Vitale, Francesco; Roncoli, Claudio

Cooperative Rerouting to Redistribute the Load of Connected and Automated Vehicles in Urban Networks

Published in:
2023 IEEE 26th International Conference on Intelligent Transportation Systems, ITSC 2023

DOI:
10.1109/ITSC57777.2023.10422516

Published: 01/01/2023

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Cooperative rerouting to redistribute the load of Connected and Automated Vehicles in urban networks

Francesco Vitale[1] and Claudio Roncoli[1]

Abstract— **In this paper, we propose a novel distributed algorithm for cooperative rerouting of Connected and Automated Vehicles (CAVs) in urban networks, in which each intersection unit manages the portion of the network for which they are in charge by sending updated routes to follow to the CAVs therein, while communicating among each other to be updated on the situation of the (in general non-homogeneous) roads of the network. The proposed approach allows to decompose the problem into subproblems, which are resolved distributively with little information exchange. The problems are constructed to obtain a fair compromise between user equilibrium and system performance. We show the results we obtained on a simulated urban network with CAVs and compared them with a baseline scenario.**

I. INTRODUCTION

Connected and automated vehicles (CAVs) are becoming a reality, and, together with opportunities [1], [2], challenges arise likewise, e.g., how to integrate CAVs with human-driven vehicles, guarantee connectivity, and develop CAV real-time planning and control strategies [3], [4]. Among such challenges is vehicle routing, i.e., determining a proper route for the CAVs to follow. Route search and planning have been vastly researched, considering their use in spatial data management and location-based social services, and multimodal route planning in transportation systems [5], [6].

A comparative study of vehicles' routing in smart cities is presented in [7], where the authors identify a main classification into three categories, namely: 1) optimal, 2) heuristic, and 3) hybrid algorithms. Optimal algorithms guarantee to find the global optimal solution through the exploration of the whole set of available solutions; examples of such methods are Dijkstra and Incremental Graphs. Heuristic-based approaches explore a subset of the available solutions and usually find an approximate optimal solution with qualities close to those of the global optimal one; examples are A*, Genetic Algorithms, Ant Colony Optimization, and Tabu Search. Finally, hybrid algorithms leverage the strengths of both of the previous; an example of a hybrid method could apply a combination of Dijkstra and Genetic Algorithms.

In the research work [8], the authors consider the problem of routing from a broader perspective, including vehicular communication, sensing, localization, internet of things, computing, and machine learning. They also present a review of route planning metrics, outlining their evolution from the traditional distance/time/cost to metrics more relevant to vehicle autonomy such as scenery and safety. Such discussion is also expanded to encompass the complexity of route planning in the presence of autonomous vehicles, and how this complexity can be managed. Finally, they highlight the necessity for considering traffic management during events or emergencies, so that both traffic management directives and autonomous inter-vehicle collaboration may lead to safer and faster resolutions of congestion.

Other major efforts focus on the optimization of multi-objective routing problems. In [9], authors offer an overview of which application-oriented multi-objective routing problems are treated and which trade-offs are investigated. Algorithmic approaches are analyzed with regard to their fitness assignment strategy, namely how the multiple objectives are handled, and their search strategy to solve the problem. A rich classification is presented, based on the objective types: 1) related to flows/tours (such as distance, reliability, environment, health risk, military risk); 2) related to nodes/arcs/edges (such as comfort, time to destination, customer cost); and 3) related to resources (such as facilities, transportation goods, goods).

Most of the previous approaches aim at minimizing a, so to speak, user-centered cost, which usually depends on factors such as the time from the origin to the destination, speed, and waiting time for every single vehicle. In this regard, e.g., an interesting concept has been proposed [10] to minimize traffic congestion through continuous-time route reservations with travel time predictions. Nevertheless, not always the current fastest/shortest routes are the most appropriate to follow, as pointed out in [11], where authors propose a double rewarded value iteration network to model current and future states and learn the experienced drivers' routing decisions based on their estimation of traffic trends. This is to take advantage of the knowledge of experienced drivers about the topology and traffic trends of the roads they are traversing. On the other hand, only a few works address the routing problem by affecting the macroscopic traffic variables, thus minimizing traffic congestion considering current measurements of the density, volume, and mean speed of the traffic. For example, authors in [12] use a discrete consensus algorithm to coordinate autonomous agents and redistribute the traffic through rerouting. However, the latter type of approach is usually under the assumption that the intersections lie in a network of roads with a common fundamental diagram, which is not realistic.

