Data-Driven Identification Based Model Predictive Control for a Ground Robot

Data-Driven Identification based Model Predictive Control for a Ground Robot

Fabio Faliero
Mechanical and Aerospace Engineering
Politecnico di Torino
Torino, Italy
fabio.faliero@polito.it

Enza Incoronata Trombetta
Mechanical and Aerospace Engineering
Politecnico di Torino
Torino, Italy
enza.trombetta@polito.it

Elisa Capello
Mechanical and Aerospace Engineering
Politecnico di Torino
Torino, Italy
elisa.capello@polito.it

***Abstract*—*Unmanned Ground Robots* (UGVs) are commonly used test beds for the design and validation of Guidance, Navigation and Control algorithms. Thanks to the onboard sensors, it is possible to extract information about the system state, including position, velocity or orientation. Additionally, data can be stored and examined for various purposes, such as the creation of a model for the system. Predictions about the system state can be exploited by a model-based predictive controller to decide the best control action to be carried out at the current time. This paper proposes a Tracking Linear Quadratic Model Predictive Controller (LQMPC) for a UGV robot, together with a Kalman Filter (KF) estimator for state retrieval. The system model is identified with a mixed approach, combining a kinematic model with a data driven model whose data have been collected during experiments with the real platform. The effectiveness of the data driven approach for MPC is then proven for a velocity and orientation tracking application in a simulative scenario.**

Keywords*—*System Identification, MPC, Autonomous Systems

I. Introduction

Autonomous operating systems are widely used in different engineering and technological fields such as industrial automation, healthcare, agriculture and space exploration. To operate as autonomous, they need to contain on board all the necessary elements for data acquisition, processing and management. Therefore, the choice of sensors, computing and storage units, payloads and actuators, must be conducted very carefully to comply with system and operational constraints. Unmanned ground vehicles (UGV) design is generally freer than other unmanned systems design, such as Unmanned Aerial Vehicles (UAVs), given the less restrictive requirements about weight, size and battery package dimension. This makes them excellent test-bed for experimenting with Guidance, Navigation and Control algorithms (GNC). The sensor set typically includes encoders, Inertial Measurement Units (IMUs), magnetometers, GPS, or alternative navigation sources such as Cameras or LiDARs. By the processing of sensor data, the system can determine its state and make inferences about the surrounding environment. The obtained information is typically exploited by a Guidance and Control System responsible for generating and tracking a reference.

In the broad spectrum of control strategies, Model Predictive Control (MPC) is a highly regarded and widely used approach due to its versatility. It enables the design of a finite horizon optimization problem that meets desired performance metrics and is subject to physical and performance constraints. Its ability to make predictions based on current system states and to optimize control in the well-established finite control horizon strategies makes it particularly suitable for driving autonomous vehicles in complex environments. MPC has found wide application in various autonomous tasks, from self-driving cars or assistive driving tasks as proposed in [1], [2] to UAVs [3] performing precision manoeuvres in the airspace. As observed in [4], the MPC strategy fits very well with the UGVs framework, given the multiple operational and safety system requirements. In the same work, it is reported a review of the design of MPCs for autonomous ground vehicles (AGVs) focusing on both individual and distributed systems. The flexibility of the predictive control algorithm makes it well suited for successful tracking of externally generated references, obtained for example from motion planner algorithms based on Artificial Potential Fields (APF) Guidance [5].

The design of an MPC is based on a comprehensive understanding of the system it is intended to control, which includes the state-space realization of the system and real-time feedback of the current states. Typically, this feedback is acquired through a State Estimation process, which involves deriving the current state of the system from measurable data. Some physical insights are exploited for the model definition, but for complex systems operating in unknown environments, data-driven models turn out to be very effective in system behaviour prediction.

