

Dynamic Obstacle Avoidance Using Nonlinear Model Predictive Control and Control Barrier Function for Ballbot Systems

Pham Minh Duc¹, Vu Duc Cuong¹, Nguyen Thi Thuy Hang^{1,2}, Nguyen Danh Huy¹,
Nguyen Thi Van Anh¹, Nguyen Tung Lam^{1*}

¹Hanoi University of Science and Technology, Ha Noi, Vietnam

²Thuy Loi University, Ha Noi, Vietnam

*Corresponding author email: lam.nguyentung@hust.edu.vn

Abstract

This research presents a tracking control system for a ballbot designed to operate in complex environments filled with both static and dynamic obstacles. The Nonlinear Model Predictive Control (NMPC) framework is formulated to predict the future positions of the ballbot and all surrounding obstacles. This predictive capability is crucial for effective navigation, as it allows the ballbot to anticipate potential collisions in the prediction horizon. The NMPC is integrated with an optimization problem that is enhanced by Control Barrier Function (CBF) constraints. These constraints ensure that the ballbot maintains a safe and consistent distance from every obstacle, thus preventing collisions. Additionally, an Extended State Observer (ESO) is implemented to observe and compensate for uncertain disturbances in the ballbot's movements, as well as to estimate immeasurable variables that might affect its performance. Various simulation scenarios are conducted to thoroughly test and validate the effectiveness of this approach in achieving precise tracking control and reliable collision avoidance in environments with a large number of obstacles.

Keywords: Ballbot, control barrier function, model predictive control, obstacles avoidance.

1. Introduction

Ballbot is a self-balancing robot designed based on the ideal of inverted pendulum motion. Ballbot is designed with many transmission mechanisms such as using inverse mouse-ball drive and rollers [1, 2]; using three omnidirectional wheels [3, 4];... but in general, the movements of ballbot are based on adjusting the position of a spherical ball in all directions and keeping the robot body balanced above that sphere. Thanks to the unique design of the moving mechanism, the ballbot moves only based on a single contact point with the surface, making this robot more flexible than other types of mobile robots. Some significant advantages can be listed as the ability to perform complex movements (turn left, turn right, turn around, etc.) in narrow environments and the ability to self-adjust the angle of the ballbot's body to maintain balance even when carrying objects or moving on inclined planes.

Although the ballbot demonstrates superior flexibility compared to other types of mobile robots, tracking and balancing control of the ballbot is still a considerable challenge due to its kinematic complexity. In most previous research, the authors have proposed the following main control methods. The first method is to linearize the ballbot model around the equilibrium point, i.e. consider state variables in balance position to approximate a linear

model and apply linear control theories for ballbot [1-3, 4-6]. Another method is to project the 3D-ballbot model into three planar models, then the controller will be designed for each ballbot model in 2D space and synthesized through a torque transformation [7-9]. These approaches illustrate good responses in balancing tasks. However, when the ballbot moves, the velocity will not be in the equivalent region, and linear controllers will no longer be effective due to the significant approximations. For the model separation method, controlling the heading angle is complicated because when changing the direction angle, some infinity components can appear and make this torque conversion matrix non-invertible.

Therefore, in this paper, Nonlinear Model Predictive Control (NMPC) is applied to overcome the above disadvantages. Besides the advantage that NMPC can control the 3D model directly by optimizing the objective function without the need for linearization or model conversion, MPC also allows for handling other complex system conditions in the form of optimization problem constraints [10]. Leveraging those advantages, we will use MPC as the central controller, whose main task is to minimize the target function while still ensuring compliance with safety conditions and obstacles. To build these collision avoidance conditions for each obstacle, we use the Control Barrier Function (CBF), initially

introduced by Ames *et al.* in [11, 12]. Instead of using the normal constraint that the distance from the ballbot to the obstacles is greater than the desired safe distance, CBF is formulated based on a safe set of distance constraints. The author in [13] has given certain conditions to determine the set consisting of all control values that render the safe set forward invariance, i.e. the safety conditions of the system will always be guaranteed throughout the moving process when applying this control signal. In addition, the states of the ballbot after a certain period of time are also predicted in advance by predictive model of NMPC. These future values are compared with the positions of obstacles and the system anticipates future collisions and makes appropriate adjustments in control signals based on CBF.

In situations where not all states can be directly measured due to sensor limitations or inherent unobservability, the Extended State Observer (ESO) steps in to estimate these states based on available measurements. Moreover, ESO is designed to estimate an additional “extended state,” called the total disturbance and includes all uncertainties, disturbances, or external impacts on the system. Motivated by [14, 15] as active disturbance rejection controls, we decided to use the ESO as the observer of NMPC in the control structure.

This study delineates three main contributions:

- Presenting a detailed mathematical model of the ballbot system based on solving the Euler-Lagrange equation.
- Introducing a control strategy combining the Model Predictive Control and Control Barrier Function based on Extended State Observer for the ballbot system operating in dynamic environments.
- Simulating the movement of ballbot in complex environments with moving obstacles and disturbances.

The rest of this paper includes 4 sections: Section 2 introduces the 3D-ballbot system model using the Euler-Lagrange equation, the next section proposes the control structure using ESO-NMPC and CBF, some simulation scenarios are implemented in Section 4 to determine the effectiveness of our proposed controller. Finally, Section 5 is the conclusion and future work.

2. Ballbot System Model

2.1. Assumptions and Coordinate Definition

To facilitate model formulation, some ideal assumptions of component interactions and working environment are considered as follows:

- The Omni wheels, the ball, and the ground are rigid;

- Interactions between ball and ground, ball and Omni wheels are point contacts and anti-slipping;
- There is no delay in the responses of actuators;
- The operating surface are horizontal plane.

