2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

Time-optimal model predictive control using feasibility governors

Alvaro Florez[1], Alejandro Astudillo[1], Wilm Decré[1] and Jan Swevers[1]

Abstract— **This paper presents a control strategy to perform time-optimal point-to-point motions using the feasibility governor (FG) strategy. This technique consists of two stages. The first stage is a feasibility governor in which an auxiliary reference is calculated inside a feasible set that drives the system toward the reference along the constraint boundaries. The second stage is a time-optimal model predictive control (MPC) formulation with an appropriate structure which is executed in a set where the reference can be reached within the prediction horizon. Feasible and reachable sets that contain system dynamics and constraints information are computed offline, allowing the methodology to run with short prediction horizons. This reduces the computational load in the online execution. An example of a car represented by a bicycle model performing a lateral movement at constant forward velocity is presented to illustrate the controller's performance.**

I. Introduction

Model predictive control (MPC) is a methodology that sets and solves an optimal control problem (OCP) at each sampling time to provide an optimized system input. This process is also known as receding horizon control. A key feature of MPC is that it explicitly handles constraints. This feature plays a major role in, e.g., handling input saturation limits, or ensuring safety regions for states.

Time-optimal types of problems in MPC are of interest in many cases, e.g., following trajectories in minimum time, rejection of disturbances in dynamic environments, and point-to-point motion to optimize production schedules. Formulations of time-optimal point to point motions with MPC are not straightforward to implement for real time execution because, usually, the required time to find a solution is larger than the sampling time. This is because of the complexity of the optimization problems that should be solved. Some works have addressed this problem by (i) optimizing the solver algorithms [1], (ii) setting a bilevel optimization with a varying horizon length [2], (iii) using variable sampling step values which are included as optimization variables that are reduced along time [3]. One of the limitations that remains is the terminal constraint. It forces that, from a given initial state, the MPC within the prediction horizon should reach the terminal state, it is also denominated target position, and for simplicity reference in this work. To accomplish that, the prediction horizon should be long in case the distance to the reference is large, increasing the number of optimization variables and therefore, increasing the optimization problem complexity. To solve these time-optimal problems with a long horizon in a reasonable time for real time execution, we need an additional procedure.

Governors, like the reference governor (RG) and command governor (CG) [4] are online solution strategies for point-to-point motion problems that can handle long distances without resulting in excessive online computations at the cost of more extensive offline pre-computation. These strategies calculate a sequence of auxiliary references that replace the original reference or destination point as inputs to the closed-loop controlled system. The calculated sequence is guaranteed to be feasible under the assumed feedback control configuration, that is, guarantee that the auxiliary references can be reached without violating specified constraints. Different possible formulations are described in [4].

The works [5]–[7] present and extended governor version, called the feasibility governor (FG) which assumes and accounts for a tracking MPC strategy with quadratic objective as the close-loop controller. For the FG the set where the initial state and reference should belong is larger, which means that the distance between the point for which feasibility and convergence properties are guaranteed can be larger than for the RG and CG. The FG can be formulated such that at least during part of the motion the system moves time-optimally to the reference. Once the system is close to the reference such that it can be reached by the applied MPC strategy within its prediction horizon, time-optimality is lost because of the chosen quadratic objective.

Therefore, this work combines the time-optimal FG with time-optimal MPC. The strategy consists of two stages. The first stage exploits the set properties of the FG to move toward the reference as fast as possible (hitting the system constraints). Once the reference can be reached within its prediction horizon the strategy switches to the second stage. It is a time-optimal formulation suitable for an MPC fashion strategy. The switching conditions are precomputed, reducing the computational burden for online execution.

An outline of the FG strategy is presented in Section II and the time-optimal MPC using governor formulation is described in Section III. In Section IV, an example illustrates the main strategy characteristics and compares it with the related methodologies. Conclusions and future ideas close the paper in Section V.

Notation: The identity matrix is denoted as $I_N \in \mathbb{R}^{N\times N}$. Given $M \in \mathbb{R}^{m\times n}$, $\mathrm{Ker}(M) = \{\boldsymbol{x} \mid M\boldsymbol{x} = 0\}$. For a given vector $\boldsymbol{x} \in \mathbb{R}^n$ and a positive definite matrix $P \succ 0 \in \mathbb{R}^{n\times n}$, the weighted norm of $\boldsymbol{x}$ is $\|\boldsymbol{x}\|_P = \sqrt{\boldsymbol{x}^\top P\boldsymbol{x}}$. For a vector

[1]MECO Research Team, Dept. of Mechanical Engineering, KU Leuven and Flanders Make@KU Leuven, 3001 Leuven, Belgium. {alvaro.florez, alejandro.astudillovigoya, wilm.decre, jan.swevers}@kuleuven.be

This work was supported by the project Flanders Make SBO DIRAC: "Deterministic and Inexpensive Realizations of Advanced Control". Flanders Make is the Flemish strategic research centre for the manufacturing industry.

