International Journal of Robotics and Control Systems
Vol. 5, No. 1, 2025, pp. 555-569
ISSN 2775-2658
http://pubs2.ascee.org/index.php/ijrcs

Autonomous Driving Model with Collision Prediction for Urban and Extra-Urban Environments

Yassine El Hafid [a,1], Tarik Ligabi [a,2], Yassine Zahraoui [a,3,*]

[a] LAGES, Hassania School of Public Works (EHTP), Casablanca 8108, Morocco
[1] el_hafid_yassine@ehtp.ac.ma; [2] ligabi.tarik.eleve@ehtp.ac.ma; [3] zahraoui.yassine@ehtp.ac.ma
[*] Corresponding Author

ARTICLE INFO

Article History
Received October 21, 2024
Revised November 30, 2024
Accepted January 23, 2025

Keywords
Autonomous Vehicle;
Collision Prediction;
Control Block;
Adaptive Cruise Control;
Urban Drive Modeling

ABSTRACT

This study introduces an architecture for an autonomous vehicle control system based on a collision detector and geometric modeling of trajectories. The goal is to develop a robust and reliable control model that can navigate metropolitan environments, often crowded with pedestrians and bicycles, as well as suburban areas, where traffic patterns can fluctuate. We have created a modular control unit that includes a collision predictor, which interacts closely with the decision module. The executed algorithm demonstrates the effectiveness of our system by ensuring the safety and comfort of the passengers. It can identify potential collisions from a distance and initiate braking preventively, following precise guidelines for deceleration and acceleration. To validate our methods, we are looking at simulations of realistic case studies. The research conducted underscores a crucial advancement in the development of safer and more flexible autonomous driving technologies.

1. Introduction

Recent studies reveal that road crashes impose a great health and monetary burden on public health. Road traffic injuries are one of the top ten causes of death worldwide, and the insurance cost of crashes for a country per year is a significant percentage of its gross national product. The aim is to reduce the predicted death rate away from the existing trend because of road traffic accidents, reducing around one-third in 2020 from 1.3 million to 0.91 million per year. Moreover, many corporations are moving towards fully integrated driverless cars that have either been developed or are still under construction and testing. Another driving force in the autonomous vehicles zone is safety, managing the ever-increasing crashes, road accidents, and untoward incidents worldwide, affected by economic gains. All these examples attest to the urgency and immense importance of collision avoidance for vehicles to dramatically affect the overall layout in terms of operator revenues and productivity as the driverless trend becomes more prevalent, potentially shutting down the automotive industry [1].

In the automotive crash scenario, control, localization, communication, vision, decision-making, environment modeling, path planning, dynamic obstacle prediction, collision detection, and finally the real-time collision-free trajectory estimation are various integral cutting-edge domains; the contribution herein revolves around collision detection and subsequent collision prediction systematically. Therefore, it is proposed to have a specialized section reserved to discuss the evolution and developmental paradigms developed for the collision avoidance system for automotive [2], [3]. Hence, assisted driving has further drawn the attention of researchers in two realization dimensions: semi-autonomous driverless vehicles and fully autonomous driverless technology, such that we aim primarily to have the first category of discussion in this essay [4], [5]. The inevitable rise in traffic load over the years has been triggering major incidents in terms of crashes, especially on roads or in the opposite lanes. It is this inner and understated dashboard of human nature, being largely unaware of one's immediate surroundings, that causes the greatest number of road incidents, accidents, or mishaps in developed countries. Such is the case with autopilot and traffic-aware cruise control in the case of commercial airplanes and self-sensing radar-guided missiles. With the technology and knowledge increment, this near certainty of continual collisions of human-driven vehicles can now be systematically reduced to injury-free available collision risk, thereby reducing the possibility of any such incidents reaching towards negation, with all cars in full automation reaching zero collision possibility [**?**], [7]. For the very purpose of such automated intelligent decision-making, special attention has to be provided to the local environment of the vehicle defined by a certain distance of operating radius, inbound for any anticipated collision, and as such integrating decision-making with the actual vehicle controls; such an enriched framework for collision prediction is essentially a multi-sensory approach [8], [9].

Collisions are a significant cause of death and injury around the world. Advanced driver assistance systems and partial or full vehicle automation have received substantial attention as means to reduce these incidents. Numerous automobile manufacturers have made substantial strides towards providing vehicles with these capabilities, enabled in part by the rapid reduction in the cost of the sensors and computation used by these systems. However, several incidents involving both partial and full vehicle autonomy have been publicized, indicating a need to develop methods for certification and validation of the technologies developed for these vehicles [10].

Numerous challenges must be resolved to align fully autonomous vehicle operation with human-in-the-loop operation [11], [12]. New sensors and actuators have been developed that can process data much more rapidly and make decisions in real time on the order of milliseconds. However, this generates significant data communication, perception, and prediction challenges for safe vehicle operation [13], [14]. The general public continues to be skeptical regarding the efficacy and safety of autonomous vehicles, and new legal frameworks will need to be developed to operate these vehicles in a cost-effective manner [15], [16]. The roles of various stakeholders in autonomous vehicle operation have been developed, and the safety of the vehicle hardware, software, and decision-making capabilities must be continually updated and validated [17], [18]. The majority of research to date has focused on trajectory estimation and prediction, collision detection, and vehicle control topics [19], [20].

The main objectives of this research are:

- Design of an Autonomous Vehicle Model: Develop a model of an autonomous vehicle based on interactive blocks. These blocks will be designed to adaptively simulate the human driving model, taking into account the variety and complexity of human behaviors while driving.
- Development of a Collision Estimator Block: Design a block specialized in estimating potential collisions. This block must be perfectly integrated and interact effectively with the decision logic system, allowing the vehicle to make safe and optimized decisions in real-time.

