2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 979-8-3503-7770-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/IROS58592.2024.10802822

Model Predictive Control for Frenet-Cartesian Trajectory Tracking of a Tricycle Kinematic Automated Guided Vehicle

Akash Subash[1, 2], Daniel Kloeser[2], Jonathan Frey[1, 3], Rudolf Reiter[1], Moritz Diehl[1, 3], Karsten Bohlmann[2]

***Abstract*— This work proposes an optimal control scheme for a trajectory-tracking Automated Guided Vehicle considering motion and collision constraints in a warehouse environment. We outline how the simpler obstacle avoidance constraints in the Cartesian Coordinate Frame (CCF) can be retained, while projecting the tricycle kinematics to the Frenet Coordinate Frame (FCF) for track progress. The Nonlinear Model Predictive Control (NMPC) scheme is subsequently implemented using acados and its real-time feasibility is demonstrated in simulation and aboard a test vehicle at a warehouse.**

I. Introduction

Real-time trajectory tracking of guided lanes for Automated Guided Vehicles (AGV) traditionally rely on PID controllers, which often limit them to fixed obstacle free routes throughout their lifetime. By introducing an analytic feed-forward model, such as within MPC, constraints can be formulated intuitively. We harness its suitability for trajectory tracking with a nonlinear spatial model reformulation while retaining Cartesian collision constraints, and motion dynamics on a test AGV.

Tuned with a data-driven model in Gazebo, the controllers' stability in the presence of underactuation and sensor noise is verified prior to the real-world deployment. We reconstruct the state with an observer followed by delay compensation to provide a delay-free model for the progress maximization NMPC formulations.

A. Outline

Section II describes candidate kinematic model formulations and Section III details the associated Optimal Control Problems (OCP). Section IV subsequently illustrates the simulation and warehouse test environment, along with metrics to measure the performance of the direct and lifted approaches, proposed in [1].

B. Related Work

Trajectory tracking for automotive applications with well-defined lanes has been vastly researched recently in the interest of autonomous driving. In this Section, we summarize the relevant literature that influenced this work, which involves modeling the kinematics and obstacles appropriately to achieve real-time deployment of the control scheme on the test AGV. Apart from introducing the tricycle kinematic model in the Cartesian frame, [2] presents global path-planning algorithms. The fundamentals for time optimal control established using a bicycle model in the Frenet frame with Sequential Quadratic Programming (SQP) algorithms was demonstrated in [3]. Further, the progress maximization objective reformulation introduced in [4], is applied to racing by explicitly modeling lateral acceleration in [5]. Improved parametrization of reference tracks to avoid singularities in the Frenet-Cartesian transforms is detailed in [6].

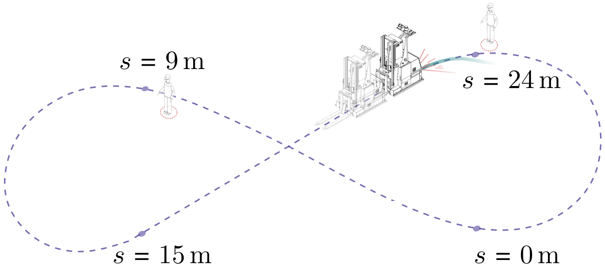

Fig. 1: Test track at ek robotics [7].

An optimization-based planning scheme with obstacle evasion known as the Time Elastic Band (TEB) in [8], demonstrated the feasibility of compute-intensive planners aboard AGVs, as indicated in [9]. This laid the groundwork for investigating the deployment of more rigorous collision avoidance techniques. TEB uses an optimization framework to penalize vehicle dynamics, and trajectory tracking, but limits to offline approximation of the multiple shooting multipliers.

By formulating the navigation problem as an elastic band attracted to way points and repelled by obstacles with an approximate kinematic model, these objectives are pitted against each other, demanding arduous tuning for solver convergence. MPC however, imposes hard constraints on dynamics, providing stronger guarantees of meeting the continuity conditions. This can be inferred from the fact that these constraints are indeed satisfied as part of the Karush-Kuhn-Tucker conditions at the solution [10], which are the first-order necessary conditions of candidate minima. The TEB is further extended to maintaining multiple candidate topologies with an approximated kinematic model for car-like vehicles in [11]. A potential field method, modeling objective terms representing repulsive fields promoting evasive maneuvers is introduced in [12] for autonomous road vehicles. Subsequently, the burden of tuning them compared to geometric obstacle constraints is discussed in [13].

