A Framework for Risk-Aware Routing of Connected Vehicles via Artificial Intelligence

Matteo Cardellini[1] and Carmine Dodaro[2] and Marco Maratea[2] and Mauro Vallati[3]

Abstract— **The advent of Connected Autonomous Vehicles can enable the use of Artificial Intelligence (AI) techniques to support urban traffic controllers in extending their control capabilities with the ability to distribute vehicles in a urban region. Vehicles can communicate their destination, and receive an optimised route by traffic controllers. While the benefits of traffic routing are clear, it is also clear that re-routing has the potential to increase risks for vehicles' and passengers' safety due to environmental or urban factors. There is however a lack of work in the area of risk-aware routing.**

To fill the above-mentioned gap, we introduce a framework to incorporate risk-awareness in the vehicle routing process. The proposed framework provides a principled structure to define and characterise different classes of risk that can arise in a region, allowing to take them into account when generating routes. We show how this framework can be implemented, and we provide an empirical analysis of its performance on two European urban areas.

I. Introduction

The advent of Connected Autonomous Vehicles (CAVs) provides a fertile ground for a fundamental transformation in urban mobility and traffic control [1], [2]. The integration of connectivity and autonomous driving capabilities in vehicles is expected to yield significant benefits, such as reducing the occurrence of accidents, curbing carbon emissions, and enhancing time savings [3], [4], [5], [6].

A critical aspect of CAVs is their ability to communicate through dedicated protocols and networks, known as Vehicular Ad Hoc Networks (VANETs). This communication capacity unleashes the use of Artificial Intelligence (AI) techniques for supporting urban traffic controllers in innovative ways. In particular, Vehicle to Infrastructure (V2I) communication permits traffic controllers to collect real-time traffic information from CAVs operating in the area, thus providing a real-time overview of traffic conditions. In turns, this allows traffic controllers to directly influence traffic by instructing vehicles on the best route to follow to reach their destination [7], [8].

While the benefits of traffic routing are clear and well investigated, there has been limited focus on the analysis of risks that this approach can bring to vehicles and to the controlled network. There has been some interest in understanding features and aspects that increase the likeliness of accidents in general urban traffic conditions [9], [10], but there is a range of factors such as weather, inadequate lighting condition, time of the day, characteristics of roads and junctions, etc. that can increase the risk of accidents or compromise the safety of passengers, particularly if a path is assigned to a potentially large number of vehicles. To ensure a beneficial exploitation of this class of approaches, it is crucial to take risk into account when routing vehicles in a urban network.

Considering the wider field of mobility and traffic control, most of the existing body of works focuses on risk-aware road movement and path planning to avoid collisions with infrastructure, other vehicles, or pedestrians [11], [12]. Additional examples include [13], that describes an architecture for supporting the risk-aware operation of autonomous vehicles; [14] defines safe and hazardous states for autonomous vehicles in an urban area, and [15] introduces an approach for minimising the possibility of collisions with pedestrians. Finally, [16] investigates how risk measures can affect the behaviours and trajectories of autonomous vehicles in an urban environment. A different line of works focus instead on risk related to the management of traffic signalised junctions. For instance, [17] defines pedestrian-safety-aware approaches for optimising traffic signals.

In this paper, we introduce a framework to deal with risk-aware vehicle routing by means of AI, more specifically Answer Set Programming (ASP) [18], [19], [20]. In particular, the proposed framework provides a structure to define and characterise the different classes of risk that can arise in the considered region, and to take them into account when routes are generated. We demonstrate in simulation, using real-world historical data, the improvements that can be achieved by using traffic routing in two very different urban areas, and we take the opportunity to discuss the benefits of embedding risk-awareness in the optimisation framework.