In [6], a linear black box parametrization with an Auto-Regressive Moving Average with eXogenous input (ARMAX) model is obtained for a small tracked UGV. The dataset for training and validation is created by collecting the Pulse Width Modulated (PWM) signals coming from the Control System and the angular speed of the DC motors recorded by encoders. The obtained data-driven based model has shown itself to be very representative of the robot behaviour in the simulated environment, letting GNC algorithms to be tested and tuned effectively before their deployment on hardware.

This paper obtains a State-Space model for a tracked robotic system using Data-Driven Identification techniques. Contrary to [6], the identified model serves a broader purpose beyond simulation alone. It is used as a plant model for the devel-

opment of the MPC algorithm together with its associated Kalman Filter (KF) estimator.
The remainder of the paper is structured as follows. In Section II, the system description and the data-driven identification process are introduced. In Section III, the Control algorithm for a linear time-invariant system is reported and discussed. The experimental process through which the model is obtained is presented in Section IV. Next, the data-driven MPC and its associated KF are applied to a reference tracking mission. Section V contains results and considerations.

II. System Identification

System Modeling and identification are fundamental steps in a Model-Based Controller design. A general state-space representation for the system can be written as:

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t + w_t \\ y_t &= Cx_t + Du_t + \nu_t \end{aligned} \tag{1}$$

where $x_t \in \mathbb{R}^n$ is the state vector, $u_t \in \mathbb{R}^m$ the input vector, $y_t \in \mathbb{R}^p$ the output vector and t the time index. The state and output noise are indicated as w_t and ν_t respectively and assumed to be stationary.
The objective of identification is the definition of the Transition matrix $A \in \mathbb{R}^{n\times n}$, the Control matrix $B \in \mathbb{R}^{n\times m}$, the output matrices $C \in \mathbb{R}^{p\times n}$ and the matrix $D \in \mathbb{R}^{p\times m}$ in the System (1) thought the elaboration of input-output data coming from experimental tests.

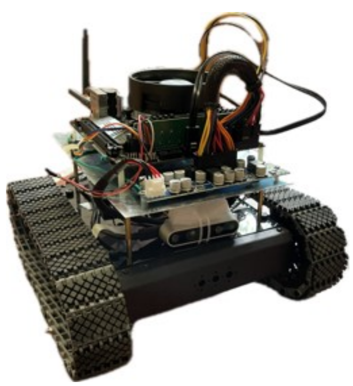

Fig. 1. Lynx Robot

A. Robot Model

The mathematical model being discussed is based on the Lynx ground robot from the STREAM Robotics Laboratory at the Department of Mechanical and Aerospace Engineering (DIMEAS) at the Politecnico di Torino (Figure 1). The Lynx Robot is a tracked UGV with two sprockets by each side, each of them is powered by a DC motor. The centre of Gravity (CoG) is assumed to be in the symmetry centre of the vehicle itself where the Body Frame ($\mathcal{F}_B : \{x, y, \theta\}$) is fixed with x axis representing the longitudinal axis and y the lateral one (Figure 2).
The Kinematic model of the UGV can be trivially derived under some assumptions on the system: (i) the motor characteristics are identical on each side and the mechanical coupling of the sprockets to each track is ideally without leakage so that the total torque transmitted to each track is the sum of each side motors contribution and (ii) the longitudinal and lateral slip can be assumed negligible. According to statement (i) the system can be modelled as a Differential Drive Robot (DDR) with the equivalent left and right wheels centred along the lateral axis y and the corresponding radius equal to the sprocket one.

