Simulation of a Two Lane Motorway to Analyze Vehicle behaviour Using Simpy

Heena Chopra
MSc in Data Analytics
National College of Ireland
Dublin, Ireland
x19205309@student.ncirl.ie

***Abstract*—In this study, our objective is to simulate the behaviour of traffic on a two lane motorway taking in consideration different real-world scenarios. We construct various different models for simulating the different scenarios that could potentially occur on a multi lane motorway. We built the different classes, each one responsible for handling different parts of the motorway construction like lanes, vehicles, etc. The classes and objects are constructed using the concepts of object oriented programming. Different simulation models were constructed by incorporating different parameters like interarrival time, number of cars, speed distribution function, etc. For each of the simulation models, four different quantities were computed using the pandas data frame of the class. These four quantities are average travelling time, throughput, average speed and the average density of the traffic. The results for each simulation were clearly reported and inferences were drawn from the position time plots of the simulations.**

I. Introduction

Traffic congestion has become a trouble for modern-day society. The number of vehicles on roads, motorways, and highways is increasing day by day that causes roughly one-third of all vehicular travel to happen under congested conditions. According to research, traffic congestion causes a delay of about six-tenths of a minute per kilometre of travel [1]. Traffic congestion results in more fuel consumption that adversely affects the environment. It has become very necessary to manage and reduce traffic congestion as the time spent on traffic congestion is miserably wasted. It causes the speed of vehicular travel to be half as that of free-flow travel. With the increasing number of vehicles on roads, traffic congestion leads to a vehicle crash. Amongst the number of factors giving rise to crashing, 94% of accidents have occurred due to drivers' mistakes such as cognitive errors, mistakes in decision making on roads, and improper operation [2].

Studies suggest that the manoeuvre of all the vehicles on road are managed by the "Law of Separation" which states that each vehicle moving should maintain a prescribed following distance from its preceding vehicle. To remain at a safe distance from its nearby vehicle, one should maintain a distance of a car length i.e. around fifteen feet for every ten miles per hour of travelling [3]. Thousands of tests have proven that the human nervous attributes and distractions confronted while driving are the major grounds for accidents happening around. It takes a whole of a second for a driver to see that the person ahead in the car has clamped on his brakes and for his brain to direct him to move his right foot from accelerator to brakes to stop the vehicle. Within this short period, the vehicle can't stop and however, the vehicle will carry on traveling without changing its speed. Therefore, it is very essential to maintain a particular distance from its preceding vehicles that are a minimum of one second's reaction to avoid any accidental situations.

The substantial increase in traffic growth on roads shows the need to have a better transporting system. Realizing this fact high-density corridors such as Multi-Lane highways have been introduced by the government. Even though it has enhanced the current highway networks but it needs serious thoughtfulness regarding the movement of traffic as various lane changing and overtaking movements may lead to severe accidents. Multi-Lane highways are a complex phenomenon to deal with and it is required to have a logical way to collect, analyse and interpret its data. As a solution to this, modeling and traffic flow simulation using an analytical and empirical approach are successful. Traffic Simulation models bring in the opportunity to estimate the traffic controls and design approaches without accomplishing a lot of expensive resources, money, and time.

The main objective of our study is to simulate a two-lane motorway with different values of the parameters like inter-arrival time, number of cars, speed variation distribution functions, etc. We will then calculate the throughput, average travelling time, average speed, and traffic density for each of the cases. The details of the model are highlighted in section IV.

Figure 1 shows the imperative lane changing scenarios where a driver demands to change the lane due to some imperious factors.