When it comes to generating routes and optimising traffic movements, here we focus on ASP for a number of reasons. ASP is a declarative programming paradigm that provides a flexible and intuitive way to model and solve complex combinatorial problems, as optimal vehicle routing. Compared to traditional imperative programming, which relies on step-by-step instructions, ASP allows users to specify the rules and constraints of a problem in a logical and intuitive way. ASP is particularly useful for solving problems involving incomplete or uncertain information, or those with a large number of possible solutions. As a matter of fact, ASP has been applied in various domains including traffic control, scheduling, diagnosis, and knowledge representation [21], [22], making it a valuable tool for researchers and practitioners in many

[1]Matteo Cardellini is with the University of Genoa, Genoa, Italy.

[2]Carmine Dodaro and Marco Maratea are with the University of Calabria, Rende, Italy.

[3]Mauro Vallati is with the University of Huddersfield, Huddersfield, UK. Mauro Vallati was supported by a UKRI Future Leaders Fellowship [grant number MR/T041196/1]. m.vallati@hud.ac.uk

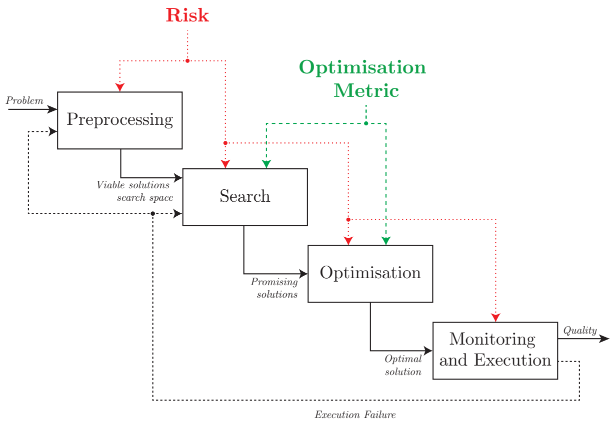

Fig. 1. The proposed framework to deal with risk and optimisation in traffic distribution.

fields. Moreover, ASP has several advantages over other solving paradigms. Firstly, ASP specifications are often more readable even to non-experts, which is essential when the solution has to be explained to sceptical users. Secondly, there are free and open-source systems like Clingo [23], whose performance is often comparable to that of industrial tools like CPLEX or Gurobi, or SAT solvers. Thirdly, if the performance of plain ASP is not satisfactory, there are several extensions available. Regarding Clingo and CPLEX, both are software tools used in optimisation, with different features and target users. Clingo is generally more user-friendly, with a simpler syntax, a semantics based on logic, and an intuitive modelling approach, while CPLEX requires a deeper understanding of mathematical programming, linear algebra and optimisation theory, and is typically used by experienced professionals. Moreover, Clingo is open-source and freely available, while CPLEX is a commercial product that requires a license to be used.

The remainder of this paper is organised as follows. First, we introduce and describe the proposed framework and the implemented components. Second, we present the results of the performed empirical analysis, followed by a detailed discussion. Finally, conclusions are given.

II. Proposed Framework and Implementation

To address risk-aware vehicles routing by means of AI we defined a framework, shown in Figure 1, composed by four phases. In the remainder of this section, we describe each of them in detail, as well as the required input and the expected mode of operation. We also describe the type and kind of risks that each step of our implementation can consider. On this regard, it is worth noting that the system will be tested using a traffic simulator, hence environmental information such as poor lighting or poor road surfaces are not available and can not be considered. However, it is straightforward to extend the proposed techniques if the corresponding data are available.

A. Preprocessing

The role of the *Preprocessing* step is to reduce the complexity of the problem by simplifying the considered network. This can include actions such as abstracting away parts of the network that are not viable to route vehicles or merging short subsequent links together, etc. It takes in input (as an XML) the network structure, the list of incoming new vehicles and the position of all the vehicles which have already a route inside the network. The preprocessor then outputs a simplified logical representation describing the network, modelling the search space in which viable solutions can be found.

