Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 2220–2225

Zonotopic-tube-based LPV Motion Planner for Safety Coordination of Autonomous Vehicles

Álvaro Carrizosa-Rendón, * **Vicenç Puig** * **and Fatiha Nejjari** *

* *Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Carrer Llorens Artigas, 4-6, 08028 Barcelona, Spain (e-mail: {alvaro.carrizosa, vicenc.puig, fatiha.nejjari}@ upc.edu).*

Abstract: In this paper an optimization-based solution to the collision avoidance challenge for autonomous vehicles is proposed. The presented approach consists in an online motion planner designed to define a feasible and efficient path which implicitly guarantees safety manoeuvres in dynamic surroundings. The fact of considering moving obstacles inside the motion planner increases the complexity of the problem while forces it to be executed more frequently as others. To reduce its computational complexity, this approach proposes a two stages translation of the commonly used non-linear optimization-based structure into a QP formulation which can be easily solved. The first stage is based on the use of LPV matrices in the dynamic constraints of the vehicle. The second stage consists in computing linear expressions by set propagation to obtain the set of permitted inputs and reachable states which guarantee safety conditions.

Keywords: Autonomous vehicles; Motion planning; Robust Planner; Tubes; Safety; Coordination.

1. INTRODUCTION

Safety coordination of autonomous vehicles is a wide topic with different aspects to be solved being the collision avoidance one of the main challenges to guarantee safety. According to this, a motion planner (from now on MP) following the predominant lane of research, which is the optimization-based solutions, has been designed. This consists in formulating a mathematical expression to evaluate the desired performance while the dynamical, safety and physical, limitations of the system are considered as a set of constraints to accomplish.

As detailed in the survey (Paden et al., 2016), many disparate optimization-based approaches can be found in the literature which use model-based movement predictions to design the motion plan. Different formulations and objectives have been presented, having all in common the difficulty of skipping complex formulations to compute the optimal solution with low computational cost. Therefore, the majority of the solutions proposed are focused on presenting approaches to simplify the problem or to reduce the computational complexity to make possible the implementation.

A very extended way of dealing with high computational costs is to design motion planners which deal with the complexity of the non-linearities reducing the computational cost by delegating the obstacles avoidance to the motion controllers (from now on MC). For example, Hegedüs et al. (2017) propose a non-linear formulation to find the optimal path using a dynamical model of the vehicle, while propose to check collision avoidance with a higher-level supervisor outside the optimization problem. Moreover, the authors consider that the computational complexity is still significant and remark that the real time execution is not viable with nowadays technology.

Another common manner of reducing the computational complexity is simplifying the expressions. One example of that is the MP for racing vehicles presented in Caporale et al. (2018). This solution uses dynamical models with a formulation based on a trade-off between the curvature of the path (to avoid slipping) and track length (to reach the goal as fast as possible). To reduce the computational cost, the authors propose to study the path as a sequence of linear segments, while use Taylor expansion to simplify the expression of the curvature. These kinds of approaches permit the implementation but sacrifice the optimality. Additionally, this solution does not consider collision avoidance inside the MP, leading it to a hypothetical external module which provides the sufficient constraints to find a collision-free path.

Other approaches, such as the MP proposed at Liu et al. (2017), avoid the complexity of the problem using kinematic models. This work presents an interesting solution based on a mixed integer problem to decide the most suitable manoeuvre selecting lanes. Once the lane is selected, the potential field associated to the lane is computed to avoid getting close to obstacles. Moreover, the collision problem is addressed by approximating the vehicle and the different obstacles as polyhedra establishing a set of constraints to exclude solutions where polyhedra intersect. It is important to remark that many MC use dynamical reference, thus this methodology would be incompatible.

Even more interesting is the proposition presented in (Scheffe et al., 2022), where the authors remark the need

* This work has been co-financed by the Spanish State Research Agency (AEI) through the projects SaCoAV (ref. MINECO PID2020-114244RB-I00) and MASHED (TED2021-129927B-I00).

10.1016/j.ifacol.2023.10.1130

of reaching high update rates and propose approaching the non-linear expression by convex approximations.