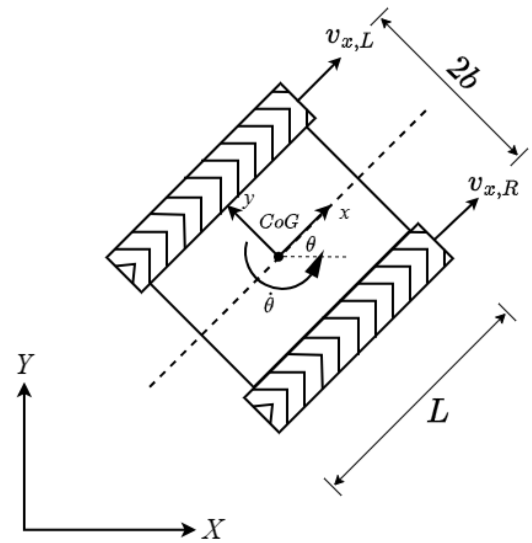

Fig. 2. Lynx Robot Kinematic model scheme

Hence, the DDR Kinematic equations can be used to represent the relation between the angular speed of the wheels, the forward velocity V_x and rotational velocity $\dot{\theta}$ in $\mathcal{F}_B : \{x, y, \theta\}$:

$$V_x = \frac{v_{x,L} + v_{x,R}}{2} \tag{2a}$$

$$\dot{\theta} = \frac{v_{x,R} - v_{x,L}}{2b} \tag{2b}$$

$$v_{x,L} = r\omega_L \tag{2c}$$

$$v_{x,R} = r\omega_R \tag{2d}$$

where r is the wheel radius, $2b$ is the distance between tracks and ω_L and ω_R in Equation (2c) and (2d) are the wheel angular speeds of the left and right side tracks.
PWM signals are used as Control signals for the system, while feedback about the angular speed of the wheels moving the tracks is obtained by two Quadrature Encoders connected to the DC motors. The relationship between the PWM signals and the angular speed of the wheels is required to directly relate the control inputs to the kinematic states of the robot. A data-driven approach is selected for its determination. As a consequence, the resulting model exhibits characteristics of a mixed identification. On one hand, a black-box identification for the PWM signals and the $\omega_{L/R}$ angular velocity. On the other hand, a kinematic relation for the angular velocity $\omega_{L/R}$ and the longitudinal and rotational speed V_x and $\dot{\theta}$.

Although assumption (ii) holds a negligible track slippage, it represents a phenomenon of great relevance in modelling the ground robot. However, the overall model is assumed to be adequate for reproducing the robot's dynamics for the study's purposes, as a significant portion of this dynamic is inherently embedded within the data-driven model.

B. Data Driven Identification

A mathematical relationship between PWM signals assumed as system inputs and the angular speed assumed as system output is obtained by exploiting a training dataset of input-output recorded data. The identification process can be summed up by the following steps [7]:

- *Data collection*: creation of training and validation dataset. Experiment design must be carefully conducted to acquire informative data.
- *Candidate model structure selection*: linear and nonlinear models are available. Here a state-space representation is selected.
- *Model fitting*: definition of the cost function to be minimized to obtain the model parameters.
- *Model validation*: definition of criteria for model quality assessment, such as the FIT Percent value, the Mean Squared Error (MSE) or the Final Prediction Error (FPE) index.

As reported in [8], the state-space representation has several advantages over the other input-output structures, like the ease of identifying Multi Input-Multi Output (MIMO) relationships and the possibility of introducing artificial variables to break up higher-order equations.
The identification problem can be formulated as follows:

Problem 1: Given a set of N observations of input u and output y data pairs, identify the state-space matrices A, B, C, D and the initial state x_0 for the system.
The subspace identification algorithms, and in particular the Numerical Algorithms for State Space Subspace System Identification (N4SID) [9], are used to estimate states and model parameters.

III. System Control

MPC is a discrete-time model-based feedback control strategy that uses a simplified model of the system to predict its future state and consequently choose the optimal control action to be applied [10], [11]. The goal is to find the best control action over a finite prediction horizon. At each sample time, the MPC gets the new measurements of the current state and it solves the optimisation problem to find the optimal input sequence concerning an objective function. A Kalman Filter [12] is proposed to retrieve all the states necessary for the Controller.
A standard application for MPC concerns the tracking of a reference, such as a desired forward speed or a specific direction. The tracking problem is formulated as follows:

Problem 2: Develop a Model Predictive Control to make a certain output vector $y(t) = Cx(t)$ track a reference signal $r(t) \in \mathbb{R}^p$ under control constraints expressed as

$$u_{min} \leq u(t) \leq u_{max}$$

A. Tracking Model Predictive Controller

The tracking Linear Quadratic MPC (LQMPC) is based on a discrete, linear and time-invariant prediction model expressed as follows:

$$\begin{cases} x_{t+k+1} &= Ax_{t+k} + Bu_{t+k} \\ y_{t+k} &= Cx_{t+k} \end{cases} \tag{3}$$

whose evolution starting from the time t is described by the relationship

$$x_k = A^k x_0 + \sum_{i=0}^{k-1} A^i B u_{k-1-i} \tag{4}$$

where x_0 is the state at time $t = 0$. For the sake of notation simplicity, the t index is omitted from here on.
The performance index is given by:

$$J(u^*, x_0) = x_N^T P x_N + \sum_{k=0}^{N_c-1} [x_k^T Q x_k + u_k^T R u_k] \tag{5}$$

$$\begin{aligned} \text{s.t.} \quad & y_{min} \leq y_k \leq y_{max}, \quad k = 1, ..., N_c, \\ & u_{min} \leq u_k \leq u_{max}, \quad k = 0, ..., N_u, \\ & x_{k+1} = Ax_k + Bu_k, \quad k \geq 0, \\ & y_k = Cx_k, \quad k \geq 0 \end{aligned}$$

where x_k is the predicted state vector at time $t+k$, obtained by applying the input optimized vector $u^* = [u_0, ..., u_{N_u-1}]^T \in \mathbb{R}^{m \times N_u}$ to Eq. (3), starting from x_0. N_c, N_u are the constraint and the control horizons, $Q > 0 \in \mathbb{R}^{n \times n}$ and $R > 0 \in \mathbb{R}^{m \times m}$ are the weighting matrices, and $P > 0 \in \mathbb{R}^{p \times p}$ the solution of the Riccati equation. The goal is to find the sequence of $u^* = [u_0, ..., u_{N_u-1}]^T$ that minimizes the index J.
By substituting Eq. (4) in Eq. (5), the following condensed form of the MPC optimization problem is obtained:

$$\begin{aligned} \min \quad & \frac{1}{2} u^T H u + x_0^T F^T u \\ \text{s.t.} \quad & Gu \leq W + Sx_0 \end{aligned} \tag{6}$$

where H, F, G, W, S depend on weights Q, R, P, upper and lower bounds y_{min}, y_{max}, u_{min}, u_{max}, and model matrices A, B, C. The optimization problem in Eq. (6) is a Quadratic Programming (QP) problem, characterized by a quadratic objective subjected to linear constraints, for both input and states.
The basic MPC setup can be extended to a tracking problem, where the objective is to make the output $y(t) = Cx(t) \in \mathbb{R}^p$ track a reference signal $r(t) \in \mathbb{R}^p$. The key point of this method is the use of the input increments Δu_k [13]:

$$\Delta u_k = u_k - u_{k-1} \tag{7}$$

where Δu_k is the new control input, whilst the value of control of the previous time step u_{k-1} becomes the new state variable. This new state variable $x_k^u = u_{k-1}$ is included in the augmented state space system, expressed as follows:

$$\begin{cases} x_{k+1} = Ax_k + Bu_{k-1} + B\Delta u_k \\ x_{k+1}^u = x_k^u + \Delta u_k \end{cases}$$

Hence, the augmented system is given by:

$$\begin{cases} \begin{bmatrix} x_{k+1} \\ x_{k+1}^u \end{bmatrix} = \begin{bmatrix} A & B \\ 0 & I \end{bmatrix} \begin{bmatrix} x_k \\ x_k^u \end{bmatrix} \\ \qquad\qquad + \begin{bmatrix} B \\ I \end{bmatrix} \Delta u_k \\ y_k = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x_k \\ x_k^u \end{bmatrix} \end{cases} \tag{8}$$