The contributions of this research are:

- Evolutionary Model of Human Behavior: Propose an evolutionary model capable of accurately simulating human behavior in terms of driving. This model must be sufficiently flexible to adapt to different driving scenarios and reflect the behavioral variations of drivers.
- Validation by Interaction with the Decision Block: Study and validate the effectiveness of the interaction between the different blocks, particularly the decision block. This validation is essential to ensuring that the proposed model functions are consistent and reliable.
- Trajectory and Collision Prediction Algorithm: Develop an advanced algorithm that supports dynamic trajectory calculation as well as collision prediction for various types of trajectories. This

approach will enable the vehicle to anticipate and avoid obstacles proactively.

The remainder of this paper is structured as follows: Section 1 presents an introduction to the subject with a full literature review. Section 2 details the proposed autonomous vehicle model, while Section 3 explores the control block properties. Section 4 discusses the adopted methodology for the calculation in the collision estimator, and Section 5 presents the calculation method in the collision estimator. Section 6 presents the simulation results and analysis. Finally, Section 7 concludes the paper with key findings and future research directions.

2. Proposed Autonomous Vehicle Model

The modeling of objects and vehicles is done in the form of a rectangle defined by a length L and a width W (Fig. 1) with a speed orientation vector and a straight or curved trajectory.

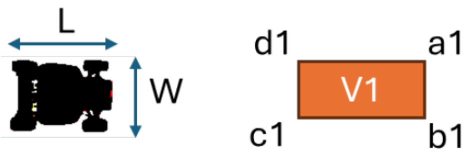

Fig. 1. Representation of the autonomous vehicle by a rectangle

The overall route is predefined at the beginning of the journey according to a well-known map. Autonomous vehicles can follow lines within the route according to speed instructions. The actions of changing stop or acceleration speed instructions are carried out according to the signage and the traffic code. The current version does not consider the latency times of the sensors or the reliability of the measurements. Furthermore, we have defined intrinsic parameters that present limitations for the autonomous vehicle:

- **Maximum braking capacity:** It represents the vehicle's deceleration capacity under optimal conditions. However, this parameter could vary depending on weather conditions, road conditions, or tire wear conditions. The deceleration capacity also includes the time for acquisition, processing, and decision-making [21].
- **Maximum acceleration capacity:** It defines the vehicle's ability to accelerate according to several parameters. Depending on the weight of the vehicle and the power of the engine. This parameter could be changed depending on the type of vehicle, as in the case of trucks. Indeed, the weight factor also influences the deceleration capacity mentioned in the previous point. The acceleration capacity also includes the time for acquisition, processing, and decision-making [22].
- **Maximum linear speed:** It is the maximum speed that a vehicle can reach while following a straight trajectory.
- **Comfort acceleration and deceleration:** They define the nominal acceleration and deceleration that allow for a comfortable ride for the driver and passengers. The acceleration and deceleration capacities include the time for acquisition, processing, and decision-making.
- **Minimum distance between vehicles:** It is the minimum safety distance between vehicles at rest or in motion. It's a dynamic parameter that can depend on the speed and weight of the vehicle, weather conditions, wheel adhesion, and traffic rules [23].
- **Detection distance or space:** It defines the distance or surrounding space at which the autonomous vehicle could make braking or acceleration decisions [24].
- **Maximum wheel rotation angle:** it is the maximum turning angle of the front wheels of the vehicle.
- **Maximum centrifugal acceleration:** it is a parameter dependent on the vehicle's speed and the angle of the front wheels. It represents a crucial factor in vehicle safety, especially for maintaining sufficient grip without loss of control [25].

- **Centrifugal comfort acceleration:** it is the nominal acceleration permissible for passenger comfort.

3. Control Block

Fig. 2 shows the proposed control block for a scaled-down model of an autonomous vehicle with three actuators. We will focus on the “Object Collision Prediction Block” and its communication with the “Logic Decision Block”.

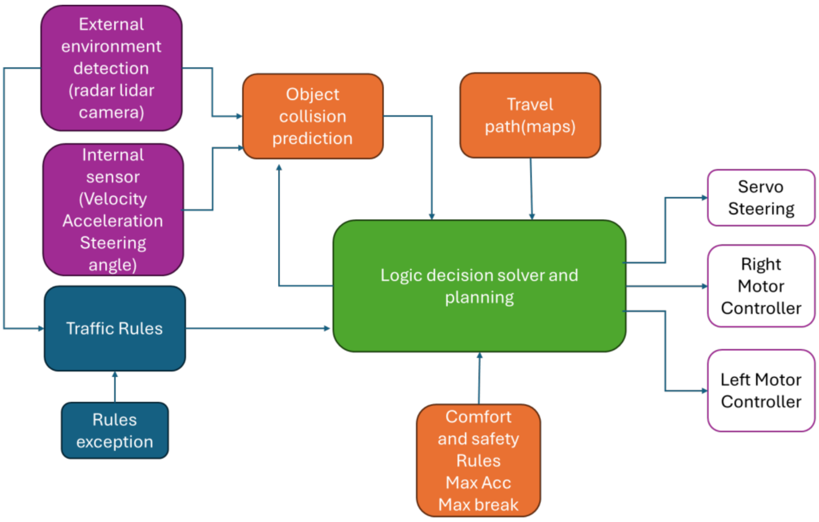

Fig. 2. Autonomous vehicle control block

Moreover, the execution of the control block follows a closed-loop iterative process. The measurement of the different parameters and the calculations are done periodically. Thus, certain parts can operate independently depending on the choice of the computing platform. Below is the functional description of each block:

- **External Environment detection:** This block represents all the sensors detecting the external environment, such as cameras, lidars, and radars. In addition, the reception of location data from satellites and road infrastructure. Furthermore, the autonomous vehicle can exchange information with other vehicles, such as emergency braking or the presence of danger, the intention to change direction, or the presence of other obstacles hidden by the communicating vehicle [26].
- **Internal sensors:** They provide the intrinsic information of the vehicle: linear and radial acceleration, absolute speed, front wheel angle, etc.
- **Traffic Rules:** They gather rules defining the vehicle’s behavior. Including stop or go signs (such as traffic lights, stop signs, or right or left priority signs). The actions of changing the permissible speed limits and reducing speeds at intersections and danger zones.
- **Traffic Rules Exception:** They represent all the specific cases in which the vehicle may not comply with traffic rules. As is the case in certain situations of danger, blockage, or under the orders of a traffic officer.
- **Travel path:** It represents the overall path the autonomous vehicle will follow to reach its destination. The vehicle could change lanes within this path. In general, the route is not changed during the journey unless the road is blocked. In this case, path re-planning is done according to a predefined strategy.
- **Comfort and Safety Rules:** They define the strategy to follow according to the situation. By default, it is the comfort rules that are adopted most of the time. However, in cases of imminent danger, these rules are no longer followed. The example of emergency braking and quick direction

change is the solution to avoid an obstacle.

