

Article

The Design of Performance Guaranteed Autonomous Vehicle Control for Optimal Motion in Unsignalized Intersections

Balázs Németh ^{1,2,*}  and Péter Gáspár ¹ 

¹ Institute for Computer Science and Control (SZTAKI), Eötvös Loránd Research Network (ELKH), 13-17. Kende utca, H-1111 Budapest, Hungary; peter.gaspar@sztaki.hu

² Department of Control for Transportation and Vehicle Systems, Budapest University of Technology and Economics, 3-7. Műegyetem rakpart, H-1111 Budapest, Hungary

* Correspondence: balazs.nemeth@sztaki.hu or nemeth.balazs@kjk.bme.hu

Abstract: The design of the motion of autonomous vehicles in non-signalized intersections with the consideration of multiple criteria and safety constraints is a challenging problem with several tasks. In this paper, a learning-based control solution with guarantees for collision avoidance is proposed. The design problem is formed in a novel way through the division of the control problem, which leads to reduced complexity for achieving real-time computation. First, an environment model for the intersection was created based on a constrained quadratic optimization, with which guarantees on collision avoidance can be provided. A robust cruise controller for the autonomous vehicle was also designed. Second, the environment model was used in the training process, which was based on a reinforcement learning method. The goal of the training was to improve the economy of autonomous vehicles, while guaranteeing collision avoidance. The effectiveness of the method is presented through simulation examples in non-signalized intersection scenarios with varying numbers of vehicles.



Citation: Németh, B.; Gáspár, P. The Design of Performance Guaranteed Autonomous Vehicle Control for Optimal Motion in Unsignalized Intersections. *Appl. Sci.* **2021**, *11*, 3464. <https://doi.org/10.3390/app11083464>

Academic Editor: Luigi Fortuna

Received: 26 February 2021

Accepted: 6 April 2021

Published: 13 April 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: coordinated control design; autonomous vehicles; learning-based control

1. Introduction and Motivation

The handling of intersection scenarios is an important challenge in the research field of autonomous vehicles. From the aspect of control design, several crucial tasks of the autonomous vehicle control in intersection scenarios should be formulated. For example, the ordering of autonomous vehicles in intersections has an impact on the energy consumption of the vehicles, traveling time, emissions and traveling comfort (due to the acceleration/deceleration maneuvers, for example) [1]. This leads to a multi-objective optimization task, which generally has a Pareto-optimal solution. Moreover, the solution of the optimization problem regarding autonomous vehicles in intersections can require lengthy computations. This poses the challenge of the minimization of the computational time, e.g., finding approximations of the optimal solution. From an application point of view, the accurate detection of a vehicle's position in an intersection can be difficult. There exist camera-based vehicle localization strategies [2], solutions with LiDAR and camera fusion [3] and GPS and stereovision-based visual odometry solutions [4]. The detection and prediction of human behavior are also difficult and complex problems; see [5,6]. Nevertheless, considering position errors caused by human participants, robust control strategies are needed with which collision avoidance can be guaranteed.

Due to the special challenges of the control of autonomous vehicles in intersections, in the architecture of the control system it can be handled as a special feature of a path planning algorithm in future vehicles. Its main differences from the path planning problem are as follows. First, in intersections the autonomous vehicles can have interactions. Since the other vehicles are also in motion, the handling of intersection scenarios is a complex problem, which requires sensing, decision and control functionalities. In intersections the

risk of the collision is increased, e.g., compared to the motion of the autonomous vehicle along a highway. Second, various types of intersections on the road can be found—they can depend on the number of lanes, directions and signalization. A challenge of the autonomous vehicles and the intelligent transportation systems is the handling of vehicle interactions in unsignalized intersections [7]. In these types of intersections, the experiences of the human drivers has a role in the avoidance of the conflicts; e.g., the speed of the surrounding vehicles and the intentions of the further drivers must be estimated. In a general path planning algorithm, the waypoints, the rules and the information on the road and on the local traffic conditions can be considered to be given, and thus, it requires less estimation. Third, in urban areas the autonomous vehicles during their routes must travel through many intersections. Since these intersections are connected to each other, the motion of the vehicles in the intersections has an impact on the global performance of the traffic system, i.e., on the average speed or on the traffic density; see, e.g., [8]. Thus, during the design of the autonomous vehicle's motion, the time minimization is an especially important at intersections. Moreover, the inefficient motion of autonomous vehicles in intersections can lead to increased energy consumption, e.g., a high number of stop and go maneuvers. Thus, during the design of the autonomous vehicle motion in intersections, a balance between the time minimization and the energy consumption minimization must be found. Although it is also a problem in a general path planning problem, it is a greater problem for intersections. Fourth, a special problem of vehicle control in intersections is the architecture of the control system. Since in intersections a high number of vehicles can be controlled and the presence of human-driven vehicles must be handled, an optimal solution of the motion problem can be found through the coordination of the autonomous vehicles in a centralized control system [9]. Nevertheless, it can require centralized control, which can lead to additional security, legal and privacy problems. Therefore, it is requested to develop distributed control structures; e.g., in [10] the problem is formulated as distributed optimal control for a system of multiple autonomous vehicles and then solved using a nonlinear model predictive control (MPC) technique. Similarly, Kloock et al. [11] investigated intersection control of multiple vehicles using a distributed MPC approach, where priorities need to be assigned to vehicles. Thus, the challenge of distributed control is also different to the general path planning and tracking problems, where the coordination of the vehicles is limited, or where it is carried out on the level of the transportation system.

The high number of challenges in the field of autonomous vehicle control in intersections has motivated increased research activity, i.e., several publications with different approaches. Methods of one of the most important groups are based on the MPC technique; see, e.g., [12–15]. Although they can provide appropriate results, the increase in the number of vehicles can make real-time computation difficult. A possible solution to the problem of increasing computational effort is the approximation of the optimal solution with neural networks; see, e.g., [16]. However, in this case it may be difficult to provide guarantees on collision avoidance for all intersection scenarios [17]. The quadratic programming method gives the possibility of real-time implementation compared to the convex optimization using space coordinates; see [18]. Collision avoidance is guaranteed through the definition of regions in the intersection with special rules, which reduces the complexity of the problem, but the conservativeness of the solution increases. Due to the ordering problem of vehicles in an intersection, the control problem can be formed as a mixed-integer linear programming (MILP) task; see, e.g., a method on the coordination in [19]. The goal of the control is to find an arrival schedule of the vehicles, which ensures safety while it reduces the number of stops and intersection delays. Muller et al. [20] presents a centralized MILP-based approach for intersection control in an urban environment of highly automated vehicles, with which the minimum vehicle delay at the intersection can be achieved. Another solution in the predictive control framework is to examine all ordering combinations of the vehicles [21], which can also be difficult for a high number of vehicles. Thus, although MPC-based methods can provide effective and handy solutions to the

control problem of autonomous vehicles in intersections, a high number of vehicles can lead to numerical difficulties.

A further group of solutions is based on the minimax formulations of the optimal control problem, together with the coordination of the vehicles. For example, Tian et al. [22] proposed a game-theoretic approach to modeling vehicle interactions, in particular, for urban traffic environments with unsignalized intersections. In general, they are traffic models with heterogeneous control, and they are used for virtual testing, evaluation and calibration of autonomous vehicle control systems. A framework for intersection control, which is based on the queuing theory, was presented by [23]. Through this framework a capacity-optimal, slot-based intersection management system for the two-road crossing configuration was presented. A slot preassigning method for connected vehicles was also proposed in [24]. The connection of intersections can also be handled by the multi-agent viewpoint [25,26]. This solution provides the ability to quickly reverse individual lanes in response to rapidly changing traffic conditions. A further multi-agent solution based on a heuristic optimization algorithm was presented by [27]. The objective of the research was to reduce total time delay for the entire intersection, while collisions were prevented. The approaches in the field of trajectory planning are possible ways to formulate the problem of autonomous vehicle motion in intersections. The actual methods and the future challenges in path planning are summarized in [28]. Some similarities between the path planning and the problem of autonomous vehicles in intersections motivated the use of Bézier-based methods in the problem. For example, in [29] a Bézier curve optimization method was proposed to cope with these constraints, and autonomous vehicles were considered that were equipped with all the necessary sensors for obstacle detection. In this way, the obstacle avoidance problem was transformed into an optimization problem under equality constraints. Another method with the use of Bézier curves was found in [30], which presents a novel method to resolve the obstacle avoidance and overtaking problems named Hybrid Planning. An MPC-based solution to track the lane centerline while avoiding collisions with obstacles was found in [31]. Further relevant works in the topic of path planning are as follows. Hu et al. [32] presented a real-time dynamic path planning method; i.e., an optimal path with appropriate acceleration and velocity profiles is the result. A graph-based planning method to generate an actionable set of multiple drivable trajectories for race vehicles, is found in [33]. For road vehicles, Werling et al. [34] proposed an optimal trajectory generation method, with which velocity maintenance, merging, following, stopping and a reactive collision avoidance functionality were achieved. In [35] a real-time safe stop trajectory planning algorithm, based on selection from a precomputed set of trajectories has been developed. Several papers deal with the selection of the velocity of autonomous vehicles, which has relevance in the problem of intersection scenarios. For example, Herrmann et al. [36] proposed an optimization-based velocity planner using the multi-parametric sequential quadratic problem, with which a spatially and temporally varying friction coefficient, and a race-focused energy strategy for the road can be handled. In [37] a solution to the mathematical problem behind the trajectory design problems has been provided; i.e., a general-purpose solver for convex quadratic programs based on the alternating direction method of multipliers is proposed. Moreover, in [38] a real-time velocity optimization algorithm for fixed paths was developed and implemented.