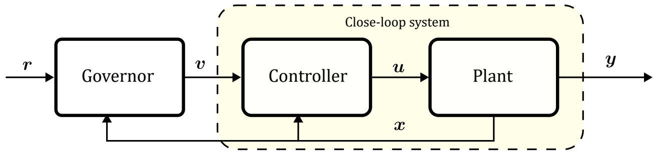

Fig. 1. Diagram of the control structure. The governor uses the reference r and considering the states and input constraints, it provides an auxiliary reference v.

$\boldsymbol{x} \in \mathbb{R}^n$ and a set $\Omega \subseteq \mathbb{R}^{n+m}$, the projection of Ω onto the domain of $\boldsymbol{x}$ is $\Pi_{\boldsymbol{x}}\Omega$ where $\Pi_{\boldsymbol{x}} = [I_n \quad \mathbf{0}_{n\times m}]$.

II. Feasibility governor

The feasibility governor (FG) for linear MPC proposed in [5], [6] enlarges the nominal MPC region of attraction. This method is an add-on unit that manipulates the reference – similar to the command governor –, while ensuring feasibility of the MPC problem in the feedback loop. It is composed of two main parts: (i) the primary controller, which is an MPC formulation, and (ii) the governor that takes the reference and provides an auxiliary reference to the system (see the general FG scheme in Fig. 1). The remainder of this section presents a brief explanation of this method.

A. System description

Let us consider a linear time-invariant (LTI) system

$$\begin{aligned} \boldsymbol{x}_{k+1} &= A\boldsymbol{x}_k + B\boldsymbol{u}_k, \\ \boldsymbol{y}_k &= C\boldsymbol{x}_k + D\boldsymbol{u}_k, \\ \boldsymbol{z}_k &= E\boldsymbol{x}_k + F\boldsymbol{u}_k, \end{aligned} \tag{1}$$

where $k \in \mathbb{N}$ is the discrete-time index, $\boldsymbol{x}_k \in \mathbb{R}^{n_x}$ are the states, $\boldsymbol{u}_k \in \mathbb{R}^{n_u}$ the control inputs, $\boldsymbol{y}_k \in \mathcal{Y} \subseteq \mathbb{R}^{n_y}$ the constrained outputs, where $\mathcal{Y}$ is a compact polyhedron, and $\boldsymbol{z}_k \in \mathbb{R}^{n_z}$ the tracking outputs. Note that the polyhedron $\mathcal{Y}$ contains the state and input constraints or a combination of them. We should consider assumptions (1-2) from [6]: (i) The pair (A, B) is stabilizable; (ii) the set $\mathcal{Y}$ is a compact polyhedron that can be described as $\mathcal{Y} = \{y | Yy \leq h\}$ and satisfies $0 \in \text{Int } \mathcal{Y}$. In case of constant references $\boldsymbol{v} \in \mathbb{R}^{n_z}$, the steady-state solution for the states, inputs, and tracking outputs can be parameterized as solutions of $Z[\boldsymbol{x}^\top \boldsymbol{u}^\top \boldsymbol{z}^\top]^\top = 0$ as [8]

$$\begin{gathered} \bar{\boldsymbol{x}}_v = G_x\boldsymbol{v}, \quad \bar{\boldsymbol{u}}_v = G_u\boldsymbol{v}, \quad \bar{\boldsymbol{z}}_v = G_z\boldsymbol{v}, \\ Z = \begin{bmatrix} A - I_{n_x} & B & 0 \\ E & F & -I_{n_z} \end{bmatrix}, \end{gathered}$$

where $G^\top = [G_x^\top \; G_u^\top \; G_z^\top]$ is a basis for $\text{Ker}(Z)$, and G_z is full rank. Thus, for given reference values $\boldsymbol{v}$ the corresponding steady-state system variables solution $[\bar{\boldsymbol{x}}_v, \bar{\boldsymbol{u}}_v, \bar{\boldsymbol{z}}_v]$ can be calculated directly.

B. The maximal output admissible set

This is the set of all states $\boldsymbol{x}$ and auxiliary references $\boldsymbol{v}$ such that the constrained outputs of the predicted closed-loop response from state $\boldsymbol{x}$ with constant auxiliary reference $\boldsymbol{v}$ remain inside the set. It is defined as follows:

$$O_\infty = \{(\boldsymbol{x}, \boldsymbol{v}) : \hat{\boldsymbol{y}}(k \mid \boldsymbol{x}, \boldsymbol{v}) \in \mathcal{Y}, \quad \forall \quad k \in \mathbb{Z}_+\},$$

$$\hat{\boldsymbol{y}}(k \mid \boldsymbol{x}, \boldsymbol{v}) = \overline{CA}^k\boldsymbol{x} + \overline{C}\sum_{j=1}^{k}\overline{A}^{j-1}\overline{B}\boldsymbol{v} + \overline{D}\boldsymbol{v},$$

where $\overline{A} := A - BK$, $\overline{B} := B(KG_x + G_u)$, $\overline{C} := C - DK$, and $\overline{D} := D(KG_x + G_u)$ represent the close loop system for a specified state feedback gain K [4], [6]. Furthermore, if $\mathcal{Y}$ is convex, O_∞ is convex and it can be expressed as

$$O_\infty := \{(\boldsymbol{x}, \boldsymbol{v}) \mid T_x\boldsymbol{x} + T_v\boldsymbol{v} \leq c\}.$$