- **Object collision Prediction:** It is the block that estimates the presence of a collision between the autonomous vehicle and objects within its detection space. It is used in two different situations. The first, using real-time speed and position data of the vehicle. And the second is used on a future projection of speed, direction, and position to respond to a request from the decision block.
- **Logic Decision Solver:** It is the heart of the decision-making system, merging all the data from the previous blocks and providing speed and wheel angle commands for the actuator part [27].

4. Methodology Adopted for the Calculation in the Collision Estimator

For two vehicles, this block uses the wheel angle position, steering, and speed data. Thus, several scenarios are grouped into two situations:

- The first in which two vehicles follow each other along a straight or curved path (Fig. 3).

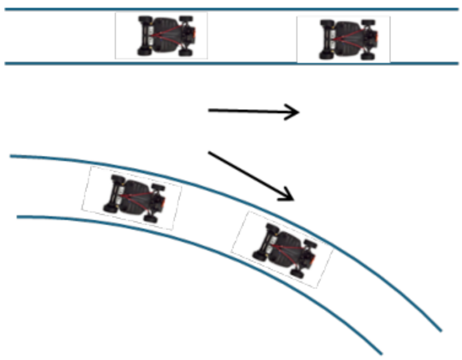

Fig. 3. The two types of straight and curved trajectories for vehicles that follow each other

- In the second case, the paths of the two vehicles intersect, whether they are straight-straight trajectories (Fig. 4A), straight-curved (Fig. 4B), or curved-curved (Fig. 4C).

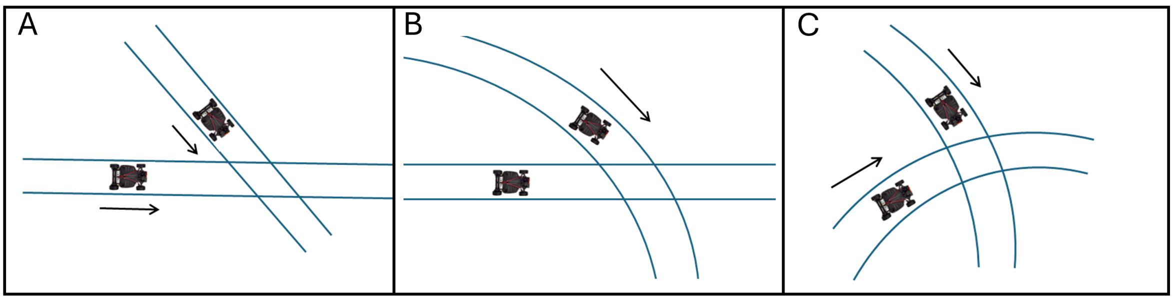

Fig. 4. The types of intersecting trajectories

4.1. Definition and Convention

In Fig. 5, the vehicle is modeled by four points a, b, c, and d. Moreover, for the case of a linear trajectory, we define two lines DL and DR (respectively left and right). And in the same way, two concentric arcs of radius RR and RL centered at O are defined for a curved trajectory. The distance between the two lines $DL1$ and $DL2$ or the two arcs is equal to the width of the vehicle.

For example, in the case of two vehicles V_1 and V_2 in the intersecting straight-line trajectories shown in Fig. 6, we define four intersection points of the lines $DL1$, $DR1$, $DL2$, and $DR2$ at four points $i1$, $i2$, $i3$, and $i4$, defining the collision zone according to expressions below:

Vol. 5, No. 1, 2025, pp. 555-569

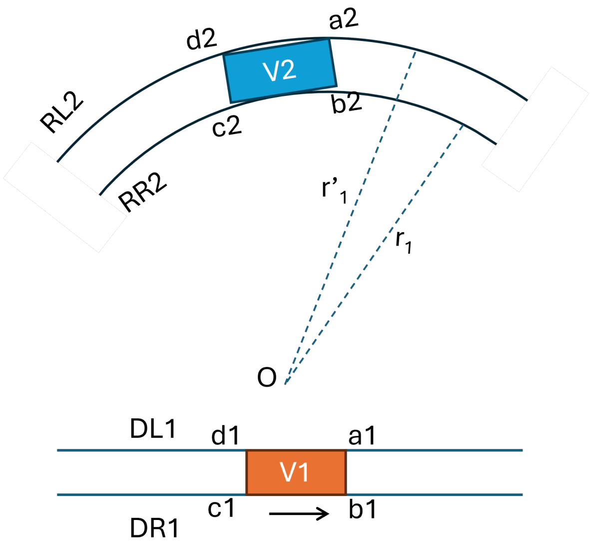

Fig. 5. Modeling of straight and curved trajectories

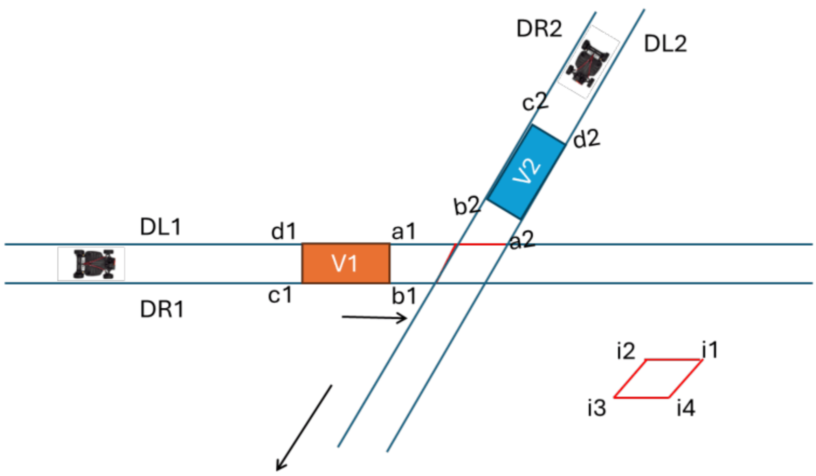

Fig. 6. Intersection points defining the collision zone defined by $i1$, $i2$, $i3$, and $i4$

$$
\begin{aligned}
i1 &= DL1 \cap DL2 \\
i2 &= DL1 \cap DR2 \\
i3 &= DR1 \cap DR2 \\
i4 &= DR1 \cap DL2
\end{aligned}
$$

