Deterministic Operating Strategy for Multi-objective NMPC for Safe Autonomous Driving in Urban Traffic

Mostafa Emam and Matthias Gerdts
Department of Aerospace Engineering, Institute of Mathematics and Applied Computing, Universitat der Bundeswehr Munchen, 85579, Neubiberg, Germany

Keywords: Autonomous Driving, Multi-objective NMPC, Path Tracking, Finite-State Machine.

Abstract: In this paper, we introduce a deterministic operating methodology based on finite-state automata to employ multi-objective Nonlinear Model Predictive Control (NMPC) in autonomous driving applications. We begin with discussing the system's dynamical behavior and the proposed constraints to guarantee safe driving. Then, we examine a typical urban scenario and dissect it into a set of interacting sequences, so that we develop and fine-tune separate MPC-based controllers for each of these sequences. Finally, we introduce a Finite-State Machine (FSM) that analyzes the current driving situation and accordingly selects the appropriate controller to compute the optimal control action. This approach is numerically simulated and tested with the software OCPID-DAE1 and results show its success in accordance with multi-objective NMPC.

NOMENCLATURE

Abbreviations

ACC	Adaptive Cruise Control
FSM	Finite-State Machine
LKA	Lane Keeping Assistant
MIP	Mixed-Integer Programming
MPC	Model Predictive Control
NMPC	Nonlinear Model Predictive Control

FSM States

ND	End (Final state)
NP	Enter Parking
PF	Path Following
PU	Pulling Up
SS	Standstill
XP	Exit Parking

1 INTRODUCTION

Over the past years, autonomous driving has become an increasingly popular topic both in academia and industry due to its many advantages, such as improving road safety, optimizing fuel/energy consumption, and enhancing the overall traveling experience (Lu et al., 2004). This interdisciplinary topic requires intensive research in diverse fields like environment perception, sensor data fusion, and control theory, which is why the quest for a system with full automation capabilities is still incomplete to this day. Nonetheless, multiple specialized systems have been successfully developed to fulfill specific objectives, such as *Adaptive Cruise Control (ACC)* and *Lane Keeping Assistant (LKA)*, and are currently being used in commercial vehicles with great success (Yurtsever et al., 2020). So, in this work we focus on the design of a path tracking control algorithm with collision-avoidance capabilities, in which we use NMPC as the control strategy due to its flexibility and wide applicability (Gerdts, 2018).

(Luo et al., 2010) proposes an ACC system based on multi-objective MPC that combines safety, comfort, and economic objectives, yet it formulates the vehicle longitudinal dynamics linearly, which greatly restricts its applicability. (Chen et al., 2021) continues with the same model and introduces a *Finite-State Machine (FSM)* to accommodate for more complex driving scenarios, which yields improved results but does not fully represent the ego-vehicle's lateral dynamics, thus producing sub-optimal results when following curved trajectories. Alternatively, (Zhang et al., 2017) uses a vehicle model with longitudinal and lateral dynamics and proposes a multi-objective FSM-based control scheme that encourages lane changing when driving behind a slower vehicle. However, the model implies independent vehicle dynamics, which may generate

Emam, M. and Gerdts, M.
Deterministic Operating Strategy for Multi-objective NMPC for Safe Autonomous Driving in Urban Traffic.
DOI: 10.5220/0011115400003191
In *Proceedings of the 8th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2022)*, pages 152-161
ISBN: 978-989-758-573-9; ISSN: 2184-495X

infeasible trajectories in real applications. (Gutjahr et al., 2017) introduces a more complex model, which formulates the vehicle movement with respect to a reference curve, then uses this model to develop a linear time-varying MPC that incorporates collision avoidance of static and dynamic obstacles. This offers a generic approach, which has also been employed by (Britzelmeier and Gerdts, 2020) to successfully control a two-wheel driven robot model.

By taking inspiration from these sources, we propose a multi-objective FSM-based framework for controlling the ego-vehicle in urban driving scenarios. Instead of constructing a monolithic controller to cover all possible cases like (Xiao et al., 2021), or utilizing a FSM to compute simplistic control actions like (Bae et al., 2020), we adopt a different perspective, in which we split the control problem into a set of sub-problems, then develop and fine-tune specialized controllers for each of these sub-problems. Accordingly, we construct a FSM that not only activates the optimal controller in any given scenario, but also guarantees optimal and smooth controls when transitioning from one controller to the other.

In this paper, we first discuss the vehicle motion model as well as the required constraints for path tracking and safe following of a leading road user in section II. We also represent the problem from a MPC perspective and discuss constructing the multi-objective cost function. In section III, we analyze a typical urban scenario and break it down into a set of driving sequences, for which different controllers can be developed and fine-tuned. Accordingly, we develop the FSM modes and examine their interrelations and transition conditions. Finally, we validate the developed framework and show the achieved results in section IV, and present our conclusions and ideas for future work in section V.

2 PROBLEM FORMULATION

2.1 Ego-vehicle Modeling

Since the success of the MPC strategy heavily relies on the formulation of the system dynamics (Grüne and Pannek, 2011), we must model the ego-vehicle's behavior using a proper motion model. The kinematic vehicle model is a simple, generic model that has already been proven effective in developing controllers for autonomous vehicles (Britzelmeier et al., 2020) and it focuses on the vehicle's geometrical movement rather than the forces acting on it. This model can be used in coordination with a curvilinear coordinate system to describe the movement of a specific point on the ego-vehicle, i.e., the rear axle's middle point, relative to a reference curve $\gamma_{ref} : [0, L] \to \mathbb{R}^2$ using:

$$s'(t) = \frac{v(t)\cos\chi(t)}{1 - d(t)\cdot\kappa_{ref}(s(t))} \tag{1a}$$

$$d'(t) = v(t)\sin\chi(t) \tag{1b}$$

$$\chi'(t) = \psi'(t) - \psi'_{ref}(t) = v(t)\kappa(s(t)) - s'(t)\kappa_{ref}(s(t)) \tag{1c}$$

$$\kappa'(t) = u_1(t) \tag{1d}$$

$$v'(t) = u_2(t) \tag{1e}$$

where s is the arc length of the projection point unto γ_{ref}, d is the lateral offset of this point to γ_{ref}, and χ is the relative course angle, i.e., the difference between the ego-vehicle's heading ψ and the reference curve's yaw angle ψ_{ref} (Burger and Gerdts, 2019). Moreover, the system inputs are formulated as generic quantities, such that we can easily map them to the actual controls required for a specific vehicle type provided that an adequate mapping $(x, u, X) \mapsto U = \mu(x, u, X)$ exists (Britzelmeier and Gerdts, 2020).

