Scenario-Optimization-Based Velocity Planning of Autonomous Vehicles for Interacting With Pedestrians

Bence Jekl, Zvonimir Dabčević, Balázs Németh, *Member, IEEE*, Branimir Škugor, *Member, IEEE*, and Péter Gáspár, *Member, IEEE*

***Abstract*— This paper presents a velocity planning method for autonomous vehicles (AVs) to guarantee safe interactions with pedestrians at unsignalized crosswalks and with surrounding vehicles on the AV's route. The method is structured within a hierarchical framework that includes robust control, a learning-based component, and a supervisory element. The learning-based component is trained using reinforcement learning techniques to reduce traveling time, minimize control interventions, and set the priority ratio between the AV and pedestrians. The supervisory element employs scenario optimization, using statistical data on pedestrian motions to ensure collision avoidance. A complex game-theory-based pedestrian model is formulated and analyzed in order to evaluate the effectiveness of the proposed velocity planning method. Extensive simulations are performed using the high-precision traffic simulator software SUMO. These simulations evaluate various aspects of the velocity planner, including computation time, traveling time, control interventions, and parameter settings. The results demonstrate the method's ability to achieve real-time implementation while maintaining safety and performance objectives.**

***Index Terms*— Autonomous vehicles, pedestrian modeling, velocity planning, scenario approach, data-driven control.**

I. Introduction and Motivation

ROADS worldwide present significant risks to vulnerable groups, especially pedestrians who make up a considerable share of traffic-related deaths [1]. The World Health Organization (WHO) reports that road traffic accidents claim over 1.19 million lives each year, with pedestrians accounting for 23% of these fatalities, highlighting the critical need for enhanced safety measures at pedestrian crossings to reduce accidents [2]. Autonomous vehicles (AVs) are seen as a potential solution to improve safety. AVs need to accurately predict the actions and intentions of nearby pedestrians in order to autonomously navigate urban streets. Unsignalized crosswalks in urban areas can increase the potential for conflicts due to the lack of traffic signals that prioritize either pedestrians or vehicles [3]. Pedestrians often make spontaneous decisions influenced by changing environmental and situational factors, which makes their behavior challenging to predict with precision. On the other hand, AVs must make real-time decisions based on these dynamic actions, requiring strategies that can effectively anticipate and adapt to pedestrian movements as they unfold. These conflict situations require the analysis of interactions between pedestrians and AVs, and the development of safe, energy-efficient control strategies for AVs.

Recent studies have explored pedestrian-vehicle interactions from several perspectives, including pedestrian and environmental factors, participant communication, and pedestrian intention analysis [4]. Pedestrian motion models take these factors into account to predict intentions, trajectories, or occupancy maps, as detailed in their taxonomy [5]. Model formulations vary depending on the scenario, such as the pedestrian's goal and the number of pedestrians, which also influence their mathematical structure [6]. Pedestrian models for pedestrian-vehicle interactions can be classified into expert-based, data-driven, and hybrid approaches, also known as physics-based or learning-based methodologies [7]. These methods are relevant for analyzing interactions at unsignalized crosswalks and are briefly discussed below.

Data-driven models analyze pedestrian-vehicle interactions using real-world data and circular polar occupancy grid maps [8] to quantify the spatial interactions and intensity based on relative positions and distances. Softmax and max pooling layers are employed to calculate interaction weights, enhancing the understanding of pedestrian-vehicle dynamics. Graph Neural Networks (GNNs) through directed graphs and Graph Convolutional Networks (GCNs) [9], [10], along with image-based models employing ego-centric views [11], advance the modeling of complex interactions, offering a detailed exploration of pedestrian-vehicle dynamics and highlighting the importance of spatial, temporal, and visual data in

Received 2 April 2024; revised 26 July 2024 and 25 November 2024; accepted 31 December 2024. Date of publication 28 January 2025; date of current version 31 March 2025. This work was supported in part by European Union within the Framework of the National Laboratory for Autonomous Systems under Grant RRF-2.3.1-21-2022-00002; in part by the National Research, Development, and Innovation Office (NKFIH) through the Project "Design of Automated Driving Systems Based on Estimation of Wheel-Road Contact Features for Handling Emergency Situations" under Grant 2019-2.1.11-TÉT-2020-00125; and in part by NKFIH through OTKA under Agreement K 135512. The Associate Editor for this article was Y. Tian.
(Corresponding author: Balázs Németh.)

Bence Jekl is with the Institute for Computer Science and Control (SZTAKI), Hungarian Research Network (HUN-REN), 1111 Budapest, Hungary (e-mail: bence.jekl@sztaki.hun-ren.hu).

Zvonimir Dabčević and Branimir Škugor are with the Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb, 10000 Zagreb, Croatia (e-mail: zvonimir.dabcevic@fsb.unizg.hr; branimir.skugor@fsb.unizg.hr).

Balázs Németh and Péter Gáspár are with SZTAKI, HUN-REN, 1111 Budapest, Hungary, and also with the Department of Control for Transportation and Vehicle Systems, Faculty of Transportation Engineering and Vehicle Engineering, Budapest University of Technology and Economics, 1111 Budapest, Hungary (e-mail: balazs.nemeth@sztaki.hun-ren.hu; peter.gaspar@sztaki.hun-ren.hu).

Digital Object Identifier 10.1109/TITS.2025.3531506

pedestrian motion prediction. However, data-driven models are often criticized for their "black-box" nature, making it difficult to understand the reasoning behind their behavior predictions. Additionally, these models heavily rely on data from specific environments, which can introduce biases and reduce their effectiveness in different settings. The need for large and diverse datasets to ensure broad applicability is a significant limitation. This is particularly challenging in environments with widely varying interactions, such as unsignalized crosswalks, where pedestrian behaviors and vehicle interactions can differ greatly depending on factors like time of day, weather conditions, and cultural norms [4].

On the other hand, expert-based approaches avoid the need for extensive, location-specific data and reduce computational demands, because they rely on clear, predefined rules to predict pedestrian motion. These approaches offer clear explanations for predictions, making them more reliable in environments with high uncertainty. Vehicles' impacts on pedestrian motion are described through explicit formulations, such as the forces outlined in the Social Force Model (SFM) [12]. These models incorporate vehicle impacts using functions that decay over distance [13] and vary with direction [14]. In SFMs, the AVs are depicted as circles [15] or ellipses [16] to capture their asymmetric impact on pedestrians, notably highlighting the greater danger posed by the front zones of vehicles [17]. A multiagent model that incorporates the SFM for improved simulations of shared spaces has been introduced in [18], while methods for model calibration and validation, adapted for diverse environments through the integration of social norms have been outlined in [19]. A conflict model designed to enhance safety by facilitating more effective conflict detection and collision prevention using property graphs has been proposed in [20]. However, modeling AV-pedestrian interactions at unsignalized crosswalks using SFM is challenging because it focuses on physical forces rather than strategic decision-making. This approach fails to capture how AVs and pedestrians anticipate and respond to each other's actions in real-time. For example, pedestrians may decide to cross based on their estimation of an AV's velocity and distance, while AVs must adjust their velocity or path based on the pedestrian's motions. This dynamic interplay of decisions is essential for accurately modeling interactions but is not adequately addressed by SFM's force-based approach. This motivates the use of a game theory (GT) framework, as it models these interactions by showing how one agent's decisions affect the other using a payoff matrix.

The advantages of expert-based models support the use of a GT-based pedestrian model in AV control strategies. While GT models do not necessarily require data for parameterization, optimizing their parameters based on real recorded data can enhance their realism. This combined method has been proven to be effective, as demonstrated in [21], resulting in accurate, collision-free trajectory predictions. Real-world data have been used in [22] to investigate and model the behavior of road users in different shared space topologies, showcasing the models' capability to capture authentic interactions. In [23] the focus is on enhancing agent-based modeling for vehicle-pedestrian interactions by integrating GT and decision models sourced from [24], to simulate scenarios that prioritize pedestrian safety and traffic flow. The research validates the model against real-world variability in decision-making, laying a foundation for its application in more complex, multi-agent traffic scenarios. This work has been further developed in [25], which reparameterizes the expectation submodel to include variables, e.g., pedestrian distance to the crosswalk and vehicle/pedestrian acceleration, enhancing the model predictive performance as evidenced by experimental validation against the recorded inD dataset [26]. However, previous GT-based models oversimplify decision-making by assuming constant velocities and limited adaptability, which reduces their realism. Additionally, without optimization using real-world data, these models may not account for the variability in pedestrian crossing behavior, diminishing their accuracy and effectiveness across diverse real-world scenarios. To address these limitations, the GT-based pedestrian model in this paper is enhanced through global parameter optimization based on recorded data, and a refined motion submodel, allowing for more realistic, variable accelerations and velocities.

The literature review shows that much effort has been devoted to pedestrian motion analysis and prediction, along with vehicle-pedestrian interaction formulation, and related AV control design. This paper contributes to these works with a novel AV velocity planning method, which is able to simultaneously handle the motion of a pedestrian on unsignalized crosswalk and the motions of surrounding vehicles on the route of AV. The proposed method has low computational time in real-time implementation. This is achieved through the proposed hierarchical structure that incorporates learning-based and model-based elements along with a supervisor. The learning-based element is trained through reinforcement learning (RL) to reduce traveling time, control intervention, and set the ratio of priority between AV and pedestrian motions under various traffic scenarios. The role of the model-based element, together with the supervisor, is to guarantee collision avoidance under all traffic scenarios. The preliminary design method of the hierarchical planning structure has been presented in [27]. This paper extends that work with the following contributions: (i) a novel method to handle both vehicle-pedestrian and vehicle-vehicle interactions, evaluated by a new GT-based pedestrian model that provides a realistic representation of interactions between a single pedestrian and a single AV at unsignalized crosswalks, (ii) an enhanced supervisor that accounts for the stochastic nature of pedestrian motion using a scenario-based approach, (iii) demonstrated computational efficiency of the hierarchical framework, enabling real-time implementation on AV systems, and (iv) utilization of scenario optimization to ensure collision avoidance while optimizing travel time and minimizing control interventions.

The organization of this paper is as follows. In Section II the hierarchical velocity planning method is formulated, introducing its overall scheme and detailing the supervisor's role in the context of AV-pedestrian interaction. In Section III the training of the learning-based component within the hierarchical structure is discussed, and the training process is evaluated. The effectiveness of the proposed planning method

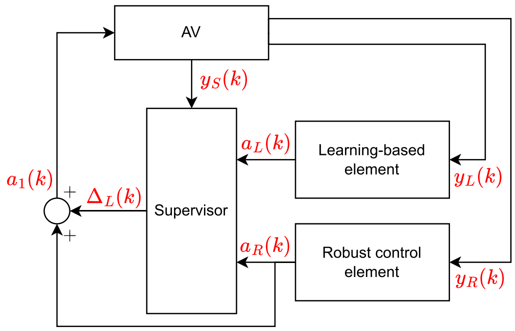


Fig. 1. Scheme of hierarchical velocity planning architecture.

