Adjacent Channel Interference and Congestion Control for Multi-Channel Operation in Vehicular Networks

Esteban Egea-Lopez, Pablo Pavon-Mariño

***Abstract*—ETSI has considered a new set of services for the Release 2, which cannot be implemented using the single control channel. Therefore, it is necessary to regulate the operation of applications on multiple channels, what is called Multi-Channel Operation (MCO). In this case the interference from the first adjacent channel is not negligible and is recommended to manage it by congestion control mechanisms. We propose an elastic channel usage scheme for MCO, based on an optimal scheduler whose goal is to maximize the traffic on the channels while minimizing the adjacent channel interference generated by that load and controlling the congestion. We formulate the interference and congestion control as a constrained convex optimization problem and derive as solution a distributed algorithm called MINOS (MultI-chaNnel operation Optimal Scheduler). MINOS seamlessly work when vehicles have a different number of network interfaces available, and constraints and priorities can be set individually and dynamically, which provides flexibility to implement more sophisticated services on top of the framework. Our results show that MINOS effectively controls the congestion and reduces the interference, achieving an increased packet reception ratio across all channels and a higher allocated traffic, compared to other proposals.**

***Index Terms*—Vehicular Communications, congestion control, scheduling, Multi-Channel Operation, MCO, adjacent interference.**

I. Introduction

COOPERATIVE Intelligent Transportation Systems (C-ITS) based on connected vehicles and road infrastructure will support a number of road safety, road efficiency, automated traffic control, cooperative driving and other advanced services. The ETSI ITS G5 European standard for vehicular communications uses IEEE 802.11p as physical and medium access control layers [1], [2]. The proposed spectrum allocation in Europe included 5 safety-related 10 MHz channels between 5875 and 5925 MHz and two additional non-safety channels between 5855 and 5875 MHz. For the Release 1, a basic set of safety applications was identified and defined in ETSI TR 102 638 [3], which were built on the Cooperative Awareness Service (CAS) [4] and the Decentralized Environmental Notification (DEN) service [5]. Both services use a single channel at 5895 MHz (control channel) and tests have shown that it is sufficient to support the use cases of the Release 1.

However, since 2017, a new set of services has been considered for the Release 2, which includes Connected and Cooperative Automated Mobility (CCAM), Collective Perception (CP) or Vulnerable Road Users (VRU), as described in [6]. Those services cannot be implemented using the single control channel [7]. Therefore, it is necessary to regulate the operation of applications on multiple channels, what is called Multi-Channel Operation (MCO) in C-ITS, including potential operation on heterogeneous radio access technologies (RAT). To this purpose, several studies have analyzed the functional requirements and technical limitations of such a system. In particular, ETSI TR 103 439 [7] evaluates and derive recommendations for MCO on a single RAT, while [6] analyzes the business, functional and safety requirements of several Release 2 applications and [8] proposes a MCO concept based on that analysis.

Simulations done in ETSI TR 103 439 [7] consider a scenario where C-ITS stations are equipped with multiple radio access interfaces operating with ITS G5 at different channels and evaluate mainly the impact of interference on adjacent channels. The results and recommendations from [7] and [8] show that distributing the load over multiple channels improve the performance of the system. They also show that, while the interference from the second and further adjacent channels is negligible, the interference from the first adjacent channel is not, and needs to be managed. Finally, they recommend to manage that interference by congestion control mechanisms instead of adjusting the transmit power, that is, by setting the appropriate load and maximum channel load on the different channels [7], [8]. But specific mechanisms for interference management are left as future research and no particular one is recommended, although they review three different potential approaches for *channel usage*: with *sequential filling* traffic load is assigned to channels in a predefined order, filling it up to a given level before passing to the next channel; with *load balancing* the traffic is distributed among the channels uniformly, for instance, with a round-robin cycle; and finally *elastic* would assign the load according to some possibly dynamic criteria. In addition to the load assignment, the *channel association* criteria would determine how the applications associates their messages to channels. Two general association policies are suggested: *predefined* and *flexible* which can be combined with the aforementioned channel usage mechanisms.

Therefore, in this paper, we address how to manage MCO adjacent channel interference for C-ITS based on those recommendations. In particular, we propose an elastic channel usage scheme based on an optimal traffic scheduler whose goal is to maximize the traffic load on the channels while minimizing

E. Egea-Lopez and P. Pavon-Mariño are with the Department of Information and Communications Technologies, Universidad Politécnica de Cartagena (UPCT), Spain, e-mail: {esteban.egea, pablo.pavon}@upct.es

the adjacent channel interference generated by that load and controlling the congestion, called MINOS (MultI-chaNnel operation Optimal Scheduler). To derive the algorithm we formulate the management of interference as a network utility maximization problem (NUM), that is, a constrained convex optimization problem where a utility function is associated to each vehicle and whose goal is to maximize its sum. By using as utility (α,w)-fairness functions, priorities as well as different types of fairness [9] can be induced on the solution. Congestion is handled by the problem constraints which ensure that the channel load on each channel will not exceed a predefined *maximum channel load, (MCL)* and that each vehicle is assigned at least a given minimum traffic, which can be set to specific channels. We solve the problem with a dual approach to derive a distributed algorithm for the MCO interference management. Even though interference depends on the transmit power, ETSI TR 103 439 [7] recommends not to use transmit power to control interference, so we assume that vehicles use a fixed transmit power and only control the allowed traffic on a channel. Our mechanism thus fulfills two goals: first, it effectively controls congestion over all channels by limiting the traffic that applications can send to a channel. The purpose of setting a MCL per channel is to keep some fraction of the capacity always available for high priority emergency messages, typically DEN packets, or other dynamically triggered messages. So, it is similar to the congestion control mechanisms for Release 1 single channel applications [10], [11]. Second, it controls the adjacent channel interference.

Let us note that we do not propose any channel association criteria but the association can be assigned by a related or independent mechanism. As mentioned before, the association policy and channel usage mechanism can be combined. For instance, a combination called *sequential filling with predefined association* is suggested in [7]: in that case, both a certain sequence of channels and a mapping of applications to channels is given. Our proposal does not prescribe any mapping but is very flexible for implementing different association schemes and assigning priorities while maximizing its utility and complying with the constraints. We do not address either how the C-ITS applications inform other stations where to listen for specific services. In both cases, it is assumed that there is a negotiation mechanism [12] used to decide and coordinate the channel for a service. MINOS is flexible enough to accommodate those mechanisms: by setting their traffic demand, applications can ensure that a minimum message rate, possibly previously negotiated with a service provider [12], is guaranteed on certain channels.

In summary, the contributions of this paper are:

- We propose a novel congestion control mechanism for MCO in C-ITS that maximizes the traffic load on the channels while minimizing the adjacent channel interference generated by that load.
- We formulate the interference and congestion control as a constrained convex optimization problem and derive its solution as a distributed algorithm called MINOS.
- Our results show that MINOS effectively controls the congestion and reduces the interference compared to the main ETSI TR 103 439 proposals. We also show how it seamlessly work when vehicles have a different number of network interfaces available.
- Constraints and priorities can be set by the vehicles individually and dynamically, which allows MINOS to be used as a lower-level congestion and interference control framework for more sophisticated MCO algorithms.

In the remainder of this paper we first review related works in section II. In section III, we formulate the optimization problem. First, we discuss a simplified but accurate enough interference model, then provide a formal description of the problem and propose a particular algorithm to solve it in a distributed manner. In section IV, it is validated and compared with other proposals in static scenarios. We extend afterwards the comparison and evaluation to a dynamic scenario and show how it can be used with different number of interfaces and assign different priorities. Finally, conclusions and future work are discussed in section V.

II. Related Work

Few proposals have specifically addressed the issues of the ETSI MCO concept so far. For a general overview, the operation and research challenges for the MCO in C-ITS have been recently discussed in [13], where authors highlight the need for MCO mechanisms to consider the different needs of applications and number of radio interfaces of vehicles. Indeed, our proposal seamlessly accommodate any different number of interfaces per vehicle, as we show in our results, and provides a flexible framework to set the demands of applications on different channels. It also addresses two of identified challenges of MCO in [13]: the need for integrated congestion control and interference management. MCO is considered by [14] but as part of the evaluation of Maneuver Coordination functionality, and only by allocating the different types of messages over two channels, that is, no specific MCO mechanism is proposed. The work in [15] is the only one that specifically addresses the ETSI MCO concept. Authors propose an extension of the Release 1 reactive DCC mechanism [16] to decide the channel assigned to messages, that is, depending on congestion of the channel measured by the *Channel Busy Time (CBT)*. However, their evaluation is quite limited and only shows that the CBT is distributed over the three channels used when the DCC is enabled. Moreover, it shows how reactive mechanisms keep the CBT oscillating around the desired threshold, achieving only the desired value on average but with large excursions. On the contrary, our proposal keeps the load under the MCL for all the channels, with minor deviations.

There are other works that have addressed multichannel operation although not specifically the ETSI MCO concept. SAMCO [12] provides a heuristic for channel switching based on service prioritization and channel load. It assumes that vehicles use two network interfaces and need to change the channel on a given network interface to send certain messages, so one of its goals is to minimize the frequency of channel switching. Load is controlled by disabling the transmission of low priority messages when the CBT is above a certain threshold. Therefore, it provides a mainly a channel association

mechanism with channel congestion capabilities. Our work, however, assumes the ETSI MCO concept of using several interfaces operating at different channels simultaneously and controls the congestion by regulating the traffic that can be sent over a channel. Note that the service provisioning mechanism described by SAMCO could be used on top of our algorithm, by using the service information to set the limits and priority parameters. CarHet [17] proposes an heuristic to select the interface to use where vehicles can use multiple Radio Access Technologies (RAT), such as ITS G5 and cellular. Vehicles select an interface according to the requirements of the applications and the load that it would add to a given interface. By minimizing the load added on a given interface, the algorithm balances the load over all interfaces, in manner similar to the *load balancing* channel usage mechanism suggested in [7], but in a multi-RAT scenario. Unlike our work, the potential adjacent interference between RATs is not considered.