When dealing with risk, in the preprocessing phase it is possible to exclude solutions which should not be taken into account due to an increased risk of accidents or gridlocks. For example, we remove

- paths including sequences of very short links separated by junctions or traffic lights. These are traditionally used for minor traffic flows and, if used by large volumes of vehicles, can dramatically increase the probability of traffic accidents and gridlocks;
- paths including narrow streets, where the probability of accidents, pedestrians crossings, or vehicles leaving parking slots is very high and can have a significant impact if traffic flows are suddenly increased.

It is worth highlighting that preprocessing is a common feature of combinatorial search approaches, and is generally needed for search-based methods to ensure that the complexity of the problem to be solved is manageable. For different classes of techniques, preprocessing may not be needed for performance-related reasons, but it should be implemented to address the corresponding risks, including those listed above.

B. Search

This phase focuses on, given a simplified representation of the network (provided by the *Preprocessing*) and the current traffic conditions (via monitoring), identifying suitable routes for every vehicle and computing time ranges in which the vehicles will enter or exit each link of the considered path.

The origin and destination of each vehicle that approaches the network is known when it enters the region, as it can be communicated via a VANET, and it is the purpose of the framework to generate a route for each vehicle. In the developed approach, for all the incoming vehicles, the search process computes all the possible (acyclic) paths in the graph network which connect their source and target streets. This results in an exponentially large number of routes, particularly critical in the case of large urban networks. To reduce complexity, the developed *Search* phase takes an additional step by grouping similar routes together and then taking a subset from each group (according to a heuristic, which states how to order each solution and how many solutions to bring to the next phase). This will allow the *Optimisation* step to consider a diversified portfolio routes to better distribute traffic.

The identification of routes also involves the assessment of corresponding time windows, i.e. the expected time at which vehicles will enter and exit a given link on the path. While an accurate computation is expensive, an approximate time window can be calculated via relaxation. Intuitively, the minimum time in which a vehicle enters and exits a street in a route can be computed as if all traffic were removed, meaning the vehicle is not slowed down and the streets are run at maximum speed. A maximum exit time, instead, can be computed by analysing how many vehicles would be expected at the street at its minimum entry time, and then reduce the expected vehicle's speed accordingly. This is a gross approximation, of course, but can still provide some information on the expected time in which a vehicle will reach a link, hence supporting the simulation of traffic evolution in the network.

It is worth reminding that the search step does not select the best route to assign to a considered vehicle, but instead provides a list of promising routes to the optimiser, that will then be in charge of performing the selection and assignment process.

In terms of risk, the *Search* can take into account the following:

- Minimising the risk of accidents by avoiding routes that are already congested. Congestion is a well-known key risk indicator for traffic accidents.
- Reducing the risk of vehicles not following the given instructions, leading to potential hazards in the controlled urban region due to unexpected traffic movements, by avoiding the generation of routes that are not within a given bound from the shortest path.

C. Optimisation

Given the portfolio of routes generated by the search step, the *Optimisation* module is in charge of selecting the best one to be assigned to each vehicle. In principle, this step can either focus on the optimisation of the route for a (set of) vehicle, or can take a network controller perspective and aims at optimising the overall behaviour of traffic. In this work we consider the latter, hence our approach aims at reducing overall congestion while taking risks into account.

After finding the most different viable shortest routes in the previous phase, it is now time to search for the best possible solution, whose purpose is optimising the flow of traffic and reducing risk inside a road network, thus subjecting the optimisation to the whole network and finding the best combination (schedule) of routes for all the vehicles in the network, and not, we remind, in finding the best possible route for every single vehicle.