Another important group of efficient solutions on the problem contains the learning-based approaches, especially reinforcement learning (RL) [39–42]. The advantage of this method is its model-free property, which can provide solutions to the problem of constraint formulation. Moreover, the training of the control agent through a high number of epochs is carried out, which can lead to improved performances. A deep RL method was applied to unsignaled intersections in [39]. It also provided an analysis to explore the ability of the system to learn active sensing behaviors and safe navigation in the case of occlusions. Tran and Bae [43] also presented a deep RL-based model that considers the effectiveness of leading autonomous vehicles in mixed-autonomy traffic at a non-signalized intersection. Yudin et al. [44] offered a new approach to training the intelligent agent that simulates

the behavior of an unmanned vehicle, based on the integration of reinforcement learning and computer vision. Using full visual information about the road intersection obtained from aerial photographs, they studied through automatic detection the relative positions of all road agents with various architectures of deep neural networks. In spite of promising achievements, the resulting neural-network-based agents cannot provide a guarantee of collision avoidance for the vehicles.

That short review of the existing literature showed that several efficient solutions exist, but there are challenges for achieving a reliable control method. The goal of the recent research activities was to find a control design method with which many autonomous vehicles can be handled, while the energy consumption and time minimization requirements, along with safety, i.e., the avoidance of collisions, are ensured. Similarly, the purpose of this paper is to provide a novel solution to the recent problems through the following concept. The increase in the number of vehicles is handled through separation, i.e., the control designed for individual autonomous vehicles, which considers limited information on the environment. In the solution the performance requirements are prioritized and the specifications are guaranteed through different methods, such as optimal control and RL-based techniques. Thus, in this way the advantages of each method, such as providing guarantees and optimization of the performance level, can be achieved.

This paper focuses on the optimal design of the longitudinal control of an autonomous vehicle. Its elements and the related problems regarding the paper can be summarized as follows.

- The objective may contain multiple criteria, the most important of which are the minimization of traveling time, energy consumption and the maximization of comfort. The variables of the optimization task are the control inputs of the vehicles, e.g., longitudinal acceleration commands or traction/braking forces.
- The dynamics of each vehicle provide constraints in the optimization problem.
- Furthermore, the safe motion of the vehicles, i.e., collision avoidance, must be guaranteed, which leads to constraints for vehicles with intersecting routes.
- The velocity of the vehicles must be kept in a predefined range, considering the speed limit.
- The control input of the vehicles must also be limited due to the physical limits of the driveline, the braking system and the tire–road contact.

In this paper the optimization problem of the vehicle's motion is divided according to the performance aspects. First, a quadratic optimization problem is proposed, whose role is to guarantee collision avoidance and the limitation of the velocities of vehicles. The constraint of collision avoidance is formed through a linear approximation of the quadratic constraint, which leads to high efficiency in computation. The quadratic optimization is solved at every time step during the motion of the vehicles. Second, the advantages of reinforcement learning in the improvement of the economy performances, e.g., the minimization of the control input, are exploited. In the training process the previously formed quadratic optimization task is applied as a part of the environment for learning. Similarly, during the operation of the control system, the trained neural network and the quadratic optimization task operate together. The effectiveness of the resulting control system through simulation examples with various vehicles is analyzed.

Thus, the contributions of the paper are as follows. The control design problem for autonomous vehicles in intersections is formed in a novel way through the division of the problem. It leads to the reduced complexity of the control problem in real-time computation. Moreover, an environment model for reinforcement learning is created, which guarantees that collision avoidance can be provided. An important benefit of the method is that the safety performance level of the resulting control structure is guaranteed; i.e., a collision of vehicles cannot be caused by the learning-based control agent. From an application point of view, an advantage of the method is that it is able to handle various situations at the intersection; i.e., the method is independent of the number of vehicles. Although this paper focuses on autonomous vehicle control, the proposed control design method may be used

for further applications, considering the similarities with the intersections. For example, in the field of microfluid research, an actual problem is to find modeling and control methods for the handling of T-junctions [45]. In [46] computational models for microfluidic bubble logic systems with T-junctions have been proposed. Learning methods have also been used for control purposes in this context; see, e.g., [47]. Thus, the contributions of the proposed control system may have relevance to further applications.

The paper is organized as follows. In Section 2 the model formulation for handling intersection scenarios is proposed. The cruise control design for the autonomous vehicle with robustness issues is found in Section 3. The application of reinforcement learning for the improvement of economy performances is presented in Section 4. In Section 5 the effectiveness of the proposed method is shown through simulation examples. Finally, the conclusions and the future challenges are provided in Section 6.

2. Model Formulation for Handling Intersection Scenarios

The goal of this section is to formulate the collision-free motion of the vehicles in intersection scenarios. The formulation includes the longitudinal motion model of the vehicles and the constraints on their motion to avoid collisions.

The longitudinal motion of the vehicles is formulated through the simplified kinematics of vehicles:

$$v_i(k+1) = v_i(k) + Ta_i(k), \quad (1)$$

$$s_i(k+1) = s_i(k) + Tv_i(k) + \frac{T^2}{2}a_i(k), \quad (2)$$

where i index represents the id of the vehicle, n is the number of vehicles, v_i is longitudinal velocity and s_i is longitudinal displacement. a_i represents the longitudinal acceleration of the vehicle, which is handled as a control input command, and T is the time step of the discrete motion model. The longitudinal displacement is related to the center point of the intersection, and thus, it is defined as $s_i = 0$ for all i at the center point. The longitudinal displacement of the approaching vehicle has a negative value, and the displacement of the vehicle moving away has a positive value.

In the environment model of the intersection, the autonomous ego vehicle is numbered 1. Thus, for example, in the case of $n = 5$ in the environment, the ego vehicle and four surrounding vehicles are incorporated. The control input of the autonomous vehicles is separated into two elements, as has been introduced in Section 1 as

$$a_1(k) = a_K(k) + \Delta(k), \quad (3)$$

where a_K is the control input command of the robust longitudinal controller and $\Delta(k)$ is the additional input from the supervisor in the model.

The purpose of the supervisor in the collision-free motion model is to select $\Delta(k)$ for the autonomous ego vehicle with the following objectives and constraints.

- The objective of the selection is to minimize the difference between the current control input $a_1(k)$ and the output of the learning-based controller $a_L(k)$. The aim of the minimization is to preserve the performance level of the learning-based controller, if possible.
- Some of the vehicles can intersect their routes at the intersection. Thus, an intervention with $a_1(k)$ must provide motion for the autonomous vehicle, with which a safe distance s_{safe} between the autonomous vehicle and the other surrounding vehicles with intersecting routes can be guaranteed. Since the number of vehicles n can be high, a limited amount of surrounding vehicle motion must be simultaneously considered; i.e., n_s represents the number of vehicles which are incorporated into the design process. It is a constraint in the selection of Δ .
- The velocity of the vehicles must be inside of a predefined range. The upper bound of the range is defined by the speed limit v_{max} or the curvature of the intersection.

The lower bound is represented by the stopping of the vehicle. Thus, it is necessary to select Δ for all vehicles to keep velocities inside of the range. It is also a constraint in the selection process.

The selection process of Δ for the autonomous ego vehicle is formed as an optimization problem as follows:

$$\min_{\Delta(k)} (a_1(k) - a_L(k))^2 \tag{4a}$$

subject to

$$(s_1(k+1) - s_j(k+1))^2 \geq s_{safe}, \quad \forall j \in n_s, \tag{4b}$$

$$0 \leq v_1(k+1) \leq v_{max}, \tag{4c}$$

$$\Delta \in \Delta, \tag{4d}$$

where j represents the surrounding vehicles, whose motion can be in conflict with the autonomous vehicle; i.e., their routes are intersected. Δ represents the domain of the optimization variable. In the optimization problem the kinematics of the vehicle motion (1) is considered through the formulation of the constraints, the separation of the control input (3) is involved in the objective function.

The objective (4a) is transformed using (3) as

$$(a_1(k) - a_L(k))^2 = (a_K(k) + \Delta(k) - a_L(k))^2 = \Delta(k)^2 + 2f(k)\Delta(k) + (a_K(k)^2 + a_L(k)^2 - 2a_L(k)a_K(k)), \tag{5}$$

where f (a vector) is formed as $f(k) = a_K(k) - a_L(k)$. Since $(a_K(k)^2 + a_L(k)^2 - 2a_L(k)a_K(k))$ is independent of $\Delta(k)$, it is omitted in the rest of the optimization problem.

The constraint for collision avoidance (4b) is formed to keep s_{safe} between the vehicles. The distance is measured in the sense of the longitudinal displacements of the vehicles on their routes. Geometrically, the quadratic constraints (4b) represent that the trajectories of the autonomous vehicle s_1 and the related surrounding vehicles must be out of a circle. The radius of the circle is defined by s_{safe} ; see Figure 1a. For example, if there are two surrounding vehicles and the autonomous vehicle the trajectories must be out of a sphere, whose radius is represented by s_{safe} ; see Figure 1b.

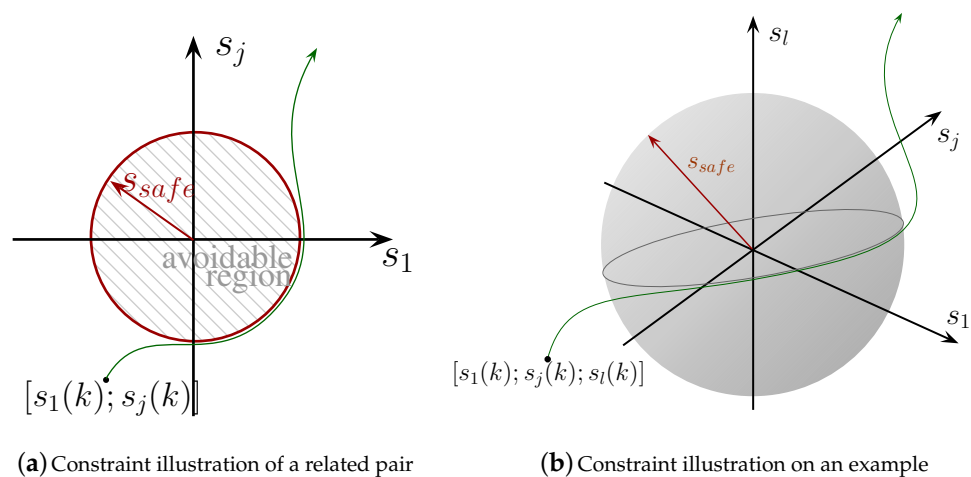


Figure 1. Geometrical illustration of the quadratic constraints.