A crucial point in this context is understanding if and how the adoption of a system optimum solution (i.e., mini-

This work was supported by the Academy of Finland project ALCOSTO (no. 349327).

[1]Francesco Vitale and Claudio Roncoli are with the Department of Built Environment, Aalto University, 02150 Espoo, Finland `francesco.vitale@aalto.fi`, `claudio.roncoli@aalto.fi`

mizing only network-level objectives) affects the behavior of an urban network with CAVs and what is its role in relation to the user optimum (i.e., minimizing only user-level objectives). A comprehensive review is offered in [13], which discusses possible ways to bridge the gap between the two optima. In particular, user optimum results from a user-centered traffic assignment where each user chooses the most convenient route selfishly. When the equilibrium is reached, all users sharing the same origin and destination will experience the same travel time, hence, guaranteeing respect for fairness. However, user optimum generally does not minimize the system's total performance. On the other hand, the system optimum results from a system-wide traffic assignment where drivers are routed on the network in such a way the total travel time is minimized. However, if system optimum is achieved, some users may experience travel times that are higher than other users traveling from the same origin to the same destination, which may be considered as unfair.

Authors in [14] develop a route control scheme to achieve the system optimum of a network by controlling a portion of the CAVs in an attempt of demonstrating the potential of automated vehicles as control actuators for improving traffic network performance. Their results show that a minimum control ratio can be identified. Furthermore, they propose a joint route control for CAVs and pricing for uncontrolled vehicles scheme that demonstrate the synergistic effect of the joint scheme on reducing the minimum control ratio and the financial burden on travelers. A multi-class model dynamic traffic assignment is developed in [15] to equally distribute the total queues over the links while considering explicitly the variations in capacity and backward wave speeds. The results show that the introduction of automated vehicles changes the traffic flow dynamics. This leads to a reduction of the spillback and the total system travel time, and even more so with increased penetration rate and total demands, when heavy traffic congestion occurs.

The studies cited above for the analysis of the system optimum assume the presence of a central unit to handle the load of information and yield the optimal solution. Therefore, they focus on redefining the optimization problem to favor its scalability and tractability. On the other hand, a distributed approach might be utilized instead. The impact of distributed dynamic routing with different market penetration rates of CAVs and congestion levels on urban roads is investigated in [16], which refers to the Downtown Toronto network case study in an agent-based traffic simulation. The results show that the higher the market penetration rates of CAVs – especially in the case of highly congested urban networks – the higher the average speed, the lower the mean travel time, and the higher the throughput.

In this work, we propose a cooperative real-time rerouting algorithm for CAVs in order to obtain an efficient redistribution of the flows in an urban network. The algorithm is constructed so that its solution leans towards a user equilibrium as far as possible while still coping with the overall system performance for larger loads of vehicles. Moreover, we argue its predisposition to be solved in a distributed fashion. In Section II we present the methodology, followed by Section III showing the results obtained by running a simulation. Finally, Section IV summarizes our work and suggests topics for future developments.

II. METHODOLOGY

In this section we underline the proposed methodology, starting from the setting and assumptions, followed by the presentation of the algorithm.

