Minimal Injury Risk Motion Planning using Active Mitigation and Sampling MPC

Luiz Alberto SERAFIM GUARDINI[1,2,3], Anne SPALANZANI[1], Philippe MARTINET[2],
Christian LAUGIER[1], Thomas GENEVOIS[1], Anh-Lam DO[3]

Abstract— **Collision mitigation is an important element in motion planning. Although Advanced Driver-Assistance Systems (ADAS) have a rich number of functionalities, they lack interchangeability. There is still a gap on finding a way to evaluate the best decision globally. This paper presents a novel motion planning framework to generate emergency maneuvers in complex and risky scenarios using active mitigation. The classical MPPI algorithm is improved to be used in a probabilistic dynamic cost map under limited perception range. A cost map with global probability of injury to all road users is used as a constraint to the problem in order to compute target selection based on the global minimum risk considering all road users. Real experiments introduce the use of virtual objects by merging simulation and real sensor data to safely produce collision and mitigation experiments. Results show that the proposed algorithm can perform correctly, by finding collision-free trajectories in complex scenarios and compute viable target selection that minimizes global injury risk when collision is inevitable.**

I. Introduction

Advanced Driver-Assistance Systems (ADAS) have been developed to raise safety and driving comfort. Primary safety has been addressed over the years with the advancement of safety technologies such as Autonomous Emergency Braking (AEB) and Advanced Evasive Steering (AES) systems.

NCAP-2025 Roadmap contemplates the development of Autonomous Emergency Braking and Steering Systems (AEBSS), which integrate steering and/or differential braking by taking action autonomously, shifting from a system based assessment to a scenario-based assessment. Its aim is to deliver improved passenger car safety but also on how it might assist other road users. The main objective is to make use of new technologies to minimize human error, which are responsible for over 90% of road accidents [1].

Many works in the literature propose motion planning for emergency scenarios considering collision avoidance, vehicle stability and path tracking supposing collision-free scenarios. However, not many works consider AEBSS application for complex scenarios mitigation, such is the case in [2], [3], These works however, limit the scope of the risk assessment to the ego-vehicle occupants, disregarding the related risk to all other road users.

On the other hand, works based in accidentology propose a framework to quantify the benefits and efficiency of AEB [4] and AEBSS [5] towards vulnerable road users. Yet, these studies comprise a posterior analysis of the probability of injury in the event of a collision.

To develop our AEBSS framework we rely on the MPPI technique developed by Williams [6]. It derives from Path Integral Control [7] which transforms the value function of the Optimal Control Problem (OCP) into an expectation over all possible trajectories. It is a method that allows Stochastic Optimal Control (SOC) problems to be solved with a Monte Carlo approximation using forward sampling of stochastic diffusion processes [8]. Thus, MPPI uses a path integral method to find a control sequence by minimizing the running cost, which corresponds to the integral of each individual cost in each step, where the solution to the Hamilton-Jacobi-Bellman equation is approximated by importance sampling of these paths using Feynman-Kac theorem and KL divergence [6].

The use of MPPI is appealing because it is a derivative-free optimization method, which allows the use of non-linear and non-convex models and cost functions. As MPPI has shown good performance in aggressive driving situations, it seems therefore particularly well adapted to emergency trajectories in the ADAS context.

The original version of the MPPI algorithm [6] considers an offline static cost map with previous knowledge of the track configuration. It has been extended to environments with dynamic obstacles in [9], and an online cost map using on-board monocular cameras was considered in [10], [11]. Also, a study on partially observable scenarios for Quadrotors was introduced in [12].

The main contribution of this work is the development of a novel motion planning technique for vehicle navigation in a risky environment considering Active Mitigation, i.e, real time target selection based on accidentology data.

In technological requirements, our methodology fulfills Euro NCAP 2025+ specs for an AEBSS motion planning, going beyond scenario-based assessment, since it can be considered a generic assessment method. Therefore, our work's emergency trajectories planning considers the global injury risk for all binary collisions between the ego vehicle and a given object in the scene by using the Probability of Collision with Injury Risk (PCIR) [13] as a constraint to the problem.

To the best of our knowledge, this is the first time PCIR is integrated into motion planning, as well as the use of MPPI considering a probabilistic dynamic cost map with limited perception range. In other words, from a classical

[1] Univ. Grenoble Alpes, INRIA, 38000 Grenoble, France, name.surname@inria.fr

[2] INRIA Sophia Antipolis-Méditerranée, ACENTAURI Team, 2004 Route des Lucioles, 06902 Valbonne, name.surmane@inria.fr

[3] Renault S.A.S, 1 av. du Golf, 78288 Guyancourt, France. name.surmane@renault.com

navigation algorithm, our work presents a novel AEBSS general framework that integrates Active Mitigation (by the use of PCIR) to compute either collision-free trajectories when available, or a mitigation trajectory that comprises a real-time target selection based on the global probability of injury concerning all road users.

