# Traffic Congestion Reduction and Accident Circumvention System via Incorporation of CAV and VANET

Mohsin Khan
*Department of Computer Science and Information Technology, Central University of Jammu, Jammu and Kashmir, India*

Bhavna Arora
*Department of Computer Science and Information Technology, Central University of Jammu, Jammu and Kashmir, India*

## ABSTRACT

*Connected Automated Vehicle (CAV) technology is the core for the new age vehicles in research phase to communicate with one another and assimilation of Vehicular Ad-hoc Network (VANET) for the transference of data between vehicles at a quantified place and time. This manuscript is an enactment of the algorithms associated to the maintenance of secure distance amongst vehicles, lane shifting and overtaking which will diminish the occurrence of collisions and congestions especially phantom jams. Those implementations are centered over CAV and VANET technology for the interconnection of the vehicles and the data transmission. The data is associated to the aspects of a vehicle such as speed, position, acceleration and acknowledgements which acts as the fundamentals for the computation of variables. In accordance with the environment of a particular vehicle i.e. its surrounding vehicles, real-time decisions are taken based on the real-time computation of the variables in a discrete system.*

Keywords: VANET, CAV Technology, Lane Shifting, Overtaking, Phantom Jams, Discrete System

## INTRODUCTION

The Connected Vehicle Technology which is also known as Vehicle Infrastructure Integration assimilates different technologies on-board like GPS navigation, auto-mobile sensors, etc. to induce ability in a vehicle for the detection of any threat or risks on the roadway and interconnect over wireless network for the transmission of that information which includes permissions, forewarns and threats. In this evolving world, the transmission of this critical data can be attained using IoT or in precise terms by the IoV (Internet of Vehicle) which is a subset of IoT (Mahmood, 2020). There are some prominent categories of IoT wireless technology which includes LPWAN, Cellular, Zigbee, BLE, WIFI, RFID (Al-Sarawi et al., 2017), and for the purpose of CAV technology cellular and WIFI can be used. As the power consumption of cellular network (3G/4G/5G) is too high (Arnold et al., 2010), thus not sustainable for the majority of IoT applications which are powered by battery-operated sensor networks but they are appropriate for CAV technology.

It is essential that the transmission of data between the vehicles need to be in real time which can be attained using IoV (Gerla et al., 2014), which is a distributed network that upkeeps the use of data generated by connected cars and Vehicle Adhoc Networks (VANETs) (Golestan et al., 2016). VANETs are the subclass of Mobile adhoc networks (MANETs) with major differences of mobile units being as vehicles and other several reasons such as network topology, mobility patterns, demographics, traffic patterns at different times of the day (Tonguz et al., 2007). For the vehicle units to exchange data in real time, the network layers such as Physical layer, MAC layer and Application layer is developed suitably.

The automation of vehicle is classified in 5 categories: Level 1-Driver Assisted; Level 2-Partial Automation; Level 3- Conditional Automation; Level 4-High Automation and Level 5-Full Automation (Harner, 2020). When CAV technology is included in the automaticity, the algorithms are crafted in such a way that a specific vehicle takes critical decisions in accordance with the surrounding vehicles. Hence, reducing the chances of traffic congestion and accidents. The reduction of congestions, especially Phantom Jams which occurs by abrupt braking of a vehicle, starts a chain reaction and turns into an unexplainable traffic jam (Won et al., 2016). A phantom jam commences when a vehicle reduces its speed even marginally, which causes the vehicles behind that vehicle to slow down the speed even more and the action of reducing speed blowouts backward through the lane of traffic like a wave (Kowszun, 2013). The effect is directly proportional to the length of the traffic lane. This scenario usually happens when a common person is unable to understand the reason behind the traffic jam taking in consideration the fact that there are no accidents or wreckage of any vehicle on the road and no extreme weather conditions.

## BACKGROUND AND PURPOSE OF RESEARCH

The main reason for this research study is the accidents caused due to unguided overtaking and lane shifting. Human errors regarding to decisions are among the prominent causes of frontal collision accidents. The factors affecting the driver during the overtaking manoeuvres is essential. The research conducted by Figueira & Larocca regarding traffic psychology and behaviour (2020) have concluded that the effect of the speed of an impeding vehicle is greater as compared to the speed of the vehicle that has to overtake or the passing sight distance. Thus, increasing the chances of accidents related to overtaking and lane changing. Not only accidents but traffic congestion has continued to upsurge over the past decade. According to traffic index ranking by TomTom, 239 of the cities are reporting increased levels of congestion between 2018 and 2019. According to study 71 percent extra travel time is experienced in Bengaluru in India and Manila in the Philippines which recorded the joint-highest congestion level. The traffic congestion does not only waste the time but also has a psychological effect on the drivers. The capital of Columbia i.e. Bogota is at third position in traffic congestion estimating 68 percent congestion while as another Indian city i.e. Mumbai, comes fourth with 65 percent. Word's top 10 list of gridlocked cities comprises of four cities of India, only. In Russia, its Moscow with the densest traffic, where travel time is increased by 59 percent due to traffic congestion. Los Angeles in USA has the nastiest congestion levels but yet the travel time is only increased by 42 percent much less than the cities described in this section.

The enhancement in understanding the characteristics of traffic flow has been increased with the surfacing of Connected and Automated Vehicle (CAV) technology. Numerous studies have been examined for traffic analysis on the basis of CAV technology such as bottleneck capacity at highways, macroscopic traffic flow, optimization of paths for individual vehicles, and routing of data (van Arem et al., 2016). The prediction about vehicles consisting of CAV technology taking over the roads in the upcoming years is true but a certain amount of research is still in progress and new scientific researches are coming forth which must be fruitful for that prediction to be a fact. Some of those studies form the foundation of this research.