Second, we define the occupancy region of the ego-vehicle (and any other vehicle) as the rectangular area that completely covers its footprint in 2-D space. We can then optimally cover this area with disks (Xiao et al., 2021), such that we detect a collision with an object (or lane markings) by simply checking if it exists inside the covering disks with *a priori* knowledge of the disks' positions and radii. This allows collision avoidance to be enforced using *Control Barrier Functions* (*CBF*) (Xiao et al., 2021). Note that determining the number and radii of disks that optimally cover a road user is an optimization problem that can be solved beforehand, for example, by taking notes from (Studier, 2022) to build a comprehensive database of road users and corresponding covering disks, such that this data is readily available during the system operation.

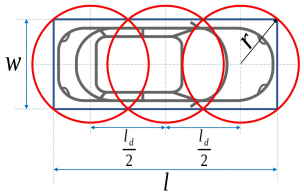

Figure 1: Optimal coverage of a vehicle's occupancy region with (3) disks of constant radius (r).

2.2 Clearance Constraints for Admissible Driving

To avoid endangering the ego-vehicle, it must always travel inside its driving lane, which can be guaranteed

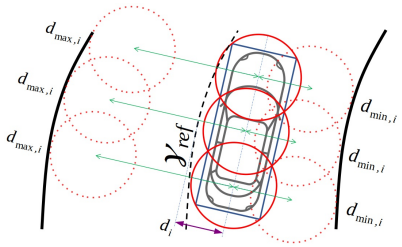

Figure 2: Representation of clearance constraints for driving within the permissible area ($clear_{LN}$).

by restricting the ego-vehicle's lateral offset d to the reference curve γ_{ref} as per the traffic lane markings on the right and left sides as depicted in Figure 2. For the lateral offset of each of the the ego-vehicle's disks d_i, we define the constraints $d_{min,i} \leq d_i \leq d_{max,i}, \forall i \in \{1,...,nrDisk_{ego}\}$, in which we calculate d_i of any disk using trigonometric functions with an assumed constant κ_{ref} across the ego-vehicle's length l (Gutjahr et al., 2017). For example, assume we have 3 disks of constant radius r covering the ego-vehicle, and they are located at $\frac{l_d}{2}$ from each other as shown in Figure 1; when driving in one permissible lane with γ_{ref} residing in the middle of it, we write the constraints using the lane width (w_{LN}) as:

$$(d)^2 \leq \left(\frac{w_{LN}}{2} - r\right)^2 \tag{2a}$$

$$\left(d + \frac{l_d}{2}\sin\chi\right)^2 \leq \left(\frac{w_{LN}}{2} - r\right)^2 \tag{2b}$$

$$\left(d + l_d \sin\chi\right)^2 \leq \left(\frac{w_{LN}}{2} - r\right)^2 \tag{2c}$$

If needed, we may use *slack*-variables $\eta_{LN} \geq 0$ (Britzelmeier and Gerdts, 2020) to *relax* the clearance constraints, which allows for reaching a better compromise between solution feasibility and system stability (Vu et al., 2021). For example, we can write this for the first disk as $clear_{LN,1} := (d)^2 - (\frac{w_{LN}}{2} - r)^2 \leq \eta_{LN}$.

2.3 Clearance Constraints for Safe following

As defined in (Bouska, 2021), a minimum longitudinal clearance (safety distance) must always be kept between the ego-vehicle and the closest leading traffic participant in the same lane. This can be determined using a modified *Constant Time Headway* policy (Swaroop and Rajagopal, 2001) as:

$$s_{SF}(t) = max(s_{SF,min}, v(t)t_h) \tag{3}$$

where $s_{SF,min}$ is the minimum safety distance, t_h is the specified time headway, and s_{SF} is the required safety distance. To enforce this metric, (Bosch, 2003) proposes to identify the closest road user in the same lane as the ego-vehicle and enforce the safety distance only for this particular user, which is both simple and valid. We achieve this by adding a clearance constraint $clear_{SF}$ between the rearmost coverage disk of the leading road user and the foremost disk of the ego-vehicle as:

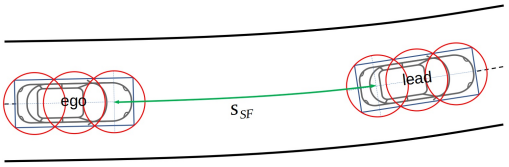

Figure 3: Required clearance for safe following ($clear_{SF}$).

$$s_{lead} - s_{ego} \geq s_{SF} \tag{4}$$

where $s_{lead/ego}$ represents the arc length of the leading road user and ego-vehicle respectively. Like $clear_{LN}$, $clear_{SF}$ can be relaxed with $\eta_{SF} \geq 0$.