Optimal resource allocation mechanisms have been studied extensively in the context of wireless networks [18]–[21] and vehicular networks in particular [22], [23]. For vehicular networks, Xu et al. [22] provide a recent survey, which shows the wide variety of techniques and approaches used depending on the specific problem. Our proposal here follows the formal approach and methodology we have used for congestion control in previous works [10], [11]: we formulate the problem as a utility maximization one and use a dual approach to find a distributed algorithm. But unlike our previous works here formulate a multichannel optimization problem and address the adjacent interference problem specific to the MCO concept by proposing a simplified but accurate enough interference model.

III. Multi-Channel Operation interference control in vehicular networks

Notation: In the remaining of the paper we use boldface letters ($\mathbf{x}$) for random variables, lowercase letters with bars ($\bar{x}$) for vectors, uppercase letters with bar ($\bar{H}$) for matrices and subscripts for the elements of a vector or matrix. A summary of variables used is provided Table in I.

In this section we model the congestion and interference control problem as a convex optimization problem. We first discuss a simplified interference model. Afterwards we formally formulate the problem and propose a distributed algorithm to solve it.

Let $\bar{v} = [v_1, .., v_V]^T$ be a set of vehicles in a vehicular network, with V the number of vehicles. Vehicles are equipped with C radio access interfaces operating with ITS G5 at different channels, so that $\bar{c} = [c_1, .., c_C]^T$ are the available channels. Let us assume for the moment that all messages have the same fixed size, b_m bytes, for the analysis of interference. This assumption has no influence later since we used normalized loads for the optimization. Moreover, in Sect. IV we show how the algorithm can be used with messages of different sizes.

In a given period T_m s, each vehicle $v \in \bar{v}$ can transmit, r_{cv} messages over channel $c \in \bar{c}$. We group the traffic variables in a matrix $\bar{R} = \{r_{cv}\}$ with dimensions $C \times V$. The goal of our algorithm is to assign the r_{cv} that maximize some utility function.

TABLE I
Variables used in the paper.

Variable	Meaning
C	Channels
c_k	Channel k
C_{cv}^{max}	Maximum channel load for vehicle v at channel c
d_{cv}	Minimum demand for vehicle v at channel c
$\bar{D}$	Traffic demand matrix
ϵ	Inner problem learning rate
γ	Outer problem learning rate
λ_{cv}	Lagrange multiplier for vehicle v at channel c
M	Number of available slots in the channel
$\bar{N}$	Neighbor matrix
r_{cv}	Packet rate for vehicle v at channel k
$\bar{R}$	Traffic matrix
t_{cv}	Log-packet rate
T_m	Measurement period
$\bar{T}$	Load matrix
v_i	Vehicle i
V	Vehicles
w_{cv}	Priority weight

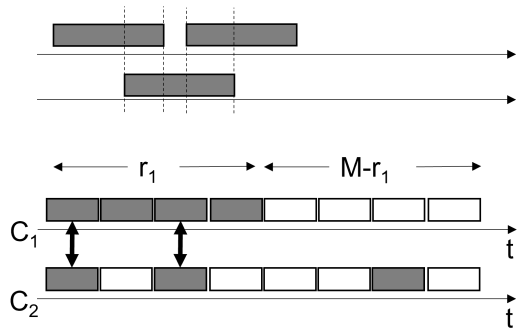


Fig. 1. *Top*: General interference model. *Bottom*: Slotted interference model between channels C_1 and C_2, when $M = 8$, $r_1 = 4$, $r_2 = 3$ and $i = 2$. Slots occupied with packets are shown as solid rectangles. Slots with adjacent interference are shown with a thick arrow.

A. Interference model

To model interference between adjacent channels it is necessary to compute the overlapping of packets sent on them (Fig. 1). But computing the overlap for a general model of traffic generation is a difficult problem, further complicated by the use of a 802.11 MAC. Instead, we will considered a simplified model where we assume that time is slotted and transmission of packets can only occur at the beginning of a timeslot. The intuitive idea behind this model is that the MAC tends to organize transmissions in this way for medium to high loads on the channel, which is the regime we are interested in, even though the slots at each channel will be shifted. For the simplified model, all channels share a common origin of time and the timeslot duration coincide with the packet duration. With this model, adjacent channel interference occurs when two packets are transmitted in the same timeslot in adjacent channels, see Fig. 1. Consider first the distribution of the number of coincident slots, $P_r(occupancy = i)$, that is, that i slots of channel 2 are occupied and the same slots of

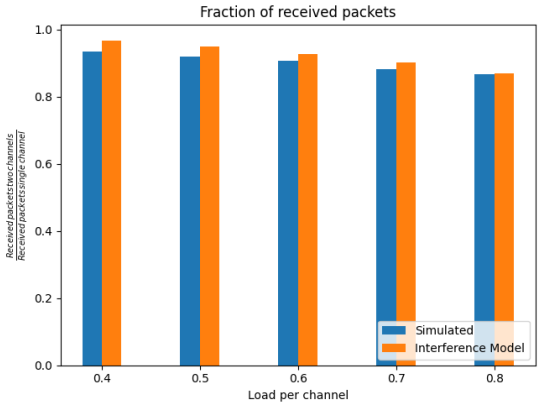


Fig. 2. Ratio of average received packets per channel when using two adjacent channels to packets received on a single channel. Comparison of the simulated and interference model expected fraction.

channel 1 are also occupied. For a given period of time, F_M, we have a fixed number of timeslots per channel, M. We can further assume that the packets are generated uniformly over the period F_M, that is, r slots out M are selected uniformly, so that each slot has 0 or 1 packets only. This corresponds to a Fermi-Dirac occupancy distribution (just one packet can occupy a slot) [24, Sect. 5.2] where each arrangement of the M slots has probability $\binom{M}{r}^{-1}$, and the probability that i packets are transmitted in the same slot in adjacent channels obeys a hypergeometric distribution. To see this, consider two adjacent channels, c_1 and c_2, where a total of r_1 and r_2 packets are transmitted respectively. If only i slots out of r_2 overlap with others in the adjacent channel, then the remaining $r_2 - i$ slots have to be distributed in the $M - r_1$ positions not occupied by packets in the adjacent channel (see Fig. 1 bottom for a particular example). The probability of this arrangement is $\frac{\binom{M-r_1}{r_2-i}}{\binom{M}{r_2}}$ and we have to add this probability for all the possible ways that the i interfering slots may be distributed over the r_1 ones, so that the final probability is $P_r(occupancy = i) = \binom{r_1}{i}\frac{\binom{M-r_1}{r_2-i}}{\binom{M}{r_2}}$, which is a hypergeometric distribution. But the probability that there is actually interference in the coincident slots depends on the Signal-to-Interference-plus-Noise Ratio (SINR) of those packets at the receiver, which in turn depends on factors such as the transmit power, distance, propagation model and the adjacent channel rejection of the receiver. That is, the probability of *not* receiving i packets in c_2, conditioned on those slots being occupied in c_1 is, $P_r(fail = i|occupancy = i)$, which is a function of those factors. Finally, the probability of interference is given by $P_r(interference = i) = P_r(occupancy = i)P_r(fail = i|occupancy = i)$. From this, taking into account that both probabilities are independent, the average interference (average number of packets lost) from channel c_k to channel c_l, from now on called I_{kl}, is

$$E[I_{kl}] = E[I_{kl}] = p\frac{r_k r_l}{M}. \quad (1)$$

where p is the expected value of the conditioned probability $P_r(fail|occupancy)$.

The expression of the latter expectation may be complicated, depending on the propagation model, but it may be approximated. We will consider it as a parameter and leave a more detailed study for future work. The reason is that, even though this is a crude model, it captures in a reasonable way the fact that adjacent channel interference depends on the load of adjacent channels. To confirm this, we have simulated, for a path loss model, a single channel first and then two adjacent channels with the simulator of Sect. IV and compared it with the expected model for different offered loads. The results are shown in Fig. 2, where we plot the ratio of received packets per channel of the two-channel and single-channel scenarios. This fraction, lower than 1, shows the expected packet losses due to interference. As can be seen, the interference model agrees well with the simulation results and improves as the load increases as expected. So in this paper we will not attempt to further model p, since for our current purpose it is enough. For a more advanced algorithm which intended to jointly control the load and the transmit power, for instance, to maximize the number of received packets, it would be necessary to fully model it. But this is left as future work.

B. Problem formulation

Once we have described our interference model, we can formulate the packet scheduler as a convex optimization problem. Our goal is to maximize the number of packets transmitted on each channel, while minimizing the adjacent channel interference. Notice that these two goals are antithetical, since the interference increases with the number of packets transmitted. Therefore, we have to add the possibility to balance and tune them. In addition, there are several reasons to limit the maximum load allowed on a given channel, MCL, usually to keep some fraction of the capacity always available for high priority emergency messages, but also to support the transmission of messages not anticipated due to unexpected vehicle or environment dynamics. We consider thus that there is a MCL for each channel and vehicle, C_{cv}^{max} messages/s, despite, in practice, all vehicles typically use the same MCL per channel. The MCL is often expressed as a limit to the CBT, which is the fraction of time that the channel is not idle. We will express it mostly as a CBT fraction in this paper. We also take into account that each vehicle has a minimum traffic demand per channel d_{cv} that the scheduler has to ensure. That traffic demand is typically generated by the applications, for instance with periodic awareness messages, which inform the scheduler of their expected traffic needs for the next period. Therefore, we group them in a *traffic demand* matrix $\bar{D}$.