The first research by Li & Zhou (2017), is related to the real-time sharing of locations and accepting assistance from infrastructure, for a CAV's arrival and request for green light at intersections which can be approximately predicted along their routes. At the intersections where CAVs and traditional vehicles approach from different directions, an intersection automation policy (IAP) is developed for capturing complicated traffic dynamics and scheduling the resources such as green light for both CAV and traditional vehicles. IAP optimization is devised first, using a mixed integer linear programming formulation to characterize heterogeneous vehicle movements and dynamic signal timing plans. Then, a phase-time-traffic (PTR) hypernetwork model is developed to represent heterogeneous traffic proliferation under traffic signal operations. Sequential branch-and-bound search algorithms are also developed for considering both CAVs and traditional vehicles in the PTR hypernetwork. For the reduction of search space and finding the exact optimum efficiently special dominance, branch-and-bound search and bounding rules are also developed which is critical to the process.

The second research taken in consideration performed by Ghiasi et al (2017), projects the preparation of satisfactory highway capacity for the vehicles in which some are CAVs and the remaining are human driven. Centered over Markov chain representation of spatial distribution of heterogeneous and stochastic headways, an analytical capacity model is formulated for highway mixed traffic. This analytical model allows an examination for the intermingling of CAV and human driven vehicles. Whether to increase or decrease the penetration rate of CAV in the market, a sufficient and compulsory conditions for the mixed traffic capacity, is provided in this study. This analytical framework assists to construct a well-ordered lane management model to proficiently conclude the optimal number of CAV lanes to increase mixed traffic throughput over a highway. A theoretical perspective in taken in consideration for the model structure to be examined and an analytical approach is recognized to resolve the optimal CAV lane number at certain common headway. This approach can be useful in future for the CAV for the decisions related to lane management and corresponding to optimal capacity.

The third research approach is related to POLARIS which comprises of studying the potential effects of several CAV technologies at the regional level (Auld et al., 2018). CAV technology can give a specific insight related to vehicles not only in transportation system but also at an individual level. Many CAV technologies increase the network throughput and efficiency such as Adaptive Cruise Control (ACC) but it also has a secondary positive effect which is reducing the burden on driver and eradicating the human errors. Few research studies have examined the latent effects on CAV technologies from a systems viewpoint. However, data related to traffic flow comprise a complicated, adaptive and nonlinear system. Thus, an advanced simulation model, POLARIS is used in this study. POLARIS includes a 3d graphics engine designed for simulating automated vehicles at regional level with the benefits of Intelligent Transportation System (ITS) (Auld et al., 2016). It can also be used to obtain the energy impact of vehicles over a small neighborhood or over a greater area.

## RESEARCH METHODOLOGY

**Phase One:** Identification of objectives

The reduction of traffic jams particularly phantom jams and minimization of the accidents can be accomplished by summoning up those three objectives:

- Maintaining a safe distance.
- Acknowledged overtaking.
- Acknowledged lane shifting.

**Phase Two:** Choosing a platform for achieving the desired objectives.

For the accomplishment of those three objectives R2019b MATLAB and Simulink has been used. According to the official website of the Mathworks, MATLAB combines a desktop environment tuned for iterative analysis and design processes with a programming language that expresses matrix and array mathematics directly while as Simulink is an add-on product to MATLAB which provides an interactive, graphical environment for modeling, simulating, and analysing of dynamic system.

**Phase Three:** Integration of components in the selected platform.

In MATLAB R2019b, different components which includes WLAN system, communications system and DSP system are integrated for the development purpose. The development also consists of incorporation of VANET toolbox for the purpose of communication of the vehicles with each other. The VANET consists of network layers such as Application layer, MAC layer and Physical layer. While the MAC layer and physical layer is used for the communication purpose, the application layer consists of the functionality of the system i.e. application of an individual component and in this case, which is a vehicle.

**Phase Four:** Formulation of algorithm and conversion of algorithm into code.

The algorithm is devised in this phase, which is specified in the next section of this article, for the procurement of the objectives. The algorithm is devised according to the MATLAB discrete event simulation system and the various components integrated in third phase. The algorithm is being developed for the Application layer of the VANET. After development of the algorithm, it is converted using code using MATLAB programming language into a discrete model.

**Phase Five:** Simulating the scenario and recording results.

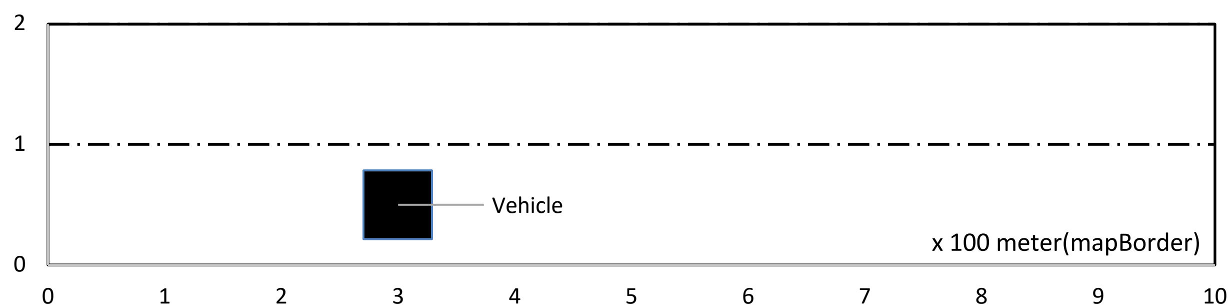
The developed discrete model is simulated using Simulink and the results are recorded in the result section providing the proof of the objectives devised in first phase. The results are recorded and converted into visual format i.e. graphs and charts. The values such as lane, speed and position in the graphs are constructed with respect to the simulated time.