Real testing validation was conducted by using Virtual Objects and Augmented Reality. It is possible then to overcome the adversity of real testing collisions with pedestrians and/or other vehicles, making it possible to conduct real experiments in safety while considering a virtual collision with objects in the scene.

The remainder of the paper is organized as follows. Section II introduces the system's architecture. Section III explains the Emergency Trajectory MPPI (ET-MPPI) methodology. Section IV presents testing results and discussions. Section V provides conclusion remarks.

II. System Architecture

Our proposed system for the motion planning framework is depicted in Figure 1.

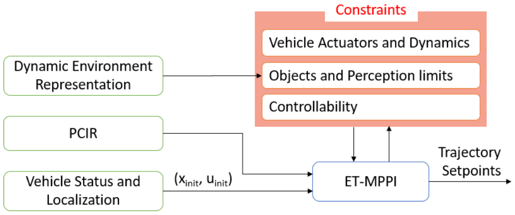

Constraints
Vehicle Actuators and Dynamics
Objects and Perception limits
Controllability
Dynamic Environment Representation
PCIR
Vehicle Status and Localization
(x_{init}, u_{init})
ET-MPPI
Trajectory Setpoints

Fig. 1: System Architecture.

Inputs are: the Dynamic Environment Representation, given by Conditional Monte Carlo Dense Occupancy Tracker (CMCDOT) [14], a generic spatial occupancy tracker that infers dynamics of the scene through a hybrid representation of the environment; PCIR, which provides the probability of collision between the ego-vehicle and an object while also considering the probability of injury as a function of the type of object (pedestrian, cyclist, another vehicle etc.) and the impact speed between the ego-vehicle and the corresponding object [13]; Vehicle Status and Localization, which refers to ego vehicle information such as position, velocity, acceleration and steering angle. In practice, the odometry information is obtained by combining information from an Xsens GPS and an Inertial Measurement Unit (IMU) present in the prototype. The odometry estimator is based on an Extended Kalman Filter (EKF) in order to infer the pose of the ego vehicle.

Constraints to the system are: Vehicle Actuators and Dynamics, i.e. vehicle model; objects in the scene and the threshold of perception; and, vehicle controllability, further discussed in IV-B.

The output of the system is an optimal trajectory that considers either a collision-free trajectory when possible or a mitigation trajectory that minimizes the global risk of injury considering all binary collisions between the ego-vehicle and each object in the scene.

III. Emergency Trajectory MPPI (ET-MPPI)

The original MPPI algorithm was developed to perform aggressive maneuvers while maintaining the ego vehicle speed as constant as possible. In emergency trajectories, besides steering angle, one also needs to change the velocity. To deal with emergency situations, our work makes use of a large range of control inputs and due to the characteristics of the MPPI approach, a significant number of trajectories would be necessary to contain all the input variation needed.

It becomes too expensive to compose the two-dimensional set comprising the desired extent for both the steering angle and the velocity for each cell in the probabilistic occupation grid [14]. For instance, let us imagine a pedestrian and the reachability set of the vehicle. A vast number of trajectories would be needed in order to contain all possible (combinations of steering angle and velocity variations within the desired scope for each cell representing an object.

The Emergency Trajectory MPPI (ET-MPPI) framework for emergency scenarios, is presented in Algorithm 1. The modifications allowing the use of MPPI in emergency scenarios are highlighted in red from the original algorithm which can be found in [6]. Due to the lack of place, it is not possible to detail it. The main idea of the proposed algorithm is to compare all admissible evasive trajectories with the full braking trajectory (by AEB) to determine the best one. Further explanation on the definition of the cost function is given later in this section.