For the tracking problem, the cost function to minimize is given by:

$$\begin{aligned} J = \ & [y_{N_y} - r_{N_y}]^T P [y_{N_y} - r_{N_y}] \\ & + \sum_{k=0}^{N_y-1} \{ [y_k - r_k]^T Q \, [y_k - r_k] + \Delta u_k^T R \Delta u_k \} \end{aligned} \tag{9}$$

Thus, the optimisation problem can now be rewritten, using the explicit form of the cost in Equation (9), as :

$$\begin{aligned} \min \quad & \frac{1}{2} \Delta u^T H \Delta u + \begin{bmatrix} x_0 \\ x_0^u \\ r \end{bmatrix}^T F^T \Delta u \\ \text{s.t.} \quad & G\Delta u \leq W + Sx_0 \end{aligned} \tag{10}$$

Hence, the minimization objective now is to obtain the optimal control sequence $\Delta u^* = [\Delta u_0, ..., \Delta u_{N_u-1}]^T$.

IV. Numerical results

Reference following is a common task for an autonomous system. A typical scenario deals with the tracking of references to successfully carry out a mission of transferring from an initial point to a final goal, avoiding obstacles. A simplified scenario is here designed and reproduced in MATLAB/Simulink environmentto evaluate the Controller behaviour with known Velocity and Orientation references. The Control algorithm is designed to be integrated in the simulative model of the system, a wider architecture composed by a Plant model, a Navigation module and a Guidance module, simplified in this case to provide offline calculated references, according to a model-in-the-loop validation approach.

To this end, two phases have been planned for the resolution of Problems 1 and 2 respectively.

In the first one, a representation of the state-space system is derived from datasets acquired through measurements on the robotic platform. The second part involves designing the LQMPC system together with its associated KF to fulfil the tracking requirements outlined in the aforementioned scenario. In the following, both are described in greater detail.

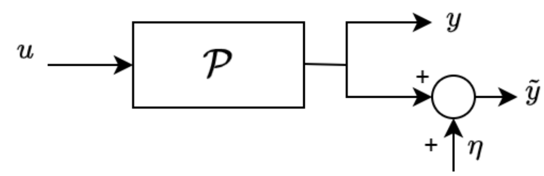

Fig. 3. Output-error identification problem scheme

A. Identification

The grey-box model derived from the UGV features PWM signals as input and longitudinal velocity V_x and orientation θ as outputs. The Lynx robot is equipped with two quadrature encoders mounted coaxially on the front left and front right DC motors, allowing for the measurement of angular speeds of the left and right tracks, i.e., ω_L and ω_R. These signals are acquired at a sampling rate of 100 Hz and processed by the FRDMK64 board. Additionally, the same board generate and acquires the PWM signals sent to the motors. Thus, for black-box identification purposes, the system features a PWM input signal u and wheel speed as the output y. This identification problem can be categorized as an Output-error problem, as the input signal remains unaffected by disturbances (Figure 3), while the measured output $\tilde{y}$ is assumed to be affected by an additional white noise η. Moreover, the input signal used for identification is sourced from a remote control, assumed to be persistently exciting the system. Additionally, the dataset employed for identification is sufficiently large to ensure accurate parameter estimation. Then data processing and fitting are carried out to obtain the state-space representation.

A satisfactory predictive capacity is obtained for a model with $n = 4$ internal states, $m = 2$ control inputs and $p = 2$ outputs. The FIT value for one step head prediction for the training set is about $\begin{bmatrix} 97.488 & 97.4229 \end{bmatrix}$ %, while the MSE value and the FPE are 0.0103 $(rad/s)^2$ and 2.6708e-05 respectively.