Here, for optimisation we use ASP [24], that is a well-known declarative programming language. The unfamiliar reader is referred to [25] for a more comprehensive description of the approach and its capabilities. In the rest of this section we will introduce only a few concepts for the sake of describing the designed encoding. Solving complex problems using ASP requires to write a set of logical rules in the ASP syntax and often involves a methodology called Guess, Check, and Optimise consisting of four steps:

- Identifying how the domain model can be expressed in terms of ASP, including input and output of the ASP encoding. ASP typically follows a relational notation, meaning that a one-to-one mapping between database tables and ASP input/output is common. For example, a vehicle can be modelled as

  `vehicle(·,·)`

  where the first parameter is its ID and the second parameter is the type of vehicle. A possible route for the vehicle can be modelled as

  `possibleRouteOfVehicle(·,·)`

  where the first parameter is the ID of the vehicle and the second parameter is the ID of the route. Note that, for each vehicle, there might be multiple possible routes.
- Defining the rules encoding to generate candidate solutions, i.e., the search space of the problem (Guess). For example, a rule of the form `{solutionRoute(V,R): possibleRouteOfVehicle(V,R)} = 1 :- vehicle(V,1)` generates a set of candidate solutions where, for each vehicle of type 1, exactly one route among the possible routes that can be followed by the vehicle is selected and stored in the relation expressed by `solutionRoute(V,R)`, indicating that the vehicle `V` follows the route `R`.
- Checking whether candidate solutions satisfy the requirements of the problem. For example, a rule of the form `:- enter(V,S,T), vehicle(V,1), capacity(S,MAX), nVehicleOnStreet(S,T,N), N > MAX` is used to discard the solutions where the maximum capacity of the street is exceeded at a given time point `T`.
- Selecting the solutions that optimise some objective function. Such preferences can be expressed by means of rules of the form `:~ solutionRoute(V,R), vehicle(V,1), cost(V,R,N). [N@2]`. In particular, this rule states that the overall sum of the routes selected for the vehicles must be minimised.

The detailed description of the encoding falls beyond the scope of this paper. Therefore, we direct the interested reader to a dedicated repository[1] for the full version of the encoding, and some examples of solutions.

D. Monitoring and Execution

This module is in charge of implementing the route(s) selected by the optimiser, and monitoring the traffic conditions of the controlled network. In this work, to test and assess the framework and the proposed implementation, we rely on SUMO [26], which is a state-of-the-art Urban Mobility Simulator.

This module is a crucial element of the architecture, as the routes have been selected using a high-level abstraction

[1] `https://github.com/matteocarde/ai-traffic`

of the considered network. For example, we leave to this component the job of simulating the flow of traffic light at intersections, cadenced by the switching phases of traffic lights. The overtakes among vehicles, the use of lanes, and the order in which vehicles are queued in traffic is another aspect overlooked in the previous phases. All these aspects can lead to discrepancies between expected and actual traffic conditions: They are provided as feedback to the search module, that takes them into account to define the current state of the network. Further, there may be significant disruptions that are modifying the viability of part of the network. For instance, a car accident can happen (they can be simulated in SUMO by modifying the behavior of drivers): This kind of events can reduce the capacity of roads, or completely block portions of the network, according to severity. This information is fed back to the Preprocessing, that needs to update its internal representation of the network and of the links.

In a sense, this module covers the more traditional roles of traffic authorities, enhanced with the ability to communicate with CAVs navigating the region.

III. Experimental Analysis

We assess the proposed approach using SUMO and by considering two scenarios: Bologna and Milton Keynes. The well-known Bologna [27] scenario considers the 7^{th} city in Italy by population ($400,000$ approx.). The considered area was constructed around the "Andrea Costa" road in Bologna, in which the football stadium is located. The network, represented in Fig. 2 (top), includes more than 110 junctions and more than 170 links. The total length of the modelled links is more than 33 kilometres. The scenario includes the demand for Bologna's peak hour (8am – 9am) in which $8,620$ vehicles roam the network. The interested reader is referred to [27] for a detailed description of how the network was constructed.

