Hindawi
Mathematical Problems in Engineering
Volume 2024, Article ID 5562771, 14 pages
https://doi.org/10.1155/2024/5562771

Research Article

Improved Model Predictive Speed Control of a PMSM via Laguerre Functions

Chala Merga Abdissa

School of Electrical and Computer Engineering, Addis Ababa Institute of Technology, Addis Ababa University, P. O. Box 385, Addis Ababa 1000, Ethiopia

Correspondence should be addressed to Chala Merga Abdissa; chalmer.abdissa@gmail.com

Received 16 October 2023; Revised 17 January 2024; Accepted 6 March 2024; Published 21 March 2024

Academic Editor: Amitava Mukherjee

This paper proposes a model predictive speed control strategy for a surface-mounted permanent magnet synchronous motor by applying Laguerre functions. The model predictive controller (MPC) incorporates an integrator. A quadratic programming procedure is applied to solve the constrained optimization problem online. The paper also provides a solution for stability. The performance efficiency of the proposed scheme is validated by comparing the results with the performance of an optimal linear quadratic regulator, conventional state-space model predictive control, and a simple MPC algorithm with integral action. Extensive simulation results confirm the efficacy of the proposed scheme, showing that it achieves good steady-state performance while maintaining a fast dynamic response.

1. Introduction

Permanent magnet synchronous motors (PMSMs) have been intensively used in applications, such as industrial robots and electrical machines. Permanent magnet motors are prevalent in everyday machining tools due to their good dynamic properties and compact structure [1–4]. In addition to high efficiency, the advantages of permanent magnet motors include their lightweight and small size, which in turn facilitate installation and maintenance. Among the many techniques used in electric drive control, the proportional integrator (PI), linear quadratic regulator (LQR), and recently developed model predictive controller (MPC) have been employed. A combination of cascaded linear controller structures and PI controllers has been used for the speed regulation of PMSMs due to their simplicity [5]. To overcome extensive overshoots and ringing, cascaded linear controllers use a limited bandwidth. Hence, a reasonably good dynamic response can be achieved [6, 7]. However, during the transient time and in the presence of a load disturbance, the dynamic performance of the PI controller is reduced. Extensive research has focused on speed controller design for adjustable-speed PMSM systems to enhance the transient response, recovery time from a load disturbance, tracking ability, and robustness [8–14]. The LQR method is a contemporary control technique that is efficient but limited to linear systems. To evaluate the performance of the LQR controller, this paper compares it with results obtained from other control techniques. This means designing the same system using another control technique which could be cumbersome. On the other hand, an MPC can be applied to control both linear and nonlinear systems [15, 16]. In an MPC, the nonlinear system and corresponding linear model can be easily compared with little modifications.

It is known that tuning an MPC is not as difficult as tuning a PI, even for multiple input multiple output (MIMO) systems. Nevertheless, to optimally regulate the response of a system to a reference while respecting the constraints, an MPC requires more computational effort than either a PI or an LQR. Since an optimal solution can be analytically accomplished, the feedback gain can be precalculated offline when the operational constraints are absent or not active. However, if the constraints are available, quadratic programing (QP) can be used to solve the optimization problem online. Hence, an MPC is used intensively in industrial control where fast sampling is not required. In contrast, an MPC has limited applications in the control of electric drives and power

converters due to the computational load and fast sampling requirements to solve the QP problem. An enhancement in processor performance has notably surged, and new faster algorithms have been developed, rendering the MPC implementation possible for a power converter and an electric drive [17–19]. The proposed scheme in this paper focuses mainly on improving the performance of PMSM systems by designing an MPC using orthonormal functions called Laguerre functions (LMPC).

The three notable contributions of the proposed scheme are illustrated below:

(1) A controller with the capability to reject disturbances from the load torque of a PMSM system is developed. To ensure zero steady-state error in the presence of a load torque disturbance, an integrator is incorporated into the design. This modification improves the steady-state performance of the PMSM system.