2.4 Identification of the Closest In-lane Road User

To appropriately enforce $clear_{SF}$, we first need to identify the leading *in-lane* road user, i.e., the closest user driving in the same lane as the ego-vehicle. (Chen et al., 2021) computes the probability that a road user is in-lane using its relative lateral position and velocity separately, then merges this information to build a combined probability map. However, this approach is lacking, as the analysis of the relative lateral velocity is inherently dependent on the position. So instead, we propose to analyze the codependent probabilities that a road user is in-lane based on its relative lateral position and velocity simultaneously, then augment this data with its longitudinal distance to get a combined probability value P_{inLN}. Here, we will only consider the rearmost coverage disk of other road users in the identification process for simplification. After computing P_{inLN} for all road users, we simply choose the user with the highest value as the leading road user. Note that this approach not only promotes safety by giving higher priority to road users closer to the ego-vehicle, but it also reacts proactively to both Cut-In and Cut-Out maneuvers, which is not guaranteed by (Chen et al., 2021).

2.4.1 Relative Lateral Distance

The possibility that a road user exists in the same lane as the ego-vehicle is inversely proportional to the relative lateral deviation between them, as calculated with respect to the reference curve γ_{ref}. We define the probability $P_{inLN,d} \in [0,1]$, where the road user is considered in-lane when $P_{inLN,d} = 1$ and not in-lane when $P_{inLN,d} = 0$, and we derive this value from the

relative lateral deviation $d_{rel} = d_{oth} - d_{ego}$ using the sigmoid function:

$$P_{inLN,d} = \frac{1}{1 + e^{(\alpha_d + \beta_d |d_{rel}|)}}, d_{rel} \in \mathbb{R} \quad (5)$$

where α_d and β_d are specified based on the lane width

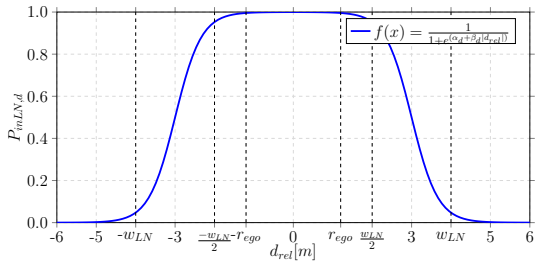

Figure 4: $P_{inLN,d}$ determines in-lane road users from d_{rel}.

w_{LN} and the radius of the ego-vehicle's coverage disks r_{ego}. An example of $P_{inLN,d}$ is depicted in Figure 4 with $r_{ego} = 1.0m$ and $w_{LN} = 4m$. Similarly, we can define functions to calculate the probability that a user is in the right or left adjacent lanes based on d_{rel}, which will be useful in the next section.

2.4.2 Relative Lateral Velocity

Here, we define the probability $P_{inLN,v} \in [0,1]$ derived from the relative lateral velocity $v_{rel,y} = v_{oth,y} - v_{ego,y}$ by using sigmoid functions, but we make a distinction between in-lane road users, i.e., whose d_{rel} is lower than a specific threshold, and users driving in the adjacent right/left lanes. For in-lane users, $P_{inLN,v}$ is inversely proportional to $v_{rel,y}$, as a low lateral velocity denotes a high probability to continue driving in-lane and a high $v_{rel,y}$ resembles a Cut-Out maneuver. We write this in a similar manner to Equation 5 to get the function demonstrated in Figure 5.

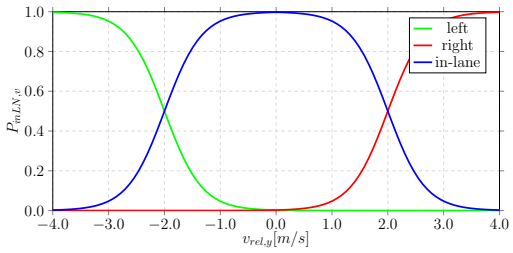

Figure 5: Calculation of $P_{inLN,v}$ from $v_{rel,y}$ for in-lane road users (blue), for users in the right lane (red) and in the left lane (green).

Alternatively, $P_{inLN,v}$ of road users driving in adjacent lanes must be calculated as a combination of both relative position and velocity. On the one hand, if the user is driving in the right adjacent lane with a positive $v_{rel,y}$, it indicates a Cut-In maneuver, i.e., a high probability to drive in-lane in an upcoming time step. Likewise, a Cut-In maneuver is plausible when the user is driving on the left with a negative $v_{rel,y}$. On the other hand, when the user is driving on the right with a negative relative velocity, it indicates a right-hand turn maneuver, i.e., $P_{inLN,v} = 0$ as the user will not realistically be driving in-lane of the ego-vehicle.

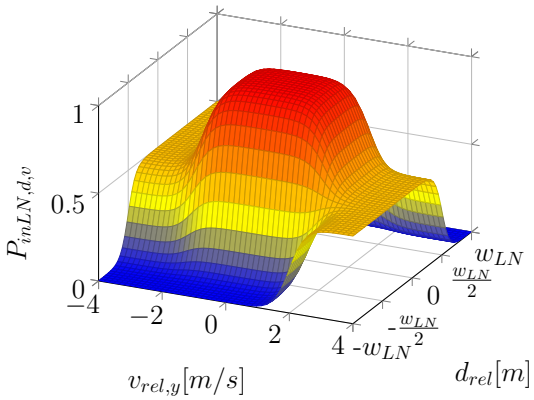

Figure 6: Combined probability $P_{inLN,d,v}$ from d_{rel} and $v_{rel,y}$.

