IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 2, 2022, accepted May 21, 2022, date of publication May 30, 2022, date of current version June 23, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3178727

Approximate Quadratic Programming Algorithm for Nonlinear Model Predictive Tracking Control of a Wheeled Mobile Robot

RAMDANE HEDJAR, (Senior Member, IEEE)
Department of Computer Engineering, College of Computer and Information Sciences (CCIS), King Saud University, Riyadh 11543, Saudi Arabia
e-mail: hedjar@ksu.edu.sa

This work was supported by a grant from the ''Research Center of the College of Computer and Information Science'', Deanship of Scientific Research, King Saud University. The author is grateful for this support.

ABSTRACT Nonlinear model predictive control (NMPC) has proven its ability to control constrained nonlinear processes. Although NMPC can achieve exemplary tracking performance, the related computation effort as well as guaranteeing tracking convergence are its main drawbacks. Indeed, constrained NMPC is a nonlinear and nonconvex optimization problem where it is difficult to find a feasible solution within a reasonable time. Motivated by these difficulties, this study proposes a procedure, using the Euler approximation, to transform the nonlinear optimization problem of NMPC into a constrained quadratic optimization problem. The proposed tracking controller is applied to the autonomous navigation problem of a wheeled mobile robot (WMR) in a constrained space. Under certain assumptions, we prove the closed-loop system stability and boundedness of the tracking error. Furthermore, we demonstrate the recursive feasibility of the solution. Simulations are performed, first to determine the adequate control parameters, and second, to demonstrate the effectiveness of the proposed algorithm, while its real-time implementation is experimentally verified using a differential drive mobile robot.

INDEX TERMS Nonlinear predictive control algorithm, state and control signal constraints, convex optimization, stability, and feasibility analysis, wheeled mobile robot.

I. INTRODUCTION

Mobile robots have been widely studied over the last few decades, owing to their inherent applications in various fields. Mobile robots are used for the simple exploration of unknown regions for search and rescue military applications and surveillance [1]. Hence, many researchers have focused on the autonomous navigation of mobile robots in unknown environments with static and dynamic obstacles. Therefore, the desired trajectory is generated online by a motion planner, based on the information acquired from the environment, to avoid obstacles and minimize the distance to the target. The control methods used to control under-actuated systems are primarily nonlinear. Therefore, several methods based on nonlinear techniques, have been developed to solve the tracking problem and allow mobile robots to track the desired trajectory [2]–[5]. The authors in [6] and [7] focused on a more complex environment (outdoor) where the wheels are subject to deformation. However, mobile robots have physical limits, and space constraints that affect the achievable closed performance in indoor and outdoor environments.

It is a known fact that model predictive control is the best technique used to cope with hard constraints on controls and states. For this reason, nonlinear model predictive control has been widely used in academic research and industry because it is simple to include different constraints in the optimization problem [8], [9].

To solve the NMPC problem various approaches, based on sequential quadratic programming (SQP) algorithms, have been developed. This direct method converts the NMPC problem into an optimal control problem. This nonlinear programming (NLP) problem requires an online solution to a receding horizon optimization problem to be implemented in practice. Thus, this is a nonlinearly constrained optimization problem with the following challenges:

The associate editor coordinating the review of this manuscript and approving it for publication was Huaqing Li.

1- The processing time of the algorithm.
2- Nonconvexity of the optimization problem.
3- Stability of the closed-loop system and feasibility of the solution.

Therefore, most NMPC applications reported in the literature have been applied to systems with slow dynamics, such as chemical processes. For instance, in [10], the authors designed an NMPC for a continuous stirred tank reactor (CSTR) using a multi-stage control approach. The authors in [11], presented a linear mismatched model-based offset-free model-predictive control approach for CSTR nonlinear system. In [12], the authors applied robust nonlinear predictive control to the CSTR process using a radial basis function network (RBF-ARX) model. In [13], the authors formulated a nonlinear optimization problem as a quadratic cost function with inequality constraints and applied it to a CSTR plant.

In recent years significant efforts have been made to cope with the real-time implementation of NMPC to be applied to highly dynamic systems with a fast sampling rate. The real-time iteration (RTI) scheme was proposed in [14], and the authors combined ideas of the RTI with multi-level iteration (MLI) to ensure high computational efficiency for real-time implementation of the NMPC. The authors in [15] proposed a homotopy-based nonlinear interior-point method for the efficient real-time implementation of NMPC. In [16], the authors developed a highly parallelizable Newton-type method to efficiently solve NMPC. In [17], the authors presented a collection of embedded optimization algorithms under a modular structure written in the C language for better real-time nonlinear optimization. A parallel optimization toolkit for the real-time implementation of NMPC was proposed in [18]. Indeed, the authors used a highly parallelizable method for efficient real-time implementation of NMPC. The authors in [19] used the RPROP algorithm based on a faster backpropagation approach for a gradient-descent optimizer. To this end, the authors in [20] used the high-performance framework ACADO software for computationally efficient implementation of NMPC to reluctance synchronous machines. Consequently, to overcome the computational burden of NMPC, numerous schemes have been developed in recent years (see excellent reviews [38]).

Despite these challenges, several applications of NMPC to mobile robots have been reported in the literature. The first applications focused on tracking the desired trajectory [19], [21], [22]. In [23], the authors proposed an NMPC using a nonlinear (nonconvex) optimization algorithm to track the centerline of the roadway while avoiding obstacles. However, the iterative method used to determine the optimal solution is highly time-consuming. The authors in [24] combined a path-following controller with obstacle avoidance using nonlinear model predictive control. Although the authors have provided stability and feasibility analysis of the algorithm, the optimization problem is nonlinear and nonconvex with a high computational burden. Nonlinear model predictive control was also used in [25] to steer the WMR to the desired pose. Despite the rigorous asymptotic stability analysis, the proposed solution must solve the nonlinear optimization problem. To avoid the local minimum of the optimization problem, using receding horizon control (RHC), the authors in [26] proposed a local-minima-free navigation function with a global minimum at the target state. The proposed algorithm solves the navigation problem by using control constraints. The authors in [27] used a genetic algorithm (GA) to solve the NMPC algorithm to avoid the computational difficulties of nonlinear programming-based MPC. This algorithm has been successfully applied to solve the steering problem of autonomous vehicles. To reduce the processing time owing to online optimization, the authors in [28] proposed a convex quadratic programming-based model predictive scheme for collision-free collision navigation of autonomous vehicles.

Consequently, the processing time of the NMPC algorithm is one of the most commonly encountered problems when it is used to control a highly dynamic nonlinear system. The time required to reach the optimal solution is high with regard to the dynamic of the process. This is because of the nonlinear optimization problem that is solved at each sample time (online) [29]. Furthermore, the optimization problem is, generally nonconvex, and the solution can converge to the local minimum [30], [31]. Hence, the convexity of the optimization problem plays a central role in the practical implementation of constrained NMPC.

In this paper, we propose a simple algorithm, based on approximated model predictive control, for safe autonomous navigation of the WMR in a constrained space. This algorithm permits the WMR to accurately track a feasible predefined trajectory. To avoid the computational burden of the NMPC, the Euler approximation is utilized to convert the NLP to constrained quadratic programming (QP). Figure 1 shows the constrained QP implementation of the proposed algorithm. This study aims to transform the tracking problem, which is a nonconvex optimization problem, into a constrained QP problem. The restricted space and constraints on the control input are expressed as convex constraints (inequality constraints). Thus, the heavy computational burden of the original optimization problem can be considerably reduced using fast-available algorithms [32], [33]. Furthermore, the stability analysis of the closed-loop system is investigated using the Lyapunov method and the positive invariance principle in constrained cases. The feasibility of the constrained QP solution is also investigated. Consequently, the contribution of the proposed algorithm solves the challenges previously cited in the application of NMPC to wheeled mobile robots.

The proposed algorithm can be used efficiently in applications where navigation space is limited, such as in robot vacuum cleaners and landmine detection robots. Moreover, undetectable obstacles, such as stairs in an indoor environment, can also be included as space constraints to prevent the robot from falling downstairs. Thus, the proposed control algorithm is simple, practical, and efficient with a feasible

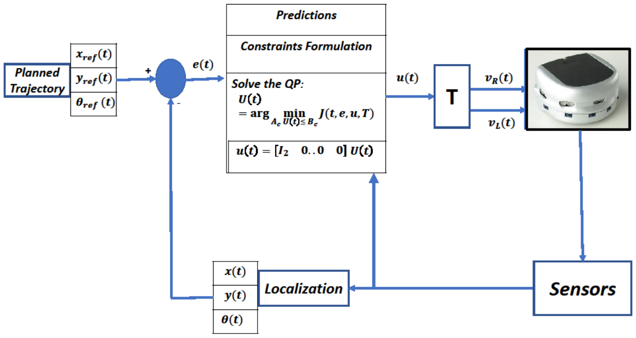

FIGURE 1. Trajectory tracking and collision-free path planning.

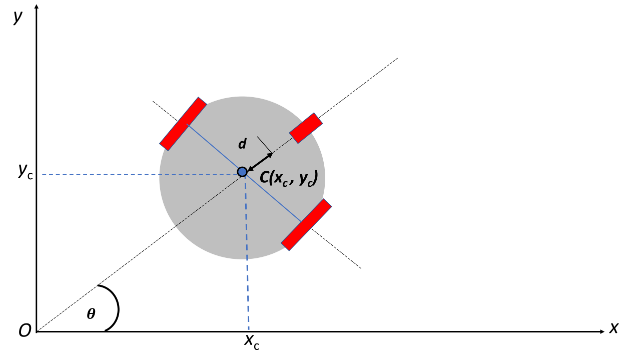

FIGURE 2. Wheeled mobile robot structure.

solution and asymptotic stability of the equilibrium point of the closed-loop system.