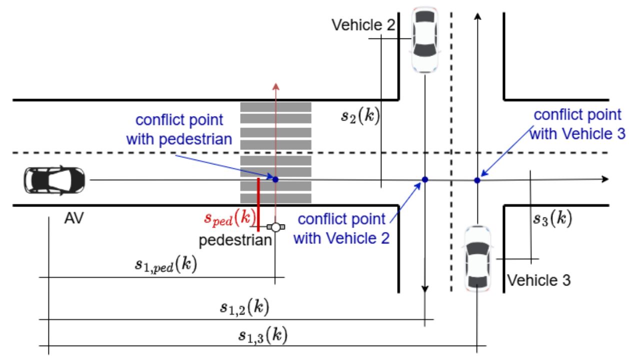


Fig. 2. Illustration of AV-pedestrian-vehicle interaction.

using a complex pedestrian motion model within the SUMO traffic simulator is also demonstrated. The complex GT-based pedestrian model is formulated in Section IV, and the implementation and simulation results are presented in Section V. Finally, the conclusions are summarized in Section VI.

II. Formulation of the Hierarchical Velocity Planning Method

This section presents an overview of the proposed hierarchical strategy and provides the formulation of the supervisor. Figure 1 illustrates the planning architecture scheme, which involves a learning-based element, a robust control element, and a supervisor. During AV planning, all these elements operate simultaneously, and each can use different measured signals on the AV, such as $y_S(k)$, $y_L(k)$, and $y_R(k)$. The control signal for the AV is the longitudinal acceleration $a_1(k)$. It is composed of the outputs of the robust controller $a_R(k)$, and the supervisor $\Delta_L(k)$, as follows:

$$a_1(k) = a_R(k) + \Delta_L(k). \tag{1}$$

The requirement for $a_1(k)$ is to guarantee the safe motion of the AV, i.e., avoiding collisions with pedestrians and surrounding vehicles and adhering to velocity limits. Moreover, the velocity planning must improve non-safety performance levels, such as limiting $a_1(k)$, reducing traveling time, and handling priority between the AV and the pedestrian. These requirements are achieved through the proposed hierarchical structure, where safety requirements are guaranteed by the robust control and the supervisor, and non-safety performance levels are improved through the learning-based element.

In this hierarchical structure, the learning-based element is designed through an RL-based training process that can incorporate the nonlinear motion characteristics of the AV and pedestrian motions. A variety of traffic scenarios can also be included in the training process, making it an appropriate method for improving performance levels. The output of the learning-based element is $a_L(k)$, which is used as an input to the supervisor. In this concept, the signal $a_L(k)$ is supervised according to the safety requirements built into the supervisor. This supervision is assisted by the signal $a_R(k)$, i.e., the supervisor output $\Delta_L(k)$ is limited by the bound $\Delta_L(k) \in [\Delta_{L,min}; \Delta_{L,max}]$, see [27].

It is considered that safety performance through $a_R(k) + \Delta_L(k)$ (see (1)) can always be guaranteed. Moreover, it is possible to find $\Delta_L(k)$ with which the non-safety performance level based on $a_L(k)$ can be improved. It leads to an optimization problem within the supervisor with the objective:

$$\min_{\Delta_L(k)\in[\Delta_{L,min};\Delta_{L,max}]} \Big(a_1(k) - a_L(k)\Big)^2. \tag{2}$$

Consequently, $a_1(k)$ is as close as possible to $a_L(k)$ through the objective, and similarly, the selection of $a_1(k)$ is limited by $a_R(k)$ through the bounds $\Delta_{L,min}$, and $\Delta_{L,max}$. The selection is also constrained by further safety criteria, detailed in the rest of this section.

A. Formulation of the supervisor using a scenario-based approach

The interaction model of the AVs is based on a previously published work [27]. This model is reformulated to the scenario of AV-pedestrian interaction. The goal of the interaction is to formulate the constraint on collision avoidance. This formulation is included in the supervisor.

The interaction model is based on the assumption that the routes of the AV and the pedestrian or surrounding vehicles intersect, as illustrated in Figure 2. These intersections are called conflict points, and the illustrated situation contains 3 conflict points. The distance of the AV and the pedestrian from the conflict point is denoted by $s_i(k)$, where index i is related to the AV ($s_{1,j}$), and j represents the selected conflict point. For pedestrians, the index i is denoted as ($i = ped$), while for surrounding vehicles, the index ranges from ($i = 2 \ldots n_s$), where $n_s - 1$ represents the number of surrounding vehicles. In this interaction model, s_i from the conflict is measured, which means that $s_{1,j} = 0$ represents the position of the AV at the conflict point related to interaction j. Consequently, $s_{1,j} < 0$ means that the AV is approaching the conflict point (e.g., crosswalk $j = ped$ or intersection ($j = 3$)), and $s_{1,j} > 0$ means that the AV is moving away from the conflict point. Similarly, if $s_{ped} < 0$, then the pedestrian is approaching the conflict point, but if $s_{ped} > 0$, then the pedestrian is moving away from it. The interaction model is formed by the following kinematic relationships:

$$v_i(k+1) = v_i(k) + T a_i(k), \tag{3a}$$

$$s_i(k+1) = s_i(k) + T v_i(k) + \frac{T^2}{2} a_i(k), \tag{3b}$$

where T is time step of the discrete motion model, and $v_i(k)$ represents the longitudinal velocity. The control input of the

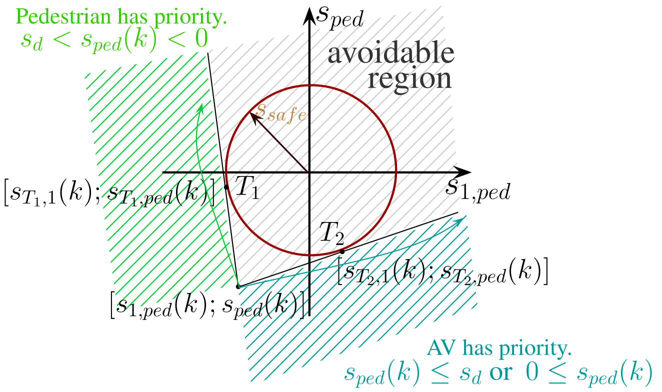


Fig. 3. Geometrical illustration of pedestrian-AV collision avoidance constraints.

AV is its longitudinal acceleration, represented as $a_1(k) = a_R(k) + \Delta_L(k)$.

The formulation of the collision avoidance constraint related to the pedestrian is illustrated in Figure 3. The avoidance of collision can be described by a quadratic expression:

$$\left(s_{1,ped}(k+1)\right)^2 + \left(s_{ped}(k+1)\right)^2 \ge s_{safe}^2, \qquad (4)$$

which ensures that the AV and the pedestrian cannot simultaneously be within a distance of s_{safe} from the conflict point at time step $k+1$. As shown in [27], this quadratic constraint can be reformulated as linear constraints, resulting in the $s_{1,ped}(k) - s_{ped}(k)$ trajectory being realizable in two half-planes. If the pair $[s_{1,ped}(l) = 0; s_{ped}(l) < 0], l > k$ is part of the trajectory, then the AV crosses the conflict point first and the pedestrian second. However, if $[s_{1,ped}(l) < 0; s_{ped}(l) = 0], l > k$ is part of the trajectory, then the pedestrian crosses the conflict point first and the AV second.

The priority between the pedestrian and the AV is fixed in the planning strategy, depending on the position of the pedestrian. If the pedestrian is closer to the conflict point than a predefined distance $s_d < 0$, specifically when $s_d < s_{ped}(k) < 0$, then priority is given to the pedestrian, as represented by the half-space at the bottom (see Figure 3). However, if the pedestrian is further from the edge ($s_{ped}(k) \le s_d$) or leaves the crosswalk ($0 \le s_{ped}(k)$), then the AV has priority, which is represented by the half-space on the left. The constraints in these cases are formed by the tangential lines from $[s_{1,ped}(k); s_{ped}(k)]$ to the circle of the avoidable region. Detailed formulation can be found in [27]. The constraint for each case is formed as follows:

$$\begin{cases} \begin{bmatrix} s_{T_2,1}(k) \\ s_{T_2,ped}(k) \end{bmatrix}^T \begin{bmatrix} -Tv_1(k) - \frac{T^2}{2}a_K(k) \\ -Tv_{ped}(k) - \frac{T^2}{2}a_{ped}(k) \end{bmatrix} \ge \\ \frac{T^2}{2}\begin{bmatrix} s_{T_2,1}(k) \\ 0 \end{bmatrix}^T \Delta_L(k), \quad \text{if } s_d < s_{ped}(k) < 0, \\ \begin{bmatrix} s_{T_1,1}(k) \\ s_{T_1,ped}(k) \end{bmatrix}^T \begin{bmatrix} -Tv_1(k) - \frac{T^2}{2}a_K(k) \\ -Tv_{ped}(k) - \frac{T^2}{2}a_{ped}(k) \end{bmatrix} \le \\ \frac{T^2}{2}\begin{bmatrix} s_{T_1,1}(k) \\ 0 \end{bmatrix}^T \Delta_L(k), \quad \text{otherwise.} \end{cases} \qquad (5)$$

The relationships in (5) require information on the motion of the pedestrian, such as velocity $v_{ped}(k)$ and acceleration $a_{ped}(k)$. Although $v_{ped}(k)$ can be measured by a LiDAR sensor, $a_{ped}(k)$ cannot be directly measured, and it depends on the pedestrian's intention, whether they are accelerating or decelerating. Therefore, $a_{ped}(k)$ is derived from a prediction of a pedestrian model. Since it is not possible to account for all potential $a_{ped}(k)$ values during prediction, a scenario approach is applied to select N possible $a_{ped,z}(k)$ scenarios, with z ranging from 1 to N. This approach transforms the formulation of (5) into N chance constraints in (6), as shown at the bottom of the next page, which is denoted by the compact form in the rest of the paper as:

$$\mathcal{P}(\Delta_L(k), s_{1,ped}(k), z) \le 0, \qquad \forall z = [1; N]. \qquad (7)$$

The incorporation of chance constraints (7) in the optimization problem (2) is carried out using the scenario-based approach. It is used to select scenarios and thus, they can be included in the optimization as chance constraints [28]. Its advantage is that scenario optimization can be effectively applied to convex optimizations with infinite number of constraints [29]. The relationship between stability and probabilistic ingredients in the context of parameter-varying systems in discrete-time is explored by [30]. Scenario optimization can also be formed for solving predictive control problems that are close to the optimization within the supervisor [31].

