

Autonomous Vehicle and Pedestrian Interaction

Leveraging The Use of Model Predictive Control & Genetic Algorithm

Elaa Elgharbi

 [0000-0002-6913-0351](https://orcid.org/0000-0002-6913-0351)

*Lycee Ibn Abbad, Université Cadi Ayyad Faculté des Sciences Semlalia,
Université Mohammed V Agdal Faculté des Sciences
Rabat, Morocco
gharbielaa0@gmail.com*

Mate Zoldy

 [0000-0003-1271-840X](https://orcid.org/0000-0003-1271-840X)

*Department of Automotive Technology, Budapest University of Technology and Economics
Budapest, Hungary
zoldy.mate@kjk.bme.hu*

Safa Bhar Layeb

 [0000-0003-2536-7872](https://orcid.org/0000-0003-2536-7872)

*LR-OASIS
National Engineering School of Tunis, University of Tunis El Manar,
Tunis, Tunisia
safa.layeb@enit.utm.tn*

Abstract

Driving assistance systems and even autonomous driving have and will have an important role in sustainable mobility systems. Traffic situations where participants' cognitive levels are different will cause challenges in the long term. When a pedestrian crosses the road, an autonomous vehicle may need to navigate safely while maintaining its desired speed. Achieving this involves using a predictive model to anticipate pedestrian movements and a strategy for the vehicle to adjust its speed proactively. This research combined model-based predictive control (MPC) with a social-force model (SFM) to effectively control the autonomous vehicle's longitudinal speed. A genetic algorithm (GA) was also integrated into the approach to address the optimisation problem. A comparison between the proposed approach (MPC-GA) and the conventional MPC technique proved the outperformance of MPC-GA.

Keywords

autonomous vehicle, MPC, GA, pedestrian safety, sustainability

1. Introduction

As the number of vehicles continues to rise, so does the potential for increased risks involving road users. Different types of accidents occur worldwide with varying intensities. However, the riskiest of them is vehicle–pedestrian accidents. The combined system of human and artificial intelligence-supported vehicles is relevant in most mobility fields. Pedestrians in the street are vulnerable to a high risk of fatal accidents when encountering vehicles. Various factors, such as age concentration, contribute to this vulnerability. The number of accidents involving pedestrians and vehicles is surging worldwide. Particularly in driving areas with no controlling traffic signs or when pedestrians illegally cross the streets, it becomes crucial to safeguard and protect these insecure commuters.

According to the Centers for Disease Control and Prevention, 3700 people die daily, and millions of people around the world have died in road accidents (CDC, 2023). This concerning number highlights the urgency to address and diminish the number of future collisions of this kind. However, places without safety precautions, such as unsignalized streets and



uncontrolled crosswalks, are more prone to see such fatalities. Another fact is that most pedestrian accidents happen close to where the victim lives, making residential neighbourhoods riskier.

Autonomous vehicles (AV) have become a breakthrough technology (Zöldy et al., 2020). AVs are classified into five levels based on their level of automation. While most studies have focused on the problem of path planning (Hegedüs et al., 2017) and (Cao and Zoldy, 2022), it is imperative to investigate the ability of AVs' communication with road users and their potential to eliminate human driver's errors. To ensure safety, AVs are expected to drive cautiously around pedestrians, which can encourage pedestrians to engage in careless behaviours like stepping onto the road, forcing AVs to slow down and yield. Hence, inspecting the interaction between AV and pedestrians is required to ensure that pedestrians cross the road safely.

Several challenging opportunities are open in the vehicle-to-pedestrian (V2P) interaction. The most critical of them is to model the behaviour of pedestrian motion when crossing the road. The key is to define AV motion planning. Therefore, the next section 2 provides background information and reviews previous research about pedestrian and AV interaction. Then, the methods and data employed in our approach are thoroughly explained in Sections 3 and 4, respectively. Finally, parameters, settings, findings, and the overall conclusion are presented.

We seek to contribute meaningfully to existing research in this area through our study. The key contributions of this paper are as follows:

- Apply MPC to the AV and pedestrian road crossing scenario with real-time optimisation using the genetic algorithm for tuning MPC problems.
- Improvements were made to the genetic algorithm's parameters, including crossover rate, mutation rate, and the incorporation of roulette wheel selection and elitism to enhance its overall performance.

2. Related works

This section presents the most relevant domains and research results in pedestrian movement modelling, autonomous vehicle motion planning and available modelling and decision-making technologies.

2.1 Residential areas: accidents prone zones

Residential areas, usually referred to as residential neighbourhoods or communities, are distinguished by their principal function as places for habitation. Compared to major thoroughfares, these regions often have a network of streets with lower traffic densities and slower speed limits. Moreover, residential districts are known for their combination of pedestrian and vehicular traffic, which includes neighbourhood commuters, walkers, and children at play.

A thorough investigation (Law, 2020) revealed some elements contributing to why residential areas are more vulnerable to such risk. The increasing frequency of commuting, speeding, and distracted driving are a few contributing causes. Other causes include underage driving, unmanaged highways, and a lack of standard traffic signals. Therefore, several studies recommended imposing a default speed limit in residential areas, such as a speed restriction of 20 mph (32 km/h) as the International Transport Forum recommended, and 30 km/h, according to Kloth (2018).

2.2 Autonomous vehicles

Besides having the potential to enhance sustainability through shared mobility, lower fuel consumption, and better driving patterns, increasing the cognitive level of vehicles leads to increased mobility and safety. Autonomous vehicles can lead the way towards pedestrian safety and efficiency on our roads. Connecting driving support systems and vehicles also increases safety for other mobility participants. The safety of motorcyclists, one of the most vulnerable, can also be greatly increased by, among other things, increasing the use of communication systems (Zöldy, 2023).

According to Palmeiro et al. (2018), it is anticipated that autonomous vehicles will help decrease the occurrence of accidents. This projection is based on their ability to replace fallible human drivers and navigate the road network with enhanced safety. Self-driving cars are a revolutionary technological advancement in the automotive industry. With advanced sensors and artificial intelligence algorithms, these vehicles can operate on roads without human intervention (Torok and Pauer, 2022).