The remainder of this paper is organized as follows. In Section 2, the kinematic model of a unicycle mobile robot is derived based on certain assumptions and properties. Section 3 describes the trajectory tracking algorithm based on the approximated nonlinear model predictive control (ANMPC). The stability of the closed-loop system and the feasibility of the solution are discussed in Section 4. Simulations and experimental tests are conducted to determine the effectiveness of the proposed scheme. Section 5 presents the results of the study. The paper ends with the conclusion and future work.

II. PRELIMINARIES

This section deals with some preliminaries required for this work.

A. MATHEMATICAL MODELING

In this study, we considered a differential drive mobile robot. It contains two independent rear wheels and a front-castor wheel. The linear speed v of the WMR is given by the combination of the left speed v_l and right speed v_r of each DC motor: $v = \frac{v_l + v_r}{2}$ and the angular speed $\omega = \frac{v_r - v_l}{L}$, where L is the distance between the left and right wheels. The transformation of these variables, it is shown in Figure 1, is given as follows:

$$\begin{bmatrix} v_r \\ v_l \end{bmatrix} = \begin{bmatrix} 1 & \dfrac{L}{2} \\ 1 & -\dfrac{L}{2} \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = T \begin{bmatrix} v \\ \omega \end{bmatrix}.$$

The Kinematic model of the mobile robot is given by [1]

$$\begin{cases} \dot{x_c} = v\, cos\,(\theta) \\ \dot{y_c} = v\, sin\,(\theta) \\ \dot{\theta} = \omega \end{cases} \tag{1}$$

This kinematic model is valid under nonholonomic constraints [1]. To overcome the controllability problem of the kinematic model (1), the coordinates of the WMR are shifted from the mass center $C(x_c,\ y_c)$ to point $C(x,\ y)$ along the sagittal axis by a distance$|d|\neq 0$ (see Figure 2). Thus, the new coordinates of the WMR are given by

$$x = x_c + d\, cos\,(\theta)$$
$$y = y_c + d\, sin(\theta)$$

Under the matrix form, the nonlinear kinematic model of WMR is

$$\dot{Z}(t) = f\,(Z, U, t) = G\,(\theta)\ u(t) \tag{2}$$

where

$$G\,(\theta) = \begin{bmatrix} cos\,(\theta) & -\,d\ sin\,(\theta) \\ sin\,(\theta) & d\ cos\,(\theta) \\ 0 & 1 \end{bmatrix},$$
$$Z(t) = \left[x(t)\ y(t)\ \theta(t)\right]^T$$

and $u(t) = \left[v(t)\ \omega(t)\right]^T$. The discrete-time version of the model (2) is

$$Z\,(k+1) = Z(k) + T_s\, G\,(\theta(k))\ u(k)$$

where T_s is the sampling time.

The tracking error dynamic can be written as

$$\dot{e}(t) = f\,(Z, U, t) - \dot{Z_{ref}} = G\,(\theta)\ u(t) - \dot{Z_{ref}} \tag{3}$$

B. ASSUMPTIONS

In order to design the ANMPC algorithm, some assumptions are required.

1- Both reference position $Z_{ref}(t)$ and its time derivative $\dot{Z}_{ref}(t)$ are bounded. So, $\exists Z_{max} > 0,\ \acute{Z}_{max} > 0$ such that $\left|Z_{ref}(t)\right| < Z_{max};\ \left|\dot{Z}_{ref}(t)\right| < \dot{Z}_{max},\ \forall\ t\ \in\ \mathfrak{R}_+$ where $Z_{ref}(t) = \left[x_{ref}(t)\ y_{ref}(t)\ \theta_{ref}(t)\right]^T \in \mathfrak{R}^3$.

2- The first-order Taylor approximation (Euler method) is used to convert the constrained nonconvex optimization problem to a constrained quadratic optimization problem. It is assumed that the approximation is valid for a small step-time h.

3- The state vector $Z(t)$ is available.

C. DEFINITIONS

Here, we define some properties that will be used in the stability and feasibility analyses.

1- The column vectors of $G(\theta)$ are linearly independent. Consequently, $G(\theta)\,u(t) = 0 \implies u(t) = 0$. Accordingly, for any positive definite (or negative definite) matrix $\sqcap$, the matrix $G^T(\theta) \ \sqcap \ G(\theta)$ is also a definite positive (or negative definite) matrix.

2- The minimal and maximal eigenvalue of a matrix Q is denoted by $\lambda_{min}(Q)$ and $\lambda_{max}(Q)$ respectively.

3- The identity matrix is denoted by I_n.

4- The Euclidean norm in $\mathbb{R}^n$ is written as $|.|$ while for a matrix $P \in \mathbb{R}^{n\times m}$, $\|P\| = \lambda_{max}\left(P\,P^T\right)^{1/2}$.

D. NOTATION

In addition to the nonholonomic constraint, the position of the WMR $P_{xy}(t) = \left[x(t)\ y(t)\right]^T \in \mathfrak{R}^2$ and the input vector $u(t) = \left[\nu(t)\ \omega(t)\right]^T \in \mathfrak{R}^2$ are constrained inside a given compact set. For the control signal: $\cup = \left\{[\nu_{min}, \nu_{max}] \times [\omega_{min}, \omega_{max}] \subset \mathbb{R}^2\right\}$.

The sequences of the control signals

$$U(t) = \left[u(t)\ u(t+h)\ \cdots\ u(t+(m-1)h)\right]^T$$

is feasible, if $U(t) \in \mho_T$, where

$$\mho_T = \left\{U(t) \in \mathfrak{R}^{2\times m} / u\,(t+ih) \in \cup \text{ for } i=0, 1, \cdots, m-1.\right\}.$$

while the restricted cartesian space is defined as compact (nonempty) set by:

$$\mathbb{Z} = \left\{[x_{min}, x_{max}] \times [y_{min}, y_{max}] \times \mathbb{R} \subset \mathbb{R}^3\right\},$$

$$\mathbb{Z}_T = \left\{\overline{Z}(t) \in \mathfrak{R}^{3\times m} / Z\,(t+ih) \in \mathbb{Z},\ \text{for } i=0, 1, \cdots, m\right\},$$

$$\overline{Z}(t) = \lfloor Z(t+h)\ Z(t+2h)\ \cdots\ Z(t+mh)\rfloor^T.$$

We can also define the constrained compact set related to the predicted tracking error as

$$\mathbb{E} = \left\{\left[x_{min} - x_{ref}(t), x_{max} - x_{ref}(t)\right] \times \left[y_{min} - y_{ref}(t), y_{max} - y_{ref}(t)\right] \times \mathbb{R} \subset \mathbb{R}^3\right\},$$

$$\mathbb{E}_T = \left\{E(t) \in \mathfrak{R}^{3\times m} / e\,(t+ih) \in \mathbb{E} \text{ for } i = 1, \cdots, m\right\}$$

where

$$E(t) = \left[e(t+h)\ \ e(t+2h)\ \ \cdots\ \ e(t+mh)\right]^T.$$

Then, for an initial feasible state error $e(t)$, the sequence of the control vector is

$$U(t) = \left[u(t)\ \ u(t+h)\ \ \cdots\ \ u(t+(m-1)h)\right]^T \in \mho_T,$$

is an admissible control vector, if the state error trajectories $E(t)$, generated by the dynamic system (3), satisfies $E(t) \in \mathbb{E}_T$.

We also define a compact set $\mathbb{Z}_U = \left\{U(t) \in \mathfrak{R}^{2\times m}\right.$, -$Z(t) \in \mathfrak{R}^{3\times m} / U(t) \in \mho_T$, -$\left.Z(t) \in \mathbb{Z}_T\right\}$, which will be converted to a polyhedral set using the Euler approximation.

III. RECEDING HORIZON CONTROL OF A WMR

The formulation of the NMPC problem for trajectory tracking is to determine the control vector $u(t)$ such that the predicted tracking error $e\,(t+\tau) = Z\,(t+\tau) - Z_{ref}\,(t+\tau)$ converges to zero along the interval $\tau \in [t,\ t+\ T]$. Note that $Z_{ref}(t)$ is the safe trajectory generated by the planner (Figure 1) to move the WMR to the desired target without colliding with the obstacles.

The above problem is reformulated as an optimization problem [8], [9]:

$$\underset{u\in\mho_T}{Min}\ J\,(t, e, u, T) = \underset{u\in\Omega}{Min}\ \frac{1}{2}\int_t^{t+T} L\,(\tau)\,d\tau, \qquad (4)$$

subject to:

- $\dot{e} = f\,(e, u, t) = G\,(\theta)\,u(t) - Z_{ref}(t)$,
- $u\,(\tau) \in \cup\mathfrak{R}^2$, $e\,(\tau) \in \mathbb{E}$,
- $L\,(\tau) = e\,(\tau)^T Q\,e\,(\tau) + u\,(\tau)^T R\,u\,(\tau)$,
- $e\,(\tau) = Z\,(\tau) - Z_{ref}(\tau),\quad \tau \in [t,\ t+T]$,

where $T > 0$ is the prediction horizon, $Q > 0$ is a positive definite matrix ($\dim(Q) = 3\ \times 3$), and $R \geq 0$ is a positive semi-definite matrix $\dim(R) = 2\ \times 2$).