(2) An LMPC is employed as a control technique for a PMSM system. The LMPC method incorporates the advantages of standard MPC algorithms such as constraint handling and online optimization and produces an algorithm with a low online computational burden. Because the LMPC method uses a simpler design, the computational load is reduced. The performance improvements of a PMSM designed using the LMPC are compared with those designed using the algorithm presented in a different studies [20, 21].

(3) To achieve a lower computational load and minimize the numerical problem, this paper proposes an exponentially decreasing objective function for the LMPC, specifically for a large prediction horizon to control the speed of the PMSM. The proposed LMPC method enhances the controller performance in an organized manner.

The structure of the proposed paper is as follows. Section 2 illustrates the drive model and linearization of the model for MPC design. Section 3 describes the MPC design. Section 4 outlines the simulation results, and Section 5 presents the conclusion.

2. Drive Model

The permanent magnet synchronous machines considered in this paper are called surface-mounted PMSMs. Models for speed control can be derived from the stator's equation and torque's equilibrium equations [22]. The following equations determine the synchronous reference d–q frame [22–24]:

$$\begin{cases} \dot{i_d}(t) = \dfrac{1}{L_d}\left(v_d - Ri_d + \omega L_q i_q\right) \\ \dot{i_q}(t) = \dfrac{1}{L_d}\left(v_d - Ri_q - \omega L_d i_d - \omega\psi\right) \\ \dot{\omega}(t) = \dfrac{p}{J}\left(T - \dfrac{B_v}{p}\omega - T_L\right) \\ T = \dfrac{3}{2}p\left[\psi i_q + \left(L_d - L_q\right)i_d i_q\right] \end{cases}. \tag{1}$$

If the permanent magnets are mounted on the rotor surface, and there is no significant internal asymmetry in the iron parts of the rotor, the direct-axis and quadrature-axis inductances of the machine are approximately equal, $L_d = L_q$. Hence, the torque in Equation (1) can be simplified to $T = \frac{3}{2}p\psi i_q$ allowing the electromagnetic torque to be controlled through i_q directly and maintaining $i_d = 0$. Maintaining $i_d = 0$ helps to achieve maximum efficiency of the PMSM, i.e., maximum torque per ampere condition in the whole operation range. $\omega = p\omega_m$ is electromechanical speed, ψ is the permanent flux linkage, p is the number of pole pairs, and ω_m is the mechanical speed. In addition, T_L is the disturbance from the load torque, while J and B_v are the motor's moment of inertia and viscous coefficient. In Equation (1), the PMSM dynamics are expressed by a nonlinear set of equations because of motional coupling terms (ωi_q and ωi_d); hence, the equations will be linearized at operating points for use in the MPC design. Using a first-order Taylor series to approximate the nonlinear coupled terms as:

$$\begin{aligned} \omega(t)i_q(t) \approx \omega_0 i_{q0} + i_{q0}(\omega(t) - \omega_0) \\ + \omega_0(i_d(t) - i_{d0}) \end{aligned}. \tag{2}$$

Substituting Equations (2) and (3) into Equation (1) yields the following equation:

$$\begin{cases} \dot{x_p}(t) = A_p x_p(t) + B_p u(t) + v \\ y(t) = D_p x_p(t) \end{cases}, \tag{3}$$

where

$$A_p = \begin{bmatrix} -\dfrac{R}{L_d} & \dfrac{L_q}{L_d}\omega_0 & \dfrac{L_q}{L_d} \\ -\dfrac{L_d}{L_q}\omega_0 & -\dfrac{R}{L_q} & -\left(\dfrac{L_d}{L_q}i_{d0} + \dfrac{\psi}{L_q}\right) \\ 0 & \dfrac{3P^2\psi}{2J} & -\dfrac{B_v}{J} \end{bmatrix}, \tag{4}$$

$$B_p = \begin{bmatrix} \dfrac{1}{L_d} & 0 \\ 0 & \dfrac{1}{L_q} \\ 0 & 0 \end{bmatrix}, v = \begin{bmatrix} -\dfrac{L_q}{L_d}\omega_0 i_{q0} \\ \dfrac{L_d}{L_q}\omega_0 i_{q0} \\ -\dfrac{pT_L}{J} \end{bmatrix}, \tag{5}$$

$$C_p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{6}$$

$v \in \Re^3$ is the steady-state parameter and load torque, $x_p = \left[i_d(t) \quad i_q(t) \quad \omega(t)\right]^T \in \Re^3$ is the state, $y = \left[i_d(t)\ \omega(t)\right]^T \in \Re^2$ is the output, and $u = \left[v_d\ v_q\right]^T \in \Re^2$ is the control input.