To recap, we have three probabilities $P_{inLN,d,i}, \forall i \in \{1,2,3\}$, with which we identify the road user's location to be in-lane or in the right/left adjacent lanes. We have also three probabilities $P_{inLN,v,i}$ that indicate whether the user will continue to drive in its current lane or perform a Cut-In/Cut-Out maneuver. So, by combining these probabilities with different weighing factors ω_i as shown in Equation 6, we finally reach the probability $P_{inLN,d,v}$, which represents the likelihood a road user is in-lane of the ego-vehicle. This is also illustrated in Figure 6.

$$P_{inLN,d,v} = \sum_{i=1}^{3} \omega_n \cdot P_{inLN,d,i} \cdot P_{inLN,v,i} \quad (6)$$

2.4.3 Relative Longitudinal Distance

Lastly, we need to augment $P_{inLN,d,v}$ with the user's relative longitudinal distance $s_{rel} = s_{oth} - s_{ego}$ to prioritize road users driving closer to the ego-vehicle for safety considerations. In urban driving scenarios, the ego-vehicle typically drives with a maximum speed of $v_{max} = 50[km/h]$ (Bouska, 2021), for which a corresponding safety distance can be computed using (Swaroop and Rajagopal, 2001). Subsequently, we define $P_{inLN,s}$ as:

$$P_{inLN,s} = \frac{1}{1 + e^{(\alpha_s + \beta_s s_{rel})}}, s_{rel} \in \mathbb{R} \quad (7)$$

which is combined with $P_{inLN,d,v}$ to yield:

$$P_{inLN} = P_{inLN,s} \cdot P_{inLN,d,v} \quad (8)$$

After calculating P_{inLN} for all road users, the user with the highest probability is selected as the leading

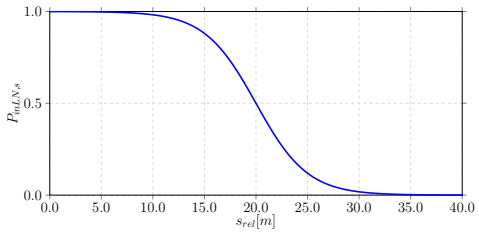

Figure 7: $P_{inLN,s}$ prioritizes close road users.

road user, for which $clear_{SF}$ will be enforced. In case multiple users have the exact same P_{inLN}, we choose the user with the smallest s_{rel} value, prioritizing closer users as intended. Note that with this formulation, we may enforce $clear_{SF}$ for a vehicle currently not in-lane but is performing a Cut-In maneuver, which increases the robustness of this approach and yields smoother system controls.

2.5 System Operational Limits

When driving in urban traffic, we must adhere to the laws and regulations, e.g., maximum/minimum speed limits. In addition, the physical construction of the ego-vehicle imposes restrictions on the system controls, as there is a cap on the maximum transmitted power by the steering actuator and the gas pedal or brakes (Gutjahr et al., 2017). We may also include safety-oriented constraints, such as limiting speed to avoid excessive lateral forces when driving in sharp curves (Gerdts, 2018), or comfort-oriented constraints, such as limiting the rate of change of the controls (Xiao et al., 2021). This has already been exhaustively discussed in literature, so we summarize by saying that the system adheres to the following constraints:

$$v_{min} \leq v(t) \leq v_{max} \tag{9a}$$
$$a_{min} \leq a(t) \leq a_{max} \tag{9b}$$
$$\kappa_{min} \leq \kappa(t) \leq \kappa_{max} \tag{9c}$$
$$u_{min} \leq u(t) \leq u_{max} \tag{9d}$$

and we refer the reader to (Xiao et al., 2021), (Luo et al., 2010), and (Chen et al., 2021) for further details.

2.6 MPC Cost Function Formulation

In this part, we focus on modeling the different goals in the multi-objective MPC cost function. We start with trajectory tracking, where our objective is to follow a desired path that spans from a start location A to a destination B, such that this path can be modeled as a spline curve $\gamma_{ref} : [s_0, s_f] \rightarrow \mathbb{R}^2$ with the way-points $\gamma_{ref}(s) := [x_{ref}(s), y_{ref}(s)]^{\mathrm{T}}$. We deliberately chose the dynamical model in Equation 1 to represent the ego-vehicle's movement with respect to a reference trajectory, as there exists a subset of the system states that mirrors tracking errors to this trajectory. In other words, for the system states vector $x(t) \in \mathbb{R}^{n_x}$, we have the error states vector $y(t) \in \mathbb{R}^{n_y}$ with $n_x \geq n_y$, and the path tracking problem is equivalent to stabilizing $y(t)$ to 0 (Xiao et al., 2021). Additionally, we model the incentive to reach the destination B as minimizing the difference between s and the total traveled distance at the destination s_f, or rather normalize it as $\frac{s_f - s}{s_f - s_0}$ for more consistent results.

Second, we employ a typical objective to improve passenger comfort and reduce fuel consumption, i.e., minimizing the system controls vector $u(t) \in \mathbb{R}^{n_u}$ (Luo et al., 2010). Finally, we include any added slack variables for constraints relaxation, such as $\eta_{LN,i}, i \in \{1, ..., n_{LN}\}$ for admissible driving $clear_{LN}$, and, if a leading in-lane road user exists, η_{SF} for safe following. This yields the combined cost function:

$$min \int_0^T \omega_y ||y|| + \omega_s \left(\frac{s_f - s}{s_f - s_0}\right)^2 + \omega_u ||u|| + \sum_{i=1}^{n_{LN}} \omega_{LN} {\eta_{LN,i}}^2 + \omega_{SF} {\eta_{SF}}^2 dt \tag{10}$$

with the weighing factors ω_y for path tracking, ω_s for maximizing travel distance, ω_u for minimizing control effort, and $\omega_{LN/SF}$ for relaxing the different clearance constraints. Note that this cost function is subject to the system dynamics (1), the operational limits (9) and the relaxed clearance constraints:

$$\begin{aligned} clear_{LN,i} &\leq \eta_{LN,i}, \forall i \in \{1, ..., n_{LN}\} \\ clear_{SF} &\leq \eta_{SF}, O_{inLN} \neq \emptyset, O_{inLN} \in O_{oth} \end{aligned} \tag{11}$$

where O_{oth} is the set containing all other road users and O_{inLN} is a single-entry set that contains the leading in-lane road user. This function serves as a basis for all controllers we intend to develop, as we can fine-tune the different ω factors for specific driving scenarios to better achieve their corresponding objectives, e.g., by prioritizing collision avoidance to path tracking in case of approaching a stationary object.