Although the circle perfectly describes the region of = state-space that must be avoided, it leads to a quadratic constraint in the optimization problem (4). Since the optimization problem must be solved at each k time step during the motion of the vehicles, the quadratic constraint can make the solution of the optimization more complex, and it can make

the real-time solution more difficult. This motivates the transformation of the quadratic constraints into linear constraints. In this paper it is performed through the approximation of the avoidable regions of circles to the avoidable regions of half-planes.

The method of the approximation is illustrated in Figure 2. First, the tangent lines to the circle from the actual state $[s_1(k) \ s_j(k)]^T$ are assigned. The avoidable half-plane is determined by the region between the tangent lines; i.e., the trajectories must be out of it. Second, two linear inequality constraints are specified, which represent that the trajectory of the state must be out of the avoidable half-plane:

$$\begin{bmatrix} s_{T1,1}(k) \\ s_{T1,j}(k) \end{bmatrix}^T \begin{bmatrix} s_1(k) \\ s_j(k) \end{bmatrix} \leq \begin{bmatrix} s_{T1,1}(k) \\ s_{T1,j}(k) \end{bmatrix}^T \begin{bmatrix} s_1(k+1) \\ s_j(k+1) \end{bmatrix}, \tag{6a}$$

or

$$\begin{bmatrix} s_{T2,1}(k) \\ s_{T2,j}(k) \end{bmatrix}^T \begin{bmatrix} s_1(k) \\ s_j(k) \end{bmatrix} \geq \begin{bmatrix} s_{T2,1}(k) \\ s_{T2,j}(k) \end{bmatrix}^T \begin{bmatrix} s_1(k+1) \\ s_j(k+1) \end{bmatrix}, \tag{6b}$$

where $[s_{T1,1}(k) \ s_{T1,j}(k)]$, $[s_{T2,1}(k) \ s_{T2,j}(k)]$ are the tangent points on the circle at time step k . Note that the linear constraints result in an outer approximation of the avoidable set. There are regions of the half-space which cannot be reached due to the linear constraints, but they are out of the circle.

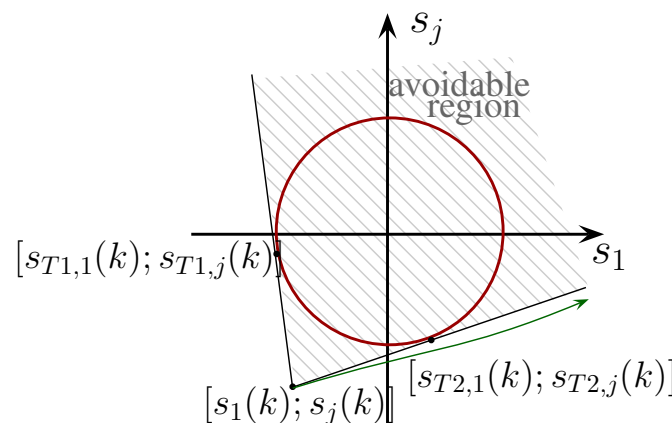


Figure 2. Illustration of constraint approximation.

The longitudinal displacement of the autonomous vehicle at $k + 1$ in (6) is transformed to express the linear constraints in terms of Δ . The transformation is based on the motion equation (1) and the relation of actuation separation (3). Moreover, the motions of surrounding vehicles for $k + 1$ with their actual velocity $v_j(k)$ are predicted this way:

$$s_1(k+1) = s_1(k) + Tv_1(k) + \frac{T^2}{2}a_K(k) + \frac{T^2}{2}\Delta(k), \tag{7a}$$

$$s_j(k+1) = s_j(k) + Tv_j(k), \tag{7b}$$

which can be substituted into (6). It leads to the linear constraints

$$\begin{bmatrix} s_{T1,1}(k) \\ s_{T1,j}(k) \end{bmatrix}^T \begin{bmatrix} -Tv_1(k) - \frac{T^2}{2}a_K(k) \\ -Tv_j(k) \end{bmatrix} \leq \frac{T^2}{2} \begin{bmatrix} s_{T1,1}(k) \\ 0 \end{bmatrix}^T \Delta(k), \tag{8a}$$

or

$$\begin{bmatrix} s_{T2,1}(k) \\ s_{T2,j}(k) \end{bmatrix}^T \begin{bmatrix} -Tv_1(k) - \frac{T^2}{2}a_K(k) \\ -Tv_j(k) \end{bmatrix} \geq \frac{T^2}{2} \begin{bmatrix} s_{T2,1}(k) \\ 0 \end{bmatrix}^T \Delta(k). \tag{8b}$$

Figure 2 illustrates that the reachable set for the state $[s_1(k+1) \quad s_j(k+1)]^T$ is non-convex, which means that (8) formulates disjunctive inequalities. However, each constraint in (8) leads to convex reachable sets, which means that the optimization problem can be divided, as it is proposed below.

Another constraint in the optimization (4) is on the velocity of the autonomous vehicle; see (4c). In the case of this constraint, $v_i(k+1)$ is expressed in terms of Δ using the motion equation (1) and the relation of actuation division (3). The linear inequality constraint is formed as

$$0 \leq v_1(k) + Ta_K(k) + T\Delta(k), \quad (9a)$$

$$v_{max} \geq v_1(k) + Ta_K(k) + T\Delta(k), \quad (9b)$$

which leads to

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix} \Delta(k) \leq \begin{bmatrix} \frac{v_1(k)}{T} + a_K(k) \\ \frac{v_{max}-v_1(k)}{T} - a_K(k) \end{bmatrix}, \quad (10)$$

where v_{max} is the maximum velocity of the autonomous vehicle. In an urban environment it is determined by the velocity regulations, and in the case of cornering maneuvers at intersections, the curvature avoids skidding.

The last constraint in the optimization problem (4) is the limitation of the resulting optimization variable; see (4d). The value of Δ is limited by the bounds of $a_1(k)$, such as a_{min} and a_{max} , which represent full braking and throttle. Since $a_1(k)$ is also influenced by $a_K(k)$, the constraints on $\Delta(k)$ are formed as

$$a_{min} - a_K(k) \leq \Delta(k), \quad (11a)$$

$$a_{max} - a_K(k) \geq \Delta(k), \quad (11b)$$

which leads to the constraint

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix} \Delta(k) \leq \begin{bmatrix} a_K(k) - a_{min} \\ a_{max} - a_K(k) \end{bmatrix}. \quad (12)$$

The optimization task (4) using (5), (8), (10) and (12) is reformulated as

$$\min_{\Delta(k)} \Delta(k)^2 + 2f^T \Delta(k) \quad (13a)$$

subject to

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix} \Delta(k) \leq \begin{bmatrix} \frac{v_1(k)}{T} + a_K(k) \\ \frac{v_{max}-v_1(k)}{T} - a_K(k) \end{bmatrix} \quad (13b)$$

and

$$\begin{bmatrix} -1 \\ 1 \end{bmatrix} \Delta(k) \leq \begin{bmatrix} a_K(k) - a_{min} \\ a_{max} - a_K(k) \end{bmatrix} \quad (13c)$$

and

$$\begin{bmatrix} s_{T1,1}(k) \\ s_{T1,j}(k) \end{bmatrix}^T \begin{bmatrix} -Tv_1(k) - \frac{T^2}{2}a_K(k) \\ -Tv_j(k) \end{bmatrix} \leq \frac{T^2}{2} \begin{bmatrix} s_{T1,1}(k) \\ 0 \end{bmatrix}^T \Delta(k), \quad \forall j \in n_s \quad (13d)$$

or

$$\begin{bmatrix} s_{T2,1}(k) \\ s_{T2,j}(k) \end{bmatrix}^T \begin{bmatrix} -Tv_1(k) - \frac{T^2}{2}a_K(k) \\ -Tv_j(k) \end{bmatrix} \geq \frac{T^2}{2} \begin{bmatrix} s_{T2,1}(k) \\ 0 \end{bmatrix}^T \Delta(k), \quad \forall j \in n_s. \quad (13e)$$

The quadratic optimization in (13) contains disjunctive inequalities. This means that the optimization task for the solution can be reformulated to a mixed-integer optimization

problem [48]. In practice, the optimization problem (13) can be solved as a set of quadratic optimization problems, which has only “and” conditions between the constraints. Thus, in the case of n_s surrounding vehicles, 2^{n_s} distinct constrained optimization problems can be formed. Since the objective function for each problem is the same, the global minimum solution can be found through a comparison of their solutions. Finally, the solution is yielded by $\Delta(k)$, which leads to the minimum value of the objective function, considering all of the optimization tasks. The optimization problem during the motion of the autonomous vehicle is continuously solved.

3. The Design of Robust Cruise Control

An intervention in the longitudinal dynamics of the autonomous vehicle at an intersection has great importance for achieving the required motion profile for time step $k + 1$. The control input of the vehicle is composed by two elements, $a_K(k)$ and $\Delta(k)$; see (3). The computation of $\Delta(k)$ in Section 2 has already been proposed. In this section the design of the control intervention $a_K(k)$ is proposed.

The most important requirements for the controller are formed as follows.

