The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM)
ISSN: 2602-3199

The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM), 2021

Volume 16, Pages 26-30

IConTES 2021: International Conference on Technology, Engineering and Science

Interactive Simulation Program for Autonomous Vehicles

Claudiu Radu POZNA
Transilvania University of Brasov
Széchenyi István University

Csaba ANTONYA
Transilvania University of Brasov

Abstract: The paper presents an interactive simulation program dedicated to autonomous vehicles. The program, developed in the form of an app and written in the Matlab programming language, is a didactic tool, which aims to better understand the phenomena that occur during locomotion. To achieve this goal, the user can easily perform interactive simulations by changing the model parameters. In addition to the mentioned flexibility, the user can perform case studies and scenarios. The structure of the program is obtained by aggregating seven objects (modules), dedicated to studying the contributions of each to the simulation. These modules are for different types of trajectories, for controllers, three types of locomotion models, sensors, different filters, environment models, and for simultaneous localization and mapping. The connection between them is possible through a database. This framework ensures the continuity of the simulations. At the end of the interactive visual simulation, the user is obtaining a holistic report with the results.

Keywords: Autonomous vehicle, Simulation, Program, Interactive, Modeling.

Introduction

Autonomous vehicles represent a hot research topic in the last years. Real-world data acquisition and testing are difficult but computer modeling and simulation represent an efficient way to analyze and design such vehicles. Modern computer hardware and the advancement of artificial intelligence made possible the development of efficient tools for algorithm testing and simulation in virtual environments.

An autonomous vehicle is reacting to the feedback received from sensors. A typical vehicle simulator needs to incorporate a vehicle model as well as models for the sensors and decision algorithms. The most important part of a virtual environment is the simulator that realistically represents the physics of the vehicle (Rosique et al. 2019). Testing of an autonomous vehicle is critical before deployment. A simulator eliminates the difficulties of the road test and allows the understanding of driving behavior, testing driver assistant systems, and for traffic research. The simulation environment must provide reproducible testing methods to accelerate the development (Huang et al. 2016). The most important autonomous vehicle simulators are AirSim, DeepDrive, Udacity, Constellation, Carcraft/Waymo, SIMLidar, Helios, GLIDAR, and CARLA (Dosovitskiy et al. 2017; Chen et al. 2018).

Autonomous vehicle testing is difficult because of its diversity and complexity. A pipeline that generates various scenarios for autonomous vehicle simulations is proposed in (Wen et al. 2020). An architecture for vehicle control is proposed in (Pozna et al. 2016), through a methodology that involves identifying primitives, combining them, abstracting them, and finally identifying patterns (Pozna et al. 2021).

- Selection and peer-review under responsibility of the Organizing Committee of the Conference

The Simulation Program

Computer modeling and simulation is a process of in-depth knowing a physical phenomenon. In general, the computer simulation of a robotic system results in a set of differential equations. Obtaining these equations is not the result but rather an intermediate step. These equations have to be transformed into algorithms and computational programs. The objective of knowledge is a priori, i.e. the possibility to anticipate the results of the causal relationship before the experiment. This goal can be achieved using computer programs through simulation. The quality of the simulation, the difference between the anticipated and the real results, depends on the quality of the model and that of the numerical methods used in solving the mentioned differential equations.

The various aspects and algorithms that are related to mobile robots and autonomous vehicles modeling led to the idea of their development in a unitary framework, which would allow a better understanding of the locomotion phenomenon of the autonomous vehicle. Consequently, the several models were brought together in the form of a complex simulation program.

The program is made in the form of an app is written in Matlab (www.mathworks.com) language. It is a didactic application aiming to better understand the phenomena that occur during locomotion. To achieve this goal, the user can easily undertake simulations by changing the model parameters. In addition to the mentioned flexibility, the user can perform case studies, scenarios, which allow the inheritance and addition of data in the program's database. In addition, the application contains the theoretical elements thus providing a unitary information and simulation environment.

The Architecture of the Simulator

The simulator has a structure obtained by aggregating seven objects. Figure 1 shows the block diagram of the mentioned architecture, the seven objects and their connection to the database. The architecture is modular, new modules (objects) can be imagined and can be added to enriched the program with new simulation methods.

Each of the objects is a module dedicated to studying its contributions to the simulation. These modules are for studying the effects on the autonomous vehicle of the trajectories, controller, locomotion models, sensors, filters, environment models, and for simultaneous localization and mapping (SLAM). The connection between them is possible through a database, this framework ensures the continuity of the simulations. The user can benefit from the data introduced in previous simulations or can add new data to the database. At the end of the interactive visual simulation, the user is obtaining a holistic report with the results.