Problem (4) is a nonconvex optimization problem that must be solved online at each sampling time with computational complexity. Furthermore, most of the existing algorithms may converge to local solutions. In this study, optimization problem (4) is converted to a convex quadratic optimization problem using the first-order Taylor approximation to solve it efficiently with a global minimum. Therefore, to obtain an explicit solution of NMPC, the prediction horizon T is divided into m subintervals ($m \geq 2$) of equal width h such that $T = m\,h$. The discrete-time version of the cost function (4) is given by

$$J(e(t)) = \sum\nolimits_{i=1}^{m-1} L(t+i\,h) + L_T(t+m\,h), \qquad (5)$$

where $e(t)$ is the initial tracking error and $L\,(t+jh) = e\,(t+jh)^T Q\,e\,(t+jh) + u\,(t+jh)^T R\,u(t+jh)$.

To ensure the stability of the closed-loop system, the terminal cost should be included in the objective function (5), as follows

$$L_T\,(t+mh) = e\,(t+mh)^T P\,e\,(t+mh)$$

where the adequate selection of the symmetric definite-positive matrix $P \in \mathfrak{R}^{3\times 3}$ is provided later. Thus, the cost function (5) can be written as

$$J\,(t, e(t), U(t), h) = \left(\sum\nolimits_{i=0}^{m-1} e\,(t+ih)^T Q\,e\,(t+ih) + u\,(t+ih)^T R\,u(t+ih)\right) + e\,(t+mh)^T P\,e(t+mh) \qquad (6)$$

To avoid the cross product of the control signals in (6), the following approximation is used:

$$\begin{cases} sin(\theta\,(t+jh)) \approx sin(\theta(t)) \\ cos(\theta\,(t+jh)) \approx cos(\theta(t)) \end{cases} \Rightarrow G(\theta\,(t+jh)) \approx G(\theta)$$

Hence, the cost function (6) becomes

$$J\,(e(t),U) = \left\{\varphi\left(e(t),\dot{Z}_{ref},h\right) + \sum_{i=1}^{m-1} e\,(t+ih)^T\,Q\,e\,(t+ih) + e\,(t+mh)^T P\,e\,(t+mh) + U(t)^T\overline{R}U(t)\right\},$$

where $\varphi\left(e(t),\dot{Z}_{ref},h\right)$ represents terms that are independent of the control vector $U(t)$ and $\overline{R} = diag(R,\ R,\ R,\ \ldots\ldots.,\ R)$.

To evaluate the effect of the control effort on the predicted tracking error, Euler approximation was used as follows

$$e\,(t+jh) = e(t) + h\,G\,(\theta)\sum_{i=0}^{j-1} u(t+jh) - h\sum_{i=0}^{j-1}\dot{Z}_{ref}(t+jh) \tag{7}$$

where $U^T(t) = \lfloor u^T(t)\ u^T(t+h)\ \cdots\ u^T(t+(m-1)h)\rfloor$, and

$$\dot{\overline{Z}}_{ref}{}^T(t) = \left[\dot{Z}^T_{ref}(t)\ \dot{Z}^T_{ref}(t+h)\ \cdots\cdots\ \dot{Z}_{ref}{}^T(t+(m-1)h)\right]^T$$

Using (7), the cost function can be rewritten under the quadratic form

$$J\,(e(t),U(t)) = \left\{\varphi\left(e(t),\dot{Z}_{ref},h\right) + 2\,h\,U^T(t)\overline{G}^T\overline{Q}\left(\overline{I}_1\,e(t) - h\overline{I}_{3m}\dot{\overline{Z}}_{ref}\right) + U^T(t)\left(\overline{R}+h^2\Pi\right)\,U(t)\right\} \tag{8}$$

where

$$\overline{I_{3m}} = \begin{bmatrix} I_3 & 0 & \cdots & \cdots & 0 \\ I_3 & I_3 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \\ I_3 & I_3 & I_3 & \cdots & I_3 \end{bmatrix};\ \dim\left(\overline{I}_{3m}\right) = 3m\times 3m;$$

$$\overline{I_1} = \left|I_3\ \cdots\ I_3\right|^T;\ \dim\left(\overline{I_1}\right) = 3m\times 3;$$

$$\Pi = \overline{G}^T\overline{Q}\,\overline{G}.$$

$$\overline{Q} = diag\,(Q,Q,\cdots,P)\,;$$

$$\overline{G} = \begin{bmatrix} G(\theta) & 0 & 0 & \cdots & 0 \\ G(\theta) & G(\theta) & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ G(\theta) & G(\theta) & G(\theta) & \cdots & G(\theta) \end{bmatrix};$$

$\dim\left(\overline{G}\right) = 3m\times 2m$.

Because $\overline{Q} > 0,\ P > 0$, and $\overline{R} \geq 0$, the matrix $\left(\overline{R}+h^2\,\Pi\right)$ is a positive definite matrix. The optimization problem (8) can be written in the following quadratic form

$$\underset{U(t)\in\mho_T}{Min}\ J\,(t,e,U(t)) = \underset{U(t)\in\mho_T}{Min}\left\{\varphi + 2\,h\,U^T(t)\overline{G^T}\,\overline{Q}\left(\overline{I_1}e(t) - h\overline{I_{3m}\dot{Z}_{ref}}\right) + U^T(t)\left(\overline{R}+h^2\Pi\right)\,U(t)\right\}, \tag{9}$$

such that

- $e\,(t+jh) = e(t) + h\,G\,(\theta)\,I_{2j}\,U(t) - h\ I_{3j}\dot{\overline{Z}}_{ref}$, for $j = 1,\cdots,m$
- $U(t)\in\mho_T$, and $E(t)\in\ \mathbb{E}_T$

A. CONSTRAINTS HANDLING

Constrained spaces are primarily utilized in the motion planning of robot manipulators. Hence, the contribution of this study is the formulation of the motion-planning problem of a mobile robot in a constrained environment as a constrained quadratic optimization problem. Therefore, two types of constraints were considered:

1) CONSTRAINTS ON CONTROL VECTOR

All real-world control problems are subject to the actuator constraints. Thus, the control signals $v(t)$ and $\omega(t)$ of the mobile robot (speed and steering angle) are bounded and cannot exceed a certain threshold level [2]. In model predictive control, the predicted control signals should be constrained within the admissible set. Thus, constraint $U(t)\in\mho_T$ can be expressed as a linear constraint

$$A_1\ U(t) \leq B_1$$

where $A_1 = diag(\overline{A},\ \overline{A},\ \cdots\cdots,\overline{A}),\ \overline{A} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 0 \\ 0 & -1 \end{bmatrix},\ B_1 = \begin{bmatrix} \overline{B} \\ \vdots \\ \vdots \\ \overline{B} \end{bmatrix}$, and $\overline{B} = \begin{bmatrix} v_{max} \\ \omega_{max} \\ -v_{min} \\ -\omega_{min} \end{bmatrix}$.

2) STATE CONSTRAINTS

The idea used in this work, which allows the mobile robot to navigate safely inside a restricted space, has been used in the handling of contacts in crowd motion simulations [34] and extended to mobile robot navigation [35]. The admissible sub-space F_R can be seen as an intersection of finitely many half-planes in space $\mathfrak{R}^2$ and can be defined by a set of nonlinear inequalities $C_r(P)$ such that

$$F_R = \left\{P_{xy}(t)\in\ \mathfrak{R}^2 : C_r(P)\leq 0\right\}\subset\mathbb{E}, \tag{10}$$

where $P_{xy}(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}$ is the Cartesian position of the mobile robot at time t. To ensure a global solution of the constrained QP, the achievable region is defined as a convex polyhedron with a set of p convex inequalities: $C_r(P) = A_rP_{xy}(t) + B_r$, where $A_r\ \in\ \mathfrak{R}^{p\times 2}$, and $B_r\ \in\ \mathfrak{R}^p$. For simplicity, this region is defined by a convex polyhedron in Cartesian space, as follows:

$$F_R = \left\{P_{xy}(t)\in\ \mathfrak{R}^2;\ x_{min}\leq x(t)\leq x_{max} \text{ and } y_{min} \leq y(t)\leq y_{max},\ t\in[t,t+T]\right\}.$$

Let us define the kinematic model of the WMR position in Cartesian space as $\dot{P}_{xy}(t) = G_{xy}\,(\theta)\ u(t)$,

$$\text{where } G_{xy}\,(\theta) = \begin{bmatrix} cos\,(\theta) & -d\ sin\,(\theta) \\ sin\,(\theta) & d\ cos\,(\theta) \end{bmatrix}.$$

It is desirable to maintain the predicted Cartesian position of the mobile robot within this region F_R. Therefore, the problem to be solved is to determine an adequate control vector $U(t)$ that satisfies the following constraints.

$$P_{xy}(t) \in \mathfrak{R}^2; \quad P_{xy}\left(t+\tau\right) \subset F_r,\ \tau \in [0, T]$$

Or under the following inequalities

$$P_{min} \le P_{xy}\left(t+\tau\right) \le P_{max}, \quad \tau \in \left[0,\ T\right], \tag{11}$$

where $P_{min} = \begin{bmatrix} x_{min} \\ y_{min} \end{bmatrix}$ and $P_{max} = \begin{bmatrix} x_{max} \\ y_{max} \end{bmatrix}$.