The Milton Keynes scenario considers the largest town in Buckinghamshire, United Kingdom, with a population of approximately $230,000$ inhabitants. A diagram of the considered network is shown in Fig. 2 (bottom): it covers an area of approximately 2.9 square kilometres. The network includes more than 25 junctions and more than 50 links. The total length of the modelled links is more than 45 kilometres. The model simulates the morning rush hour, and has been built by considering historical traffic data collected between 8am and 9am on non-holiday weekdays. Data has been provided by the Milton Keynes Council, and gathered by sensors distributed in the region between December 2015 and December 2016. Traffic signal control information has been provided by the Council. The model has been calibrated and validated. During the morning rush hour, $1,900$ vehicles are entering the controlled region, and the main traffic flows are from North to South-East, and from West to East. This is because large residential areas are located at the North and West of the modelled region.

Our approach uses the TraCI interface[2] [28] to interact with the SUMO simulation environment, in order to get the current network status, communicate with approaching vehicles, and inform vehicles of re-routing. In both scenarios, the simulation is run until all the vehicles left the network. For each set of experiments, the simulation is run five times and results are averaged. All the experiments were run on a MacBook Pro with a 2.5 GHz Intel Core i7 quad-core, with 16 GB of RAM.

[2] https://sumo.dlr.de/docs/TraCI.html

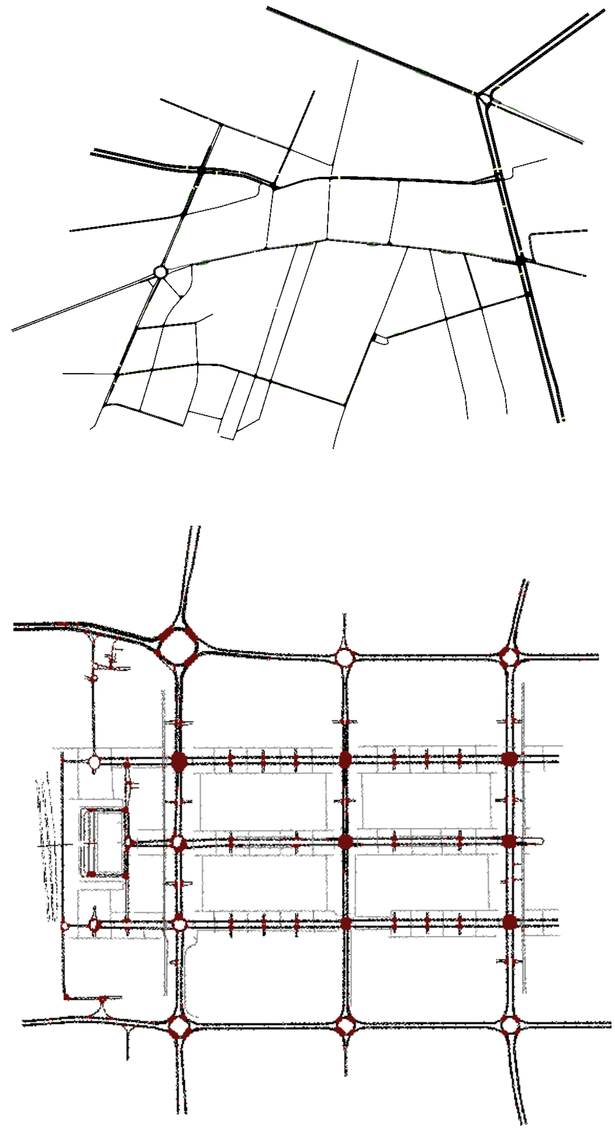

Fig. 2. The networks of the considered urban areas: Bologna, Italy (top) and Milton Keynes, United Kingdom (bottom).

To empirically assess the performance of the proposed framework, in this paragraph, we compare real traffic data of the Milton Keynes and Bologna urban areas with a simulation in which the same vehicles are routed using our proposed approach. Table I shows a comparison between the two approaches in terms of a number of traditionally considered KPIs, that focus on delay, waiting time, speed and path duration. As it can be seen from the comparison, in the Milton Keynes urban area, the proposed approach is able

TABLE I
PERFORMANCE OBSERVED WHEN THE PROPOSED APPROACH IS USED (OPTIMISED) OR NOT (ORIGINAL) IN SIMULATION WITH REAL-WORLD HISTORICAL TRAFFIC DATA FROM THE MILTON KEYNES AND BOLOGNA'S URBAN AREA.