The energy of ballbot is formulated based on the state variables and their velocities on different coordinates. We define these coordinate conservations as Fig. 1, where frame $\{b\}$ is the translation frame of inertial frame $\{I\}$ and fixed to the centre of the ball, frame $\{b'\}$ are obtained when rotating the frame $\{b\}$ around its center, frame $\{a'\}$ is rigidly fixed in the body at center of mass (COM) and frame $\{a\}$ is translated from frame $\{a'\}$ alongside a'_z axis to the COM of the ball.

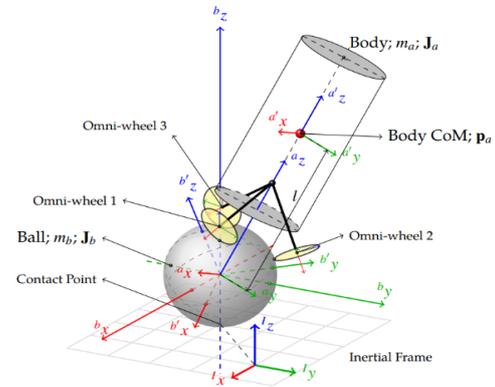


Fig. 1. Ballbot coordinate definition

2.2. Equation of Motion

We define x, y are ball position in xy – plane of the inertial frame $\{I\}$, ϕ_x, ϕ_y and ϕ_z are the roll-pitch-yaw Euler angles of the body in frame $\{b\}$. The generalized coordinate of the ballbot is selected as:

$$\mathbf{q} = [x \quad y \quad \phi_x \quad \phi_y \quad \phi_z]^\top \quad (1)$$

The kinetic energy of the ball includes the rotation kinetic energy, and translation kinetic energy is given as follows:

$$T_b = \frac{1}{2} \dot{\mathbf{p}}_b^\top m_b \dot{\mathbf{p}}_b + \frac{1}{2} \dot{\boldsymbol{\phi}}^\top \mathbf{J}_b \dot{\boldsymbol{\phi}} \quad (2)$$

where, $\mathbf{p}_b = [x \quad y \quad 0]^\top$; $\boldsymbol{\phi}$ is the rotation of the ball; m_b, J_b and $\mathbf{J}_b = \text{diag}(J_b \quad J_b \quad J_b)$ are the mass, the moment of inertia in one direction and the moment of inertia vector, respectively. Then, the potential energy of the ballbot is calculated by:

$$V_b = [0 \quad 0 \quad g] m_b \mathbf{p}_b = 0 \quad (3)$$

The position of the body's COM is found by rotation of the frame $\{a\}$ to $\{b\}$, that is

$$\mathbf{p}_a = \mathbf{R}_{a \rightarrow b} \begin{bmatrix} 0 & 0 & l \end{bmatrix}^\top \quad (4)$$

where, l is the distance from the COM of the ball to the COM of the body, and $\mathbf{R}_{a \rightarrow b}$ is the rotation matrix from frame $\{a\}$ to frame $\{b\}$, it is defined by rotation of the frame $\{b\}$ sequence as follows

$$\mathbf{R}_{a \rightarrow b} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi_x & -\sin \phi_x \\ 0 & \sin \phi_x & \cos \phi_x \end{bmatrix} \dots \begin{bmatrix} \cos \phi_y & 0 & \sin \phi_y \\ 0 & 1 & 0 \\ -\sin \phi_y & 0 & \cos \phi_y \end{bmatrix} \begin{bmatrix} \cos \phi_z & -\sin \phi_z & 0 \\ \sin \phi_z & \cos \phi_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5)$$

The rotation state vector of the body is defined as $\Phi = [\phi_x \ \phi_y \ \phi_z]^\top$ and the kinetic and potential energy of body-wheels is:

$$\begin{aligned} T_a &= \frac{1}{2} \dot{\mathbf{p}}_a^\top m_a \dot{\mathbf{p}}_a + \frac{1}{2} \dot{\Phi}^\top \mathbf{J}_a \dot{\Phi} \\ &= \frac{1}{2} \left(\dot{\mathbf{R}}_{a \rightarrow b} \begin{bmatrix} 0 \\ 0 \\ l \end{bmatrix} \right)^\top m_a \dot{\mathbf{R}}_{a \rightarrow b} \begin{bmatrix} 0 \\ 0 \\ l \end{bmatrix} + \frac{1}{2} \dot{\Phi}^\top \mathbf{J}_a \dot{\Phi} \end{aligned} \quad (6)$$

$$V_a = [0 \ 0 \ g] m_a \mathbf{p}_a = m_a g l \cos \theta_x \cos \theta_y \quad (7)$$

where m_a and \mathbf{J}_a are the total mass and moment of inertia of rigid body ball- wheels, respectively.

Each omniwheel has the individual rotation energy around the yaw axis and can be formulated:

$$T_c = \frac{1}{2} \dot{\Psi}^\top \mathbf{J}_c \dot{\Psi} \quad (8)$$

where Ψ is the rotation velocities of the omniwheels, $\mathbf{J}_c = \text{diag}(J_c, J_c, J_c)$ is the moment of inertial of the omniwheel. Then, based on the Lagrangian mechanics, the dynamic behavior of the ballbot can be described in the following:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\mathbf{q}}} - \frac{\partial L}{\partial \mathbf{q}} = \mathbf{Q} \mathbf{u} \quad (9)$$

where $L = T_a + T_b + T_c - V_a - V_b$ is the Lagrange function and $\mathbf{u} = [\tau_1 \ \tau_2 \ \tau_3]^\top$ is input torque vector. Equation (9) can be also described in the matrix form:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{D}(t, \mathbf{q}) = \mathbf{Q}(\mathbf{q})\mathbf{u} \quad (10)$$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^5$ is mass matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^5$ is Coriolis matrix, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^5$ is gravity matrix, $\mathbf{Q} \in \mathbb{R}^{5 \times 3}$ is Jacobian matrix and $\mathbf{D}(t, \mathbf{q}) \in \mathbb{R}^5$ is disturbance vector.