The $m-$ steps predicted Cartesian position of the WMR is given by: $\overline{P_m}(t) = \overline{P}(t) + h\ AG_{xy}\ U(t)$,

where $$AG_{xy} = \begin{bmatrix} G_{xy} & 0 & \cdots 0 \\ G_{xy} & G_{xy} & 0\cdots 0 \\ \cdots & \cdots & \cdots & \cdots \\ G_{xy} & G_{xy} & G_{xy} & G_{xy} \end{bmatrix};$$

$$\overline{P}(t) = \begin{bmatrix} P_{xy}(t) \\ P_{xy}(t) \\ \vdots \\ P_{xy}(t) \end{bmatrix}, \text{ and } \overline{P}_m(t) = \begin{bmatrix} P_{xy}(t+h) \\ P_{xy}(t+2h) \\ \vdots \\ P_{xy}(t+mh) \end{bmatrix}.$$

Constraints given in equation (11) can be written as:

$$h\,AG_{xy}\ U(t) \le \overline{P}_{max}\overline{-P}(t)$$
$$-h\,AG_{xy}\ U(t) \le \overline{-P}_{min} + \overline{P}(t)$$

where $\overline{P}_{min}(t) = \begin{bmatrix} P_{min} \\ P_{min} \\ \\ \vdots \\ P_{min} \end{bmatrix}$ and $\overline{P}_{max}(t) = \begin{bmatrix} P_{max} \\ P_{max} \\ \\ \vdots \\ P_{max} \end{bmatrix}$.

Under matrix-vector form

$$h\,A_2\ U(t) \le B_2,$$

which are convex constraints with $A_2 = \begin{bmatrix} AG_{xy} \\ -AG_{xy} \end{bmatrix}$, and $B_2 = \begin{bmatrix} \overline{P}_{max} - \overline{P}(t) \\ -\overline{P}_{min} + \overline{P}(t) \end{bmatrix}$.

Consequently, the state constraints $Z(t + ih) \in \mathbb{Z}_T$ are approximated by a linear inequality (with regard to $U(t)$) $h\,A_2\ \ U(t) \le B_2$. Therefore, if assumption 2 holds, the optimization problem can be formulated as a constrained QP problem as follows

$$U^*(t) = arg \min_{A_c\ U(t) \le B_c} J(t, e, u, T) \tag{12}$$

where $A_c = \begin{bmatrix} A_1 \\ A_2 \end{bmatrix}$, $B_c = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$.

The optimization problem (12) is a convex quadratic problem because of the quadratic cost and linear inequality constraints (with regards to vector $U(t)$). Furthermore, the optimization problem is feasible because the feasible set, defined by $\mho_T = \left\{U(t) \in \mathfrak{R}^{2\times m} / A_c\ U(t) < B_c\right\}$ is nonempty (the matrix A_c has a full rank). Hence, many algorithms exist for solving this convex optimization problem within a few steps [31]–[33].

IV. STABILITY AND FEASIBILITY ANALYSIS

The stability of constrained model predictive control of nonlinear systems has been the main subject of research for many years. The closed-loop stability of the plant under the cost function with state and input constraints (12) can be guaranteed independently from the nonlinear mathematical model of the plant [36], [37]. This is realized by adding an equality constraint to the optimization problem, which forces the tracking error to lie inside a terminal region at the end of the finite horizon. This terminal region is the region of attraction for a nonlinear system controlled by a local controller.

A. STABILITY ANALYSIS

In this work, the Lyapunov method is used to prove the stability, and feasibility of the system closed by the model predictive control. The proposed controller drives the state error to the terminal set. Inside this terminal set, a local controller was used to steer the error state to the equilibrium point.

Lemma: For system (3), the compact set $\Lambda_F = \{e(t) \in \mathfrak{R}^3, V_F(e) = e\left(\tau\right)^T P\, e\left(\tau\right) \le\ \alpha,\ \ for\ \tau \ge t_F = t + mh\}$ is a positively invariant set with $\alpha = {\delta_M}^2$, under the local controller:

$$u_F\left(\tau\right) = -\frac{1}{h}\Big(G\left(\theta\right)^T P\, G\left(\theta\right)\Big)^{-1} G\left(\theta\right)^T P\, e\left(\tau\right), \tag{13}$$

Proof: To determine the current control $u_F\left(\tau\right)$ that reduces the tracking error along with the interval $\tau \in [t_F t_F + h]$, we consider a dynamic performance index, that penalizes the tracking error in the form

$$J_{Ind}\left(e, u_F\right) = e\left(t_f + h\right)^T P\, e\left(t_f + h\right),$$

where $t_f = t + T$. The minimization of $J_{Ind}\left(e, u_F\right)$ with respect to $u_F\left(\tau\right)$, using the first-order Taylor approximation of the one-step-ahead predicted tracking error and setting $\frac{\partial J_{Ind}}{\partial u_F} = 0$ yields:

$$u_F\left(\tau\right) = -\frac{1}{h}\Big(G\left(\theta\right)^T P\, G\left(\theta\right)\Big)^{-1} G\left(\theta\right)^T P\, e\left(\tau\right),$$

where $\left(G\left(\theta\right)^T P\, G\left(\theta\right)\right)^{-1}$ is a positive-definite matrix.

Let's take the following Lyapunov function

$$V(e) = \frac{1}{2} e(t_f)^T P\, e\left(t_f\right). \tag{14}$$

Given that the planned trajectory can be considered as a path or set of points with $\dot{Z}_{ref} = 0$, the time derivative of the Lyapunov function using the local controller (13) yields,

$$\begin{aligned} \dot{V}(e) &= e(t_f)^T P \dot{e}\left(t_f\right) \\ &= -\frac{1}{h} e\left(t_f\right)^T\ P\, G\left(\theta\right) \left(G\left(\theta\right)^T\ P\, G\left(\theta\right)\right)^{-1} \\ &\quad \times G\left(\theta\right)^T\ \ P\, e\left(t_f\right) \\ \dot{V}(e) &= -\frac{1}{h} e\left(t_f\right)^T\ \Xi\left(\theta\right) e\left(t_f\right) < 0, \end{aligned}$$

since $\Xi(\theta) = P\,G(\theta)\left(G(\theta)^T P\,G(\theta)\right)^{-1} G(\theta)^T P$ is also positive definite. The function $s^T\left(G(\theta)^T P\,G(\theta)\right)^{-1} s > 0,\ \forall\, s \neq 0 \in \Re^2$. Using the transformation $s = G(\theta)^T P\,e$, the function $e^T\,\Xi(\theta)\,e > 0,\ \forall\, e \in \Re^3$ provided that $e \neq 0$. Thus, the set $\Lambda_F = \left\{e(t) \in \Re^3, V_F(e) = e(\tau)^T P\,e(\tau) \le \alpha,\right.$ for $\left.\tau \ge t_F = t + mh\right\}$ is a positively invariant set under the local controller (13). The tracking error entersset Λ_F, then it belongs to this terminal set for $\tau \ge t_F$. Subsequently, the tracking error asymptotically converges to the origin.

Theorem 1: Assume that the initial solution of the optimization problem is feasible, $e(t + ih) \in \mathbb{E}$, and $u(t + (i - 1)h) \in \cup\, for\ i = 1,\ m$. Suppose that the following inequality holds:

$$G^T(\theta)\,P\,G(\theta) > G^T(\theta)\,Q\,G(\theta) + \frac{1}{h^2}R. \qquad (15)$$

Then, under terminal controller (13), the optimal cost function $J(t, e(t), h)$ is a control Lyapunov function (CLF) for system (3). Consequently, the equilibrium point becomes asymptotically stable.

Proof: See the appendix.

B. FEASIBILITY ANALYSIS

To guarantee the closed-loop stability and convergence derived previously, we assume that the predictions by the tail lie inside the constrained space. In other words, the optimal solution at time t is feasible.

$$U^*(t) = \left[\, u^*(t)\ u^*(t+h)\ \cdots\ u^*(t + (m-1)h)\,\right]^T,$$

where $u(t + ih) \in \cup,\ i = 1,\ \cdots, m.$

$$\overline{P^*}_{xy}(t) = \left[\, P^*_{xy}(t+h)\ P^*_{xy}(t+2h)\ \cdots\ P^*_{xy}(t+mh)\,\right]$$

where $\overline{P}^*_{xy}(t)$ is the predicted cartesian position of the mobile robot using the optimal solution $U^*(t)$ with $P^*_{xy}(t + ih) \in \mathbb{Z}_T \subset \Re^2, i = 1, \cdots, m.$

We must ensure the feasibility of the solution in the next step, that is, at time $t + h$ [36], [37]. Hence, the next solution to the optimization problem must be feasible (not necessarily optimal). The solution at time $t + h$ is

$$\tilde{U}(t+h) = \left[\, u*(t+h) \cdots u*(t + (m-1)h)\ u_F(t+mh)\,\right]^T,$$

$$\tilde{P_{xy}}(t+h) = \left[\, P^*_{xy}(t+h)\ P^*_{xy}(t+2h) \cdots P^*_{xy}(t+mh+h)\,\right]^T.$$

Therefore, the necessary and sufficient conditions for the predictions generated by tilde $\tilde{U}(t+h)$ and $\tilde{P_{xy}}(t+h)$ to be feasible at time $t + h$, whenever the solution of the optimization problem at time t is feasible, are

i. $P_{min} \le P^*_{xy}(t + mh + h) \le P_{max}$ or $P^*_{xy}(t + mh + h) \in \mathbb{Z}$.
ii. $u_{min} \le u_F(t + mh) \le u_{max}oru_F(t + mh) \in \cup$,

For the solution given by (i), it is assumed that the predicted tracking error at time $t + mh$ is feasible, then there exists a feasible subset of $\mathbb{E}$, named Λ_F such that $e^*(t + mh) \in \Lambda_F \subseteq \mathbb{E}$. Hence, using the Lemma, the tracking error at the next time is also feasible $e^*(t + mh + h) \in \Lambda_F \subseteq \mathbb{E}$. Thus, the optimal solution will drive the state error from the set $\mathbb{E}$ to the terminal set Λ_F. To constraint the control signal within its limits (ii) when the tracking error lies inside the terminal region Λ_F, the following inequality should be satisfied

$$|u_F(\tau)| \le \frac{1}{h}\left\|\left(G(\theta)^T P\,G(\theta)\right)^{-1} G(\theta)^T P\right\| |e(\tau)| \overline{\le} u,$$

using assumption 6, the inequality becomes

$$|e(\tau)| \le \frac{h\bar{u}}{\lambda_{max}\left(\Psi\ \Psi^T\right)^{\frac{1}{2}}}, \qquad (16)$$

where $\Psi = \left(G(\theta)^T P\,G(\theta)\right)^{-1} G(\theta)^T P,\ \bar{u} = Max\ (u_{max}, |u_{min}|)$.