Meanwhile, the number of optimization-based solutions which apply set-theory are becoming more and more frequent. Between them, different approaches using different techniques and objectives have been presented. The approach presented in Danielson et al. (2020) computes robust positive-invariant sets to define the subset of states where it is safe to generate the MC reference. The solution proposes a methodology to deal with disturbances and parametric uncertainties. Even more interesting, the solution can cope with dynamic obstacles by bounding the time where the vehicle can transit between sets.

Also with set-theory, different approaches propose solutions based on generating safety corridors or propagating the states to define regions where safety is guaranteed for every vehicle movement allowed. For example, Manzinger et al. (2020) proposes a methodology to define corridors by reachability analysis and integrate them inside the MP. To reduce the computational complexity some assumptions to achieve linear expressions for the kinematics aspects of the vehicle are done. Also Schäfer et al. (2021) propose a solution with similar objectives. The authors describe a methodology to identify collision-free driving corridors using reachability analysis. Once developed, they are approximated by polyhedra to reduce the complexity of the solution for using them as constraints of the MP. The authors validated their methodology combining it with existing optimization-based MP.

Besides that, other propositions reduce the computational cost without linearizing or simplifying the model by using LPV matrices. For example, in Alcalá et al. (2020a), the non-linearities are embedded inside a linear model with time-varying terms using an LPV state-space representation. By this way, the MP solves a problem where the dynamic constraints of the vehicle are linear. However, the MP can only deal with static obstacles by tightening the boundaries of the states to avoid collisions. Those techniques can be perfectly combined with tube-based solutions as in Alcalá et al. (2020c), where LPV matrices and tubes are combined to solve a related problem associated to the MC of an autonomous vehicle, highlighting the utility that such a combination could have in the field of motion planning.

On the basis of the aforementioned approaches, the aim of this paper is to present a novel Motion Planner for an autonomous vehicle able to generate references for a dynamic Motion Controller in dynamic surroundings. It consists in the generation of kinematic and dynamic references for the MC guaranteeing the avoidance of dynamic obstacles in real time, proposing a translation of the non-linear optimization problem into a QP problem combining LPV state-space representation for the non-linear constraints associated to the dynamics of the vehicle, and tube-propagation for linear constraints to guarantee safety conditions. By this way, the complexity of the problem is reduced which enables its implementation in real time.

2. PROBLEM STATEMENT

The starting point of this work is an autonomous vehicle with V2V communication and a level of autonomy 5. That is, the vehicle can drive autonomously with any human intervention and counts with a communication system to send and receive information from other vehicles. As commented, the objective is to design a MP able to generate the kinematic and dynamic references for the MC guaranteeing collision avoidance.

The optimization-based MP proposed for coordination are generally non-linear and are formulated with a structure similar to the one of an MPC controller. This structure is typically composed by a cost function, used to evaluate the performance of the path designed along a prediction horizon (H_p) with a determined sampling time (T_s), and a set of constraints, used to ensure feasibility by accomplishing the physical and dynamical limitations of the vehicle while guaranteeing not compromising safety conditions.

$$\min_{u(0)\ldots u(H_p-1)} J(x(k),u(k)) \tag{1}$$

subject to the following discretized constraints:

$$x(k+1) = f(x(k),u(k)) \cdot T_s + x(k) \tag{2}$$

$$g(x(k),u(k)) \leq 0 \tag{3}$$

$$x(k) \in [\underline{x}, \overline{x}] \tag{4}$$

$$u(k) \in [\underline{u}, \overline{u}] \tag{5}$$

$$\Delta u(k) \in [\underline{\Delta u}, \overline{\Delta u}] \tag{6}$$

Equation (1) is a cost function with linear and quadratic terms rewarding or penalizing the values of the states, inputs and their slew rates. In comparison with an MPC controller, the MP does not count with references; it generates them. Expressions (4), (5) and (6) delimit the upper and lower bounds of the states, inputs and slew rates according to safety and physical limitations.

The first non-linearities come from equation (2), which are the constraints associated to the dynamic aspects of the vehicle. The second source of non-linearities are the inequalities (3), which are non-linear expressions used to ensure that there is no collision between the vehicle studied and the obstacles nearby.