- The control system must guarantee safe longitudinal motion for the vehicle, even if a_L has degradation or a fault scenario. The safe motion is guaranteed through a_K and Δ .
- The longitudinal controller must have robust characteristics, because $a(k) \neq a_K(k)$, if $\Delta(k) \neq 0$. Robustness must also be guaranteed in extreme vehicle dynamic scenarios: for example, if $a_K(k)$ suggests full throttle, but it has been overridden by $\Delta(k)$ to be full braking. Similarly, robustness must be guaranteed if $a_K(k)$ suggests full braking, but it has been overridden by $\Delta(k)$ to be full throttle.
- The control signal $a_K(k)$ must be computed to avoid the saturation of the control actuation $a(k)$. Since in the computation process the computation of $a_K(k)$ is prior to $\Delta(k)$, the result of (13) is not considered in the computation of $a_K(k)$. Thus, the role of the cruise controller is to guarantee $a_{min} \leq a_K(k) \leq a_{max}$.

The design of the longitudinal controller is based on the simplified vehicle model (1) with one state, such as the longitudinal velocity $v_1(k)$. The controller is formed as a gain P for velocity tracking, whose input is the velocity error; see Figure 3. The reference signal for the tracking is v_{ref} . The idea behind the formulation of the control design as a tracking problem is as follows. The vehicle must adapt to the environment, which determines the achievable velocity of the vehicle, e.g., through maximum speed limits. The aim of the cruise control is to consider the regulations and the motion of the other vehicles to achieve safe motion. The maximum speed limit information is provided by $v_{ref}(k)$, and information on the motion of the vehicles in the environment in the computation of $\Delta(k)$ is included. Thus, through the selections of $v_{ref}(k)$ and $\Delta(k)$, safe motion is guaranteed.

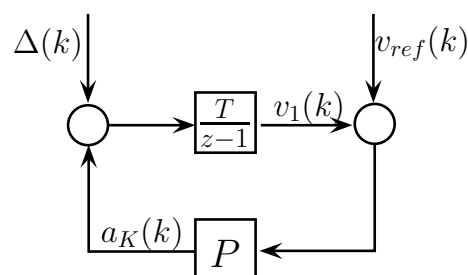


Figure 3. Closed-loop structure of the longitudinal control.

The selection of P is determined by the requirement of robustness by the controlled system. In the presented structure, $\Delta(k)$ is handled as a disturbance and the performance of the system is the minimization of the tracking error, such as $|v_{ref}(k) - v_1(k)|$. Thus, for the robustness issue, it is necessary to limit the impact of $\Delta(k)$ disturbance on the performance. The robustness criterion is based on the small-gain theorem [49]. As a consequence, the \mathcal{H}_∞

norm transfer function from $\Delta(k)$ on the velocity error must be smaller than 1, which guarantees disturbance attenuation.

In the computation of the transfer function the maximum value of the $\bar{\Delta} = |\Delta(k)|$ is also considered. The maximum is determined by the vehicle dynamic scenario, if $\Delta(k)$ implies full throttle instead of full braking or full braking is implied instead of full throttle. Thus, the value of $\bar{\Delta}$ is $|a_{min}| + |a_{max}|$. The transfer function from $\Delta_i(k)$ to the velocity error is

$$G(z) = \frac{\bar{\Delta} \frac{T}{z-1}}{1 + \frac{P \cdot T}{z-1}} \tag{14}$$

The criterion on the selection of P is to guarantee $\|G(z)\|_\infty < 1$.

The constraint on the control input $a_1(k)$ is influenced by the limitation of the reference signal. The reference velocity in Figure 3 represents the maximum speed limit $v_{max}(k)$ for the vehicle on the given road section. Nevertheless, it is necessary to provide a reference velocity for the vehicle, with which the constraint on $a(k)$ is kept. Thus, $v_{ref}(k)$ is computed as

$$v_{ref}(k) = v_1(k) + \min \left(\frac{a_{max}}{P}, \max \left(\frac{a_{min}}{P}, v_{max} - v_1(k) \right) \right) \tag{15}$$

The expression represents that the reference velocity can differ from that actual velocity with a limited value. The value of P has a role in the computation of the reference signal.

Finally, the method of the selection of P is as follows. It is recommended to select a value of P as high as possible, which can result in fast operation of the tracking control. However, the value of P is constrained by the robustness criterion:

$$\max P, \tag{16a}$$

subject to

$$\|G(z)\|_\infty < 1. \tag{16b}$$

The output of the robust longitudinal controller is $a_K(k)$, which is used in the computation of the control input (3) and in the computation of $\Delta(k)$ (13).

4. Design of Motion Profile Using Reinforcement Learning

In the previous sections a control strategy was formed with which the safe motion of the autonomous vehicle in an intersection can be guaranteed. Nevertheless, in the design problem of the vehicle motion there are economy aspects that should be considered during the design process; i.e., it is recommended to minimize the control energy of the vehicles. In this section the consideration of the economy performance in the design process is proposed.

The goal of the design process is to find an agent which is able to generate the control signal $a_L(k)$. The agent is trained through a reinforcement learning process. The structure of the design process is illustrated in Figure 4. The model for the learning process contains the optimization task (13) together with the result of the robust control design (16). The model guarantees the avoidance of the collision in the intersection for every $a_L(k)$ signal. Thus, during the training process of the agent in every episode the safety performances are guaranteed, and similarly, the economy performance is improved. The output of the motion model is reward $r(k)$, which is composed by $a(k)$ and $v_1(k)$ as follows:

$$r(k) = -Q_1 a_i^2(k) + Q_2 (s_1(k) - s_1(0)), \tag{17}$$

where Q_1 and Q_2 positive values are design parameters, which scale the importance of each term in $r(k)$. The reward contains the control inputs in the vehicles $a(k)$, which represent the economy performance of the vehicles. If the reward contains only $a(k)$, it can result

in unacceptably slow motion for the autonomous vehicle, because $a(k) = 0$ is the best choice for the maximization of the reward. Thus, in the reward the traveled distance of the autonomous vehicle ($s_1(k) - s_1(0)$) is also incorporated, where $s_1(0)$ is the initial position of the autonomous vehicle. The observation for the agent includes the position of the autonomous vehicles $s_1(k)$ and of the surrounding vehicles $s_j(k)$ $j \in n_s$, and their velocities $v_1(k), v_j(k)$ $j \in n_s$.

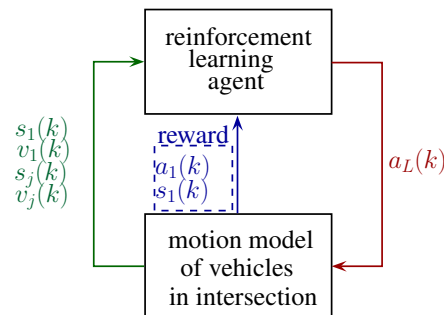


Figure 4. Control structure in the learning process.

The goal of the reinforcement learning process is to maximize reward (17) during episodes. In this paper the training process through the deep deterministic policy gradient (DDPG) is carried out, which is a model-free, online, off-policy reinforcement learning method [50]. A DDPG agent is an actor-critic reinforcement learning agent that computes an optimal policy that maximizes the long-term reward. In this method, actor and critic approximators are used. Both approximators use the observations, which are represented by S . The purpose of the actor approximator $\mu(S)$ is to find the action A with $a_L(k)$, which maximizes the long-term future reward. The role of the critic $Q(S, A)$ is to find the expected value of the long-term future reward for the task.

The result of the training process is an agent whose output is $a_L(k)$. In the control process of the autonomous vehicle, the agent works together with the control strategy (13). As a result, the collision avoidance of the vehicles in the intersection is guaranteed, and similarly, the economic performance of the vehicles is improved.

5. Simulation Results

Four simulation examples are shown to illustrate the effectiveness of the designed controller. The first and second examples show simplified scenarios, in which three vehicles are at the intersection. The goal of the simulations in scenario 3 is to show that the proposed control system is able to adapt to various traffic environments, depending on the initial velocities and positions of the vehicles. The fourth example has increased complexity, in which seven vehicles are incorporated. In these simulations the closest three vehicles in the constraints of the optimization tasks are incorporated.

The RL-based agent was trained through 500 simulation scenarios, in which the number of vehicles, their initial velocities and positions were selected randomly; i.e., the vehicle number was varied between 1 and 7, the initial positions were selected between -20 m ... -40 m and the initial velocities were varied between 10 km/h ... 50 km/h. In the learning process of the neural networks, the following structures have been trained. The actor network had eight neurons in the input layer; three fully connected layers with 48 neurons and Rectified Linear Unit (ReLU) functions in each layer; and one neuron with hyperbolic tangent functions in the output layer. The critical network has the same structure. In the structure and parameter selection of the neural networks, the k-fold cross-validation technique [51] and a hidden-layer number optimization process [52] can be used. The sampling time in each episode is selected at $T = 0.05$ s. The terms in the reward function are considered with the same design parameters, such as $Q_1 = -0.1$ and

$Q_2 = 10$. The training process has been performed through Matlab 2020a Reinforcement Learning Toolbox [53] on an Intel i7 CPU.

For the simplified examples, the first and second scenarios are illustrated in Figure 5. In these cases, vehicle 1 is the ego vehicle; vehicle 2 and vehicle 3 are the surrounding vehicles on the perpendicular road section. Both surrounding vehicles cross the route of vehicle 1, which results in a conflict situation. In the first scenario the following initial conditions were set: $s_1 = -40$ m, $s_2 = -36$ m, $s_3 = -41$ m and all of the vehicles have 50 km/h initial velocity. In the second scenario, vehicle 3 had different settings, while the other vehicles had the same initial conditions, i.e., $s_2 = -48$ m and $v_2 = 30$ km/h. The safe distance was set to $s_{safe} = 8$ m.