Algorithm 1: ET-MPPI Algorithm

```
1  Procedure computeMppiControl($x_{init}$,$u_{init}$,$\Delta t$)
2      for $k \leftarrow 0$ to $K-1$ do   // For all sampled trajectories
3          Sample $\varepsilon_t^k \in \mathcal{N}(0, \nu\Sigma)$   // Generate Gaussian noise
4          for $t \leftarrow 0$ to $T-1$ do   // For prediction horizon
5              $\mathbf{v}_t = \mathbf{u}_t + \varepsilon_t^k$   // Adding noise to control
6              $\mathbf{v}_{t_{brake}} = \mathbf{u}_t + \varepsilon_{t_{brake}}^k$   // Emergency braking control
7              $\mathbf{x} \leftarrow \mathbf{x} + F(\mathbf{x}, g(\mathbf{v}_t))\Delta t$   // State update
8              $\mathbf{x}_{brake} \leftarrow \mathbf{x}_{brake} + F(\mathbf{x}_{brake}, g(\mathbf{v}_{t_{brake}}))\Delta t$
9              if collision($\mathbf{x}_k$) then
10                 $\mathbf{x}_k \leftarrow \mathbf{x}_{k_{brake}}$   // Replace by braking state
11             $\widetilde{S}_k = \widetilde{S}_k + q(\mathbf{x_{t,k}}, \mathbf{u_{t,k}})$   // Trajectory Cost
12         $\widetilde{S}_k = \widetilde{S}_k + \phi(\mathbf{x})$   // State dependent final cost
13     $\rho \leftarrow \min_k(\widetilde{S}_k)$   // Find minimum cost
14     $\eta \leftarrow \sum_{k=0}^{K-1} exp\left(-\frac{1}{\lambda}(S_k - \rho)\right)$   // Normaliser
15     for $k \leftarrow 0$ to $K-1$ do
16         $w_k \leftarrow \frac{1}{\eta} exp\left(-\frac{1}{\lambda}(S_k - \rho)\right)$   // Probability Weight
17     for $k \leftarrow 0$ to $K-1$ do
18         $\mathbf{U} \leftarrow \left(\mathbf{U} + \sum_{k=0}^{K-1} w_k \varepsilon^k\right)$   // Weighted sum control
             update
19     return etmppiTrajectory($x_{init}$,$U$)   // Return trajectory
```

To start, the algorithm requires the status and localization of the ego vehicle at instant t_{init}, here defined as (x_{init}, u_{init}). In our case, it comes from the Vehicle Status and Localization block. Then, KxT random control variations are generated using CUDA's random number generation library (line 3). The variance ν of the noise $\mathcal{N}$ can be increased to avoid local minima by shifting the range of the trajectory cost, which leads to chattered control solutions.

Perturbation is added to the initial control sequence (line 5). The Gaussian noise generated for the acceleration in the original algorithm is replaced by the maximum deceleration in ε_{brake} (line 6), whereas the Gaussian noise for the steering angle remains the same.

The next state is found for each sampled trajectory (line 7) and updated according to each timestep t. The Same is valid for K conservative trajectories $\mathbf{x}_{brake}$ (line 8), which always consider the maximal braking and the corresponding random noise for the steering angle. $F(\cdot)$ represents the vehicle model, in our case, a kinematic bicycle model and $g(\cdot)$ the clamping function for actuators saturation.

Every trajectory $\mathbf{x}_k$ is checked for collisions. If the k^{th} trajectory collides against one of the objects of the scene, it is replaced by the conservative trajectory $\mathbf{x}_{k_{brake}}$ (line 10).

A running cost for each sample trajectory k at a given timestep t is then computed and added to the final cost (lines 11:12). The cost of each sampled trajectory k is converted to a probability weight w_k (line 16), taking into account the minimum weight trajectory (line 13). The optimal control law is then defined via the probability weight w_k averaged over all the perturbation sequences (line 18).

The final trajectory is obtained by passing the final control sequence trough a vehicle model (line 19), a kinematic bicycle model in our case. Since the final optimal trajectory depends on the probability weight of each k trajectory, it is possible to compute pure evasive steering trajectories (or a composition of braking and steering), otherwise, the ET-MPPI algorithm guarantees that the worst-case scenario will always have the maximal braking (same as in AEB), although with the advantage of all possible range of steering for the collision trajectories, which allows some maneuverability.

For the latter, since we apply PCIR as a constraint to the system, it is possible to have a better contextualization of the scene when dealing with mitigation scenarios and compute a target selection considering the global minimal risk. So, if the evaluation considers that no collision-free trajectories can be computed, we need to compute the minimal global risk associated with all K possible trajectories. Those criteria are addressed by defining and tuning the cost function $\widetilde{S}_k$ coefficients as presented in the next subsection.

A. ET-MPPI Cost Functions

Our cost map takes into consideration both static and dynamic objects given by a probabilistic grid. Furthermore, the PCIR is used as a novel constraint to the problem. The elements taken into consideration in order to develop our cost map are the vehicle's actuators and dynamics constraints, objects and PCIR, perception limits and the ego-vehicle predicted trajectory.

The actuator and dynamic constraints and variance cost are already taken into consideration in the original algorithm presented in [15] in the form of a convex cost function.