A. Setting

We consider a network of intersections connecting roads, where each road is characterized by a possibly different fundamental diagram. We assume the presence of intersection unit devices (IUs) able to communicate with each other and with CAVs. We define a route to be followed by a CAV as the sequence of intersections (or, interchangeably IUs) to drive through from an origin intersection to a destination intersection. Despite the so-defined route is not realistic since CAVs do not have origin and destination exactly at intersections, we keep such a definition for the sake of simplicity and consider the needed modification for future developments. Fig. 1 gives a practical example of the network considered. The role of the IUs is to sense the presence of the CAVs within their area of competence and/or be informed by them via communication to reinforce the reliability of the sensed data. Furthermore, the IUs communicate and propagate information among each other to make sure that each of them has an updated *situation* of the network, which is described by the number of vehicles driving on each road at a given time. Other information about the roads in the network are necessary, namely the capacity volume and density (or the number of vehicles at capacity, for simplicity), and the jam density (or number of vehicles at jam, for simplicity). The latter are, in general, resulting from a fundamental diagram that is different for each road and is uploaded to the IUs beforehand. The IUs update and deliver the route to be followed by every CAV within its *range*, which encompasses the vehicles within its inflowing roads

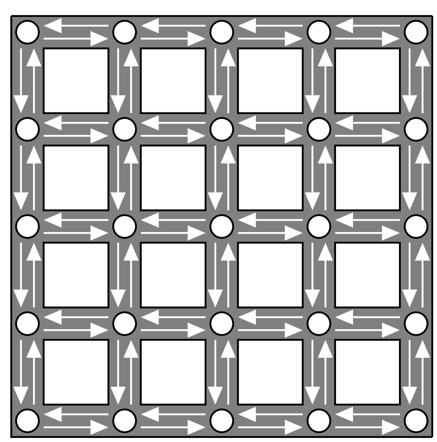

Fig. 1. The network considered. The white circles represent the IUs.

Algorithm 1 Rerouting algorithm.

Require: $G(I,E), o, d, N, N^c, V, L$

Ensure: P

```
1:  d_o ← 0
2:  for all i ∈ I \ o do
3:      d_i ← ∞
4:      Q.add(i, d_i)
5:      P_i ← null
6:  while Q ≠ ∅ do
7:      l ← argmin_{i∈Q}{d_i}
8:      Q ← Q \ l
9:      for all (l, j) ∈ E do
10:         c_lj ← d_l + σ_{N^C_lj}(κ_1, N_lj, N^C_lj) + κ_2 L_lj/V_lj
11:         if c_lj < d_j then
12:             d_j ← c_lj
13:             P_j ← l
14: if just entered or exited then
15:     Update N_lj
16:     Update V_lj
```

driving towards the IU itself. The CAVs can employ their communication capabilities to inform an IU when arriving within its area of competence, as already mentioned. Moreover, the CAVs will receive the updated route and actuate some techniques to track thereof. The tracking techniques are not the object of this study, but they can, e.g., address intersection coordination and minimization of the error to the reference route as in [17], [18], [19], [20].

B. Proposed algorithm

The algorithm run by the IUs is a modified version of the Dijkstra algorithm, where we customize the cost function. Though we believe that it is possible to utilize other routing algorithms with such a cost, in the present paper we focus on Dijkstra, while we intend to investigate other algorithms in future works. A CAV driving towards an intersection communicates with the relative IU as soon as it is within its communication range (and/or will be sensed by the IU itself). At this point, and at every time instant thereafter, the IU runs Algorithm 1 (Rerouting Algorithm) for the CAV until it exits its range. The input needed for the algorithm to run includes the map G of intersections I and roads connecting them E, the origin o which is the next intersection that will be visited by the CAV (i.e., the IU itself), the destination d which is the last intersection that the CAV wishes to visit through its route, a matrix N containing the most recent values of the number of vehicles on each road, a matrix N^c containing the values of a critical number of vehicles (i.e., the number of vehicles for which the traffic is flowing at capacity) on each road according to their fundamental diagram, a matrix V containing the most recent values of mean speed on each road, and a matrix L containing the value of the length of each road. Matrices N (used as a proxy for the density), N^c (relating to the critical density), and V are uploaded beforehand and updated via communication with the surrounding IUs. Indeed, after every periodical measurement, the IUs spread the information to the other IUs, together with a timestamp, so that each of them has an estimation of the most recent traffic situation. The rest is a variation of the Dijkstra algorithm where the initialization requires the distance to the origin to be 0, the distance to the other intersections to be infinity for every intersection, such a distance to be added in a queue, and the predecessor list to be null (lines 1–5). Unless the queue is empty, the least *distant* to the origin neighbor (intersection l) is chosen and removed from the queue (lines 6–8). For all the neighbors j of the latter, the cost is computed as the summation of three terms. The first term, i.e., d_l, is the cost to l. The second term is