Strict collision avoidance, promising safety in human-machine interaction unlike in the former approaches, can be better realized when formulated with hard constraints such as ellipses in [14], multiple covering circles in [15],

*This work was supported by ek robotics GmbH

[1] Department of Microsystems Engineering, University of Freiburg
[2] ek robotics GmbH, Germany
[3] Department of Mathematics, University of Freiburg

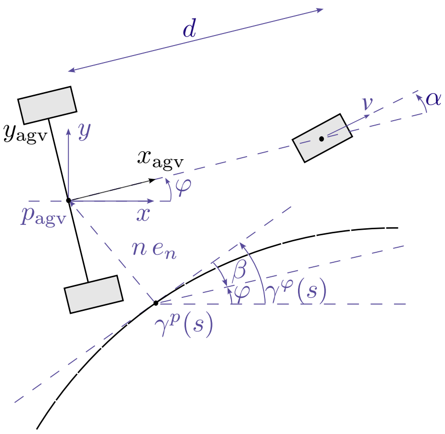

Fig. 2: Relation between Cartesian and Frenet frames for a reference curve $\gamma(s)$.

or hyperplanes in [16]. Collision constraints specified as road boundary deflection in [5] are extended with Ordinary Differential Equation (ODE) formulations in the CCF and the FCF in [1], motivated by retaining convex constraint specification while using the implicit curvilinear states of the FCF. Employing the native Differential Algebraic Equation (DAE) formulation stemming from this joint state representation, [13] uses linear MPC for trajectory tracking.

C. Contribution

The contribution of this work consists firstly of the index reduction for the lifted formulation for the tricycle kinematics, indicating its relevance to other models. Secondly, utilizing the real-time feasible MPC implementation with acados [17], we provide quantifiable performance metrics from the first real vehicle implementation of the direct and lifted controllers. To limit the bias in these benchmarks, we define a generic observer for state reconstruction in both controllers.

II. System Modelling

In this section, we outline various formulations of the tricycle kinematic model for motion control excluding highly dynamic maneuvers, and within the velocity bounds where wheel slippage can be safely ignored.

A. Tricycle kinematics

The Cartesian state vector of the AGV with tricycle kinematics described by $\zeta^c = [x,\, y,\, \varphi]^\top \in \mathbb{R}^3$ represents the position and heading in the CCF. It is further augmented by the vector $\zeta^u = [v,\, \alpha]^\top \in \mathbb{R}^2$ to constrain the rate of ζ^u, by defining the control input as $u = [a,\, \omega]^\top \in \mathbb{R}^2$. The control vector represents acceleration at the wheel a, and wheel turning rate ω which are the time derivatives of the wheel speed v and heading φ respectively. This vector u defined in the vehicle reference frame with axes x_{agv} and y_{agv} is projected to the global reference frame using the rotation matrix $M(\varphi)$ at the center of gravity marked by p_{agv} in Figure 2. Although the resulting extended state $\zeta^p = [x,\, y,\, \varphi,\, v,\, \alpha]^\top \in \mathbb{R}^5$ for pose tracking and control u vectors are continuous-time functions, this dependence is omitted further for notational convenience. The kinematics are defined in the CCF as

$$\dot{\zeta}^p(t) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \\ \dot{v} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} v\, \cos(\alpha)\, \cos(\varphi) \\ v\, \cos(\alpha)\, \sin(\varphi) \\ \dfrac{v}{d}\, \sin(\alpha) \\ a \\ \omega \end{bmatrix}. \tag{1}$$

For a more intuitive representation of track progress s and lateral displacement n along a reference curve γ with a heading angle difference β between the vehicle φ and tangent $\gamma^{\varphi}(s)$, we switch to the spatial reformulation in the FCF with $\kappa(s) = \dfrac{d\gamma^{\varphi}(s)}{ds}$ defined as the signed curvature. Here, the dynamic model using the state vector $\zeta^f = [s,\, n,\, \beta]^\top \in \mathbb{R}^3$ from [5] is adapted as $\zeta^{d^T} = [\zeta^{f^T},\, \zeta^{u^T}] \in \mathbb{R}^5$ to the tricycle kinematics

$$\dot{\zeta}^d(t) = \begin{bmatrix} \dot{s} \\ \dot{n} \\ \dot{\beta} \\ \dot{v} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} \dfrac{v\, \cos(\alpha)\, \cos(\beta)}{1 - n\, \kappa(s)} \\ v\, \cos(\alpha)\, \sin(\beta) \\ \dfrac{v}{d}\, \sin(\alpha) - \dfrac{\kappa(s)\, v\, \cos(\alpha)\, \cos(\beta)}{1 - n\, \kappa(s)} \\ a \\ \omega \end{bmatrix}. \tag{2}$$

From the reference curve $\gamma^p(s)^T = [\gamma^x(s)^T,\, \gamma^y(s)^T]$, we define the transforms between the coordinate frames as

$$s^* = \underset{s}{\text{minimize}} \quad \|p_{\text{agv}} - \gamma^p(s)\|_2^2 \tag{3a}$$

$$\zeta^f = \mathcal{F}_\gamma(\zeta^c) = \begin{bmatrix} s^* \\ e_n^T\, (p_{\text{agv}} - \gamma^p(s^*)) \\ \varphi - \gamma^{\varphi}(s^*) \end{bmatrix} \tag{3b}$$

$$\zeta^c = \mathcal{F}_\gamma^{-1}(\zeta^f) = \begin{bmatrix} \gamma^x(s) - n\, \sin(\gamma^{\varphi}(s)) \\ \gamma^y(s) + n\, \cos(\gamma^{\varphi}(s)) \\ \gamma^{\varphi}(s) + \beta \end{bmatrix}. \tag{3c}$$