The MPC requires a discrete-time model, and Equation (3) is thus discretized with a sampling period using

a zero-order hold, resulting in the discrete-time model as follows:

$$\begin{cases} x_p(k+1) = A_d x_p(k) + B_d u(k) + \mathrm{d}_v \\ y(k) = C_d x_p(k) \end{cases}, \tag{7}$$

where $A_d = e^{A_p T_s}, B_d = \int_0^{T_s} e^{A_p \tau} B_p d\tau, C_d = C_p$, and d_v has constant entries. For speed regulation of a PMSM, $n_u = 2$, $n_y = 2$, and $n_x = 3$ will be used, where n_u, n_y, and n_x correspond to input, output, and state, respectively.

Let $\Delta x_p(k) = x_p(k) - x_p(k-1)$ and $\Delta u(k) = u(k) - u(k-1)$ denote increment on state and input variables, respectively, determined from the corresponding variables in Equation (7). The state dynamics in the incremental model are as follows:

$$\Delta x_p(k+1) = A_d \Delta x_p(k) + B_d \Delta u(k). \tag{8}$$

In a similar manner, the output incremental dynamics are given as follows:

$$y(k+1) = y(k) + C_d A_d \Delta x_p(k) + C_d B_d \Delta u(k). \tag{9}$$

By choosing a new state, $x(k) = \left[\Delta x_p(k)^T \; y(k)^T\right]^T$ the augmented state-space model is obtained by combining Equation (8) with Equation (9):

$$\begin{cases} x(k+1) = Ax(k) + B\Delta u(k) \\ y(k) = Cx(k) \end{cases}, \tag{10}$$

where $A = \begin{bmatrix} A_d & 0_1 \\ D_d A_d & I \end{bmatrix}, B = \begin{bmatrix} B_d \\ D_d B_d \end{bmatrix}, C = [0_2 \quad I] \cdot 0_1$ and 0_2 denote the zero matrices and I is identity matrices of appropriate dimensions.

Using the augmented model in Equation (10) has two advantages. First, the augmented model removes the d_v of Equation (7) including any uncertain parameters. Second, it eliminates the unknown torque.

3. Model Predictive Control Design

To achieve good controller performance while operational constraints are present, use an MPC scheme. The performance of a PMSM speed controller employing Laguerre functions, i.e., an LMPC, is compared with an optimal discrete LQR (DLQR) in this and the following section. In later sections, the performance of an LMPC is compared with the MPC-IA and with a more conventional MPC based on a state-space design (SS-MPC).

3.1. State-Space MPC. An SS-MPC is formulated using the state-space approach. An SS-MPC with a convex quadratic performance index is a QP problem, which is appealing because QP must be solved online [25]. The state variable $x(k)$ is estimated through an observer to obtain an optimal solution in the presence of operational constraints [26]. Thus, apply an observer of the form:

$$\hat{x}(k+1) = Ax(k) + B\Delta u(k) + K_{\mathrm{ob}}(y(k) - C\hat{x}(k)), \tag{11}$$

where K_{ob} is the observer gain matrix, it is apparent that the observer gain K_{ob} can be used to manipulate the convergence rate of the error. K_{ob} is used to place the closed-loop eigenvalues of the error system matrix $A - K_{\mathrm{ob}}C$ at a desired location of the complex plane. The estimate of the output yields the following equation:

$$\hat{y}(k) = C\hat{x}(k). \tag{12}$$

Model in Equation (10) is used to calculate the future state variables through $\Delta u(k), \cdots, \Delta u(k+1), \cdots, \Delta u(k+N_c-1)$, leading to following equation:

$$\begin{cases} \hat{x}(k+1|k) = A\hat{x}(k) + B\Delta u(k) \\ \hat{x}(k+2|k) = A^2\hat{x}(k) + AB\Delta u(k) + \\ \qquad B\Delta u(k+1) \\ \qquad \vdots \\ \hat{x}(k+N_p|k) = A^{N_p}\hat{x}(k) + A^{N_p-1}B\Delta u(k) + \\ \cdots + A^{N_p-N_c}B\Delta u(k+N_c-1) \end{cases}, \tag{13}$$

where N_p is the prediction horizon and N_c (control horizon) is a parameter that determines the number of future control inputs to be included in optimization. This, therefore, assumes $N_c \leqslant N_p$. In its compact form, the output prediction for the next N_p instants is as follows:

$$\hat{Y} = E\hat{x}(k) + \Theta\Delta U, \tag{14}$$

where

$$\begin{aligned} \hat{Y} &= \left[\hat{y}(k+1)^T \hat{y}(k+2)^T \cdots \hat{y}(k+N_p)^T\right]^T \\ \Delta U &= [\Delta u(k)^T \Delta u(k+1)^T \cdots \Delta u(k+N_c-1)^T]^T \\ E &= [(CA)^T (CA^2)^T \cdots (CA^{N_p})^T]^T, \end{aligned} \tag{15}$$

$$\Theta = \begin{bmatrix} \mathrm{CB} & 0 & \cdots & 0 \\ \mathrm{CAB} & \mathrm{CB} & \cdots & 0 \\ \mathrm{CA}^2 B & \mathrm{CAB} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{CA}^{N_p-1}B & \mathrm{CA}^{N_p-2}B & \cdots & \mathrm{CA}^{N_p-N_c}B \end{bmatrix}. \tag{16}$$

Consider the cost function:

$$J_s = \left(R_s - \hat{Y}\right)^T Q_s \left(R_s - \hat{Y}\right) + \Delta U^T \overline{R} \Delta U, \tag{17}$$

where $R_s = \overline{R}_s r(k) = \overbrace{I \quad I \quad \cdots \quad I}^{n_y \times N_p} r(k)$ with reference $r(k)$ of constant entries, $Q_s \geqslant 0(n_y \times N_p) \times (n_y \times N_p)$ and $\overline{R} > 0$ is a $(n_u \times N_c) \times (n_u \times N_c)$. Upon minimizing the cost function in Equation (17), obtain the optimal control vector $\Delta U =$

$(\Theta^T Q_s \Theta + \overline{R})^{-1} \Theta^T Q_s (\overline{R}_s - E\widehat{x}(k))$. The receding horizon principle to obtain the control law:

$$\begin{aligned} \Delta u(k) &= [I \quad 0 \quad \cdots \quad 0]\Delta U \\ &= k_y r(k) - k_{\mathrm{mpc}}\widehat{x}(k), \end{aligned} \tag{18}$$

where k_y is the matrix correspond to change in set-point and k_{mpc} is the state feedback gain matrix. Operational constraints in control algorithms come from physical systems. In PMSM, the input voltages and are limited by the DC bus voltage. Further, space vector pulse width modulator modulates the maximum voltage to $V_{\mathrm{dc}}/\sqrt{3}$. The cost function in Equation (17) is minimized with respect to ΔU:

$$\begin{aligned} &D_1(u^{\min} - u(k-1)) \le D_2 \Delta U \\ &\le D_1(u^{\max} - u(k-1)) \\ &D_1 \Delta u^{\min} \le \Delta U \le D_1 \Delta u^{\max} \end{aligned}, \tag{19}$$

where

$$D_1 = \begin{bmatrix} I \\ I \\ \vdots \\ I \end{bmatrix}, D_2 = \begin{bmatrix} I & 0 & \cdots & 0 \\ I & I & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I & I & \cdots & I \end{bmatrix}. \tag{20}$$

The constraints in compact form is given as follows:

$$M\Delta U \le \gamma. \tag{21}$$

And the optimization problem with constraints is solved by QP [25].