Additionally, let us now define (i) the set of steady-state admissible auxiliary references

$$\mathcal{V} := G_y^{-1}\mathcal{Y} = \{v \mid G_yv \in \mathcal{Y}\},$$

where $G_y = \mathrm{CG}_x + \mathrm{DG}_u$, $\mathcal{V}$ is the set of constant $\boldsymbol{v}$'s that yields in steady-state values for $\boldsymbol{y}_k \in \mathcal{Y}$, and (ii) the set of admissible references

$$\mathcal{R} := G_z\mathcal{V} = \{G_zv \mid v \in \mathcal{V}\},$$

which is the corresponding set of steady values for $\boldsymbol{z}_k$.

C. MPC controller

Following the approach in [5], [6], we formulate the controller as a linear MPC that optimizes an objective and handles the constraints. Knowing the states and inputs in steady-state for a given $\boldsymbol{v}$, the strategy will solve each sampling time the following optimal control problem (OCP)

$$\min_{\boldsymbol{x}, \boldsymbol{u}} \quad \sum_{k=0}^{N-1} \|\boldsymbol{x}_k - \bar{\boldsymbol{x}}_v\|_Q^2 + \|\boldsymbol{u}_k - \bar{\boldsymbol{u}}_v\|_R^2 + \|\boldsymbol{x}_N - \bar{\boldsymbol{x}}_v\|_P^2 \tag{2a}$$

$$\begin{aligned} \text{s.t.} \quad & \boldsymbol{x}_0 = \boldsymbol{x}_{\text{ini}}, \quad (\boldsymbol{x}_N, \boldsymbol{v}) \in O_\infty, && \text{(2b)} \\ & \boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \quad \text{for} \quad k = 0, \ldots, N-1, && \text{(2c)} \\ & \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{D}\boldsymbol{u}_k \in \mathcal{Y} \qquad \text{for} \quad k = 0, \ldots, N-1, && \text{(2d)} \end{aligned}$$

where $N \in \mathbb{N}$ is the prediction horizon, $\boldsymbol{x} := \{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ and $\boldsymbol{u} := \{\boldsymbol{u}_0, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_{N-1}\}$ are the sets of decision variables, $\boldsymbol{x}_{\text{ini}}$ is the measurement or estimate of the initial states. Q, R and P are weighting matrices. Generally, the stage cost matrices satisfy $Q \succeq 0$, with (A, Q) detectable, and $R \succ 0$. Moreover, the abovementioned state feedback gain K is the associated LQR gain $K = (R + B^\top PB)^{-1}(B^\top PA)$ where P is the solution to the discrete algebraic Riccati equation $P = Q + A^\top PA - (A^\top PB)(R + B^\top PB)^{-1}(B^\top PA)$. Note that the control law K is needed to calculate O_∞ but is not used as the actual system controller. Hence, we redefine the reachable maximal output admissible set

$$\Gamma_N := \{(\boldsymbol{x}_{\text{ini}}, \boldsymbol{v}) \in \mathbb{R}^{n_x} \times \mathbb{R}^{n_v} \mid \exists \;\; (\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{u}) : (2b) - (2d)\} \tag{3}$$

as the N-step backward reachable set of O_∞. Γ_N is polyhedral since O_∞ is also polyhedral, and can be computed offline. Two polyhedral calculation methods are presented in [6, Sec IV.C]. This calculation is computationally expensive, and its complexity increases exponentially with the size of states and horizon.

D. Governor

The governor has the role of modifying the reference $\boldsymbol{r}$, providing an auxiliary reference $\boldsymbol{v}$ such that system (1) with MPC (2) in the feedback is always feasible and does not violate the constraints. The auxiliary reference $\boldsymbol{v}$ provided by the governor is calculated by solving the following optimization problem:

$$\min_{\boldsymbol{v} \in \mathcal{V}} \quad \|G_z \boldsymbol{v} - \boldsymbol{r}\|_2^2 \tag{4a}$$

$$\text{s.t.} \quad (\boldsymbol{x}, \boldsymbol{v}) \in \Gamma_N . \tag{4b}$$

Thus, having a state measurement or estimate we compute the auxiliary reference v in the set of admissible auxiliary references that is closest to r and belongs to the reachable maximal output admissible set Γ_N, meaning that all constraints $\mathcal{Y}$ will hold.

Solving, accordingly, problems (2) and (4) ensures constraint satisfaction. Moreover, a state calculated with the form $\boldsymbol{x} = G_x \boldsymbol{v}$ is asymptotically stable and has finite time convergence property [5], [6].

III. Time-optimal feasibility governor MPC (TOFG)

As mentioned above, the presented approach consists of two stages. In the first stage, a FG is used to drive the system in minimal time towards the given reference $\boldsymbol{r}$. When the system is close enough to the reference and the FG is not time-optimal anymore stage two is initiated, which is an actual time-optimal strategy. Further details on this approach are given below.