Our path cost $q(\mathbf{x})$ is defined as:

$$q(\mathbf{x}) = c_{pcir} + c_{whiteline} + c_{limits} + c_{reference} \tag{1}$$

c_{pcir} penalizes collision while considering injury risk:

$$c_{pcir} = w_{pcir}(PCIR) \tag{2}$$

where w_{pcir} is the weight and PCIR is computed according to the impact speed and the type of object in the scene [13]. To avoid collisions as much as possible, w_{pcir} gets a high value, and if mitigation is needed, the value of PCIR will become predominant over all other costs, which sets the target selection.

$c_{whiteline}$ assigns for ego-vehicle trajectory to remain on the right side of the road. Trajectories surpassing the boundaries are penalized as defined in equation (3).

$$c_{whiteline} = w_{whiteline} \tag{3}$$

Where $w_{whiteline}$ corresponds to the correct road lane cost. It is set as a soft constraint to appease an emergency eviction maneuver that requires crossing the middle lane to avoid a collision. This is achieved by setting the corresponding weight to a value that is between Reference and PCIR costs.

c_{limits}, defined in Equation (4) is used to ensure that the ego-vehicle will not cross undesired areas, such as sidewalks, which can lead to loss of control. In cases where a behavior planning is used, this constraint can be replaced by the ones issued by the behavior planner.

$$c_{limits} = w_{limits} \tag{4}$$

Where w_{limits} corresponds to the road limit cost and its value shall surpass w_{PCIR} to assure that even in mitigation cases the constraint will be respected.

Figure 2 shows a diagram to exemplify $c_{whiteline}$ and c_{limits}. Trajectories which are generated and which remain in the green zone (trajectory 3) are not penalized by either costs. Trajectories that go beyond the green zone to the yellow zone (trajectory 2) are penalized with $c_{whiteline}$. Trajectories which are in the red zone (trajectories 1 and 4) are penalized with c_{limits}.

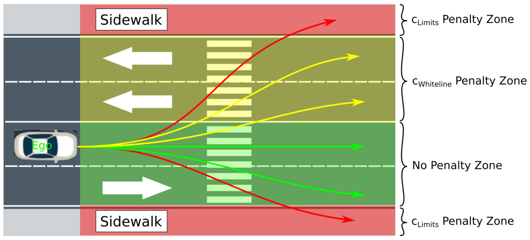

Sidewalk
Ego
Sidewalk
c_{Limits} Penalty Zone
$c_{Whiteline}$ Penalty Zone
No Penalty Zone
c_{Limits} Penalty Zone

Fig. 2: Road Boundaries and Limits.

$c_{reference}$ is used for having the evasive trajectory closer to the projected real trajectory of the vehicle. It is defined as:

$$c_{reference} = (\mathbf{x} - \mathbf{x}_{ref})^T \mathbf{w}_{ref} (\mathbf{x} - \mathbf{x}_{ref}) \tag{5}$$

Where $\mathbf{w}_{ref}$ corresponds to a positive define weight matrix and $\mathbf{x}_{ref}$ are the reference states in a three seconds horizon trajectory prediction from a kinematic bicycle model and the initial states provided by an odometry estimator based on an Extended Kalman Filter (EKF).

IV. Experiments and Testing

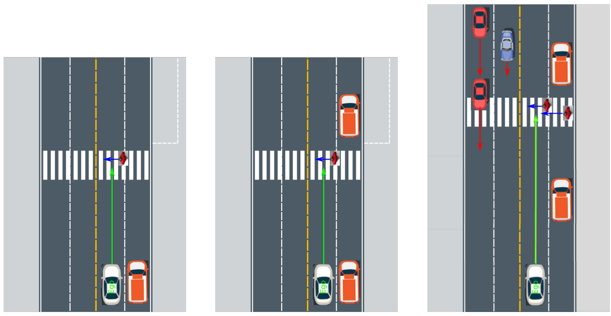
The ET-MPPI algorithm has been tested both in simulation and real experiments. Table I presents the corresponding parameters for the testing.

TABLE I: ET-MPPI Parameters Values

K	4000	T	45
λ	0.004	α	0.7
$w_{whiteline}$	5000	w_{PCIR}	1.6E5
ν	Diag(1.0, 1.2)	w_{ref}	Diag(0, 400, 0, 0)
Σ_{C_x}	Diag(1.0, 2.0)	Σ_{C_y}	Diag(1.0, 22.0)
$R_{terminal}$	Diag(0, 2.5, 0, 400)		

A. Experimental scenarios

We propose a pedestrian crossing scenario, shown in Figure 3, for our analysis. Scenario complexity grows from (a) to (c) to evaluate ET-MPPI algorithm performance in generating either collision free or mitigation optimal trajectories.