	Milton Keynes		Bologna	
	Original	Optimised	Original	Optimised
Total Duration [s]	15.729	5.065	5.692	5.647
Avg. Distance [m]	2,465	2,107	1,636	1,678
Avg. Speed [m/s]	2.49	5.28	6.37	6.58
Avg. Duration [s]	3,718.95	515.82	283.93	281.13
Avg. Wait Time [s]	3,132.36	259.39	97.96	95.33
Avg. Dep Delay [s]	791.78	55.69	192.23	188.33

to greatly increase the overall performance of the network, spreading traffic and reducing congestion, increasing the average speed of vehicles and allowing the network to free faster. In Bologna, instead, the KPIs are only slightly increased with respect to the ones compute on real traffic data, but still showing how our proposed approach is able to capture all the nuances of urban traffic control and to deal with the risk of congesting the network. The high differences in improvement which can be seen in Milton Keys with respect to Bologna can be explained by the two very different topologies of the networks. As it can be easily seen, in Milton Keynes the streets form a sort of *Manhattan Grid* in which parallel streets are more or less equal in terms of number of lanes, length and intersections. For this reason, a car entering the network has a plethora of possible routes to choose, which are more or less of the same length, giving the possibility to better spread the traffic through the whole map. In Bologna, instead, it can be noted how streets in the outer ring are more structured to deal with traffic (with a high number of lanes and roundabouts to reduce traffic), while streets at the centre of the map (which constitute the residential area) have mostly one lane and a high number of intersections. For this reason, vehicles have a smaller number of promising routes to chose from since residential area's streets tend to be filled early, leaving no choice to the planner to let vehicles move through the streets of the outer ring.

IV. DISCUSSION

It is worth discussing some potential additional risks that could be taken into account by the instantiated framework, but were not included due to the lack of information in the considered scenarios. A number of different types of risks can potentially be dealt with in the Preprocessing step, by updating the structure and the information of the network accordingly. These include:

- Roads that cannot be considered during certain time frames (e.g., streets near an elementary school which becomes a no-traffic zone when children are entering or leaving the school, or in streets nearby a political/social/sport event which must be freed for security reasons).
- Streets that are not suitable to some kind of vehicles, such as streets in the city centre in which the circulation of polluting vehicles is forbidden, or bridges and underpasses which have a limitation on weight, height and width of vehicles.
- Paths that are not viable due to the hazardous nature of goods in vehicles, e.g., vehicles which contain inflammable liquids should avoid entering highly populated areas due to the risk of accidental explosions.

Additionally, the search phase could also be extended considering risks related to

- sequence of streets that can result in a significant increase in accidents potential (e.g, routes in which trucks or long vehicles perform tight turns which have a limited visibility and could risk injuring pedestrians or smaller vehicles). This can be done by modifying the heuristic function guiding the search, or by including hard constraints to be satisfied for a plan to be considered valid;
- paths in which a particular street is not present (e.g., routes which do not include a petrol pump for vehicles which have a small amount of fuel left). This can be suitable to be addressed also by tweaking the heuristic function accordingly.

Finally, the optimisation metric right now is mainly focused on the distribution of vehicles to avoid congested roads. It can be extended by considering also pollution, which is an important element of mobility in urban regions. Plans that allow to distribute pollutants away from residential areas can be preferred under some environmental circumstances, such as non-windy or non-rainy days. However, pollutants dispersion and emission models at the state of the art are computationally expensive, as they need to take into account a variety of chemicals, and cannot be easily incorporated in a system that aims at being deployed for real-time traffic control [29]. In a straightforward implementation, a pollution value can be assigned to each vehicle, and the emissions depend on such pollution value and on the distance travelled by the corresponding vehicle. This information can be taken into account by the search and the Clingo optimiser to, respectively, generate solutions by extending the heuristic search to take pollution into account, and by adding and/or updating rules accordingly in the respective Optimize step of the ASP encoding.