## APPLICATION LAYER ALGORITHM

### Algorithm Prerequisites

The algorithm generates the results on the basis of the input provided. The inputs in case of a vehicle are its speed, position, acceleration but those input needs to be provided in a sequence or in a particular order and at a precise time. If the order or timely presence of that data is not acquired by the algorithm then the results or the decisions of the algorithm would be worthless. For a particular sequence of data input to the algorithm, the vehicle is inclined to select its environment in two dimensions with a boundary.

*Figure 1. X-Y (Lane-Roadlength) axis of a Vehicle*




1. **Basic Service Message (BSM):** BSM is generated by a vehicle at every instant and as a discrete event system, it is generated after every 0.1 seconds. It consists of the following variables:
   - Vehicle Id.
   - Position X.
   - Position Y.
   - Lane Number.
   - Speed X.
   - Speed Y.
   - Acceleration.

2. **LCM (Lane Changing Message):** LCM is generated in the case when vehicle is commanded and meets a certain criterion to change the lane and consists of the following variables:
   - Vehicle Id.
   - Position X
   - Speed X.
   - Acceleration.

3. **ackLCM (Acknowledgement Lane Changing Message**): ackLCM is generated when the permission is granted to the vehicle that generated the LCM. This message generates 1, if and only

if the vehicle is not overtaking or lane changing or it is not ready to overtake or change lane. It consists of only 2 digits 0 and 1. 1 means that the permission is granted and 0 means denied.

4. **Database:** A database is created for the storage of the data generated by vehicle and the data received from the other vehicles in its surrounding. The data stored in the database is accessed for calculation purpose of variable which results in the outcome or a specific decision.

## Algorithm

```
A zone up to which a vehicle receives BSM from other vehicles, mapBorder= 1000
Friction Coefficient, coEff= 0.8
Maximum deceleration, decelMax = -6.5
The delay in reaction period, reactionDelay = 0.67
Input consecutive lane number through which the car will overtake or to which
the car will change lane = conLane
Lane to be taken in consideration = curLane

Loop
In Database, following data related to vehicle is stored i.e.
vehicle ID= v_id
position X= pos_x
position Y= pos_y
lane number= lane
speed X= speed_x
speed Y= speed_y
acceleration = acc

The received BSM of a vehicle is also stored in database i.e.
Received Vehicle ID= recV_id
Received Position X= recPos_x
Received Position Y= recPos_y
Received Lane number= recLane
Received Speed X= recSpeed_x
Received Speed Y= recSpeed_y
Received Acceleration= recAcc
The value of the lane is stored as variable curLane i.e.
curLane =lane

Four different matrices are generated:
Same_lane_dist
Same_lane_info
Consec_lane_dist
Consec_lane_info

Four different arrays are generated:
Array_A []
Array_B []
Array_C []
Array_D []

If mapBorder< 1000 & recLane = lane

If recPos_x > pos_x
The received BSM vehicle is in the front.
```

Distance is calculated with that vehicle using equation,
calDist=((speed_x)/3.6+acc)*2+$pos_x$-$recPos_x$-(($recSpeed_x$)/3.6+recAcc)*2

If calDist < minimum element in Array_A
Value of recV_id is stored in matrix Same_lane_dist.
Values of recV_id, $recPos_x$, $recPos_y$, recLane, $recSpeed_x$, $recSpeed_y$, recAcc
are stored in matrix Same_lane_info.
End if

calDist is stored in an Array_A
Array_A = {Array_A, calDist}
FrontVehicleDist = Minimum element in Array_A
The vehicle with whom the least calDist is calculated will be the immediate Front vehicle.

Else $recPos_x$ < $pos_x$
The received BSM vehicle is at the posterior position.
calDist=($recSpeed_x$)/3.6*2+$recPos_x$-$Pos_x$-(($speed_x$)/3.6*2+0.5*acc*2^2)

If calDist < minimum element in Array_B
Value of recV_id is stored in matrix Same_lane_dist.
Values of recV_id, $recPos_x$, $recPos_y$, recLane, $recSpeed_x$, $recSpeed_y$, recAcc
are stored in matrix Same_lane_info.
End if

calDist is stored in an Array_B
Array_B = {Array_B, calDist}
BackVehicleDist = Minimum element in Array_B
The vehicle with whom the most least calDist is calculated will be the immediate posterior vehicle.
End If

If mapBorder < 1000 & recLane = conLane
If $recPos_x$ > $pos_x$
The received BSM vehicle is in the front but in next consecutive lane.
calDist=(($speed_x$)/3.6+acc)*2+$pos_x$-$recPos_x$-(($recSpeed_x$)/3.6+recAcc)*2

If calDist < Minimum element in Array_C
Value of recV_id is stored in matrix Consec_lane_dist.
Values of recV_id, $recPos_x$, $recPos_y$, recLane, $recSpeed_x$, $recSpeed_y$, recAcc are stored in matrix Consec_lane_info.
End if

calDist is stored in an Array_C
Array_C = {Array_C, calDist}
FrontVehicleDistNextLane = Minimum element in Array_C
The vehicle with whom the most least calDist is calculated will be the immediate Front vehicle in next lane.