Now, let us first consider a deterministic propagation model, such as a path loss one. We then can define a symmetric *neighbor matrix*, $\bar{N}$ with dimensions $V \times V$, where N_{ij} is 1 if vehicle i is in range of vehicle j and 0 otherwise. We set $N_{ii} = 1$, that is, consider that a vehicle is a neighbor of itself. Finally, abusing of notation, we define $N(i)$ as the set of neighbors of vehicle i, that is, $N(i) = \{j \in \bar{v}|N_{ij} = 1\}$. This way we can define the *load matrix*, $\bar{T}$ with dimensions $C \times V$, as the product of the traffic and neighbor matrices, $\bar{T} = \bar{R}\bar{N}$, so that T_{cv} is the load that vehicle v measures on channel c, including its own traffic.

We will use α,w-fair utility functions U_v to control the fairness and priority of the optimal traffic allocation,

$$U_v(x) = \begin{cases} w_v x & \text{if } \alpha = 0 \\ w_v \log x & \text{if } \alpha = 1 \\ w_v \frac{x^{1-\alpha}}{1-\alpha} & \text{if } \alpha > 0, \alpha \neq 1 \end{cases} \quad (2)$$

The α parameter induces a given fairness notion [9], while the w_v parameters can be used to achieve weighted fairness and assign priority to channels. For simplicity, we next only consider proportional fairness and restrict ourselves to the case $\alpha = 1$.

With all the previous definitions, we can formulate the optimization problem. First, to obtain a convex problem, we do the change of variable: $t_{cv} = log(r_{cv})$, that is, we consider the traffic in log-scale.We also assume that all the variables and parameters are normalized to the maximum channel capacity, so that $r_{cv} \in [0, 1]$. Let us note that after the change of variable, the load matrix has also changed, since now the elements of $\bar{R}$ are $R_{cv} = e^{t_{cv}}$ and so the elements of $\bar{T}$ are sum of exponentials of t_{cv}. Now, we can formulate the problem as follows:

$$\mathbf{E-CI}: \quad \max_{t_{cv}} \sum_v \left\{ \sum_c \left[\omega_{cv} log(e^{t_{cv}}) \right] - 2p \sum_{c=1}^{C-1} \left[T_{c,v} T_{c+1,v} \right] \right\} \quad (3a)$$

subject to:

$$T_{cv} \leq C_{cv}^{max} \quad \forall c \in \bar{c}, \forall v \in \bar{v}, \quad (3b)$$

$$t_{cv} \geq log(d_{cv}) \quad \forall c \in \bar{c}, \forall v \in \bar{v}, \quad (3c)$$

The objective function (3a) includes our two goals: on the one hand, the first term maximizes the assigned load to a channel, using the α-fair utility function to induce proportional fairness and assign priorities. On the other hand, the second term minimizes the average adjacent interference. The average interference is given by eq. (1). Since we only consider the interference of the first adjacent channel, as recommended by [7], for a vehicle v, the average interference on a channel c by their adjacent channels is given by eq. (1), that is, using the load matrix we have $pT_{c-1,v}T_{c,v} + pT_{c+1,v}T_{c,v} = pT_{c,v}(T_{c-1,v} + T_{c+1,v})$, except for the edge channels that are only mutually interfered by the following or previous channel respectively. Notice that, since we used normalized traffic, the interference is also normalized in the previous expression. By adding the previous expression for all the channels, we get the $-2p\sum_{c=1}^{C-1}[T_{c,v}T_{c+1,v}]$ term in the objective function. As discussed previously, the additional parameter p reflects that, due to adjacent channel rejection, not all packets transmitted in adjacent channels interfere with each other. Though it can also be simply treated as a weight to balance the importance of the interference term with respect to the load term.

Constraints (3b) are used for congestion control, that is, to ensure that the load measured on each channel by a vehicle is below the MCL of the channel, and (3c) ensure that the assigned traffic covers the minimum demand and the traffic is positive. The values of these constraints can be set by each vehicle dynamically and independently if necessary, and may be used as part of a higher level context dependent mechanism or to fulfill some quality of service requirements.

C. Dual decomposition and decentralized algorithm

We proceed to derive a distributed algorithm to solve problem $\mathbf{E-CI}$ (3) by taking its dual [18]. Previously, to use the standard formulation in optimization, we convert it to a minimization problem by changing the sign of sum in (3a). Then we relax the constraints (3b) to construct the Lagrangian function L of (3a):

$$L(\lambda_{cv}, t_{cv}) = \sum_v \left[-\sum_c \omega_{cv} t_{cv} + 2p \sum_{c=1}^{C-1} \left[T_{c,v} T_{c+1,v} \right] \right] + \quad (4)$$

$$+ \sum_{cv} \lambda_{cv} (T_{cv} - C_{cv}^{max}) \quad (5)$$

where the $\lambda_{cv} \geq 0$ are the Lagrange multipliers associated with the relaxed constraints. The Lagrange dual is the minimum value of the Lagrangian function over the domain of the variables. That is, given a set of non-negative multipliers λ_{cv},

$$g(\lambda_{cv}) = \min_{t_{cv} \geq log(d_{cv})} L(\lambda_{cv}, t_{cv}) \quad (6)$$

And the dual problem is:

$$\mathbf{D}: \max_{\lambda_{cv} \geq 0} g(\lambda_{cv}) = \max_{\lambda_{cv} \geq 0} \left\{ \min_{t_{cv} \geq log(d_{cv})} L(\lambda_{cv}, t_{cv}) \right\} \quad (7)$$

The objective function in (3a) has the *strong duality property* since is concave and the Slater condition holds, and the Karush-Kuhn-Tucker (KKT) conditions characterize its optimum solution [25]. Then, when the optimal multipliers λ_{cv}^* have been found, at least one of the associated rates $\bar{t}_{cv}^*(\lambda_{cv}^*)$ are the optimal solution of the original problem (3).

In order to solve the problem with the dual method we use a projected gradient descent algorithm to iteratively find the dual optimal multipliers λ_{cv}^*, and obtain the corresponding optimal traffic $t_{cv}^*(\lambda_{cv}^*)$ allocations from eq. (6): at each iteration, vehicles broadcast their multipliers λ_{cv} and measured loads T_{cv} per channel, receive multipliers and loads from one-hop neighbors and use them as input to the minimization problem (6). The latter has no analytical solution, but can be solved independently by each vehicle, using local information, either with a solver or with a gradient algorithm.

Our algorithm MINOS follows the above procedure as sketched in Algorithm 1. Algorithm 2 is used to solve the local problem (6), with just a standard minimization via gradient descent. In both algorithms the gradient descent is projected on the constrain set, but notice that, since that set is made of box-constraints, that projection is immediate and equivalent to the element-wise set of the minimum values when they are below the constraint.

We end with a clarification the algorithm steps and other practical aspects:

Algorithm 1 - MultI-chaNnel operation Optimal Scheduler (MINOS).

At $k = 0$, set initial vehicle multipliers λ_{cv}^0, logarithmic rate t_{cv}^0, $\forall c, v$

Then, at each time k:

Step 1. Each vehicle i receives from its neighbor vehicles their multipliers λ_{cv}^k and measured loads T_{cv}^k, $\forall c, v \in N(i)$.

Step 2. Each vehicle updates its own multiplier λ_{cv}^{k+1} as: $\lambda_{cv}^{k+1} = \left[\lambda_{cv}^k + \gamma\left(T_{cv} - C_{cv}^{max}\right)\right]^+_{\lambda_{cv} \geq 0}$

Step 3. Each vehicle updates t_{cv}^{k+1} with the output of Algorithm (2):

$t_{cv}^{k+1} = LagrangianMinimization(\lambda_{cv}^k, T_{cv}^k, t_{cv}^k)$

Algorithm 2 - LagrangianMinimization

procedure LAGRANGIANMINIMIZATION(λ_{cv}, T_{cv}, t_{cv})
  $t_{cv}^1 = t_{cv}$, $T_{cv}^1 = T_{cv}$, $j = 1$
  repeat
    $t_{cv}^{j+1} = \left[t_{cv}^j - \epsilon \frac{\partial L(\lambda_{cv}^j, t_{cv}^j)}{\partial y}\right]^+_{t_{cv} \geq log(d_{cv})}$
    $j = j + 1$
  until $||\nabla L|| = 0$
  return t_{cv}^{j+1}
end procedure

- To update the multiplier in Step 2 of Algorithm 1, we use the fact [26, T. 17.4] that a subgradient of the dual function g is given by

$$\nabla g(\lambda_{cv}) = T_{cv} - C_{cv}^{max}, \quad \forall c, v \tag{8}$$

  The subgradient is just the difference between the measured load by a vehicle and the MCL constraint per channel. Notice that, since we use normalized loads, the load (T_{cv}) is measured using directly the CBT on the channel per period.
- The gradient of the Langrangian function in Algorithm 2 is given by:

$$\frac{\partial L(\lambda_{cv}, t_{cv})}{\partial t_{ij}} = -w_{ij} + \\ + e^{t_{ij}} \sum_{v' \in N(j)} \left[\lambda_{i,v'} + 2p(T_{i-1,v'} + T_{i+1,v'})\right] \tag{9}$$

  where again, the channel loads received from the neighbors are measured using the CBT.
- The use of the CBT to measure the load is key for the algorithm to adapt to different fading conditions as shown in the Results.
- Similarly, although we formally use the neighbors of a vehicle in the formulation, in practice, a vehicle just receive the loads and multipliers from the neighbors that are effectively in range, depending on the fading conditions, and thus adapt to them. That is, it is not necessary to know the neighbors of a vehicle to apply the algorithm. Moreover, let us remark that all the required information is local, that is, no two-hop information is needed.
- MINOS requires broadcasting the multipliers and loads of every channel, two real numbers per channel. Assuming it each one is coded in 4 bytes, it makes 8 bytes per channel, that is, it typically requires a maximum of 64 bytes per vehicle. But it is not necessary to continuously broadcast the information, since the convergence is usually reached in a few iterations, and the conditions are not expected to change rapidly, as we further discuss in Sect. IV-F.
- From a practical perspective, p could be periodically estimated by measurements by vehicles. As said, it can also be treated as a weight for the importance given to the adjacent interference, that is, by letting it go to zero we can switch off the interference part of the algorithm and just maximize the allocated load subject to the MCL and minimum traffic constraints.