II. Literature Review

Extensive research has been done on traffic flow to understand its behaviour under interrupted and uninterrupted conditions. One of the interesting studies done by Srikanth et. al [4], in which they used one of the best traffic simulation software i.e., VISSIM. The model generated the traffic flow data to obtain all the important parameters. The VISSIM model was calibrated and validated based on field data. They analysed the traffic behaviour on various types of lanes divided by highways i.e., four lanes, six lanes, and eight lanes using the

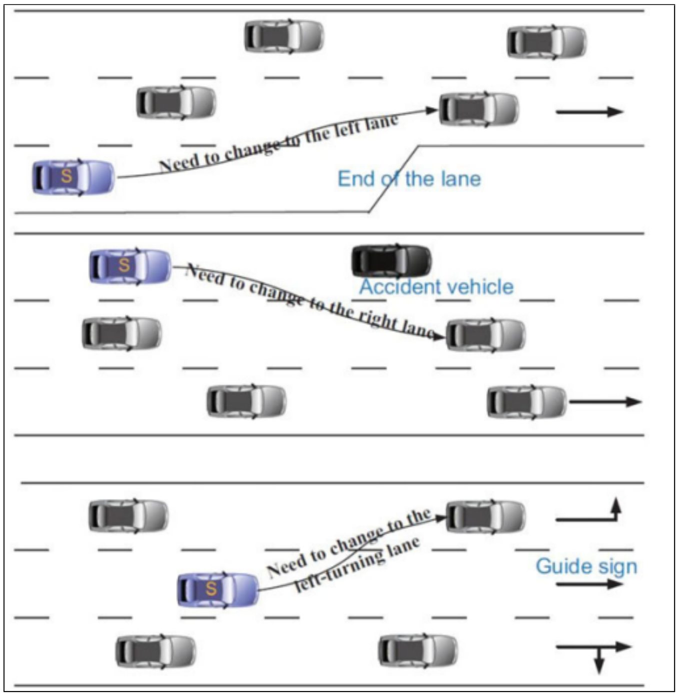

Fig. 1. Situations where a driver must change its lane, a) reaching the end of the current lane, (b) an accident vehicle appears in front in the current lane, and (c) a guidance sign appears at the road crossing

VISSIM model. They found that the number of lane changes is correlated with two major factors: 1). Traffic Volume 2). The number of lanes available for any particular direction of travel. The traffic volume is proportional to the number of lanes available by the third-degree polynomial. In another similar study conducted by Harb et. al [5], the VISSIM model was applied to simulate the two-to-one work zone lane closure configurations using the different Maintenance of Traffic (MOT) plans of the software. Dynamic lane merging systems (DLM) were introduced to enhance the safety of roadway zones. Early merge and late merge are the two types of lane merging techniques designed by them to advise and direct the drivers on merging locations. The three Maintenance of traffic (MOT) plans used are explained in brief below:

- The conventional plans were used in work zones of Florida.
- Simplified dynamic early merging system (early SDLMS).
- Simplified dynamic late merging systems (late SDLMS).

They observed that the early SDLMS outperformed the other two plans i.e., the late SDLMS and conventional MOT in terms of throughput and travel times. To deal with traffic congestion and traffic flow several traffic management algorithms have been brought into practice, one of the effective ones is Ramp Metering (RM). Ramp metering controls the rate of vehicles entering by installing traffic signals on all the slip roads. Al-Obaedi [6] developed a micro simulation model for motorway merge sections. The main objective of the research is to study the effectiveness of applying traffic management control systems and especially the Ramp Metering (RM) on systems. Their model was developed based on car-following, lane changing, and gap acceptance rules. Nearly 85% of drivers decide their merging movement within 50m of their lane and accept their first available gap after reaching the auxiliary lane.Zhou et. al [7] built a novel framework of multi-lane factors (MLFs) for bridge traffic loading that can be calibrated for any traffic stream. The framework established consists of a combination coefficient, a shared factor, and a lane correction coefficient.