The scenario approach is applied to minimize the cost in (2) as follows. A random selection of N number of $a_{ped,z}$ acceleration values is used for formulating the constraints in (7). The random selection of $a_{ped,z}$ is based on the probability density function of pedestrian acceleration, see [28]. The selection of $\Delta_L(k)$ is constrained by further conditions resulting from the motion of surrounding vehicles [27]. The constrained optimization problem is formulated as:

$$\min_{\Delta_L(k) \in [\Delta_{L,min}; \Delta_{L,max}]} \quad \Delta_L(k)^2 + 2f^T\Delta_L(k) \qquad (8a)$$

$$\text{subject to} \quad \mathcal{P}(\Delta_L(k), s_{1,ped}(k), z) \le 0, \quad \forall z = [1; N], \qquad (8b)$$

$$\mathcal{F}(\Delta_L(k), i) \le 0, \qquad \forall i = [2; n_s], \qquad (8c)$$

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix} \Delta_L(k) \le \begin{bmatrix} \frac{v_1(k)}{T} + a_K(k) \\ \frac{v_{max} - v_1(k)}{T} - a_K(k) \end{bmatrix} \qquad (8d)$$

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix} \Delta_L(k) \le \begin{bmatrix} a_K(k) - a_{min} \\ a_{max} - a_K(k) \end{bmatrix}, \qquad (8e)$$

where (8c) forms constraint for avoiding collision to $n_s - 1$ number of vehicles in the surrounding. Analysis for selecting n_s can be found in [27]. Eq. (8d) expresses the constraint on avoiding backward motion of the AV, and (8e) is the constraint on control input limits a_{min}, and a_{max}.

The selection of the number of chance constraints N in scenario optimization is highly important because it impacts the size of the optimization problem and the confidence of the solution regarding constraint violation. Two parameters are defined for selecting N: confidence $\nu \in (0; 1)$, and violation $\epsilon \in (0; 1)$. These parameters evaluate the relationship between the chance-constrained solution ($\Delta_L(k)$) of (8) with $N < \infty$, and the theoretical solution with $N \to \infty$. If the relationship

$$N \geq \frac{2}{\epsilon}\left(\ln\frac{1}{\nu} + 1\right) \tag{9}$$

is satisfied, then $\Delta_L(k)$ satisfies all possible constraints with probability no smaller than $1 - \nu$, with at most an ϵ-fraction [28]. Consequently, the trade-off between N and the confidence in $\Delta_L(k)$ can be managed through (9). For example, with increased ν and ϵ parameter selection, the computed $\Delta_L(k)$ can lead to reduced confidence regarding constraint violation of all possible scenarios. However, the resulting size of the scenario optimization problem is reduced, facilitating the numerical solution of (8).

III. Design of the Learning-Based Element for Improving Vehicle Motion Performance

In this section, the training of the learning-based element for the planning system is presented. The goal of the element is to provide $a_L(k)$ to improve the performance level of the controlled vehicle. In this study, the learning-based element is trained using the model-free, online, and off-policy deep deterministic policy gradient (DDPG) reinforcement learning method [32].

The training process involves the entire planning structure (supervisor, robust control) and the environment and thus, the dynamic interaction during training can be evaluated [27]. The dynamic interaction is performed between the vehicle and the pedestrian. The modeling of pedestrian motion is based on the work of [33], which provides a simplified formulation for pedestrian motion and decision-making. The simplified model provides the benefit of reduced training time, which is important for accelerating the training process in the DDPG framework.

During the learning process, safe vehicle motion is guaranteed by the supervisor, along with the simultaneous enhancement of the performance level. The $r(k)$ reward for the learning process is formed as follows:

$$\begin{aligned} r(k) = Q_1 v_1(k) - Q_2|a_L(k)| - Q_3|a_1(k) - a_L(k)| \\ + Q_4 v_{ped}(k), \end{aligned} \tag{10}$$

where Q_1, Q_2, Q_3, Q_4 are scalar weights for tuning the performance level of the controlled system. The role of $Q_1 > 0$ is to improve the velocity of the AV, while Q_4 impacts the improvement of pedestrian velocity. These impacts can oppose each other because increased AV velocity can reduce the pedestrian's willingness to cross, i.e., $v_{ped}(k)$ is reduced. The role of $Q_2 > 0$ is to reduce control intervention. Weight Q_3 impacts the learning process, because if $a(k)$ is closer to $a_L(k)$, the RL-based element imitates the supervisor's operation.

The RL training process is carried out through simulation-based episodes. The goal of the training is to maximize the cumulative reward (10), which is achieved through the evaluation of simulation results in each episode. In the DDPG framework, training is performed in an actor-critic structure. During training, an optimal policy is calculated to maximize the long-term cumulative reward, where finding an optimal policy is equivalent to learning the Bellman equation. The actor and critic approximators use observations to achieve this goal. The actor approximator aims to find the action $a_L(k)$ that maximizes the long-term reward, while the critic approximator estimates the expected value of this long-term reward.

A. Simulation-based analysis and comparison

In the rest of this section, a comparison of the proposed planning system to a stochastic model predictive (SMPC) solution [33] is presented. The goal of the comparison is to show that a similar performance level can be achieved with the proposed hierarchical planning. Moreover, it is shown that the training process can impact the ratio between AV prior to pedestrian and pedestrian prior to AV outcomes.

The environment for training purposes is illustrated in Figure 4. In this case the AV and the pedestrian approach the crosswalk simultaneously. The velocity of the AV is controlled by the proposed structure, and the goal of this environment is to train the learning-based element through the maximization of (10). The environment incorporates the supervisor algorithm (8) to ensure that the resulting element adapts to the operation of the supervisor. Moreover, using the supervisor, collisions between the pedestrian and the AV are avoided throughout the entire training process. In all cases the initial conditions of the pedestrian and the AV are the same as presented in [33]. In all scenarios, the pedestrian starts approaching the crossing from a distance of $s_{ped}(0) = 4$m,

$$\begin{cases} \begin{bmatrix} s_{T_2,1}(k) \\ s_{T_2,ped}(k) \end{bmatrix}^T \begin{bmatrix} -Tv_1(k) - \frac{T^2}{2}a_K(k) \\ -Tv_{ped}(k) - \frac{T^2}{2}a_{ped,z}(k) \end{bmatrix} \geq \\ \frac{T^2}{2}\begin{bmatrix} s_{T_2,1}(k) \\ 0 \end{bmatrix}^T \Delta_L(k), \quad \text{if } s_d < s_{ped}(k) < 0, \\ \begin{bmatrix} s_{T_1,1}(k) \\ s_{T_1,ped}(k) \end{bmatrix}^T \begin{bmatrix} -Tv_1(k) - \frac{T^2}{2}a_K(k) \\ -Tv_{ped}(k) - \frac{T^2}{2}a_{ped,z}(k) \end{bmatrix} \leq \\ \frac{T^2}{2}\begin{bmatrix} s_{T_1,1}(k) \\ 0 \end{bmatrix}^T \Delta_L(k), \quad \text{otherwise}, \end{cases} \quad \forall z = [1; N], \tag{6}$$

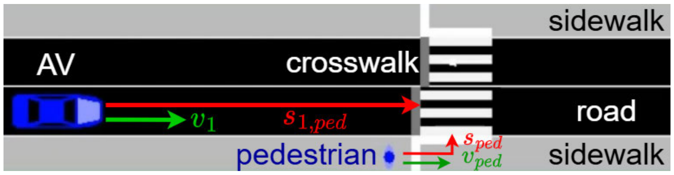


Fig. 4. Training environment setup for AV-pedestrian interaction.

TABLE I
EVALUATION OF THE PROPOSED PLANNING SYSTEM WITH DIFFERENT SETTINGS

Setting of the velocity planning system	AV prior to pedestrian cases (from 100)	Mean of negative accelerations m/s^2	Average traveling time s
SMPC	38	-0.09	7.37
$Q_4 = -200$	51	-1.26	7.25
$Q_4 = -0.3$	26	-0.16	7.83
$Q_4 = 0.3$	14	-0.7272	8.66
$Q_4 = 0.7$	6	-1.66	9.59
$Q_4 = 1$	3	-1.78	9.25

while the initial position and velocity of the AV are varied. The comparison of the proposed method with different Q_4 settings to the SMPC is shown in Table I. In the case of SMPC, there are 38 scenarios out of 100 where the AV crosses before the pedestrian. For the proposed strategy, the parameter Q_4 significantly impacts this ratio; as Q_4 increases, the ratio decreases. The decreased ratio indicates that the AV's motion is more considerate, but it also results in a longer average traveling time. The other weights in (10) remain constant during the training process, with $Q_1 = Q_2 = Q_3 = 1$. Analyses on the impact of these weights on the operation of the AV can be found in [27] and [34]. Table I shows that selecting $Q_4 = -0.3$ provides an optimal balance between various objectives, achieving both low acceleration and reasonable traveling time. Furthermore, the mean negative accelerations are computed for a specific section of AV-pedestrian interaction when the pedestrian has priority and crosses the crosswalk. This scenario, defined as Action 3 of the control strategy in [33], serves as an appropriate basis for comparison.

Figure 5 illustrates the reward function during the training process with the setting $Q_4 = -0.3$. Each episode is randomized with respect to the initial position and velocity of the AV. It is shown that the reward converges to 400 while its deviation continuously decreases.

Finally, two simulation examples illustrating the impact of Q_4 on the velocity and control signals are shown in Figure 6. These simulations correspond to the planner settings $Q_4 = -200$ and $Q_4 = 0.7$. Figure 6a and Figure 6b pertain to the $Q_4 = -200$ setting, highlighting the ratio of scenarios where the AV has priority. These velocity profiles are depicted in green in Figure 6. The results for the $Q_4 = 0.7$ setting are presented in Figure 6c and Figure 6d, demonstrating that the pedestrian has priority in almost all cases. Although the AV experiences increased deceleration compared to the previous case, it does not stop in any of the scenarios. As a result, the traveling time is not significantly increased.