A. Feasibility governor with time-optimal behavior

To generate a time-optimal trajectory with the FG strategy, the set within which the auxiliary reference $\boldsymbol{v}$ is searched is Γ_N, the reachable maximal output admissible set. Hence, in order to achieve time-optimality, Γ_N should contain the time-optimal trajectory values toward the reference. The set Γ_N depends on O_∞ which, in turn, depends on Q and R.

Assumption 1: The values Q, R and N are designed such that $\Gamma_N \cap \text{bd}(\mathcal{Y}) \neq \emptyset$ where bd() represents the boundary of a set.

Assumption 1 enables the FG to reach the boundaries of the system defined by $\mathcal{Y}$. This is a necessary condition to achieve time-optimality. By minimizing the difference between the auxiliary reference $\boldsymbol{v}$ and the reference $\boldsymbol{r}$ in (4), the solution to this problem lies on the boundary of Γ_N. This, in addition to Assumption 1, makes the system hit state and/or input constraints – defined by $\mathcal{Y}$ – and drives the system with a time-optimal behavior toward the reference.

B. Time-optimal MPC

To implement a time-optimal strategy in an MPC fashion, a time-optimal OCP should be solved every sampling time. In each OCP, it is necessary that the reference is reachable within the prediction horizon because the terminal constraint specifying to be on target at the end of the horizon provides the guarantee of convergence and stability of the MPC [9]. The OCP is defined as follows:

$$\begin{aligned}
\min_{\boldsymbol{x}, \boldsymbol{u}, N_{TO}} \quad & N_{TO} \\
\text{s.t.} \quad & \boldsymbol{x}_0 = \boldsymbol{x}_{\text{ini}}, \\
& \boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \quad \text{for} \quad k = 0, \dots, N_{TO} - 1, \\
& \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{D}\boldsymbol{u}_k \in \mathcal{Y} \qquad \text{for} \quad k = 0, \dots, N_{TO} - 1 \ , \\
& \boldsymbol{x}_{N_{TO}} = \bar{\boldsymbol{x}}_r \qquad\qquad\qquad\qquad\qquad\qquad , \\
& N_{TO} \leq N_{TO_{\max}} .
\end{aligned}$$

The optimum value $N_{TO}^* \in \mathbb{N}_{\leq N_{TO_{\max}}}$ is the minimum horizon length to reach the steady-state system state corresponding to the reference $\bar{\boldsymbol{x}}_r = G_x r$ in less than $N_{TO_{\max}}$ steps, where $N_{TO_{\max}}$ is the maximum prediction horizon that allows a practical implementation satisfying real time constraints. From a practical approach, this formulation has disadvantages because it is a mixed integer problem with changing horizon length. The changing horizon length modifies the problem dimensions at each sampling time, making it inconvenient for implementations. Therefore, the problem is reformulated for a fixed horizon length $N_{TO} \geq N_{TO}^*$ [9]

$$\begin{aligned}
\min_{\boldsymbol{x}, \boldsymbol{u}} \quad & \sum_{k=0}^{N_{TO}-1} \theta^k \|\boldsymbol{x} - \bar{\boldsymbol{x}}_v\|_1 \\
\text{s.t.} \quad & \boldsymbol{x}_0 = \boldsymbol{x}_{\text{ini}}, \\
& \boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \quad \text{for} \quad k = 0, \dots, N_{TO} - 1, \\
& \boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{D}\boldsymbol{u}_k \in \mathcal{Y} \qquad \text{for} \quad k = 0, \dots, N_{TO} - 1 \ , \\
& \boldsymbol{x}_{N_{TO}} = \bar{\boldsymbol{x}}_r .
\end{aligned}$$

where $\theta > 1$ is a fixed parameter that creates exponential weights along the horizon, inducing sparsity in the state error at the horizon end. In this formulation, the system reaches the reference at N_{TO} at the latest. But this objective function is not differentiable. Therefore, we apply the sum of absolute residuals approximation [10] using a slack variable to cast the problem into a linear program:

$$\min_{\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{s}} \quad \sum_{k=0}^{N_{TO}-1} \theta^k \ \mathbf{1}^\top \boldsymbol{s} \tag{5a}$$

$$\text{s.t.} \quad \boldsymbol{x}_0 = \boldsymbol{x}_{\text{ini}}, \tag{5b}$$

$$-\boldsymbol{s} \leq \boldsymbol{x} - \bar{\boldsymbol{x}}_v \leq \boldsymbol{s} \qquad , \tag{5c}$$