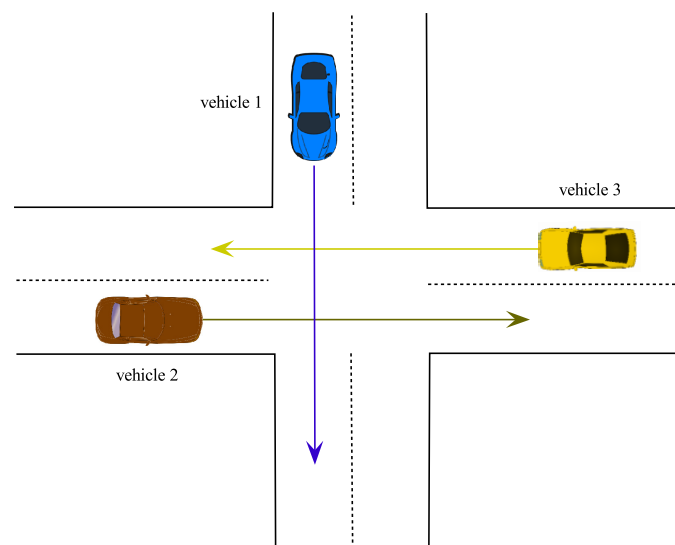
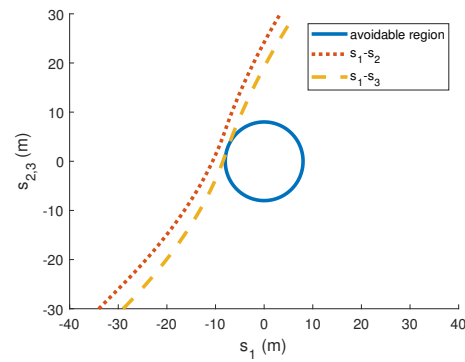
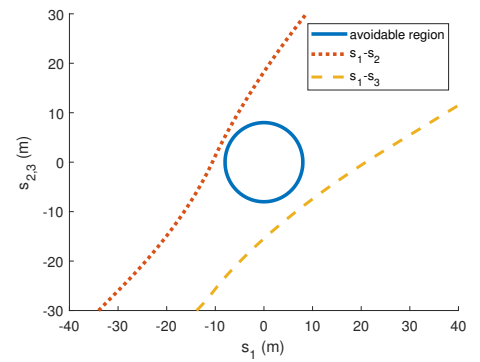


Figure 5. Illustration of the simplified intersection scenarios.

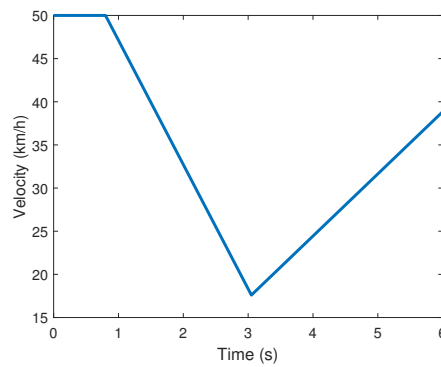
The results of the simplified scenarios are found in Figure 6. Figure 6a,b shows that the autonomous ego vehicle is able to guarantee keeping s_{safe} . Nevertheless, in the two scenarios, the ordering of the vehicles in the intersection is different. Since in scenario 1 the surrounding vehicles have increased velocity, the autonomous vehicle decides to reduce its velocity to 18 km/h and allows the surrounding vehicles with right of way to enter the intersection. In scenario 2 the autonomous vehicle has enough time to go through the intersection before vehicle 3 enters it. Thus, in this case v_1 can be increased after vehicle 2 leaves the intersection, see Figure 6d. The differences in the motion of vehicle 1 also influence the trajectories in Figure 6a,b. Moreover, the control input signals for each scenario are found in Figure 6e,f. In case of scenario 2 the acceleration of the ego vehicle is increased to 3 m/s^2 in 2.6 s to avoid a collision with vehicle 2, while in scenario 1 the acceleration is increased later than 3 s by a lower value.



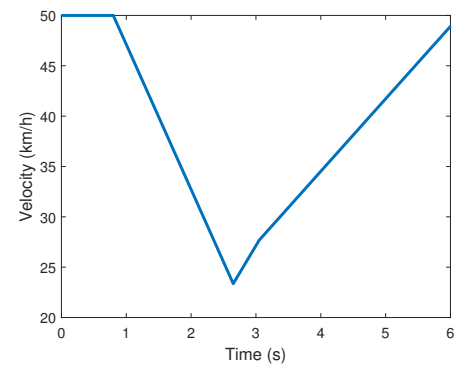
(a) Trajectories of the vehicles in scenario 1



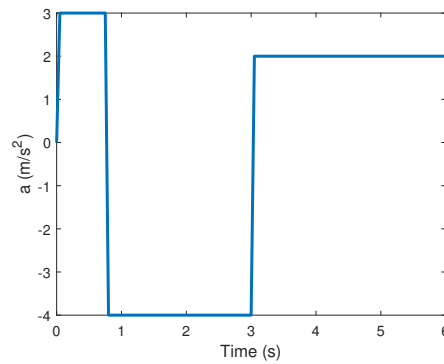
(b) Trajectories of the vehicles in scenario 2



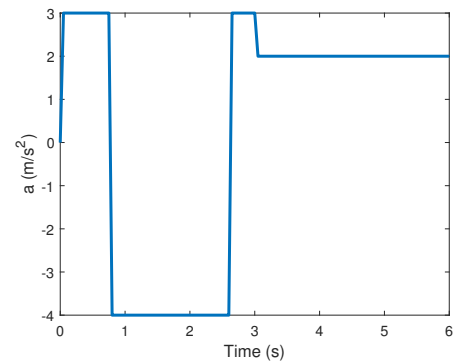
(c) Velocity in scenario 1



(d) Velocity in scenario 2



(e) Control input in scenario 1



(f) Control input in scenario 2

Figure 6. Signals of the simplified scenarios.

In the next simulation examples, it is proposed that the proposed algorithm provides safe motion for the autonomous vehicle in various scenarios. The intersection scenario is illustrated in Figure 7, where vehicle 1 represents the autonomous vehicle and the other vehicles are the surrounding vehicles. In the next four simulations the initial positions and velocity values of the vehicles are varied, and thus, a varying ordering in the intersection is yielded. In all of the simulations, vehicle 1 has conflicts with the surrounding vehicles because of their route crossing. The goal of the simulations are to show that the autonomous vehicle is able to adapt to various traffic scenarios, and the proposed optimization (13) together with the reinforcement learning is efficient. The initial position and velocity values of the scenarios are listed in Table 1.

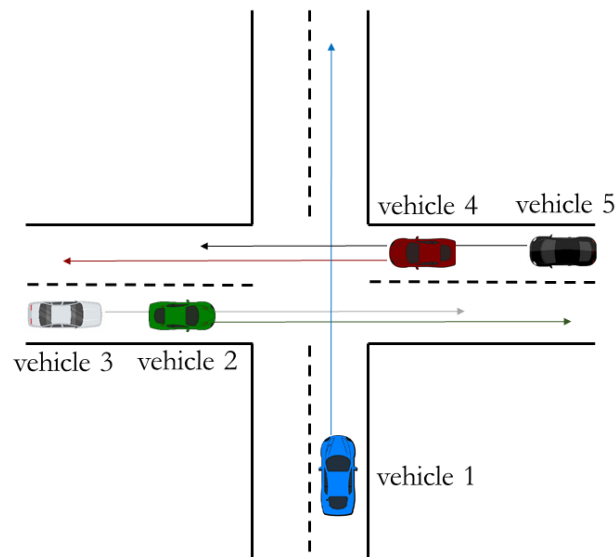


Figure 7. Illustration on the intersection scenario with 5 vehicles.

Table 1. Initial conditions for scenario 3.

Initial Condition	Scenario 3a	Scenario 3b	Scenario 3c	Scenario 3d
$s_1(0)$	−8 m	−25 m	−26 m	−30 m
$s_2(0)$	−20 m	−10 m	−20 m	−10 m
$s_3(0)$	−30 m	−30 m	−8 m	−25 m
$s_4(0)$	−25 m	−8 m	−25 m	−8 m
$s_5(0)$	−35 m	−35 m	−45 m	−30 m
$v_1(0)$	50 km/h	50 km/h	40 km/h	35 km/h
$v_2(0)$	50 km/h	50 km/h	50 km/h	50 km/h
$v_3(0)$	35 km/h	35 km/h	50 km/h	50 km/h
$v_4(0)$	50 km/h	50 km/h	50 km/h	50 km/h
$v_5(0)$	45 km/h	45 km/h	40 km/h	45 km/h

Figure 8 shows the results of each scenario in the third simulation setup, related to Figure 7. Figure 8a illustrates the trajectories of scenario 3a, where vehicle 1 could go through the intersections first. Thus, the velocity of vehicle 1 was set to the constant maximum 50 km/h, with which the avoidance of the collision was guaranteed; see Figure 8b and the related control inputs in Figure 8c. In scenario 3b the autonomous vehicle was further from the intersection at the beginning of the simulation; see Table 1. Thus, it was not possible to go through the intersection first, and therefore, vehicle 1 was the third in line; see the trajectories in Figure 8d. v_1 was slightly reduced before 2 s to guarantee the safe distance. Figure 8f shows that u was close to u_L during the entire simulation, which means that the learning process and the optimization (13) are efficient. In scenario 3c, the vehicle had reduced velocity related to the surrounding vehicles, and thus, it was fourth at the intersection (see Figure 8g). While guaranteeing safe distance, the velocity was reduced, and at 5 s the maximum velocity was achieved; see Figure 8h. Although u is close to u_L , it is necessary to modify it to avoid the collision, e.g., between 1.5 s ... 3.8 s (Figure 8i). Figure 8j shows the trajectories of scenario 3d, where vehicle 1 is the last in the ordering. Its reason is that the initial velocity is smaller than in scenario 3c; see Figure 8k. Since for vehicle 5 the priority must be guaranteed, vehicle 1 must accelerate with a reduced value between 2 and 6 s; see the velocity profile in Figure 8k and the acceleration command in Figure 8l. Nevertheless, u can be selected to be close to u_L , which can provide an improved level of economy performance, while the safety performance is simultaneously guaranteed.