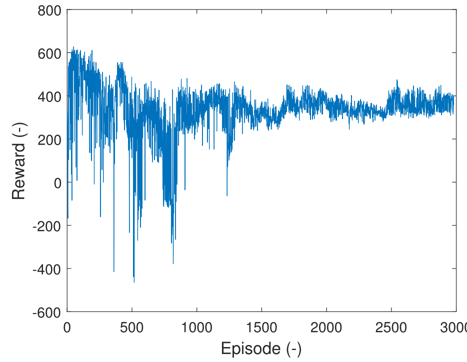
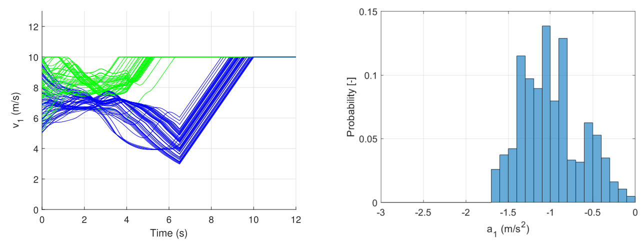
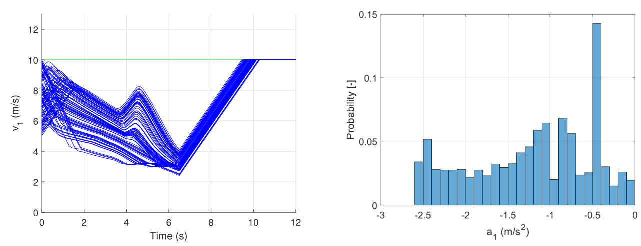


Fig. 5. Convergence of the reward function during the training process.



(a) Velocity ($Q_4 = -200$). (b) Control inputs ($Q_4 = -200$).



(c) Velocity ($Q_4 = 0.7$). (d) Control inputs ($Q_4 = 0.7$).

Fig. 6. Impact of Q_4 settings on velocity and control signals [33].

It can be concluded that the planning strategy's character can be effectively influenced by adjusting the reward function. The proposed training process is effective in the context of pedestrian-AV interaction, and the results achieved are close to those obtained with the SMPC method.

IV. COMPLEX GAME THEORY-BASED PEDESTRIAN MODEL FOR EVALUATION PURPOSES

The remainder of the paper focuses on applying the proposed hierarchical velocity planning strategy to an advanced pedestrian motion model, which is introduced and analyzed in this section. Given that the interaction between vehicles and pedestrians at unsignalized crosswalks is fundamentally seen as a strategic competition for limited time and space, GT is identified as an appropriate modeling method for modeling pedestrian behavior [23], [24]. This approach reduces the need for extensive, location-specific data, minimizes computational demands due to fewer model parameters, and is able to effectively capture pedestrian decision-making processes [24]. Therefore, the previously developed GT-based pedestrian model [25] is expanded through both global optimization of its parameters using a genetic algorithm framework and structural changes to the pedestrian motion submodel, allowing

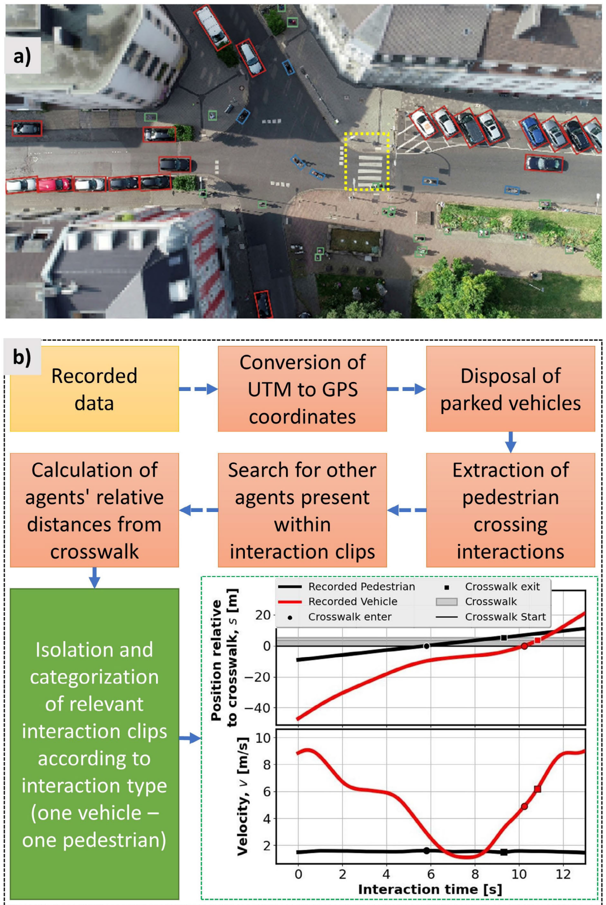


Fig. 7. Aerial view of unsignalized crossing [26] (a) and data processing methodology (b).

for more realistic pedestrian accelerations. The global optimization statistically captures the mean pedestrian behavior from real-world data and generates a 3D Pareto frontier, with each axis representing a distinctive objective function: the first aligns simulated position time profiles with recorded ones, and the second and third minimize the Kullback–Leibler (KL) divergence for velocity and acceleration probability density functions, respectively, ensuring the simulated distributions closely match the recorded ones. Furthermore, unlike simplified model, the GT-based pedestrian model allows for velocity changes and decision-making throughout the entire interaction, capturing strategic behavior and dynamic adaptation of pedestrians.

A. Analysis of Recorded Dataset

The publicly available inD dataset [26] consists of around 9 hours of drone footage capturing interactions at four intersections in Germany. For this analysis, a specific intersection with an unsignalized crossing is chosen, as shown in Figure 7a. This dataset records various signals for different agents, including position (in Universal Transverse Mercator (UTM) coordinates), velocity, acceleration (in both longitudinal and lateral directions), heading angle, and the classification of road users (e.g., pedestrian, vehicle, cyclist).

TABLE II
SUMMARY OF EXTRACTED ONE-TO-ONE INTERACTIONS AND THEIR OUTCOMES FOR IND DATASET

Outcome	**Number of interactions**	**Number of interaction points**
Pedestrian prior to vehicle	29 (52%)	9,945
Vehicle prior to pedestrian	27 (48%)	5,046
Total	56 (100%)	14,991

Preprocessing of this data is required to accurately model pedestrian behavior during vehicle interactions, as outlined in Figure 7b. Initially, the data is transformed from UTM to GPS coordinates. Active interactions are then isolated by removing instances of parked vehicles (identified as those with zero velocity throughout the interaction and positioned beyond the interacting vehicle's path), and focusing solely on those involving a single pedestrian and a single vehicle, while excluding any interactions where additional agents, such as other pedestrians, bicyclists, or vehicles, are present. These interactions are identified when pedestrians begin crossing within 10 meters of the crosswalk, capturing critical driver-pedestrian decisions [35], while also encompassing pedestrian behavior phases necessary for precise modeling [36]. The analysis also notes any additional agents involved in these interactions. The distances of pedestrians and vehicles from the crosswalk are calculated using the GPS coordinates, focusing on encounters between a single pedestrian and a single vehicle to streamline the modeling effort and exclude the impact of extra agents on crossing decisions. The relative distance, transitioning from 2D spatial to 1D metric analysis, is calculated based on recorded GPS points for both pedestrian and vehicle. First, for each individual interaction, the agents' GPS positions at the crosswalk edge are determined. Then, using the Haversine formula [37], the distance is calculated backwards in time by accumulating the incremental distances between consecutive GPS points. Furthermore, to avoid the influence of vehicle type, the vehicle distance relative to the crosswalk edge is calculated with respect to its front point, determined from the recorded center GPS point and vehicle length. The resulting interactions are summarized in Table II, indicating whether pedestrians cross before or after vehicles, showcasing nearly equal instances of both scenarios. The distribution of pedestrian and vehicle velocities for isolated interactions is shown in Figure 8, illustrating the diversity and heterogeneity of recorded motions.

B. Optimization of Pedestrian Model

The GT-based pedestrian model is comprised of four submodels: perception, expectation, decision, and motion. The perception submodel is designed to simulate how pedestrians perceive their surroundings, including their own position and velocity as well as those of approaching vehicles. In reality, these observations are often marred by errors, leading to common misjudgments where the actual velocity and distance of vehicles might be inaccurately assessed by pedestrians. To replicate these inaccuracies, random errors are introduced into the genuine values of vehicle velocity (v_{veh}) and distance relative to the beginning of the crosswalk (s_{veh}). The degree

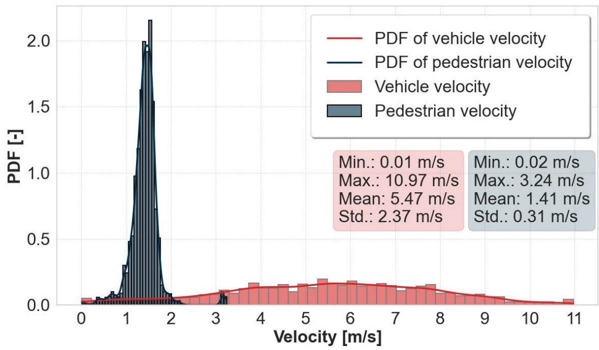


Fig. 8. Recorded vehicle and pedestrian interaction velocity probability density functions (PDF).

to which these real values are modified is determined by a perturbation parameter ε. Herein, the observations of vehicle velocity ($v_{veh,est}$) and distance $d_{veh,est}$ from the perspective of pedestrian are obtained from the Gaussian distributions: $v_{veh,est} \sim \mathcal{N}(v_{veh}, (\varepsilon \cdot v_{veh})^2)$ and $s_{veh,est} \sim \mathcal{N}(s_{veh}, (\varepsilon \cdot s_{veh})^2)$.

Information is received from the perception model by the pedestrian's expectation submodel, through which the probabilities of the vehicle's next actions, either crossing or yielding, are calculated. A binary logistic regression model is utilized due to its effectiveness in handling binary outcomes, incorporating several explanatory variables. These include the pedestrian's distance (s_{ped}), velocity (v_{ped}), and acceleration (a_{ped}); the estimated distance ($s_{veh,est}$) and velocity ($v_{veh,est}$) of the vehicle; the vehicle's acceleration (a_{veh}); and a binary variable indicating the vehicle's lane position (L_{veh}), with $L_{veh} = 1$ signifying a lane further from the pedestrian and 0 indicating a closer lane. The calculation of probabilities is based on a linear combination of these variables, together with an intercept term (β_0) [25]:

$$\begin{aligned} U = \beta_0 + \beta_1 s_{ped} + \beta_2 v_{ped} + \beta_3 a_{ped} + \beta_4 s_{veh,est} \\ + \beta_5 v_{veh,est} + \beta_6 a_{veh} + \beta_7 L_{veh}. \end{aligned} \tag{11}$$

This combination is then input into the sigmoid function to yield a probability value (p) within the range of [0, 1]:

$$P(y = 1|U) = p_{veh,c} = \frac{e^U}{(1 + e^U)}. \tag{12}$$

Here, β_i are the model parameters, with $p_{veh,c}$ representing the probability of a crossing decision ($y = 1$). The probability of a yielding decision is subsequently derived as $p_{veh,y} = 1 - p_{veh,c}$, ensuring all outcomes are accounted for.