2.3 Social force model

One of the most popular approaches to pedestrian behaviour modelling is physics-based models. Hence, several methods have been used to simulate the behaviour of pedestrians towards autonomous vehicles, such as the social force model (SFM). This method is a widely used technique for simulating pedestrians, which was first introduced by Helbing and Molnar (1998) to model crowd behaviour in evacuation analysis. The resultant force of a pedestrian is the combination of the psychological and the physical force. Psychological force is used to describe the social properties of the pedestrian. In collision and friction, physical force takes effect (Zhou et al., 2021). These models consider “physical” and “social” forces acting on the agent. Based on this, an agent’s movement is driven by a combination of all the forces acting on it at each time step. The basis of the model relied on three main collections of forces acting on an agent: an attracting force drawing the agent towards their goal, a repulsive force away from each agent in the scene, and repulsive forces from all walls and borders in the environment. The motion resulting from the sum of these forces was shown to model human movements in their presented scenarios accurately.

Several authors have studied the V2P interaction in varied scenarios using different approaches. A recent study by Prédhumeau et al. (2022) investigated V2P interaction using SFM and a decision model that predicts such interactions. The simulation was done in controlled experiments in a parking lot at The Ohio State University. Another study by Zeng et al. (2014) analysed pedestrian behaviour at a signalised intersection using the modified social force theory.

In an article by Rashid et al. (2022), an interaction framework between AV and pedestrians was established by employing a rule-based social force model to replicate pedestrian movement during road crossings. Similarly, a modified SFM was suggested for pedestrian crossing by Ren et al. (2014). In their study, the authors suggested some modifications to the SFM by applying the concepts of anisotropy and relative velocity.

2.4 Model predictive control

Notably, model predictive control (MPC) is a category of control algorithms utilising a dynamic process model to anticipate and enhance process performance (Henson, 1998). Yang and Özgüner (2019) studied unsignalized scenarios. The MPC method was used for longitudinal AV speed control and solved using a quadratic programming (QP) toolbox to control pedestrian crowds. The method was supplemented and compared with a proportional integral derivative (PID) controller. Likewise, SFM was used to predict pedestrian motion. Preliminary results of the research clearly demonstrated the benefits of the used MPC solution compared to the classical PID method.

Furthermore, a similar study by Jayaraman et al. (2020) used a behavior-aware MPC model to regulate the AV to navigate safely in crosswalks, proving the efficiency of their suggested model. However, the solving process of NMPC (nonlinear model predictive control) is complex and harder than LMPC (linear model predictive control) and requires sophisticated algorithms.

Besides, a recent paper by Pan et al. (2023) proposes a trajectory planning algorithm that combines A*, the artificial potential field method, and MPC with the addition of a dynamic obstacle potential field in the objective function of the MPC controller. This method enables vehicles to actively avoid collisions with pedestrians and improve traffic safety while maintaining smooth, constraining trajectories.

Overall, the application of MPC is valuable for managing pedestrian-vehicle interactions and facilitating secure pedestrian crossings. Researchers have leveraged MPC to regulate autonomous vehicle behaviour when encountering pedestrians, particularly during road crossings. These studies showcase MPC’s potential to enhance AV safety and pedestrian interaction, ensuring smoother navigation and collision avoidance in crosswalks.

2.5 Genetic algorithm

Several studies investigated possible approaches to solve MPC optimisation efficiently. To this end, several algorithms have been proposed to solve the optimal control parameters of MPC, including the Genetic Algorithm (GA). GA is a family of search algorithms inspired by natural selection and genetics, which presents a solution to solve highly nonlinear optimisation problems (Vermuyten et al., 2018).

Moreover, the optimisation of MPC using GA was tested and applied in various cases, such as lane keeping and autonomous vehicle parking. For instance, in Son et al. (2017), GA was used to tune the MPC parameters for a lane-keeping scenario. Results were promising and showed that the optimised MPC exhibited better performance than the human-tuned MPC. Similarly, as suggested by (Arrigoni et al. 2022), a novel genetic algorithm strategy was proposed to solve the NMPC problem for an autonomous vehicle path planner in an urban scenario. The simulation was conducted using the CarMaker environment, and the results demonstrated the successful performance of the model. The GA was crucial in preventing the solution from getting stuck in local minima. Additionally, the results confirmed the feasibility of a real-time controller implementation.

In essence, GA-enhanced MPC improved performance in tasks such as autonomous vehicle control. Some strategies have been proposed to optimise MPC, including Particle Swarm Optimization (PSO) (Abdolahi et al., 2023; Kebbati et al., 2021).

3. Problem formulation

For the test conducted in this study, a residential area was hypothesised, where streets are non-controlled with traffic lights, and the AV encounters an unsignalized crosswalk. Pedestrians nearing the crosswalk must make decisions regarding crossing or waiting for the AV to pass. To prioritise the safety and comfort of AV riders, the AV employs a pedestrian crossing model to predict pedestrian behaviour and plan its actions accordingly.

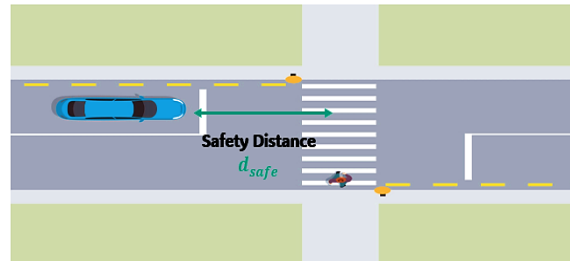


Figure 1 Hypothetical scenario

The proposed approach incorporates a GA to address the numerical solution of the optimal control problem (OCP) within the MPC framework. The method utilises the SFM to handle pedestrian motion while employing MPC to control and regulate the longitudinal speed of the autonomous vehicle.

3.1 Motion Predictions: vehicle dynamics

In our analysis, we consider the AV as a point mass. We focus on the longitudinal vehicle dynamics for crosswalk interactions and adopt a discrete-time kinematic model. Hence, our focus will be dedicated to longitudinal dynamics for the vehicle:

$$M\ddot{s}(t) + \alpha \dot{s}(t) = u(t), \quad (1)$$

where

s – is the longitudinal position,

M – is the vehicle mass,

α – is a drag coefficient

$u(t)$ – is the control action.

Thus, let $X = [x_1, x_2]^T = [s, \dot{s}]^T$ be the state vector which comprises the vehicle's position and velocity, denoted by x_1 and x_2 , respectively. Furthermore, by rewriting the equation into discretised vehicle dynamics using the Euler method, we have:

$$x(k+1) = Ax(k) + Bu(k), \quad (2)$$

where

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 - \frac{\alpha \Delta t}{M} \end{bmatrix}, B = \begin{bmatrix} 0 \\ \frac{\Delta t}{M} \end{bmatrix},$$

Δt – represents the discretisation time step,

$u(k)$ – is the discretised control action.