A proof of the convergence of MINOS is given in the Appendix.

IV. RESULTS

The performance of MINOS is evaluated in this section. We first validate the algorithm in a static scenario, where vehicles do not move, and then we test it in a dynamic scenario. Since, to the best of our knowledge, there is no other proposal that addresses MCO scheduling specifically at the moment, we compare it with the channel usage mechanisms proposed in ETSI TR 103 439 [7]: *Load Balancing* (LB) and *Sequential Filling* (SF). LB assigns each packet to a channel in a round-robin fashion, while SF keeps using a channel until is is completely filled before sending on another channel. It does not use consecutive channels, but a fixed sequence described below. The simulations have been implemented with the OMNET++ 6.0.3 simulator [27], and the INET 4.5 library [28] which implements the IEEE 802.11p standard. This library also implements a realistic propagation and interference model for computing the Signal to Interference-plus-Noise Ratio (SINR) and determining the packet reception probabilities, considering also capture effect. We have extended the library with an implementation of the MCO. Our implementation considers adjacent channel interference by adding the interference of the first and second adjacent channels to the SINR of each reception. We set a data rate of 6 Mbps and assume an equal beacon size of 500 bytes. This results in a total message size of 537 bytes, including MAC and physical headers. Because the physical-layer trailer duration depends on the payload size via the padding, the resulting packet duration is 752 μs, according to [1]. To get the total channel capacity we add to that duration the IFS (58μs) and the average contention window duration (195 μs) [2], which results in a total channel capacity of around 995 messages per second.

The simulation parameters that are common to the simulations in the following sections are summarized in Table II, where we have adopted most of the parameters used in the ETSI TR 103 439 [7] simulations. The logical numbering of channels by ETSI standards does not correlate with the physical numbering of the channels by the IEEE 802.11p standard. The ETSI ITS standards consider 8 physical channels separated by 10 MHz, starting at 5.855 GHz. The first channel

TABLE II
COMMON PARAMETERS FOR SIMULATIONS

Parameter	Value
Data rate	6 Mbps
Transmit power	23 dBm
Sensitivity (S)	-98 dBm
Noise	-110 dBm
SINR threshold	3.1 dB
Measurement period (T_m)	250 ms
Message size (b_s)	500 bytes
Message duration (T_b)	752 μs
Minimum demand (d_{cv})	16 packets/s
w_{cv}	1
λ_v^0	0.1
γ	0.1
ϵ	10^{-6}

is numbered 172 and the last one 186 by IEEE standards, while the ETSI standard names them SCH0 to SCH7, with SCH0 corresponding to channel 180. To avoid confusion, in the results and figures of this paper we will refer to the channels by their physical ordering, starting by channel 0, which corresponds to channel 172 in the ETSI standard. We will use at most 6 channels, from 0 to 5, which are the ones that are used by the ETSI TR 103 439 [7]. That is, we will not use logical channels SCH6 and SCH7 which correspond to physical channels 184 and 186. As an example, with our terminology, SF uses the channels in the order 4 2 3 5 0 1, corresponding to physical channels 180, 176, 178, 182, 172 and 174, respectively. In all the scenarios, we simulate 6 applications on top of the MCO which generate traffic with exponentially distributed inter-packet generation times with a given rate. This way we do not restrict ourselves to periodic traffic. The packets are then distributed among the channels according to the MCO allocation. The interference model decreases the packet received power on an adjacent channel by 25.9 and 40.6 dB for the first and second adjacent channels, respectively, as in [7]. All simulations have been replicated 5 times and 95% confidence intervals are shown as shaded bands in the plots, though for some simulations are not visible due to their low value. To facilitate the reproducibility of our results, the implementation of the algorithms, our implementation of MCO and the rest of the code used for the paper are freely available in our public repository[1], where any other parameter value can be checked.

A. Static scenarios

We first test the validity of our algorithm and assumptions in static scenarios where vehicles do not move which allows us an accurate control of the vehicle interactions. Since the use of 6 channels adds a considerable computational burden to the simulations we just use 20 vehicles. We start with a scenario where the vehicles 15 m apart and use a path loss model with a path loss exponent of 2.5. In this scenario all the vehicles are in range and close to each other which allows to highlight the effects of interference. That is, adjacent channel interference mainly provokes packets losses by reducing the SINR below

[1]https://gitlab.com/esteban.egea/minos

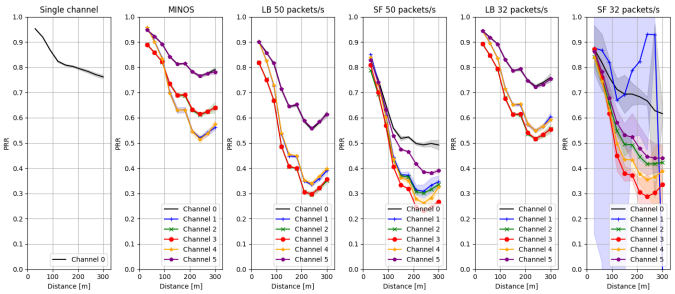


Fig. 3. All vehicles in range. Average Packet Reception Ratio (PRR) versus distance. Shaded bands show 95% confidence intervals.

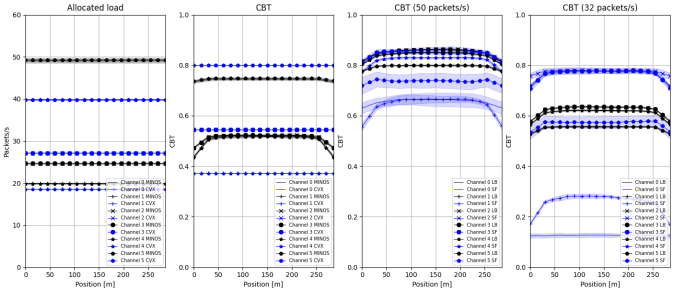


Fig. 4. All vehicles in range. First: Allocated load by MINOS and exact optimal values by CVXPY versus vehicle position. Second: Average CBT versus vehicle position for MINOS and CVXPY optimal. Third and Fourth: Average CBT versus vehicle position for LB and SF with 50 and 32 packets/s respectively. Shaded bands show 95% confidence intervals.

the threshold and increases the energy on the adjacent channels making them appear as busy. Since it is difficult to measure it directly, we use the Packet Reception Rate (PRR) as a proxy of the interference. We use the definition of PRR from [7] as the ratio of the number of vehicles receiving a packet to the number of vehicles located at a certain distance interval. To illustrate it, we simulate a single channel as a baseline. Interpacket generation time is exponentially distributed with a rate of 65 packets/s, so that the channel is saturated. With all the vehicles in range and a deterministic propagation model, packets are only lost due to collisions. In Fig. 3 left we show the PRR for the single channel case. As can be seen, the PPR decreases with the distance, since collisions affect distant vehicles more. In the right plots of the figure we show the PRR for the 6 channels case. As can be seen, the PRR decreases more with distance than in the single channel case, which is due to the effect of the adjacent channel interference in the SINR.

For MINOS, since vehicles are very close in these scenarios, we keep the p parameter fixed to 1, and $C^{max} = [0.8, 0.8, 0.8, 0.8, 0.6, 0.8]$, that is, we set the MCL to 80% to all channels except channel 4, which is set to 60%. We simulate 6 applications on top of the MCO which generates a load with a rate of 50 packets/s initially. In Fig. 4 we show the results of a simulation run for 20 s. In addition to the algorithms, the exact optimal values have been computed with the CVX numerical solver in Python, CVXPY [29]. As can be seen in the first plot, the load allocated by MINOS is similar to the optimal values computed by CVXPY. The discrepancy is due to the fact that MINOS is using the actual CBT measured

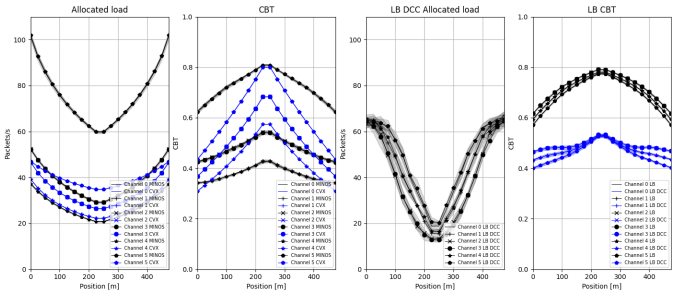


Fig. 5. Not all vehicles in range. First: Allocated load by MINOS and exact optimal values by CVXPY versus vehicle position. Second: Average CBT versus vehicle position for MINOS and CVXPY optimal. Third: Average allocated load by LB with DCC versus vehicle position. Fourth: Average CBT for LB with 65 packets/s and DCC. Shaded bands show 95% confidence intervals.

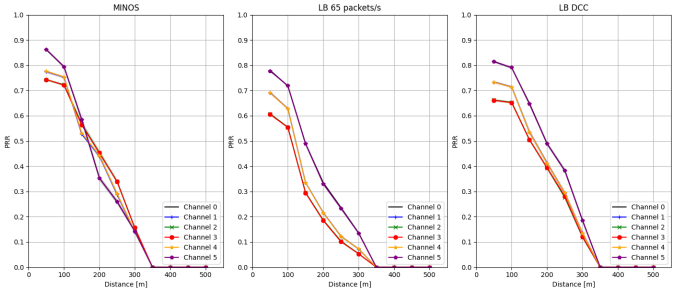


Fig. 6. Not all vehicles in range. Average Packet Reception Ration (PRR) versus distance. Shaded bands show 95% confidence intervals.

on the channel, which is lower than in the theoretical model, since the latter does not consider the effects of the IFS time in the MAC. It can be checked in the second plot on Fig. 4, as even though the allocated load for channels 0 and 5 is higher than the CVX optimal, the average CBT is below the target MCL of the channels. The convergence is quick, with the optimal values attained for MINOS after 4 to 8 iterations, which depending on the measurement period, takes around 1 to 2 s. Let us note how MINOS and CVX do not allocate all the allowed load to the channels, in order to reduce the interference. In fact, it does allocates the full available MCL to the edge channels, 0 and 5, which are less affected by interference.