V. CONCLUSION

In this paper, we presented a framework for performing optimal and risk-aware vehicle routing in urban areas, based on Answer Set Programming. The proposed framework is capable of generating routing plans that account for various types of risks that may be present in an urban area, such as restricted areas, streets with limited capacity, and junctions with limited turning capabilities. We also discussed some potential additional risks that could be taken into account by the instantiated framework, such as restricted streets during certain time frames and unsuitable streets for certain types of vehicles.

The performed empirical analysis indicates that the framework, while being able of considering risks during routes generation, can significantly increase the overall performance of the network according to a range of considered KPIs. The obtained improvement on KPIs is of course a function of the structure of the network and of the potential risks that it may induce on traffic navigating through it.

We see several avenues for future work. First, we are interested in testing the framework on areas where more risk-related data are available. Second, we plan to enhance the framework to better consider the possibility of vehicles not following the provided instructions, hence improving the overall security of the approach. Finally, we are interested in incorporating traffic light optimisation into the framework, for instance by leveraging on existing AI-based work [30], to provide a more complete approach to urban traffic management and control.

REFERENCES

[1] S. E. Shladover, "Connected and automated vehicle systems: Introduction and overview," *Journal of Intelligent Transportation Systems*, vol. 22, no. 3, pp. 190–200, 2018.

[2] A. Nikitas, E. T. Njoya, and S. Dani, "Examining the myths of connected and autonomous vehicles: analysing the pathway to a driverless mobility paradigm," *International Journal of Automotive Technology and Management*, vol. 19, no. 1-2, pp. 10–30, 2019.

[3] T. Campisi, A. Severino, M. A. Al-Rashid, and G. Pau, "The development of the smart cities in the connected and autonomous vehicles (cavs) era: From mobility patterns to scaling in cities," *Infrastructures*, vol. 6, no. 7, p. 100, 2021.

[4] M. Vallati and L. Chrpa, "A principled analysis of the interrelation between vehicular communication and reasoning capabilities of autonomous vehicles," in *Proceedings of ITSC*, 2018, pp. 3761–3766.

[5] P. Kopelias, E. Demiridi, K. Vogiatzis, A. Skabardonis, and V. Zafiropoulou, "Connected & autonomous vehicles–environmental impacts–a review," *Science of the total environment*, vol. 712, 2020.

[6] A. Nikitas, N. Thomopoulos, and D. Milakis, "The environmental and resource dimensions of automated transport: A nexus for enabling vehicle automation to support sustainable urban mobility," *Annual Review of Environment and Resources*, vol. 46, pp. 167–192, 2021.

[7] M. Vallati, E. Scala, and L. Chrpa, "A hybrid automated planning approach for urban real-time routing of connected vehicles," in *Proceedings of ITSC*. IEEE, 2021, pp. 3821–3826.

[8] M. H. Mansourianfar, Z. Gu, S. T. Waller, and M. Saberi, "Joint routing and pricing control in congested mixed autonomy networks," *Transportation Research Part C: Emerging Technologies*, vol. 131, p. 103338, 2021.

[9] V. Najafi Moghaddam Gilani, S. M. Hosseinian, M. Ghasedi, and M. Nikookar, "Data-driven urban traffic accident analysis and prediction using logit and machine learning-based pattern recognition models," *Mathematical problems in engineering*, vol. 2021, pp. 1–11, 2021.

[10] Z. Zhou, "Attention based stack resnet for citywide traffic accident prediction," in *2019 20th IEEE International Conference on Mobile Data Management (MDM)*, 2019, pp. 369–370.

[11] C. Chen, X. Liu, H.-H. Chen, M. Li, and L. Zhao, "A rear-end collision risk evaluation and control scheme using a bayesian network model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 1, pp. 264–284, 2018.