Additionally, for each iteration, we compute the arrival time for points a, b, c, and d at locations $i1$, $i2$, $i3$, and $i4$ based on the instantaneous speed at time T. The calculations assume the vehicle will sustain a constant speed and trajectory.

5. Calculation Method in the Collision Estimator

5.1. Case of Vehicles Following Each Other

In this situation, the two vehicles are following each other with overlapping trajectories, as shown in Fig. 7. In the case of straight-line trajectories, the calculation is based on the distances between the front of the following vehicle and the rear of the leading vehicle; the speeds of the two vehicles determine whether or not there is a collision, especially in the case where $V_1 > V_2$. However, the distance to the point of impact and the moment of impact are calculated based on the difference in speeds of the two vehicles [28].

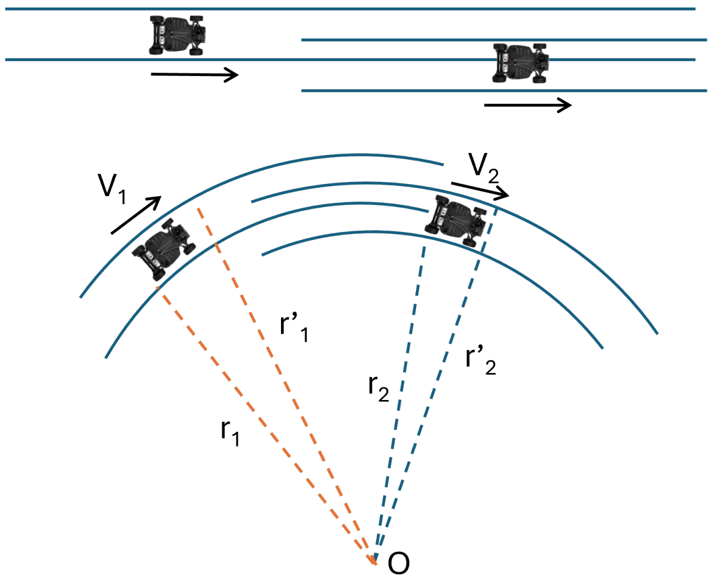

Fig. 7. Modeling of linear and curved trajectories for two following vehicles

Furthermore, in the case of concentric trajectories with different curvature radii and a center O, the calculation is based on the angular speeds $\dot{\theta}_1$ point and $\dot{\theta}_2$ point derived from the speeds of the vehicles V_1 and V_2 and the curvature radii. Thus, if $\dot{\theta}_1$ is greater than $\dot{\theta}_2$ with overlapping trajectories, there will be a collision.

5.2. Case of Intersecting Trajectories

The principle of the method is to define the arrival times Ta, Tb, Tc, and Td of the different points a, b, c, and d of the two vehicles V_1 and V_2 at points $i1$, $i2$, $i3$, and $i4$. The idea is to define the arrival time intervals of the front and rear points calculated based on the speeds of vehicles V_1 and V_2. The absence of collision is mainly due to the lack of intersection of the time intervals of arrival moments according to the conditions below. The method remains the same regardless of the type of trajectory (straight or curved) and regardless of the direction of the vehicle's movement. However, there is a difference in the method of calculating arrival times depending on the type of movement.

NO COLLISION IF

At $i1$: $[Ta1, Td1] \cap [Ta2, Td2]$=$\emptyset$

AND

At $i2$: $[Ta1, Td1] \cap [Tb2, Tc2]$=$\emptyset$

AND

At $i3$: $[Tb1, Tc1] \cap [Tb2, Tc2]$=$\emptyset$

AND

At $i4$: $[Tb1, Tc1] \cap [Ta2, Td2]$=$\emptyset$

5.3. Collision Estimator Algorithm

For the collision detector, it starts by checking if the two trajectories intersect. If there is no intersection, he calculates for two cars that follow each other, regardless of their straight or curved trajectories. Otherwise, it calculates the times and compares the intervals, then determines whether or not there is a collision. The collision estimator will be able to have real-time data on the speeds, trajectories, and positions of the vehicles. Or hybrid data based on a projection of a future action with speed, position, or trajectory commands.

5.4. Case of Two Vehicles Following Each Other

Fig. 8 represents the pseudo-code of the estimator in the case where two vehicles follow each other along a straight trajectory. The algorithm determines if a collision would occur between two vehicles moving in a linear manner. He starts by calculating the relative distance between the vehicles, taking their lengths into account, then he determines their relative speed projected onto the $X-$axis. If a positive relative speed is detected (indicating an approach), the algorithm calculates the time required before a potential collision. The future positions of the vehicles are then estimated, and the distance traveled by the first vehicle up to that point is calculated, limited to a maximum value. (supVal). Finally, the algorithm returns the time of the potential collision as well as this distance. If the relative speed is negative or zero, no collision is possible according to this simplified model, and a default value (supVal) is returned. This algorithm is based on a simplified model of linear motion and does not take into account factors such as acceleration or changes in direction.

```
Get positions of car1 and car2.
Calculate relative distance between car1 and car2.
Calculate relative velocity between car1 and car2.
if relative velocity ≤ 0 then
    return no collision (SupVal, SupVal).
end if
Calculate time remaining until collision, ΔT.
Get positions of both cars after ΔT.
if distance between cars after ΔT > ε then
    return no collision (SupVal, SupVal).
end if
Calculate the distance traveled by car1 before collision, d_i.
Calculate the time of collision, t_i.
return time of collision t_i and distance traveled d_i.
```