Paulsen et. al [8] performed an interesting study in which a bicycle traffic simulation model is developed to manage the cyclist speed heterogeneity and its inherent interactions between the cyclists. The model built was validated against the data of the locations that have the possibility of having the highest number of bicycle trafficking. Microscopic traffic simulation models have become a very important tool for the analysis and management of traffic incidents such as accidents and vehicle crashes. P Hidas [9] introduced a massive simulation system i.e.; Simulation of Intelligent Transport Systems (SITRAS), in which the driver-vehicle objects are designed as independent agents. There are several applications of the outputs obtained from the Transport systems such as congestion, incident management, and route guidance. For developing the SITRAS model, lane changes and merging algorithms are used by the authors. These models developed two i.e. forced and cooperative methods for lane changing that are necessary to be considered under congested and traffic conditions. The results show that the forced and cooperative lane changing procedures turned out to give the best values during traffic-congested conditions. While driving, if the intention to change their lanes, they often indicate it through motion cues such as eye contact, lowering their speed, and adjusting their position. Bansal et. al [10] proposed a novel framework with mixed autonomy traffic-driving technique where a human-driven vehicle (HV) and an automated driven vehicle (AV) move together. They considered this approach on a two-lane traffic highway in which HV and AV drive off to merge into each other's lane. They optimize a collective reward and find that it can adapt to a human's behaviour. The methodology followed for simulating an autonomous driving scenario is shown in the figure 2

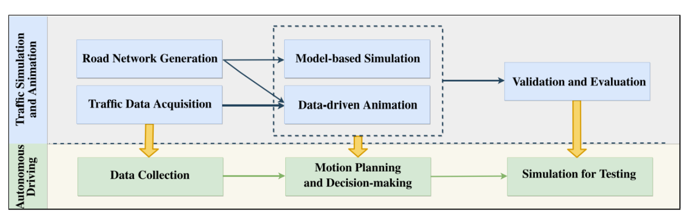

Fig. 2. Schema for traffic simulation of an autonomous driving scenario

Today's biggest struggle with the simulation and lane merging methods is to handle the real-time traffic situations as they are not scaled well with that complexity. However, this limitation can be abolished by learning all the behaviour offline using the data-driven approaches. This is when machine learning algorithms come into practice, to train the model's behaviour using the collected experiences. Hart et. al [11] proposed a policy-based reinforcement learning-based model framework for Lane merging scenarios. They combined local optimization and policy-based reinforcement learning to nurture the best of the two methodologies and make it relevant for all the safety-critical implementations. The model works in such a way that an initial solution is provided by the policy-based reinforcement learning algorithm that guides the optimizer for further optimizations.

Smog causes low visibility to drivers on the road and may lead to traffic safety issues. Many existing models available for traffic simulation consider that the driver has an optimal vision, but this is not correct in this scenario and therefore these simulation models are not successful for smog weather conditions. To overcome this limitation of the existing simulation models, Xu et. al [12] developed a Smog Full Velocity Difference Model (SMOG-FVDM) for better simulation of traffic in smog conditions. In this model, a stadia model for drivers is created in the smog conditions. They observed that the model resulted in an efficient and realistic traffic simulation in the smog weather conditions.

III. Methodology

The Methodology is a very crucial section for any research. It gives a well-defined and suitable path for the study by making it manageable, smooth, and effective. The main objective of our study is to simulate the stream of traffic on a two-lane motorway using simpy simulation software in python. The four basic Object-oriented programmings (OOPS) concepts encapsulation, abstraction, inheritance, and polymorphism are used in the framework of the study for better simulation. Various Classes and objects are created to incorporate all the elements and entities required for simulation.

- Lane Class: The first and foremost part of the methodology is Construction of the lanes for the motorway traffic in the lane class. This class defines all the enter and leave lane events of the vehicles. There are two ways for construction of Lanes:
  - Construction of one big lane by attaching the smaller lanes.
  - Manoeuvring a single long lane segment for the entire lane.