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k \ \text{for} \ k = 0, \dots, N_{TO} - 1, \tag{5d}$$

$$\boldsymbol{C}\boldsymbol{x}_k + \boldsymbol{D}\boldsymbol{u}_k \in \mathcal{Y} \qquad \text{for} \ k = 0, \dots, N_{TO} - 1 \ , \tag{5e}$$

$$\boldsymbol{x}_{N_{TO}} = \bar{\boldsymbol{x}}_r . \tag{5f}$$

This OCP formulation can be implemented in an MPC fashion, i.e., it is solved at each sampling time without changing the problem dimension. The set of states for which (5d)-(5f) holds is definded as the reachable reference set

$$\mathcal{R}_{TO} := \{\boldsymbol{x}_{\text{ini}} \mid \exists \ \boldsymbol{u} : (5\text{d}) - (5\text{f})\} \subseteq \mathbb{R}^{n_x} , \tag{6}$$

which is the set of states from which the reference can be reached within a maximum N_{TO} time steps without violating imposed constraints. This set is calculated as the N_{TO}-step backward dynamics propagation of the reference point, taking into account input constraints. This procedure is similar to the computation of Γ_N in (II-C). The horizon N_{TO} should be large enough such that the reachable reference set $\mathcal{R}_{TO}$ reaches bd($\mathcal{Y}$) to allow that the switching of stage happens while the system is hitting the constraints.

C. Algorithm

With all the elements and sets defined, the feasibility governor is applied such that it drives the system as fast as possible toward the reference, and once the system is inside the reachable reference set, we switch the problem to a time-optimal scheme. The time-optimal feasibility governor (TOFG) is summarized in the Algorithm 1.

Algorithm 1 Time-Optimal Feasibility Governor

Calculation of G_x, G_u and G_z
Define Q,R,N,N_{TO}
Calculation of K and P
Calculate O_∞ and then calculate Γ_N
Calculate $\mathcal{R}_{TO}$
for *start* to *maximum execution time* **do**
  if $\boldsymbol{x}_0 \in \Gamma_N$ **then**
    if $\boldsymbol{x}_0 \notin \mathcal{R}_{TO}$ **then**
      Solve (4) and get $\boldsymbol{v}$
      Solve (2) to get $\boldsymbol{u}_0$ and apply it to the system
    else
      Solve (5) to get $\boldsymbol{u}_0$ and apply it to the system
    end if
  else
    The system is out of the region of attraction
  end if
  update the current system state to $\boldsymbol{x}_0$
end for

D. Implementation

For a given N_{TO} the FG should be designed such that it remains time-optimal until it reaches $\mathcal{R}_{TO}$. Note that $\mathcal{R}_{TO} \cap \text{bd}(\mathcal{Y}) \neq \emptyset$ is needed to guarantee a transition between stages keeping time-optimality. This condition can be verified offline by increasing N_{TO} until a maximum value $N_{TO_{\max}}$. After that, if FG loses the time-optimality before reaching $\mathcal{R}_{TO}$, the design variables for FG (Q, R, N) should be tuned to extend the time-optimal behavior until it reaches $\mathcal{R}_{TO}$.

Set calculations are performed offline before the actual system execution, which is an advantage of this strategy. The sets Γ_N and $\mathcal{R}_{TO}$ can be expressed as the projection $\Pi_{x_0,v}$ and Π_{x0} of the variables $\boldsymbol{v}$ and $(\boldsymbol{x}, \boldsymbol{u})$ along the prediction horizon (see Appendix I). For Γ_N, the N-step backward propagation of O_∞ can be represented in inequality form. We consider the constraints from (2) in the form $L[\boldsymbol{x}, \boldsymbol{v}]+M\boldsymbol{u} \leq$

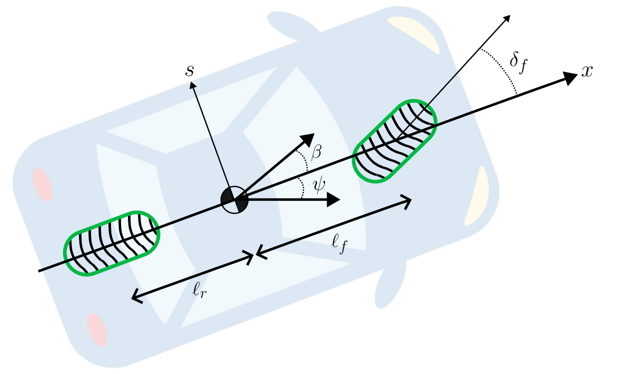

Fig. 2. Car representation by a bicycle model

b. Therefore, the set from (3) can be expressed as

$$\Gamma_N = \Pi_{x_0,v}\{(\boldsymbol{x}, \boldsymbol{v}, \boldsymbol{u}) \mid L[\boldsymbol{x}, \boldsymbol{v}]^\top + M\boldsymbol{u} \leq b\}. \tag{7}$$

Similarly, we define the N_{TO}-step backward propagation of the reference to represent $\mathcal{R}_{TO}$ from (6) as

$$\mathcal{R}_{TO} = \Pi_{\boldsymbol{x}_0}\{(\boldsymbol{x}, \boldsymbol{u}) \mid L_1\boldsymbol{x} + M_1\boldsymbol{u} \leq b_1\}. \tag{8}$$

Details about the structure and components of the matrices above mentioned are presented in Appendix I. The projection $\Pi_{\boldsymbol{x}_0}$ is the most computationally expensive operation. As the number of states or horizon increases, it becomes impractical since the algorithm complexity grows exponentially. To perform this operation, the package `Bensolve tools` was used [11], [12]. The implementation was made in `MATLAB` 2021b using the toolchain for model predictive control `IMPACT` [13], which provides a framework for easy and meaningful implementations of MPC allowing the user to interface different numerical optimization solvers.