3. Control Structure

3.1. Nonlinear Model Predictive Control (NMPC)

NMPC is a type of optimal control method with a cost function built from control signals as well as errors between the response and the desired value. The optimization problem minimizes this cost function while still ensuring all constraint conditions. For conventional MPC and NMPC, the most necessary constraint is the dynamic equation of the system. The main difference between MPC and NMPC lies in the predictive model, the use of nonlinear predictive model and nonlinear constraints of NMPC bring higher accuracy and efficiency compared to MPC.

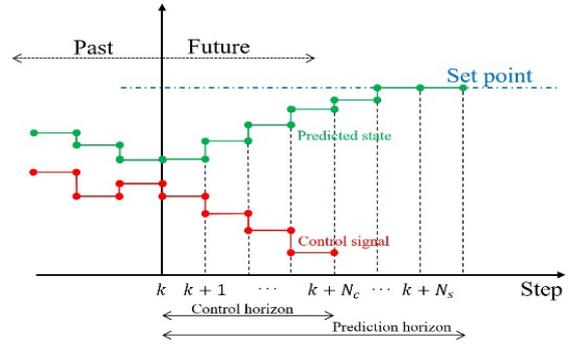


Fig. 2. NMPC principle

The basic principle of NMPC is depicted in Fig. 2. By getting measurement values at time t_0 , the future state can be determined in the prediction horizon of time T based on the system model. A cost function must be given to minimize the control signal \mathbf{u} for the next control horizon T_c ($T_c \leq T$). The larger the prediction horizon and control horizon, the more future state variables can be predicted. Still, it will increase the complexity and processing time of the optimization problem as well as waste computation when unexpected situations arise. This process is worked iteratively to update future states continuously, and the control signals calculated at this step can be applied to all subsequent steps if the model accuracy is guaranteed. Based on the equation of motion (10), the control signal \mathbf{u} is given by the solution of the optimization problem:

$$\begin{aligned} \mathbf{u} &= \arg \min_{\mathbf{u}} \int_{t_0}^{t_0+T} (\mathbf{e}(\tau)^\top \mathcal{R} \mathbf{e}(\tau) + \mathbf{u}(\tau)^\top \mathcal{U} \mathbf{u}(\tau)) d\tau \\ \text{s.t. } \ddot{\mathbf{q}} &= \mathbf{M}(\mathbf{q})^{-1} (\mathbf{Q}(\mathbf{q})\mathbf{u} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{G}(\mathbf{q})) \end{aligned} \quad (11)$$

where \mathbf{u} is the control signal, T is the time of predictive horizon; $\mathcal{R} = \mathcal{R}^\top > 0$, $\mathcal{T} = \mathcal{T}^\top > 0$ are the weight matrices of NMPC and $\mathbf{e}(t) = \mathbf{q}(t) - \mathbf{q}_r(t)$ is the error of states and reference \mathbf{q}_r .

3.2. NMPC with Extended State Observer (ESO-NMPC)

To achieve effective future prediction, NMPC must receive feedback states from the system model. However, the mathematical model of the system also includes uncertain parameters and disturbances as well as lack of measurement ability. Therefore, the optimization controller must be implemented step by step in infinity loops, and ESO can be used as a high-performance observer for handling uncertain disturbances and hard-to-observe variables in ballbot system. Consider the ballbot system (10) with $\mathbf{q}_1 = \mathbf{q}$ and $\mathbf{q}_2 = \dot{\mathbf{q}}$, we obtain:

$$\begin{aligned} \dot{\hat{\mathbf{q}}}_1 &= \mathbf{q}_2 \\ \dot{\hat{\mathbf{q}}}_2 &= \mathbf{M}^{-1}\mathbf{Q}\mathbf{u} - \mathbf{M}^{-1}(\mathbf{C}\mathbf{q}_2 + \mathbf{G} + \mathbf{D}(t, \mathbf{q}_1)) \end{aligned} \quad (12)$$

The linear ESO [14] for system observation state variables is designed as follows

$$\begin{aligned} \dot{\hat{\mathbf{q}}}_1 &= \hat{\mathbf{q}}_2 + \alpha_1 \frac{\mathbf{q}_1 - \hat{\mathbf{q}}_1}{\varepsilon} \\ \dot{\hat{\mathbf{q}}}_2 &= \hat{\mathbf{q}}_3 + \alpha_2 \frac{\mathbf{q}_1 - \hat{\mathbf{q}}_1}{\varepsilon^2} + \mathbf{M}^{-1}\mathbf{Q}\mathbf{u} \\ \dot{\hat{\mathbf{q}}}_3 &= \alpha_3 \frac{\mathbf{q}_1 - \hat{\mathbf{q}}_1}{\varepsilon^3} \end{aligned} \quad (13)$$

where $\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2$ and $\hat{\mathbf{q}}_3$ are the estimated values of $\mathbf{q}_1, \mathbf{q}_2$ and the total disturbance is defined as $\mathbf{q}_3 = \mathbf{f} = -\mathbf{M}^{-1}(\mathbf{C}\mathbf{q}_2 + \mathbf{G}) - \mathbf{M}^{-1}\mathbf{D}$, respectively.

Theorem 1: By assuming that the total disturbance $\|\dot{\mathbf{f}}\| \ll f_{d\max}$ ($f_{d\max} > 0$) and the chosen parameters $\alpha_1, \alpha_2, \alpha_3, \varepsilon > 0$ satisfying

$$\mathbf{H} = \begin{bmatrix} -\alpha_1 & 1 & 0 \\ -\alpha_2 & 0 & 1 \\ -\alpha_3 & 0 & 0 \end{bmatrix} \text{ is Hurwitz,} \quad (14)$$

then the error of estimated ESO variables $\mathbf{e} = [(\hat{\mathbf{q}}_1 - \mathbf{q}_1)^\top \quad (\hat{\mathbf{q}}_2 - \mathbf{q}_2)^\top \quad (\hat{\mathbf{q}}_3 - \mathbf{q}_3)^\top]^\top$ is bounded and $\lim_{t \rightarrow \infty} \mathbf{e} = 0$