We observe that the standard Cartesian tricycle kinematics ensue from index reducing the DAE $[\mathcal{F}_\gamma^{-1}(\zeta^f)^T,\, \zeta^{f^T},\, \zeta^{u^T}]$, similar to the single track model in [1] as

$$\begin{aligned} \dot{\zeta}^c &= \frac{d\mathcal{F}_\gamma^{-1}(\zeta^f)}{dt} && \text{(4a)} \\ &= \frac{\partial \mathcal{F}_\gamma^{-1}(\zeta^f)}{\partial \zeta^f}\, f(\zeta^f, u) && \text{(4b)} \\ &= \begin{bmatrix} v\, \cos(\alpha)\, \cos(\varphi) \\ v\, \cos(\alpha)\, \sin(\varphi) \\ \dfrac{v}{d}\, \sin(\alpha) \end{bmatrix}. && \text{(4c)} \end{aligned}$$

Subsequently, to profit from the advantages of both coordinate frames, i.e the FCF for trajectory tracking and the CCF for collision avoidance as shown in [1], we define an extended state vector $\zeta^{l^T} = [\zeta^{c^T},\, \zeta^{d^T}] \in \mathbb{R}^8$, leading to

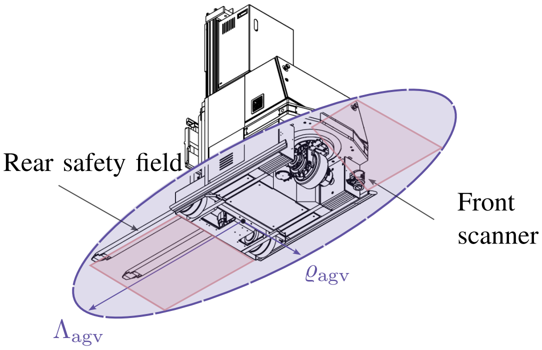

Fig. 3: Front-bottom view of the enlarged elliptical AGV footprint [7].

$$\dot{\zeta}^l(t) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\varphi} \\ \dot{s} \\ \dot{n} \\ \dot{\beta} \\ \dot{v} \\ \dot{\alpha} \end{bmatrix} = \begin{bmatrix} v\,\cos(\alpha)\,\cos(\varphi) \\ v\,\cos(\alpha)\,\sin(\varphi) \\ \dfrac{v}{d}\,\sin(\alpha) \\ \dfrac{v\,\cos(\alpha)\,\cos(\beta)}{1 - n\,\kappa(s)} \\ v\,\cos(\alpha)\,\sin(\beta) \\ \dfrac{v}{d}\,\sin(\alpha) - \dfrac{\kappa(s)\,v\,\cos(\alpha)\,\cos(\beta)}{1 - n\,\kappa(s)} \\ a \\ \omega \end{bmatrix}. \quad (5)$$

For the dual frame representation, we consider a Frenet primary frame with Cartesian algebraic states

$$\begin{aligned} \dot{\zeta}^d &= f(\zeta^d, u) \\ 0 &= \zeta^c - \mathcal{F}_\gamma^{-1}(\zeta^f). \end{aligned} \quad (6)$$

Motivated by real-time feasibility, we avoid the alternate bilevel problem as in [1], limiting to a single instance of the optimization problem (3a) within the observer for the state reconstruction, as illustrated in Figure 5.

B. Obstacle avoidance formulation

The intuitive Euclidean distance formulation between circular obstacles and an AGV circumscribing circle is unsuitable for this application due to the unnecessarily restricted solution space, which can be inferred from the vehicle dimensions in Table I. The rectangular footprint of the AGV better approximated by a bounding ellipse with axes Λ_{agv} , ϱ_{agv}, for $i \in [1,\, n_{ob}]$ circular obstacles of radius r_{ob}, and the difference between their centers $[\Delta x,\, \Delta y]$ for k shooting nodes, is implemented here by constraining the enlarged ellipse with axes λ , ρ as

$$\begin{bmatrix} \Delta x_{i,k} \\ \Delta y_{i,k} \end{bmatrix}^T \Sigma(\zeta_k) \begin{bmatrix} \Delta x_{i,k} \\ \Delta y_{i,k} \end{bmatrix} - 1 = 0. \quad (7a)$$

$$\Sigma(\zeta_k) = M(\varphi_k)^T \begin{bmatrix} 1/\lambda_i^2 & 0 \\ 0 & 1/\rho_i^2 \end{bmatrix} M(\varphi_k) \quad (7b)$$

$$\lambda_i = \Lambda_{\text{agv}} + r_{i,\text{ob}} \quad (7c)$$

$$\rho_i = \varrho_{\text{agv}} + r_{i,\text{ob}} \quad (7d)$$

Quantity	**Value**
track curvature κ	[-0.25, 0.25] m^{-1}
wheelbase d	1.03 m
AGV length l_{agv}	2.914 m
AGV breadth b_{agv}	1.115 m