  There are two lanes in the motorway, the left lane, and the right lane. The left lane of the motorway is more frequently used by the drivers as compared to the right lane and therefore, the left lane is called as the ***'slow lane'*** and the right lane called as the ***'fast lane'***. Various functions have been applied in the code for merging lanes and segments, explained as below:
  - ***attachLeft()*** : To attach a parallel lane towards the left of the current lane.
  - ***attachRight()*** : To attach a parallel lane towards the right of the current lane.
  - ***widenLeft()***: To attach a lane segment towards the left of the current lane.
  - ***widenRight()*** : To attach a lane segment towards the right of the current lane.
- ***Vehicle Class***: The behaviour of the vehicles on the motorways is defined using Vehicle class in the code. In this class, all the possible behaviours of the vehicles are taken into consideration such as overtaking, crashing, and emergency braking. This class is mainly divided into two major parts:
  - Assigning initial values to all the vehicles like their initial position, acceleration, velocity, etc.
  - Updating the existing initial position, acceleration, velocity values of vehicles using Euler's integration method.

  Overtaking, one of the most complex yet important manoeuvre of the vehicles is controlled in this class of the code. Any vehicle is considered to perform overtaking only if a proper threshold distance is maintained with the vehicle in the front and back of the vehicle and there is no vehicle present in the neighbouring lane. We have defined an adjust velocity function in this class that allows the vehicle to adjust the velocity of the vehicle in such a way that, there is no crashing possibility of the vehicle with any other vehicle that is moving closer.
- ***Surround Class***: This class undertakes a view of the vehicles in place. It allows a vehicle to access another vehicle from the Top, Front, Back and Side edges of the vehicle.The different functions of the Class are explained as below (refer to figure 3:
  - ***leftLane()***: To refer to the next left lane present. It returns a None if no lane exists to the left.
  - ***rightLane()***: To refer to the next right lane present. It returns a None if no lane exists to the right.
  - ***Front***: To refer to an object that is present in the front of the current position.
    - ***leftFront()***: For any vehicle in the left front from the current position.
    - ***rightFront()***: For any vehicle in the right front from the current position.
  - ***Back***: To refer to an object that is present in the back of the current position.
    - ***leftBack()***: For any vehicle in the left Back from the current position.
    - ***rightBack()***: For any vehicle in the right Back from the current position.
- ***Recorder Class***: This class concerns recording all the Simulations. It records all the events that come about during the run time of the simulation process. We have created different frameworks in Python for all the sub-events that take place such as emergency braking, lane changing, crashing, any vehicle reaching the end of the simulation, etc.A Few of the possible events are explained as below:
  - ***Change Fast***: The event triggers when any vehicle in a slow lane overtakes another vehicle to move to the fast lane.

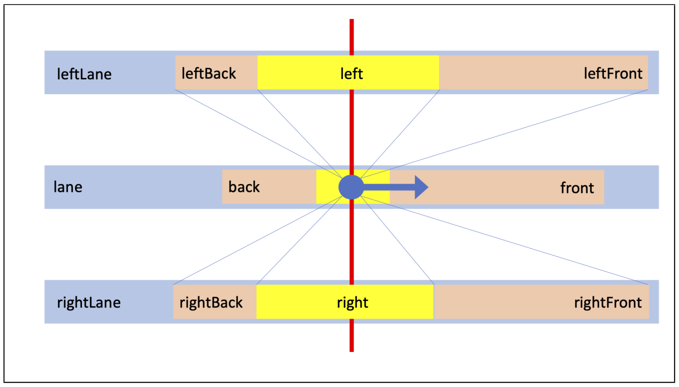

Fig. 3. Depiction of surround class

- *Change Slow*: The event triggers when any vehicle in a fast lane overtakes another vehicle to move to the slow lane.

Other events like Brake, Crash, End, etc. also happen during the run time of the simulation, that records the start and end of simulations due to emergency braking or any other event.

IV. Simulation Models

After defining and constructing all the classes, various simulation models are executed, each having its parameters and set of elements.