Proof. The error dynamic of the above ESO can be determined as

$$\varepsilon \dot{\hat{\mathbf{e}}} = \mathbf{A}\hat{\mathbf{e}} + \varepsilon \mathbf{B}\dot{\mathbf{f}} = \begin{bmatrix} -\alpha_1 \mathbf{I}_5 & \mathbf{I}_5 & \mathbf{0} \\ -\alpha_2 \mathbf{I}_5 & \mathbf{0} & \mathbf{I}_5 \\ -\alpha_3 \mathbf{I}_5 & \mathbf{0} & \mathbf{0} \end{bmatrix} \hat{\mathbf{e}} + \varepsilon \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{I}_5 \end{bmatrix} \dot{\mathbf{f}} \quad (15)$$

Then Lyapunov candidate function is selected with positive define matrix \mathcal{P} as:

$$\mathcal{V} = \varepsilon \hat{\mathbf{e}}^\top \mathcal{P} \hat{\mathbf{e}} \quad (16)$$

Therefore, the derivative of the Lyapunov candidate can be calculated in the following

$$\begin{aligned} \frac{d}{dt} \mathcal{V} &= \varepsilon \dot{\hat{\mathbf{e}}}^\top \mathcal{P} \hat{\mathbf{e}} + \varepsilon \hat{\mathbf{e}}^\top \mathcal{P} \dot{\hat{\mathbf{e}}} \\ &= (\mathbf{A}\hat{\mathbf{e}} + \varepsilon \mathbf{B}\dot{\mathbf{f}})^\top \mathcal{P} \hat{\mathbf{e}} + \hat{\mathbf{e}}^\top \mathcal{P} (\mathbf{A}\hat{\mathbf{e}} + \varepsilon \mathbf{B}\dot{\mathbf{f}}) \\ &= \hat{\mathbf{e}}^\top \mathbf{A}^\top \mathcal{P} \hat{\mathbf{e}} + \varepsilon (\mathbf{B}\dot{\mathbf{f}})^\top \mathcal{P} \hat{\mathbf{e}} + \hat{\mathbf{e}}^\top \mathcal{P} \mathbf{A} \hat{\mathbf{e}} + \varepsilon \hat{\mathbf{e}}^\top \mathcal{P} \mathbf{B} \dot{\mathbf{f}} \end{aligned} \quad (17)$$

By choosing $\alpha_1, \alpha_2, \alpha_3, \varepsilon$ as positive constant, positive define matrix \mathcal{Q} and \mathbf{H} is Hurwitz, the Lyapunov equation $\mathbf{A}^\top \mathcal{P} + \mathcal{P} \mathbf{A} + \mathcal{Q} = 0$ is satisfied. The equation (17) becomes:

$$\begin{aligned} \frac{d}{dt} \mathcal{V} &= -\hat{\mathbf{e}}^\top \mathcal{Q} \hat{\mathbf{e}} + 2\varepsilon \hat{\mathbf{e}}^\top \mathcal{P} \mathbf{B} \dot{\mathbf{f}} \\ &\leq -\|\hat{\mathbf{e}}\|^2 \lambda_{\min}(\mathcal{Q}) + 2\varepsilon \|\hat{\mathbf{e}}\| \cdot \|\mathcal{P} \mathbf{B}\| f_{d\max} \\ &\leq -\mathcal{V} \frac{\lambda_{\min}(\mathcal{Q})}{\varepsilon \lambda_{\min}(\mathcal{P})} + 2\varepsilon \sqrt{\frac{\mathcal{V}}{\varepsilon \lambda_{\min}(\mathcal{Q})}} \|\mathcal{P} \mathbf{B}\| f_{d\max} \end{aligned} \quad (18)$$

where the function $\lambda_{\min}(\bullet)$ and $\lambda_{\max}(\bullet)$ are the maximum eigenvalue of matrix (\bullet) . We define $c_1 = \varepsilon \lambda_{\min}(\mathcal{P})$ and $c_2 = \lambda_{\min}(\mathcal{Q})$ to obtain:

$$\frac{d}{dt} \sqrt{\mathcal{V}} \leq \frac{-c_2}{2c_1} \sqrt{\mathcal{V}} + \frac{\sqrt{c_1}}{c_1} \varepsilon \|\mathcal{P} \mathbf{B}\| f_{d\max} \quad (19)$$

then,

$$\begin{aligned} \|\hat{\mathbf{e}}\| &\leq \sqrt{\frac{\mathcal{V}}{c_1}} \\ &\leq \frac{\sqrt{c_1} \mathcal{V}(\hat{\mathbf{e}}(0)) e^{-\frac{c_2}{2c_1} t}}{c_1} + \frac{\varepsilon \|\mathcal{P} \mathbf{B}\| f_{d\max}}{c_1} \int_0^t e^{\frac{-c_2}{2c_1}(t-\tau)} d\tau \\ &\leq \frac{\sqrt{c_1} \mathcal{V}(\hat{\mathbf{e}}(0)) e^{-\frac{c_2}{2c_1} t}}{c_1} + \frac{2\varepsilon \|\mathcal{P} \mathbf{B}\| f_{d\max}}{c_2} \left(e^{\frac{-c_2}{2c_1} t} + 1 \right) \end{aligned} \quad (20)$$

By selecting positive number $\varepsilon \ll 1$, the error between estimated states and output states is bounded and convergence to zero, i.e. $\lim_{\varepsilon \rightarrow 0} (\lim_{t \rightarrow \infty} \hat{\mathbf{e}}) = 0$. The theorem is proven.