Fig. 8. The Pseudo-Code for the algorithm to calculate distance and the moment of collision for the case of vehicles following each other on a straight trajectory

5.5. Case of Two Vehicles Crossing Paths at an Intersection

Fig. 9 represents the pseudo-code of the algorithm that detects collisions between two cars using a geometric and temporal approach. He starts by defining reference points on each vehicle to then calculate their trajectories. By determining the intersection points of these trajectories and comparing the arrival times of each car at these points, the algorithm identifies if there is a collision. The angle between the trajectories is also taken into account to determine the type of collision (frontal, lateral, rear). If a collision is detected, the algorithm precisely calculates the point and moment of impact. Finally, it categorizes the collision according to different possible scenarios (based on the angle and other factors), allowing for a more detailed analysis of the event. Additional information on the trajectories can also be provided. The accuracy of the algorithm depends on the quality of the input data regarding the position, speed, and direction of the vehicles.

The two algorithms remain valid for cases of curved or straight trajectories; the main difference is the method of calculating distances and arrival times based on the radii of curvature.

6. Results and Discussion

This section showcases two Python simulations that utilize the Pygame library to replicate our model. We employed a time step of 17 milliseconds for each iteration, with pixels as the fundamental unit of displacement. The algorithm is comparatively straightforward with few branches. Consequently, we will be capable of simulating a substantial number of cars on GPU architectures, as demonstrated in our prior research on the NaSCH traffic model implementation [29] [30].

Require: List of cars, current time
Ensure: Impact point and time, collision paths if collision is detected
Define delimitation points for both cars:
Extract points $a1, b1, c1, d1$ for first car.
Repeat for second car with points $a2, b2, c2, d2$.
Calculate heading lines for each car:
Determine $dl1$ and $dr1$ for first car from points $a1$ and $b1$.
Determine $dl2$ and $dr2$ for second car from points $a2$ and $b2$.
Find intersection points between these lines:
Calculate intersections $i1, i2, i3, i4$ from line combinations.
Calculate angle between directions of two cars:
Calculate $\Delta\theta$ based on heading differences.
Calculate arrival times for each intersection point:
Compute arrival times like $Ta11, Td11$ for car 1 at point $i1$, etc.
Test if time intervals overlap to determine collision:
For each pair of intervals (Ta, Td), check for intersection.
Display result and determine if there is point of impact:
If collision is detected, further calculate exact impact point p_i and time t_i.
Based on $\Delta\theta$, determine different collision scenarios:
Execute specific checks and calculations for each scenario.
Optionally determine additional information on collision paths.

Fig. 9. The Pseudo-Code of the algorithm detecting collisions between two cars using a geometric and temporal approach

6.1. Simulation 1: Case of Two Following Vehicles

In this simulation (Fig. 10), we used a scenario where two vehicles follow each other along a straight trajectory. The blue following car is equipped with an active braking and acceleration system according to the algorithm in Fig. 11. The speeds of the two vehicles are plotted in Fig. 12. Thus, the following vehicle adjusts its speed according to the leading vehicle to maintain a safe distance (adaptive cruise control). The inter-vehicle distance (d_{iv}) remains stable thanks to dynamic adjustments, thus avoiding potential collisions. The braking moments are indicated by the peaks on the “Brake” curve, which are activated to manage the speed changes of the lead vehicle. The braking distance (d_f) represents the distance required for the following vehicle to come to a complete stop safely. Simultaneously, the system manages the distance from the potential impact point (D_v) to prevent collisions by proactively adjusting speed and distance. Together, these elements highlight the effectiveness of the active braking system in maintaining road safety.

6.2. Simulation 2: Crossing Case with Dynamic Braking

Fig. 13 shows a sequence from the simulation of a vehicle (blue) approaching an intersection where a convoy of two cars (red) is passing through. The convoy of red vehicles moves at a constant speed, unlike the blue vehicle, which adjusts its speed several times.

The curve in Fig. 14 shows the dynamic change in speed as a function of the distance to the impact point located in the collision zone of the intersection. The blue curve represents the autonomous vehicle, which initially maintains a speed of about 100 pix/s. However, it slows down twice when it detects the first and second vehicles of the convoy, which are moving at a stable speed of 80 pix/s. The first slowdown reduced the blue car’s velocity to 90 pix/s. During the second slowdown, the car drastically reduces its velocity to the point of total braking, maintaining a safe distance of 100 pix between the blue car and the convoy. Otherwise, the orange curve indicates the distance to the point of impact, which gradually decreases with two notable drops. These drops correspond to the blue vehicle’s successive detection of other vehicles, which prompts a reduction in its speed to prevent an accident. The ”‘brake”’ curve illustrates the points at which the brakes engage, thereby conveying the control system’s reaction to the varying distances before impact and during braking.