Else recPos_x < pos_x
The received BSM vehicle is in the back but in next consecutive lane.
calDist=($recSpeed_x$)/3.6*2+$recPos_x$-$pos_x$-(($speed_x$)/3.6*2+0.5*acc*2^2 )

If calDist < Minimum element in Array_D
Value of recV_id is stored in matrix Consec_lane_dist.
Values of recV_id, $recPos_x$, $recPos_y$, recLane, $recSpeed_x$, $recSpeed_y$, recAcc are stored in matrix Consec_lane_info.

```
End if

calDist is stored in an Array_D
Array_D = {Array_D, calDist}
BackVehicleDistNextLane = Minimum element in Array_D
The vehicle with whom the most least calDist is calculated will be the
immediate posterior vehicle in next lane.
End if
End Loop

Case 1: Maintaining a safe distance
While FrontVehicleDistance < (speed_x/3.6+2*acc)*3.6
deceleration=(((revSpeed_x)/3.6)^2-((speed_x)/3.6)^2)/(2*FrontVehicleDistance)
if deceleration < ceil(-9.8*coEff)
acceleration = Max(deceleration, -9.8*coEff)
else
acceleration = Min(acceleration, deceleration)
End If
Acceleration = Min(acceleration, -1)
End While

Case 2: Overtaking
Distance_Braking = speed_x/3.6*reactionDelay+(speed_x/3.6)^2/(2*|decelMax|)
While FrontVehicleDistance < Distance_Braking
Send LCM to immediate Front vehicle.
If FrontVehicleDistanceNextLane > FrontVehicleDistance & ackLCM=1
speed_y = (4-mod(((pos_y)/4),2)*8)/2
End If
When lane = conLane
speed_y is decreased to 0.
End When
End While
Loop continuously executes and fresh data is taken in consideration.
If FrontVehicleDistance > 2 * FrontVehicleDistanceNextLane
Repeat acceleration at a specific value.
Until pos_x is greater than the position of the overtaking vehicle.
If posx > position of the overtaking vehicle
speed_y = (4-mod(((pos_y)/4),2)*8)/2
End If
When lane = curLane
speed_y is decreased to 0.
End When
End If

Case 3: Lane Changing
Distance_Braking = speed_x/3.6*reactionDelay+(speed_x/3.6)^2/(2*abs(decelMax))
While FrontVehicleDistance < Distance_Braking
Send LCM to immediate Front vehicle.
If FrontVehicleDistanceNextLane > FrontVehicleDistance & ackLCM=1
speed_y = (4-mod(((posy)/4),2)*8)/2
End If
End while
When lane = conLane
speed_y is decreased to 0.
End When
End While
```

## Algorithm Elucidation

The BSM is generated by a vehicle at every discrete time period and stored in the vehicle's database. The received BSM of other vehicle is also stored in the database which forms the basis of calculation. A loop is executed in which front vehicle and back vehicle is recognized at every discrete time period and distance is calculated with front vehicle at regular discrete time intervals.

The first step, is the comparison of the lanes, when BSM is received. If the lanes are similar then the received position X is compared with the position X of the vehicle. If the received Position X is greater, then the vehicle is in the front else, it is at the back. Yet, it is unknown whether it is at the immediate front or at the immediate back. So, distance is calculated with that vehicle and stored in an array. The distance is calculated in every iteration or for every BSM received. If the vehicle is at front, distance is calculated using equation:

$$calDist = \left(\frac{speed_x}{3.6} + acc\right) * 2 + pos_x - recPos_x - \left(\frac{recSpeed_x}{3.6} + recAcc\right) * 2$$

And if the vehicle is at back, distance is calculated using equation:

$$calDist = \frac{recSpeed_x}{3.6} * 2 + recPos_x - Pos_x - \left(\frac{speed_x}{3.6} * 2 + 0.5 * acc * 2^2\right)$$

The arrays are separate for the front vehicles and the back vehicles. The vehicle having minimum distance, which is obtained by comparing the values in those arrays, will be the immediate front and back vehicle, respectively. i.e. array for front vehicles contains the distances of the front vehicles only while as the array for back vehicles contains the distances of the back vehicles only but for the same lane. Thus, when the immediate front vehicle and back vehicle are recognized, the information which comprises of distance, vehicle id, position, speed, lane and acceleration regarding to the front vehicle and back vehicle are stored in matrices.

If the lanes are different but consecutive then the position X is compared with the position X of the vehicle whose BSM was received. If the received position X is greater, then the vehicle is in the front but in next lane and vice versa. Distances are calculated with the vehicle whose BSM was received and two separate arrays are generated. Same equations given above are used for the calculation of distances with front vehicles and back vehicles, respectively. Similarly, separate arrays are generated, one array consists of the distances with the front vehicle and other one contains with the back vehicle. From each array minimum distance is obtained which will be the immediate front and back car but in the next consecutive lane. Third condition can arise, in which received BSM consists of position X which will be same as that of the position X of vehicle in consideration. In this case, the distance will be zero and not included in any array. Thus, the information which comprises of vehicle id, position, speed, lane and acceleration regarding to immediate front vehicle and back vehicle in next lane are stored in matrices.