Scenario (a) Scenario (b) Scenario (c)

Fig. 3: Pedestrian Crossing Scenarios

The ego vehicle (white), has its field of view occluded by the truck on its right. Therefore, the pedestrian (coming from the right of the ego vehicle) perception happens for Time-to-Collision (TTC) between 0.5 and 2 seconds.

B. Metrics

Two metrics are considered. The first is related to vehicle controllability. Two different control constraints based on the definition of controllability given by ISO 26262 standard [16] had been chosen. C_x (simply controllable) and C_y (difficult to control or uncontrollable), whose parameters for the steering angle δ, the steering rate $\dot{\delta}$ and the acceleration a are presented in Table II. These parameters are inserted in line 3 of Algorithm 1 in order to generate the noise ε_t^k that will be added to the control input and thus influence the variance importance sampling. The control input perturbation along with the clamping function $g(\cdot)$ compose the vehicle reachable set.

The second metric is a performance comparison between AEB systems, C_x and C_y in terms of collision avoidance and/or mitigation.

Each scenario was simulated twenty times for each control constraint. A small offset was added to each of the objects' initial position to observe the robustness of the method with respect to avoidance/mitigation and provide a comparison with the full longitudinal braking AEB system.

TABLE II: Control Constraints C_x and C_y

Constraint	δ $[degrees]$	$\dot{\delta}$ $[degrees/s]$	a $[m/s^2]$
C_x	[-3 3]	[-3 3]	[-3 3]
C_y	[-30 30]	[-30 30]	[-9 3]

C. Simulation Experiments

Simulations take into consideration noisy localization sensors, such as GPS and IMU. Besides, the uncertainty in the perception is also taken into consideration. All simulation results were obtained with the following specs: Intel® Core™ i9-9880H CPU @ 2.30GHz x 16 with NVIDIA Quadro RTX 3000/PCIe/SSE2 GPU under Ubuntu 18.04 ROS Melodic [17] and Gazebo 9 [18].

1) Simulation Results: The graph in Figure 4 presents the percentage of collision-free (blue) and mitigation trajectories (red) for AEB case and those obtained with the proposed ET-MPPI method with control constraints Cx and Cy, considering an ego-vehicle longitudinal velocity of 50 km/h .

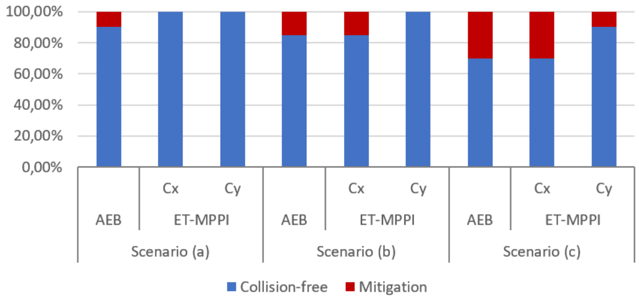


Fig. 4: Percentage of collision-free or mitigation cases for 20 simulation runs for the given scenarios for AEB system and ET-MPPI control constraints C_x and C_y.

Control constraint C_y showed a better performance in avoiding objects in the scene when compared to the other control constraint C_x and the AEB system. This is due to the fact that C_y presents a larger reachability set and an avoidance is possible in many more cases. In most of the cases, configuration C_x does as good as the AEB system, since its output usually corresponds to braking, since its reachable set is much more limited.

Another outcome is that an increase in scenario complexity will lead to situations where collision might become inevitable, and consequently, mitigation is needed.

Next, we discuss and present the output of the ET-MPPI algorithm compared to the AEB for a collision-free trajectory for scenario (a), and a mitigation case for scenario (c), both considering the control constraint C_y at 50 km/h ego-vehicle speed. Scenario (b) will not be discussed since its outcomes tend to one of the presented results.

Scenario (a) is shown in Figure 5 on the left. The system perception and output for the ET-MPPI generated trajectory for Scenario (a) is shown in Figure 5 on the right. The AEB trajectory is represented by the black band and the ET-MPPI trajectory by the yellow band.