The identified model matrices, solution to Problem 1, are reported:

$$A = \begin{bmatrix} -2.7960 & -0.2164 & -0.7850 & 7.7403 \\ 0.7837 & -2.2623 & -8.5600 & 1.0264 \\ 3.0200 & 3.8240 & -10.6964 & 12.3116 \\ -7.7748 & 0.9730 & -2.1405 & -21.0519 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.0000 & -0.0003 \\ -0.0003 & -0.0003 \\ -0.0003 & -0.0016 \\ 0.0010 & 0.0020 \end{bmatrix}$$

$$C = \begin{bmatrix} 43.4251 & -56.6888 & 2.0322 & 1.5480 \\ 77.7576 & 32.4848 & -1.7242 & 2.2110 \end{bmatrix}$$

The D matrix is assumed as the zero matrix with appropriate dimensions. In Figure 4, the validation set is compared with the model predicted output. The FIT value is about 97.14%

for the left DC motor and 96.79% for the right one.

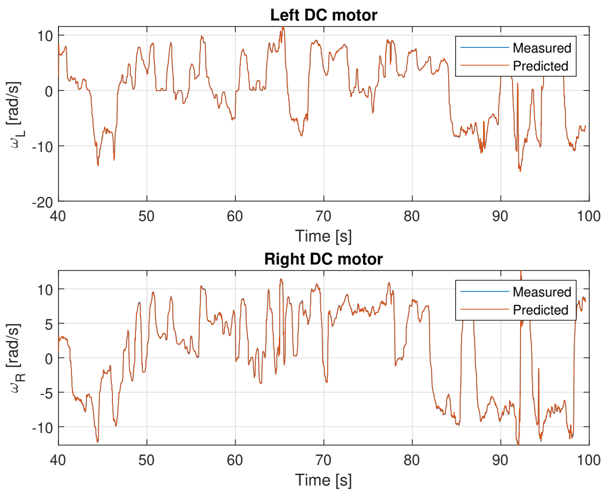

Fig. 4. Validation datasets

B. Control

As discussed in Section II-A, the data-driven identified model needs to be integrated with the DDR Kinematic model. A reformulation is required to produce as system output the forward velocity from Equation (2a) and the evolution of θ angle according to Equation (2b). The system state vector needs to be augmented, i.e. $\bar{x} = [x^T \; \theta \; u^T]^T$, where x and u are respectively the states of the black-box model and the PWM inputs. Moreover, to formulate the control problem as in Equation (8), the system input is reformulated as $\bar{u} = \Delta u$ (defined in (7)) and the output as $\bar{y} = [V_x \; \theta]^T$. The augmented system matrices are reported:

$$\bar{A} = \begin{bmatrix} A & \mathbf{0}_{4\times1} & B \\ \left[-\frac{r}{2b} \quad \frac{r}{2b}\right] C & 0 & \mathbf{0}_{1\times2} \\ \mathbf{0}_{2\times4} & \mathbf{0}_{2\times1} & \mathbf{I}_{2\times2} \end{bmatrix}$$

$$\bar{B} = \begin{bmatrix} B \\ 0 \\ \mathbf{I}_{2\times2} \end{bmatrix}$$

$$\bar{C} = \begin{bmatrix} \left[\frac{r}{2} \quad \frac{r}{2}\right] C & 0 & \mathbf{0}_{1\times2} \\ \mathbf{0}_{1\times4} & 1 & \mathbf{0}_{1\times2} \end{bmatrix}$$

Once the model is obtained, a further step is the definition of the parameters in the optimization problem of Equation (10), i.e. the weights Q, R, P, the upper and lower bounds $\bar{u}_{max}$ and $\bar{u}_{min}$, the constraint and the control horizons N_c and N_u. Several factors have been taken into account in the selection of the time horizon, in particular the computational cost, to ensure a fast execution, and the predictive capability of the model, tested during identification. Control constraints come from the hardware, capable of generating PWM values in a range of 1000 and -1000.