$$\sigma_{N^C_{lj}}(\kappa_1, N_{lj}, N^C_{lj}) = \frac{1}{1+e^{-\kappa_1(N_{lj}-N^C_{lj})}}, \tag{1}$$

where the subscripts lj refers to the road examined and $\sigma_{N^C_{lj}}(\cdot)$ is a sigmoid function centered at the capacity N^C_{lj} and skewed by the slope constant κ_1. The third term ($\kappa_2 \frac{L_{lj}}{V_{lj}}$) is the time needed to drive through the road (l,j), namely $\frac{L_{lj}}{V_{lj}}$, resized by a factor κ_2 to give more or less importance with respect to the $\sigma_{N^C_{lj}}(\cdot)$ function (line 10). Notice that, while the second element allows a load-sharing process among the roads, the third one takes into account the time needed to traverse a road itself. The constants κ_1 and κ_2 play an important role in tuning the algorithm to lean towards the user or the system optima. The idea behind this formulation is to let the CAVs give higher importance to following the user equilibrium as long as the number of vehicles is lower enough than the critical value. This would translate into allowing $\kappa_2 \frac{L_{lj}}{V_{lj}}$ to be higher than $\sigma_{N^C_{lj}}(\cdot)$ until the latter reaches a value slightly smaller than 0.5 which is, by construction, at the critical number of vehicles. Therefore, supposed that one wants to promote the user equilibrium for, e.g., $\sigma_{N^C_{lj}}(\cdot) \leq 0.4$, then κ_2 could be calculated as $\kappa_2 = 0.4 \frac{V_{lj}}{L_{lj}}$ at free-flow speed. For higher values of $\sigma_{N^C_{lj}}(\cdot)$, the system performance should be prioritized, hence $\kappa_2 \frac{L_{lj}}{V_{lj}}$ should be lower (see Fig. 2 for a clarification of the shape of these two functions).

The remaining updates the least distant neighbor and sets the predecessor (lines 11–13). Finally, if the CAV has just entered or exited road (l,j) (line 14), the number of vehicles on the road (l,j) is updated (line 15) and the mean speed is updated according to a fundamental diagram $f^D_{lj}(N_{lj})$ (line 16). In particular, consider the i^{th} CAV entering the road $(l,j) \in r_i$, where r_i is the route of CAV i, at time t. Define T as a sufficiently large number to be used as the time horizon. Then the update on the situation is

$$\begin{aligned} N_{lj}(t,\cdots,T) &= N_{lj}(t)+1, \quad \text{if } N_{lj}(t) < N^J_{lj}(t) \\ V_{lj}(t,\cdots,T) &= f^D_{lj}(N_{lj}(t)), \end{aligned} \tag{2}$$

which is evaluated every time a vehicle requests to enter the road (l,j), either from outside, i.e., for entering the network itself, or traveling from another road of the network (x,l). Analogously, the update rule for CAV i when exiting

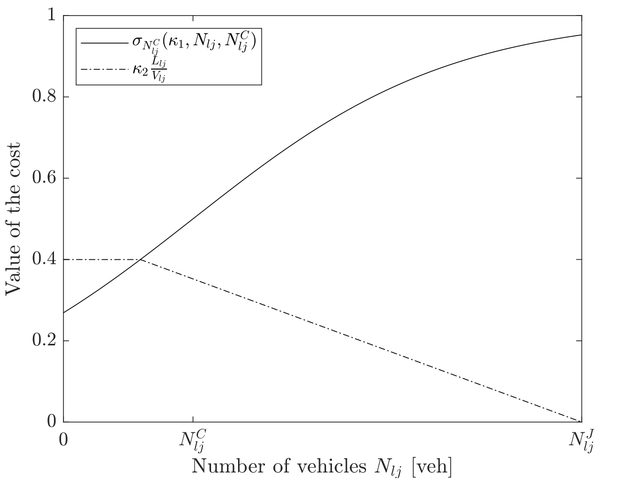

Fig. 2. An example of the two main cost terms for traversing a road (l, j), considered a triangular fundamental diagram.