The fact of considering avoidance of moving obstacles inside the MP, forces it to be executed in real time and with higher frequency, as usually. Therefore, the computational time has to be reduced. Otherwise, the resulting planner would not be implementable.To reduce the computational complexity of the approach, a translation of typical non-linear solutions into a QP solution which could be solved in lower computational time has been proposed.

In a first stage, the non-linear expressions associated to the dynamic aspects of the vehicle from (2) are represented as linear expressions with LPV matrices embedding the non-linearities inside time-varying terms.

Moreover, in a second stage, the set of states, inputs and slew rates admissible is refined by a zonotopic propagation of the set of possible states along time skipping the non-linear expressions of the collision avoidance (3). The objective of the set-propagation based on a specific type of polytopes called zonotopes is to efficiently compute linear expressions of a tube of all reachable states and inputs that can be applied accomplishing the obstacle avoidance constraints which originally were non-linear.

By this way, the problem gets redefined as a QP problem constrained by expressions (9), (10) and (8) which can be solved with lower computational cost because of not containing non-linear terms.

$$\min_{u(0)\ldots u(H_p-1)} J(x(k),u(k)) \tag{7}$$

subject to the following constraints:

$$x(k+1) = A(k)x(k) + B(k)u(k) \tag{8}$$

$$x(k) \in S_x(k) \subseteq [\underline{x}, \overline{x}] \tag{9}$$

$$u(k) \in S_u(k) \subseteq [\underline{u}, \overline{u}] \tag{10}$$

In this proposal the vehicle has been modeled with a dynamical bicycle model with the following states: linear, lateral and angular velocities ($v_x(t)$, $v_y(t)$ and $\omega(t)$), distance to the center of the road ($e_y(t)$), difference between the orientation of the vehicle and the curvature of the road ($\theta_e(t)$), and the distance traveled ($s(t)$) measured projecting the position of the vehicle over the center of the road. The mathematical expressions of the different states are defined in equations (11-16), while a more detailed description of the expressions and the reasoning behind them can be consulted in Alcalá et al. (2020b).

$$\begin{aligned}\dot{v}_x(t) = a(t) - C_f \frac{\delta(t) sin(\delta(t))}{m} + \omega(t) v_y(t) \\ -\mu v_x(t) + C_f sin(\delta(t)) \frac{\omega(t) l_f + v_y(t)}{m v_x(t)}\end{aligned} \tag{11}$$

$$\begin{aligned}\dot{v}_y(t) = -\omega(t) \frac{C_f l_f cos(\delta(t)) - C_r l_r}{v_x(t) m} - v_x(t)\omega(t) \\ + C_f \frac{\delta(t) cos(\delta(t))}{m} - v_y(t) \frac{C_r + C_f cos(\delta(t))}{v_x(t) m}\end{aligned} \tag{12}$$

$$\begin{aligned}\dot{\omega}(t) = \frac{C_f \delta(t) cos(\delta(t)) l_f}{I} - v_y(t) \frac{C_f cos(\delta(t)) l_f}{v_x(t) I} \\ + v_y(t) \frac{l_r C_r}{v_x(t) I} - \omega(t) \frac{C_f cos(\delta(t)) {l_f}^2 + C_r {l_r}^2}{v_x(t) I}\end{aligned} \tag{13}$$

$$\dot{e}_y = v_x(t) sin(\theta_e(t)) + v_y(t) cos(\theta_e(t)) \tag{14}$$

$$\dot{\theta}_e(t) = \omega(t) + \kappa \frac{v_y(t) sin(\theta_e(t)) - v_x(t) cos(\theta_e(t))}{1 - e_y(t)\kappa} \tag{15}$$

$$\dot{s}(t) = \frac{v_x(t) cos(\theta_e(t)) - v_y(t) sin(\theta_e(t)))}{1 - e_y(t)\kappa} \tag{16}$$

As it can be noticed in this preliminary study, some aspects have been let for further studies such as the inclusion of the uncertainties of the model, the uncertainties of the sensors or issues related with the communication.

3. METHODOLOGY

3.1 LPV approach

The benefits of using LPV matrices is the achievement of linear expressions to describe the non-linear reality of the vehicle model defined by equations (11-16). This LPV representation reduces the complexity of the problem without introducing the uncertainties associated to inaccurate simplifications of the model or linearizations around operational points.