LB and SF, without further congestion control mechanisms, do not consider the MCL constraint, so they do not reduce the packet rates. As can be seen in the third plot of Fig. 4, the average CBT for 50 packets/s is higher than the MCL and evenly distributed by LB while higher for the first channels in the sequence on SF and lower in the last ones which are underutilized. In Fig. 3 the PRR is lower for LB and SF than for MINOS. The effects of adjacent channel interference are clearly shown in the fact that the edge channels have consistently a higher PRR even for LB that distributes the load. Moreover, both effects of interference can be seen for channel 3 of SF, as it can be seen that the PRR on channel 3 is lower than for channels 4 and 2, and the CBT is higher, despite less load being allocated to it.

To provide a fairer comparison, we have also tested LB and SF with a lower load of 32 packets/s, which corresponds to the

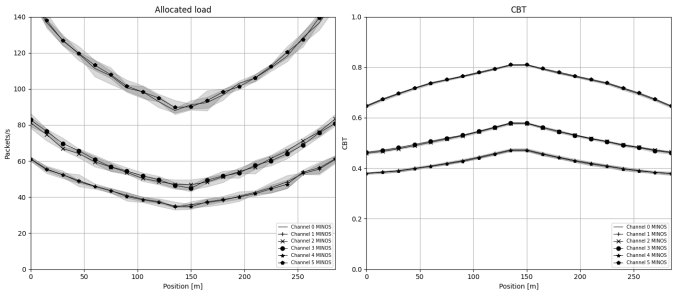


Fig. 7. Not all vehicles in range. Allocated load and CBT versus vehicle position for MINOS with different size of packets. Shaded bands show 95% confidence intervals.

average channel rate of the optimal load achieved by MINOS. As can be seen in the right plots of Fig. 3, the PRR of LB has increased, but the uneven distribution of load in channels by MINOS improves the PRR on the central channels compared to LB, that is, achieves its intended goal of reducing adjacent channel interference. SF, on the other hand, is not able to increase the PRR, since it keeps filling totally the channels in sequence, which is always highly detrimental to PRR. It also results in a high variability of the PRR, since the last channels in the sequence are rarely used. Moreover, the fact that the sequence used by SF uses alternating channels does not help to reduce the interference. On the contrary, the high load in the first two channels in the sequence (4 and 2) highly degrades the PRR in the central channel 3, even though it is less loaded. The performance of LB and MINOS is now similar but let us remark that these values of load are not known a priori. MINOS, unlike LB, converges to them.

In the second scenario, not all vehicles are in range. We place the vehicles at 25 m apart and set the path loss exponent to 3.29 so that the maximum range is 254 m. In all our tests SF performs consistently worse than LB, so we only show the results for LB in the remaining of the paper. All algorithms start with a load of 65 packets/s, but in this case, instead of artificially set another test with the optimal load for LB, we enable the DCC mechanism in the ETSI EN 302 663 standard [1], as described in [7]. In Fig. 5 again we show how MINOS achieve the optimal load values similar to those computed by CVXPY. Again, the differences come from the use of the actual CBT, since the CBT does not exceed the target MCL for any channel. Regarding LB, the results are similar to those of the previous scenario. When DCC is enabled the channel load is reduced uniformly for all channels as shown in Fig. 5 third plot, improving the PRR although still below than MINOS for the central channels. Actually, although not shown in the figures, the load and CBT keeps oscillating between high and low rates in all the channels, while MINOS keep a stable load and CBT.

To finish this section, we show that the size of the packet has little influence in the results, as previously discussed. We simulate again the second scenario, but now the applications select uniformly a packet size from the set $\{200, 300, 360, 455\}$ bytes [30] before transmitting. Applications use the average packet duration to set a packet rate from the allocated normalized load. As can be seen in Fig. 7, the number of packets/s is

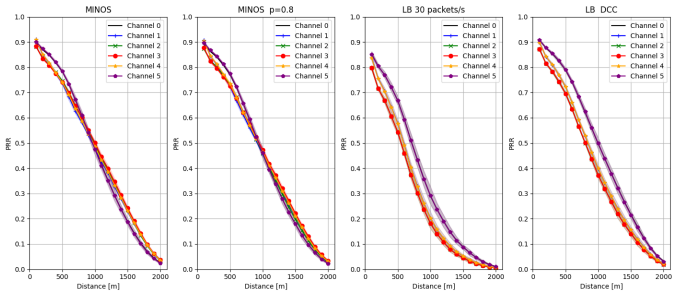


Fig. 8. Average Packet Reception Rate (PRR) for the highway scenario. Shaded bands show 95% confidence intervals.

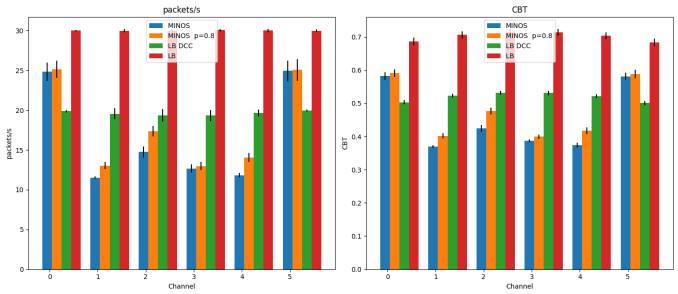


Fig. 9. Average allocated load and CBT for the highway scenario. Bars show 95% confidence intervals.

increased since the packet duration is lower, compared to the previous scenarios, but the CBT is kept below the MCL. It is possible, again, because we use the measured CBT in the computation of the subgradient, eq. (8). That is, the algorithm is not really aware of the number of neighboring vehicles, or the size of messages used, but only of the resulting load on the channel through the measured CBT. For the same reason, the size of the message does have influence on the allocated message rate but not on the channel load. To see this, consider the following example. Assume that a vehicle is sending messages with a size equal to 3 slots of our interference model. It would be equivalent to having 3 consecutive slots occupied, so it would be equivalent to a vehicle using a message rate 3 times higher than the real rate it uses. MINOS will just adapt to that measured channel load, probably allocating the vehicle a lower message rate. In this case, the real interference process deviates from the assumed model because the slots will not really be selected uniformly, but the model is just an approximation, as we have remarked.

B. Highway scenario

In this section we replicate one of the simulations done on ETSI TR 103 439 [7], a representative one, with slight modifications to reduce the simulation time. We use a 2 km straight road with 6 lanes, 3 in each direction, 4 m wide each. We place 100 vehicles on the road, that is, a relatively high density of 50 vehicles/km, evenly distributed over the lanes and uniformly distributed over the road length. Each vehicle moves at a constant speed, randomly selected using a Gaussian distribution with mean 30 m/s and standard deviation 3 m/s. As propagation model, we too use the modified ECC68 rural [7], but add a Nakagami-m fading model with $m = 3$, since the fading model is not fully specified. With this model, the average range is 250 m, so not all vehicles are in range.

We only compare MINOS with LB, for two variants of each one: LB with a fixed packet rate of 30 packets/s and with DCC enabled, and MINOS with the p parameter set to 1 and 0.8. For MINOS, we set $C^{max} = [0.6, 0.6, 0.6, 0.4, 0.6, 0.6]$, to make it comparable to LB DCC which reduces the load when the CBT exceeds 62%.

As shown shown in Fig. 8, MINOS achieves a higher PRR than LB. When the parameter p is lowered, that is, the importance of interference is lowered, it behaves slightly more aggressively, increasing the load in all the channels, which

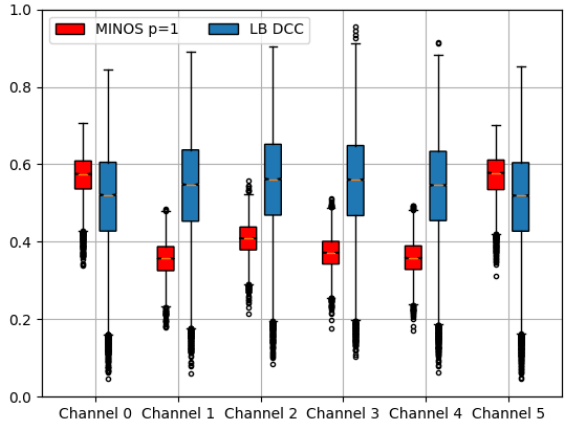


Fig. 10. Box-and-whisker plot of the CBT for the highway scenario.

results in a lower PRR. When DCC is enabled, LB increases the PRR to a level similar to MINOS, but, since the load is equally distributed, the central channels suffer the effects of adjacent channel interference, in contrast to MINOS, which keeps balanced the effect of the interference in all channels. In Fig. 9 we show the average allocated load and CBT for the different algorithms. As can be seen, LB with DCC reduces the load to an average of 20 packets/s in all channels and the CBT to around 0.5. MINOS again assigns more packets to the edge channels and less to the central channels to reduce the interference. When the p parameter is lowered, the load is increased in all channels, but in all cases the CBT is kept below the given MCL. When the DCC is enabled for LB it reduces the load in average, but since it uses a reactive mechanism, it keeps oscillating between high and low values. The box-and-whisker plot in Fig. 10 shows that effect, in contrast to MINOS.