To further introduce uncertainty into the model, each parameter β_i) is modified by adding random noise. This adjustment involves a random value κ drawn from a uniform distribution within the [0,1] interval and a perturbation parameter ε, affecting β_i as follows:

$$\beta_i = \beta_i (1 + (1 - 2\kappa)\varepsilon). \tag{13}$$

This addition of noise aims to reflect real-world uncertainty in decision-making processes, making the model more robust by accounting for variations in pedestrian and vehicle behavior.

In the decision submodel, the prospect values for a pedestrian's actions, specifically crossing ($V_{ped,cross}$) and yielding ($V_{ped,yield}$), are calculated using cumulative prospect theory. These values are derived from the probabilities provided by the expectation submodel:

$$V_{\text{ped,cross}} = w^-(p_{\text{veh,c}})v(b_{11}) + w^+(p_{\text{veh,y}})v(b_{22}), \tag{14}$$

$$\begin{aligned} V_{\text{ped,yield}} = \; & w^-(p_{\text{veh,c}})v(b_{12}) \\ & + \left(w^-(p_{\text{veh,c}} + p_{\text{veh,y}}) - w^-(p_{\text{veh,c}})\right) v(b_{22}). \end{aligned} \tag{15}$$

Here, w^- and w^+ are the cumulative probability weighting functions that adjust the perceived probability of losses and gains, v is the value function, and b_{11}, b_{12} and b_{22} represent the payoff values for different agent decision combinations [23]. The weighting and value functions used in (14) and (15) are defined as:

$$w^+(p) = \frac{p^\gamma}{(p^\gamma + (1-p)^\gamma)^{\frac{1}{\gamma}}}, \quad w^-(p) = \frac{p^\delta}{(p^\delta + (1-p)^\delta)^{\frac{1}{\delta}}} \tag{16}$$

$$v(x) = \begin{cases} x^\alpha, & x \geq 0 \\ -\lambda(-x)^\beta, & x < 0 \end{cases} \tag{17}$$

where α, β, γ, δ, and λ are parameters for the weighting and value functions. By relying on cumulative prospect theory and optimizing parameters to fit real-world data, bounded rationality is inherently accounted for in the model [38]. The final decision between pedestrian crossing or yielding decision is made by comparing two calculated values: $V_{ped,cross}$ and $V_{ped,yield}$. The pedestrian decides to cross if $V_{ped,cross} > V_{ped,yield}$; otherwise, they opt to yield. Additionally, it is adjusted that the calculated prospect values, $V_{\text{ped,cross}}$ and $V_{\text{ped,yield}}$, are further modified by a perturbation parameter ε, introducing uncertainty into the decision-making process. This adjustment is made by multiplying each prospect value by $(1 + (1 - 2\kappa)\varepsilon)$.

Based on the calculated prospect values, the pedestrian's dynamics in the current time step k are determined by the motion submodel. The pedestrian will yield if $V_{ped,yield} \geq V_{ped,cross}$, resulting in a deceleration given by:

$$a_{ped}(k) = \frac{-v_{ped}(k)^2}{2|s_{ped}(k)|}. \tag{18}$$

Conversely, if $V_{ped,yield} < V_{ped,cross}$, the pedestrian decides to cross, and acceleration is determined by several factors: the estimated vehicle arrival time at the crosswalk to anticipate conflicts, the pedestrian's current velocity and position for necessary adjustments, and predefined velocity and acceleration limits for realism. By incorporating these elements, the model considers a wide range of possible pedestrian accelerations to simulate realistic behavior:

$$a_{ped}(k) = \min\left(\max\left(\frac{v_{tar}(k) - v_{ped}(k)}{\frac{s_{veh}(k)}{v_{veh}(k)} \cdot \mu}, 0\right), a_{ped,max}\right), \tag{19}$$

where $v_{ped}(k)$ is the pedestrian velocity, $s_{veh}(k)$ is the vehicle position relative to the crosswalk edge, $v_{veh}(k)$ is the vehicle

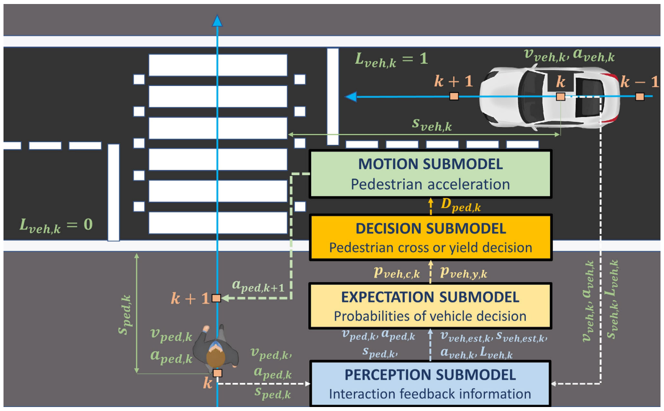


Fig. 9. Pedestrian crossing behavior model.

velocity, μ is a motion model parameter for time adjustment, and $a_{ped,max}$ is the maximum pedestrian acceleration. The target pedestrian velocity $v_{tar}(k)$ is calculated at the k-th time step as follows:

$$v_{tar}(k) = \max\left(v_{ped,min}, \min\left(\frac{2\left(|s_{ped}(k)| + CW_{height}\right)}{\frac{s_{veh}(k)\cdot\mu}{v_{veh}(k)} + \omega} - v_{ped}(k), v_{ped,max}\right)\right) \quad (20)$$

where CW_{height} is the height of the crosswalk (here equal to 5.5 meters), $v_{ped,min}$ and $v_{ped,max}$ are the minimum and maximum pedestrian velocities, respectively, and ω is a small positive constant added for numerical stability.

Finally, the pedestrian state is updated using the following state equations:

$$s_{ped}(k+1) = s_{ped}(k) + v_{ped}(k)\Delta T + a_{ped}(k)\Delta T^2/2, \quad (21)$$
$$v_{ped}(k+1) = v_{ped}(k) + a_{ped}(k)\Delta T, \quad (22)$$

The optimization of the expectation, decision, and motion submodel parameters is conducted to replicate realistic pedestrian crossing behaviors using pedestrian-vehicle data from the inD dataset. The parameters optimized include eight for the expectation submodel (β_0 to β_7), nine for the decision submodel ($b_{11}, b_{12}, b_{21}, b_{22}, \alpha, \beta, \gamma, \delta$, and λ), and five for the motion submodel (time reserve factor upon crossing decision μ, minimum velocity threshold $v_{ped,min}$, maximum pedestrian velocity $v_{ped,max}$, and limits on maximum acceleration $a_{ped,max}$ and deceleration $a_{ped,min}$), where the perturbation parameter ε is set to zero.

In this process, three objective functions are minimized. The first function aligns the simulated pedestrian positions with the recorded data:

$$J_1 = \sum_{i=1}^{M} \frac{1}{P} \sum_{k=1}^{P} (s_{sim,i,k} - s_{rec,i,k})^2, \quad (23)$$

where M is the count of vehicle-pedestrian interactions in the training set, P is the number of discrete time steps for each interaction, $s_{sim,i,k}$ represents the simulated pedestrian position, and $s_{rec,i,k}$ is the corresponding recorded position at each time step k within interaction i.

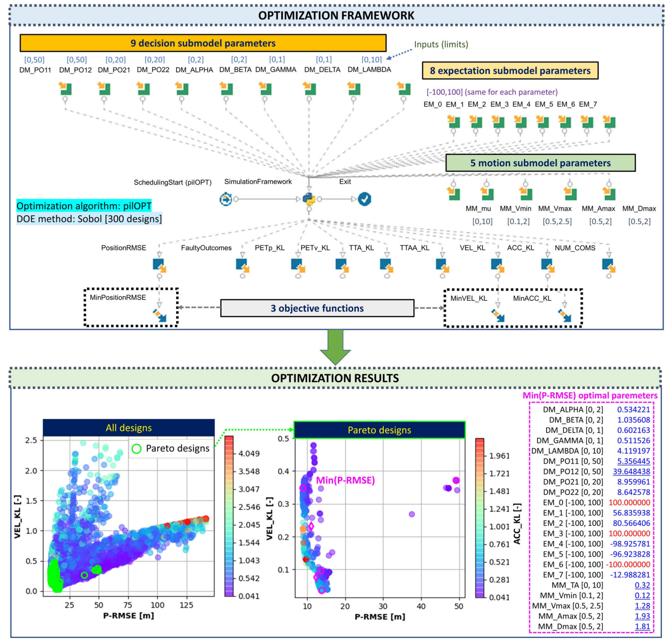


Fig. 10. Schematic of pedestrian model optimization framework realized in modeFRONTIER software.

The second and third functions aim to minimize the KL divergence for velocity and acceleration probability density functions, ensuring the simulated distributions closely match the recorded data:

$$J_2 = D_{KL}(PDF_{VEL,rec}, PDF_{VEL,sim}), \quad (24)$$
$$J_3 = D_{KL}(PDF_{ACC,rec}, PDF_{ACC,sim}). \quad (25)$$

Here, D_{KL} represents the KL Divergence Measure [39], $PDF_{VEL,rec}$ and $PDF_{VEL,sim}$ are the recorded and simulated probability density functions of pedestrian velocities, respectively, while $PDF_{ACC,rec}$ and $PDF_{ACC,sim}$ are the recorded and simulated probability density functions of pedestrian accelerations.