3 URBAN DRIVING AND VEHICLE CONTROL

To construct the system controller, we revise various sources that describe what a typical urban driving scenario entails, such as (Yurtsever et al., 2020), (Bae et al., 2020), and (Bosch, 2003). We also employ sources like (Bouska, 2021) and (ISO Central Secretary, 2018) to develop performance tests, with which we can guarantee the system credibility. In general, the ego-vehicle must travel across $[s_0, s_f]$

while adhering to the driving regulations and, when other road users exist, it must correctly identify the leading in-lane user and follow it at a safe distance to avoid collisions. Furthermore, it must adequately react to situations like Cut-In/Cut-Out maneuvers and traffic jams (Automatic Stop maneuver), and operate reliably both in straight and curved trajectories. Accordingly, we define four major sequences (*Exit Parking*, *Path Following*, *Pulling Up*, and *Enter Parking*), such that during a typical scenario, the ego-vehicle is either in one of these sequences or transitioning from one to the other. For each sequence, we develop and fine-tune a NMPC-based controller (with calibrated objectives and constraints), and we introduce a FSM to activate and switch between the different sequences and their corresponding controllers. This will be discussed in detail in the upcoming paragraphs.

3.1 FSM Architectural Approach

The FSM model is one of the fundamental methods for designing rule-based controllers (Macedo et al., 2015). From a mathematical perspective, it is a deterministic model of computation that portrays a system using states (also known as modes), inputs, and transition dynamics, and can be expressed as a quintuple $M := (S, \Sigma, \delta, S_0, F)$, where S is the finite (non-empty) set of states with $S_0 \in S$ as the initial state, Σ is the finite (non-empty) set of inputs, $\delta : S \times \Sigma \to S$ represents the inter-state transition function, and $F \subseteq S$ is the set of final states. For its operation, the FSM starts in S_0 and, according to the system input σ and transition function δ, it makes a transition to a different state $f \in S$. Then, f is set as S_0 and possible transitions are checked for the new σ and δ; this process is repeated until the final state is reached, for which F is an empty set (or is endlessly repeated if the FSM is cyclic) (Klose and Mester, 2018).

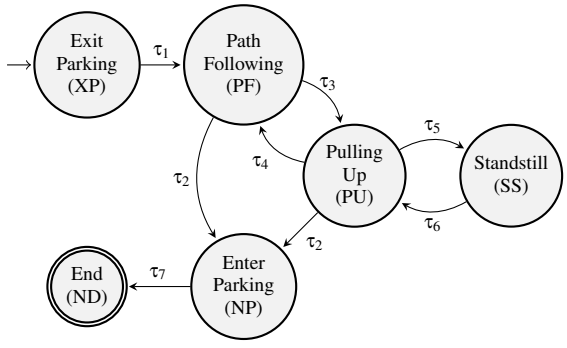

Figure 8: Configuration of the FSM modes and their possible transitions.

The FSM depicted in Figure 8 incorporates the aforementioned driving sequences, with the addition of the state (*Standstill*), which gets activated when the ego-vehicle drives with a very slow speed $v \leq 0.5[m/s]$ to avoid infeasible and aggressive system controls, and the final state (*End*) which promptly stops the ego-vehicle upon reaching its destination. Also, the conditions for switching between the FSM modes are portrayed in Table 1 with the symbols τ_i, such that a currently active state remains unchanged when all of its exit conditions are not satisfied (denoted with OW).

Table 1: FSM mode switch conditions.

S_0 \ f	XP	PF	PU	NP	SS	ND
XP	OW	τ_1	-	-	-	-
PF	-	OW	τ_3	τ_2	-	-
PU	-	τ_4	OW	τ_2	τ_5	-
SS	-	-	-	τ_6	OW	-
NP	-	-	-	OW	-	τ_7

Exit Parking (XP). This is the initial state that gets activated when starting the trip at A and it denotes that the ego-vehicle is currently driving inside the parking area. This imposes a restriction on the maximum speed (no faster than walking speed) and allows for sharper turns (due to traveling slowly), and, accordingly, we can calibrate the system constraints (cf. Equation 9) and objectives (cf. Equation 10). Since we already have γ_{ref} in accordance with geographical data, we can detect whether the ego-vehicle is inside the parking area or not by comparing s to a threshold s_{XP}, such that the condition for leaving the parking area τ_1 is fulfilled when $s \geq s_{XP}$. However, this yields aggressive controls, so we replace the hard equality condition with a transitioning phase between s_{XP} and a relaxed threshold $s_{XP,\eta}$. Consequently, we compute a transitioning factor ω_{τ_1} from s_{XP} and $s_{XP,\eta}$ as shown in Figure 9, and we revise τ_1 to be:

$$(s \geq s_{XP,\eta} \to \tau_1 = 1) \wedge (s < s_{XP,\eta} \to \tau_1 = 0) \quad (12)$$

where τ_1 is only fulfilled after passing the transitioning phase threshold $s_{XP,\eta}$.

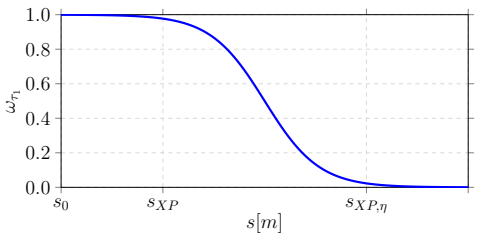

Figure 9: ω_{τ_1} yields smooth controls when transitioning from XP to PF.