TABLE I: Parameters of the kinematic model

To retain the narrow AGV footprint afforded by the AGV ellipse and yet reduce nonlinearity in the constraint formulation, we also consider AGV covering-circles [18], [19] of equal radii r_c to obtain the minimum number of circles $n_c = \left\lceil \dfrac{l_{\text{agv}}}{b_{\text{agv}}} \right\rceil = 3$ centered at

$$x_j = x + (\delta_{\text{cg}} + j\,\delta_{\text{c}})\cos(\varphi),\;\; j = -1, \ldots, n_c - 2 \quad (8a)$$

$$y_j = y + (\delta_{\text{cg}} + j\,\delta_{\text{c}})\sin(\varphi),\;\; j = -1, \ldots, n_c - 2. \quad (8b)$$

The $i \in [1,\, n_{ob}]$ obstacle ellipses are enlarged by the j AGV covering circles with centers offset by δ_c and δ_{cg}, to obtain an elliptical constraint

$$\begin{bmatrix} \Delta x_{i,j,k} \\ \Delta y_{i,j,k} \end{bmatrix}^T \Sigma \begin{bmatrix} \Delta x_{i,j,k} \\ \Delta y_{i,j,k} \end{bmatrix} - 1 = 0 \quad (9a)$$

$$\Sigma = M(\theta_i)^T \begin{bmatrix} 1/\lambda_i^2 & 0 \\ 0 & 1/\rho_i^2 \end{bmatrix} M(\theta_i). \quad (9b)$$

III. Trajectory tracking controller

The below Nonlinear Programs (NLP) overapproximate the physically motivated motion constraints imposed by the application module depicted in Figure 5, and satisfy geometric constraints for obstacle avoidance along the warehouse track; while tracking the trajectories specified by a reference path augmented with the temporal information

$$s_{k,\text{ref}} = s_0 + \frac{s_{\text{ref}}}{N}\,k \quad (10a)$$

$$v_{k,\text{ref}} = v_{\text{ref}}. \quad (10b)$$

We track a constant wheel velocity to simplify overapproximating the application layer's safety field indicated in red

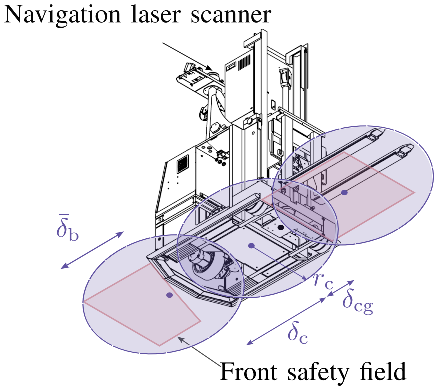

Fig. 4: Rear-bottom view of the AGV's covering circles depicting their lateral offsets [7].

in Figures 3, and 4, which grows dynamically as a function of v to account for safe braking distances.

A. Direct Elimination Frenet Optimal Control Problem

The direct elimination NLP with collision constraints specified as a nonlinear state mapping $\Gamma_{\mathrm{d}}(\zeta^d) = \mathcal{F}_\Upsilon^{-1}(\zeta^d)$ for an initial guess $\bar{\zeta}_0^d$ is defined as

$$\begin{aligned}
&\underset{u_0,\zeta_0^d,\dots,u_{N-1},\zeta_N^d}{\text{minimize}} \quad E\left(\zeta_N^d\right) + \sum_{k=0}^{N-1} L\left(\zeta_k^d, u_k\right) && \text{(11a)}\\
&\text{subject to} && \\
&\qquad \zeta_0^d - \bar{\zeta}_0^d \qquad\qquad = 0, && \text{(11b)}\\
&\qquad \zeta_{k+1}^d - \phi(\zeta_k^d, u_k) \quad = 0,\ k = 0,\dots,N-1, && \text{(11c)}\\
&\underline{u} \leq \quad u_k \qquad\qquad\quad \leq \overline{u},\ k = 0,\dots,N-1, && \text{(11d)}\\
&\underline{w} \leq \quad v_k\,\cos(\alpha_k) \qquad \leq \overline{w},\ k = 0,\dots,N, && \text{(11e)}\\
&\underline{\dot{\varphi}} \leq \quad \frac{v_k}{d}\,\sin(\alpha_k) \qquad \leq \overline{\dot{\varphi}},\ k = 0,\dots,N, && \text{(11f)}\\
&\qquad n_k\,\kappa(s) \qquad\qquad \leq 1,\ k = 0,\dots,N, && \text{(11g)}\\
&1 \leq \|\Gamma_{\mathrm{d}}(\zeta_k^d) - p_{i,\mathrm{ob}}\|_{\Sigma(\cdot)}^2, \qquad k = 0,\dots,N, && \text{(11h)}\\
&\qquad\qquad\qquad\qquad\qquad i = 0,\dots,n_{ob}-1. &&
\end{aligned}$$