The optimization framework is implemented within the modeFRONTIER 2018R2 software and is interfaced with Python version 3.7, as illustrated in Figure 10. Simulations are conducted in the Python environment, where objective functions are assessed. Parameter adjustments are performed by the modeFRONTIER software, which utilizes the pilOPT optimization algorithm in response to the simulation outcomes. The Sobol Design of Experiments (DOE) method with 300 designs is employed. Results from the 23,017 Genetic Algorithm iterations are displayed as a 3D Pareto frontier, with each axis representing an objective function, and the acceleration KL divergence is denoted by a color scale. The final parameter selection is based on the identification of the minimal position Root Mean Square Error (RMSE) on the Pareto frontier, ensuring the model alignment with the position data is optimized (see (23)). In Figure 11, the model results are presented by comparing simulated and recorded interactions. The pedestrian RMSE between recorded and simulated trajectories is shown in Figure 11a, where each value represents the RMSE calculated for all interaction points of a single interaction. The mean value is equal to 0.16 m,

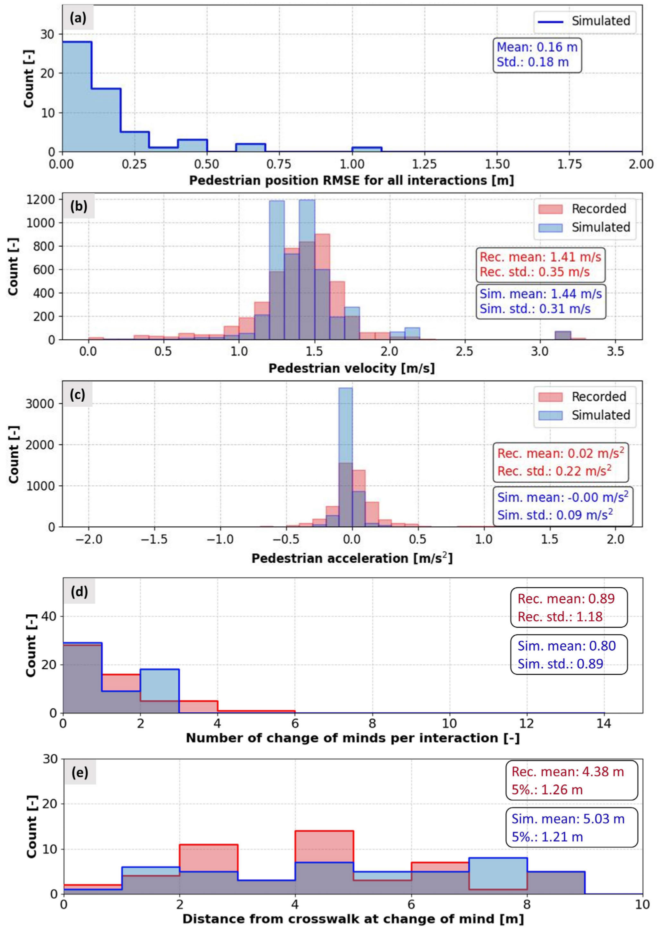


Fig. 11. Comparison of simulated and recorded datasets: pedestrian position root mean squared error (RMSE) (a), velocity distributions (b), acceleration distributions (c), distributions of change of mind events (CoMs) (d), and distances from crosswalk at which CoMs occurred (e).

indicating a good match between recorded and simulated trajectories. Furthermore, the distributions of recorded and simulated velocities are shown in Figure 11b for all interaction points. The means are 1.41 m/s for recorded and 1.44 m/s for simulated, with standard deviations of 0.35 m/s and 0.31 m/s, respectively, showing similar distributions. Although the simulated pedestrian acceleration distribution has a higher peak around 0 m/s^2 compared to the recorded distribution as illustrated in Figure 11c, most of the simulated acceleration values closely align with the recorded values, demonstrating the similarity between the distributions. To gain insights into a pedestrian decision-making process, a change of mind (CoM), defined as a time instant when switching from one decision to another, i.e., from yield to cross and vice versa, is analyzed. In recorded data, underlying pedestrian decisions governing pedestrian motion are not directly available and thus should be reconstructed from recorded motion data. It is done here by detecting shifts where a pedestrian, at any point in the interaction, noticeably accelerates (above 0.1 m/s^2) and at any later time decelerates (below -0.1 m/s^2), or vice versa, indicating a shift in movement intent and a change in decision. On the other hand, in simulations, pedestrian decisions are readily available in each time step as an output of the decision model (14) and (15). The processed results show that a mean time between two CoMs in recorded data is 0.89 s with

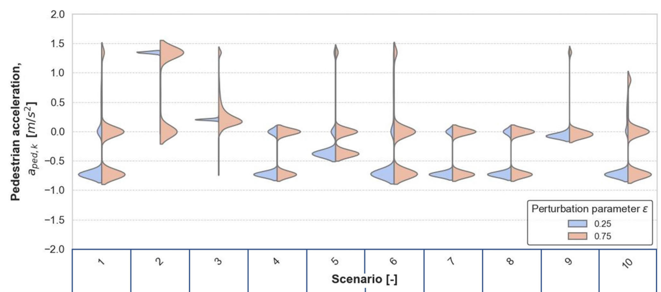
TABLE III
PEDESTRIAN MODEL TRAINING AND VALIDATION RESULTS

Dataset	J_1/M [m]	Faulty outcomes [-]	J_2 [-]	J_3 [-]
Training	0.158	0 / 56	0.349	0.527
Validation	0.171	0 / 4032	0.280	0.488

a standard deviation of 1.18 s, while simulations exhibit a slightly lower mean of 0.80 s (-10%) and a standard deviation of 0.89 s (-25%), which may still be considered very close to recorded ones (see Figure 11d). Furthermore, to determine when a clear crossing decision is made, the distances of all CoM events from the start of the crosswalk are analyzed. As shown in Figure 11e, the distribution of these distances reveals that 95% of CoM events occur at least 1.20 m from the crosswalk edge, indicating that pedestrians in both recorded and simulated scenarios generally commit to a decision before reaching the area near the crossing point. This demonstrates the model accuracy in capturing pedestrian decision dynamics with minimal strategy changes.

The pedestrian model is validated using the same dataset of recorded pedestrian-vehicle interactions used for training and optimization due to the limited number of interactions. To ensure unbiased evaluation and generalization, data augmentation is applied by taking 72 different initial starting times for each interaction in the simulation. Therefore, only the pedestrian features at the initial simulation time step $k = 0$ (position $s_{ped,0}$, velocity $v_{ped,0}$, and acceleration $a_{ped,0}$) are taken from the recorded data, while their subsequent motions ($k > 0$) are simulated (see Figure 9). In contrast, all vehicle features are recorded throughout the simulation to enable comparison between recorded and simulated outcomes. The results, shown in Table III, indicate that the validation dataset has a similar position RMSE per interaction J_1/M (see (23)) to the training set. Noticeably, lower J_2 (KL divergence of velocity distributions, see (24)) and J_3 (KL divergence of acceleration distributions, see (25)) values for validation indices are observed, likely due to the much larger number of data samples used for reconstructing distributions/histograms. Importantly, there are no faulty outcomes (the outcome of pedestrian crossing prior or after the vehicle is the same in simulation as in recorded interactions). Despite similar performance across various training and testing scenarios, it should be noted that the model still may underperform in extreme situations that deviate significantly from typical ones which may be mitigated by employing a larger dataset for model development.

To demonstrate the uncertainty in pedestrian behavior, the perturbation parameter in the optimized pedestrian model is set to non-zero values, specifically to show its impact. Ten different pedestrian-vehicle scenarios are established, each featuring distinct pedestrian and vehicle attributes and occurring near a crosswalk where pedestrian behavior tends to vary the most (refer to Figure 12). These scenarios capture a single discrete time step k. For each scenario, 10,000 simulations are run to ascertain the pedestrian acceleration state $a_{ped,k}$, with perturbation parameters ε assigned values of 0.25 and



	1	2	3	4	5	6	7	8	9	10
$s_{ped,k}$ [m]	-0.72	-1.53	-1.32	-1.41	-1.13	-1.56	-1.33	-0.62	-1.83	-1.51
$v_{ped,k}$ [m/s]	1.51	0.12	1.28	2.25	0.91	1.47	2.22	1.82	0.53	1.64
$s_{veh,k}$ [m]	-1.64	-0.60	-0.94	-1.25	-1.42	-1.29	-1.47	-1.72	-1.32	-0.90
$v_{veh,k}$ [m/s]	14.36	1.85	0.28	7.30	14.98	12.22	11.74	12.03	13.23	10.63
$a_{veh,k}$ [m/s²]	-0.42	-0.29	1.99	-0.87	2.37	1.40	1.29	-1.74	1.24	1.56

Fig. 12. Pedestrian acceleration distributions in near-crosswalk scenarios under different perturbation parameter ε.

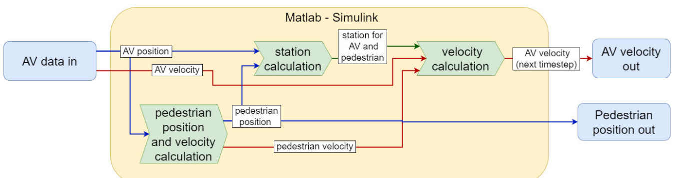


Fig. 13. Communication between Matlab and SUMO using TraCI protocol.

0.75. According to Figure 12, larger perturbation parameters, indicative of greater model uncertainty, can lead to an increase in the number of peaks in the acceleration state distribution. For instance, in scenario 2, a Gaussian distribution emerges when $\varepsilon = 0.25$, but a bimodal distribution occurs at $\varepsilon = 0.75$, representing a pedestrian's decision to either accelerate or maintain velocity. Similarly, in scenario 6, the distribution transitions from bimodal at $\varepsilon = 0.25$ to multi-modal at $\varepsilon = 0.75$, spreading the probability across decelerating, accelerating, or maintaining velocity.

V. Simulation Results With the GT-Based Pedestrian Model

The effectiveness of the proposed velocity planning structure is demonstrated through simulations using the GT-based pedestrian motion model in the SUMO traffic simulator [40]. This section presents the implementation details and provides two simulation examples showcasing the system's operation.

A. Implementation of the Planning in SUMO Simulator

The SUMO simulator is interconnected with Matlab to influence AV velocity and pedestrian position. The communication between Simulink and SUMO is implemented using the TraCI protocol [41]. The communication architecture is illustrated in Figure 13. The blue rectangles illustrate the data sent through the TraCI protocol. The Simulink model reads the AV position, velocity, and orientation values from SUMO to calculate the pedestrian's position and orientation. The constraints (6) are based on the distances $s_{1,ped}(k)$ and $s_{ped}(k)$ between the AV, pedestrian, and the conflict point on the crosswalk, which are used to compute the control input $a_1(k)$ of the AV. This computation step is performed in Matlab.

The GT-based pedestrian model is developed in Python, which communicates with Matlab to achieve low computation time during simulations. The time to solve the equations

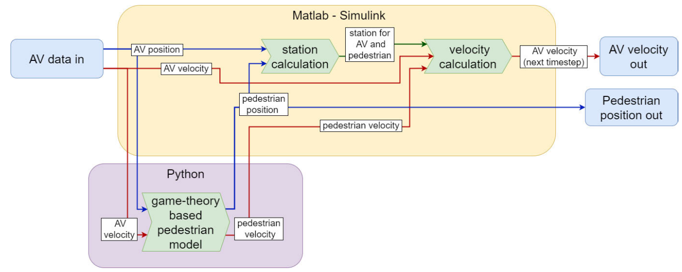


Fig. 14. Communication flow between Matlab and Python for integrating SUMO simulator.

of motion for the GT-based pedestrian model in one computational step is 0.001s. In this case, the AV position and velocity received from SUMO are forwarded to Python through the TCP/IP protocol. The pedestrian model calculates the pedestrian's position and velocity based on this information, as shown in Figure 14.

B. Simulation Results on AV-Pedestrian Interaction

The implementation within the SUMO environment is utilized for evaluation purposes, specifically in the scenario illustrated in Figure 4. For evaluation purposes, 100 simulations have been conducted with random initial positions and velocities for the GT-based pedestrian model. In each scenario, the vehicle starts with an initial velocity of 0 m/s. The value of s_{safe} in these simulations is set to 2 m. The simulations have three objectives: assessing the impact of N on AV motion, examining the computational time of the hierarchical velocity planning, and illustrating the results in the time domain. The pedestrian motion scenarios for the scenario optimization are generated using the GT-based pedestrian model with the inD dataset.