This process of moving from continuous time to discrete time using the Euler method was proposed by Yang and Özgüner (2019).

3.2 Pedestrian motion prediction

At time t , pedestrian motion can be iteratively obtained using the pedestrian dynamics in the equation below, from which the pedestrian state $x_p(k)$ is derived.

$$\ddot{x}_p = \frac{d^2 x_p}{dt^2} = a = \frac{1}{m_p} f_{total} \quad (3)$$



Furthermore, the model relies on the principles of social force. It is applied to predict the motion of pedestrians under the influence of vehicles. In this model, the movement of each pedestrian is represented as x_p follows the 2D planar point-mass Newtonian dynamics, influenced by a total force f_{total} , comprising various sub-forces (Yang and Özgüner, 2019).

Where $x_p \in R^4$ represents the pedestrian state vector, mainly the positions (x_p, y_p) and velocities (v_x, v_y) in (x, y) axes. Whereas m_p is the pedestrian's mass and $f_{total} \in R^2$ is the total force. The velocities and acceleration are expressed as below:

$$\dot{x}_p = v_x \text{ and } \dot{y}_p = v_y \quad (4)$$

$$\dot{v}_x = a_x \text{ and } \dot{v}_y = a_y \quad (5)$$

Hence, the acceleration is equal to: $a_x = \ddot{x}$ and $a_y = \ddot{y}$ (6)

The total longitudinal force acting on the pedestrian is the sum of two main forces: the attraction force (or destination force), which drives the pedestrian toward their desired goal, and the repulsive force (or vehicle force), which repels the pedestrian from the vehicle. Hence, the total force is equal to:

$$f_{total} = f_{destination} + f_{vehicle} \quad (7)$$

3.3 Cost function design

Based on the vehicle dynamics previously explained, the future state vector of the vehicle position $x_1(k+n)$ will be obtained. Likewise, the future pedestrian state $x_p(k+n)$ will be obtained from the previously mentioned pedestrian dynamics.

A safety requirement is incorporated into the MPC constraint architecture to guarantee the safe functioning of the autonomous vehicle and avoid any potential collisions with persons crossing the road. This standard stipulates that, about the longitudinal position of the vehicle, a specific distance designated as d_{safe} must be maintained between the autonomous vehicle and the pedestrian. To maintain this defined d_{safe} throughout N future steps:

$$|x_1(k+n) - x_p(k+n)| > d_{safe} \quad (8)$$

This formula ensures that the projected distance between the vehicle's future position and the pedestrian's anticipated location stays within the allowed safety margin. Also, limitations are placed on the velocity, control action and the rate of change of control action due to the physical constraints of the vehicle. Therefore, the following constraints will be considered. For $\forall i = k+1, \dots, k+N$, we have:

$$v_{min} < v(k+n) \leq v_{max} \quad \forall n \in \{1, 2, \dots, N_p\} \quad (9)$$

$$|u(k+n)| \leq u_{max} \quad \forall n \in \{0, 1, 2, \dots, N_p - 1\} \quad (10)$$

$$|\Delta u(k+n)| \leq \Delta u_{max} \quad \forall n \in \{0, 1, 2, \dots, N_p - 1\} \quad (11)$$

The MPC's final objective is to identify the control actions $U = [u(k), u(k+1), \dots, u(k+N-1)]^T$ that will ensure that the vehicle meets state and safety constraints. Considering this, the cost function is referred to as follows:

$$U^* = \arg \min_U \sum_{n=0}^{N_p} w_v (v(k+n) - v_{0,veh})^2 + \sum_{n=0}^{N_p-1} w_u (u(k+n))^2 \quad (12)$$

where W_u and W_v are the weights of the cost of the velocity and control, respectively. Subject to the previous constraints (2), (7), (8), (10) and (11).

Algorithm: MPC for longitudinal speed

Input: MPC parameters, initial state and input $u(0)$.

Result: control signal $u^*(k)$

Initialization

For each time step k **do**

obtain $x(k)$ and pedestrian state;

 predict pedestrian motion using pedestrian dynamics;

 solve for $U^* = \arg \min_U \sum_{n=0}^{N_p} w_v (v(k+n) - v_{0,veh})^2 + \sum_{n=0}^{N_p-1} w_u (u(k+n))^2$;

if MPC is feasible **then**

 apply $u(k) = U^*(1)$;

else

 infeasible

end

end

Figure 2 MPC algorithm

3.4 Model predictive control based on genetic algorithm

The genetic algorithm searches for the optimal control sequences that optimise the cost function and comply with the model constraints. The following sections explain the optimisation process.

3.4.1 Encoding

Each chromosome ($c_i = \{gene_1, \dots, gene_{N_c-1}\}$) in the population constitutes a control action $[u(t), u(t+1), \dots, u(t+N-1)]$, with $i = 1$, population size. The selection of each chromosome is done within the defined range of control interval $[u_{\min}, u_{\max}]$ and with $\{\Delta_i u(t+j), j = 1, \dots, N_c - 1\}$ not exceeding the limit $\Delta u_{\max}$.

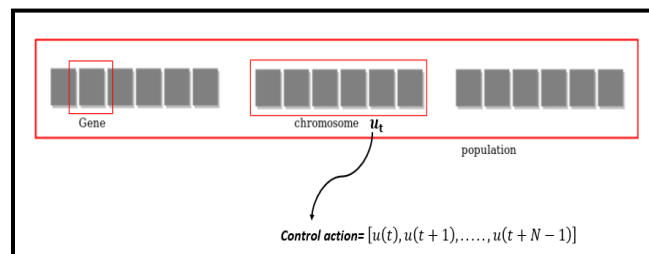


Figure 3 Control input



3.4.2 Initialisation

- Begin by generating an initial control value within the defined constraint space.
- If the individual adheres to the constraints and terminal constraints, include it in the initial population.
- Continue to iterate through the steps mentioned above until a total of individuals reaches the population size that has been selected.

3.4.3 Evaluation and Selection

- The fitness value for each chromosome is defined as: $\frac{1}{(1+J)}$ (13)
- With J being the value of the cost function of the MPC problem.