Following the substitution of the system variables in (11) with the output from ESO, a generalized ESO-NMPC for the ballbot is outlined as follows:

$$\begin{aligned} \mathbf{u} &= \arg \min_{\mathbf{u}} \int_{t_0}^{t_0+T} (\hat{\mathbf{e}}(\tau)^\top \mathcal{R} \hat{\mathbf{e}}(\tau) + \mathbf{u}(\tau)^\top \mathcal{T} \mathbf{u}(\tau)) d\tau \\ \text{s.t. } \ddot{\mathbf{q}} &= \mathbf{M}(\hat{\mathbf{q}}_1)^{-1} (\mathbf{Q}(\hat{\mathbf{q}}_1) \mathbf{u} - \mathbf{C}(\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2) \hat{\mathbf{q}}_2 - \mathbf{G}(\hat{\mathbf{q}}_1)) \end{aligned} \quad (21)$$

where $\mathbf{x} = [\mathbf{q}_1 \quad \mathbf{q}_2]^\top$ is observation state vector and $\hat{\mathbf{e}}(t) = \hat{\mathbf{q}}_1(t) - \mathbf{q}_r(t)$ is the error of observation state and reference.

3.3. Obstacles Avoidance Using ESO-NMPC and Control Barrier Function (CBF)

We define the position of i -th obstacle in ground as $\mathbf{O}_i(t) = [x_i(t) \quad y_i(t)]^\top$. Motivated by CBF [13], barrier function for i -th obstacle can be chosen:

$$D_i(\mathbf{q}_1(t), \mathbf{O}_i(t)) = d_i(\mathbf{q}_1(t), \mathbf{O}_i(t))^2 - d_{\max}^2 \quad (22)$$

From this CBF certificate, the super level safe set of the distance can be formulated as follow

$$\mathcal{D} = \{\mathbf{q}_1 \in \mathcal{Q} \subset \mathbb{R}^5 : D \geq 0\} \quad (23)$$

Definition 1. Consider the dynamic system (10) with the control signal \mathbf{u} is locally Lipschitz. For any initial condition $\mathbf{q}_1 \in \mathcal{D} \subset \mathbb{R}^{10}$, there are existed a maximum interval $\mathcal{I}(\mathbf{q}_1(0)) = [0, t_{\max})$ such that $\mathbf{q}_1(t) \in \mathcal{I}(\mathbf{q}_1(0))$ is the unique solution of (10). The set \mathcal{D} defined in (23) is forward invariance if $\mathbf{q}_1(t) \in \mathcal{D} \quad \forall t \in \mathcal{I}(\mathbf{q}_1(t))$. The system (10) is safe under the constraint condition of set \mathcal{D} if \mathcal{D} is forward invariance [13].

Definition 2. Consider the set \mathcal{D} as the super level set of continuous differential function $D_i : \mathcal{X} \rightarrow \mathbb{R}$, the function D_i is a CBF if existing a class κ_∞ function $\gamma_i(\bullet)$ such as [11]

$$\begin{aligned} \frac{\partial D_i}{\partial \mathbf{q}_1} &\neq \mathbf{0} \quad \forall \mathbf{q}_1 \in \partial \mathcal{D} \\ \dot{D}_i(\mathbf{q}_1, \mathbf{O}_i) &\geq -\gamma_i(D_i(\mathbf{q}_1, \mathbf{O}_i)) \end{aligned} \quad (24)$$

By choosing a positive barrier function D if the ballbot is not in contact with the obstacle, then when the ballbot approaches the obstacle, the derivative of the barrier function \dot{D} , will tend to push the ballbot far away. That is if the selected control signal satisfies that $\dot{D}_i(\mathbf{q}, \mathbf{O}_i, \mathbf{u}) \geq -\gamma_i D_i(\mathbf{q}, \mathbf{O}_i)$ and $\gamma_i \in \kappa_\infty$, ballbot will not collide with obstacles. Therefore, we achieved

final obstacle avoidance based on the NMPC-CBF by the optimization problem as follows:

$$\begin{aligned} \mathbf{u} &= \arg \min_{\mathbf{u}} \int_{t_0}^{t_0+T} (\hat{\mathbf{e}}(\tau)^\top \mathcal{R} \hat{\mathbf{e}}(\tau) + \mathbf{u}(\tau)^\top \mathcal{T} \mathbf{u}(\tau)) d\tau \\ \text{s.t. } \ddot{\mathbf{q}} &= \mathbf{M}(\hat{\mathbf{q}}_1)^{-1} (\mathbf{Q}(\hat{\mathbf{q}}_1) \mathbf{u} - \mathbf{C}(\hat{\mathbf{q}}_1, \hat{\mathbf{q}}_2) \hat{\mathbf{q}}_2 - \mathbf{G}(\hat{\mathbf{q}}_1)) \\ \mathbf{O}_i^{(k)}(t) &= f(\mathbf{O}_i^{(k-1)}, \mathbf{O}_i^{(k-2)}, \dots, \mathbf{O}_i) \\ \dot{D}_i(\mathbf{q}_1, \mathbf{O}_i, \mathbf{u}) &\geq -\gamma_i(D_i(\mathbf{q}_1, \mathbf{O}_i)) \end{aligned} \quad (25)$$

Finally, the control structure for driving the ballbot balancing, tracking, and avoiding a moving or static obstacle is achieved in Fig. 3. At the time t_0 the sensor measures the state \mathbf{O}_i of n obstacles and output of the ballbot system. The observer estimates the output $\hat{\mathbf{q}} = \hat{\mathbf{q}}_1$ and the first derivative of output $\dot{\hat{\mathbf{q}}} = \hat{\mathbf{q}}_2$. Then the Predictive Model tries to predict the state of ballbot and the obstacles in the future of T_s . This information is used to formulate the barrier function (22) and the CBF constraint $\dot{D}_i \geq -\gamma_i(D_i)$ in the MPC optimization problem. The final control signal \mathbf{u} ensures that the ballbot tracks reference trajectory and avoids obstacles.