Initially, we attempted to compute the transitioning

control action as a weighted average of the two state controllers with $u_{\tau_1} = \omega_{\tau_1} \cdot u_{XP} + (1 - \omega_{\tau_1}) \cdot u_{PF}$. Yet, this yielded a sub-optimal u_{τ_1}, despite both u_{XP} and u_{PF} being optimal. Alternatively, we attain optimal controls during the transitioning phase by introducing a new problem, at which ω_{τ_1} gradually modifies the system objectives and constraints. For example, we adjust the objective for path following to be:

$$\omega_{y,\tau_1} = \omega_{y,XP} + (\omega_{y,PF} - \omega_{y,XP}) \cdot (1 - \omega_{\tau_1}) \quad (13)$$

which is similarly done for all other objectives. As for the system constraints, we have no problem when transitioning to a less restricted environment, e.g., when $v_{max,XP} \leq v_{max,PF}$, as the constraint $v \leq v_{max,PF}$ is already fulfilled, so we use the same approach as Equation 13. Nevertheless, this is invalid when $\kappa_{max,XP} \geq \kappa_{max,PF}$, as an optimal control may already be at the operational limit $\kappa = \kappa_{max,XP}$, yielding an infeasible problem with $\kappa \nleq \kappa_{max,PF}$. Instead, we relax this constraint (and all other similar cases) using:

$$\begin{aligned} &\kappa_{max,\tau_1} = \kappa_{max,XP} + (\kappa_{max,PF} - \kappa_{max,XP}) \cdot (1 - \omega_{\tau_1}), \\ &\kappa - \kappa_{max,\tau_1} \leq \eta_{\kappa_{max}} \end{aligned} \quad (14)$$

thus we have a feasible control problem that can be optimally solved to achieve a smooth state transition.

Path Following (PF). This is the core urban driving sequence, where the ego-vehicle travels with an admissible speed and tries to maximize the traveled distance to its destination. Here, the turning curvature is restricted more than XP to avoid excessive lateral forces when turning with high speeds (promotes safety and comfort), and similarly for acceleration. There are multiple exit transitions from PF, thus we need to define a priority, or rather an order of operations (Zhang et al., 2017), with which we can sequentially evaluate these transitions. We specify the transition τ_2 to NP as the highest priority condition, as it implies that we are close to our destination B. Similar to τ_1, we use geo-data to determine a threshold s_{NP} based on s_f, and add a transitioning phase between a relaxed threshold $s_{NP,\eta}$ and s_{NP} using the transitioning factor ω_{τ_2}, with:

$$(s \geq s_{NP} \rightarrow \tau_2 = 1) \wedge (s < s_{NP} \rightarrow \tau_2 = 0) \quad (15)$$

The transition τ_3 to PU denotes that the ego-vehicle travels much slower than the speed limit either due to following a slow leading road user or approaching a traffic jam. So, we use the speed limits to construct the transitioning factor ω_{τ_3} as shown in Figure 11, with:

$$\tau_3 := \neg\tau_2 \wedge \left(v \leq \frac{v_{max} - v_{min}}{2}\right) \wedge (O_{inLN} \neq \emptyset) \quad (16)$$

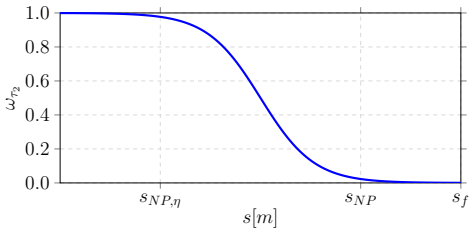

Figure 10: ω_{τ_2} for transitioning from PF to NP.

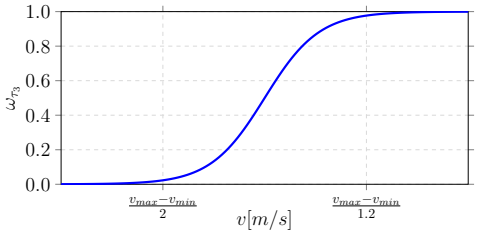

Figure 11: ω_{τ_3} for transitioning from PF to PU.

Pulling Up (PU). This sequence is not only suitable for following slow road users or approaching stationary ones, but can also be used for stopping at traffic lights by modeling the traffic junction as a stationary leading road user. Here, we prioritize ω_{SF} compared to ω_s to guarantee safety, and likewise adapt other objectives' weights. We also allow for more aggressive controls due to traveling relatively slowly. Similar to PF, the transition τ_2 to NP has the highest priority and follows the formula in Equation 15 with the same transitioning factor ω_{τ_2}. Afterwards, we have the transition τ_4 to PF, which has a higher priority than the transition τ_5 to SS to discourage the system from performing a complete stop when a stationary leading road user starts to accelerate. Complementary to Equation 16, we construct the transition phase elements as:

$$\begin{aligned} &\tau_4 := \neg\tau_2 \wedge \left(\left(v \geq \frac{v_{max} - v_{min}}{1.2}\right) \vee (O_{inLN} = \emptyset)\right), \\ &\omega_{\tau_4} = 1 - \omega_{\tau_3} \end{aligned} \quad (17)$$

Standstill (SS). Next, we have the utility state SS, which is responsible for completely stopping the ego-vehicle behind a stationary object, and acts as a buffer that prevents needless start-stop maneuvers behind a very slow leading user. First, we have the transition τ_5 from PU to SS, which we define as:

$$\tau_5 := \neg\tau_4 \wedge (v \leq 0.5[m/s]) \wedge (a \leq 0[m/s^2]) \quad (18)$$

and implement as a hard constraint, which is acceptable at this speed. This transition promptly stops the ego-vehicle if it is traveling very slowly while decelerating. Next, we have the transition τ_6 from SS back to PU, which we compute from the relative distance to

the leading user $s_{rel} = s_{lead} - s_{ego}$, the required safety distance s_{SF}, and an additional threshold $s_{SF,\eta}$ by:

$$\tau_6 := (s_{rel} \geq s_{SF} + s_{SF,\eta}) \vee (O_{inLN} = \emptyset) \quad (19)$$

which will either be fulfilled when the leading user starts moving (and cover a certain distance), or when no leading user can be detected anymore (relevant for modeling traffic lights).