3.4.4 Genetic operators

Roulette Wheel Selection

The selection strategy used in this paper utilises the roulette selection algorithm. As a result, individuals with higher fitness levels are more likely to be selected and retained during the evolutionary iteration process. If an individual is denoted as i and its fitness is f_i , the probability that it will be selected is expressed as:

$$P_t = \frac{f_t}{\sum_{t=1}^n f_t} \quad (14)$$

Mutation

Genes are chosen based on their likelihood of mutation. They are subsequently replaced at random while adhering to permissible boundaries defined by control signals $|u(i)| \leq u_{max}$ and limits on the change in control signals $|\Delta u(i)| \leq \Delta u_{max}$.

Crossover

This approach creates two offspring while ensuring that the control signals remain within acceptable constraints.

Elitism

The elite-preservation strategy is implemented to ensure the current population's quality and the MPC algorithm's steadiness. Where the top 1% of high-fitness individuals in the population are retained.

Termination condition

Repeating the preceding step and entire calculation at subsequent control interval $k+1$.

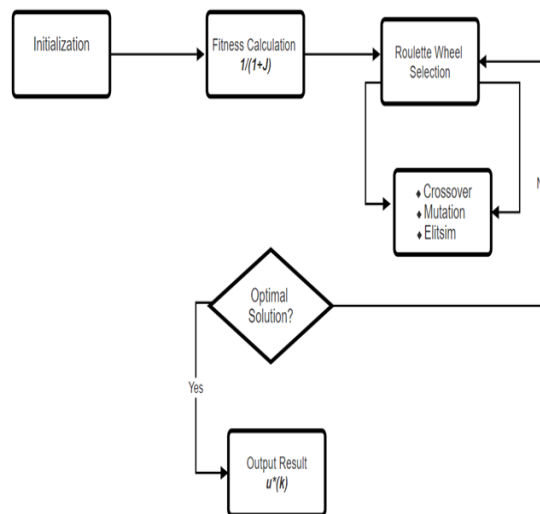


Figure 4 GA flowchart

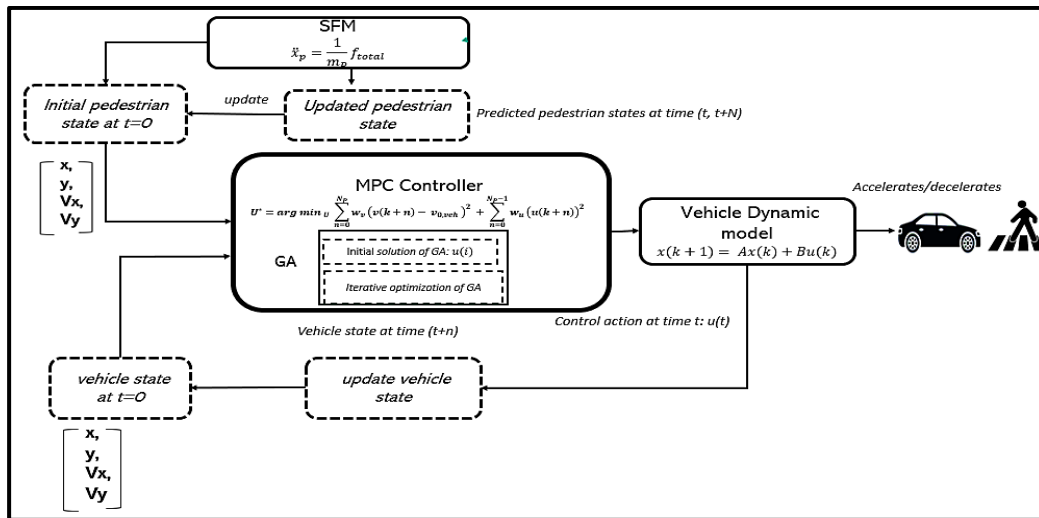


Figure 5 Model diagram

Finally, the overall algorithm is stated as follows:

Algorithm: MPC-GA for longitudinal speed

Input: MPC parameters, initial state and input $u(0)$.

Result: control signal $u^*(k)$

Initialization

For each time step k **do**

 obtain $x(k)$ and pedestrian state;

 predict pedestrian motion using pedestrian dynamics;

for $n=0$ to N_{p-1} **do**

 // Genetic Algorithm Optimization

 gen_counter = 0

while gen_counter < MaxGenerations **do**

 // generate set of possible control moves

 // find the process output for all control moves

 // evaluate the fitness for each solution

 // apply genetic operators

 //repeat until number of generation is reached

 gen_counter = gen_counter + 1

end

 solve for $U^* = \arg \min_U \sum_{n=0}^{N_p} w_v (v(k+n) - v_{0,veh})^2 + \sum_{n=0}^{N_p-1} w_u (u(k+n))^2$;

if MPC is feasible **then**

 apply $u(k) = U^*(1)$;

else

 // Handle infeasible case (maximum deceleration)

 infeasible

end

end

end

Figure 6 MPC-GA algorithm



4. Parameter settings

To conduct the simulation, CVXPY, a Python-based modelling language for convex optimisation, was chosen to solve the optimisation problem (Diamond and Boyd, 2016). A computer equipped with an Intel Core i5-1135G7 CPU, 8GB of RAM, running on the Windows 11 operating system, and employing Python programming language was used. The tables below contain the values chosen for the parameters based on the literature found in the work of (Yang and Özgüner, 2019) and (Jayaraman et al., 2020).

Table 1 MPC parameters

Symbol	Value	Unit
Vehicle mass M	2000	Kg
Minimum control input u_{min}	-7	m/s ²
Maximum control input u_{max}	7	m/s ²
Minimum control input change Δu_{min}	-5	m/s ³
Maximum control input change Δu_{max}	5	m/s ³
Drag coefficient α	100	-
Minimum speed v_{min}	0	m/s
Maximum speed v_{max}	16	m/s
Safety distance d_{safe}	5	M
Prediction horizon N_{pred}	15	-
Weight on the control w_v	1	-
Weight on control Rate w_u	1	-



Table 2 GA parameters

GA parameter	Setting used
Population size (default)	100
Crossover rate (default)	0.9
Mutation rate(default)	0.1
Elite count	1%
Maximum number of generations(default)	100

The default value for the number of generations is set to 100, a choice aligned with previous studies (Ahmadpour et al., 2021; Ramasamy et al., 2019; Mohammadi et al., 2018).