The key aspect of this procedure consists in embedding the non-linearities of the system inside time-varying parameters precomputed at each iteration. Those parameters are introduced inside the linear expressions as varying terms of matrices A and B obtaining a set of linear constraints for the MP.

As the resulting state-space representation is not fully controllable for all state values, some adaptations have been performed to obtain a fully controllable system. On one hand, it has been considered that the MP will not be used with linear velocities around zero. Additionally, the orientation difference between the vehicle and the curvature of the road ($\theta_e(k)$) can be approximated as in equations (17) and (18) adding new terms A_{ij} solving controllability problems introducing lower errors as other simplifications and approaches.

$$sin(\theta_e(t)) \simeq \frac{sin(\theta_e(t))}{2} + \frac{\theta_e(t)}{2} \tag{17}$$

$$cos\,(\theta_e(t)) \simeq \frac{cos(\theta_e(t))}{2} + \frac{1}{2} \tag{18}$$

Applying both expressions and discretizing via the Forward Euler Method, the LPV state-space representation obtained is the one described at the constraint (8) with the LPV matrices (19) and (20).

$$B(k) = \begin{bmatrix} T_s & -C_f \frac{sin(\delta(k))}{m} T_s \\ 0 & -C_f \frac{cos(\delta(k))}{m} T_s \\ 0 & C_f \frac{l_f cos(\delta(k))}{I} T_s \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{19}$$

$$A(k) = \begin{bmatrix} 1 & A_{12}(k) & A_{13}(k) & 0 & 0 & 0 \\ A_{21}(k) & A_{22}(k) & A_{23}(k) & 0 & 0 & 0 \\ 0 & A_{32}(k) & A_{33}(k) & 0 & 0 & 0 \\ A_{41}(k) & A_{42}(k) & 0 & 1 & A_{45}(k) & 0 \\ A_{51}(k) & A_{52}(k) & 1 & 0 & A_{55}(k) & 0 \\ A_{61}(k) & A_{62}(k) & 0 & 0 & A_{65}(k) & 1 \end{bmatrix} \tag{20}$$

where the different time-varying terms $A_{ij}(k)$ are defined as:

$$A_{12}(k) = C_f \frac{sin(\delta(k))}{v_x(k) m} T_s \tag{21}$$

$$A_{13}(k) = v_y(k) T_s + C_f \frac{l_f sin(\delta(k))}{v_x(k) m} T_s \tag{22}$$

$$A_{21}(k) = -\omega(k) T_s \tag{23}$$

$$A_{22}(k) = -\frac{C_r + C_f cos(\delta(k))}{v_x(k) m} T_s + 1 \tag{24}$$

$$A_{23}(k) = \frac{C_r l_r - C_f l_f cos(\delta(k))}{v_x(k) m} T_s \tag{25}$$

$$A_{32}(k) = -\frac{C_f l_f cos(\delta(k)) + C_r l_r}{I v_x(k)} T_s \tag{26}$$

$$A_{33}(k) = -\frac{C_f {l_f}^2 cos(\delta(k)) + C_r {l_r}^2}{I v_x(k)} T_s + 1 \tag{27}$$

$$A_{41}(k) = \frac{sin(\theta_e(k))}{2} T_s \tag{28}$$

$$A_{42}(k) = cos(\theta_e(k)) T_s \tag{29}$$

$$A_{45}(k) = \frac{v_x(k)}{2} T_s \tag{30}$$

$$A_{51}(k) = -\kappa \frac{cos(\theta_e(k))}{1 - e_y(k)\kappa} T_s \tag{31}$$

$$A_{52}(k) = \frac{\kappa sin(\theta_e(k))}{2 - 2e_y(k)\kappa} T_s \tag{32}$$

$$A_{55}(k) = \frac{\kappa v_y(k)}{2 - 2e_y(k)\kappa} T_s + 1 \tag{33}$$

$$A_{61}(k) = \frac{cos(\theta_e(k))}{1 - e_y(k)\kappa} T_s \tag{34}$$

$$A_{62}(k) = -\frac{sin(\theta_e(k))}{2 - 2e_y(k)\kappa} T_s \quad (35)$$

$$A_{65}(k) = -\frac{v_y(k)}{2 - 2e_y(k)\kappa} T_s \quad (36)$$

As it can be observed, values of the sixth state of the system (s), do not affect to the dynamics of the other states. In fact, this state could be computed outside the model at each iteration simplifying it, but its integration in the model is useful for the prediction step and to easily weight the distance travelled inside the cost function.