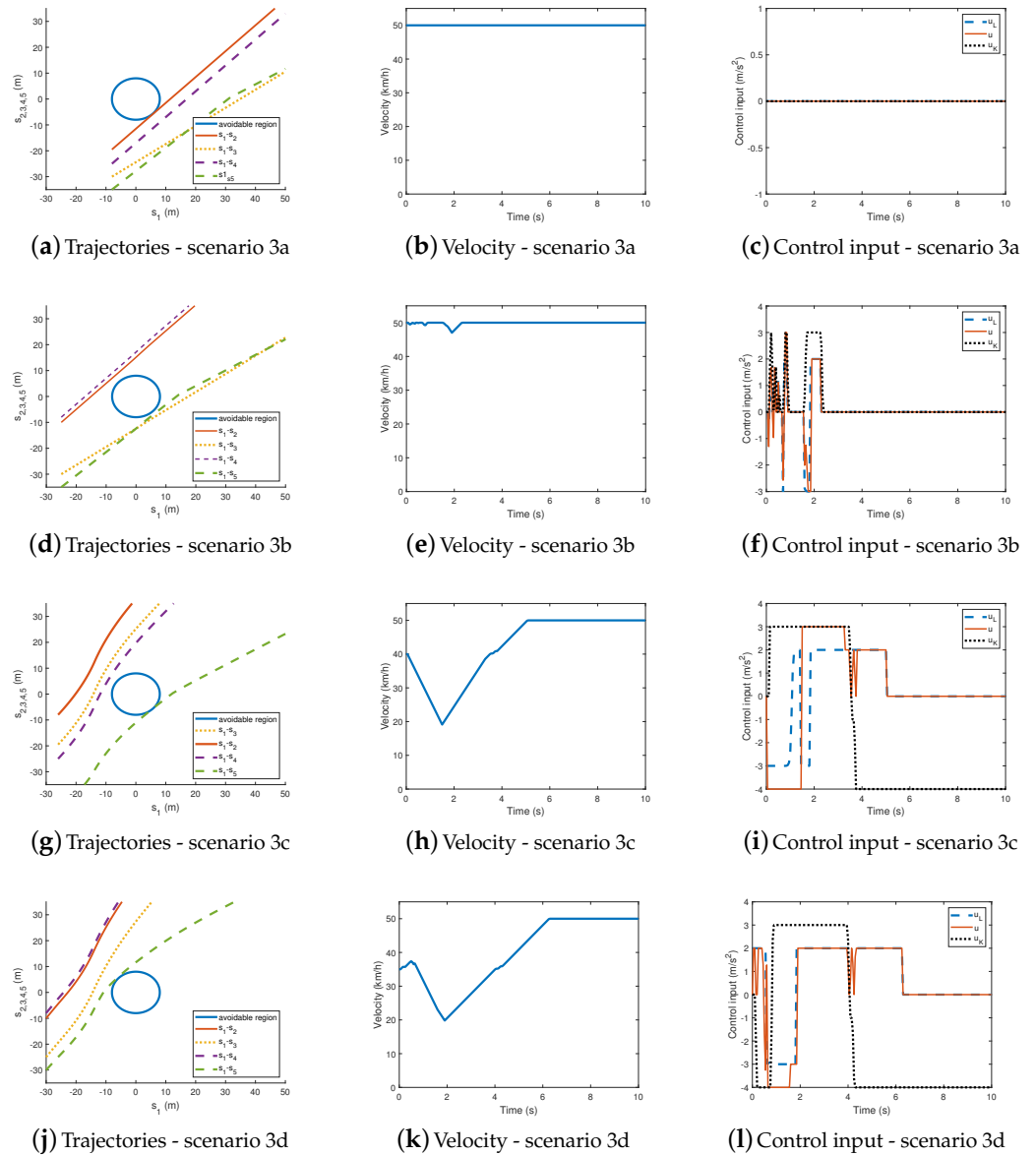


Figure 8. Signals of third scenarios.

Finally, the operation of the designed autonomous vehicle control system in a complex simulation scenario in urban context is analyzed. In the example, seven vehicles took part, i.e., the ego vehicle and six other surrounding vehicles. The illustration of the fourth scenario is found in Figure 9. In this scenario the autonomous ego vehicle is illustrated as vehicle 1. In the example, all of the other vehicles are in conflict with vehicle 1, which means that the signals of vehicles 2–7 are used during the design of the control input a . The initial positions and velocity values are as follows: $s_1 = -34$ m, $s_2 = -25$ m, $s_3 = -4$ m, $s_4 = -30$ m, $s_5 = -75$ m, $s_6 = -102$ m, $s_7 = -128$ m, $v_1 = 50$ km/h, $v_2 = 50$ km/h, $v_3 = 40$ km/h, $v_4 = 50$ km/h, $v_5 = 50$ km/h, $v_6 = 40$ km/h and $v_7 = 50$ km/h. In the scenario, $n_s = 3$ surrounding vehicles during the solution of the optimization problem are considered.

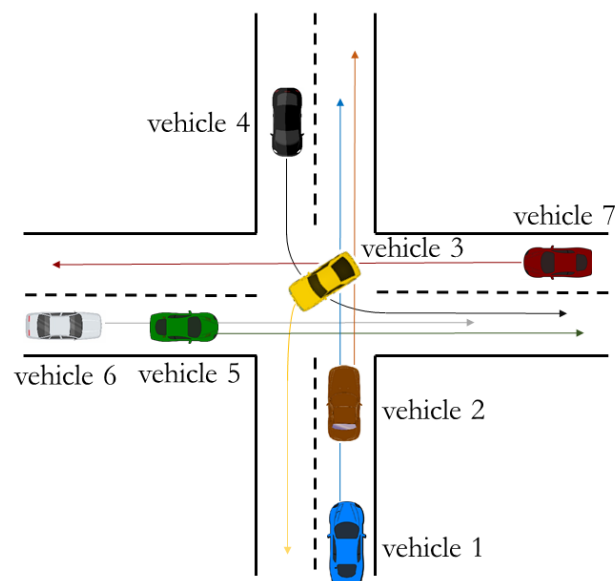


Figure 9. Illustration on the complex intersection scenario.

Some scenes of scenario 4 can be found in Figure 10. Moreover, the video of the entire scenario in <https://youtu.be/VGqcoX-9YTY> can be seen. Figure 10a shows that vehicle 2 and vehicle 3 already left the intersection and vehicle 4 was in the intersection performing a left turn. Since vehicles 2–4 were closer to the intersection than vehicle 1, the ego vehicle had to reduce its velocity to give way; see Figure 11b. It was yielded by the actuation; see the acceleration command u in Figure 11c. Moreover, after vehicle 4 left the intersection, vehicle 5 arrived at 50 km/h; see Figure 10b. Due to the reduced velocity of vehicle 1, it was not possible to go through between vehicle 4 and vehicle 5, and thus, vehicle 1 stopped. Figure 10c shows when vehicle 5 left the intersection, the velocity of vehicle 1 was increased to go through the intersection before vehicle 6 and vehicle 7 arrived (Figure 10d).

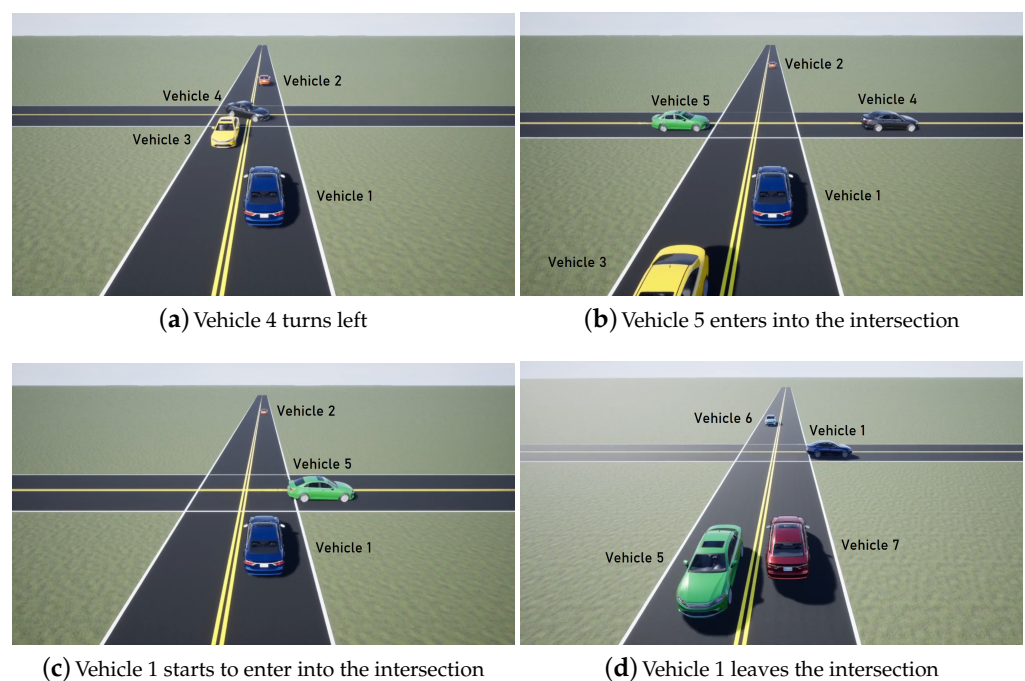
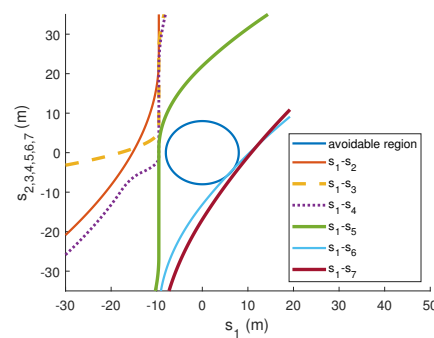
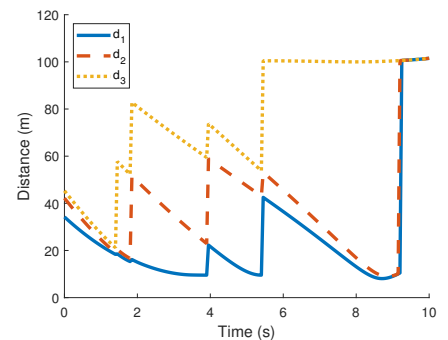


Figure 10. Illustration of the complex fourth scenario.