- **Model For Traffic Generation**: Four quantities are measured and analyzed for all the simulation models. These quantities give us insights into how our input parameters are affecting the simulated models.
  - ***Travelling Time (Average)***: The average travelling time of any vehicle is the time taken by the vehicle to reach the end of the motorway. In this study, we have calculated the average travelling time by finding the difference between the entering lane and the end lane events of the vehicles that have reached the end of the motorway successfully. Later, these times are averaged as per the number of vehicles considered.
  - ***Net Throughput***: Net throughput is the volume of the vehicles moving in one direction, that reach the end lane of the motorway successfully. The throughput for this study is calculated by finding the difference between the last vehicle exiting and the first vehicle entering the motorway.
  - ***Speed (Average)***: Average Speed of the vehicle is the ratio of the average length of the motorway and the average time of travelling of the vehicle.
  - ***Density Of Traffic (Average)***: The density of the traffic is defined as the ratio of the total volume of the vehicles travelling to the average speed of the lane. In other words, it is the ratio of net throughput and the average speed of the vehicle.

  Along with the above parameters, several other global constants were also passed as parameters in the vehicle class. A few of them are listed below:
  - ***Critical_Time_Tolerance***: This is the minimum time difference required between two vehicles to avoid the crash. This is fixed as 4 seconds for the model.
  - ***Lane_Change_Time***: This is the maximum time required by any vehicle to change a particular lane. This is fixed as 3 seconds for the model but in actual it is also dependent on the type of vehicle and the behaviour of its driver.
  - ***Min_Time_Diff***: This is the minimum time difference considered to trigger the overtaking behaviour of the model. This is taken about 1 second for the model.
  - ***Min_Speed_Diff***: This is the minimum speed difference considered to trigger the overtaking behaviour of the vehicle. This is taken about 2 m/s for the model.

  Our simulation study is divided into two major segments, first is the simulation of the two-lane motorway to calculate all the parameters and quantities measured in the previous section and the second is integrating the *freemotorwaySpeed()* function in the model to calculate the velocities of all the vehicles in the flow.
- **Model for Behaviour of Human Drivers:** Simulation of the behaviour of the human drivers is done using the below components:
  - **Uniform Distribution**: In this type of distribution, *random.uniform()* function of python is applied that returns a random floating-point number between any given range of functions. A distribution made out of these floating points number is termed Uniform Distribution.
  - **Expovariate Distribution**: In this type of distribution, *random.expovariate()* function of python is applied that returns a pseudo-random numbers between any given range of functions. Random Numbers generated through this function can be determined. A distribution made out of these random points number is termed Expovariate Distribution.

  In our study, Inter arrival time is calculated using both Uniform and Expovariate distribution. The second task for simulating the human driver behaviour is an extension of the first model. In this, we have used the *freemotorway speed()* function, as provided in the class in the traffic generation file that leads to the speed of the vehicles.

A. Simulation 1

In this Simulation model, we have used single value for computing Inter Arrival Time (IAT) of both lanes, which is calculated using *uniform()* random distribution.The parameters that we have used are enlisted below along with the values. The position-time graph is shown in figure 4. The values for the throughput, average travelling time, average speed and traffic density is shown in figure 5.

- Number of cars (N): 1000
- Inter Arrival Time (IAT): 8 seconds
- Distribution: Uniform
- Range: IAT/10 to IAT+10
- Interpretation: Use of Uniform distribution has displayed two of the most important facts. First,There is a very less variation in the inter-arrival time. Second, the inter-arrival time is very low, resulting in maximum throughput. The low inter-arrival time shows that maximum of the vehicles reach the end of the simulation. The graphs also depicts that there are very less number of crashes in the simulation model.