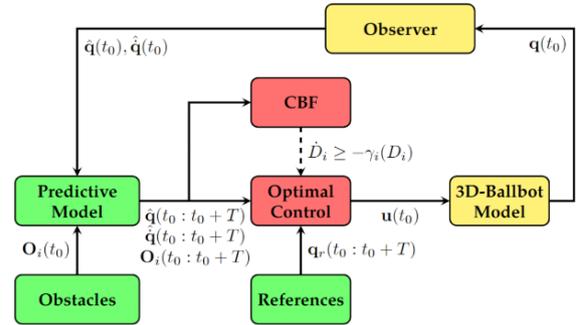


Fig. 3. NMPC control structure

4. Simulation

Ballbot is operated in two scenarios: static obstacles avoidance and moving obstacles avoidance. All the obstacles are considered in green and blue circular objects, the red circle represents for ballbot. The disturbance applied in system is defined as:

$$\mathbf{D}(t, \mathbf{q}) = \begin{bmatrix} 5 \sin(20\pi t) \\ 5 \sin(20\pi t) \\ 0.45 \sin(90\pi t) \\ 0.45 \sin(90\pi t) \\ 0.55 \sin(90\pi t) \end{bmatrix} \quad (26)$$

All parameters for system model and controller are shown in Table 1:

Table 1. Parameters for model and control structure

System parameters		Control parameters	
m_a	4.65 kg	α_1	4.1
m_b	0.6 kg	α_2	8.1
r_b	0.11 m	α_3	1.5
r_c	0.05 m	ϵ	0.01
J_a	$diag(0.2,0.2,0.05)$	prediction horizon T	0.5 s
J_b	0.006 kgm ²	control horizon T_c	1 s
J_c	0.0002 kgm ²	Q	$\mathbf{I}_{5 \times 5}$
l	0.5 m	λ	10

4.1. Static Obstacles Avoidance

In the first scenario, ballbot must track the desired line trajectory from (0, 0) to point (3,3). There are two obstacles blocking this path with the position $\mathbf{O}_1(t) = [1 \ 1.2]^\top$ and $\mathbf{O}_2(t) = [2 \ 1.8]^\top$, and the same radius of 0.3 m. The CBF certificates in avoiding obstacles are selected for each obstacle as follows:

$$D_i(\mathbf{q}(t), \mathbf{O}_i(t)) = \|\mathbf{p}_b(t) - \mathbf{O}_i(t)\|^2 - 0.3^2 \quad (27)$$

It can be observed that if the ballbot continues to follow the required trajectory without CBF, it will collide with two static obstacles at times 4.5s and 7.5s. However, two CBF constraints have changed the trajectory at these times to keep safe distances from obstacles. After leaving the dangerous area, the ballbot continues following the original trajectory (as shown in Fig. 4, Fig. 5 and Fig. 6).

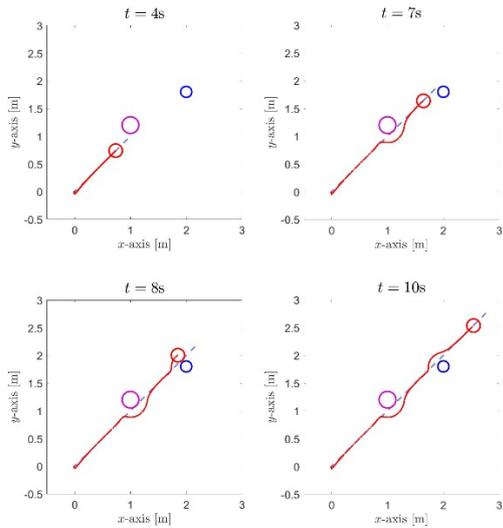


Fig. 4. Trajectory of ballbot in avoiding double static obstacles

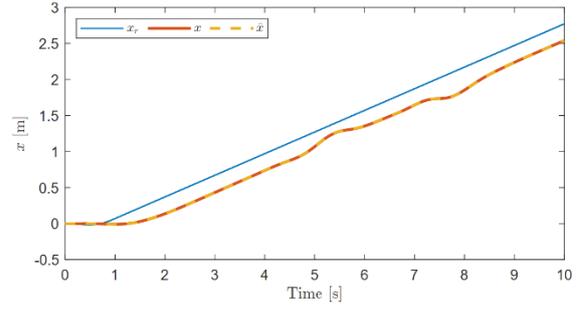


Fig. 5. Position response in Ox-axis

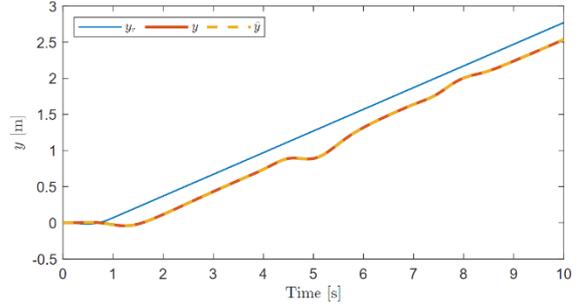


Fig. 6. Position response in Oy-axis

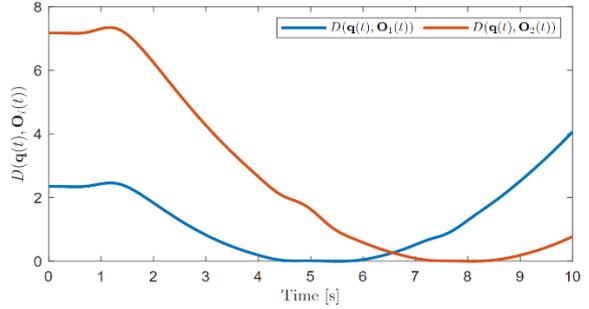


Fig. 7. CBF certificates for each static obstacle

The CBF certificates in Fig. 7 illustrates that the distance between ballbot and each obstacle are never under point of 0. That means the collisions don't occur in the entire operating process.