IV. Numerical Example

In this section, an example implementation of TOFG is presented. The case considered for design and simulation is the numerical example from [6], [14], which is the lateral movement of a car at a constant forward velocity V_x. This illustrates, for example, a part of a lane changing maneuver.

In Fig. 2, the diagram illustrates the bicycle model used in this example. The system state vector is denoted as $\boldsymbol{x} := [s, \psi, \beta, \omega]^\top$, where s represents the lateral position of the vehicle, ψ is the yaw angle, $\beta = \dot{s}/V_x$ stands for the sideslip angle, and $\omega = \dot{\psi}$ denotes the yaw rate. The control input to the system is the front steering angle $u := \delta_f$. Additionally, the system is subject to constraints on the vector $\mathbf{y} := [\alpha_f, \alpha_r, \delta_f]^\top$, where α_f and α_r represent the front and rear slip angles, respectively. The tracking output is designated as $z = s$. The matrices that characterize the continuous-time model of the system as in (1) are as follows

$$A := \begin{bmatrix} 0 & V_x & V_x & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{2C_\alpha}{mV_x} & \frac{C_\alpha(\ell_r-\ell_f)}{mV_x^2} - 1 \\ 0 & 0 & \frac{C_\alpha(\ell_r-\ell_f)}{I_{zz}} & -\frac{C_\alpha\left(\ell_r^2+\ell_f^2\right)}{I_{zz}V_x} \end{bmatrix}, B := \begin{bmatrix} 0 \\ 0 \\ \frac{C_\alpha}{mV_x} \\ \frac{C_\alpha \ell_f}{I_{zz}} \end{bmatrix},$$

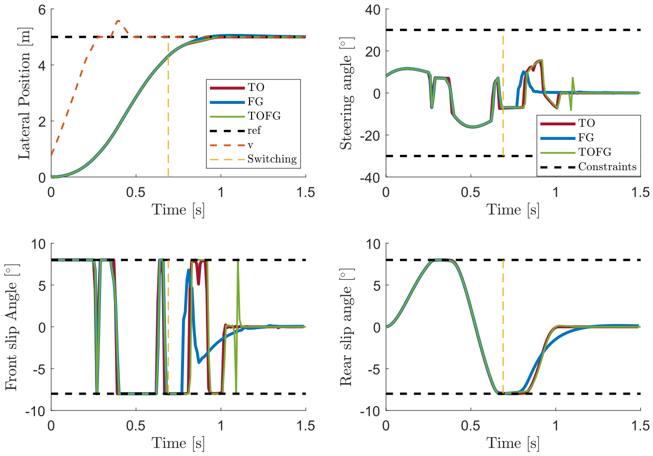

Fig. 3. Results for the lateral movement of the vehicle. Three strategies are compared: the time-optimal (TO) with time window $N = 104$, the feasibility governor (FG) for $N = 5$, and the time-optimal feasibility governor (TOFG) for $N = 5$, $N_{TO} = 35$. The yellow dashed vertical line indicates where the system reaches $\mathcal{R}_{TO}$ and the TOFG is switched to a time-optimal MPC strategy.

$$C := \begin{bmatrix} 0 & 0 & -1 & -\frac{\ell_f}{V_x} \\ 0 & 0 & -1 & \frac{\ell_r}{V_x} \\ 0 & 0 & 0 & 0 \end{bmatrix}, D := \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix},$$

$E = [1000]$, and $F = 0$, where $m = 2041$ kg is the mass of the vehicle, $I_{zz} = 4964\ \mathrm{kg} \cdot \mathrm{m}^2$ is the moment of inertia around the yaw axis, $\ell_f = 1.56$ m and $\ell_r = 1.64$ m are the moment arms of the front and rear wheels relative to the center of mass, respectively, and $C_\alpha = 246994$ N/rad is the tire stiffness. The continuous-time system matrices are converted to discrete-time using a zero-order hold with a sampling time of $t_s = 0.01$ s. The constraint set is $\mathcal{Y} = [-8°, 8°] \times [-8°, 8°] \times [-30°, 30°]$, which represents limits on the front and rear slip angles (to prevent tire slip and drifting), and a mechanical limit on the steering angle. The initial condition is $\boldsymbol{x}_0 = \mathbf{0}$, the reference is $\boldsymbol{r} = 5$ m. The terminal penalty P and gain K are computed using the LQR equations. The matrices $Q = \mathrm{diag}([1, 0.01, 0.01, 1])$ and $R = 0.192$ were fine-tuned to achieve input saturation at the beginning of the trajectory.

Fig. 3 shows results for a time-optimal OCP formulation with a horizon of $N = 104$, which is the minimum value needed to have a feasible problem, i.e., to reach the reference within the prediction horizon. Results of FG, $N = 5$, and TOFG for same N and $N_{TO} = 35$ are also shown in Fig. 3.