This loop continuously executes and depending upon the input, three cases arises:

**Case 1:** If only a safe distance needs to be maintained between the vehicles then the distance is acquired at regular intervals with the immediate front vehicle in the same lane. As the distance is measured at discrete time intervals in a loop with the vehicle at immediate front, it is compared with a threshold value and that threshold value is calculated using equation:

$$\left(\frac{speed_x}{3.6} + 2 * acc\right) * 3.6$$

If it is less than threshold value then the vehicle decelerates consistently until the distance is attained to be greater than the threshold value. The deceleration can be calculated using equation:

$$deceleration = \frac{\left(\frac{revSpeed_x}{3.6}\right)^2 - \left(\frac{speed_x}{3.6}\right)^2}{2 * (distance\ with\ front\ vehicle)}$$

**Case 2:** As the distance is obtained with the immediate front vehicle, lane of the vehicle is stored as Current lane. If the distance is less than threshold value called as brake distance, an LCM message is transmitted to the front vehicle by obtaining its vehicle id which is stored in the matrix. The brake distance is calculated using following equation:

$$Brake\ Distance = \frac{speed_x}{3.6} * reactionDelay + \frac{\left(\frac{speed_x}{3.6}\right)^2}{2 * |decelMax|}$$

In this equation Reaction Delay is used which is the interval of time between relative speed and acceleration and maximum deceleration is set. An ackLCM message is received by that vehicle which consists either 0 or 1. If it is 1 then the vehicle can overtake and if it is 0, then the vehicle is not permitted to overtake. 0 is transmitted if the vehicle at the front is overtaking or ready to overtake. As the value of every vehicle is updated at a regular time interval, position Y changes and lane of the vehicle is checked. When the lane of the vehicle is same as that of the input consecutive lane, the speed Y decreases to zero and the vehicle starts to accelerate if and only if the updated front vehicle has a distance greater than the double of previous distance with immediate front vehicle in former lane. The acceleration continues until the position X will be greater than that of the vehicle overtaken, which was at immediate front before changing the lane. When that position is achieved the vehicle again starts to change lane and stops changing lane when the value of Current Lane is achieved. Thus, completing the process of successful overtaking.

**Case 3:** This case of lane shifting is similar to that of overtaking up to a specific point. As distance with the front vehicle is updating at regular discrete time intervals. The distance is compared with the threshold value which in this case is also Brake Distance and as the threshold value is attained an LCM is transmitted to the front vehicle for the lane changing option. The front vehicle is recognized using the vehicle Id which is already stored in the matrix. If an ackLCM is received with a value of 1 then the lane changing starts. Speed Y is generated and the speed Y is set to be 0 when value when the updated lane of the vehicle is equal to the input concurrent lane which was set at the beginning of the lane changing. Thus, completing the process of acknowledged lane shifting.

## RESULTS

### Maintaining a safe distance

A scenario in which four vehicles are selected, which are present in same lane are jammed together. From the Table 1 it can be addressed that the vehicle with ID 1 has a speed of 100 km/h while as the vehicle having ID 2 has a speed of 70 km/h. Thus, there are chances of ramming into each other but the algorithm makes sure that a safe distance is maintained. The vehicle decelerates and reduces the speed in order to fulfil that criteria. The change of speed over the period of simulation time due to deceleration, maintaining a safe distance between vehicles has been illustrated by the graph in Figure 2 obtained as a result at the end of the simulation. The graph in Figure 3 illustrates the presence of those four vehicles in same lane.

*Table 1. Initial description of four vehicles for simulation of maintaining a safe distance*

| Vehicle ID | Initial Speed | Initial Position |
|---|---|---|
| 1 | 100 km/h | 110 |
| 2 | 70 km/h | 150 |
| 3 | 50 km/h | 190 |
| 4 | 40 km/h | 240 |

*Figure 2. Result of speed and position with respect to simulation time*

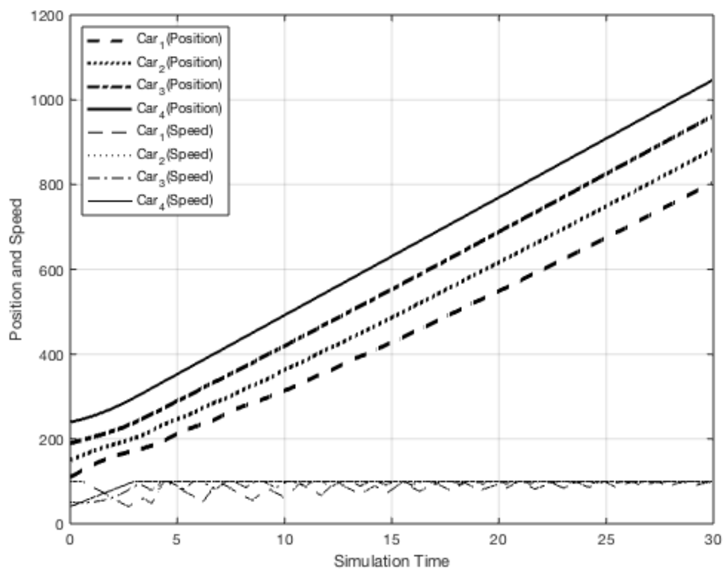
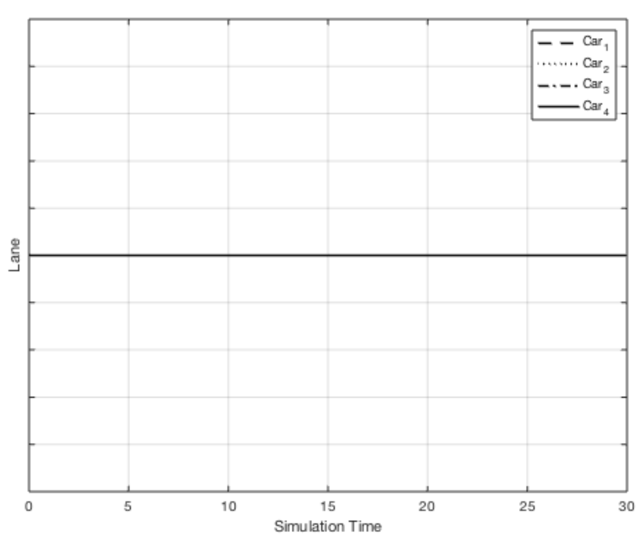



*Figure 3. Result of rate of changing of lane with respect to simulation time*



## Overtaking

The initial criteria or description of four vehicles are given in Table 2. In Figure 4 the crossing over line of Car 1 over Car 2 represents a successful acknowledged overtaking without any collision or congestion. As it can be seen from Figure 4 and Figure 5, Car 1 overtaking Car 2 and being successful while as after 5 second, Car 2 is only changing lane and being unsuccessful in overtaking as the conditions are not fulfilled.