C. Simulation

The objective of this study is to derive a reliable UGV model for the design of an MPC. To this end, two simulations are designed with offline generated reference assumed to be known a priori, i.e. without a path planner in the loop. The performance of the proposed algorithm are evaluated by comparing a PID controller in two reference tracking missions. Mission 1 is designed to track a constant longitudinal velocity of 0.35 m/s and a constant orientation angle of 5 deg. Mission 2, in contrast, aims to guide the UGV through a sinusoidal path, thus varying the orientation angle while maintaining a constant velocity of 0.15 m/s. In Figure 5 it is possible to analyse the behaviour of the system in terms of longitudinal velocity V_x and angular position θ with respect to the reference for Mission 1. The MPC controller demonstrates effective

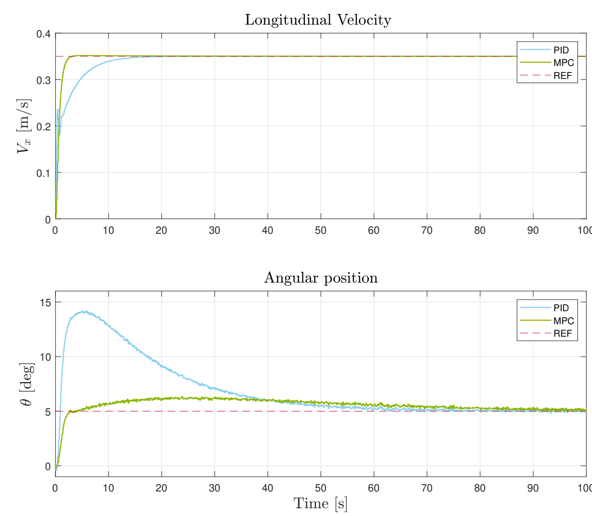

Fig. 5. Velocity and Orientation, Mission 1

tracking behaviour, with steady-state RMSE values of 0.0242 m/s for velocity and 0.8809 deg for orientation. In contrast, the PID controller exhibits RMSE values of 0.0275 m/s and 3.3465 deg for velocity and orientation, respectively. The MPC has been designed with soft constraints on the tracking errors to provide a precise response without being overly aggressive, thereby allowing accurate position tracking. This let the system to track the orientation reference avoiding the overshoot that instead characterizes the response of the system with the PID. For the velocity tracking, the settling time of the MPC is shorter than the time it takes for the PID to reach the reference value. This results in better responsiveness to dynamic events. In Figure 6 the longitudinal velocity V_x and angular position θ with respect to the reference are reported for the Mission 2.

In this scenario, the MPC is also shown to be effective, achieving a steady-state RMSE of 7.102e-05 m/s for velocity and 1.6999 deg for orientation, in comparison to the PID's RMSE values of 3.49e-04 m/s and 3.6575 deg, respectively. Furthermore, the MPC exhibits a shorter settling time for velocity and more precise orientation tracking compared to

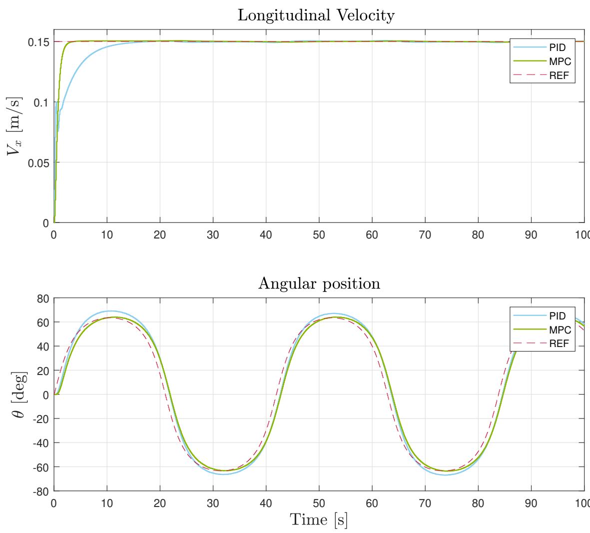

Fig. 6. Velocity and Orientation, Mission 2

the PID controller. Additionally, the aforementioned tracking performance of the MPC allows it to fulfil the objective of Mission 2 by following the sinusoidal path as shown in Figure 7, while the PID reveals its ineffectiveness.