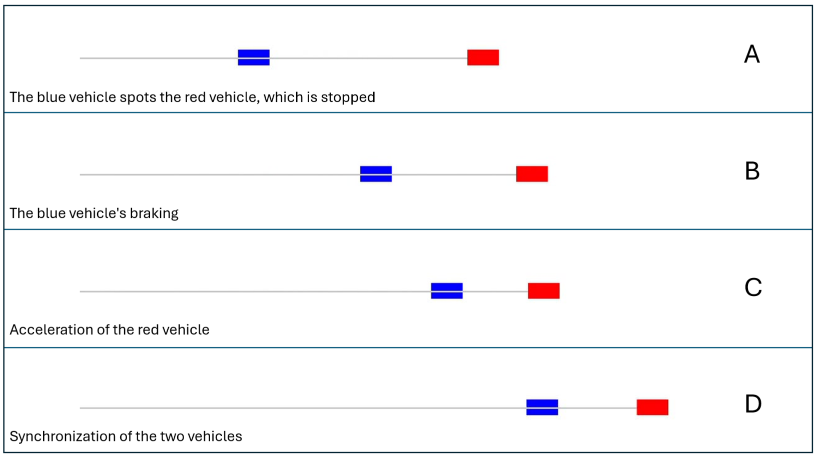

Fig. 10. The Pseudo-Code for the speed change algorithm based on the safety distance and the distance to the impact point

```
if inter-vehicle distance < DETECTING_RANGE then
    if distance between vehicle and point of impact − braking distance of vehicle > SAFETY_DISTANCE then
        if first vehicle's speed < CRUISING_SPEED and inter-vehicle distance ≥ SAFETY_DISTANCE then
            Update acceleration of the first vehicle to ACCELERATION
            Disable braking for the first vehicle
        else
            Update deceleration of the first vehicle to DECELERATION
            Enable braking for the first vehicle
        end if
    else
        Enable braking for the first vehicle
        Update deceleration of the first vehicle to DECELERATION
    end if
else
    if first vehicle's speed < CRUISING_SPEED then
        Update acceleration of the first vehicle to ACCELERATION
        Disable braking for the first vehicle
    else
        Update deceleration of the first vehicle to DECELERATION
        Enable braking for the first vehicle
    end if
end if
```

Fig. 11. Different stages of the active speed change of the blue vehicle based on the behavior of the red one

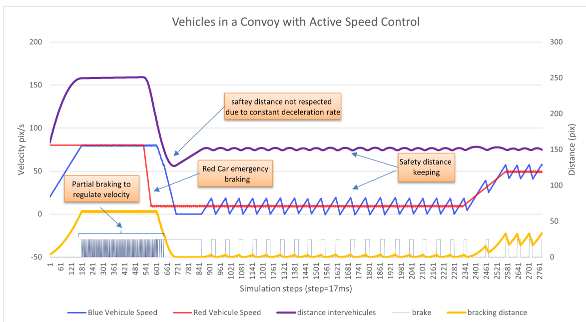

Fig. 12. Chart of the results of speed and inter-vehicle distance showing the reliability of the braking and active acceleration of vehicle two

Fig. 15 represents the pseudo-code of the loop used to manage the speed of the autonomous vehicle. At the beginning of this function, the necessary global variables are initialized. The function checks if the autonomous vehicle or other vehicles in its immediate environment are approaching a potential impact point. If such an impact point is detected, the autonomous vehicle activates its brakes to avoid a collision, and a message indicating the detection of the impact point is displayed.

If no impact point is detected, the function checks if the vehicles involved have exited the danger zone. If the vehicles have not exceeded the point of impact and are within a distance shorter than the defined safety distance, braking is also activated to maintain safety. However, if the distance is sufficient and there is no immediate danger, the vehicle adjusts its acceleration to reach or maintain a defined cruising speed. Throughout this logic, messages are displayed to monitor the vehicle's status, indicating whether safety distances are being maintained or if the vehicle continues to maintain its cruising speed without restriction.

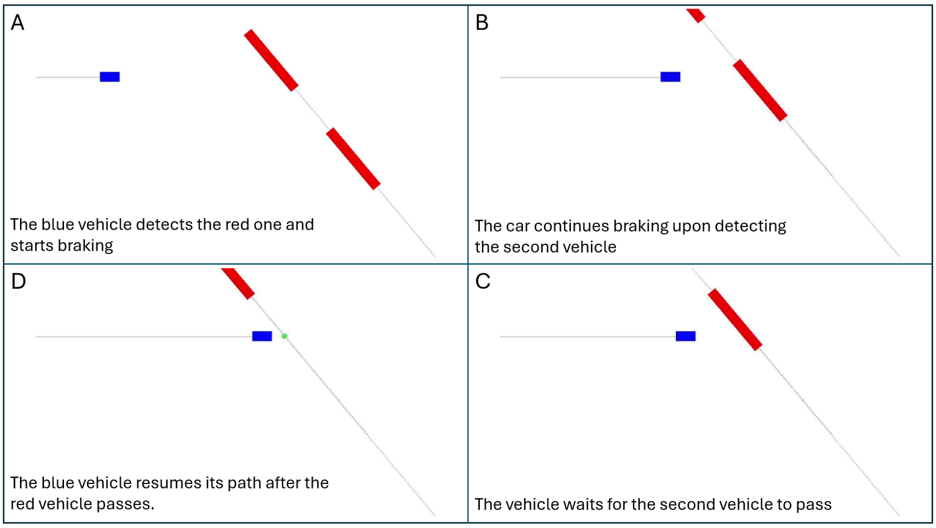

Fig. 13. Simulation of active speed change of a vehicle approaching an intersection with the passage of two vehicles

6.3. Limitations and Future Work

The suggested predictive model, as previously stated, does not account for measurement errors or misinterpretations of information. The existence of hidden vehicles or objects with hypothesized direction and velocity may provide big challenges due to the temporary or complete lack of data. Furthermore, we plan in future work to integrate the prediction of a neighboring vehicle's change of direction based on its behavior (normal or aggressive driving) and the analysis of the surrounding traffic. We also intend to incorporate logic into the decision-making block when there is a complete lack of information, such as in instances of sensor disruption or camera glare.

7. Conclusion and Perspectives

We developed a concept of an autonomous vehicle using a modular control architecture, and we established and validated collision prediction algorithms via simulation. The main aim of our research is to guarantee safety in autonomous vehicles for both urban and rural driving conditions. Due to their capacity to respond in under a millisecond, these systems greatly surpass human reaction rates, which often fall between 1 and 1.5 seconds, resulting in a decrease in accidents by 40% or greater. Furthermore, comfort is an essential element of urban driving. The collision prediction system facilitates the prevention of abrupt braking, reserving it exclusively for emergencies. A forthcoming article will detail subsequent research, focusing on the interaction of the decision block with other components described in our model. This ongoing project will examine the prediction of collisions between concealed items and cars. The current version omits considerations of latency and measurement reliability; addressing these aspects could improve the comprehensiveness of our study. Subsequent iterations could be enhanced by using error margins or probabilistic models, like Kalman filters, to enhance forecast accuracy.