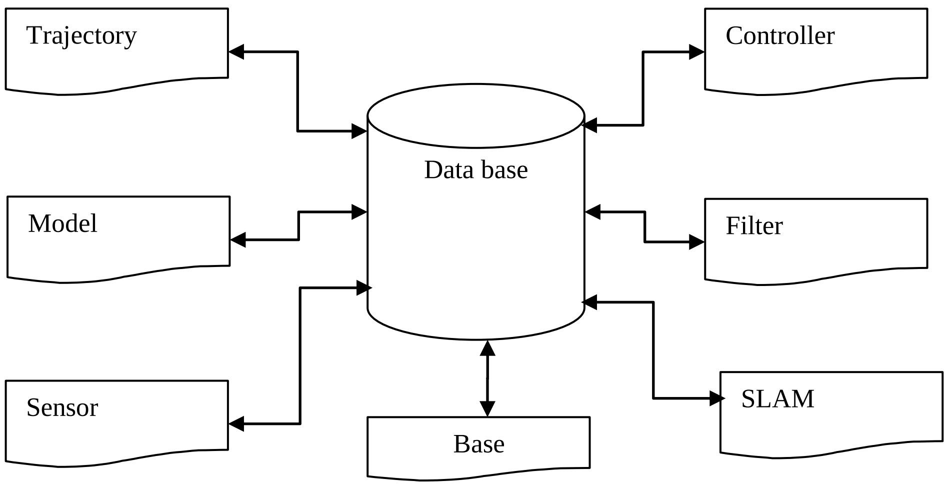


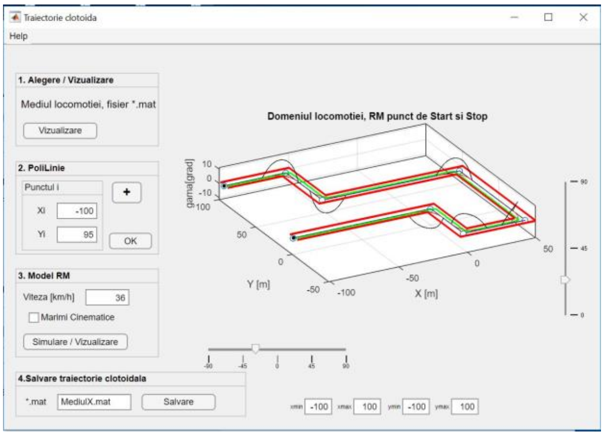
Figure 1. The autonomous vehicle simulator's architecture

The Modules of the Simulator

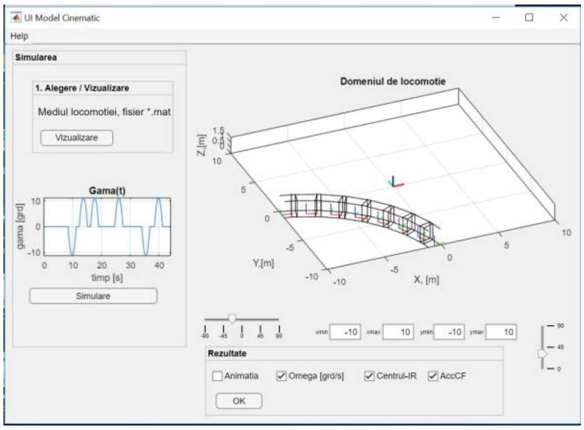
The modules and their features are presented in Table 1. Each of the modules can be used for testing and comparing different algorithms in the same area. Also, the user can change the parameters of the algorithm and visualize the results with plots and animations.