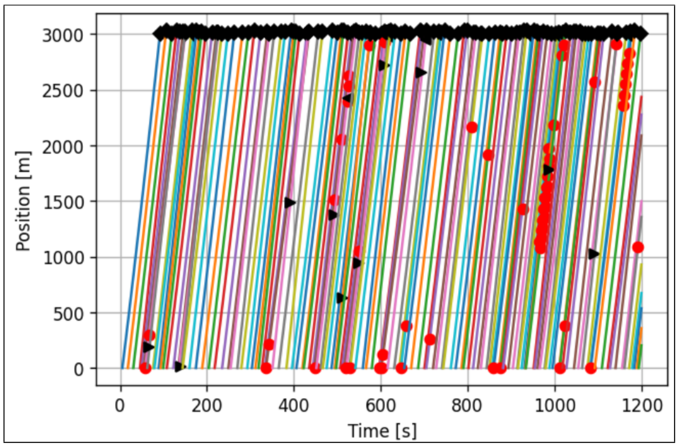

Fig. 4. Position time graph for simulation 1

```
Average travelling time 89.85564318600467
Average Speed 33.38688471451797 m/s
Throughput 400.3616636528029
Traffic Density:  11.991584931513826
```

Fig. 5. Values of different outputs for simulation 1

B. Simulation 2

In this Simulation model, we have used single value for computing Inter Arrival Time (IAT) of both lanes, which is calculated using *expovariate()* random distribution.The parameters that we have used are enlisted below along with the values.The position-time graph is shown in figure 6. The values for the throughput, average travelling time, average speed and traffic density is shown in figure 7.

- Number of cars (N): 1000
- Inter Arrival Time (IAT): 13 seconds
- Distribution: expovariate(1/IAT)
- Interpretation: The expovariate distribution causes the values of inter arrival time to fluctuate very rapidly. The smallest value for interarrival time can be found to be as low as 0.005. This causes crashes in the simulation because there is not enough time for the vehicle for overtaking and move onto the other lane. The average travelling time was found out to be around 90 seconds while the average speed comes out to be around 33.16 metres per second.

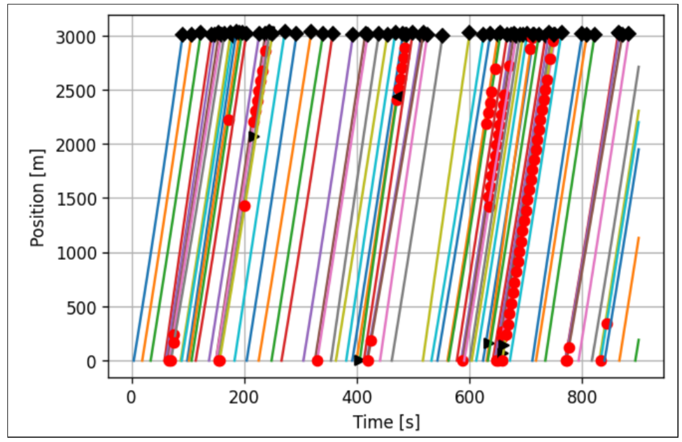

Fig. 6. Position time graph for simulation 2

```
average travelling time 90.45207155970675
average speed: 33.166736242405705 m/s
Throughput 304.4268125650395
Traffic Density:  9.178678611608795
```

Fig. 7. Values of different outputs for simulation 2

C. Simulation 3

In this Simulation model, we have used single value for computing Inter Arrival Time (IAT), which is calculated using uniform random distribution. The *freemotorway speed()* function is used for generating the velocity of the vehicles. The parameters that we have used are enlisted below along with the values. The position-time graph is shown in figure 8. The values for the throughput, average travelling time, average speed and traffic density is shown in figure 9

- Number of cars (N): 1000
- Inter Arrival Time (IAT): 10 seconds
- Distribution: uniform
- Speed function: *freemotorwaySpeed()*
- Range: IAT/10 to IAT+10
- Interpretation: This simulation makes use of the freemotorwaySpeed function to calculate the speeds of the vehicles. The average travelling time attains the maximum value of around 100 seconds in this scenario. The values of throughput came out to be 356 cars/hour. the freemotorwayspeed functions calculates the speeds of the vehicles using a gaussian distribution which is a type of normal distribution.