The inputs remain inside the feasible range without violating the constraints. Before the yellow line, the TOFG performs in the same way as the FG and they follow closely the TO solution. After the yellow line, The TOFG follows the TO solution, reaching the reference at the same time. The values of $\boldsymbol{v}$ display how the auxiliary reference created by the governor drives the system toward the reference. After reaching the reference, $\boldsymbol{v}$ has a small peak to keep the states in the feasible region. There are small differences between TOFG and TO that could be attributed to the Q and R tuning, or to the ℓ_1 norm approximation made in the OCP (5). These effects can be corrected by adding penalization to the signal increments. However, this is out of the scope of this work.

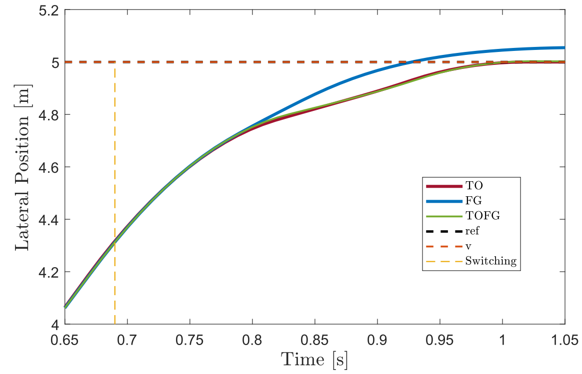

Fig. 4. Zoom of the results for the lateral movement of the vehicle. Here is shown in detail the lateral position transition between FG and TO in the TOFG strategy. It can be seen how TO and TOFG reach the reference at the same time.

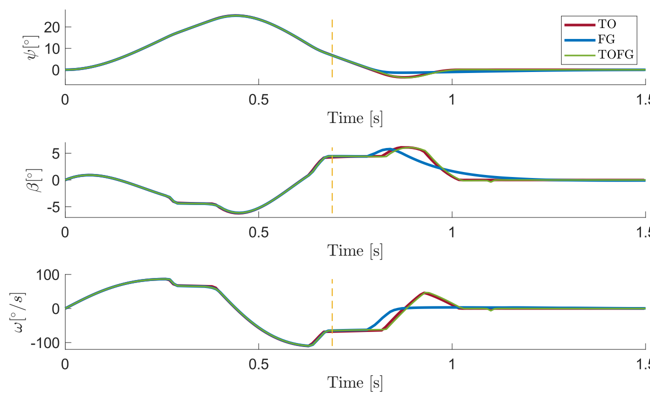

Fig. 5. Yaw, sideslip and yaw rate for the lateral movement of the vehicle. The strategies time-optimal (TO) for $N = 104$, feasibility governor (FG) for $N = 5$, and time-optimal feasibility governor (TOFG) for $N = 5$, $N_{TO} = 35$ are shown. The yellow dashed vertical line indicates where the system reaches $\mathcal{R}_{TO}$ and the TOFG is switched to a time-optimal strategy.

The details of Fig. 4 show that the problem switching takes place at 0.69 s and then, the behavior of TOFG mimics a time-optimal solution, showing how the proposed strategy can be time-optimal while making a safe lateral movement (no constraint violation). The TO MPC requires a horizon of only $N_{TO} = 35$, while if the TO MPC would have been applied from the initial state, it would require a horizon $N = 104$. The results for yaw, sideslip and yaw rate in Fig. 5 show that the TOFG follows the time-optimal solution after the switching point (after the system reaches $\mathcal{R}_{TO}$) indicated by the yellow dashed vertical line.

V. Conclusion

The methodology time-optimal feasibility governor MPC (TOFG) was presented as an extension of the Feasibility Governor (FG) strategy. The proposed method allows us to make time-optimal point-to-point motions using a short MPC horizon. This is achieved by computing offline the maximal output admissible set and the reachable reference set.

These sets guarantee properties like feasibility, no constraint violation (safety), and reference reachability. An auxiliary reference is calculated by the governor, and the MPC horizon for the online calculation is significantly shorter than that needed in traditional formulations.

Future work includes the exploration of cases involving piecewise constant references to generalize the methodology behavior, taking advantage of linear properties. Additionally, we aim to develop set approximations to simplify definitions for hardware implementations and develop a systematic design method.