With the cost weighted by the diagonal matrices Q, Q_N, R

$$\begin{aligned}
L\left(\zeta_k^d, u_k\right) &= \|\zeta_k^d - \zeta_{k,\mathrm{ref}}^d\|_Q^2 + \|u_k\|_R^2 && \text{(12a)}\\
E\left(\zeta_N^d\right) &= \|\zeta_N^d - \zeta_{N,\mathrm{ref}}^d\|_{Q_N}^2. && \text{(12b)}
\end{aligned}$$

Here, $(\zeta_0^d,\dots,\zeta_N^d)$, and $(u_0,\dots,u_{N-1})$ denote the discrete time state and control trajectories for the horizon length N respectively, and $\phi(\zeta_k^d, u_k)$, the discretized dynamics with a Runge-Kutta order 4 (RK4) integrator.

We choose suitable limits for the AGV speed w and its angular velocity $\dot{\varphi}$, as seen in (11e) and (11f). This is done to ensure the feasibility of the OCP solution, despite the MPC being agnostic to the underlying application safety bounds. Finally, while the obstacle avoidance constraints might demand straying from the center line, the constraint (11g) still needs to be fulfilled to guarantee the uniqueness of the projection onto the centerline. An indeterminate solution of (3c) could result in the inner high curvature arch while performing an evasive maneuver, if the AGV is driven to the center of the circle with radius κ^{-1}.

Quantity	Value
sampling time τ_s	$6 \cdot 10^{-2}\,\mathrm{s}$
measurement delay τ_m	$6 \cdot 10^{-2}\,\mathrm{s}$
actuation delay τ_a	$6 \cdot 10^{-2}\,\mathrm{s}$
reference progress s_{ref}	$8\ \mathrm{m}$
reference wheel speed v_{ref}	$8 \cdot 10^{-1}\,\mathrm{m\,s^{-1}}$
state weights Q	$\mathrm{diag}(\,[10^{-8}, 25,\ 10^{-8},\ 10^2,\ 10]\,)$
terminal state weights Q_N	$\mathrm{diag}(\,[10^{-1}, 25,\ 10^{-8},\ 10^{-8},\ 5]\,)$
control weights R	$\mathrm{diag}(\,[5,\ 25]\,)$
horizon length N	100

TABLE II: Parameters for NMPC formulation. Units in SI units, if not specified explicitly.

B. Lifted Frenet Optimal Control Problem

Utilizing the index-reduced state ζ^l to define the obstacle avoidance constraints in the Cartesian frame and the cost function (12), we arrive at the mathematically equivalent lifted formulation

$$\begin{aligned}
&\underset{u_0,\zeta_0^l,\dots,u_{N-1},\zeta_N^l}{\text{minimize}} \quad E\left(\zeta_N^d\right) + \sum_{k=0}^{N-1} L\left(\zeta_k^d, u_k\right) && \text{(13a)}\\
&\text{subject to} && \\
&\qquad \zeta_0^l - \bar{\zeta}_0^l \qquad\qquad = 0, && \text{(13b)}\\
&\qquad \zeta_{k+1}^l - \phi(\zeta_k^l, u_k) \quad = 0,\ k = 0,\dots,N-1, && \text{(13c)}\\
&\underline{u} \leq \quad u_k \qquad\qquad\quad \leq \overline{u},\ k = 0,\dots,N-1, && \text{(13d)}\\
&\underline{w} \leq \quad v_k\,\cos(\alpha_k) \qquad \leq \overline{w},\ k = 0,\dots,N, && \text{(13e)}\\
&\underline{\dot{\varphi}} \leq \quad \frac{v_k}{d}\,\sin(\alpha_k) \qquad \leq \overline{\dot{\varphi}},\ k = 0,\dots,N, && \text{(13f)}\\
&\qquad n_k\,\kappa(s) \qquad\qquad \leq 1,\ k = 0,\dots,N, && \text{(13g)}\\
&1 \leq \|\Gamma_{\mathrm{l}}(\zeta_k^l) - p_{i,\mathrm{ob}}\|_{\Sigma(\cdot)}^2, \qquad k = 0,\dots,N, && \text{(13h)}\\
&\qquad\qquad\qquad\qquad\qquad i = 0,\dots,n_{ob}-1. &&
\end{aligned}$$

The obstacle avoidance formulations in (7a), (9a), define $\Sigma(\cdot)$ either as $\Sigma(\zeta_k)$ for an AGV bounding ellipse with circular obstacles, or Σ for AGV covering circles with elliptical obstacles. These constraints are imposed via the non-linear $\Gamma_{\mathrm{d}}(\zeta^d)$, and linear $\Gamma_{\mathrm{l}}(\zeta^l)$ state mappings within the direct (11), and lifted (13) controllers respectively. These collision constraints are relaxed with slack variables with a quadratic penalization to guarantee feasibility.