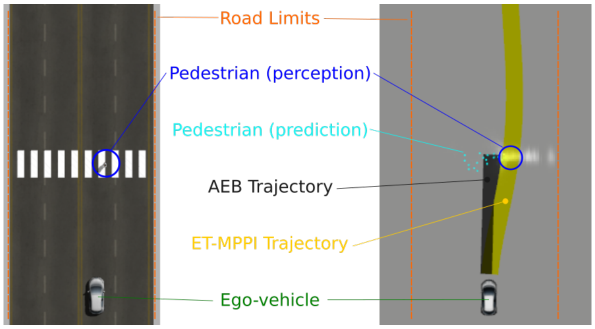


Fig. 5: Scenario (a): On the left the gazebo simulation, on the right, perception, object prediction and ET-MPPI generated trajectory

Although the given scenario is not complex, it is possible to observe that the AEB system will only be able to mitigate the collision, whereas an avoidance trajectory (yellow band with footprint of the ego-vehicle) was generated by the ET-MPPI algorithm. Besides the point of avoiding the obstacle, the ET-MPPI trajectory might also avoid a rear collision to a tailing vehicle, increasing the road safety. For the simulation we consider that the pedestrian will cross the road at constant speed. However, if the pedestrian reacts to the incoming vehicle, the emergency trajectory is replanned (iteration time of 0.05 seconds) and a new emergency trajectory is computed in order to cope with the new changes.

Scenario (c) is depicted in Figure 6, where the scene is shown on the left and the output with the elements of the scene are displayed on the right. The truck on the right, which is occluding the field of perception is pushed closer to the pedestrian passing to decrease even further the reaction time. The goal is to analyse a mitigation scenario by observing how the algorithm generates the ET-MPPI trajectory according to the injury risk associated with each of the objects.

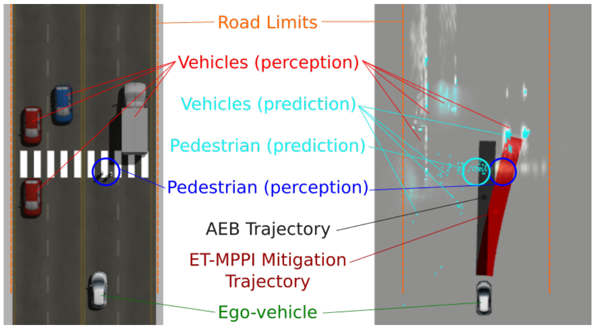


Fig. 6: Scenario (c) after small time: On the left the gazebo simulation, on the right, perception, object prediction and ET-MPPI mitigation trajectory

The ET-MPPI trajectory for this case is displayed in red, since a mitigation trajectory is considered for the scene. Both AEB and ET-MPPI trajectories will result in a collision, however, for the latter, the stationary vehicle on the right hand side is selected through the cost function parameter c_{pcir}, resulting in a trajectory that minimizes the global injury risk.

Figure 7 displays the corresponding PCIR cost map based on the type of object and the prediction. This cost map gives us an indication of the global probability of injury with respect to the highest probability of injury in the scene. For instance, the pedestrian presents the reference PCIR of 1 (or 100%), meaning that it is the most vulnerable road user. The vehicles on the top (blue vehicle) present a very low probability of collision with injury risk (close to white) if compared to the pedestrian. The stationary vehicle on the bottom presents a PCIR close to 40%. It takes precedence over the blue vehicle but does not take precedence over the pedestrian, but since the path to the blue vehicle is blocked by the pedestrian prediction, target selection goes towards the parked truck on the right.

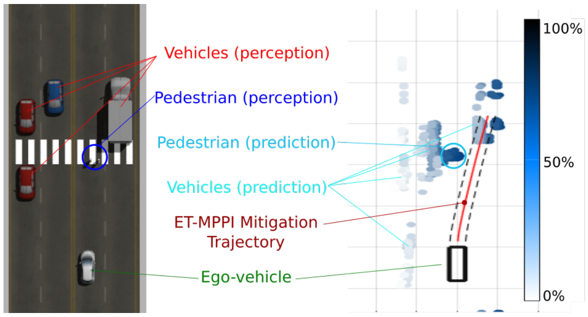


Fig. 7: Scenario (c): PCIR cost map generated from ET-MPPI trajectory cost

In short, AEB trajectory will result in an ego-to-pedestrian collision, which to the corresponding impact speed and assumptions brings a higher probability of injury (around 80% probability of slight and 20% severe injury) to the pedestrian. ET-MPPI algorithm had a better performance in mitigating the collision due to the target selection based on PCIR. For the given scenario, instead of colliding with the pedestrian our algorithm selects the truck on the right, which brings a less than 2% probability of injury to the occupants of the vehicles.

D. Real Experiments

Experiments have been conducted on a dedicated test vehicle, a Renault Zoe equipped with a Velodyne HDL64 LiDAR on the top, 3 Ibeo Lux on the front and 1 on the back, which provides dense 3D point clouds of threshold measurement. Besides, a Xsens GPS, an IMU and a SP90 RTK GPS system provide accurate position and orientation. Same as in simulation, perception relies on the CMCDOT spatial occupancy tracker [14]. The localization is based on the fusion of the odometry and the RTK GPS to obtain a coherent position, orientation and speed estimation.