Table 1. The modules of the interactive simulation program

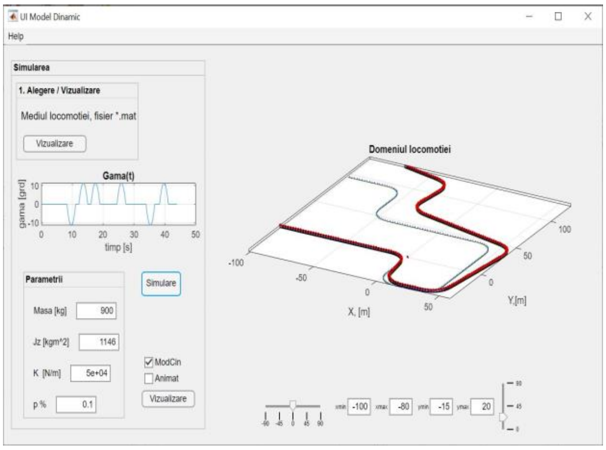
Module	Components	Observations
Trajectories	The domain of locomotion	It allows the definition of the locomotion environment by specifying multiple obstacles in the form of polylines. Here the environment discretion can be set.
	A* trajectory	The locomotion environment can be imported; starting position and destination points can be defined; one can calculate the trajectory between these points using the A* algorithm. The obtained trajectory avoids the obstacles that are in the field of locomotion.
	Clothoid trajectory	The locomotion environment can be imported. Limit points and guiding points can be specified, as well as the vehicle's speed. These allow the computation of the trajectory with clothoid segments and of the steering angle as a function of time.
Models	Geometric Model	The geometric model of the autonomous vehicle can be defined. The visualization can take several forms of the geometric model associated with the vehicle: a reference system; a triangle; a rectangle and a box shape. Here the posture can be changed.
	Cinematic Model	It allows the definition of the steering angle, as the control value, which in the end will lead to the representation of the trajectory and of the associated kinematic quantities. It allows the import of the steering angle of the trajectory with clothoid segments and the simulation of the vehicle's kinematics.
	Dynamic Model	Allows the definition of the steering angle, the control parameter and the inertial properties (mass, rolling inertia), which will ultimately lead to the presentation of the vehicle's trajectory. Here the over and understeering effects are highlighted. It allows the import of the steering angle of the trajectory with clothoid segments and the simulation of the vehicle's dynamics.
Sensors	LIDAR	It allows the import of the locomotion environment and the trajectory of the vehicle. The LIDAR parameters can be changed. It simulates the measurement of obstacles, the definition of the map and the occupancy grid.
	Odometer	Using the dynamic model with slip, the trajectory of the vehicle, obtained with odometric measurements is simulated.
Controllers	PID trajectory	Allows the import of the locomotion environment of the autonomous vehicle, the trajectory, the LIDAR sensor, the particle filter and simulates the controlled trajectory of the vehicle.
	Lateral dynamics	It permits the import or definition of the command parameter to control the vehicle for the elimination of the effects of lateral dynamics. Also, permits the tracking of the desired trajectory. The dynamic and desired controlled trajectories are illustrated.
Filters	Bayesian	The control values and measurements are defined, the probability distribution of the vehicle's position is shown.
	Extended Kalman Filter	The control quantities are defined, the probability distributions of the vehicle's position are calculated for various parameters of the sensors.
	Particle filter	
SLAM	Extended Kalman Filter method	The control quantities are defined, the probability distributions of the vehicle's position are calculated for various parameters of the sensors.
	Particle filter method	
Environment model		The general physical parameters are defined: absolute reference frame, friction, gravity, etc.