*Table 2. Initial description of four vehicles for simulation of acknowledged overtaking*

| Vehicle ID | Initial Speed | Initial Position | Lane |
|---|---|---|---|
| 1 | 70 km/h | 110 | 1 |
| 2 | 40 km/h | 150 | 1 |
| 3 | 50 km/h | 60 | 2 |
| 4 | 50 km/h | 250 | 2 |

*Figure 4. Result of speed and position with respect to simulation time*

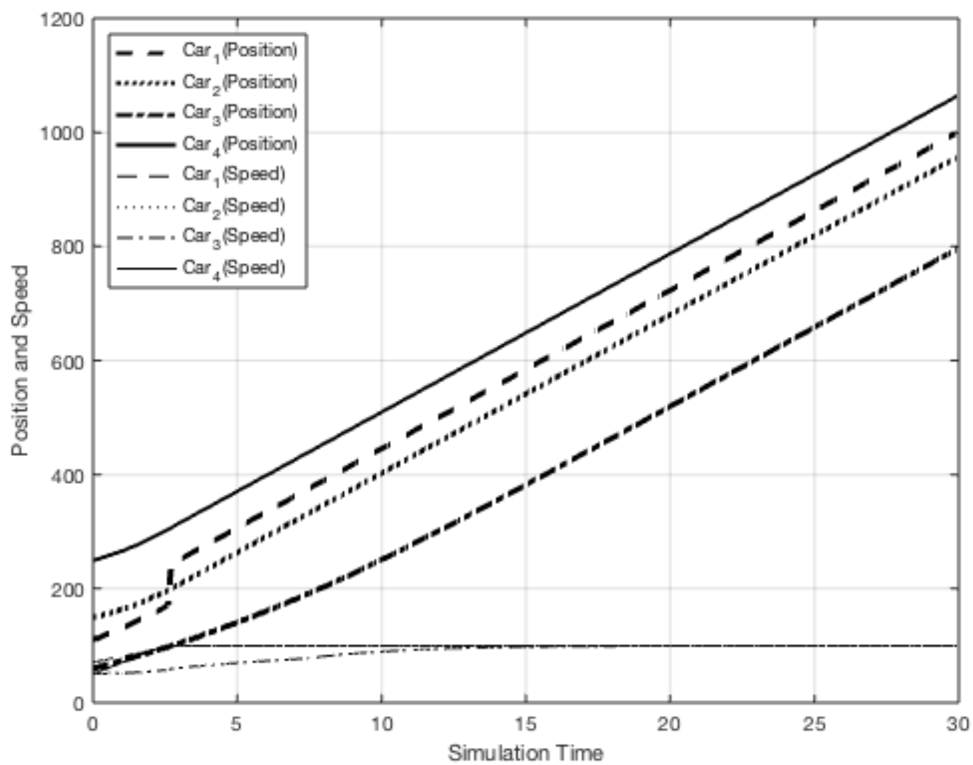
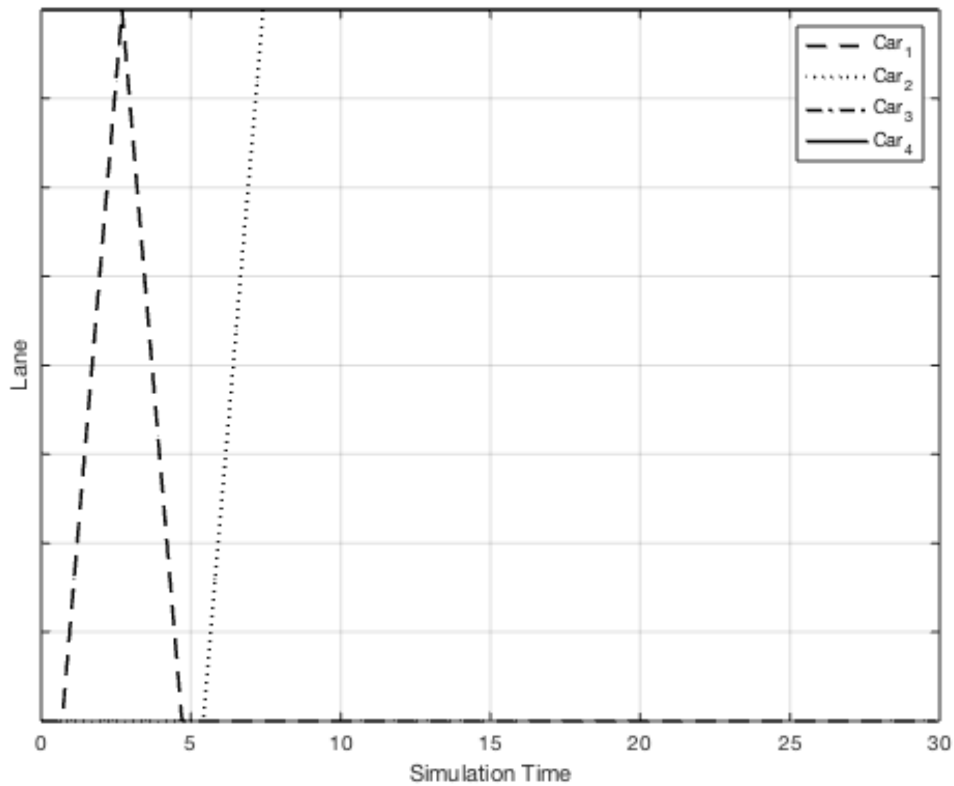