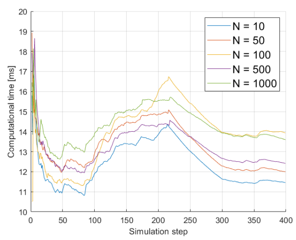
The impact of N on the simulation results is summarized in Table IV. In the simulations, the scenario-based velocity planning is set with $\epsilon = 0.05$, see (9). As the number of constraints increases, the confidence value ν also rises, as shown in the first and fourth columns of Table IV. It is also evident that the actual confidence value during the simulations exceeds the selected ν, as indicated in the fifth column of Table IV. For instance, with $N = 1000$ samples, 95% $(1 - \epsilon)$ of the constraints (6) are met with a 99.35% probability, surpassing the expected 99.2%. Although the actual confidence value is less than 100%, no collisions between the pedestrian and the AV occurred. Another conclusion of the analysis is that varying N has a low impact on the priority between the AV and the pedestrian, with values ranging between 30 and 41. This range is slightly higher than the results obtained with the simplified pedestrian model (see Table I). Additionally, the mean negative acceleration values are nearly identical and remain low.

The analysis of the impact of constraint number shows that high confidence can be achieved with high N. However, increasing the number of constraints also enlarges the size of the optimization problem (8). Therefore, the implementation of the algorithm requires an analysis of the computation time, ensuring that the computation time of the planning algorithm is less than the simulation step time of 0.05s. During each

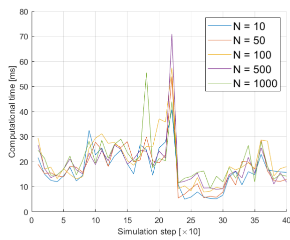
TABLE IV

ANALYSIS ON THE IMPACT OF CONSTRAINT NUMBER

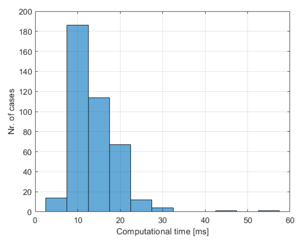
Number of constraints N	Vehicle prior to pedestrian cases (from 100)	Mean of negative accelerations m/s^2	Confidence value (ν) using (9) with $\epsilon = 0.05$	Real confidence value
50	36	-0.659	0.84	0.9834
100	41	-0.643	0.92	0.9817
500	30	-0.673	0.984	0.9862
1000	39	-0.665	0.992	0.9935



(a) Average computational time for all N values.

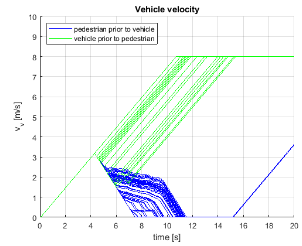


(b) Maximum computational time for all N values.

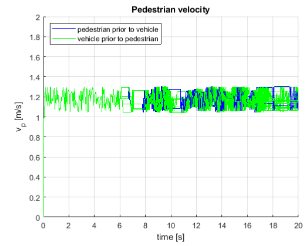


(c) Statistics on computational time for $N = 1000$.

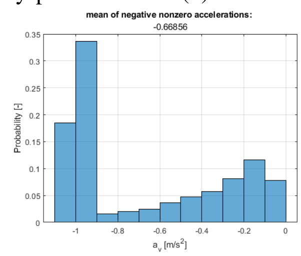
Fig. 15. Analysis of computational time.



(a) AV velocity profiles.



(b) Pedestrian velocity profiles.



(c) AV acceleration distribution.

Fig. 16. Simulations results of AV and pedestrian interactions using the GT-based pedestrian model.

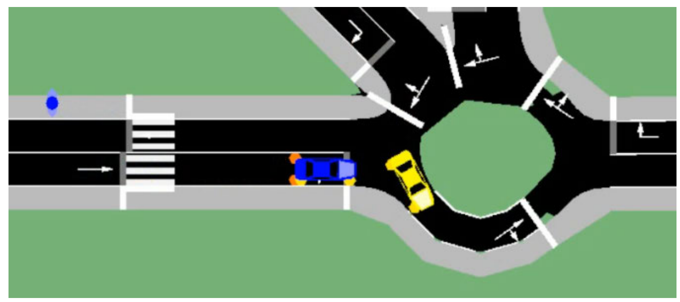


Fig. 17. Illustration of traffic scenario involving AV, pedestrian, and another vehicle at roundabout.

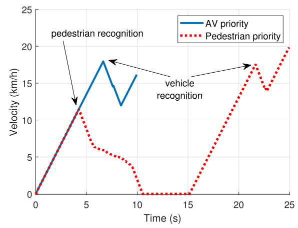
simulation, the computational time has been measured in Matlab. At the end of the simulation, this data has been post-processed to calculate the average computational time. The results are shown in Figure 15. The trend indicates that as N increases, the average computational time also increases (see Figure 15a). Figure 15b shows the maximum computational time, indicating that the impact of N on time is also limited. Around the 220^{th} simulation step, the computational time increases significantly. This is because the pedestrian and the AV are closer to each other, making it more challenging to find a solution to (8). After this point, either the AV or the pedestrian leaves the crosswalk, simplifying the calculation because there is no pedestrian nearby. Figure 15c shows the distribution of computational time values for $N = 1000$. In most cases, the computational time is between 0.01s and 0.02s, but there are rare outliers where the time increases up to 0.055s. Although this is higher than the simulation step time, real-time computation can be achieved in the majority of simulations.

The proposed analyses show that $N = 1000$ is able to provide high confidence and acceptable computational time. Therefore, the results of this scenario are illustrated in the time domain as shown in Figure 16. Figure 16a demonstrates that the velocity profiles of the AV differ significantly in the 39 scenarios where the AV has priority compared to the other scenarios. This high ratio may positively impact traffic flow, and the pedestrian's motion is not significantly disturbed (see Figure 16b). The need for high deceleration can also be avoided using the proposed velocity planning structure, as illustrated in its distribution in Figure 16c.

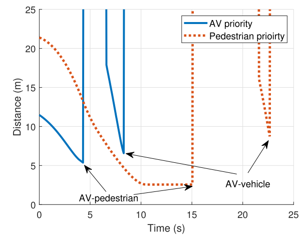
C. Simulation Results on a Complex Scenario

Finally, the effectiveness of the proposed velocity planning strategy is demonstrated through simulation examples involving another vehicle. In these examples $n_s = 4$ is selected. The goal of the simulations is to show that the AV can adapt to more complex situations, such as interacting with pedestrians and surrounding vehicles. A screenshot of the scenario is shown in Figure 17, where the AV moves to a roundabout after leaving the crosswalk. Entering the roundabout is determined by the motion of another vehicle. It is assumed that the position and velocity of the other vehicle are available for handling the interaction [27].

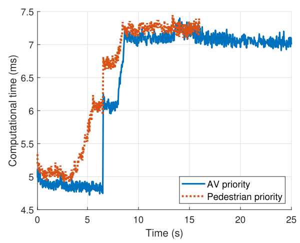
The results of two simulations related to the scenario are presented, distinguished by the priority between the AV and the pedestrian. Figure 18a shows the velocity profiles for both cases. In each scenario, the vehicle starts from a standing position and accelerates. In the second case, the acceleration transitions to deceleration when the pedestrian is recognized. However, in the first case, the AV maintains its priority and continues to accelerate. The recognition of another vehicle



(a) AV velocity profiles.



(b) Minimum distance values.



(c) Computational time.

Fig. 18. Simulation results of complex scenario involving AV, pedestrian, and another vehicle.

in the roundabout occurs at different times, specifically at $7s$ and $22s$. In both cases, the AV is required to decelerate slightly to avoid a collision with the other vehicle. In Figure 18b, the minimum distance between the AV and the pedestrian or the other vehicle is shown. It is observed that the minimum values are achieved when the AV or the pedestrian surpasses the crosswalk and when the other vehicle moves in front of the AV. Figure 18b confirms that collisions are avoided in all cases. Finally, the computational times for the hierarchical strategy are illustrated in Figure 18c. The results show that the computation time is significantly less than the 0.05s simulation step time, indicating that the velocity planning computations can be performed in real-time.

VI. Conclusion

The presented simulations illustrate the effectiveness of the proposed velocity planning method for AVs, i.e., the method is able to guarantee collision avoidance and optimize performance metrics such as traveling time and control intervention. It has been shown that the interactions between the AV and the pedestrians, the surrounding vehicles can be effectively influenced by the tuning parameter selection in the scenario optimization and in the reward of RL. The analysis of the computational time for the provided velocity planning has shown that it is possible to implement the algorithm in real-time. Moreover, the formulated complex GT-based pedestrian model can be effectively used for evaluating the proposed velocity planning strategy through simulations in the SUMO traffic simulator. It is shown that the proposed parameter selection method results in simulated pedestrian motion that is close to the real pedestrian motion recorded by the dataset.

One of the future challenges of the proposed method is to extend the hierarchical strategy to consider more than one pedestrian. This extension could cover more configurations, such as multiple individual pedestrians moving along the vehicle's route or multiple pedestrians moving on the same crosswalk. Due to the variety of configurations, the learning-based element and the scenario optimization will need to be reformulated to accommodate the actual situation. Furthermore, the effectiveness of the method has provided an appropriate starting point for its implementation on a real test vehicle. However, disturbances in the measured signals, such as the position and velocity of the pedestrian, must be considered. This may require an extension of the scenario optimization to achieve robustness against disturbances.

References

[1] J. E. Rod, O. Oviedo-Trespalacios, T. Senserrick, and M. King, "Older adult pedestrian trauma: A systematic review, meta-analysis, and GRADE assessment of injury health outcomes from an aggregate study sample of 1 million pedestrians," *Accident Anal. Prevention*, vol. 152, Mar. 2021, Art. no. 105970.

[2] *Global Status Report on Road Safety 2023: Summary*, World Health Organization, Geneva, Switzerland, 2023.

[3] Y. Yang, Y. Wang, S. M. Easa, and X. Zheng, "Analyzing pedestrian behavior at unsignalized crosswalks from the drivers' perspective: A qualitative study," *Appl. Sci.*, vol. 12, no. 8, p. 4017, Apr. 2022.

[4] A. Rasouli and J. K. Tsotsos, "Autonomous vehicles that interact with pedestrians: A survey of theory and practice," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 3, pp. 900–918, Mar. 2020.

[5] M. Gulzar, Y. Muhammad, and N. Muhammad, "A survey on motion prediction of pedestrians and vehicles for autonomous driving," *IEEE Access*, vol. 9, pp. 137957–137969, 2021.