C. Feedback loop

The state $\hat{\zeta}_1^p$ estimate from the AGV's localization and encoder modules are encapsulated in the application module in Figure 5. Since the OCP additionally requires an estimate of the Frenet state vector ζ^f to solve the OCPs (11) and (13), we utilize an observer to solve the optimization problem in (3b) outlined in Figure 5. Since this feedback mechanism constrains the state drift of the index-reduced ODE solution in III-B, the required equality (14) is trivially satisfied.

$$\zeta_0^f - \mathcal{F}_\gamma(\zeta_0^c) = 0 \qquad (14)$$

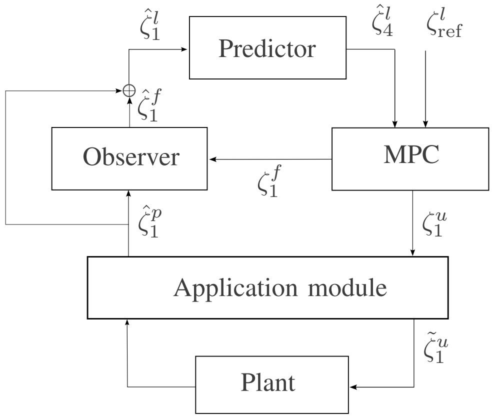

Fig. 5: Closed loop architecture for the lifted Frenet formulation using discrete time indices k for states ζ_k

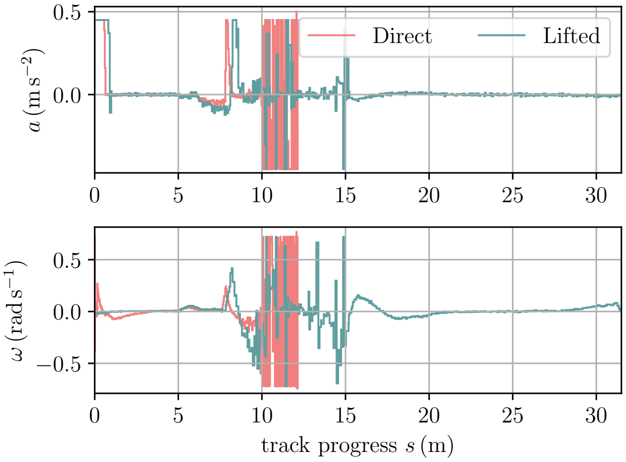

Fig. 6: Control u comparison for the controllers using the AGV bounding ellipse constraint in Gazebo.

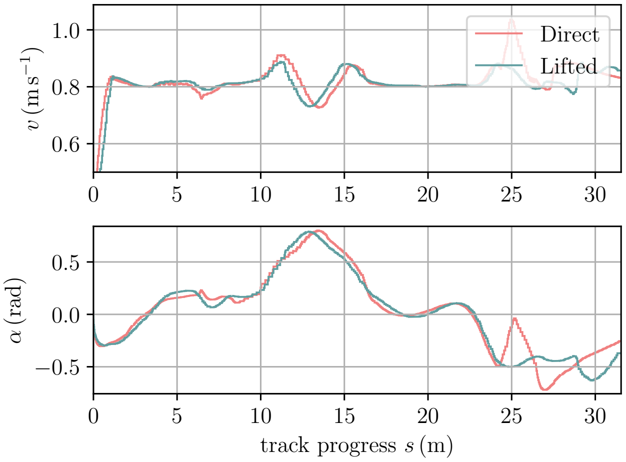

Fig. 7: State ζ^u comparison with the AGV covering circles, illustrating reference velocity tracking in Gazebo.

To realistically handle latency on the real AGV, we identify the dominating delays due to the expensive nonlinear MPC node and communication networks detailed in Table II. The inherent challenge of measuring one-way latency is overcome by considering the round trip delay observed experimentally in Controller Area Network (CAN) bus. Assuming that it equally influences the actuation and measurement paths as τ_a, and τ_m respectively, the predictor in the feedback path compensates these delays to retain optimality.

IV. Experiments

In this section, we evaluate the direct and lifted controllers first in the Gazebo simulation framework, followed by a real AGV in the warehouse. Despite successful tests on several tracks with varying number of obstacles, we limit the following discussion only to the configuration indicated in Figure 1 due to the challenge at $s = 24\,\mathrm{m}$, where the collision and projection uniqueness constraints are active simultaneously. In Gazebo, we identify the required delay compensation scheme in the predictor for a round trip delay $\tau_{\mathrm{rt}} = \tau_{\mathrm{s}} + \tau_{\mathrm{a}} + \tau_{\mathrm{m}}$, the appropriate trajectory references, and also the more advantageous covering circles constraint formulation. We characterize the performance for the experiment duration $[0, M \cdot \tau_s]$ by introducing the centerline deviation metric using the Frenet state n using

$$\varepsilon_{n,\mathrm{avg}} = \frac{1}{M}\sum_{l=0}^{M} |n_l|. \tag{15}$$

Subsequently, we utilize an oscillation metric that approximates the finite difference second-order derivative of an arbitrary quantity z as

$$\varepsilon_{z,\mathrm{osc}} = \sqrt{\frac{1}{M-2}\sum_{l=0}^{M-2}(z_{l+1} - 2z_l + z_{l-1})^2}. \tag{16}$$