a road (l, j) is

$$
\begin{aligned}
N_{lj}(t+1, \cdots, T) &= N_{lj}(t+1) - 1, \qquad \text{if } \{\exists (j,x) \in r_i \\
&\qquad \text{and} \\
&\qquad N_{jx}(t) < N^J_{jx}(t)\} \\
&\qquad \text{or } \nexists (j,x) \in r_i \\
V_{lj}(t+1, \cdots, T) &= f^D_{lj}(N_{lj}(t+1)).
\end{aligned} \tag{3}
$$

Notice that the increase update in (2) takes place immediately at time t to avoid conflicts with multiple CAVs entering, while the decrease update in (3) takes place at the following instant $t+1$ for the same reason. The updates are set until T so that they hold even for following instants where the CAVs are neither entering nor exiting a new road.

Given the nature of the proposed solution, with the possibility that every IU takes care of the CAVs within its respective range while communicating to achieve better performances, we can conclude that this algorithm falls in the category of distributed routing via sharing of the vehicle load among the roads in the urban network.

III. RESULTS

We present here the results obtained by running a set of simulation experiments. First, we describe the setting, then we show the results for a baseline technique, namely a basic Dijkstra, where routes are computed considering the time needed to travel in free-flow, hence without checking the actual situation of the congestion, and those applying the proposed algorithm for comparison.

A. Setting

We generate an urban network with 4000 CAVs driving, where the first vehicle enters at 0 s, and the last at 240 s, and show the results until 360 s. We choose such a controlled scenario to better assess the behavior of all CAVs and give enough time for all of them to exit the network. The network utilized is the same as the one in Fig. 1. The CAVs are assigned random origin, destination, and initial time at which they enter the network. We adopt a triangular fundamental diagram for each road whose macroscopic values are: number of vehicles at capacity $N^C = 7$, number of vehicles at jam $N^J = 20$, free-flow speed $V^F = 50$ km/h, and length of the road $L = 0.1$ km. For the proposed algorithm, we set $\kappa_1 = 0.1$ and $\kappa_2 = 0.05$.

In what follows, we focus on the results obtained for the actual total time traveled as compared to the total time traveled that would be experienced in free-flow, and the occupancy of the road, where we define the occupancy as N_{lj}/N^J_{lj}, namely the ratio between the number of vehicles and the number of vehicles at jam density on road (l, j).

B. Baseline Dijkstra

Fig. 3 shows that the average occupancy of the network does not reach very high levels. However, after about 40 s, there is often at least one road suffering from maximum occupancy reached. Let us consider a portion of the network, depicted in Fig. 4, more in detail. Results reported in Fig. 5 show that road A reaches maximum occupancy and the congestion propagates backward to roads B and D as well. Roads C and E show signs of milder congestion coming from road B.

C. Proposed algorithm

Fig. 6 shows that the average occupancy of the network does not reach very high levels and it is rather similar to the baseline. On the other hand, the maximum occupancy reached by any road is always remarkably lower than that of the baseline simulation, and, in particular, none of the roads ever reaches its maximum occupancy. This suggests that the load of CAVs has been effectively redistributed across the network. Let us consider the portion of the network as in Fig. 4 once again. Fig. 7 shows that roads A, B, and D, which were among the most affected by congestion, have lower occupancy values and operate in a congested state