Hence, condition (16) is satisfied if the terminal region is selected to be.

$$\Lambda_F = \left\{ e(\tau) \in \mathbb{E} \subset \Re^3\ / V_F(e) \le \alpha,\ \text{sucht hat}\ \alpha = \frac{\lambda_{max}(P) h^2 \overline{u^2}}{\lambda_{max}\left(\Psi\Psi^T\right)^{1/2}} \right\}.$$

Consequently, feasibility can be achieved provided that the control problem admits a feasible solution at the initial time t, in order to maintain the feasibility of all solutions. It is well known that feasibility at one time instant leads to a feasible solution at the next instant, and the value of the cost function decreases [36], [37].

Theorem 2: Suppose that assumptions 1-3 hold, and matrix P is chosen such that inequality (15) holds. Starting from a feasible initial position $Z(t) \in \mathbb{Z}$ and a feasible control prediction sequence $U(t) \in \mho_T$, then the solution $U^*(t)$ of the quadratic problem (12) steers any initial tracking error from $\mathbb{E}$ to Λ_F. Furthermore, the terminal control signal $u_F(\tau)$ steers the tracking error to the equilibrium point e^* when $\tau \ge (t + T + h)$.

Proof: Assume that the initial control sequence $U(t)$ and the initial state are feasible at time t. Using the constrained QP problem (12), the optimal solution $U^*(t)$ is feasible. From Theorem 1, the control Lyapunov function is a decreasing function. Then, the polyhedron set $\mathbb{Z}_T$ is a positively invariant set and the tracking error converges to the terminal region within the interval $[t,\ t + T]$. For $\tau \ge (t + T + h)$, the terminal controller $u_F(\tau)$steers the tracking error to the equilibrium point e^*. Thus, the equilibrium point of the closed-loop system is asymptotically stable.

According to the feasibility or unfeasibility of the reference trajectory $Z_{ref}(\tau)$, there are two cases:

1) CASE 1: $Z_{ref}(\tau) \in \mathbb{Z}$ (FEASIBLE TRAJECTORY)

In this case, the terminal region is an attractive region with an equilibrium point at the origin. Indeed, the solution provided by the QP algorithm (12) is feasible, because $Z(\tau) \in \mathbb{Z}$ and

$Z_{ref}(\tau) \in \mathbb{Z}$, the equilibrium point is $e^* = \lim_{\tau\to\infty} Z(\tau) - Z_{ref}(\tau) = 0$. The dynamic of the tracking error inside the terminal region is given by

$$\begin{aligned} \grave{e}(\tau) &= -\frac{1}{h} H\, e(\tau) \text{ for } \tau \geq t+mh \\ &\Longrightarrow e(\tau) = e(t+mh)\, e^{-\frac{1}{h}H\tau}, \end{aligned}$$

where $H = \left(G(\theta)^T P G(\theta)\right)^{-1} G(\theta)^T P$. Hence, we have

$$\lim_{\tau\to\infty} e(\tau) = e(t+mh)\, e^{-\frac{1}{h}H\tau} = e^* \to 0$$

2) CASE 2: $Z_{ref}(\tau) \notin \mathbb{Z}$ (UNFEASIBLE TRAJECTORY)

In this case, Z(τ) $\in \mathbb{Z}$ and $Z_{ref}(\tau) \notin \mathbb{Z}$, the equilibrium point is $e^* = Z(t+T) - Z_{ref}(t+T) = Z_s - Z_{ref} = e_s \neq 0$, where Z_s is the value of $Z(t)$ on the border of the feasible set $\mathbb{Z}$. Thus, the solution converges to an equilibrium point defined by $e_s \neq 0$.

In conclusion, the solution to the convex optimization problem (12) asymptotically steers the tracking error to the equilibrium point e^* defined by

$$e^* = \begin{cases} 0 & \text{if } Z_{ref}(\tau) \in \mathbb{Z} \\ e_s & \text{if } Z_{ref}(\tau) \notin \mathbb{Z}. \end{cases}$$

V. SIMULATION AND EXPERIMENTAL RESULTS

This section illustrates the effectiveness of the proposed control algorithm, given by QP (12), through simulation using MATLAB software (2021b) and a real-time implementation using the Khepera III platform [39]. This section addresses the different performances achieved by the proposed algorithm in terms of the:

1- Achieved control performances (tracking performance and feasibility of the solution).
2- Processing time comparison.
3- Real-time implementation of the approximated NMPC.

For the simulation part, the control parameters Q, R, and h were tuned to determine the optimal values that achieved reasonable tracking performance. It is noticed that the step time h is a key parameter that should be chosen carefully to validate the Euler approximation used in this work. Therefore, many simulations have been performed to determine the adequate control parameters that achieve an admissible tracking performance.

A. CONTROL PERFORMANCE RESULTS

For this part, we have stated two different scenarios [19]:

I. Tracking an eight trajectory. In this case, the constraints on the y-axis and control signals are included in the optimization problem.
II. Tracking a circular trajectory in which the admissible region is a square-shaped space. In this case, both state and input constraints are included in the optimization problem.

For simplicity, we assume that the matrices Q and R are diagonals. After tuning the control parameters, the numerical values are selected as follows:

$$h = 0.1s; \quad Q = \begin{bmatrix} 10^4 & 0 & 0 \\ 0 & 10^4 & 0 \\ 0 & 0 & 1 \end{bmatrix};$$
$$Ts = 0.01s; \quad P = 2\,Q; \; d = 0.2; \; T = 0.4s; \; m = 4;$$
$$\mathrm{R} = \begin{bmatrix} 10^{-4} & 0 \\ 0 & 10^{-2} \end{bmatrix}.$$

With these control parameters, condition (15) is satisfied.

1) FIRST SCENARIO

The eight-shaped desired trajectory defined by

$$\begin{cases} x_{ref}(t) = x_c + R_1 \sin(\omega_1\, t) \\ y_{ref}(t) = y_c + R_2 \sin(\omega_2\, t), \end{cases} \quad t \in \left[0,\ t_f\right];$$

where $(x_c = 3, y_c = 3)$ represents the center of the eight-shaped trajectory. The reference trajectory of the orientation angle is deduced from

$$\theta_{ref}(t) = tan^{-1}\left(\frac{\dot{y}_{ref}}{\dot{x}_{ref}}\right), \quad t \in [0,\ t_f]$$

Trajectory parameters are

$$R_1 = R_2 = 2m; \quad \omega_r = 0.05\, rad/s; \; \omega_1 = \omega_r;$$
$$\omega_2 = 2\omega_r; \quad t_f = 42\pi\, s.$$

Because of the bounded velocity of each wheel, constraints on the control signals should be included in the optimization problem. In this simulation, the same bounds provided by the authors in [2] were used.

$$\Omega = \{0 \leq v(t) \leq 0.3\, m/s; -0.5\, rad/s \leq \omega(t) \leq 0.5\, rad/s\}.$$

The constrained space is delimited over the y-axis, where $F_R = \{y(t) \in \Re;\ 1.5m \leq y(t) \leq 4.5\, m,\ t \in [t, t+T]\}$.

Figure 3 shows the good tracking performance achieved by the proposed controller. In this scenario, the mobile robot is represented by a black triangle and the moving target using a green ball. It can be observed that the desired eight-shaped trajectory was precisely tracked inside the constrained space. The applied control signals, which are within the saturation limits, are shown in Figure 4 for t $\in$ [0, 75s]. Small oscillations in the steering angle were observed during start-up.

2) SECOND SCENARIO

In this case, the WMR must follow a circular trajectory generated by the planner. The circle-shaped trajectory is defined as follows

$$\begin{cases} x_{ref}(t) = x_c + R_o sin(\omega_o t) \\ y_{ref}(t) = y_c - R_o cos(\omega_o t) \\ \theta_{ref}(t) = \omega_o t \end{cases}$$

where $R_o = 2m;\ \omega_o = 0.05\frac{\text{rad}}{\text{s}}$.

The mobile robot must navigate inside the restrained region; therefore, it is necessary to determine the control

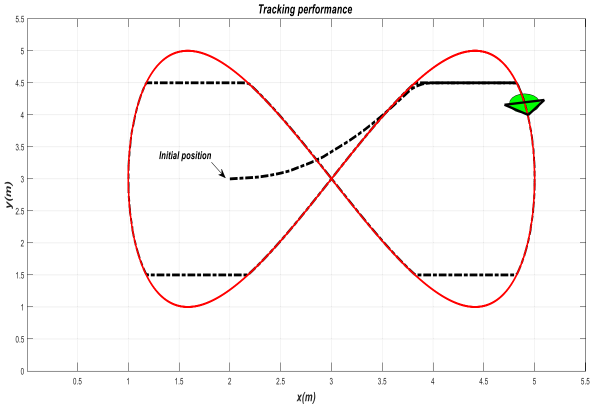

FIGURE 3. Tracking performance for the first scenario.