*Figure 5. Result of rate of changing of lane with respect to simulation time*




## Lane Shifting

The initial representation of the four vehicles are illustrated in Table 3. The algorithm works as expected as the vehicles change lane when the criterion is met. Figure 6 shows a smooth transition of position with respect to time and vehicles maintaining a distance with each other. Figure 7 depicts the time when the lane shifting is initiated by the vehicle and the flexibility of the dynamic system is demonstrated by the graph as more than one vehicle is changing lane at the same time.

*Table 3. Initial description of four vehicles for simulation of acknowledged lane shifting*

| **Vehicle ID** | **Initial Speed** | **Initial Position** | **Lane** |
| --- | --- | --- | --- |
| 1 | 80 km/h | 200 | 1 |
| 2 | 40 km/h | 240 | 1 |
| 3 | 100 km/h | 160 | 1 |
| 4 | 90 km/h | 100 | 2 |

*Figure 6. Result of speed and position with respect to simulation time*

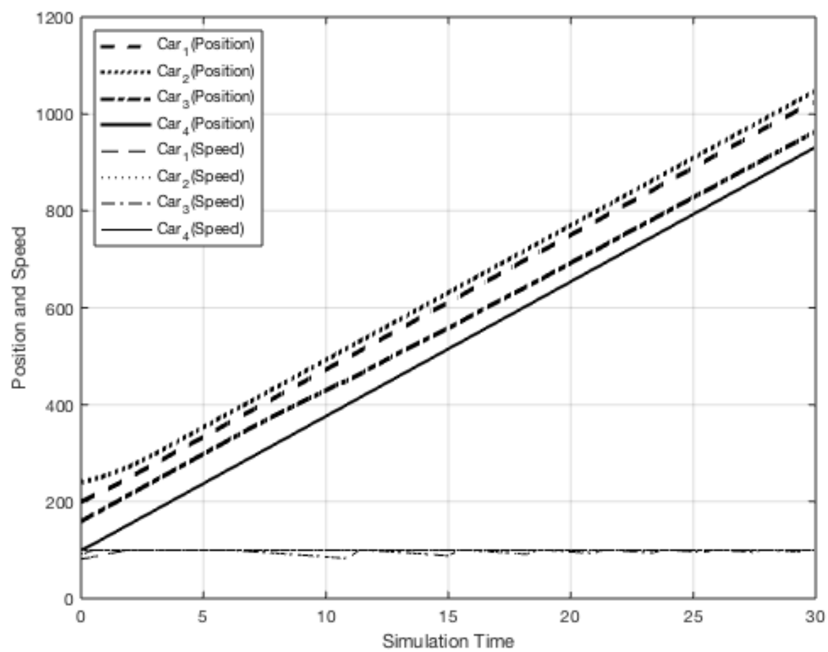
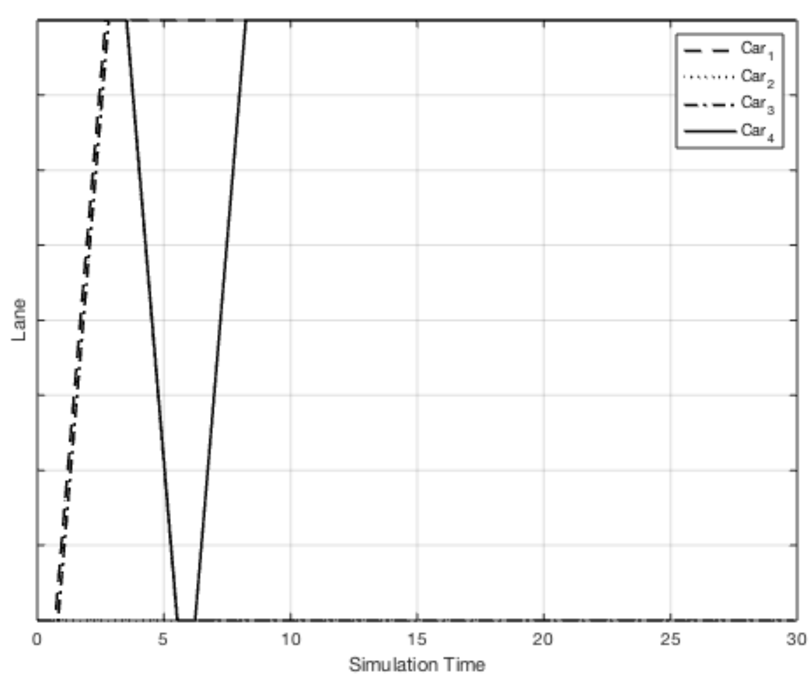