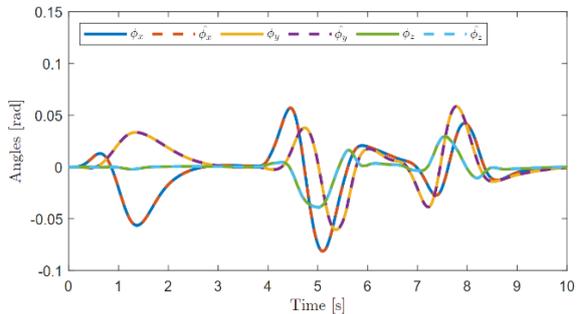


Fig. 8. Euler angle responses of ballbot

Simultaneously, there is an increase in deflection angles for guiding the robot's trajectory as shown in Fig. 8, but the system still guarantees not-too-large deflection angles and returns to the balanced point after completing the collision avoidance task. The predictive capability of NMPC in anticipating

collisions ensures smooth movement, preventing abrupt changes in tilt angles and control signals.

4.2. Moving Obstacles Avoidance

In the second scenario, beside the tracking task, ballbot need to predict collisions and adjust this trajectory to avoid the moving obstacles which tend to come across the desired path. The first position, velocity and radius of each obstacle are defined as:

$$\begin{aligned} \mathbf{O}_1(0) &= \begin{bmatrix} 0 \\ 1 \end{bmatrix}; \dot{\mathbf{O}}_1(t) = \begin{bmatrix} 0.23 \\ -0.09 \end{bmatrix}; r_{o1} = 0.3 \\ \mathbf{O}_2(0) &= \begin{bmatrix} 2 \\ 0 \end{bmatrix}; \dot{\mathbf{O}}_2(t) = \begin{bmatrix} -0.04 \\ 0.25 \end{bmatrix}; r_{o2} = 0.25 \end{aligned} \quad (28)$$

The CBFs for each obstacle can be formulated as:

$$\begin{aligned} D_1(\mathbf{q}(t), \mathbf{O}_1(t)) &= \|\mathbf{p}_b(t) - \mathbf{O}_1(t)\|^2 - r_{o1}^2 \\ D_2(\mathbf{q}(t), \mathbf{O}_2(t)) &= \|\mathbf{p}_b(t) - \mathbf{O}_2(t)\|^2 - r_{o2}^2 \end{aligned} \quad (29)$$

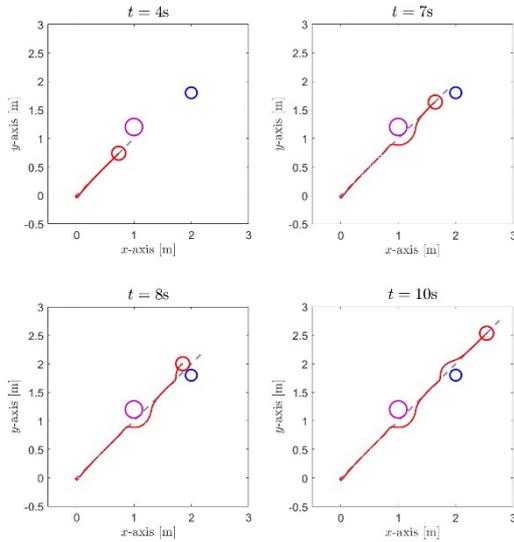


Fig. 9. Trajectory of ballbot in avoiding double moving obstacles

The proposed controller also shows excellent effectiveness in the case of moving obstacles. The future state of ballbot and obstacles are predicted and the result trajectory has some small change by circumventing the rear of the object for the minimum error with the desired path as shown in Fig. 9. The time-varying positions in Ox -axis and Oy -axis are presented in Fig. 10 and Fig. 11, respectively.

Similar to the scenario 1, CBF certificates in Fig. 12 decrease to nearly 0 when the ballbot is in the process of avoiding collisions while the Euler angle responses still ensure the balancing task of ballbot. Even when subjected to disturbance, the approximated values of deflection angle are very close to the respective model output response (Fig. 13). That shows that ESO has done a good job of estimating noise and state variables.