C. Street crossing scenario

We discuss another dynamic scenario, which illustrates the behavior when channel conditions change quickly. We simulate a street crossing, where we have a cluster of 14 vehicles that move with different velocities in a straight line and cross a street where another cluster of 14 vehicles is static as shown in Fig. 11. The scenario comprises a 100 m by 100 m plane with four 40 m by 40 m buildings at the corners. The

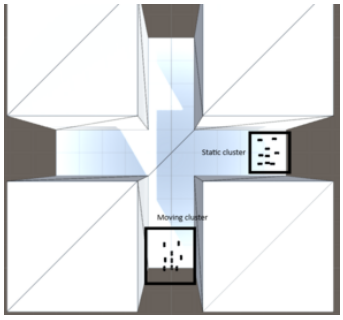


Fig. 11. Street crossing scenario. A cluster of moving vehicles move from down to up and cross a street where a cluster of static vehicles is placed.

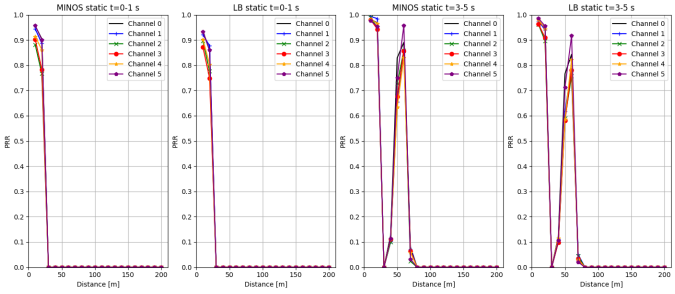


Fig. 12. Average Packet Reception Rate (PRR) for the street crossing scenario for static cluster. Figures show the PRR at times 0 to 1 s and 3 to 5 s. Shaded bands show 95% confidence intervals.

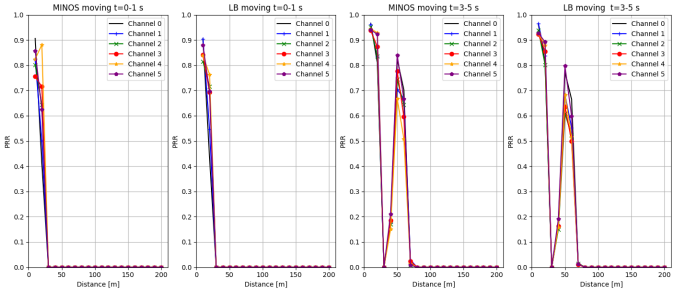


Fig. 13. Average Packet Reception Rate (PRR) for the street crossing scenario for moving cluster. Figures show the PRR at times 0 to 1 s and 3 to 5 s. Shaded bands show 95% confidence intervals.

streets are 20 m wide. We have simulated the scenario with Opal [31], a ray tracing propagation simulator, configured to use 1 reflections and 1 diffraction. With this configuration, clusters are initially only in range via diffraction and moving vehicles become in range as they approach the intersection, and then go out of range as they move away. Vehicles in the moving cluster move at a constant speed selected randomly from 10 to 15 m/s. We compare MINOS and LB with with DCC enabled using the previous section configuration.

We show in Fig. 12 and 13 the PRR for the static and moving clusters, respectively at two snapshots in time, at the beginning of the simulation and after 3 to 5 s, when the moving cluster is crossing the intersection. In the first snapshot, vehicles can only reach other vehicles in their cluster, while in the second snapshot, vehicles from the moving cluster are able each vehicles from the static cluster as shown in the second peaks for distances between 40 and 60 m. In

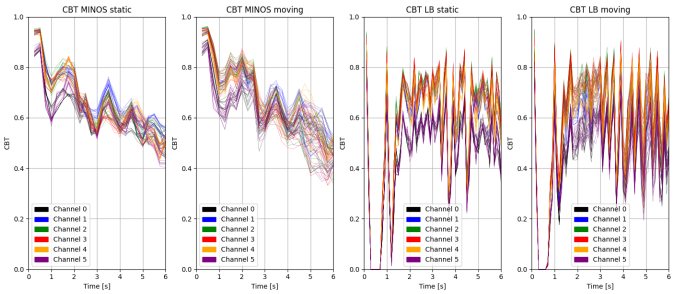


Fig. 14. Time evolution of CBT for all vehicles in the street crossing scenario.

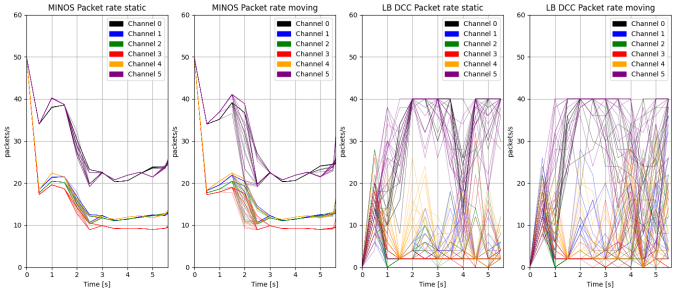


Fig. 15. Time evolution of allocated rate for all vehicles in the street crossing scenario.

the second snapshot the PRR is slightly better for MINOS in both cases. In the first snapshot the PRR is slighttly better for MINOS only in the moving cluster. But if we look at Fig. 14 and Fig. 15 where we show the time evolution of the CBT and the allocated packet rate, we see the large oscillations in the CBT and packet rate for LB. Thus, its apparently high PRR is actually an average of the periods where it is very high and very low due to the oscillations. Those oscillations are unavoidable since the DCC used by LB is a reactive procedure, which reduces the rate when the CBT is high. Since all channels are equally loaded by LB, all vehicles reach the DCC threshold around the same time.

MINOS, on the other hand, keeps a stable allocated rate and CBT. We can see how the allocated rate is quickly decreased for both clusters, which measure a similar load. There is a period of oscillation in CBT for MINOS between 1 and 2 s due to two factors: the algorithm has not reached convergence yet (only 4 iterations have been executed) and there are still packets that were enqueued at the beginning of simulation, when channel was saturated, that are being transmitted at this point. From t=1.5 s approximately, inter cluster packets start to be received, and so the CBT for MINOS increases, even though all vehicles keep decreasing their packet rates. By t=5.5 s, both clusters have converged and the MCL is not exceeded. We can see that convergence is reached after 11 iterations in this scenario even though there is a serious disturbance in the middle when both clusters are in range and the number of neighbors duplicates almost instantly.

D. Computational complexity

The computational complexity of MINOS is mainly due to the gradient descent algorithm used to solve the local problem.

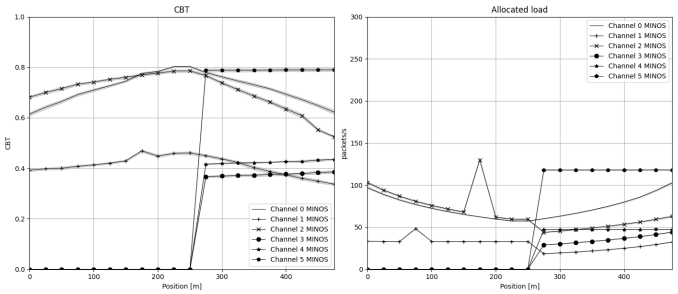


Fig. 16. Average CBT and allocated load versus position in the static scenario when half of the vehicles (from 0 to 150 m), have 3 interfaces and the rest have 6 interfaces. Shaded bands show 95% confidence intervals.

The complexity of the gradient descent algorithm is $O(Ck)$, where k is the number of iterations needed to converge and C the number of channels. In our simulations we have set a maximum number of iterations of 10^4, but on the average, the required number of iterations to solve the inner problem is an order of magnitude lower. As seen from eq. (9), the gradient of the Lagrangian function is linear in the number of neighbors, but the values of the Lagrange multipliers and the measured loads do not change during the iterations, that is, only have to be computed once before the gradient descent iterations.

E. Priorities and framework flexibility

As we have mentioned, MINOS offers a flexible framework that allows to set different constraints and parameters, as required by application. In this section we exemplify it, showing how it can be seamlessly used even if the vehicles use a different number of network interfaces and different priorities. So we come back to the second static scenario, where not all vehicles are in range. We simulate it again, but now half the vehicles, on the left part until the 10th vehicle, use only 3 network interfaces, while the rest use 6 interfaces. For the vehicles with 3 interfaces, we set the MCL to 0.8 for all the channels and $d_{cv} = 33$ packets/s, while the rest of the parameters are the same for all vehicles. In addition, we set the priority parameters $w_{13} = 2$ and $w_{27} = 2$, that is, for vehicle 3 and 7 and channels 1 and 2 respectively, we set it to 2.

Fig. 16 shows the average allocated load and CBT for all the vehicles. As can be seen, MINOS accommodates the different number of interfaces with a non-trivial allocation of load among the vehicles: Channels 4 and 5 are equally loaded for the right vehicles since they do not contribute to the load or interference of vehicles on the left, while the load is decreased in channels 1, 2 and 3 to compensate and the load is always below the MCL. The effect of priorities is also clearly shown as the assigned packet rates is approximately doubled for vehicles 3 and 7 in the corresponding channels, compared to its close neighbors. This is what is expected for proportional fairness ($\alpha = 1$) [32], where the assigned rate is proportional to the priority parameter for equal channel conditions, that is, by looking at eq. (12) in Appendix, we see that the ratio of rates assigned to close neighbors that measures approximately equal channel conditions is equal to the ratio of the priority parameters. For vehicle 7 it is clear in the figure, shown as an spike of channel 2 at the vehicle position (around 100 m). For vehicle 3, the effect is not so clear, because in this case, the minimum demand of 33 packets is enforced by MINOS for all the neighbors. But it can be seen if compared with the rate of the corresponding vehicle in the right part for which the minimum demand is not enforced. Let us also note how the effect of the increase in rate of vehicle 7 is reflected in the CBT of the adjacent channel 1, with no consequence since it is below the MCL, while it is not reflected in the CBT of channel 2 and 3, which are adjusted by the algorithm to be below MCL.