[12] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," *arXiv preprint arXiv:1708.06374*, 2017.

[13] M. Khonji, J. Dias, R. Alyassi, F. Almaskari, and L. Seneviratne, "A risk-aware architecture for autonomous vehicle operation under uncertainty," in *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*, 2020, pp. 311–317.

[14] M. Gleirscher and S. Kugele, "Reaching Safe States in Autonomous Road Vehicles," in *Fast abstracts at International Conference on Computer Safety, Reliability, and Security (SAFECOMP)*, 2016.

[15] K. Mokhtari, A. Ayub, V. Surendran, and A. R. Wagner, "Pedestrian density based path recognition and risk prediction for autonomous vehicles," in *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2020, pp. 517–524.

[16] T. Nyberg, C. Pek, L. Dal Col, C. Norén, and J. Tumova, "Risk-aware motion planning for autonomous vehicles with safety specifications," in *2021 IEEE Intelligent Vehicles Symposium (IV)*, 2021, pp. 1016–1023.

[17] Y. Zhang, Y. Zhang, and R. Su, "Pedestrian-safety-aware traffic light control strategy for urban traffic congestion alleviation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 1, pp. 178–193, 2021.

[18] M. Gelfond and V. Lifschitz, "The stable model semantics for logic programming," in *Proceedings of the Fifth International Conference and Symposium , Seattle, Washington, August 15-19, 1988 (2 Volumes)*. MIT Press, 1988, pp. 1070–1080.

[19] ——, "Classical Negation in Logic Programs and Disjunctive Databases," *New Generation Comput.*, vol. 9, no. 3/4, pp. 365–386, 1991.

[20] I. Niemelä, "Logic Programs with Stable Model Semantics as a Constraint Programming Paradigm," *Ann. Math. Artif. Intell.*, vol. 25, no. 3-4, pp. 241–273, 1999.

[21] T. Eiter, A. Falkner, P. Schneider, and P. Schüller, "Asp-based signal plan adjustments for traffic flow optimization," in *ECAI 2020*, 2020, pp. 3026–3033.

[22] E. Erdem, M. Gelfond, and N. Leone, "Applications of answer set programming," *AI Magazine*, vol. 37, no. 3, pp. 53–68, 2016.

[23] M. Gebser, B. Kaufmann, and T. Schaub, "Conflict-driven answer set solving: From theory to practice," *Artificial Intelligence*, vol. 187, pp. 52–89, 2012.

[24] F. Calimeri, W. Faber, M. Gebser, G. Ianni, R. Kaminski, T. Krennwallner, N. Leone, M. Maratea, F. Ricca, and T. Schaub, "ASP-Core-2 input language format," *Theory and Practice of Logic Programming*, vol. 20, no. 2, pp. 294–309, 2020.

[25] G. Brewka, T. Eiter, and M. Truszczyński, "Answer set programming at a glance," *Communications of the ACM*, vol. 54, no. 12, p. 92–103, 2011.

[26] P. Á. López, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. WieBner, "Microscopic traffic simulation using SUMO," in *21st International Conference on Intelligent Transportation Systems, ITSC*. IEEE, 2018, pp. 2575–2582.

[27] L. Bieker, D. Krajzewicz, A. P. Morra, C. Michelacci, and F. Cartolano, "Traffic simulation for all: a real world traffic scenario from the city of bologna," in *SUMO 2014*, 2014.

[28] A. Wegener, M. Piórkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux, "Traci: an interface for coupling road traffic and network simulators," in *Proceedings of the 11th communications and networking simulation symposium*, 2008, pp. 155–163.

[29] H. Forehead and N. Huynh, "Review of modelling air pollution from traffic at street-level - the state of the science," *Environmental Pollution*, vol. 241, pp. 775–786, 2018.

[30] T. L. McCluskey and M. Vallati, "Embedding automated planning within urban traffic management operations," in *Proceedings of ICAPS*, 2017, pp. 391–399.