Table 3 SFM parameters

Symbol	Value	Unit
Maximum pedestrian speed $v_{p,max}$	2.5	m/s
Maximum pedestrian acceleration $a_{p,max}$	5	m/s ²
Pedestrian mass m_p	80	Kg
$k_{destination}$	300	-
$A_{vehicle}$	200	-
B	2.6	-

5 Results and analysis

For crossover and mutation rates, we used 0.9 and 0.1 at the beginning of the simulation, respectively. Following the guidelines provided by most sources ((Abdolahi et al., 2023; Kebbati et al., 2021; Vermuyten et al., 2018; Arrigoni et al., 2022; Hang et al., 2021), we used (0.7, 0.3) and (0.6, 0.4) as crossover and mutation rates, respectively. Subsequently, we manually adjusted these rates to obtain the optimal combination based on CPU time usage, aiming for computational efficiency across 200 simulation runs. Computer systems require energy for almost all computational tasks involved in running algorithms, which adds to their carbon footprint. Consequently, it highlights how important it is to maximise computational efficiency to minimise environmental impact and promote sustainable computing practices.

5.1 Step 1: crossover and mutation rates

Step 1 shows the crossover and mutation rates based on GA rate tuning.

Table 4 GA rate tuning

	MPC-GA					
	CPU					
(Crossover, Mutation)	0.9	0.1	0.7	0.3	0.6	0.4
Average	75.3611		74.7704		81.7025	

Hence, based on the results above, the crossover and mutation rates used are 0.7 and 0.3, respectively.

Now, tune the number of generations using different settings: [20, 30, 40, 50, 100]. We kept the population size fixed to its default value (100) and tested the above combinations for 200 simulation runs.

5.2 Step 2: generations number

The CPU time gradually rises with the number of generations. As expected, given that the algorithm runs through additional cycles to evolve a solution, more generations demand more processing power. For every 100 generations, the CPU time rises from 64.087 seconds to 74.770 seconds.

However, the standard deviation (SD) does not follow a strictly linear pattern. For instance, the standard deviation after 20 generations is 3.402, which is relatively high and shows large variations in CPU time across runs. However, the value of SD significantly drops to 1.159 when the number of generations rises to 30, indicating a more steady and reliable performance. After 40 and 50 generations, it rises once more while staying in a modest range.

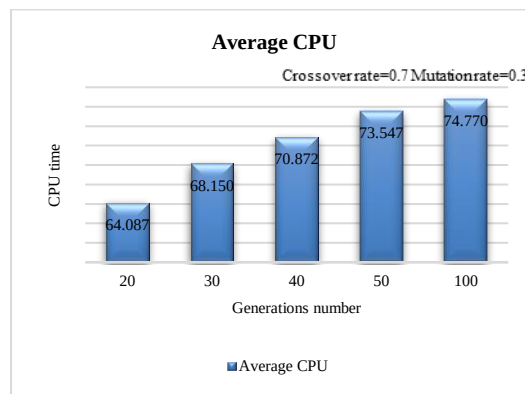


Figure 6 Generation number tuning

In summary, selecting a value with reduced variability, as indicated by a smaller standard deviation, is advantageous. This choice leads to computational performance that is more predictable and consistent, a crucial aspect in maintaining system reliability. Moreover, although running the algorithm with more generations takes longer, the results are more reliable, and the time required for each run is less variable. Hence, the selected number of generations is 30.

5.3 Step 3: population size

The findings show that computational efficiency is significantly affected by the population size. With values of 78.976 seconds and 78.939 seconds, respectively, the average CPU times are particularly close for population sizes 20 and 30. The average CPU time, however, noticeably rises to 87.957 seconds and 87.673 seconds, respectively, when the population reaches 40 and 50. This shows that larger populations have longer average calculation times because they use more processing resources.

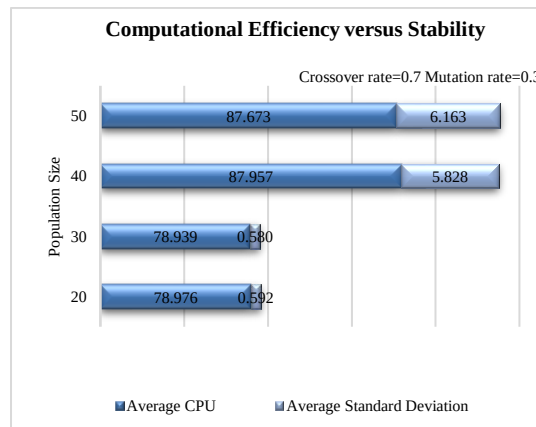


Figure 7 Population size tuning

Moreover, the standard deviations, 0.592 and 0.580 for population sizes 20 and 30, respectively, are very low, suggesting that the CPU time is largely constant and stable between runs. Nevertheless, the standard deviation significantly rises to 5.828 and 6.163, respectively, as the population grows to 40 and 50, respectively. This implies more variation in the computation time across various runs for larger populations.

In summary, the best option is to use a smaller population size (30), as it ensures stable and consistent performance, along with lower average CPU times and standard deviations. This approach reduces unnecessary computational burden, maximises resource efficiency, and lowers energy consumption. As a result, it reduces the carbon footprint associated with prolonged algorithmic processing (IBM Cloud Education, 2023). This result will be very useful, especially when dealing with more complex scenarios and processes that demand higher computational resources.

Table 5 Tuned GA parameters

Tuned Parameters			
Crossover Rate	Mutation rate	Number of Generations	Population Size
0.7	0.3	30	30

6 Comparison between MPC versus MPC-GA

6.1 Quantitative Assessment: RMSE Analysis and Computational Efficiency

Then, a final comparison between the two models using the best combination of GA parameters was done. Furthermore, Total time and root mean squared error (RMSE) were compared to evaluate the performance and efficiency. The total average RMSE over all cycles was calculated as follows:

$$RMSE = \frac{1}{n} \left[\sum_{i=1}^n (y_i - \hat{y})^2 \right]^{\frac{1}{2}} \quad (14)$$



Table 6 MPC versus MPC-GA performance evaluation

	MPC	MPC-GA
Average elapsed CPU time (s)	89.21	62.71
Total average RMSE for a control method	0.65	0.32

Numerous simulation runs were conducted, and the results consistently showed that MPC-GA had lower RMSE than MPC and significantly reduced CPU processing time. MPC-GA performed better in percentage terms, with an RMSE roughly 50.76% lower (0.32 RMSE) than the MPC. These results confirm the superior performance of MPC-GA and show that it provides a more efficient solution, ultimately resulting in improved overall performance. Simultaneously, MPC-GA contributes to sustainability by adhering to sustainable computing principles outlined by Lannelongue et al. (2023) by reducing processing requirements for running algorithms, consequently minimising energy consumption and resource utilisation.