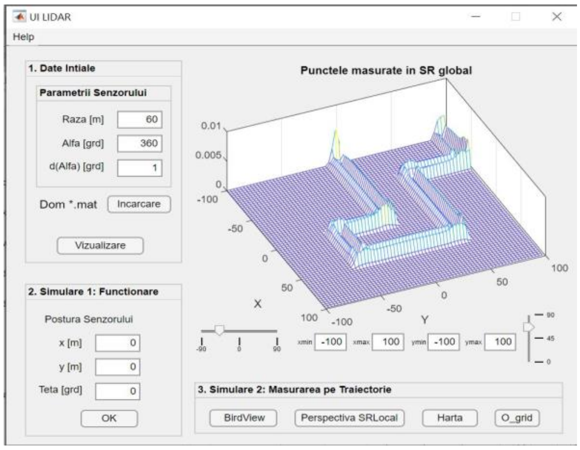
a



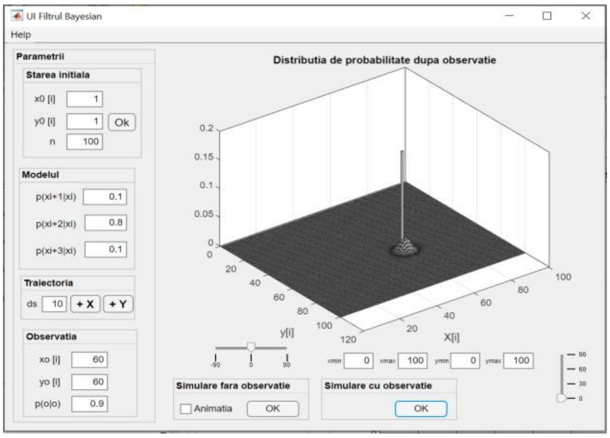
b



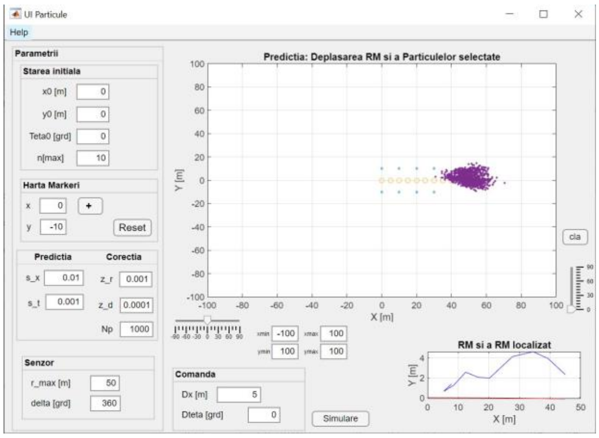
c



d



e



f

Figure 2. The interface of the simulation program

Some screenshots of the user interface of the simulator are shown in Figure 2. These are for the (a) trajectory with clothoid segments, (b) the kinematic model, (c) the dynamic model, (d) LIDAR model, (e) Bayes filter, and (f) particle filter.

With the simulator, the user has the possibility to study a certain phenomenon (filters, kinematic, dynamic, control, etc.) by modifying the model parameters (length, mass, resolution, tuning parameters, etc.) and the control values (linear speed, angular velocity, etc.) which, in the end, will allow the visualization of their influence.

In order to understand the holistic phenomenon of locomotion, the simulator also allows case studies. It is about linking the mentioned phenomena with the help of a database that takes over the effects of one phenomenon and transforms them into the causes of other phenomena.

Conclusion

The program is dedicated to autonomous vehicles (a mobile robot with four wheels) and allows the visualization of different locomotion scenarios. It is made in the form of an app is written in Matlab language. It is a didactic application aiming to better understand the phenomena that occur during locomotion, data processing, and decision making. The user can easily undertake simulations by changing the model parameters, can perform case studies, scenarios, which allow the inheritance and addition of data in the program's database. In addition, the application contains the theoretical elements thus providing a unitary information and simulation environment.

Acknowledgements

This work was supported by a grant of the Romanian Ministry of Research, Innovation and Digitization, CNCS/CCCDI – UEFISCDI, project number PN-III-P2-2.1-PED-2019-4366 within PNCDI III (431PED).

Scientific Ethics Declaration

The authors declare that the scientific ethical and legal responsibility of this article published in EPSTEM journal belongs to the authors.

References

Chen, Y., Chen, S., Zhang, T., Zhang, S., & Zheng, N. (2018, June). Autonomous vehicle testing and validation platform: Integrated simulation system with hardware in the loop. In *2018 IEEE Intelligent Vehicles Symposium (IV)* (pp. 949-956). IEEE.

Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., & Koltun, V. (2017). CARLA: An open urban driving simulator. In *Conference on robot learning* (pp. 1-16). PMLR.

Huang, W., Wang, K., Lv, Y., & Zhu, F. (2016). Autonomous vehicles testing methods review. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 163-168). IEEE.

The MathWorks Inc (2020). MATLAB version 9.8.0. Natick.

Pozna, C. R., & Antonya, C. (2016). Issues about autonomous cars. In *2016 IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI)* (pp. 13-18). IEEE.

Pozna, C. R., & Antonya, C. (2021). Proposal of an autonomous vehicle control architecture. In *2021 IEEE 25th International Conference on Intelligent Engineering Systems (INES)* (pp. 37-42). IEEE.

Rosique, F., Navarro, P. J., Fernández, C., & Padilla, A. (2019). A systematic review of perception system and simulators for autonomous vehicles research. *Sensors*, *19*(3), 648.

Wen, M., Park, J., & Cho, K. (2020). A scenario generation pipeline for autonomous vehicle simulators. *Human-centric Computing and Information Sciences, 10*, 1-15.

Author Information

Claudiu Radu POZNA
Transilvania University of Brasov
Department of Automation and Information Technology
29 Eroilor Blvd, 500036 Brasov, Romania
Széchenyi István University
Department of Computer Science
Egyetem tér 1, 9026 Győr, Hungary

Csaba ANTONYA
Transilvania University of Brasov
Department of Automotive and Transport Engineering
29 Eroilor Blvd
500036 Brasov, Romania
Contact e-mail: *antonya@unitbv.ro*

To cite this article:

Pozna, C.R. & Antonya, C. (2021). Interactive simulation program for autonomous vehicles. *The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM), 16*, 26-30.