The experimental validation of this work demands complex test scenarios, where collision with pedestrians and vehicles must be contemplated. For safety reasons, such tests cannot be realized in real conditions. Therefore a novel capability has been developed to make it possible to conduct these experiments. Thus, virtual sensor data is merged with actual sensor data to add pedestrians and vehicles in the test scene using Augmented Reality (AR) by merging simulated and vehicle's LiDAR point cloud information. This representation of the AR is fed seamlessly to the perception software of the actual vehicle, as depicted in Figure 8.

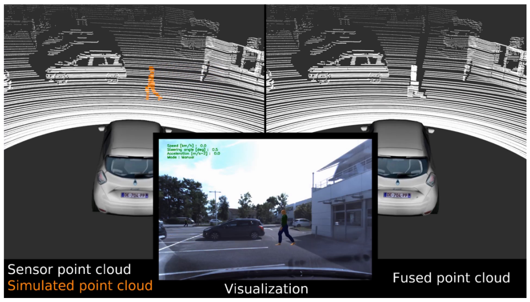


Fig. 8: Illustration of the augmented reality testing technique with an additional virtual pedestrian. On the left, the Point cloud generated by sensors and the point cloud of virtual objects. In the center, the Visualization of the scene. On the right, the Augmented reality point cloud which is directly used by the perception software.

While simulated experiments are less meaningful and real experiments too dangerous, experiments with AR present an interesting compromise. As in simulation, it allows to realize test scenarios that can include collisions with one or several objects. The motion of the agents can be defined exactly upon the test requirements and is repeatable. As in actual experiments, perception provides dense and complex data while localization is slightly uncertain.

Even if the simulated data in AR may be less representative than actual data, it is assumed that it will not affect the experiments since perception is not the purpose of this work. Upon these considerations, it appears that this testing environment is relevant to evaluate the behavior of the MPPI based motion planning in highly realistic conditions.

E. Real Experimental Results

The results for real experiments are shown in Figure 9. On the left, the augmented reality includes the virtual objects to the vehicle camera and on the right the virtual and real objects perception and PCIR. The red band represents the ET-MPPI mitigation trajectory. As in simulation, target selection presents the pedestrian as the most vulnerable road user and a collision with the parked truck on the right brings the least global injury risk.

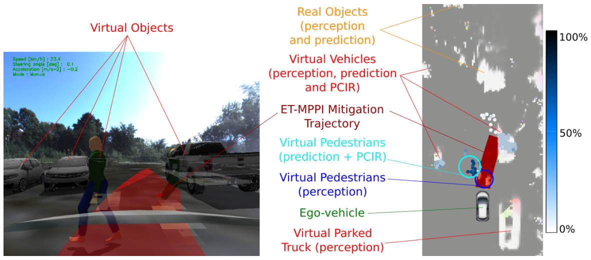

Virtual Objects
Real Objects (perception and prediction)
Virtual Vehicles (perception, prediction and PCIR)
ET-MPPI Mitigation Trajectory
Virtual Pedestrians (prediction + PCIR)
Virtual Pedestrians (perception)
Ego-vehicle
Virtual Parked Truck (perception)
100%
50%
0%

Fig. 9: Scenario (c) real testing. On the left, camera view with objects in Augmented Reality. on the right, perception, PCIR for object prediction and ET-MPPI mitigation trajectory

The virtual objects setup allowed real testing considering collision while keeping driver security and prototype integrity. Terrain noise affected the result due to false-positive detection. However, due to replanning we noticed that once the perception grid filters the noise a new trajectory is promptly generated.

Another remark is related to object prediction. We have noticed a small delay that might be the result of GPU usage for perception, virtual object simulation and MPPI algorithm combined.

Considering all challenges of real time experiments, the ET-MPPI algorithm has produced results consistent with simulation for generating collision-free trajectories when possible or mitigating the global probability of injury for a collision in mitigation scenarios.

V. Conclusions

The present work presents a novel motion planning technique for vehicle navigation on risky environments considering active mitigation. Trajectory planning is based on the classical MPPI framework, which is significantly improved to contemplate a probabilistic dynamic cost map and global probability of injury constraints.

Results show that trajectory generations complies with Euro NCAP 2025+ requirements and with realistic driving scenarios by providing an AEBSS scenario based framework which generates collision-free trajectories when possible or computes active mitigation target selection based on preexisting accidentology data.

Virtual objects are introduced in our work as an alternative to conduct safe real experiments. By merging virtual sensor data to actual real data to virtually present objects in the scene, which allowed us to carry out real experiments without putting in risk the driver or menacing prototype integrity.