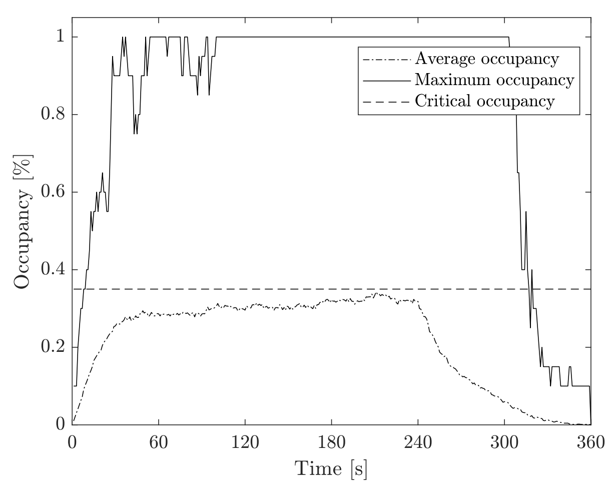

Fig. 3. Occupancy averaged by the whole network and maximum occupancy reached by any road in the network for the baseline algorithm simulation.

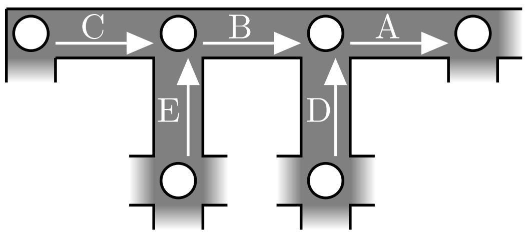

Fig. 4. The portion of network referred to in Figs. 5 and 7.

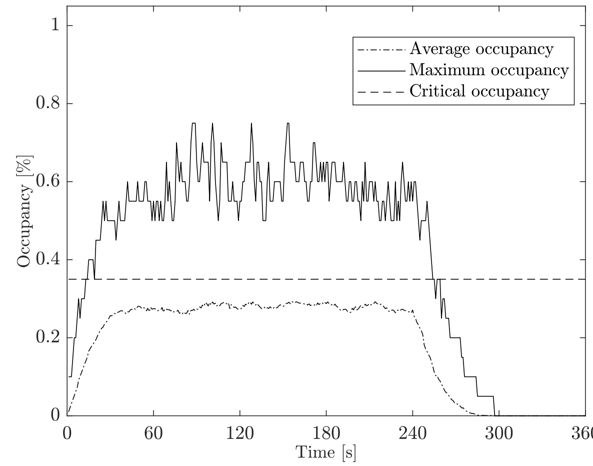

Fig. 6. Occupancy averaged by the whole network and maximum occupancy reached by any road in the network for the proposed algorithm simulation.

only for a short time throughout the entire simulation. Fig. 8 shows further insight into the cumulative occupancy, namely the sum of the occupancy of every road throughout the time horizon. Obviously, the two curves overlap as long as the network is in free flow. We recall that this result is induced by properly setting the κ_2 constant as previously discussed. As soon as the congestion occurs, the baseline algorithm leads to accumulating more and more vehicles on the roads keeping them occupied for longer. On the other hand, the proposed algorithm exhibits a lower accumulation, suggesting a more effective dispersion of the congestion. As a final remark, the average time spared by each CAV with the proposed algorithm to go from origin to destination as compared to the time spent in the baseline simulation is 6.552 s. Table I compares the total time traveled and delay between the baseline and the proposed algorithm, where one can see that the proposed algorithm allows a strong improvement in network-level metrics. Indeed, the proposed algorithm has improved the time delay by 87.76% and the total travel time by 21.22%.

IV. CONCLUSIONS

This study showed the ability of congestion dispersion utilizing a modified version of the Dijkstra algorithm that constructs the distance between intersections considering both the current travel time and the situation of the network given by the number of vehicles on each road. We designed our technique to be solved in a distributed fashion allowing scalability at the cost of a small amount of data exchange. We finally showed convincing results of the effectiveness of the proposed algorithm as compared to a baseline technique.

Future research on this topic might include: 1) a sensitivity analysis of parameters of the algorithms, such as the slope of the sigmoid and the resizing factor of the cost for the time needed to traverse a road; 2) considering additional costs and/or modifying the sigmoid function; 3) considering bigger networks and possibly heuristic or hybrid approaches (even though keeping the same considerations on the cost); and 4) investigating the resilience to communication delays, packet loss, and altered information exchange.