All the algorithm parameters, including constraints and priorities, can be set by the vehicles individually and dynamically, and will be incorporated by the algorithm, without the need for informing the rest of vehicles. This provides a great deal of flexibility to implement more sophisticated services on top of the framework. As an example of this, in [32] we designed an integrated safety service that uses the priority parameter to give more resources to vehicles that are closer to a dangerous situation, such as a traffic jam or an accident.

F. Practical use

As we said in the previous subsection, designing safety services and integrating with concrete channel association policies is left as future work, but we briefly discuss next how we envision the practical use of MINOS. MINOS will be possibly used in conjunction with a service association mechanism. Upon receiving Services Association Message (SAM) from a service provider, vehicles can set the traffic demand required for a given channel according to the service requirements, something that vehicles can do individually without coordinating with other service users. Once the parameters are set, a few MINOS packet exchanges (algorithm iterations) are typically enough to converge to the optimal allocation. Let us remark again here that the required information is local and it can be piggybacked on existing packets or use dedicated control packets. In any case, the overhead is remarkably low for this kind of mechanisms: only the load and Lagrange multiplier per channel. It may be compared to the overhead of similar proposals, such as CarHet [17], which requires that each vehicle repeatedly broadcast its position, the load it measures per RAT (channel in our case) and timestamp of those measurements, as well as the position of its one-hop neighbors and their channel load measurements per RAT. In our simulations, $F_m = 0.25$ s and the convergence is achieved in 4 to 8 iterations, that is, 1 to 2 s. This procedure can be repeated periodically, let us say every 20 s, or when the conditions change appreciably, as detected by the vehicles. Once the optimal allocation is achieved, vehicles may send the result back to the services provider as feedback, so that it can adjust the quality of service to the situation. For instance, assume that the service provider is a platoon leader computing the capacity requirements of a coordinated maneuver [14], it may have set a certain minimum demand for a channel, but if the allocated load by MINOS exceeds adequately that initial demand, it may increase the frequency of coordination messages knowing that there is margin.

In summary, channel association can be integrated by using three steps: first, the service provider sets the minimum demand for each channel according to the needs of the service or traffic type. For instance, it may decide that all the periodic beaconing traffic must be transmitted on a certain channel. Second, the vehicles use MINOS to compute the optimal load for each channel according to the constraints. Third, the service provider actually assigns the traffic to the channels according to the admissible load allocated by MINOS. At this point, it may decide, for example, to assign all the Collective Perception Messages (CPM) to the channel with the highest allocated load. Let us remark that MINOS will converge to a solution where some extra traffic may be potentially allocated by the applications, but they may choose not to do so and stay at a low load regime.

V. Conclusion

The operation of applications on multiple channels in ITS G5 vehicular networks requires to manage the adjacent channel interference by congestion control mechanisms. In this paper, we propose an elastic channel usage scheme for MCO, based on an optimal scheduler whose goal is to maximize the traffic on the channels while minimizing the adjacent channel interference generated by that load and controlling the congestion. We formulate the interference and congestion control as a constrained convex optimization problem and derive as solution a distributed algorithm called MINOS. Our results show that MINOS seamlessly work when vehicles have a different number of network interfaces available and effectively controls the congestion and reduces the interference compared to other proposals. Moreover, it is flexible enough to be used as a lower-level congestion and interference control framework for more sophisticated MCO algorithms.

Several future research directions are possible. An immediate one is to study the influence of the fairness parameter in the outcomes of the algorithm. A more interesting one is to refine the interference model and explicitly incorporate the effects of the transmit power and the propagation model, which would allow to set the power as a control variable and derive a joint traffic and power control algorithm. We also plan to use context information to derive suitable channel association and priority policies and integrate them into the framework control.

Appendix

An optimal solution to problem $\mathbf{E-CI}$ (3) is obtained as long as the gradient projection of Algorithm 1 converges. And the convergence follows from the Lipschitz continuity of the gradient of the Lagrange dual function (6), $\nabla g(\lambda_{cv})$ [33, Prop. 3.4], given by eq. (8) . Therefore we have to prove that

$$\|\nabla g(\bar{x}) - \nabla g(\bar{y})\|_2 \leq M\|\bar{x} - \bar{y}\|_2 \quad \forall \bar{x}, \bar{y} \geq 0 \tag{10}$$

Now, given any $\bar{x}, \bar{y} \geq 0$, using Taylor theorem we have that $\|\nabla g(\bar{x}) - \nabla g(\bar{y})\|_2 \leq \|\nabla^2 g(\bar{z})\|_2 \|\bar{x} - \bar{y}\|_2$ for some $\bar{z} = t\bar{x} + (1-t)\bar{y} \geq 0, t \in [0,1]$. Therefore, we have to find a bound for the Euclidean norm of the Hessian $\|\bar{H}\|_2 = \|\nabla^2 g(\bar{z})\|_2$.

From differentiating eq. (8), each element of the Hessian has the form $\frac{\partial^2 g(\lambda_{cv})}{\partial \lambda_{ij} \partial \lambda_{kl}} = \frac{\partial T_{ij}}{\partial \lambda_{kl}} = \sum_{v' \in N(j)} e^{t_{iv'}} \frac{\partial t_{iv'}}{\partial \lambda_{kl}}$, and we arrange them in the following matrix $\bar{H}$, with dimensions $(C \times V) \times (C \times V)$:

$$\bar{H} = \begin{bmatrix} \frac{\partial T_{11}}{\partial \lambda_{11}} & \frac{\partial T_{12}}{\partial \lambda_{11}} & \cdots & & \frac{\partial T_{CV}}{\partial \lambda_{11}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{\partial T_{11}}{\partial \lambda_{CV}} & \frac{\partial T_{12}}{\partial \lambda_{CV}} & \cdots & & \frac{\partial T_{CV}}{\partial \lambda_{CV}} \end{bmatrix} \tag{11}$$

Now, to bound it we need the following relations. For each vehicle and at each algorithm iteration, given a set of multipliers λ_{cv}, the unique minimizers in (6), t^*_{cv} are computed from the stationarity condition of the Lagrange dual (6), that is, setting $\nabla L = 0$ in eq. (9) for the Lagrangian function (5), we get

$$e^{t_{ij}} = \frac{w_{ij}}{\sum_{v' \in N(j)} [\lambda_{i,v'} + 2p(T_{i-1,v'} + T_{i+1,v'})]} \tag{12}$$

Then we take the partial derivative with respect to each of the multipliers to get the next set of equations

$$e^{t_{ij}} \frac{\partial t_{ij}}{\partial \lambda_{kl}} = \frac{-w_{ij}}{(\sum_{v' \in N(j)} [\lambda_{i,v'} + 2p(T_{i-1,v'} + T_{i+1,v'})])^2} \cdot \tag{13}$$

$$\cdot \left(\sum_{v' \in N(j)} [\frac{\partial \lambda_{i,v'}}{\partial \lambda_{kl}} + 2p(\frac{\partial T_{i-1,v'}}{\partial \lambda_{kl}} + \frac{\partial T_{i+1,v'}}{\partial \lambda_{kl}})) \right)$$

We will express the above equations in matrix form. First, we arrange the terms of the left side of eq. (13) in a matrix $\bar{E}$, with dimensions $(C \times V) \times (C \times V)$, as follows

$$\bar{E} = \begin{bmatrix} e^{t_{11}} \frac{\partial t_{11}}{\partial \lambda_{11}} & e^{t_{12}} \frac{\partial t_{12}}{\partial \lambda_{11}} & \cdots & & e^{t_{CV}} \frac{\partial t_{CV}}{\partial \lambda_{11}} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ e^{t_{11}} \frac{\partial t_{11}}{\partial \lambda_{CV}} & e^{t_{12}} \frac{\partial t_{12}}{\partial \lambda_{CV}} & \cdots & & e^{t_{CV}} \frac{\partial t_{CV}}{\partial \lambda_{CV}} \end{bmatrix} \tag{14}$$

Now, for the right side of the equation, we define the following matrices. First, for the multiplying factor in the right side of eq. (13) we define the symbol $K_{ij} = \frac{-w_{ij}}{(\sum_{v' \in N(j)} [\lambda_{i,v'} + 2p(T_{i-1,v'} + T_{i+1,v'})])^2}$ and define the matrix $\bar{K}$, with dimensions $(C \times V) \times (C \times V)$, as follows

$$\bar{K} = \begin{bmatrix} K_{11} & 0 & \cdots & 0 \\ 0 & K_{12} & \cdots & 0 \\ \cdots & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & K_{CV} \end{bmatrix} \tag{15}$$

Then we define the block diagonal matrix $\bar{N_D}$, with dimensions $(C \times V) \times (C \times V)$, as follows

$$\bar{N_D} = \begin{bmatrix} \bar{N} & 0 & \cdots & 0 \\ 0 & \bar{N} & \cdots & 0 \\ \cdots & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & \bar{N} \end{bmatrix} \tag{16}$$

where each block $\bar{N}$ is the neighbor matrix. Similarly, we define a block band matrix $\bar{N_B}$, with dimensions $(C \times V) \times (C \times V)$, as follows

$$\bar{N_B} = \begin{bmatrix} 0 & \bar{N} & \cdots & 0 \\ \bar{N} & 0 & \bar{N} & 0 \\ 0 & \bar{N} & \cdots & \bar{N} \\ 0 & 0 & \cdots & 0 \end{bmatrix} \tag{17}$$

With the previous definitions we can arrange the equations defined in eq. (13) as

$$\bar{E} = \bar{N_D}\bar{K} + 2p\bar{H}\bar{N_B}\bar{K} \tag{18}$$

If we multiply on the right by $\bar{N_D}$ both sides of the above equation we obtain the Hessian also on the right side, that is

$$\bar{E}\bar{N_D} = \bar{H} = \bar{N_D}\bar{K}\bar{N_D} + 2p\bar{H}\bar{N_B}\bar{K}\bar{N_D} \tag{19}$$

$$\bar{H}(I - 2p\bar{N_B}\bar{K}\bar{N_D}) = \bar{N_D}\bar{K}\bar{N_D} \tag{20}$$