*Figure 7. Result of rate of change of lane with respect to simulation time*



## CONCLUSION AND FUTURE RECOMMENDATION FOR RESEARCH

The outcome of the implementation of the algorithm using MATLAB and Simulink in a discrete environment clearly minimizes the likelihood of congestions especially phantom jams and condense the accidents caused due to unacknowledged overtaking and lane changing. The graphs provided in the result section is a clear proof that a safe distance is conserved between vehicles, an acknowledged overtaking and calculated lane changing is accomplished as the vehicles are collaborating with each other for formulating a decision according to the algorithm. As the main cause of the phantom jam is abrupt braking or abrupt reduction in speed which is circumvented, as the vehicle consistently obtains the information regarding the speed it has to maintain at every discrete time point for maintaining a safe distance. Thus, slashing the probability of congestions specifically phantom jams. In case of vehicles overtaking or shifting lane which is attained by mutually deciding with the surrounding vehicles. As shown by the results, the vehicle considers the overtaking manoeuvre, if and only if the impeding vehicle (i.e. front vehicle) permits it and the impeding vehicle permits, if it is not overtaking or lane changing. Hence, the overtaking manoeuvre or lane shifting occurs while acknowledging from the impeding vehicle and on the basis of the information, regarding the vehicles in the next lane. This research focusses on the safe passage of vehicle during the process of overtaking as the passing sight distance of the overtaking vehicle is not always clear enough which either results a risky manoeuvre, congestion or worst case scenario, a collision.

This implementation has a real world application and needs to go through many phases of testing for applying in the real world. The testing phases needs to be performed in real world as it has passed the tests on a simulator which was the key purpose of this research conducted on a simulator. For passing the real the world environment, the algorithm represented in this paper could do some change according to different scenarios but the result achieved would be similar as that of acquired over the simulator which is presented in this paper. In real world new components such as sensors can be integrated in the algorithm for more accuracy of data and accordingly decisions, thus, not only minimizing congestions and accidents but a time will come when complete eradication of congestion and accidents would be possible. On the basis of simulation, this research can be further extended and furnished by integrating the sensors in the algorithm, increasing the capability of the vehicles.

## REFERENCES


Al-Sarawi, S., Anbar, M., Alieyan, K., & Alzubaidi, M. (2017, May). Internet of Things (IoT) communication protocols. In *2017 8th International conference on information technology (ICIT)* (pp. 685-690). IEEE.

Arnold, O., Richter, F., Fettweis, G., & Blume, O. (2010, June). Power consumption modeling of different base station types in heterogeneous cellular networks. In *2010 Future Network & Mobile Summit* (pp. 1-8). IEEE.

Auld, J., Hope, M., Ley, H., Sokolov, V., Xu, B., & Zhang, K. (2016). POLARIS: Agent-based modeling framework development and implementation for integrated travel demand and network and operations simulations. *Transportation Research Part C: Emerging Technologies, 64,* 101-116.

Auld, J., Verbas, O., Javanmardi, M., & Rousseau, A. (2018). Impact of privately-owned level 4 CAV technologies on travel demand and energy. *Procedia computer science, 130,* 914-919.

Figueira, A. C., & Larocca, A. P. C. (2020). Analysis of the factors influencing overtaking in two-lane highways: a driving simulator study. *Transportation research part F: traffic psychology and behaviour, 69,* 38-48.

Gerla, M., Lee, E. K., Pau, G., & Lee, U. (2014, March). Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. In *2014 IEEE world forum on internet of things (WF-IoT)* (pp. 241-246). IEEE.

Ghiasi, A., Hussain, O., Qian, Z. S., & Li, X. (2017). A mixed traffic capacity analysis and lane management model for connected automated vehicles: A Markov chain method. *Transportation Research Part B: Methodological, 106,* 266-292.

Golestan, K., Soua, R., Karray, F., & Kamel, M. S. (2016). Situation awareness within the context of connected cars: A comprehensive review and recent trends. *Information Fusion, 29,* 68-83.

Harner, I. (2020). *The 5 Autonomous Driving Levels Explained.* https://www.iotforall.com/5-autonomous-driving-levels-explained.

Kowszun, J. (2013). Jamitons: Phantom Traffic Jams. *School Science Review, 95*(350), 53-61.

Li, P. T., & Zhou, X. (2017). Recasting and optimizing intersection automation as a connected-and-automated-vehicle (CAV) scheduling problem: A sequential branch-and-bound search approach in phase-time-traffic hypernetwork. *Transportation Research Part B: Methodological, 105,* 479-506.

Mahmood, Z. (2020). Connected Vehicles in the IoV: Concepts, Technologies and Architectures. In *Connected Vehicles in the Internet of Things* (pp. 3-18). Springer, Cham.

Mathworks, (2020). *Simulink Documentation.* https://www.mathworks.com/help/simulink/

The MathWorks (2019b). *VANET Toolbox. Natick, Massachusetts, United States*. Retrieved from https://www.mathworks.com/matlabcentral/fileexchange/68437-vanet-toolbox-a-vehicular-network-simulator-based-on-des

Mathworks, (2020). *What is MATLAB.* https://www.mathworks.com/discovery/what-is-matlab.html

TomTom traffic index, (2019). *The world according to traffic.* https://www.tomtom.com/en_gb/traffic-index.

Tonguz, O., Wisitpongphan, N., Bait, F., Mudaliget, P., & Sadekart, V. (2007, May). Broadcasting in VANET. In *2007 mobile networking for vehicular environments* (pp. 7-12). IEEE.

Van Arem, B., Abbas, M. M., Li, X., Head, L., Zhou, X., Chen, D., ... & Orosz, G. (2016). Integrated traffic flow models and analysis for automated vehicles. In *Road Vehicle Automation 3* (pp. 249-258). Springer, Cham.

Won, M., Park, T., & Son, S. H. (2016). Toward mitigating phantom jam using vehicle-to-vehicle communication. *IEEE transactions on intelligent transportation systems, 18(5),* 1313-1324.