Selection of the LPV matrices: A critical aspect of the methodology proposed is the selection of the LPV matrices to obtain an accurate representation of the vehicle. In this proposal, it has been assumed that the surroundings will change progressively and no big changes happening suddenly are expected. By this way, the proposed path computed at each execution is expected to be similar to the path designed in the previous iteration adding a new further step. Consequently, the LPV matrices are computed based on the estimated location of the vehicle at each time instant extracted from the previous designed path. This procedure can be graphically seen in Figure 1: The vehicle is represented as a big square while the green curve represents the future locations of the vehicle in case it follows the last path designed. The LPV matrices are computed according to the estimated values of the states at each time instant.

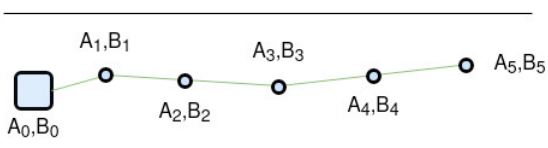

Fig. 1. Selection of the LPV matrices for the LPV-MP.

As it can be expected, this assumption may induce some errors in the model. Therefore, the MP should be combined with a closed-loop MC to minimize the error when tracking the path. Additionally, the MP has to be executed frequently in order to reduce the difference between expectations and reality, reacting rapidly to unexpected changes in the surroundings. In the same way, the time length of the path $(H_p \cdot T_s)$ does not have to be excessively large as in typical off-line MP, because the uncertainty of the model is higher the further we plan, while the dynamic surroundings do not ensure the designed path would be optimal or even feasible in future iterations. Also including uncertainties inside the model would be interesting to guarantee robustness.

3.2 Computation of safety Constraints

The second step to skip the non-linearities of the non-linear optimization problem is based on merging the constraints associated to the inputs, slew rates and states boundaries with the collision avoidance restrictions by means of set-theory obtaining new linear expressions constraining the set of admissible values. In this proposal, the sets are defined as zonotopes because of its basic definition and its capability of being propagated with low computational complexity.

Firstly,the set of all inputs that are candidates to be applicable is computed. As shown in (37), it is computed amplifying the set of inputs applied (or applicable) at previous time instant and the maximal slew rate allowed $(S_{\Delta u})$. Then, all values outside the boundaries are excluded. This step is performed intersecting the obtained set with a set containing the upper and lower bounds. By this way, all inputs that are not applicable due to the slew rate or constraints are directly excluded from the study.
Secondly, the set of possible states is propagated using the computed set of inputs and the precomputed LPV matrices, which are those used for the first stage based on the expected location of the vehicle at each time instant. Once the set of all reachable states is computed, it is intersected with the set of feasible solutions as formulated in (38). This intersection is performed with the set of all valid states excluding those which do not guarantee safety conditions for this time instant $(S_x^C(k))$.

$$\hat{S}_u(k) = (S_u(k-1) + S_{\Delta u}) \cap S_u^C \quad (37)$$

$$S_x(k+1) = \Big(A(k)S_u(k) + B(k)\hat{S}_u(k)\Big) \cap S_x^C(k) \quad (38)$$

Once the set of reachable states which guarantees the fulfillment of safety conditions has been computed, a back-propagation stage using the pseudo-inverse matrix of the LPV matrix B and the Minkowski difference of sets is performed to obtain the subset of the inputs that provides valid states $(S_x(k+1))$. In other words, (39) refines the subset of applicable inputs excluding those values which lead the vehicle into unsafe regions. By this way, the resulting subsets of inputs and states for a determined time instant k is contained in the initial set of applicable inputs and fulfill requirements (40).

$$S_u(k) = B^+(k)\,(S_x(k+1) \sim A(k)S_x(k)) \quad (39)$$

$$S_u(k) \subseteq \hat{S}_u(k) \qquad S_x(k) \subseteq S_x^C(k) \quad (40)$$

Up from this point, the constraints associated to the slew rate (6), boundaries of inputs (5), boundaries of states (4) and the verification of collision avoidance (3) can be substituted by limiting the search of solutions inside the tube containing the reachable safety states (9) and applicable inputs (10) skipping previous existing non-linearities.
The proposition for avoiding dynamic obstacles is to generate tubes not only for the vehicle studied, but also for the moving obstacles considering the uncertainties of the estimation of their motions, excluding from the safety subsets those values that intersect with the tubes generated for each obstacle nearby.