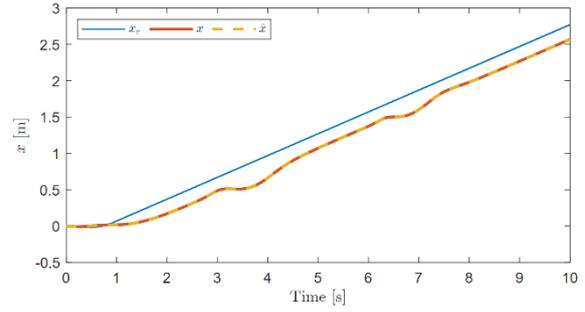


Fig. 10. Position response in Ox -axis

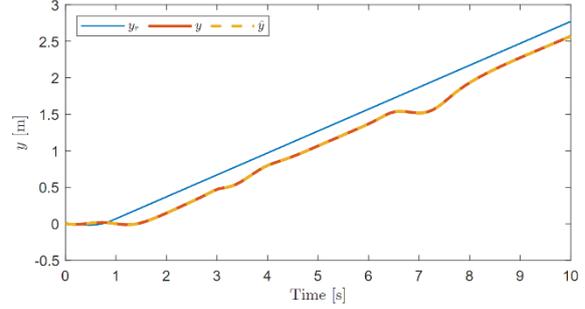


Fig. 11. Position response in Oy -axis

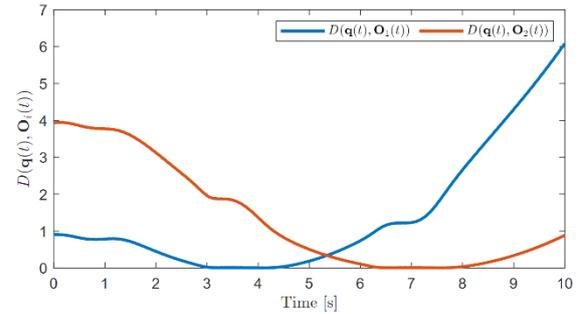


Fig. 12. CBF certificates for each static obstacle

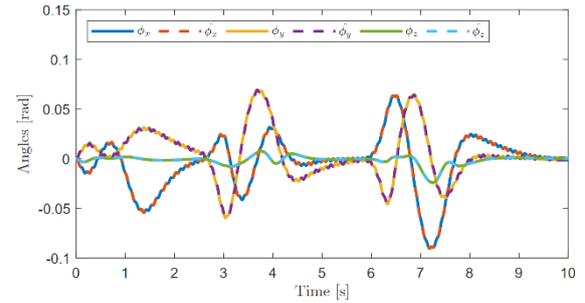


Fig. 13. Euler angle responses of ballbot

5. Conclusion

In this research, NMPC is implemented to maintain balance and navigate ballbot in operating environments with obstacles. ESO is applied as a disturbance rejection observer and overcoming the immeasurable variables. Leveraging NMPC predictability of ballbot's output response and behavior of dynamic obstacles, we applies CBFs to ensure collision avoidance between the ballbot and each obstacle in the prediction horizon. The numerical simulations are given to demonstrate the effectiveness

of the proposed method with both static and moving obstacles in disturbed environments.

In future work, we have a plan to improve our proposed controller by ensuring the safety guarantee with the tilt angle constraints as well as handling the complex-shaped obstacles. On the other hand, we will also try to handle with unknown motion equation of moving obstacles by using curve-fitting algorithm or machine learning.

Acknowledgments

This research is funded by the Hanoi University of Science and Technology (HUST) under project number T2023-TĐ-002.

References

- [1] T. B. Lauwers, G. A. Kantor, and R. L. Hollis, A dynamically stable single-wheeled mobile robot with inverse mouse-ball drive, in Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006, Orlando, FL, USA, May. 15-17, 2006, pp. 2884–2889. <https://doi.org/10.1109/ROBOT.2006.1642139>
- [2] T. Lauwers, G. Kantor, and R. Hollis, One is enough!, in Robotics Research: Results of the 12th International Symposium ISRR, 2007, pp. 327–336. https://doi.org/10.1007/978-3-540-48113-3_30
- [3] M. Kumagai and T. Ochiai, Development of a robot balanced on a ball - Application of passive motion to transport, in 2009 IEEE International Conference on Robotics and Automation, 2009, Kobe, Japan, May. 12-17, 2009, pp. 4106–4111. <https://doi.org/10.1109/ROBOT.2009.5152324>
- [4] P. Fankhauser and C. Gwerder, Modeling and Control of a Ballbot, Bachelor Thesis, Swiss Federal Institute of Technology, Zürich, Switzerland, 2010.
- [5] D. C. Vu, T. H. Nguyen Thi, D. D. Vu, T. L. Nguyen, and D. H. Nguyen, H-infinity full-state feedback control for a ball-balancing robot, in Hanoi Industrial University Journal of Science and Technology, vol. 59, no. 2A, pp. 96–100, Mar. 2023. <https://doi.org/10.57001/huih5804.2023.047>
- [6] D. C. Vu, T. H. N. Thi, D. D. Vu, V. P. Pham, D. H. Nguyen, and T. L. Nguyen, H-infinity Optimal full-state feedback control for a ball-balancing robot, in The International Conference on Intelligent Systems & Networks, Agu. 2023, pp. 326–334. https://doi.org/10.1007/978-981-99-4725-6_41
- [7] D. B. Pham, H. Kim, J. Kim, and S.-G. Lee, Balancing and transferring control of a ball segway using a double-loop approach [applications of control], IEEE Control Systems Magazine, vol. 38, no. 2, pp. 15–37, Apr. 2018. <https://doi.org/10.1109/MCS.2017.2786444>
- [8] M. D. Pham *et al.*, Auto-balancing ballbot systems: A fractional-order sliding mode based radial-basis neural network approach, in Proceedings of the International Conference on Engineering Research and Applications, ICERA 2022, 2022, pp. 270–280. https://doi.org/10.1007/978-3-031-22200-9_29
- [9] C.-W. Liao, C.-C. Tsai, Y. Y. Li, and C.-K. Chan, Dynamic modeling and sliding-mode control of a ball robot with inverse mouse-ball drive, in 2008 SICE Annual Conference, Chofu, Japan, Oct. 2008, pp. 2951–2955. <https://doi.org/10.1109/SICE.2008.4655168>
- [10] M. D. Pham, D. C. Vu, T. T. H. Nguyen, T. V. A. Nguyen, D. D. Vu, and T. L. Nguyen, Nonlinear model predictive control for ballbot systems: A benchmark with hierarchical sliding mode and linear quadratic controls, in 2023 12th International Conference on Control, Automation and Information Sciences (ICCAIS). IEEE, Ha Noi, Vietnam, Nov. 2023, pp. 411–416. <https://doi.org/10.1109/ICCAIS59597.2023.10382349>
- [11] A. D. Ames, J. W. Grizzle, and P. Tabuada, Control barrier function based quadratic programs with application to adaptive cruise control, in 53rd IEEE Conference on Decision and Control, Los Angeles, CA, USA, Dec. 2014, pp. 6271–6278. <https://doi.org/10.1109/CDC.2014.7040372>
- [12] A. D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle, Rapidly exponentially stabilizing control Lyapunov functions and hybrid zero dynamics, IEEE Transactions on Automatic Control, vol. 59, iss. 4, Apr. 2014, pp. 876–891. <https://doi.org/10.1109/TAC.2014.2299335>
- [13] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, Control barrier functions: Theory and applications, in 2019 18th European Control Conference (ECC), Naples, Italy, Jun. 2019, pp. 3420–3431. <https://doi.org/10.23919/ECC.2019.8796030>
- [14] J. Han, A class of extended state observers for uncertain systems, Control and Decision, vol. 10, no. 1, pp. 85–88, 1995.
- [15] M. Ran, J. Li, and L. Xie, A new extended state observer for uncertain nonlinear systems, Automatica, vol. 131, pp. 109772, 2021. <https://doi.org/10.1016/j.automata.2021.109772>