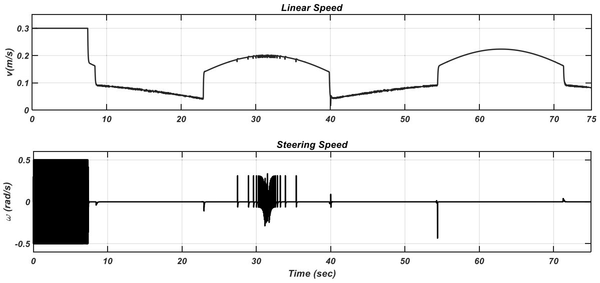

FIGURE 4. The time variation of the applied control signals.

vector $U(t)$ to track the desired trajectory only inside the free space. Thus, the following constraints are included in the optimization problem

$$F_R = \Big\{P_{xy}(t) \in \mathfrak{R}^2;\ 1.2m \le x(t) \le 4.8\,m \times \text{ and } 1.5\,m \le y(t) \le 4.5m\ ,\ t \in [t, t+0.4s]\Big\}.$$

The initial position of the WMR is $\left[2m\ 4m\ 0m\right]^T$ and the desired trajectory starts at $\left[x_{ref}(0) = 3m\ y_{ref}(0) = 1m\right]^T$.

The tracking performance of the WMR under the control algorithm (12) is shown in Figure 5. It was observed that when the desired trajectory was inside the feasible region, the robot followed it precisely. However, when the reference trajectory was outside the desired region (unfeasible), the mobile robot navigated close to the border of the admissible region.

Figure 6 illustrates the applied control signals. Therefore, the linear speed and steering angle are within the saturation limits. Here, oscillations were also observed at the beginning of the movement. Figure 7 shows that the tracking error converges to zero when the reference trajectory is inside the desired region and is different from zero when it is outside the desired region.

Furthermore, to show the effect of the feasibility of the desired trajectory, Figures 8 and 9 show the tracking performance for two different cases: the desired trajectory is feasible and completely inside the restricted region ($R = 1$m). In the second case, the desired trajectory is unfeasible and completely outside the restricted region ($R = 3$m),

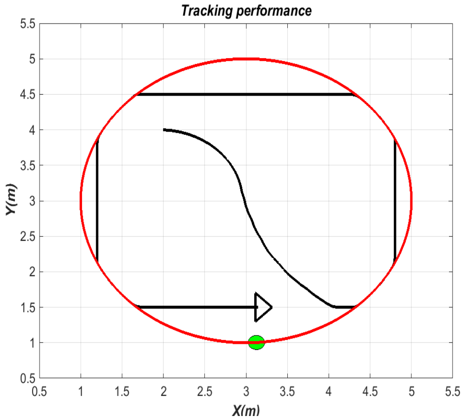

FIGURE 5. Tracking performance for the second scenario.

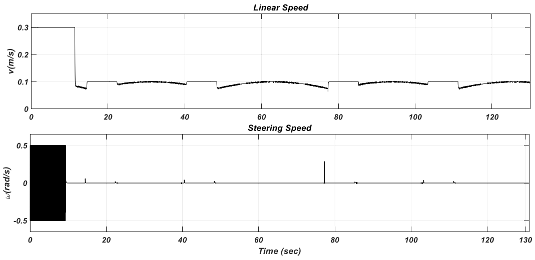

FIGURE 6. Optimal control signals.

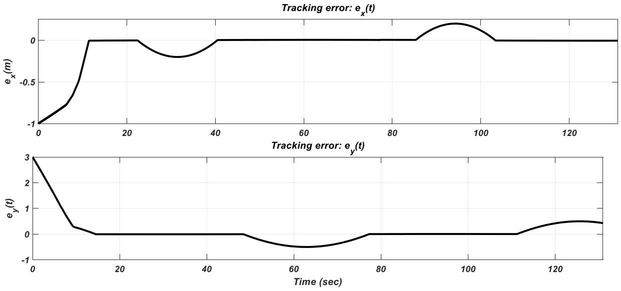

FIGURE 7. Tracking performances in Cartesian coordinates.

but the state constraints are satisfied. Figure 8 shows that the equilibrium point is the origin, and the tracking error is zero ($e^* = 0$), whereas in Figure 9, the equilibrium point is different from zero ($e^* \neq 0$).

B. EXECUTION TIME COMPARISON

The proposed NMPC formulation based on the approximation of NMPC (ANMPC) was compared with the processing time of the NLP formulation (4) and the algorithm that uses the CasADI tool to accelerate the operation of NMPC [40]. The NLP problem, termed by NMPC in this comparison part, is solved using MATLAB "*fmincon*" solver with SQP.

To perform a valid comparison, we used three algorithms with the same sampling time, prediction horizon, and control parameters. Furthermore, the regulation for the same target in free space was conducted for $t \in [0,\ 2s]$. During the

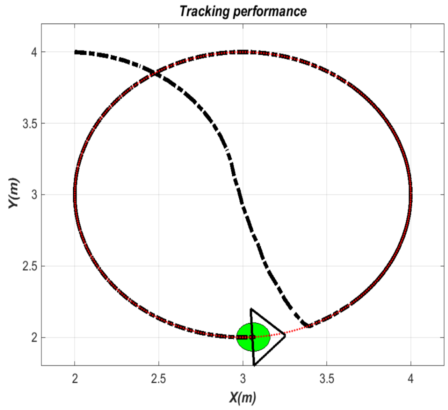

FIGURE 8. The desired trajectory is inside the restricted region.

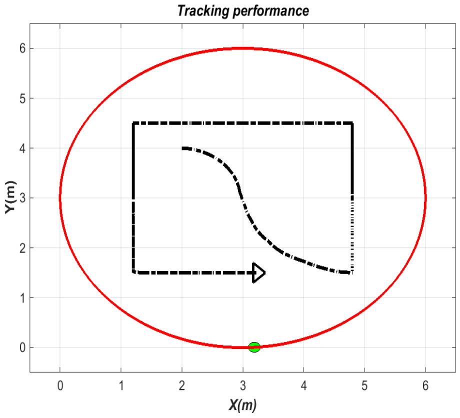

FIGURE 9. The desired trajectory is outside the restricted region.

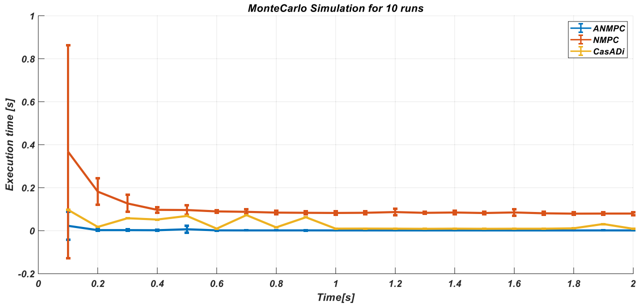

FIGURE 10. Computation time analysis of the three algorithms.

simulation, we noticed that the initial guess of the algorithm affected the processing time in the first steps of the simulation. To this end, a Monte Carlo simulation was utilized in these simulations with ten different runs. The initial vector estimate was sampled from the normal distribution $\mathbb{N}(0, 1)$.

The simulation results are shown in Figure 10. As clearly shown the proposed algorithm (ANMPC) achieves a better processing time (2.9 *ms*) per iteration.

Table 1 summarizes the mean and standard deviation of the processing time per iteration for the three algorithms. This comparison demonstrates that the ANMPC-based approach

TABLE 1. The mean and standard deviation of the execution time per iteration.

Method	Processing time per iteration	Standard deviation per iteration
NMPC	0.1062 s	0.1086
CASADI	0.0290 s	4.2564e-18
ANMP	0.0029 s	0.0148

FIGURE 11. Khepera III mobile robot.

is an attractive alternative for efficiently solving the NMPC problem.

C. EXPERIMENTAL RESULTS

In the second part, we consider the real-time implementation of the proposed algorithm on a real mobile robot platform. The experiment was conducted using the TOSHIBA laptop with Intel Core i7 processors. The platform was the mobile robot Khepera III (Figure 11), which is a differential drive mobile robot [39]. The Khepera platform is an automated differential-drive guided mobile robot designed and equipped for performing autonomous tasks. It is equipped with infrared, and ultrasonic sensors for navigation, and encoders for localization. The distance between the left and right wheels is equal to $L = 88.4$mm. The diameter of the wheels is 41mm. The connection between the platform and the laptop was established using Bluetooth communication technology. The MATLAB software was also used to implement the proposed algorithm because MATLAB functions support communication with devices via Bluetooth.