A. Simulation

The physics engine in Gazebo is used to test the controller with a high-fidelity model before moving to the test vehicle. The accuracy of the data driven measurement noise models in the simulated plant is indicated by the similarity of the metrics between simulation and reality in Table III. In this environment, a realistic AGV footprint can be factored in along with the forces of gravity, friction and other contact forces. This environment is interfaced with the ROS Noetic framework, for which the MPC is developed as an independent node using the acados cython interface. Within this node, the previously formulated NLPs are solved with the QP solver HPIPM [20] using a condensing horizon of $N/10$, and the Real Time Iteration (RTI) [21] scheme to ensure real-time feasibility. On inspecting Figure 6 and Table IV, the oscillatory controls indicate poor solution convergence for the severely nonlinear elliptical footprint constraint, as anticipated from the arguments in Section II-B. Since the solution stagnates at the poor local optima in the direct elimination scheme at $s = 12.5\,\mathrm{m}$, the additionally nonlinear elliptical footprint highlights the superiority of the lifted formulation. However, since both, the direct and lifted controllers complete the track with the covering circle constraint as seen in Figure 7 with lower oscillation in the controls highlighted in Table III, we consider them for the real AGV tests in the next section.

	Direct	**Lifted**	**Simulation**
$\varepsilon_{n,\,\mathrm{avg}}\,(\mathrm{m})$	$2.24 \cdot 10^{-1}$	$2.35 \cdot 10^{-1}$	✓
$\varepsilon_{a,\,\mathrm{osc}}\,(\mathrm{m\,s^{-2}})$	$1.58 \cdot 10^{-2}$	$1.56 \cdot 10^{-2}$	✓
$\varepsilon_{\omega,\,\mathrm{osc}}\,(\mathrm{rad\,s^{-1}})$	$1.57 \cdot 10^{-2}$	$2.25 \cdot 10^{-2}$	✓
$\varepsilon_{n,\,\mathrm{avg}}\,(\mathrm{m})$	$3.04 \cdot 10^{-1}$	$2.87 \cdot 10^{-1}$	
$\varepsilon_{a,\,\mathrm{osc}}\,(\mathrm{m\,s^{-1}})$	$1.97 \cdot 10^{-2}$	$8.15 \cdot 10^{-2}$	
$\varepsilon_{\omega,\,\mathrm{osc}}\,(\mathrm{rad\,s^{-1}})$	$1.70 \cdot 10^{-1}$	$2.53 \cdot 10^{-1}$	

TABLE III: Centerline and oscillation metrics with AGV covering circles in Gazebo and in reality.

	Direct	**Lifted**
$\varepsilon_{n,\,\mathrm{avg}}(\mathrm{m})$	$6.30 \cdot 10^{-1}$	$4.17 \cdot 10^{-1}$
$\varepsilon_{a,\,\mathrm{osc}}(\mathrm{m\,s^{-2}})$	$7.79 \cdot 10^{-1}$	$1.88 \cdot 10^{-1}$
$\varepsilon_{\omega,\,\mathrm{osc}}(\mathrm{rad\,s^{-1}})$	1.14	$3.52 \cdot 10^{-1}$

TABLE IV: Centerline and oscillation metrics with the AGV bounding ellipse in Gazebo.

B. Warehouse tests

Both formulations are tested with circular and rectangular obstacle footprints, detected with the ROS package from [22]. The closed-loop state trajectories of the directed and lifted formulation are presented in Figure 8.

We continue to use the RTI scheme to solve the OCPs aboard the AGV equipped with an Intel i7 9700TE @ 1.8 Ghz running ROS Noetic. The average timings in Table V show the controls computed within a fourth of the sampling time, indicating real-time feasibility on modern embedded platforms. The MPC node encapsulates the times for reference and obstacle parameter updates, and the RTI step in acados. The computationally inexpensive prediction achieved with the acados RK4 integrator, is overshadowed by the significant overhead for state reconstruction in the observer. Additionally, the majority of the computation time spent in the interface can be reduced with a leaner implementation. Despite the larger state space of the lifted formulation, the corresponding algorithm has lower average runtime indicating better convergence properties.

V. Conclusion

We evaluated two controllers for trajectory tracking aboard a real tricycle kinematic AGV and experimentally verified better convergence behavior for the nonlinear optimization solver in the lifted formulation. We further validated the superior performance of the lifted formulation for severely nonlinear elliptical constraints, highlighted by the direct formulation's failure to complete the track in simulation.