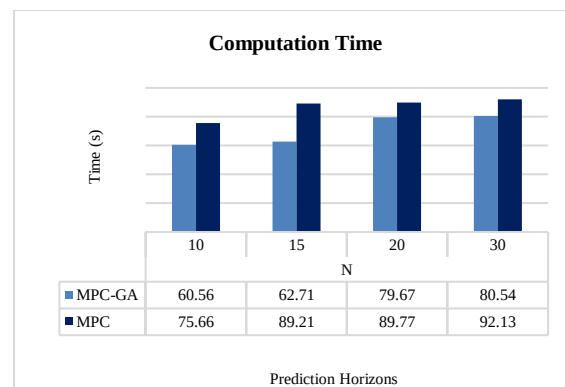


Figure 8 Computation time comparison between MPC and MPC-GA

Furthermore, we examined a comparison analysis between MPC versus MPC-GA, using different prediction horizons, to conduct a more thorough evaluation of the performance of the two models. The outcomes of this comparative assessment show that, regarding time efficiency, MPC-GA outperformed the traditional MPC strategy. When compared to pure MPC, MPC-GA showed much shorter execution times. The percentage improvements in CPU time were observed to be 20%, 29%, 11%, and 13% for prediction horizons of 10, 15, 20, and 30, respectively. This finding highlights how MPC-GA's superior optimisation capabilities can be used to boost overall operational effectiveness.

6.2 Simulation-based evaluation

The evaluation of the simulation results, which utilised Model Predictive Control enhanced with Genetic Algorithm (MPC-GA), reveals a promising outcome. Notably, during simulations involving pedestrian interactions, the autonomous vehicle demonstrated a significant capacity for ensuring safety.

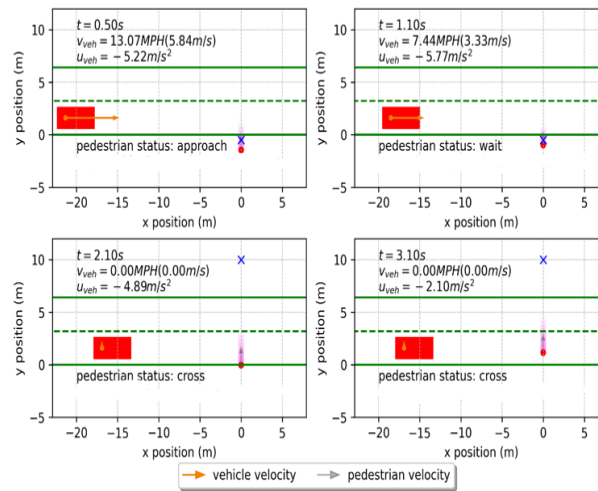


Figure 9 MPC-GA simulation result

When a pedestrian started a crossing, the vehicle promptly and effectively came to a complete stop ($V_{veh} = 0 \text{ m/s}$), showcasing the robustness of the MPC-GA system. These results underscore the potential of MPC-GA in real-world scenarios, illustrating its ability to prioritise safety and successfully manage interactions with pedestrians. This also demonstrated its robust safety measures, ensuring pedestrians' safety during vehicle interactions. Notably, the vehicle consistently stopped at a safe distance in front of the pedestrian, emphasising its safety commitment.

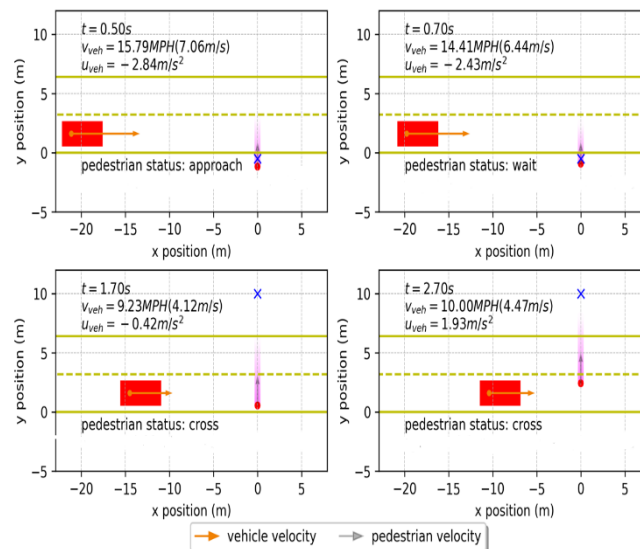


Figure 10 MPC-GA simulation result

We can see no collision between the AV and the pedestrian using MPC. Nevertheless, the AV failed to attempt to stop at a safe distance when the pedestrian was crossing the street. Rather, the AV decreases its speed from 14.41 MPH to 9.23 MPH. In addition, the AV accelerates once more as soon as the pedestrian crosses the driving lane.

6.3 Comparison of Control Actions and state evolution

It is crucial to highlight the significant disparity in control strategies between the two approaches, as shown in the figure below. The control actions employed by MPC-GA displayed notable fluctuations, signifying an ongoing optimisation process involving actively fine-tuning control instruction. This dynamic behaviour played a pivotal role in achieving the abrupt stop. On the other hand, the smoothness of the MPC control actions implied a more cautious and conservative

approach, emphasising gradual speed reduction to enhance passenger comfort. Thus, the MPC-GA approach emphasised precision and safety, leading to a more sudden stop when required. Hence, our analysis reveals that the choice between MPC-GA and MPC involves a trade-off between speed smoothness and precision. While MPC offers a consistent and comfortable speed profile for passengers, MPC-GA demonstrates a more dynamic behaviour in speed control. The volatility in speed observed in the MPC-GA strategy directly results from the real-time optimisation process. It ensures that the vehicle can swiftly adapt to changing situations, such as pedestrians entering the crosswalk.