4. SIMULATIONS

For validating the methodology proposed, the explained procedure has been implemented in MATLAB and evaluated through simulations in different scenarios. For the implementation the toolbox for rapid prototyping optimization problems YALMIP, Löfberg (2004), and a toolbox called CORA, Althoff (2015), useful for defining and dealing with set operations have been used. The solvers used for the non-linear MP have been "fmincon" while the MP with LPV matrices have been solved with "quadprog".
The simulations have been performed in different closed maps, being the results presented those of the simulations performed in the map presented in Figure 2. The vehicle

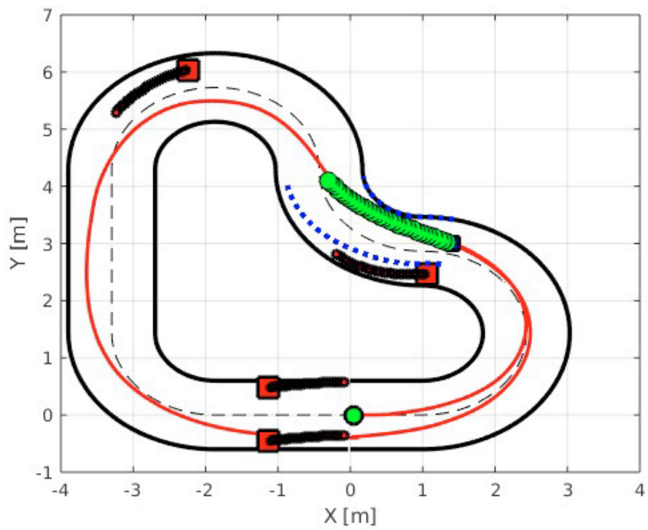

Fig. 2. Simulation of an LPV-MP with a H_p of 40 samples. Current position and predicted path (green square and dots), safety margins (blue curves), path already followed (red curve), obstacles position and future estimations (red squares and dots).

has been defined with the parameters listed in Table 1, corresponding to a small car-like robot. An open-loop controller has been implemented applying the first input proposed by the MP while the vehicle has been simulated with its ODEs.

Some dynamic obstacles have been added: four vehicles driving at constant linear velocity varying its distance to the center of the road by a sinusoidal function, one located at the left side, another at the right side and two driving in parallel. For these simulations the current position and future movements of the obstacles have been considered accurately known, reducing their tubes to specific positions. Therefore, no tube computation for the obstacles is needed using directly the specific estimated location of the obstacles for a tightening of the set of valid states ($S_x^C(k)$) at each time instant in order to ensure safety conditions.

Table 1. Parameters used for defining the car-like robot.

Parameters	**Value**	**Parameters**	**Value**
l_r	0,125 m	l_f	0,125 m
C_r	65 N/rad	C_f	65 N/rad
μ	0,05	I	0,03 kg/m^2
m	1,98 kg		

Comparison between a NL-MP and an LPV-MP: In a first study, a non-linear MP (from now on NL-MP) and a linear MP thanks to LPV matrices (from now on LPV-MP) has been compared. For the comparison the same initial conditions and scenario have been used.

As already mentioned, for both cases the future location of the surrounding obstacles were supposed to be known being represented with red squares and dots in Figure 2. Then, the boundaries of the lateral distance to the center of the road has been already redefined before executing each iteration thanks to the estimated locations of the vehicle and obstacles at each time instant based in the path designed in the previous iteration. This can be observed with the blue curve in the representation of the map.