APPENDIX I
BACK PROPAGATION SET MATRICES

These matrices represent the constraints of the OCP problem (2) in an inequality structured form. First, the equality constraints corresponding to the system dynamics are expressed as double inequalities. For each state $\boldsymbol{x}_a$ and input $\boldsymbol{u}_a$ we have

$$A\boldsymbol{x}_a - \boldsymbol{x}_{a+1} + B\boldsymbol{u}_a \leq 0 \quad , \quad -A\boldsymbol{x}_a + \boldsymbol{x}_{a+1} - B\boldsymbol{u}_a \leq 0.$$

The output feasible set can be described in terms of the states and inputs as

$$Y\boldsymbol{y}_a \leq h \quad \rightarrow \quad YC\boldsymbol{x}_a + YD\boldsymbol{u}_a \leq h.$$

Now using the above defined inequalities and the set definitions from Section II, we can construct the following matrices:

$$L_d = \begin{bmatrix} A & -I & 0 & \cdots & 0 \\ 0 & A & -I & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & A & -I \end{bmatrix} M_d = \begin{bmatrix} B & 0 & \cdots & 0 \\ 0 & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & B \end{bmatrix}$$

$$L_i = \begin{bmatrix} YC & 0 & \cdots & 0 \\ 0 & YC & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & YC \end{bmatrix},$$

$$M_i = \begin{bmatrix} YD & 0 & \cdots & 0 \\ 0 & YD & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & YD \end{bmatrix}.$$

Therefore

$$L = \begin{bmatrix} L_d & 0 \\ -L_d & 0 \\ L_i \; 0 & 0 \\ 0 \;\; T_x & T_v \end{bmatrix}, M = \begin{bmatrix} M_d \\ -M_d \\ M_i \\ 0 \end{bmatrix}, b = \begin{bmatrix} 0 \\ 0 \\ 1_N \otimes h \\ c \end{bmatrix},$$

for variables $\boldsymbol{x} = [\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$, $\boldsymbol{v}$, and $\boldsymbol{u} = [\boldsymbol{u}_0, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_{\boldsymbol{N-1}}]$, and

$$L_1 = \begin{bmatrix} L_d \\ -L_d \\ L_i \;\; 0 \\ 0 \;\;\; I \\ 0 \; -I \end{bmatrix}, M_1 = \begin{bmatrix} M_d \\ -M_d \\ M_i \\ 0 \\ 0 \end{bmatrix}, b_1 = \begin{bmatrix} 0 \\ 0 \\ 1_{N_{TO}} \otimes h \\ \text{ref} \\ -\text{ref} \end{bmatrix},$$

for variables $\boldsymbol{x} = [\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_{N_{TO}}]$ and $\boldsymbol{u} = [\boldsymbol{u}_0, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_{\boldsymbol{N_{TO}-1}}]$. The interested reader is referred to [6, Appendix B] for an additional version of a condensed formulation.

REFERENCES

[1] R. Verschueren, S. De Bruyne, M. Zanon, J. V. Frasch, and M. Diehl, "Towards time-optimal race car driving using nonlinear mpc in real-time," in *53rd IEEE Conference on Decision and Control*, 2014, pp. 2505–2510.

[2] L. Van den Broeck, M. Diehl, and J. Swevers, "Model predictive control for time-optimal point-to-point motion control," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 2458–2463, 2011, 18th IFAC World Congress.

[3] C. Rosmann, A. Makarow, F. Hoffmann, and T. Bertram, "Time-optimal nonlinear model predictive control with minimal control interventions," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 2017, pp. 19–24.

[4] E. Garone, S. Di Cairano, and I. Kolmanovsky, "Reference and command governors for systems with constraints: A survey on theory and applications," *Automatica*, vol. 75, pp. 306–328, 2017.

[5] T. Skibik, D. Liao-McPherson, T. Cunis, I. Kolmanovsky, and M. M. Nicotra, "Feasibility governor for linear model predictive control," in *2021 American Control Conference (ACC)*, 2021, pp. 2329–2335.

[6] ——, "A feasibility governor for enlarging the region of attraction of linear model predictive controllers," *IEEE Transactions on Automatic Control*, vol. 67, no. 10, pp. 5501–5508, 2022.

[7] T. Skibik, D. Liao-McPherson, and M. M. Nicotra, "A terminal set feasibility governor for linear model predictive control," *IEEE Transactions on Automatic Control*, vol. 68, no. 8, pp. 5089–5095, 2023.

[8] D. Limon, I. Alvarado, T. Alamo, and E. Camacho, "Mpc for tracking piecewise constant references for constrained linear systems," *Automatica*, vol. 44, no. 9, pp. 2382–2387, 2008.

[9] R. Verschueren, H. J. Ferreau, A. Zanarini, M. Mercangöz, and M. Diehl, "A stabilizing nonlinear model predictive control scheme for time-optimal point-to-point motions," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, 2017, pp. 2525–2530.

[10] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[11] A. Löhne and B. Weißing, "Equivalence between polyhedral projection, multiple objective linear programming and vector linear programming," *Mathematical Methods of Operations Research*, vol. 84, no. 2, pp. 411–426, 2016.

[12] D. Ciripoi, A. Löhne, and B. Weißing, "A vector linear programming approach for certain global optimization problems," *Journal of Global Optimization*, vol. 72, no. 2, pp. 347–372, 2018.

[13] A. Florez, A. Astudillo, W. Decré, J. Swevers, and J. Gillis, "Impact: A toolchain for nonlinear model predictive control specification, prototyping, and deployment," in *IFAC world congress*, 2023.

[14] J. Wurts, J. L. Stein, and T. Ersal, "Collision imminent steering using nonlinear model predictive control," in *2018 Annual American Control Conference (ACC)*, 2018, pp. 4772–4777.