Using the ∞-norm we have $||\bar{H}(I - 2p\bar{N_B}\bar{K}\bar{N_D})||_\infty = ||\bar{N_D}\bar{K}\bar{N_D}||_\infty \leq ||\bar{N_D}||_\infty||\bar{K}||_\infty||\bar{N_D}||_\infty$ And so

$$\frac{||\bar{H}(I - 2p\bar{N_B}\bar{K}\bar{N_D})||_\infty}{||\bar{N_D}||_\infty||\bar{K}||_\infty||\bar{N_D}||_\infty} \leq 1 \tag{21}$$

and

$$\frac{||\bar{H}||_\infty||(I - 2p\bar{N_B}\bar{K}\bar{N_D})||_\infty}{||\bar{N_D}||_\infty||\bar{K}||_\infty||\bar{N_D}||_\infty} \leq 1 \tag{22}$$

so we finally obtain

$$||\bar{H}||_\infty \leq \frac{||\bar{N_D}||_\infty||\bar{K}||_\infty||\bar{N_D}||_\infty}{||(I - 2p\bar{N_B}\bar{K}\bar{N_D})||_\infty} \tag{23}$$

The matrix in the denominator is a tridiagonal block matrix with a identity block in the diagonal and blocks corresponding to the product of $\bar{K}$ and the neighbor matrix in the off-diagonal, as can be checked. Therefore, each row only has a 1 in the diagonal element and various elements that are the sum of a number of K_{ij} factors. So $||(I - 2p\bar{N_B}\bar{K}\bar{N_D})||_\infty \geq 1 + 2pK_{min}$, where K_{min} is the minimum value of the K_{ij} factors. But, since the λ_{cv} are not bounded above, that term can be arbitrarily small, so the denominator is bounded below by 1. For the numerator, K_{max} is maximum when the λ_{cv} are zero, for a border channel and when only a vehicle transmits the minimum rate, that is $K_{max} = \frac{w_{max}}{2pd_{min}}$. So finally, we have that $||\bar{H}||_\infty \leq V^2\frac{w_{max}}{2pd_{min}}$, and $||\bar{H}||_2 \leq \sqrt{CV}||\bar{H}||_\infty \leq \sqrt{CV}V^2\frac{w_{max}}{2pd_{min}}$. Therefore, we conclude that $\nabla g(\bar{\lambda})$ is Lipschitz continuous with $M = \sqrt{C}V^{\frac{5}{2}}\frac{w_{max}}{2pd_{min}}$. From this follows that Algorithm 1 converges to the optimal values provided $0 < \gamma < 2/M$.

ACKNOWLEDGMENT

This work was supported by Grants PID2020-112675RB-C41 funded by MCIN/AEI/10.13039/501100011033 and PID2023-148104OB-C42 funded by MICIU/AEI/10.13039/501100011033 and by ERDF/EU.

REFERENCES

[1] ETSI, *Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band*, ETSI EN 302 663 v1.3.1 Std., 2020.

[2] IEEE, *IEEE Standard for Information Technology–Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks–Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016), Std., Feb 2021.

[3] ETSI, "Intelligent transport systems (ITS); vehicular communications; basic set of applications; definitions," ETSI TR 102 638 v1.1.1, Tech. Rep., 2009.

[4] ——, *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, ETSI EN 302 637-2 v1.4.1 Std., 2019.

[5] ——, *Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Decentralized Environmental Notification Basic Service*, ETSI EN 302 637-3 v1.2.1 Std., 2014.

[6] Car 2 Car Communication Consortium, "Multi-channel operation (MCO); part 1; functional requirements," Car 2 Car Communication Consortium, Tech. Rep., 2020.

[7] ETSI, "Intelligent transport systems (ITS); multi-channel operation study; release 2," ETSI TR 103 439 v2.1.1, Tech. Rep., 2021.

[8] Car 2 Car Communication Consortium, "Multi-channel operation (MCO); part 3; MCO Concept," Tech. Rep., 2022.

[9] J. Mo and J. Walrand, "Fair end-to-end window-based congestion control," *IEEE/ACM Transactions on Networking*, vol. 8, no. 5, pp. 556–567, 2000.

[10] E. Egea-Lopez and P. Pavon-Mariño, "Distributed and fair beaconing rate adaptation for congestion control in vehicular networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 12, pp. 3028–3041, 2016.

[11] E. Egea-Lopez, P. Pavon-Mariño, and J. Santa, "Optimal joint power and rate adaptation for awareness and congestion control in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 25 033–25 046, 2022.

[12] M. Boban and A. Festag, "Service-actuated multi-channel operation for vehicular communications," *Computer Communications*, vol. 93, pp. 17–26, 2016.

[13] A. Bazzi *et al.*, "Multi-channel operation for the release 2 of etsi cooperative intelligent transport systems," *IEEE Communications Standards Magazine*, vol. 8, no. 1, pp. 28–35, March 2024.

[14] K. Garlichs, A. Willecke, A.-C. Hagau, and L. C. Wolf, "Maneuver coordination for cavs: Its-g5 channel and service performance evaluation," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–15, December 2023.

[15] O. Váczi and L. Bokor, "Modeling and evaluation of a dynamic channel selection framework for multi-channel operation in its-g5," *Telecom*, vol. 4, no. 2, pp. 313–333, 2023.

[16] ETSI, "Intelligent transport systems (ITS); decentralized congestion control mechanisms for intelligent transport systems operating in the 5 ghz range; access layer part," ETSI TS 102 687, Tech. Rep., 2011.

[17] M. Sepulcre and J. Gozalvez, "Heterogeneous v2v communications in multi-link and multi-rat vehicular networks," *IEEE Transactions on Mobile Computing*, vol. 20, no. 1, pp. 162–173, January 2021.

[18] P. Pavon-Mariño, *Optimization of Computer Networks: Modeling and Algorithms. A Hands-On Approach*. John Wiley & Sons, 2016.

[19] M. Noor-A-Rahim *et al.*, "A survey on resource allocation in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 701–721, 2022.

[20] Y. Zhao, Y. Li, D. Wu, and N. Ge, "Overlapping coalition formation game for resource allocation in network coding aided d2d communications," *IEEE Transactions on Mobile Computing*, vol. 16, no. 12, pp. 3459–3472, 2017.

[21] D. Wu, X. Huang, X. Xie, X. Nie, L. Bao, and Z. Qin, "Ledge: Leveraging edge computing for resilient access management of mobile iot," *IEEE Transactions on Mobile Computing*, vol. 20, no. 3, pp. 1110–1125, 2021.

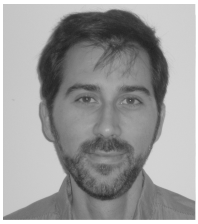
[22] Y. Xu, G. Gui, H. Gacanin, and F. Adachi, "A survey on resource allocation for 5g heterogeneous networks: Current research, future trends, and challenges," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 668–695, 2021.

[23] G. Ma, M. Hu, X. Wang, H. Li, Y. Bian, K. Zhu, and D. Wu, "Joint partial offloading and resource allocation for vehicular federated learning

tasks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 8, pp. 8444–8459, 2024.
[24] W. Feller, *An Introduction to Probability Theory and Its Applications, Volume 1*, 3rd ed. New York: John Wiley & Sons, 1991.
[25] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.
[26] R. Baldick, *Applied Optimization*. Cambridge University Press, 2006.
[27] "OMNeT++ simulation library," [online] Available: https://omnetpp.org/, accessed: June, 2024.
[28] "INET framework," [online] Available: https://inet.omnetpp.org/, accessed: June, 2024.
[29] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 82, pp. 1–5, 2016.
[30] R. Molina-Masegosa, M. Sepulcre, J. Gozalvez, F. Berens, and V. Martinez, "Empirical models for the realistic generation of cooperative awareness messages in vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5713–5717, 2020.
[31] E. Egea-Lopez, J. M. Molina-Garcia-Pardo, M. Lienard, and P. Degauque, "Opal: An open source ray-tracing propagation simulator for electromagnetic characterization," *PLOS ONE*, vol. 16, no. 11, p. e0260060, 2021. [Online]. Available: https://doi.org/10.1371/journal.pone.0260060
[32] J. Aznar-Poveda, E. Egea-Lopez, A.-J. Garcia-Sanchez, and P. Pavon-Mariño, "Time-to-collision-based awareness and congestion control for vehicular communications," *IEEE Access*, vol. 7, pp. 154 192–154 208, 2019.
[33] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Upper Saddle River, NJ: Prentice-Hall, Inc., 1989.

Esteban Egea Lopez received the Telecommunications Engineering degree in 2000, from the Polytechnic University of Valencia (UPV), Spain, the Master Degree in Electronics in 2001, from the University of Gävle, Sweden, and Ph.D. in Telecommunications in 2006 from the Universidad Politécnica de Cartagena (UPCT), Spain. He is an associate professor of the Department of Information Technologies and Communications at Universidad Politécnica de Cartagena (UPCT).

His research interest includes vehicular networks and MAC protocols.



Pablo Pavon-Mariño was born in Ourense (Spain). He received his Telecommunications Engineering degree from the University of Vigo (Spain) in 1999. In 2000 he joined the Technical University of Cartagena (Spain), where he received his Ph.D. degree in Telecommunications in 2004. In 2010 he received his degree in Mathematics from the National University of Distance Learning (Spain). Since 2007, he has been working as an Associate Professor, and since 2017 as Full Professor in the Department of Information Technologies and Communications of the Technical University of Cartagena. His research interests include performance evaluation and optimization of communication networks.