V. Conclusion

In this study, we have simulated various aspects of a two-lane motorway. We constructed three different simulation

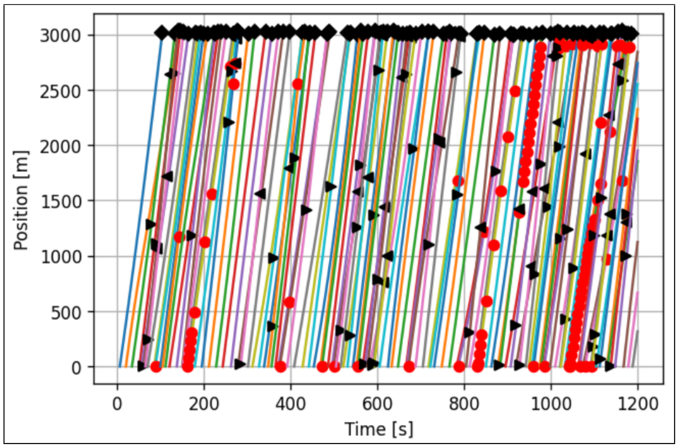

Fig. 8. Position time graph for simulation 3

```
average travelling time 100.01543262863237
average speed: 29.99537092579812 m/s
Throughput 356.3367252543941
Traffic Density:  11.879723912596111
```

Fig. 9. Values of different outputs for simulation 3

models taking different values of parameters like inter-arrival time and speed distribution. The values of throughput, average travelling time, average speed, and traffic density were computed for each simulation. The first simulation model consisting of a uniform speed distribution function attains the maximum value of 400 cars per hour for throughput and the average velocity of 33.38 m/s. The maximum average travelling time of 100 seconds was reported for simulation 3 in which the *freemotorwayspeed()* function was used for generating the speed data for the vehicles. The average traffic density remains more or less constant for each simulation.

References

[1] R. Arnott and K. Small, "The economics of traffic congestion," *American scientist*, vol. 82, no. 5, pp. 446–455, 1994.

[2] S. Singh, "Critical reasons for crashes investigated in the national motor vehicle crash causation survey," Tech. Rep., 2015.

[3] L. A. Pipes, "An operational analysis of traffic dynamics," *Journal of applied physics*, vol. 24, no. 3, pp. 274–281, 1953.

[4] S. Srikanth, A. Mehar, and K. G. N. V. Praveen, "Simulation of traffic flow to analyze lane changes on multi-lane highways under non-lane discipline," *Periodica Polytechnica Transportation Engineering*, vol. 48, no. 2, pp. 109–116, 2020.

[5] R. Harb, E. Radwan, and V. V. Dixit, "Comparing three lane merging schemes for short-term work zones: A simulation study," *International Scholarly Research Notices*, vol. 2012, 2012.

[6] J. Al-Obaedi *et al.*, "Development of traffic micro-simulation model for motorway merges with ramp metering," Ph.D. dissertation, Salford: University of Salford, 2011.

[7] J. Zhou, X. Shi, C. C. Caprani, and X. Ruan, "Multi-lane factor for bridge traffic load from extreme events of coincident lane load effects," *Structural Safety*, vol. 72, pp. 17–29, 2018.

[8] M. Paulsen, T. K. Rasmussen, and O. A. Nielsen, "Fast or forced to follow: A speed heterogeneous approach to congested multi-lane bicycle traffic simulation," *Transportation research part B: methodological*, vol. 127, pp. 72–98, 2019.

[9] P. Hidas, "Modelling lane changing and merging in microscopic traffic simulation," *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 5-6, pp. 351–371, 2002.

[10] S. Bansal, A. Cosgun, A. Nakhaei, and K. Fujimura, "Collaborative planning for mixed-autonomy lane merging," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 4449–4455.

[11] P. Hart, L. Rychly, and A. Knoll, "Lane-merging using policy-based reinforcement learning and post-optimization," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 3176–3181.

[12] M. Xu, H. Wang, S. Chu, Y. Gan, X. Jiang, Y. Li, and B. Zhou, "Traffic simulation and visual verification in smog," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 1, pp. 1–17, 2018.