As can be observed comparing the results in Table 2, translating the non-linear dynamical constraints into a set of linear expressions significantly reduces the computational cost of finding the optimal solution. In fact, the computational time required for computing a solution using the NL-MP is higher than the duration of the designed path. In case of LPV-MP, this proportion varies depending on the sampling time (T_s) and the selected prediction horizon (H_p), but it is possible to obtain better results which are closer to a real-time implementation, verifying the high benefits of applying LPV matrices inside the MP to avoid those non-linearities.

Table 2. Comparison of NL-MP and LPV-MP during a 10 seconds simulation.

MP	H_p	T_s [s]	$H_p \cdot T_s$ [s]	$T_{com}[s]$	$\Delta s[m]$
NL	10	0,03	0,3	1,7368	22,3457
NL	30	0,03	0,9	14,7916	22,8840
LPV	10	0,03	0,3	0,0242	18,8204
LPV	30	0,03	0,9	0,0792	20,4379

The mean squared error along the 10 seconds of simulation between the distance travelled by the vehicle using the LPV-MP compared with the NL-MP is 5,5105 m^2 for a prediction horizon of 10 samples and 2,4904 m^2 for 30 samples, but the mean time required for computing a path at each iteration gets reduced from 1,7368 seconds to 0,0242 and from 14,7916 seconds to 0,0792, respectively.

Comparison between an LPV-MP and a ZTB LPV-MP: A comparison between an LPV-MP without the stage of states propagation (the LPV-MP of the previous comparison) and another with the tube computation based in zonotopes (from now on ZTB-MP) has also been performed to proof that the computational cost of increasing the tube generation does not increase significantly if the computational cost is compared with the results of the initial NL-MP.

Table 3. Results of different MP with (ZTB) and without (LPV) safety constraints computation during 10 seconds simulation.

MP	H_p	T_s [s]	$H_p \cdot T_s$ [s]	$T_{com}[s]$	$\Delta s[m]$
LPV	10	0,03	0,3	0,0242	18,8204
LPV	15	0,03	0,45	0,0315	19,3215
LPV	20	0,03	0,6	0,0401	19,6298
LPV	30	0,03	0,9	0,0792	20,4379
LPV	35	0,03	1,05	0,0943	20,6456
ZTB	10	0,03	0,3	0,0362	18,3190
ZTB	15	0,03	0,45	0,0543	19,2598
ZTB	20	0,03	0,6	0,0726	19,5009
ZTB	30	0,03	0,9	0,1313	20,5650
ZTB	35	0,03	1,05	0,1584	20,8241

The results of the new simulations are shown in Table 3. The simulation scenario and initial conditions are the same as in previous simulations. As can be observed in Figures 3 and 4, where both MP have been graphically compared, the inclusion of the tube computation does not infer negatively to the distance travelled obtaining similar results, while the computational cost increases linearly but remains in the same order of magnitude, which is lower than the one of the NL-MP.

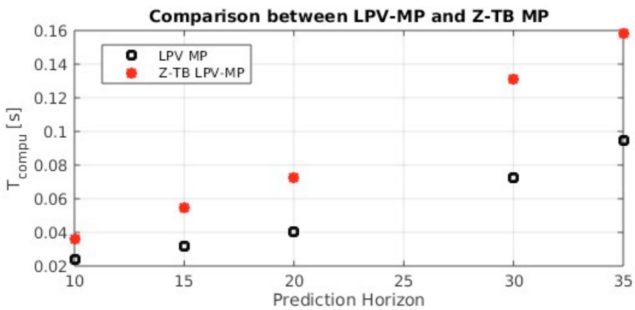

Fig. 3. Comparison between LPV-MP with (LPV-MP) and without zonotopic propagation (ZTB-MP): Computational cost.

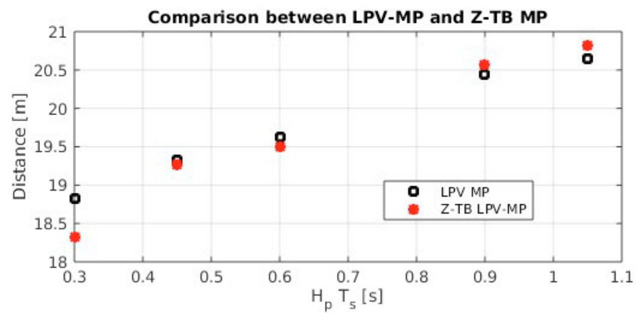

Fig. 4. Comparison between LPV-MP with (LPV-MP) and without zonotopic propagation (ZTB-MP): Distance travelled.