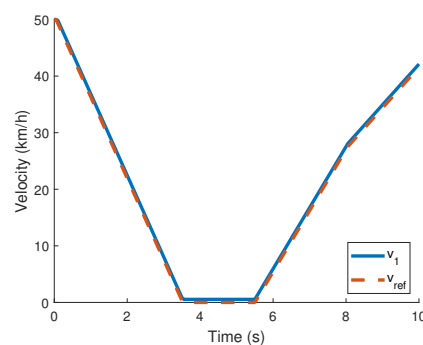
Figure 11 shows an insight into the operation of the control system. Figure 11a illustrates the trajectories of the vehicles and the avoidable region. The illustration shows that all of the trajectories are outside of the avoidable region, and thus, the safety constraint on the motion of the autonomous vehicle can be guaranteed. Figure 11b shows the distances between vehicle 1 and the closest $n_s = 3$ vehicles, where d_1 is related to the distance between vehicle 1 and the closest vehicle and d_3 is related to the distance between vehicle 1 and the third closest vehicle. Note that the vehicles in n_s vary during the simulation, e.g., directly before $t = 4$ s distance d_1 is related to s_1-s_4 , d_2 is related s_1-s_5 and d_3 is related to s_1-s_6 ; while directly after $t = 4$ s, d_1 is related to s_1-s_5 , d_2 is related s_1-s_6 and d_3 is related to s_1-s_7 . If there are less than three close vehicles, virtual vehicles with constant 100 m positions are defined; see, e.g., d_3 after 5.5 s. Figure 11b shows that all of the distances were above $s_{safe} = 8$ m during the entire simulation, independently from the selection of the closest vehicle. This was yielded by the control input u , which is illustrated in Figure 11d. Figure 11d also shows the further inputs, such as u_K and u_L . Since the role of u_K is to provide velocity tracking, its value was 3 m/s^2 constantly, because $v_1 < 50 \text{ km/h}$ during the entire simulation, except the initial velocity. The role of u_L is to provide vehicle motion, with which the reward function is maximized, so energy consumption is minimized and the traveled distance of the vehicle is maximized. However, none of the control inputs u_K and u_L are acceptable, because the safety constraints are not guaranteed. Thus, u_K is modified through Δ , which results in the control input signal u . The maximum deceleration of the vehicle before 5.5 s guaranteed the avoidance of collision with vehicles 2–5, and the maximum acceleration between 5.5 s . . . 8 s guaranteed the avoidance of the collision with vehicles 6 and 7. After 8 s, the conflict between the vehicles ceased, and thus, vehicle 1 could move with $u = u_L$ to minimize the objective in (13) to zero. As a result, the velocity of vehicle 1 was slightly increased after 8 s; see Figure 11c. Moreover, Figure 11c presents that the difference between v_{ref} and v_1 is very small, which illustrates the effectiveness of the designed robust cruise control.



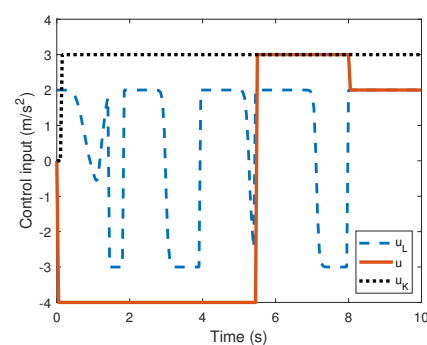
(a) Trajectories of the vehicles



(b) Distances between vehicle 1 and the closest other vehicles



(c) Velocity of the autonomous vehicle



(d) Control input of the autonomous vehicle

Figure 11. Signals of the complex fourth scenario.

The simulation examples have shown that the proposed control algorithm is able to guarantee the safe motion of the autonomous vehicle; i.e., the collisions in the intersection are avoided. The contribution of the complex scenario is that the autonomous vehicle is able to handle the presence of increased number of vehicles in the intersection.

6. Conclusions

In this paper a motion control design method for autonomous vehicles in intersection scenarios has been proposed. The effectiveness of the proposed method through various simulation examples has been presented. The scenarios show that collisions between the autonomous and the surrounding vehicles is avoided, and moreover, the autonomous vehicle can adapt to the current traffic situation. Consequently, the designed control strategy is able to guarantee safe and economical motion of the autonomous vehicle.

The challenge of the future work is to analyze the traffic scenarios in which more autonomous vehicles with the same control strategy are incorporated. It is necessary to examine how it is possible to handle the concurrent situations, e.g., with the extension of the reward function in the reinforcement learning process. It requests the training of the same agents for each vehicles, but in the learning process the interaction of the vehicles must be considered. In this way, the proposed method can be an alternative solution of the distributed control solutions in the field of intersection scenarios. Moreover, a further challenge of the method has arisen from the aspect of the analysis; i.e., the design of velocity profile for the autonomous vehicle if it must pass through several consecutive intersections. This means that the training of the RL-based agent must be carried out on an extended traffic network. In this case, further global traffic level performances in the reward function can be incorporated, e.g., the maximization of the traffic outflow on the network.

Author Contributions: Conceptualization, B.N.; methodology, B.N. and P.G.; supervision, P.G. All authors have read and agreed to the published version of the manuscript.

Funding: The research was supported by the Ministry of Innovation and Technology NRD Office within the framework of the Autonomous Systems National Laboratory Program. The research was supported by the Hungarian Government and cofinanced by the European Social Fund through the project “Talent management in autonomous vehicle control technologies” (EFOP-3.6.3-VEKOP-16-2017-00001). The work of Balázs Németh was partially supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences and the ÚNKP-20-5 New National Excellence Program of the Ministry for Innovation and Technology from the National Research, Development and Innovation Fund.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

References

1. Mihály, A.; Farkas, Z.; Gáspár, P. Multicriteria Autonomous Vehicle Control at Non-Signalized Intersections. *Appl. Sci.* **2020**, *10*, 7161. [[CrossRef](#)]
2. Mattern, N.; Wanielik, G. Camera-based vehicle localization at intersections using detailed digital maps. In Proceedings of the IEEE/ION Position, Location and Navigation Symposium, Indian Wells, CA, USA, 4–6 May 2010; pp. 1100–1107.
3. Quack, T.; Hesseler, F.; Abel, D. Infrastructure-Based Tracking of Road Users in Urban Intersections for Partially Available, Variable-Delay Sensor Data. In Proceedings of the 2018 26th Mediterranean Conference on Control and Automation (MED), Zadar, Croatia, 19–22 June 2018; pp. 1–7.
4. Wei, L.; Cappelle, C.; Ruichek, Y.; Zann, F. GPS and Stereovision-Based Visual Odometry: Application to Urban Scene Mapping and Intelligent Vehicle Localization. *Int. J. Veh. Technol.* **2011**, *2011*, 1–17. [[CrossRef](#)]
5. Li, J.; He, Q.; Zhou, H.; Guan, Y.; Dai, W. Modeling Driver Behavior near Intersections in Hidden Markov Model. *Int. J. Environ. Res. Public Health* **2016**, *13*, 1265. [[CrossRef](#)]