A scenario of tracking a circular trajectory in constrained space was adopted in this experiment. The reference trajectory was the same as that used in the simulation, with the following parameters:

$$x_c = 0; \quad y_c = 0; \; R_o = 650\,mm, \; \omega_o = 2 * \mathrm{pi}/35; \qquad t \in [0s\,,\,146s]\,.$$

Constraints on control signals are:

$$v_{max} = 300\frac{mm}{s}; \quad v_{min} = 0; \; \omega_{max} = 5\frac{rad}{s}; \; \omega_{min} = -5\frac{rad}{s}.$$

Space constraints are formulated as follows:

$$F_R = \Big\{P_{xy}(t) \in \Re^2; \;\; -580\,mm \le x(t)$$

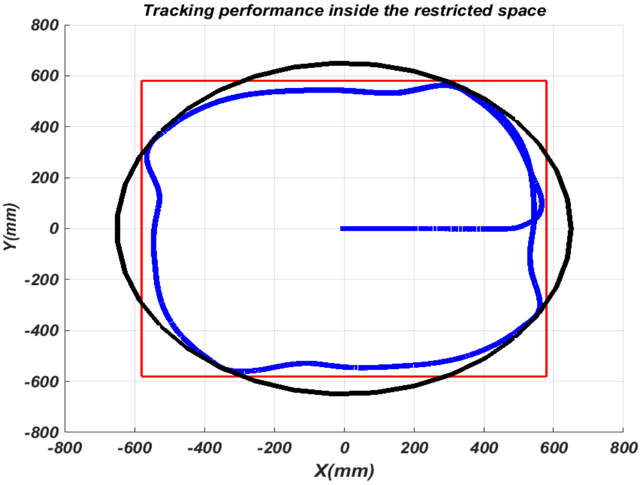

FIGURE 12. Trajectory tracking performance in the constrained square.

$$\leq 580\,mm \text{ and } -580\,mm \leq y(t) \leq 580\,mm,\ t \in [0s\ ,\ 146s]\Big\}$$

Because of the presence of noisy measurements, and the inaccuracies that arise from residual errors (quantization of the velocity controls), the constrained space used by the control algorithm is diminished with regard to the real constrained region (hard constraints). Consequently, to increase safety, the soft constraints used in the algorithm are as follows:

$$Fs_R = \Big\{P_{xy}(t) \in \Re^2;\ -550\,mm \leq x(t) \leq 550\,mm \times \text{and } -550\,mm \leq y(t) \leq 550\,mm,\ \ t \in [0s\ ,\ 146s]\Big\}$$
$$\text{where } Fs_R \subset F_r.$$

The robot starts from the feasible region, which is given by the position: $\left[x(0)\ y(0)\ \theta(0)\right] = \left[0\ 0\ 0\right]^T$ and the desired trajectory (also feasible) starts at $\left[x_{ref}(0) = 650\,mm\ \ y_{ref}(0) = 0mm\ \theta_{ref}(0) = 0^o\right]^T$. The numerical values of the control parameters for this experiment are as follows:

$$Q = 10^3 I_3;\quad R = 10^{-3} I_2;\ h = 0.1s;$$
$$d = 80\,mm;\quad m = 4;\ T = 0.4s$$

Figures 12-14 show the experimental results obtained by solving the constrained quadratic optimization problem given by (12).

Figure 12 shows the desired trajectory (circle), restricted space region (square), and the trajectory of the mobile robot. The tracking of the reference trajectory inside the restricted space is accurate and when the desired trajectory is outside the restricted space, the mobile robot remains close to the border of the square.

Figure 13 shows the solution obtained from the constrained optimization problem. The control signals, linear velocity, and steering velocity were within the saturation limits.

The command signals sent to the WMR are the velocities of the right and left wheels$(v_R,\ v_L)$, which are deduced from the following equation

$$\begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix} = \begin{bmatrix} \frac{1}{2}\left(v_R(t) + v_L(t)\right) \\ \frac{1}{L}\left(v_R(t) - v_L(t)\right) \end{bmatrix} \Rightarrow \begin{bmatrix} v_R(t) \\ v_L(t) \end{bmatrix} = T \begin{bmatrix} v(t) \\ \omega(t) \end{bmatrix},$$

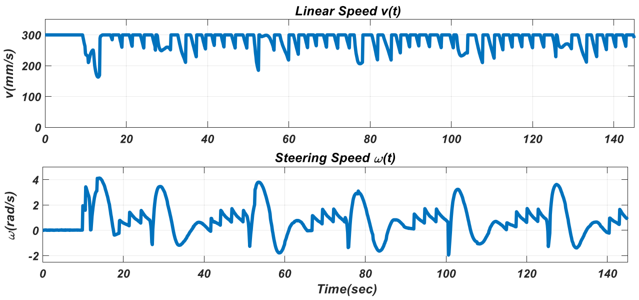

FIGURE 13. Linear velocity and angular velocity.

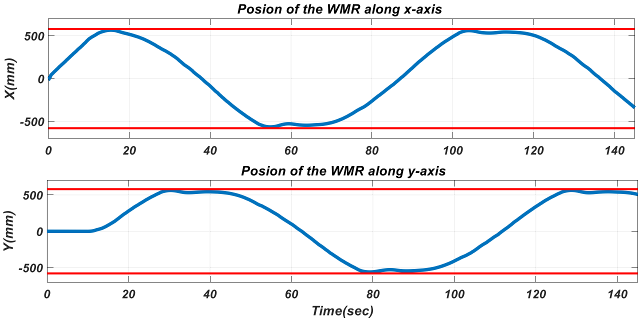

FIGURE 14. The current evolution of the WMR along x-y axes.

Figure 14 illustrates the Cartesian positions of the mobile robot along the *x-axis* and *y-axis*. It is clear that the mobile robot navigates practically inside the restricted region and that the constrained optimization problem has forced the mobile robot to move inside the tolerated space, which is a square in this case.

Finally, we note that the main drawback of the proposed control algorithm is finding adequate tuning control parameters ($h,\ Q,\ R$, etc.) that satisfy the desired tracking performance for different tracking trajectories.

VI. CONCLUSION

In this study, an approximated nonlinear model predictive control was applied to the autonomous navigation of a differential-drive mobile robot in a constrained region. Euler approximation was used to overcome the nonconvex and nonlinear optimization problem in the control design. The stability of the closed-loop system is determined using the Lyapunov method. The autonomous navigation of the mobile robot in the constrained region was formulated as a quadratic optimization problem or QP where both physical constraints (control vector) and space constraints were included in the design of the controller. Therefore, all the constraints are explicitly included in the optimization problem. Thus, the obtained constrained optimization problem is convex with a fast update and many numerical solutions exist that can efficiently solve this type of optimization problem. It was proven that the solution is feasible and that the trajectory tracking error is bounded and may converge to the origin when the desired trajectory navigates inside the feasible region.

The proposed method offers many advantages over existing nonlinear methods, including simplicity, fast computation, and the ability to include different constraints in the controller design.

Simulations and experimental results demonstrated the effectiveness of the proposed control algorithm and good tracking performance was achieved.

Future work will focus on the application of the proposed approach to three-dimensional vehicles, such as aerial unmanned vehicles.

APPENDIX

PROOF OF THEOREM 1

Let the initial solution $U^*(t)$ be feasible, and let the equivalent optimal cost function (9) be given in the form of $J\left(t, e(t), U^*(t), h\right)$ at time t. For the next step, we have

$$J\left(t+h, e\left(t+h\right), \tilde{}U(t+h), h\right) - J\left(t, e(t),\ U^*(t), h\right)$$

where

$$U^*(t)=\left[u^*(t)\ u^*(t+h)\ \cdots\ u^*(t+(m-1)h)\right]^T$$

$$\tilde{U}\left(t+h\right)=\left[u*(t+h)\ \cdots\ u*(t+(m-1)h)\ u_F(t+mh)\right]^T$$

$$J\left(t+h, e\left(t+h\right), \tilde{U}(t+h), h\right) - J\left(t, e(t),\ U^*(t), h\right)$$
$$= -L\left(e(t), u^*(t)\right) + P_T(t, m)$$

$$P_T\left(t, m\right) = L\left(e\left(t+mh\right), u_F\left(t+mh\right)\right) + e\left(t+h+mh\right)^T$$
$$\times Pe\left(t+mh+h\right) - e\left(t+mh\right)^T Pe\left(t+mh\right),$$

which can be transformed into the quadratic form using the terminal controller

$$P_T\left(t, m\right) = e(t+mh)^T\left\{Q - 2Q\,G\left(G^TQG\right)^{-1}G^TP\right.$$
$$+ Q\,G(G^TQG)^{-1}\left(\frac{R}{h^2} + G^TPG\right)$$
$$\left.\left(G^TQG\right)^{-1}G^TQ\right\}e\left(t+mh\right)$$
$$= e\left(t+mh\right)^T\prod e(t+mh)$$

The following equality is obtained

$$G^T\prod G = G^TQG - G^TPG + \frac{R}{h^2}$$

From the definition (1), the following inequality is obtained

$$\sqcap < 0 \iff G^TQG - G^TPG + \frac{R}{h^2} < 0,$$

Consequently, function $P_T\left(t, m\right)$is negative. This inequality can be simplified if we take diagonal matrices as $P = p\,I_3; Q = q\,I_3,$ and $R = r\,I_2$, then the previous inequality becomes:

$$p > q + \frac{r}{h^2(d^2+1)}.$$

The cost $J\left(t+h, e\left(t+h\right), \tilde{U}(t+h), h\right)$ is feasible and not necessarily optimal, and we have

$$J\left(t+h, e\left(t+h\right),\ U^*(t+h), h\right)$$
$$< J\left(t+h, e\left(t+h\right), \tilde{U}\left(t+h\right), h\right)$$

Thus, the optical cost (6) is decreasing since

$$J\left(t+h, e\left(t+h\right),\ U^*(t+h), h\right) - J\left(t, e(t),\ U^*(t), h\right) < 0$$

Further, all sublevel sets of the optimal cost are bounded. Consequently, the optimal cost function is a Lyapunov function [41].

ACKNOWLEDGMENT

The author would like to thank the Research Center of the College of Computer and Information Sciences at King Saud University.

REFERENCES

[1] B. Siciliano and O. Khatib, Eds., *Springer Handbook of Robotics*, 2nd ed. Cham, Switzerland: Springer, 2016.

[2] G. Oriolo, A. D. Luca, and M. Vendittelli, "WMR control via dynamic feedback linearization: Design, implementation, and experimental validation," *IEEE Trans. Control Syst. Technol.*, vol. 10, no. 6, pp. 835–852, Nov. 2002.

[3] A. Bloch and S. Drakunov, "Tracking in nonholonomic dynamic systems via sliding modes," in *Proc. 34th IEEE Conf. Decis. Control*, Dec. 2002, pp. 2103–2106.