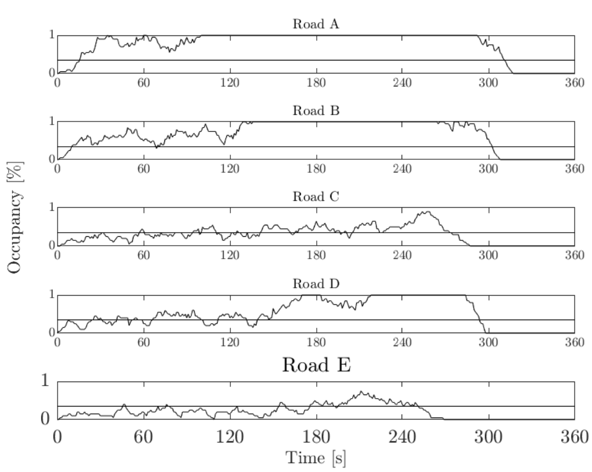

Fig. 5. Occupancy of the roads A, B, C, D, and E referred to in Fig.4 showing evidence of congestion. The straight lines in the plots represent the critical occupancy. Results for the baseline algorithm simulation.

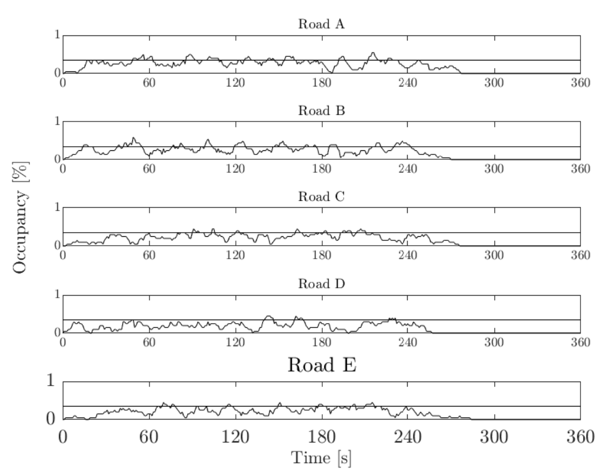

Fig. 7. Occupancy of the roads A, B, C, D, and E referred to in Fig.4 showing lower levels of congestion. The straight lines in the plots represent the critical occupancy. Results for the proposed algorithm simulation.

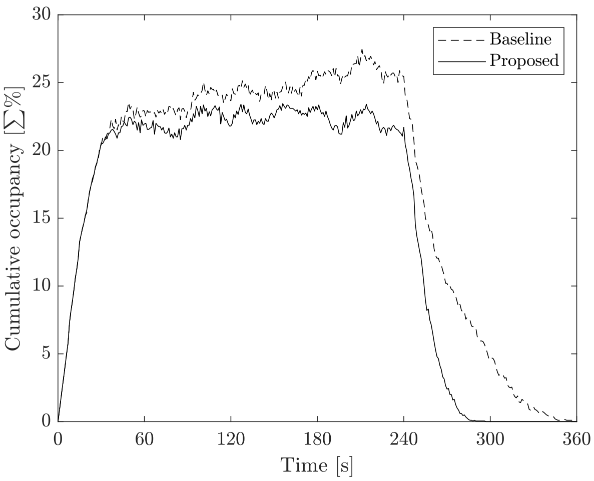

Fig. 8. Cumulative occupancy of the roads throughout the time.

TABLE I

NUMERICAL COMPARISON BETWEEN THE BASELINE AND THE PROPOSED ALGORITHM

	Baseline	Proposed algorithm
Total Travel Time	34h 18m 39s	27h 01m 52s
Total Delay	08h 17m 42s	01h 00m 54s

REFERENCES

[1] A. Sarker, H. Shen, M. Rahman, M. Chowdhury, K. Dey, F. Li, Y. Wang, and H. S. Narman, "A review of sensing and communication, human factors, and controller aspects for information-aware connected and automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 1, pp. 7–29, 2019.