One limitation of the study is the use of constant velocity objects in simulation. A suggestion for improvement includes a more liable simulation scenario with non-constant velocities objects and pedestrians with random trajectories. Also, for prototype experiments, false-positives on detection interfered with the algorithm output, although the replanning made it possible to correct it once filtering removed the noise.

Future development includes dealing with the mentioned limitation on object tracking to improve object prediction, and noise filtering to improve trajectory planning output stability. It also includes the use of the generated trajectories as set points for a motion controller to be used in a feedback loop in order to observe its feasibility in prototype applications.

VI. Acknowledgment

This work is supported by a CIFRE fellowship from Renault S.A.S. We thank the CHROMA-IRT team for their support in model simulation and integration tools for prototype testing.

References

[1] "Euro ncap launches road map 2025 - in pursuit of vision zero," Euro NCAP, Tech. Rep., 09 2017. [Online]. Available: https://www.euroncap.com/en/press-media/press-releases/euro-ncap-launches-road-map-2025-in-pursuit-of-vision-zero/

[2] H. Wang, Y. Huang, A. Khajepour, Y. Zhang, Y. Rasekhipour, and D. Cao, "Crash mitigation in motion planning for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3313–3323, Sep. 2019.

[3] K. Lee and D. Kum, "Collision avoidance/mitigation system: Motion planning of autonomous vehicle via predictive occupancy map," *IEEE Access*, vol. 7, pp. 52 846–52 857, 2019.

[4] M. Schachner, W. Sinz, R. Thomson, and C. Klug, "Development and evaluation of potential accident scenarios involving pedestrians and aeb-equipped vehicles to demonstrate the efficiency of an enhanced open-source simulation framework," *Accident Analysis & Prevention*, vol. 148, p. 105831, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0001457520316511

[5] J. Kovaceva, A. Bálint, R. Schindler, and A. Schneider, "Safety benefit assessment of autonomous emergency braking and steering systems for the protection of cyclists and pedestrians based on a combination of computer simulation and real-world test results," *Accident Analysis & Prevention*, vol. 136, p. 105352, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0001457519306736

[6] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, "Aggressive driving with model predictive path integral control," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 1433–1440.

[7] H. Kappen, "Linear theory for control of nonlinear stochastic systems," *Physical review letters*, vol. 95, p. 200201, 12 2005.

[8] G. Williams, A. Aldrich, and E. A. Theodorou, "Model predictive path integral control: From theory to parallel computation," *Journal of Guidance Control and Dynamics*, vol. 40, pp. 344–357, 2017.

[9] A. Buyval, A. Gabdullin, and A. Klimchik, "Model predictive path integral control for car driving with dynamic cost map," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 01 2018, pp. 258–264.

[10] P. Drews, G. Williams, B. Goldfain, E. A. Theodorou, and J. M. Rehg, "Aggressive deep driving: Combining convolutional neural networks and model predictive control," in *Proceedings of the 1st Annual Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, S. Levine, V. Vanhoucke, and K. Goldberg, Eds., vol. 78. PMLR, 11 2017, pp. 133–142. [Online]. Available: http://proceedings.mlr.press/v78/drews17a.html

[11] A. Buyval, A. Gabdullin, K. Sozykin, and A. Klimchik, "Model predictive path integral control for car driving with autogenerated cost map based on prior map and camera image," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 2109–2114.

[12] I. S. Mohamed, G. Allibert, and P. Martinet, "Model predictive path integral control framework for partially observable navigation: A quadrotor case study," in *2020 16th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2020, pp. 196–203.

[13] L. A. S. Guardini, A. Spalanzani, C. Laugier, P. Martinet, A.-L. Do, and T. Hermitte, "Employing Severity of Injury to Contextualize Complex Risk Mitigation Scenarios," in *IV 2020 – 31st IEEE Intelligent Vehicles Symposium*. Las Vegas / Virtual, United States: IEEE, 10 2020, pp. 1–7. [Online]. Available: https://hal.inria.fr/hal-02945325

[14] L. Rummelhard, A. Négre, and C. Laugier, "Conditional monte carlo dense occupancy tracker," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems*, Sep. 2015, pp. 2485–2490.

[15] G. Williams, B. Goldfain, P. Drews, K. Saigol, J. Rehg, and E. Theodorou, "Robust sampling based model predictive control with sparse objective information," in *Robotics Science and Systems*, 06 2018.

[16] ISO 26262, "Road vehicles - functional safety," 2018. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:68383:en

[17] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng, "Ros: an open-source robot operating system," in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, may 2009.

[18] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, Sep. 2004, pp. 2149–2154 vol.3.