Enter Parking (NP). This state is very similar to XP, with one difference being that we are almost at our destination. So, we have only one transition τ_7 to the final state ND, which we calculate from s, s_f, and some threshold s_{ND} as:

$$(s_f - s \leq s_{ND} \rightarrow \tau_7 = 1) \wedge (s_f - s > s_{ND} \rightarrow \tau_7 = 0) \quad (20)$$

such that when τ_7 is fulfilled, we directly transit to ND and promptly decelerate the ego-vehicle until a complete stop, thus concluding our trip.

Finally, it is worth mentioning that the proposed FSM architecture is loosely coupled with the NMPC strategy implemented in this work, i.e., we can make slight adaptations and employ this architecture for other control techniques as well, which may be investigated in future work.

4 NUMERICAL SIMULATION AND RESULTS

The proposed control architecture is implemented entirely in Fortran and the software OCPID-DAE1 is used to solve the formulated optimal control problem. To adequately evaluate the developed controller, we designed two test scenarios that combine different maneuvers, e.g., Cut-In and Cut-Out, as well as require transitioning between the different FSM states.

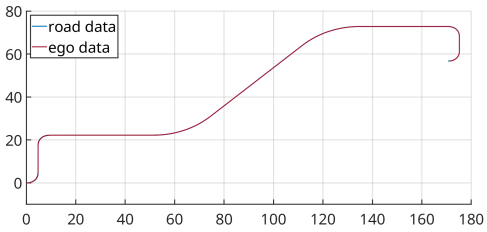

Figure 12: Travel path for the first test.

The first scenario includes a full test run from start to finish, where the ego-vehicle follows a trajectory with different turns and attempts to reach its destination as fast as possible. The travel path is shown in Figure 12, such that the ego-vehicle traverses this path with minimal error as illustrated in Figure 13. Moreover, the ego-vehicle travels with the speed trajectory

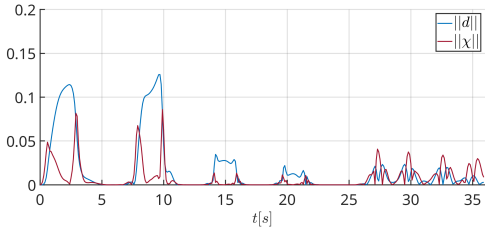

Figure 13: Tracking error states for the first test.

depicted in Figure 14, which denotes a smooth trajectory with comfortable acceleration and deceleration. Finally, the currently active FSM state and state transitions are demonstrated in Figure 15.

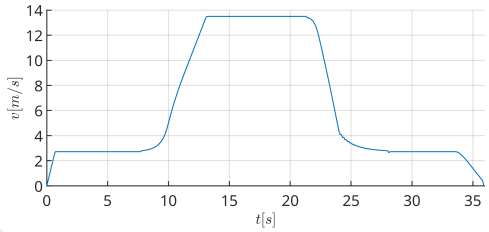

Figure 14: Speed trajectory for the first test.

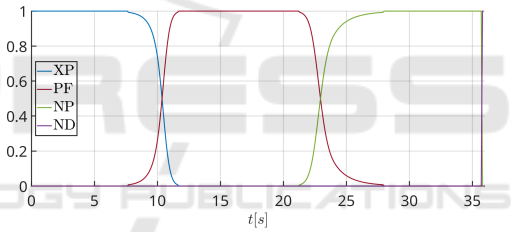

Figure 15: Active FSM states for the first test.

The second test focuses on evaluating the system performance with multiple road users, while the ego-vehicle is traversing a straight line trajectory. Moreover, this scenario directly starts in the PF state for simplification. Initially, the ego-vehicle is traveling with $v = 13.5[m/s]$, and we have two users in the scenario: one in-lane at $s_{rel,o1} = 41[m]$, and the other in the left lane at $s_{rel,o2} = 36[m]$, such that they both travel with $v = 12[m/s]$. In the first few seconds, the ego-vehicle continues to travel with its initial speed as $s_{rel} > s_{SF}$, but as it approaches the in-lane vehicle, it decelerates to match its velocity of $12[m/s]$ as illustrated in Figure 16.

At $t = 16[s]$, the vehicle in the left lane performs a Cut-In maneuver to the driving lane, and the ego-vehicle decelerates to maintain the safety distance s_{SF}, then accelerates afterwards to match the speed v_{o2} of the new in-lane leader. Around $t = 34[s]$, this user starts to decelerate and the ego-vehicle decreases its speed to match v_{o2} until the user performs a Cut-Out maneuver around $t = 40[s]$, at which the ego-vehicle

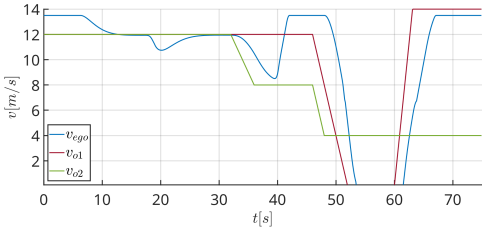

Figure 16: Speed trajectories of all road users in the second test.

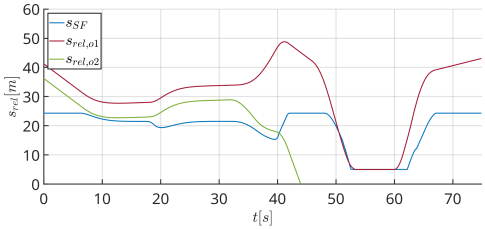

Figure 17: Safety distance s_{SF} is kept to the leading in-lane road user during the second test.

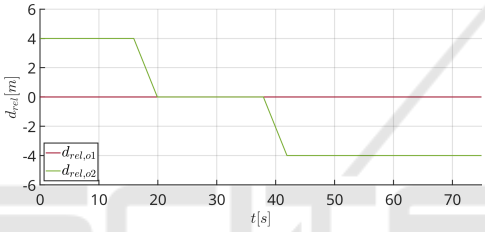

Figure 18: d_{rel} of other road users in the second test.

accelerates till it reaches the maximum allowed speed. Finally, we test a Closing maneuver as the ego-vehicle recognizes a stationary object in-lane at $t = 48[s]$ and appropriately decelerates until it reaches a complete stop at $t = 54[s]$. The ego-vehicle remains stationary until the leading in-lane road user has passed the relaxed safety distance $s_{rel} >= s_{SF} + s_{SF,\eta}$, after which it starts moving again and accelerates appropriately until it reaches its maximum allowed speed, thus concluding our numerical experiments.