[6] F. Camara et al., "Pedestrian models for autonomous driving—Part II: High-level models of human behavior," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 9, pp. 5453–5472, Sep. 2021.

[7] M. Golchoubian, M. Ghafurian, K. Dautenhahn, and N. L. Azad, "Pedestrian trajectory prediction in pedestrian-vehicle mixed environments: A systematic review," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 11, pp. 11544–11567, Nov. 2023.

[8] H. Cheng, W. Liao, M. Y. Yang, M. Sester, and B. Rosenhahn, "MCENET: Multi-context encoder network for homogeneous agent trajectory prediction in mixed traffic," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2020, pp. 1–8.

[9] C. Wong, B. Xia, Q. Peng, W. Yuan, and X. You, "MSN: Multi-style network for trajectory prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 9751–9766, Sep. 2023.

[10] S. Carrasco, D. F. Llorca, and M. Á. Sotelo, "SCOUT: Socially-consistent and understandable graph attention network for trajectory prediction of vehicles and VRUs," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jul. 2021, pp. 1501–1508.

[11] K. Kim, Y. K. Lee, H. Ahn, S. Hahn, and S. Oh, "Pedestrian intention prediction for autonomous driving using a multiple stakeholder perspective model," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2020, pp. 7957–7962.

[12] R. Borsche and A. Meurer, "Microscopic and macroscopic models for coupled car traffic and pedestrian flow," *J. Comput. Appl. Math.*, vol. 348, pp. 356–382, Mar. 2019.

[13] D. Yang, Ü. Özgüner, and K. Redmill, "A social force based pedestrian motion model considering multi-pedestrian interaction with a vehicle," *ACM Trans. Spatial Algorithms Syst.*, vol. 6, no. 2, pp. 1–27, Feb. 2020.

[14] D. Yang, Ü. Özgüner, and K. Redmill, "Social force based microscopic modeling of vehicle-crowd interaction," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1537–1542.

[15] M. Prédhumeau, L. Mancheva, J. Dugdale, and A. Spalanzani, "Agent-based modeling for predicting pedestrian trajectories around an autonomous vehicle," *J. Artif. Intell. Res.*, vol. 73, pp. 1385–1433, Apr. 2022.

[16] Z. Zhang and D. Fu, "Modeling pedestrian–vehicle mixed-flow in a complex evacuation scenario," *Phys. A, Stat. Mech. Appl.*, vol. 599, Aug. 2022, Art. no. 127468.

[17] D. Yang, A. Kurt, K. Redmill, and Ü. Özgüner, "Agent-based microscopic pedestrian interaction with intelligent vehicles in shared space," in *Proc. 2nd Int. Workshop Sci. Smart City Operations Platforms Eng.* New York, NY, USA: Association for Computing Machinery, Apr. 2017, pp. 69–74.

[18] F. T. Johora and J. P. Müller, “Modeling interactions of multimodal road users in shared spaces,” in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2018, pp. 3568–3574.

[19] F. T. Johora and J. P. Müller, “On transferability and calibration of pedestrian and car motion models in shared spaces,” *Transp. Lett.*, vol. 13, no. 3, pp. 172–182, Mar. 2021.

[20] S. Hossain, F. T. Johora, J. P. Müller, and S. Hartmann, “A conceptual model of conflicts in shared spaces,” in *Proc. 6th Int. Conf. Ind. Bus. Eng.* New York, NY, USA: Association for Computing Machinery, 2020, pp. 228–235.

[21] F. T. Johora, H. Cheng, J. P. Müller, and M. Sester, “An agent-based model for trajectory modelling in shared spaces: A combination of expert-based and deep learning approaches,” in *Proc. 19th Int. Conf. Auto. Agents MultiAgent Syst.*, 2020, pp. 1878–1880.

[22] F. T. Johora and J. P. Müller, “Zone-specific interaction modeling of pedestrians and cars in shared spaces,” *Transp. Res. Proc.*, vol. 47, pp. 251–258, Jan. 2020.

[23] B. Skugor, J. Topic, J. Deur, V. Ivanovic, and E. Tseng, “Analysis of a game theory-based model of vehicle-pedestrian interaction at uncontrolled crosswalks,” in *Proc. Int. Conf. Smart Syst. Technol. (SST)*, Oct. 2020, pp. 73–81.

[24] P. Chen, C. Wu, and S. Zhu, “Interaction between vehicles and pedestrians at uncontrolled mid-block crosswalks,” *Saf. Sci.*, vol. 82, pp. 68–76, Feb. 2016.

[25] L. Pavelko, B. Skugor, J. Deur, V. Ivanovic, and H. E. Tseng, “Modification and experimental validation of a logistic regression vehicle-pedestrian model,” in *Proc. Int. Conf. Smart Syst. Technol. (SST)*, Oct. 2022, pp. 109–114.

[26] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein, “The inD dataset: A drone dataset of naturalistic road user trajectories at German intersections,” in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Oct. 2020, pp. 1929–1934.

[27] B. Németh and P. Gáspár, “Hierarchical motion control strategies for handling interactions of automated vehicles,” *Control Eng. Pract.*, vol. 136, Jul. 2023, Art. no. 105523.

[28] M. C. Campi, S. Garatti, and M. Prandini, “The scenario approach for systems and control design,” *Annu. Rev. Control*, vol. 33, no. 2, pp. 149–157, Dec. 2009.

[29] G. C. Calafiore and M. C. Campi, “The scenario approach to robust control design,” *IEEE Trans. Autom. Control*, vol. 51, no. 5, pp. 742–753, May 2006.

[30] A. Sala, “Stability analysis of LPV systems: Scenario approach,” *Automatica*, vol. 104, pp. 233–237, Jun. 2019.

[31] J. Matuško and F. Borrelli, “Scenario-based approach to stochastic linear predictive control,” in *Proc. IEEE 51st IEEE Conf. Decis. Control (CDC)*, Dec. 2012, pp. 5194–5199.

[32] T. P. Lillicrap et al., “Continuous control with deep reinforcement learning,” in *Proc. 4th Int. Conf. Learn. Represent. (ICLR)*, San Juan, Puerto Rico, Y. Bengio and Y. LeCun, Eds., May 2016. [Online]. Available: http://arxiv.org/abs/1509.02971

[33] B. Skugor, J. Deur, V. Ivanovic, and H. E. Tseng, “Stochastic model predictive control of an autonomous vehicle interacting with pedestrians at unsignalized crosswalks,” in *Proc. Eur. Control Conf. (ECC)*, Jun. 2023, pp. 1–7.

[34] B. Németh, Z. Farkas, Z. Antal, and P. Gáspár, “Hierarchical control design of automated vehicles for multi-vehicle scenarios in roundabouts,” in *Proc. Eur. Control Conf. (ECC)*, Jul. 2022, pp. 1964–1969.

[35] A. Várhelyi, “Drivers’ speed behaviour at a zebra crossing: A case study,” *Accident Anal. Prevention*, vol. 30, no. 6, pp. 731–743, Nov. 1998. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0001457598000268

[36] A. Gorrini, L. Crociani, G. Vizzari, and S. Bandini, “Observation results on pedestrian-vehicle interactions at non-signalized intersections towards simulation,” *Transp. Res. F, Traffic Psychol. Behaviour*, vol. 59, pp. 269–285, Nov. 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1369847817302000

[37] J. Doe and J. Smith, *Geographical Information Systems: Principles and Applications*, 2nd ed., New York, NY, USA: GeoScience, 2020.

[38] A. Tversky and D. Kahneman, “Advances in prospect theory: Cumulative representation of uncertainty,” *J. Risk Uncertainty*, vol. 5, no. 4, pp. 297–323, Oct. 1992.

[39] J. E. Contreras-Reyes and R. B. Arellano-Valle, “Kullback–Leibler divergence measure for multivariate skew-normal distributions,” *Entropy*, vol. 14, no. 9, pp. 1606–1626, Sep. 2012.

[40] P. A. Lopez et al., “Microscopic traffic simulation using SUMO,” in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2575–2582.

[41] T. Wágner, T. Ormándi, T. Tettamanti, and I. Varga, “SPaT/MAP V2X communication between traffic light and vehicles and a realization with digital twin,” *Comput. Electr. Eng.*, vol. 106, Mar. 2023, Art. no. 108560.

Bence Jekl received the master’s degree in vehicle engineering from Budapest University of Technology, Budapest, Hungary, in 2025. He is currently a Young Research Fellow with the Institute for Computer Science and Control (SZTAKI), HUN-REN. His current research interests include software implementation and simulation-based verification methods for autonomous vehicles.

Zvonimir Dabčević received the master’s degree in mechanical engineering from the Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb, Zagreb, Croatia, in 2022, where he is currently pursuing the Ph.D. degree. His current research interests include machine learning with automotive applications, synthesis of driving cycles, and city bus transport electrification planning.

Balázs Németh (Member, IEEE) received the Ph.D. degree in transportation sciences from Budapest University of Technology and Economics in 2013, the Habilitation degree in 2023, and the D.Sc. degree from Hungarian Academy of Sciences in 2024. He is currently pursuing the Ph.D. degree in theology with the Károli Gáspár University of the Reformed Church in Hungary, Budapest, Hungary. He is also a Research Advisor with the Institute for Computer Science and Control (SZTAKI), HUN-REN. His research interests include coordinated control design, autonomous vehicle systems, nonlinear analysis and synthesis of control systems, and ethical aspects of autonomous systems. He is a member of the IFAC Intelligent Autonomous Vehicles Committee.

Branimir Škugor (Member, IEEE) received the Ph.D. degree in mechanical engineering from the University of Zagreb in 2016. He is currently an Assistant Professor with the University of Zagreb. He has participated in numerous projects supported by Croatian Science Foundation, European Commission, and Ford Motor Company. His research interests include advanced controls of hybrid electric vehicle powertrains, modeling and charging optimization of electric vehicle fleets, synthesis of driving cycles, machine learning with automotive applications, and autonomous vehicle control.

Péter Gáspár (Member, IEEE) is currently the Head of the Systems and Control Laboratory, Institute for Computer Science and Control (SZTAKI), HUN-REN. He is also a Full Professor with the Department of Control for Transportation and Vehicle Systems, Budapest University of Technology and Economics. Since 2022, he has been a Full Member with Hungarian Academy of Sciences. His research interests include linear and nonlinear systems, robust control, multi-objective control, system identification, identification for control, and artificial methods. His research and industrial works have involved mechanical systems, vehicle structures, and vehicle dynamics and control. He is the Chair of the International Federation of Automatic Control (IFAC) Hungary National Member Organization and a member of the IFAC Automotive Control and Transportation Systems Technical Committees.