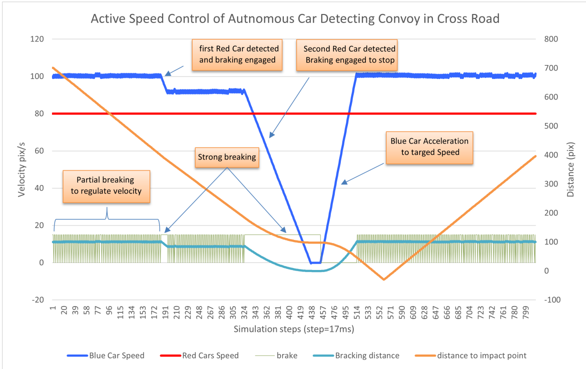

Fig. 14. Dynamic speed change of vehicle 1 (Blue Car) upon detecting the passage of two vehicles at the intersection

```
function LOOP
    Initialize global variables: target_speed, state, old_direction, data_list, cu-
mulate_steering_error
    Check impact points for the autonomous vehicle with other vehicles
    Obtain results: impact_point_vehicle1, vehicle1_passed_impact
                    impact_point_vehicle2, vehicle2_passed_impact
    if impact_point_vehicle1 or impact_point_vehicle2 then
        Activate the autonomous vehicle's brake
        Display "flag: impact point detected"
    else
        if no vehicle has passed the impact point then
            if distance to autonomous vehicle < SAFETY_DISTANCE then
                Activate the autonomous vehicle's brake
                Display "flag: safety distance breached"
            else
                if autonomous vehicle's speed < CRUISE_SPEED then
                    Increase the autonomous vehicle's acceleration
                    Deactivate the autonomous vehicle's brake
                else
                    Decrease the autonomous vehicle's acceleration
                    Maintain brake status
                end if
                Display "flag: safety respected"
            end if
            Display "flag: vehicle(s) at impact point"
        else
            if autonomous vehicle's speed < CRUISE_SPEED then
                Increase the autonomous vehicle's acceleration
                Deactivate the autonomous vehicle's brake
            else
                Decrease the autonomous vehicle's acceleration
                Maintain brake status
            end if
            Display "flag: constant speed maintained"
        end if
    end if
```

Fig. 15. The Pseudo-Code for the Speed Change Based on Vehicles Passing Through the Intersection

Author Contribution: Yassine El Hafid: Conceptualization, Methodology, Software, Writing-Reviewing. Tarik Ligabi: Software, Validation. Yassine Zahraoui: Writing-Original draft preparation, Visualization, Investigation, Supervision.

Funding: There was no external funding for this study.

Acknowledgment: We extend our gratitude to the administration of the Hassania School of Public Works (EHTP) for their support and assistance in the completion of this work, as well as for facilitating our team's participation in the international autonomous vehicle competition, NXPCUP.

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] A. Chand, S. Jayesh, and A. B. Bhasi, "Road traffic accidents: An overview of data sources, analysis techniques and contributing factors," *Materials Today: Proceedings*, vol. 47, pp. 5135–5141, 2021, https://doi.org/10.1016/j.matpr.2021.05.415.

[2] H. Wang and B. Liu, "Path Planning and Path Tracking for Collision Avoidance of Autonomous Ground Vehicles," in *IEEE Systems Journal*, vol. 16, no. 3, pp. 3658-3667, 2022, https://doi.org/10.1109/JSYST.2021.3085479.

[3] L. Zheng, P. Zeng, W. Yang, Y. Li, and Z. Zhan, "Bézier curve-based trajectory planning for autonomous vehicles with collision avoidance," *IET Intelligent Transport Systems*, vol. 14, no. 13, pp. 1882–1891, 2020, https://doi.org/10.1049/iet-its.2020.0355.

[4] M. Ammour, R. Orjuela and M. Basset, "A MPC Combined Decision Making and Trajectory Planning for Autonomous Vehicle Collision Avoidance," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 24805-24817, 2022, https://doi.org/10.1109/TITS.2022.3210276.

[5] H. Alghodhaifi and S. Lakshmanan, "Autonomous Vehicle Evaluation: A Comprehensive Survey on Modeling and Simulation Approaches," in *IEEE Access*, vol. 9, pp. 151531-151566, 2021, https://doi.org/10.1109/ACCESS.2021.3125620.

[6] P. S. Perumal *et al*., "An insight into crash avoidance and overtaking advice systems for Autonomous Vehicles: A review, challenges and solutions," *Engineering Applications of Artificial Intelligence*, vol. 104, p. 104406, 2021, https://doi.org/10.1016/j.engappai.2021.104406.

[7] J.-B. Receveur, S. Victor, and P. Melchior, "Autonomous car decision making and trajectory tracking based on genetic algorithms and fractional potential fields," *Intelligent Service Robotics*, vol. 13, no. 2, pp. 315–330, 2020, https://doi.org/10.1007/s11370-020-00314-x.

[8] T. Zhang, W. Song, M. Fu, Y. Yang, X. Tian and M. Wang, "A Unified Framework Integrating Decision Making and Trajectory Planning Based on Spatio-Temporal Voxels for Highway Autonomous Driving," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 10365-10379, 2022, https://doi.org/10.1109/TITS.2021.3093548.

[9] P. Ghorai, A. Eskandarian, Y. -K. Kim and G. Mehr, "State Estimation and Motion Prediction of Vehicles and Vulnerable Road Users for Cooperative Autonomous Driving: A Survey," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 16983-17002, 2022, https://doi.org/10.1109/TITS.2022.3160932.

[10] N. Klinjun, M. Kelly, C. Praditsathaporn, and R. Petsirasan, "Identification of Factors Affecting Road Traffic Injuries Incidence and Severity in Southern Thailand Based on Accident Investigation Reports," *Sustainability*, vol. 13, no. 22, p. 12467, 2021, https://doi.org/10.3390/su132212467.