[4] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot: Backstepping kinematics into dynamics," in *Proc. 34th IEEE Conf. Decis. Control*, Dec. 1995, pp. 3805–3810.

[5] S. M. Ahmadi, M. B. Taghadosi, and A. Haqshenas M, "A state augmented adaptive backstepping control of wheeled mobile robots," *Trans. Inst. Meas. Control*, vol. 43, no. 2, pp. 434–450, Jan. 2021.

[6] S. Li, L. Ding, H. Gao, C. Chen, Z. Liu, and Z. Deng, "Adaptive neural network tracking control-based reinforcement learning for wheeled mobile robots with skidding and slipping," *Neurocomputing*, vol. 283, pp. 20–30, Mar. 2018.

[7] L. Ding, L. Huang, S. Li, H. Gao, H. Deng, Y. Li, and G. Liu, "Definition and application of variable resistance coefficient for wheeled mobile robots on deformable terrain," *IEEE Trans. Robot.*, vol. 36, no. 3, pp. 894–909, Jun. 2020.

[8] R. Findeisen, F. Allgöwer, and L. T. Biegler, Eds., *Assessment and Future Directions of Nonlinear Model Predictive Control* (Lecture Notes in Control and Information Sciences), vol. 358. Berlin, Germany: Springer, 2007.

[9] L. Magni, D. M. Raimondo, and F. Allgöwer, Eds., *Nonlinear Model Predictive Control: Towards New Challenging Applications* (Lecture Notes in Control and Information Sciences), vol. 384. Berlin, Germany: Springer, 2009.

[10] S. Subramanian, Y. Abdelsalam, S. Lucia, and S. Engell, "Robust tube-enhanced multi-stage NMPC with stability guarantees," *IEEE Control Syst. Lett.*, vol. 6, pp. 1112–1117, 2022.

[11] L. Xie, H. Su, S. Lu, and L. Xie, "Linear mismatched model based offset-free MPC for nonlinear constrained systems with both stochastic and deterministic disturbances and its application to CSTR," *IEEE Access*, vol. 6, pp. 69172–69184, 2018.

[12] F. Zhou, H. Peng, G. Zhang, X. Zeng, and X. Peng, "Robust predictive control algorithm based on parameter variation rate information of functional-coefficient ARX model," *IEEE Access*, vol. 7, pp. 27231–27243, 2019.

[13] N. Saraf, M. Zanon, and A. Bemporad, "A fast NMPC approach based on bounded-variable nonlinear least squares," in *Proc. 6th IFAC Conf. Nonlinear Model Predictive Control*, Madison, WI, USA, vol. 51, no. 20, Aug. 2018, pp. 337–342.

[14] A. Nurkanovic, A. Zanelli, S. Albrecht, and M. Diehl, "The advanced step real time iteration for NMPC," in *Proc. IEEE 58th Conf. Decis. Control (CDC)*, Dec. 2019, pp. 11–13.

[15] A. Zanelli, R. Quirynen, J. Jerez, and M. Diehl, "A homotopy-based nonlinear interior-point method for NMPC," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 13188–13193, Jul. 2017.

[16] H. Deng and T. Ohtsuka, ‘‘A parallel Newton-type method for nonlinear model predictive control,’’ *Automatica*, vol. 109, Nov. 2019, Art. no. 108560.

[17] R. Verschueren, G. Frison, D. Kouzoupis, N. Duijkeren, A. Zanelli, R. Quirynen, and M. Moritz, ‘‘Towards a modular software package for embedded optimization,’’ in *Proc. 6th IFAC Conf. Nonlinear Model Predictive Control (NMPC)*, vol. 51, no. 20. Madison, WI, USA, Aug. 2018, pp. 374–380.

[18] H. Deng and T. Ohtsuka, ‘‘ParNMPC—A parallel optimisation toolkit for real-time nonlinear model predictive control,’’ *Int. J. Control*, vol. 95, no. 2, pp. 390–405, Feb. 2022.

[19] T. P. Nascimento1, C. E. T. Dorea, and L. M. G. Goncalves, ‘‘Nonlinear model predictive control for trajectory tracking of nonholonomic mobile robots: A modified approach,’’ *Int. J. Adv. Robotic Syst.*, vol. 15, Jan./Feb. 2018, Art. no. 1729881418760461.

[20] A. Zanelli, J. Kullick, H. M. Eldeeb, G. Frison, C. M. Hackl, and M. Diehl, ‘‘Continuous control set nonlinear model predictive control of reluctance synchronous machines,’’ *IEEE Trans. Control Syst. Technol.*, vol. 30, no. 1, pp. 130–141, Jan. 2022.

[21] A. S. Conceição, A. P. Moreira, and P. J. Costa, ‘‘A nonlinear model predictive control strategy for trajectory tracking of a four-wheeled omnidirectional mobile robot,’’ *Optim. Control Appl. Methods*, vol. 29, no. 5, pp. 335–352, Sep. 2008.

[22] H. N. Huynh, O. Verlinden, and A. V. Wouwer, ‘‘Comparative application of model predictive control strategies to a wheeled mobile robot,’’ *J. Intell. Robotic Syst.*, vol. 87, no. 1, pp. 81–95, Jul. 2017.

[23] U. Rosolia, S. de Bruyne, and A. G. Alleyne, ‘‘Autonomous vehicle control: A nonconvex approach for obstacle avoidance,’’ *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 2, pp. 469–484, Mar. 2017.

[24] I. Sánchez, A. D’Jorge, G. V. Raffo, A. H. González, and A. Ferramosca, ‘‘Nonlinear model predictive path following controller with obstacle avoidance,’’ *J. Intell. Robotic Syst.*, vol. 102, no. 1, pp. 1–18, Apr. 2021.

[25] K. Worthmann, M. Mehrez, M. Zanon, and G. Mann, ‘‘Model predictive control of nonholonomic mobile robots without stabilizing constraints and costs,’’ *IEEE Trans. Control Syst. Technol.*, vol. 24, no. 4, pp. 1394–1406, Jul. 2016.

[26] M. Seder, M. Baotic, and I. Petrovic, ‘‘Receding horizon control for convergent navigation of a differential drive mobile robot,’’ *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 2, pp. 653–660, Mar. 2017.

[27] X. Du, K. K. K. Htet, and K. K. Tan, ‘‘Development of a genetic-algorithm-based nonlinear model predictive control scheme on velocity and steering of autonomous vehicles,’’ *IEEE Trans. Ind. Electron.*, vol. 63, no. 11, pp. 6970–6977, Nov. 2016.

[28] Z. Wang, G. Li, H. Jiang, Q. Chen, and H. Zhang, ‘‘Collision-free navigation of autonomous vehicles using convex quadratic programming-based model predictive control,’’ *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 3, pp. 1103–1113, Jun. 2018.

[29] M. N. Zeilinger, C. N. Jones, and M. Morari, ‘‘Real-time suboptimal model predictive control using a combination of explicit MPC and online optimization,’’ *IEEE Trans. Autom. Control*, vol. 56, no. 7, pp. 1524–1534, Jul. 2011.

[30] V. Valls and D. J. Leith, ‘‘A convex optimization approach to discrete optimal control,’’ *IEEE Trans. Autom. Control*, vol. 64, no. 1, pp. 35–50, Jan. 2019.

[31] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[32] Y. Wang and S. Boyd, ‘‘Fast model predictive control using online optimization,’’ *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 2, pp. 267–278, Mar. 2010.

[33] R. Hedjar, ‘‘Application of nonlinear rescaling method to model predictive control,’’ *Int. J. Control, Autom. Syst.*, vol. 8, no. 4, pp. 762–768, Aug. 2010.

[34] B. Maury and J. Vend, ‘‘Handling of contacts in crowd motion simulations,’’ in *Traffic Granular Flow 07*, vol. 1. Berlin, Germany: Springer, 2007, pp. 171–180.

[35] R. Hedjar and M. Bounkhel, ‘‘Real-time obstacle avoidance for a swarm of autonomous mobile robots,’’ *Int. J. Adv. Robotic Syst. (IJARS)*, vol. 11, no. 67, pp. 1–12, 2014.

[36] D. Q. Mayne and P. Falugi, ‘‘Stabilizing conditions for model predictive control,’’ *Int. J. Robust Nonlinear Control*, vol. 29, no. 4, pp. 894–903, 2019.

[37] B. R. James, D. Q. Mayne, and M. Diehl, *Model Predictive Control: Theory Computation and Design*, 2nd ed. Madison, WI, USA: Nob Hill Publishing, LLC, 2018.

[38] I. J. Wolf and W. Marquardt, ‘‘Fast NMPC schemes for regulatory and economic NMPC—A review,’’ *J. Process Control*, vol. 44, pp. 162–183, Aug. 2016.

[39] Accessed: Nov. 1, 2018. [Online]. Available: www.k-team.com

[40] Accessed: Jul. 20, 2020. [Online]. Available: https://github.com/MMehrez/MPC-and-MHE-implementation-in-MATLAB-using-Casadi.git

[41] H. K. Khalil, *Nonlinear Control*, 1st ed. London, U.K.: Pearson, 2014.

RAMDANE HEDJAR (Senior Member, IEEE) received the Ph.D. degree in electronic and electrical engineering from the USTHB University, Algeria, in collaboration with SUPELEC, Paris, France, in 2002.

He joined the Computer Engineering Department at King Saud University as an Assistant Professor. From 1992 to 2000, he was a Lecturer in the Department of Electronics at Djelfa University. He is currently a Professor with the Department of Computer Engineering, King Saud University. His research interests include robust control, nonlinear predictive control, robotics, neural network control, and networked control systems.