5 CONCLUSION AND FUTURE WORK

In this work, we introduced a FSM architectural model for operating multiple multi-objective NMPC-based controllers in urban driving scenarios. We simulated our approach and successfully tested it against different scenarios, proving its effectiveness in generating smooth control trajectories, while minimizing the path tracking errors and the overall travel time. Moreover, we integrated a probabilistic approach to determine the closest in-lane leading road user, with which we enforced collision avoidance constraints to guarantee the ego-vehicle's safety.

For future work, we may investigate a more generic approach for tuning the weighing factors of the different objectives in the MPC multi-objective cost function, e.g., using machine learning. We may also extend the path tracking problem so as to enable overtaking of slower vehicles, for which new FSM states and transitions will be necessary. Finally, we may formulate our problem using a *Mixed-Integer Programming (MIP)* approach then validate it using the same test scenarios, such that we can comparatively evaluate the FSM and MIP methodologies in terms of robustness, computational complexity, and overall performance.

ACKNOWLEDGEMENTS

This research paper is funded by dtec.bw – Digitalization and Technology Research Center of the Bundeswehr as part of project MORE – Munich Mobility Research Campus.

REFERENCES

Bae, S.-H., Joo, S.-H., Pyo, J.-W., Yoon, J.-S., Lee, K., and Kuc, T.-Y. (2020). Finite state machine based vehicle system for autonomous driving in urban environments. In *2020 20th International Conference on Control, Automation and Systems (ICCAS)*, pages 1181–1186.

Bosch, R. (2003). *ACC Adaptive Cruise Control*. Bentley Pub.

Bouska, W. (2021). *StVO Straßenverkehrs-Ordnung*. C.F. Müller.

Britzelmeier, A. and Gerdts, M. (2020). A nonsmooth newton method for linear model-predictive control in tracking tasks for a mobile robot with obstacle avoidance. *IEEE Control Systems Letters*, PP:1–1.

Britzelmeier, A., Gerdts, M., and Rottmann, T. (2020). Control of interacting vehicles using model-predictive control, generalized nash equilibrium problems, and dynamic inversion. *IFAC-PapersOnLine*, 53(2):15146–15153. 21st IFAC World Congress.

Burger, M. and Gerdts, M. (2019). *DAE Aspects in Vehicle Dynamics and Mobile Robotics*, pages 37–80. Springer International Publishing, Cham.

Chen, C., Guo, J., Guo, C., Chen, C., Zhang, Y., and Wang, J. (2021). Adaptive cruise control for cut-in scenarios based on model predictive control algorithm. *Applied Sciences*, 11(11):5293.

Gerdts, M. (2018). Numerical experiments with multistep model-predictive control approaches and sensitivity updates for the tracking control of cars.

Grüne, L. and Pannek, J. (2011). *Nonlinear Model Predictive Control: Theory and Algorithms*. Springer London Ltd.

Gutjahr, B., Gröll, L., and Werling, M. (2017). Lateral vehicle trajectory optimization using constrained linear time-varying mpc. *IEEE Transactions on Intelligent Transportation Systems*, 18(6):1586–1595.

ISO Central Secretary (2018). Intelligent transport systems — Adaptive cruise control systems — Performance requirements and test procedures. Standard, International Organization for Standardization, Geneva, CH.

Klose, P. and Mester, R. (2018). Simulated autonomous driving in a realistic driving environment using deep reinforcement learning and a deterministic finite state machine. *CoRR*, abs/1811.07868.

Lu, M., Wevers, K., van der Heijden, R., and Heijer, T. (2004). Adas applications for improving traffic safety. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No.04CH37583)*, volume 4, pages 3995–4002 vol.4.

Luo, L., Liu, H., Li, P., and Wang, H. (2010). Model predictive control for adaptive cruise control with multi-objectives: Comfort, fuel-economy, safety and car-following. *Journal of Zhejiang University SCIENCE A*, 11:191–201.

Macedo, B. H. F., Araujo, G. F. P., Silva, G. S., Crestani, M. C., Galli, Y. B., and Ramos, G. N. (2015). Evolving finite-state machines controllers for the simulated car racing championship. In *2015 14th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, pages 160–172.

Studier, R. (2022). *Straßenverkehrs-Zulassungs-Ordnung StVZO*. Epubli.

Swaroop, D. and Rajagopal, K. (2001). A review of constant time headway policy for automatic vehicle following. In *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, pages 65–69.

Vu, T. M., Moezzi, R., Cyrus, J., and Hlava, J. (2021). Model predictive control for autonomous driving vehicles. *Electronics*, 10(21):2593.

Xiao, W., Mehdipour, N., Collin, A., Bin-Nun, A. Y., Frazzoli, E., Tebbens, R. J. D., and Belta, C. (2021). Rule-based evaluation and optimal control for autonomous driving. *CoRR*, abs/2107.07460.

Yurtsever, E., Lambert, J., Carballo, A., and Takeda, K. (2020). A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, 8:58443–58469.

Zhang, M., Li, N., Girard, A., and Kolmanovsky, I. (2017). A finite state machine based automated driving controller and its stochastic optimization. volume Volume 2: Mechatronics; Estimation and Identification; Uncertain Systems and Robustness; Path Planning and Motion Control; Tracking Control Systems; Multi-Agent and Networked Systems; Manufacturing; Intelligent Transportation and Vehicles; Sensors and Actuators; Diagnostics and Detection; Unmanned, Ground and Surface Robotics; Motion and Vibration Control Applications of *Dynamic Systems and Control Conference*.