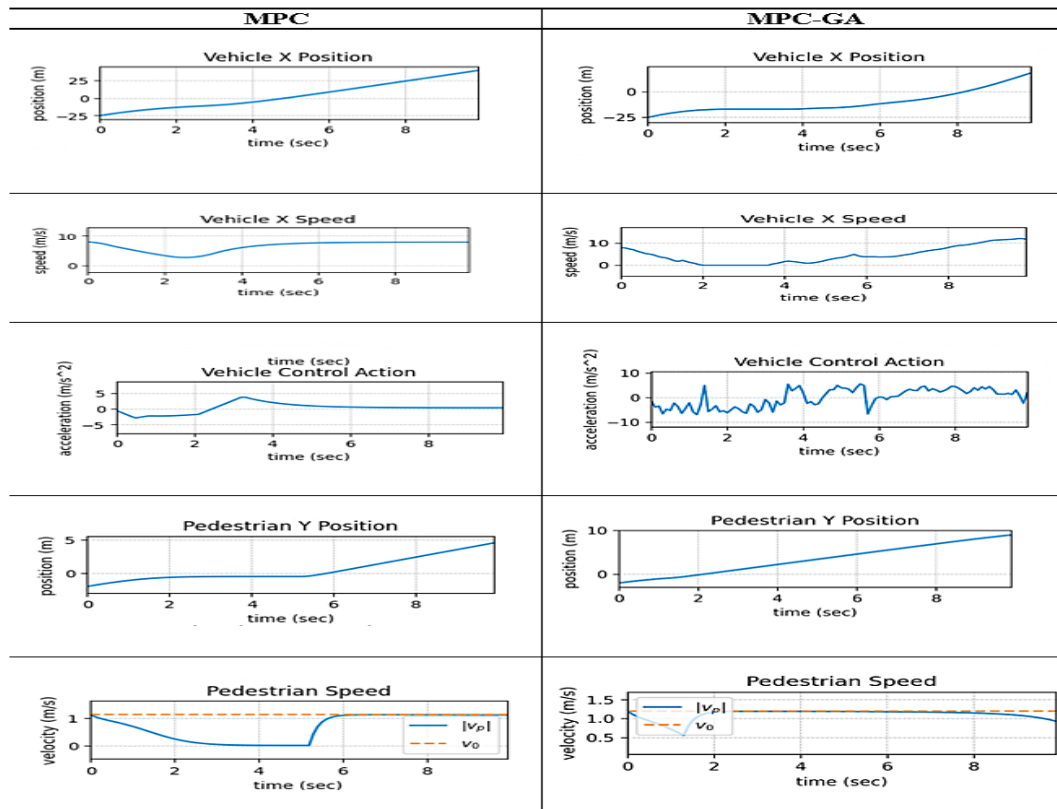


Figure 11 Graphical results comparison of MPC and MPC-GA results

7 Conclusion and future work

One of the most important steps toward sustainability, particularly in promoting sustainable mobility, is the integration of autonomous vehicles into transportation networks. In addition to transforming transportation, these cars can protect pedestrian safety at crosswalks. This paper's investigation is driven by an innovative emphasis on pedestrian safety and sustainability, two essential aspects of contemporary urban living. To achieve this overarching goal, the paper explores the effectiveness of MPC in ensuring pedestrian safety and investigates the potential of integrating GA to optimise MPC for more robust safety outcomes.

To assess pedestrian safety in this context, the SFM was employed, using Newton's second law to simulate pedestrian motion. The study then systematically evaluated the effectiveness and efficiency of both MPC and MPC-GA in ensuring pedestrian safety. MPC was used to model the behaviour of the autonomous vehicle and its interaction with the crossing pedestrian through acceleration and deceleration. Additionally, GA was utilised to optimise the process of solving MPC. Specifically, genetic evolution was used to find the optimal control input in each iteration. The findings underscored the substantial advantages MPC-GA offers, with slightly lower execution times signalling enhanced computational efficiency, which is particularly valuable in complex control systems. The lower running times required in each iteration would reduce the need for energy use and, ultimately, more sustainable computing.



Furthermore, MPC-GA consistently reduced computing times across various prediction horizons, emphasising the efficacy of GA optimisation. This approach also significantly improved predictive accuracy, achieving control objectives more efficiently. Regarding the comparison between simulations, MPC and MPC-GA resulted in safe interactions, with no recorded collisions. However, MPC-GA exhibited a superior interaction response, wherein the autonomous vehicle yielded to pedestrians and came to a full stop when required, underlining an exceptional level of safety. In the case of local emission-free drivetrain solutions like electric cars with increasing safety, it has an overall positive effect on sustainability.

However, the study was conducted and limited to a simulated environment, thereby leaving the influence of real-world factors like unpredictable pedestrian behaviour and varying environmental conditions unaddressed.

Additional research directions could involve investigating programming languages other than Python and evaluating their effects on computer performance, which would expand the study of sustainable computing techniques. Analysing the interactions between various language codes and algorithms may provide ways to maximise computing efficiency, aligning with sustainability objectives by reducing the amount of energy and resources used. Another improvement could be to test the scenario in a more complex road structure to investigate the robustness of MPC-GA to ensure pedestrian safety, which is crucial in sustainable mobility and contributes to smoother traffic flow.