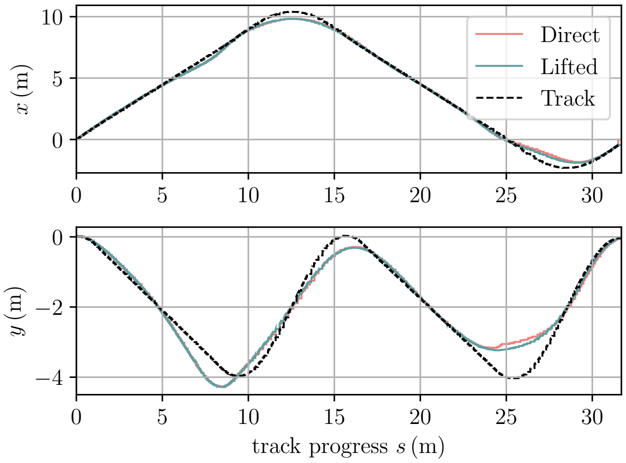

Fig. 8: State ζ^c comparison for the AGV covering circles constraint indicating obstacle avoidance during reference tracking on the real AGV.

	Direct	**Lifted**
interface	6.07 ms	6.32 ms
mpc	4.50 ms	4.45 ms
predictor	0.32 ms	0.40 ms
observer	2.64 ms	2.65 ms

TABLE V: Averaged MPC timings on the real AGV.

References

[1] R. Reiter, A. Nurkanović, J. Frey, and M. Diehl, "Frenet-Cartesian model representations for automotive obstacle avoidance within nonlinear MPC," *European Journal of Control*, 2023.

[2] M. Ljubi, G. Klančar, and A. Zdešar, "Path Planning of Multiple Automatic Guided Vehicles with Tricycle Kinematics Considering Priorities and Occupancy Time Windows," *Intelligent Autonomous Systems 17*, 2023.

[3] 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," *53rd IEEE CDC*, 2014.

[4] M. Werling, L. Gröll, and G. Bretthauer, "Invariant Trajectory Tracking With a Full-Size Autonomous Road Vehicle," *IEEE Transactions on Robotics*, 2010.

[5] D. Kloeser, T. Schoels, T. Sartor, A. Zanelli, G. Prison, and M. Diehl, "NMPC for Racing Using a Singularity-Free Path-Parametric Model with Obstacle Avoidance," *IFAC*, 2020.

[6] R. Reiter and M. Diehl, "Parameterization Approach of the Frenet Transformation for Model Predictive Control of Autonomous Vehicles," *IEEE ECC*, 2021.

[7] Malitzky, Markus and ek robotics GmbH, "AGV mechanical specification," 2023. *Restricted access.

[8] C. Rösmann, W. Feiten, T. Wösch, F. Hoffmann, and T. Bertram, "Efficient trajectory optimization using a sparse model," in *IEEE ECMR*, 2013.

[9] J. Wu, X. Ma, T. Peng, and H. Wang, "An Improved TEB Algorithm of AGV in Complex Environment," *Sensors*, 2021.

[10] Rawlings, James, Mayne, David, and Diehl, Moritz, "Model Predictive Control: Theory, Computation, and Design," 2020.

[11] C. Rösmann, F. Hoffmann, and T. Bertram, "Integrated online trajectory planning and optimization in distinctive topologies," *Robotics and Autonomous Systems*, 2017.

[12] H. Jiang, Z. Wang, Q. Chen, and J. Zhu, "Obstacle avoidance of autonomous vehicles with CQP-based model predictive control," *IEEE SMC*, 2016.

[13] X. Xing, B. Zhao, C. Han, D. Ren, and H. Xia, "Vehicle Motion Planning With Joint Cartesian-Frenét MPC," *IEEE Robotics and Automation Letters*, 2022.

[14] B. Brito, B. Floor, L. Ferranti, and J. Alonso-Mora, "Model Predictive Contouring Control for Collision Avoidance in Unstructured Dynamic Environments," 2020.

[15] S. I. Galiev and A. V. Khorkov, "Optimization of the Number and Arrangement of Circles of Two Radii for Forming a k-Covering of a Bounded Set," *Computational Mathematics and Mathematical Physics*, 2019.

[16] S. Brossette and P.-B. Wieber, "Collision avoidance based on separating planes for feet trajectory generation," *17th Int. Conf. Humanoid Robotics (IEEE-RAS)*, 2017.

[17] R. Verschueren et al., "acados: a modular open-source framework for fast embedded optimal control," 2020.

[18] A. V. Khorkov and S. I. Galiev, "Optimization of a k-covering of a bounded set with circles of two given radii," *Open Computer Science*, 2021.

[19] A. Heppes and H. Melissen, "Covering a Rectangle With Equal Circles," *Periodica Mathematica Hungarica*, 1997.

[20] G. Frison and M. Diehl, "HPIPM: a high-performance quadratic programming framework for model predictive control," *IFAC*, 2020.

[21] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, "From linear to nonlinear MPC: bridging the gap via the real-time iteration," *International Journal of Control*, 2020.

[22] D. Habermann and C. Garcia, "Obstacle Detection and Tracking Using Laser 2D," *IEEE LARS*, 2010.