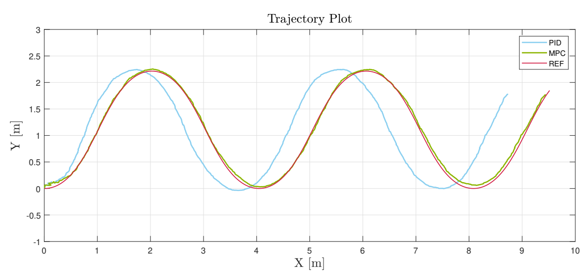

Fig. 7. Sinusoidal Trajectory tracking, Mission 2

V. Conclusion

A Linear Quadratic Model Predictive Controller for a UGV robot is presented and discussed for a Reference Tracking application. A mixed approach is adopted to create the model on which the Controller is based. The subspace system identification method is applied to obtain the state-space representation of the MIMO subsystem with the PWM signals as input and the angular velocity of the sprockets as output. Then, the DDR kinematic model is used to augment the black-box model and infer the longitudinal and rotational velocities of the robot. Finally, the effectiveness of the Controller is demonstrate in simulated missions.

As a future improvement, the use of more accurate sensors along with external tracking devices (e.g., external tracking system or motion capture cameras) can be considered to create a data-driven model describing the evolution of system states such as pose, velocity, and orientation in the inertial reference system. A direct relationship between the inputs PWM and the states in the inertial frame could be advantageous in capturing more effectively all aspects of the robot dynamics. Moreover, the implemented algorithm proves to be fast enough to be deployed on hardware. So, future steps will concern the development of the LQMPC on board the UGV robot.

References

[1] F. Faliero, "Nmpc design for autonomous driving applications," 2019.

[2] C. V. Samak, T. V. Samak, and S. Kandhasamy, "Control strategies for autonomous vehicles," in *Autonomous driving and advanced driver-assistance systems (ADAS)*. CRC Press, 2021, pp. 37–86.

[3] M. A. Kamel, A. T. Hafez, and X. Yu, "A review on motion control of unmanned ground and aerial vehicles based on model predictive control techniques," *Journal of Engineering Science and Military Technologies*, vol. 2, no. 1, pp. 10–23, 2018.

[4] S. Yu, M. Hirche, Y. Huang, H. Chen, and F. Allgöwer, "Model predictive control for autonomous ground vehicles: A review," *Autonomous Intelligent Systems*, vol. 1, pp. 1–17, 2021.

[5] O. Khatib, *Real-Time Obstacle Avoidance for Manipulators and Mobile Robots*, I. J. Cox and G. T. Wilfong, Eds. New York, NY: Springer New York, 1990.

[6] E. I. Trombetta, I. D. D. M. de Pierrepont, D. Carminati, M. Scanavino, E. Capello *et al.*, "Data-driven identification method and simulation modeling of a ground robot," in *2021 IEEE 8th International Workshop on Metrology for AeroSpace (MetroAeroSpace)*. IEEE, 2021, pp. 602–607.

[7] L. Ljung, "System identification," in *Signal analysis and prediction*. Springer, 1998, pp. 163–173.

[8] A. K. Tangirala, *Principles of system identification: theory and practice*. Crc Press, 2018.

[9] P. Van Overschee and B. De Moor, *Subspace identification for linear systems: Theory—Implementation—Applications*. Springer Science & Business Media, 2012.

[10] J. B. Rawlings and D. Q. Mayne, "Model predictive control: theory and design, nob hill pub," *Madison, Wisconsin*, vol. 825, 2009.

[11] A. Bemporad, "Model predictive control design: New trends and tools," in *Proceedings of the 45th IEEE Conference on Decision and Control*. IEEE, 2006, pp. 6678–6683.

[12] R. E. Kalman, "A new approach to linear filtering and prediction problems," 1960.

[13] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, no. 1, pp. 3–20, 2002.