References

- Abdollahi, Y., Yousefi, S., Tavoosi, J. (2023). A New Self-Tuning Nonlinear Model Predictive Controller for Autonomous Vehicles. *Complexity*. January 2023:1-9 DOI: <https://doi.org/mkpp>
- Ahmadpour, M. R., Ghadiri, H., Hajian, S. R. (2021). Model predictive control optimisation using the metaheuristic optimisation for blood pressure control. *IET Systems Biology*. 15(2), 41–52. DOI: <https://doi.org/mkpg>
- Alharasees, O., Adali, O. H., Kale, U. (2023, June). Human Factors in the Age of Autonomous UAVs: Impact of Artificial Intelligence on Operator Performance and Safety. *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE. 798–805 DOI: <https://doi.org/mkpr>
- Arrigoni, S., Braghin, F., Cheli, F. (2022). MPC trajectory planner for autonomous driving solved by genetic algorithm technique. *Vehicle System Dynamics*. 60(12), 4118–4143. DOI: <https://doi.org/mkps>
- Cao H., Zoldy, M. (2022). Implementing B-Spline Path Planning Method Based on Roundabout Geometry Elements. *IEEE Access*. 10. 81434–81446, 2022, DOI: <https://doi.org/mkpt>
- Centers for Disease Control and Prevention. URL: <https://www.cdc.gov/injury/features/global-road-safety/>
- Diamond, S., Boyd, S. (2016). CVXPY: A Python-embedded modeling language for convex optimisation. *Journal of Machine Learning Research*. 17(1), 2909–2913. DOI: <https://doi.org/mkpv>
- Cao, H., and Zoldy, M. (2021). MPC tracking controller parameters impacts in roundabouts. *Mathematics*, 9(12), 1394. DOI: <https://doi.org/jc52>
- Hegedüs, F., Bécsi, T., Aradi, S., Gápár, P. (2017). Model based trajectory planning for highly automated road vehicles. *IFAC-PapersOnLine*, 50(1), 6958–6964. DOI: <https://doi.org/djzn>
- Helbing, D., Molnar, P. (1995). Social force model for pedestrian dynamics. *Physical Review E*. 51(5), 4282. DOI: <https://doi.org/fn55b2>
- Henson, M. A. (1998). Nonlinear model predictive control: current status and future directions. *Computers & Chemical Engineering*, 23(2), 187–202. DOI: <https://doi.org/d6k4cz>
- IBM Cloud Education (2023, January 9). Why Green Coding is a Powerful Catalyst for Sustainability Initiatives. *IBM*. URL: <https://www.ibm.com/blog/green-coding/>
- Jayaraman, S. K., Robert, L. P., Yang, X. J., Pradhan, A. K., Tilbury, D. M. (2020, July). Efficient behavior-aware control of automated vehicles at crosswalks using minimal information pedestrian prediction model. *2020 American Control Conference (ACC)*. IEEE. 4362–4368 DOI: <https://doi.org/mkpw>
- Kebbaty, Y., Puig, V., Ait-Oufroukh, N., Vigneron, V., Ichalal, D. (2021, November). Optimised adaptive MPC for lateral control of autonomous vehicles. *2021 9th International Conference on Control, Mechatronics and Automation (ICCMA)*. IEEE. 95–103. DOI: <https://doi.org/mkpx>
- Kloth, M. (2018, March 29). Lower Speed Means Fewer Road Deaths. *International Transport Forum*. URL: <https://www.itf-oecd.org/lower-speed-means-fewer-road-deaths>
- Lannelongue, L., Aronson, H. E. G., Bateman, A., Birney, E., Caplan, T., Juckes, M., ... Inouye, M. (2023). GREENER principles for environmentally sustainable computational science. *Nature Computational Science*. 3(6), 514–521. DOI: <https://doi.org/mkpy>
- Law, A. (2020, November 1). Why Do Most Accidents Occur So Close to Our Homes? *AUGER & AUGER*. URL: <https://www.augerlaw.com/blog/why-do-most-accidents-occur-so-close-to-our-homes/>
- Mohammadi, A., Asadi, H., Mohamed, S., Nelson, K., Nahavandi, S. (2018). Optimising model predictive control horizons using genetic algorithm for motion cueing algorithm. *Expert Systems with Applications*. 92, 73–81. DOI: <https://doi.org/mkp2>



- Palmeiro, A. R., van der Kint, S., Vissers, L., Farah, H., de Winter, J. C., Hagenzieker, M. (2018). Interaction between pedestrians and automated vehicles: A Wizard of Oz experiment. *Transportation Research Part F: Traffic Psychology and Behaviour*. 58, 1005–1020. DOI: <https://doi.org/gd6d8h>
- Pan, W., He, Q., Huang, Y., & Qin, L. (2023). Four-dimensional trajectory planning for urban air traffic vehicles based on improved RRT* algorithm. *IEEE Access*. January 2023 PP(99):1-1 DOI: <https://doi.org/mkp3>
- Prédhumeau, M., Mancheva, L., Dugdale, J., Spalanzani, A. (2022). Agent-Based Modeling for Predicting Pedestrian Trajectories Around an Autonomous Vehicle. *Journal of Artificial Intelligence Research*. 73, 1385–1433. DOI: <https://doi.org/mkp4>
- Ramasamy, V., Sidharthan, R. K., Kannan, R., Muralidharan, G. (2019). Optimal tuning of model predictive controller weights using genetic algorithm with interactive decision tree for industrial cement kiln process. *Processes*. 7(12), 938. DOI: <https://doi.org/mkp5>
- Rashid, M. M., Seyedi, M., Jung, S. (2022). Simulation of Pedestrian Interaction with Autonomous Vehicles Via Social Force Model. Available at SSRN 4017139. DOI: <https://doi.org/gpfkdh>
- Ren, G., Li, H. Y., Lu, L. L., Zhang, T. (2014). Modeling and Simulation for Pedestrian Crossing Based on a Modified Social Force Model. *CICTP 2014: Safe, Smart, and Sustainable Multimodal Transportation Systems*. 3194–3204. DOI: <https://doi.org/mkp6>
- Son, R., Cho, J., Huh, K. (2017). Genetic Algorithm based Parameters Optimisation of Model Predictive Control in Lane Keeping. 한국자동차공학회 춘계학술대회. 333–335.
- Torok, A., Pauer, G. (2022). *Cognitive Sustainability*, 1(3). DOI: <https://doi.org/jswn>
- Vermuyten, E., Meert, P., Wolfs, V., & Willems, P. (2018). Combining model predictive control with a reduced genetic algorithm for real-time flood control. *Journal of Water Resources Planning and Management*, 144(2), 04017083 DOI: <https://doi.org/mkp8>
- Yang, D., Özgüner, Ü. (2019). Combining Social Force Model with Model Predictive Control for Vehicle's Longitudinal Speed Regulation in Pedestrian-Dense Scenarios. arXiv Safety aspects of critical scenario identification for autonomous transport. preprint arXiv:1907.05178.
- Zeng, W., Chen, P., Nakamura, H., Iryo-Asano, M. (2014). Application of social force model to pedestrian behavior analysis at signalised crosswalk. *Transportation Research Part C: Emerging Technologies*. 40, 143–159. DOI: <https://doi.org/gn2s9q>
- Zhou, R., Cui, Y., Wang, Y., & Jiang, J. (2021). A modified social force model with different categories of pedestrians for subway station evacuation. *Tunnelling and Underground Space Technology*, 110, 103837. DOI: <https://doi.org/gq6ww7>
- Zöldy, M., Szalay, Z., Tihanyi, V. (2020). Challenges in homologation process of vehicles with artificial intelligence. *Transport*, 35(4), 447–453. DOI: <https://doi.org/f9wq>
- Zöldy, M. (2023) The era of the cognitive mobility, FISITA - Tecnology and Mobility Conference Europe 2023.09.8.10 Barcelona, URL: https://www.researchgate.net/publication/377718172_The_era_of_the_cognitive_mobility/