6. Aoude, G.; Desaraju, V.; Stephens, L.; How, J. Driver Behavior Classification at Intersections and Validation on Large Naturalistic Data Set. *IEEE Trans. Intell. Transp. Syst.* **2012**, *13*, 724–736. [[CrossRef](#)]
7. He, Z.; Zheng, L.; Lu, L.; Guan, W. Erasing Lane Changes From Roads: A Design of Future Road Intersections. *IEEE Trans. Intell. Veh.* **2018**, *3*, 173–184. [[CrossRef](#)]
8. Tettamanti, T.; Mohammadi, A.; Asadi, H.; Varga, I. A two-level urban traffic control for autonomous vehicles to improve network-wide performance. *Transp. Res. Procedia* **2017**, *27*, 913–920. [[CrossRef](#)]
9. Wuthishuwong, C.; Traechtler, A. Vehicle to infrastructure based safe trajectory planning for Autonomous Intersection Management. In Proceedings of the 2013 13th International Conference on ITS Telecommunications (ITST), Tampere, Finland, 5–7 November 2013; pp. 175–180.
10. Mohseni, F.; Frisk, E.; Nielsen, L. Distributed Cooperative MPC for Autonomous Driving in Different Traffic Scenarios. *IEEE Trans. Intell. Veh.* **2020**, *1*. [[CrossRef](#)]
11. Kloock, M.; Scheffe, P.; Marquardt, S.; Maczjewski, J.; Alrifaae, B.; Kowalewski, S. Distributed Model Predictive Intersection Control of Multiple Vehicles. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 1735–1740.
12. Kim, K.; Kumar, P.R. An MPC-Based Approach to Provable System-Wide Safety and Liveness of Autonomous Ground Traffic. *IEEE Trans. Autom. Control.* **2014**, *59*, 3341–3356. [[CrossRef](#)]
13. Riegger, L.; Carlander, M.; Lidander, N.; Murgovski, N.; Sjöberg, J. Centralized MPC for autonomous intersection crossing. In Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil, 1–4 November 2016; pp. 1372–1377.
14. Bichiou, Y.; Rakha, H.A. Real-time optimal intersection control system for automated/cooperative vehicles. *Int. J. Transp. Sci. Technol.* **2019**, *8*, 1–12. [[CrossRef](#)]
15. Hult, R.; Zanon, M.; Gros, S.; Falcone, P. Optimal Coordination of Automated Vehicles at Intersections: Theory and Experiments. *IEEE Trans. Control. Syst. Technol.* **2019**, *27*, 2510–2525. [[CrossRef](#)]
16. Németh, B.; Gáspár, P.; Szocs, D.; Mihály, A. Design of the optimal motions of autonomous vehicles in intersections through neural networks. *IFAC-PapersOnLine* **2018**, *51*, 19–24. [[CrossRef](#)]
17. Szilassy, P.; Németh, B.; Gáspár, P. Design and robustness analysis of autonomous vehicles in intersections. *IFAC-PapersOnLine* **2019**, *52*, 321–326. [[CrossRef](#)]
18. Murgovski, N.; de Campos, G.R.; Sjöberg, J. Convex modeling of conflict resolution at traffic intersections. In Proceedings of the 2015 54th IEEE Conference on Decision and Control (CDC), Osaka, Japan, 15–18 December 2015; pp. 4708–4713.
19. Fayazi, S.A.; Vahidi, A.; Luckow, A. Optimal scheduling of autonomous vehicle arrivals at intelligent intersections via MILP. In Proceedings of the 2017 American Control Conference (ACC), Seattle, WA, USA, 24–26 May 2017; pp. 4920–4925.
20. Muller, E.R.; Carlson, R.C.; Junior, W.K. Intersection control for automated vehicles with MILP. *IFAC-PapersOnLine* **2016**, *49*, 37–42. [[CrossRef](#)]
21. Németh, B.; Gáspár, P. Coordination of automated and human-driven vehicles in intersection scenarios. In Proceedings of the 2019 American Control Conference (ACC), Philadelphia, PA, USA, 10–12 July 2019; pp. 5278–5283.
22. Tian, R.; Li, N.; Kolmanovsky, I.; Yildiz, Y.; Girard, A.R. Game-Theoretic Modeling of Traffic in Unsignalized Intersection Network for Autonomous Vehicle Control Verification and Validation. *IEEE Trans. Intell. Transp. Syst.* **2020**, 1–16. [[CrossRef](#)]
23. Tachet, R.; Santi, P.; Sobolevsky, S.; Ignacio Reyes-Castro, L.; Frazzoli, E.; Helbing, D.; Ratti, C. Revisiting Street Intersections Using Slot-Based Systems. *PLoS ONE* **2016**, *11*, e0149607. [[CrossRef](#)]
24. Chai, L.; Cai, B.; ShangGuan, W.; Wang, J. Connected and autonomous vehicles coordinating method at intersection utilizing preassigned slots. In Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 16–19 October 2017; pp. 1–6.
25. Dresner, K.; Stone, P. A Multiagent Approach to Autonomous Intersection Management. *J. Artif. Intell. Res.* **2008**, *31*, 591–656. [[CrossRef](#)]
26. Hausknecht, M.; Au, T.C.; Stone, P. Autonomous Intersection Management: Multi-intersection optimization. In Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, San Francisco, CA, USA, 25–30 September 2011; pp. 4581–4586.
27. Zohdy, I.; Rakha, H. Optimizing driverless vehicles at intersections. In Proceedings of the 19th ITS World Congress, Vienna, Austria, 22–26 October 2012.
28. Katrakazas, C.; Quddus, M.; Chen, W.H.; Dekka, L. Real-time motion planning methods for autonomous on-road driving: State-of-the-art and future research directions. *Transp. Res. Part Emerg. Technol.* **2015**, *60*, 416–442. [[CrossRef](#)]
29. Moreau, J.; Melchior, P.; Victor, S.; Cassany, L.; Moze, M.; Aioun, F.; Guillemard, F. Reactive path planning in intersection for autonomous vehicle. *IFAC-PapersOnLine* **2019**, *52*, 109–114. [[CrossRef](#)]
30. Lattarulo, R.; Pérez Rastelli, J. A Hybrid Planning Approach Based on MPC and Parametric Curves for Overtaking Maneuvers. *Sensors* **2021**, *21*, 595. [[CrossRef](#)]
31. Carvalho, A.; Gao, Y.; Gray, A.; Tseng, H.E.; Borrelli, F. Predictive control of an autonomous ground vehicle using an iterative linearization approach. In Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, 6–9 October 2013; pp. 2335–2340.

32. Hu, X.; Chen, L.; Tang, B.; Cao, D.; He, H. Dynamic path planning for autonomous driving on various roads with avoidance of static and moving obstacles. *Mech. Syst. Signal Process.* **2018**, *100*, 482–500. [[CrossRef](#)]
33. Stahl, T.; Wischnewski, A.; Betz, J.; Lienkamp, M. Multilayer Graph-Based Trajectory Planning for Race Vehicles in Dynamic Scenarios. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019.
34. Werling, M.; Ziegler, J.; Kammel, S.; Thrun, S. Optimal trajectory generation for dynamic street scenarios in a Frenet Frame. In Proceedings of the 2010 IEEE International Conference on Robotics and Automation, Anchorage, Alaska, 3–8 May 2010; pp. 987–993. [[CrossRef](#)]
35. Svensson, L.; Masson, L.; Mohan, N.; Ward, E.; Brenden, A.P.; Feng, L.; Törngren, M. Safe Stop Trajectory Planning for Highly Automated Vehicles: An Optimal Control Problem Formulation. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 517–522.
36. Herrmann, T.; Wischnewski, A.; Hermansdorfer, L.; Betz, J.; Lienkamp, M. Real-Time Adaptive Velocity Optimization for Autonomous Electric Cars at the Limits of Handling. *IEEE Trans. Intell. Veh.* **2020**, *1.10.1109/TIV.2020.3047858*. [[CrossRef](#)]
37. Stellato, B.; Banjac, G.; Goulart, P.; Bemporad, A.; Boyd, S. OSQP: An operator splitting solver for quadratic programs. *Math. Program. Comput.* **2020**, *12*, 637–672. [[CrossRef](#)]
38. Lipp, T.; Boyd, S. Minimum-time speed optimisation over a fixed path. *Int. J. Control.* **2014**, *87*, 1297–1311. [[CrossRef](#)]
39. Isele, D.; Rahimi, R.; Cosgun, A.; Subramanian, K.; Fujimura, K. Navigating Occluded Intersections with Autonomous Vehicles Using Deep Reinforcement Learning. In Proceedings of the 2018 IEEE International Conference on Robotics and Automation (ICRA), Guangzhou, China, 17–19 November 2018; pp. 2034–2039.
40. Wu, T.; Jiang, M.; Zhang, L. Cooperative Multiagent Deep Deterministic Policy Gradient (CoMADDPG) for Intelligent Connected Transportation with Unsignalized Intersection. *Math. Probl. Eng.* **2020**, *2020*, 1820527. [[CrossRef](#)]
41. Chen, X.; Xu, B.; Qin, X.; Bian, Y.; Hu, M.; Sun, N. Non-Signalized Intersection Network Management With Connected and Automated Vehicles. *IEEE Access* **2020**, *8*, 122065–122077. [[CrossRef](#)]
42. Zhou, M.; Yu, Y.; Qu, X. Development of an Efficient Driving Strategy for Connected and Automated Vehicles at Signalized Intersections: A Reinforcement Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 433–443. [[CrossRef](#)]
43. Quang Tran, D.; Bae, S.H. Proximal Policy Optimization Through a Deep Reinforcement Learning Framework for Multiple Autonomous Vehicles at a Non-Signalized Intersection. *Appl. Sci.* **2020**, *10*, 5722. [[CrossRef](#)]
44. Yudin, D.A.; Skrynnik, A.; Krishtopik, A.; Belkin, I.; Panov, A.I. Object Detection with Deep Neural Networks for Reinforcement Learning in the Task of Autonomous Vehicles Path Planning at the Intersection. *Opt. Mem. Neural Netw.* **2019**, *28*, 283–295. [[CrossRef](#)]
45. Wu, Y.; Fu, T.; Ma, Y.; Li, H. Active control of ferrofluid droplet breakup dynamics in a microfluidic T-junction. *Microfluid Nanofluid* **2015**, *18*, 19–27. [[CrossRef](#)]
46. Anandan, P.; Gagliano, S.; Bucolo, M. Computational models in microfluidic bubble logic. *Microfluid Nanofluid* **2015**, *18*, 305–321. [[CrossRef](#)]
47. Huang, D.; Wang, K.; Wang, Y.; Sun, H.; Liang, X.; Meng, T. Precise control for the size of droplet in T-junction microfluidic based on iterative learning method. *J. Frankl. Inst.* **2020**, *357*, 5302–5316. [[CrossRef](#)]
48. Belotti, P.; Liberti, L.; Lodi, A.; Nannicini, G.; Tramontani, A. Disjunctive Inequalities: Applications And Extensions. Available online: <https://www.lix.polytechnique.fr/~liberti/disjineqs.pdf> (accessed on 19 February 2021).
49. Zhou, K.; Doyle, J.; Glover, K. *Robust and Optimal Control*; Prentice Hall: Englewood Cliffs, NJ, USA, 1996.
50. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv* **2015**, arXiv:1509.02971.
51. Beale, H.D.; Demuth, H.B.; Hagan, M. *Neural Network Design*; PWS Publishing: Boston, MA, USA, 1996.
52. Xu, S.; Chen, L. A novel approach for determining the optimal number of hidden layer neurons for FNN's and its application in data mining. In Proceedings of the 5th International Conference on Information Technology and Applications (ICITA 2008), Cairns, Australia, 23–26 June 2008; pp. 683–686.
53. The MathWorks, Inc. *Reinforcement Learning Toolbox User's Guide*; Mathworks Inc.: Natick, MA, USA, 2020.