5. CONCLUSIONS

An optimization-based solution to the collision avoidance problem for autonomous vehicles has been proposed and presented in this work. The proposed approach consists in designing an online motion planner to define a feasible and efficient path which implicitly guarantees safety manoeuvres in dynamic surroundings.

Firstly, it is widely known that non-linear MP needs to be simplified or adapted to be implementable with the technology of nowadays in case it is desired to be used online. The proposed translation to a QP problem seems to be a good alternative for this aim. In this case, a translation of the dynamical constraint into linear expressions thanks to the LPV matrices has been performed reducing significantly the computational cost. Secondly, it has been designed and proved that the zonotope-tube-based states propagation for defining safety regions does not compromise the computational cost while increases the safety of the solutions and opens the MP up to include dynamic obstacles avoidance. For future work, an interesting field to study and analyse more in-depth is the introduction of uncertainty in the problem formulation.
It is also interesting to study the possibility of changing the LPV matrices by a set of LPV matrices for operating the states propagation to increase the robustness of the safety region provided by this stage.

REFERENCES

Alcalá, E., Puig, V., and Quevedo, J. (2020a). Lpv-mp planning for autonomous racing vehicles considering obstacles. *Robotics and Autonomous Systems*, 124, 103392.

Alcalá, E., Puig, V., Quevedo, J., and Rosolia, U. (2020b). Autonomous racing using linear parameter varying-model predictive control (lpv-mpc). *Control Engineering Practice*, 95, 104270.

Alcalá, E., Puig, V., Quevedo, J., and Sename, O. (2020c). Fast zonotope-tube-based lpv-mpc for autonomous vehicles. *IET Control Theory & Applications*, 14(20), 3676–3685.

Althoff, M. (2015). An introduction to cora 2015. In *Proc. of the workshop on applied verification for continuous and hybrid systems*, 120–151.

Caporale, D., Fagiolini, A., Pallottino, L., Settimi, A., Biondo, A., Amerotti, F., Massa, F., De Caro, S., Corti, A., and Venturini, L. (2018). A planning and control system for self-driving racing vehicles. In *2018 IEEE 4th international forum on research and technology for society and industry (RTSI)*, 1–6. IEEE.

Danielson, C., Berntorp, K., Weiss, A., and Di Cairano, S. (2020). Robust motion planning for uncertain systems with disturbances using the invariant-set motion planner. *IEEE Transactions on Automatic Control*, 65(10), 4456–4463.

Hegedüs, F., Bécsi, T., Aradi, S., and Gáspár, P. (2017). Model based trajectory planning for highly automated road vehicles. *IFAC-PapersOnLine*, 50(1), 6958–6964.

Liu, C., Lee, S., Varnhagen, S., and Tseng, H.E. (2017). Path planning for autonomous vehicles using model predictive control. In *2017 IEEE Intelligent Vehicles Symposium (IV)*, 174–179. IEEE.

Löfberg, J. (2004). Yalmip : A toolbox for modeling and optimization in matlab. In *In Proceedings of the CACSD Conference*. Taipei, Taiwan.

Manzinger, S., Pek, C., and Althoff, M. (2020). Using reachable sets for trajectory planning of automated vehicles. *IEEE Transactions on Intelligent Vehicles*, 6(2), 232–248.

Paden, B., Čáp, M., Yong, S.Z., Yershov, D., and Frazzoli, E. (2016). A survey of motion planning and control techniques for self-driving urban vehicles. *IEEE Transactions on intelligent vehicles*, 1(1), 33–55.

Schäfer, L., Manzinger, S., and Althoff, M. (2021). Computation of solution spaces for optimization-based trajectory planning. *IEEE Transactions on Intelligent Vehicles*.

Scheffe, P., Henneken, T.M., Kloock, M., and Alrifaee, B. (2022). Sequential convex programming methods for real-time optimal trajectory planning in autonomous vehicle racing. *IEEE Transactions on Intelligent Vehicles*.