[11] Y. Huang, J. Du, Z. Yang, Z. Zhou, L. Zhang and H. Chen, "A Survey on Trajectory-Prediction Methods for Autonomous Driving," in *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 3, pp. 652-674, 2022, https://doi.org/10.1109/TIV.2022.3167103.

[12] F. Leon and M. Gavrilescu, "A Review of Tracking and Trajectory Prediction Methods for Autonomous Driving," *Mathematics*, vol. 9, no. 6, p. 660, 2021, https://doi.org/10.3390/math9060660.

[13] P. Kothari, S. Kreiss and A. Alahi, "Human Trajectory Forecasting in Crowds: A Deep Learning Perspective," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 7386-7400, 2022, https://doi.org/10.1109/TITS.2021.3069362.

[14] S. Wang, Z. Bao, J. S. Culpepper, and G. Cong, "A Survey on Trajectory Data Management, Analytics, and Learning," *ACM Computing Survey*, vol. 54, no. 2, pp. 1–36, 2021, https://doi.org/10.1145/3440207.

[15] S. Capobianco, L. M. Millefiori, N. Forti, P. Braca and P. Willett, "Deep Learning Methods for Vessel Trajectory Prediction Based on Recurrent Neural Networks," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 6, pp. 4329-4346, 2021, https://doi.org/10.1109/TAES.2021.3096873.

[16] L. Lin, W. Li, H. Bi and L. Qin, "Vehicle Trajectory Prediction Using LSTMs With Spatial–Temporal Attention Mechanisms," in *IEEE Intelligent Transportation Systems Magazine*, vol. 14, no. 2, pp. 197-208, 2022, https:/ddoi.org/10.1109/MITS.2021.3049404.

[17] X. Song *et al.*, "Pedestrian Trajectory Prediction Based on Deep Convolutional LSTM Network," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 6, pp. 3285-3302, 2021, https:/doi.org/10.1109/TITS.2020.2981118.

[18] R. Quan, L. Zhu, Y. Wu and Y. Yang, "Holistic LSTM for Pedestrian Trajectory Prediction," in *IEEE Transactions on Image Processing*, vol. 30, pp. 3229-3239, 2021, https://doi.org/10.1109/TIP.2021.3058599.

[19] C. Vishnu, V. Abhinav, D. Roy, C. K. Mohan and C. S. Babu, "Improving Multi-Agent Trajectory Prediction Using Traffic States on Interactive Driving Scenarios," in *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 2708-2715, 2023, https://doi.org/10.1109/LRA.2023.3258685.

[20] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings and A. Mouzakitis, "Deep Learning-Based Vehicle Behavior Prediction for Autonomous Driving Applications: A Review," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 33-47, 2022, https://doi.org/10.1109/TITS.2020.3012034.

[21] R. Hajiloo, M. Abroshan, A. Khajepour, A. Kasaiezadeh and S. -K. Chen, "Integrated Steering and Differential Braking for Emergency Collision Avoidance in Autonomous Vehicles," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 5, pp. 3167-3178, 2021, https://doi.org/10.1109/TITS.2020.2984210.

[22] A. P. Carrone, J. Rich, C. A. Vandet, and K. An, "Autonomous vehicles in mixed motorway traffic: capacity utilisation, impact and policy implications," *Transportation*, vol. 48, no. 6, pp. 2907–2938, 2021, https://doi.org/10.1007/s11116-020-10154-4.

[23] C. Wang, X. Zhao, R. Fu, and Z. Li, "Research on the Comfort of Vehicle Passengers Considering the Vehicle Motion State and Passenger Physiological Characteristics: Improving the Passenger Comfort of Autonomous Vehicles," *International Journal of Environmental Research and Public Health*, vol. 17, no. 18, p. 6821, 2020, https://doi.org/10.3390/ijerph17186821.

[24] D. Parekh, N. Poddar, A. Rajpurkar, M. Chahal, N. Kumar, G. P. Joshi, and W. Cho, "A Review on Autonomous Vehicles: Progress, Methods and Challenges," *Electronics*, vol. 11, no. 14, p. 2162, 2022, https://doi.org/10.3390/electronics11142162.

[25] Y. Li, Y. Cai, X. Sun, H. Wang, Y. Jia, Y. He, L. Chen, and Y. Chao, "Trajectory tracking of four-wheel driving and steering autonomous vehicle under extreme obstacle avoidance condition," *Vehicle System Dynamics*, vol. 62, no. 3, pp. 601–622, 2024, https://doi.org/10.1080/00423114.2023.2186249.

[26] A. Boubakri and S. M. Gamar, "A New Architecture of Autonomous Vehicles: Redundant Architecture to Improve Operational Safety," *International Journal of Robotics and Control Systems*, vol. 1, no. 3, pp. 355–368, 2021, https://doi.org/10.31763/ijrcs.v1i3.437.

[27] W. Farag, M. Abouelela, and M. Helal, "Finding and Tracking Automobiles on Roads for Self-Driving Car Systems," *International Journal of Robotics and Control Systems*, vol. 3, no. 4, pp. 704–727, 2023, https://doi.org/10.31763/ijrcs.v3i4.1022.

[28] M. H. Harun, S. S. Abdullah, M. S. M. Aras, M. B. Bahar, and F. Ali Ibrahim, "Recent Developments and Future Prospects in Collision Avoidance Control for Unmanned Aerial Vehicles (UAVS): A Review," *International Journal of Robotics and Control Systems*, vol. 4, no. 3, pp. 1207–1242, Jul. 2024, https://doi.org/10.31763/ijrcs.v4i3.1482.

[29] T. Feng, K. Liu, and C. Liang, "An Improved Cellular Automata Traffic Flow Model Considering Driving Styles," *Sustainability*, vol. 15, no. 2, p. 952, 2023, https://doi.org/10.3390/su15020952.

[30] Y. E. Hafid, A. E. Rharras, M. Wahbi, and R. Saadane, "GPU optimized parallel implementation of NaSch traffic model," *Proceedings of the Third International Conference on Computing and Wireless Communication Systems*, 2019, http://dx.doi.org/10.4108/eai.24-4-2019.2284233.