[2] D. Elliott, W. Keen, and L. Miao, "Recent advances in connected and automated vehicles," *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 6, no. 2, pp. 109–131, 2019.

[3] J. Guanetti, Y. Kim, and F. Borrelli, "Control of connected and automated vehicles: State of the art and future challenges," *Annual Reviews in Control*, vol. 45, pp. 18–40, 2018.

[4] I. Papamichail, N. Bekiaris-Liberis, A. I. Delis, D. Manolis, K.-S. Mountakis, I. K. Nikolos, C. Roncoli, and M. Papageorgiou, "Motorway traffic flow modelling, estimation and control with vehicle automation and communication systems," *Annual Reviews in Control*, vol. 48, pp. 325–346, 2019.

[5] K. Li, X. Rao, X. Pang, L. Chen, and S. Fan, "Route search and planning: A survey," *Big Data Research*, vol. 26, p. 100246, 2021.

[6] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, and R. F. Werneck, "Route planning in transportation networks," *Algorithm Engineering: Selected Results and Surveys*, pp. 19–80, 2016.

[7] V. T. N. Nha, S. Djahel, and J. Murphy, "A comparative study of vehicles' routing algorithms for route planning in smart cities," in *2012 First International Workshop on Vehicular Traffic Management for Smart Cities (VTM)*. IEEE, 2012, pp. 1–6.

[8] A.-E. Taha and N. AbuAli, "Route planning considerations for autonomous vehicles," *IEEE Communications Magazine*, vol. 56, no. 10, pp. 78–84, 2018.

[9] S. Zajac and S. Huber, "Objectives and methods in multi-objective routing problems: A survey and classification scheme," *European Journal of Operational Research*, vol. 290, no. 1, pp. 1–25, 2021.

[10] C. Menelaou, S. Timotheou, P. Kolios, C. G. Panayiotou, and M. M. Polycarpou, "Minimizing traffic congestion through continuous-time route reservations with travel time predictions," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 1, pp. 141–153, 2018.

[11] J. Li, D. Fu, Q. Yuan, H. Zhang, K. Chen, S. Yang, and F. Yang, "A traffic prediction enabled double rewarded value iteration network for route planning," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 5, pp. 4170–4181, 2019.

[12] C. Wuthishuwong and A. Traechtler, "Distributed control system architecture for balancing and stabilizing traffic in the network of multiple autonomous intersections using feedback consensus and route assignment method," *Complex & Intelligent Systems*, vol. 6, no. 1, pp. 165–187, 2020.

[13] V. Morandi, "Bridging the user equilibrium and the system optimum in static traffic assignment: A review," *4OR*, pp. 1–31, 2023.

[14] Z. Chen, X. Lin, Y. Yin, and M. Li, "Path controlling of automated vehicles for system optimum on transportation networks with heterogeneous traffic stream," *Transportation Research Part C: Emerging Technologies*, vol. 110, pp. 312–329, 2020.

[15] D. Ngoduy, N. Hoang, H. Vu, and D. Watling, "Multiclass dynamic system optimum solution for mixed traffic of human-driven and automated vehicles considering physical queues," *Transportation Research Part B: Methodological*, vol. 145, pp. 56–79, 2021.

[16] L. Alfaseeh, S. Djavadian, and B. Farooq, "Impact of distributed routing of intelligent vehicles on urban traffic," in *2018 IEEE International Smart Cities Conference (ISC2)*. IEEE, 2018, pp. 1–7.

[17] F. Vitale and C. Roncoli, "Distributed trajectory optimization for signal-free intersection control with connected and automated vehicles," *under review*, 2023.

[18] ——, "Distributed formation control for managing cav overtaking and intersection maneuvers," *IFAC-PapersOnLine*, vol. 55, no. 13, pp. 198–203, 2022.

[19] ——, "Reference tracking optimization with obstacle avoidance via task prioritization for automated driving," *under review*, 2022.

[20] ——, "An mpc-based task priority management approach for connected and automated vehicles reference tracking with obstacle avoidance," in *2021 European Control Conference (ECC)*. IEEE, 2021, pp. 813–819.