3.2. Laguerre-Based Model Predictive Control. The general procedure for MPC designed using Laguerre function was explained in a previous study [27]. The author addressed this design method for single input single output. This paper presents an LMPC design method for MIMO systems specifically for the use to control the speed of PMSM systems.

Design framework: Let $\Delta u(k) = [\Delta u_1(k) \quad \Delta u_2(k) \quad \cdots \quad \Delta u_r(k)]^T$ and the partitioned input matrix be $B[B_1 \quad B_2 \quad \cdots \quad B_r]$, where r is the number of inputs and B_i is the ith column of the matrix. Use orthonormal basis Laguerre functions to model the control trajectory. The Laguerre function in the z-transform is as follows:

$$\Gamma_N^i = \frac{\sqrt{1-a_i^2}}{1-a_i z^{-1}} \left[\frac{z-a_i}{1-a_i z^{-1}}\right]^{N-1}, \tag{22}$$

where $0 \leqslant a_i \leqslant 1$ is the pole of the Laguerre network.

The control trajectory can be described using the Laguerre functions:

$$\Delta u_i(k) \approx \sum_{j=1}^{N} d_j^i(k) l_j^i(k) \ , \tag{23}$$

where $l_j^i(m)$ is the inverse z-transform of jth term in the discrete Laguerre network, i.e., Γ_j^i and the coefficients d_j^i are unknowns and must be obtained from systems data.

The tuning parameters a_i and N_i are related to N_c [26] by the following equation:

$$a_i \approx e^{-N_i/N_c}. \tag{24}$$

Now, rewriting Equation (23):

$$\Delta u_i(k) = L_i(k)^T \xi_i \ , \tag{25}$$

where ξ_i and $L_i(k)$ are $\xi_i = \left[d_1^i \quad d_2^i \quad \cdots \quad d_{N_i}^i\right]^T$ and $L_i(k) = \left[l_1^i(k) \quad l_2^i(k) \quad \cdots \quad l_{N_i}^i(k)\right]^T$.

The state prediction can be written as follows:

$$x(k_i + h|k_i) = A^h x(k_i) + \phi(h)^T \xi \ , \tag{26}$$

where the vector ξ and the matrix $\phi(h)^T$ are given by $\xi^T = \left[\xi_1^T \quad \xi_2^T \quad \cdots \quad \xi_h^T\right]$ and $\phi(h)^T = \sum_{j=0}^{h-1} A^{h-j-1}\left[B_1 L_1(j)^T \cdots B_h L_h(j)^T\right]$.

Similarly, the output is described as follows:

$$\begin{aligned} y(k_i + h|k_i) = \mathrm{C}A^h x(k_i) + \sum_{j=0}^{h-1} \mathrm{C}A^{h-j-1} \\ \times [B_1 L_1(j)^T B_2 L_2(j)^T \cdots B_h L_h(j)^T]\xi. \end{aligned} \tag{27}$$

3.2.1. Solution without Constraints. The objective is to determine the parameter vector that minimizes the disparity between the predicted output and the desired set point signal. To achieve this objective, a cost function designed for this specific purpose is formulated. This cost function serves as a guiding metric to measure how closely the predicted output aligns with the set point signal. The cost function for this purpose is as follows:

$$\begin{aligned} J_L = \sum_{h=1}^{N_p} x(k_i + h|k_i)^T Q_L x(k_i + h|k_i) \\ + \xi^T R_L \xi, \end{aligned} \tag{28}$$

where $Q_L \ge 0$ and $R_L \ge 0$. Performing the partial derivative on J_L, i.e., $\partial J_L / \partial \xi = 0$, the optimal Laguerre coefficients vector in absence of constraints is found as follows:

$$\xi = -\Pi^{-1} \Psi x(k_i), \tag{29}$$

With $\Pi = \sum_{h=1}^{N_p} \Theta(h) Q_L \Theta(h)^T + R_L$ and $\Psi = \Theta(h) Q_L A^h$.

After calculating the optimal Laguerre coefficients vector ξ, the receding horizon control (RHC) law is realized as follows:

$$\Delta u(k_i) = \begin{bmatrix} L_1(0)^T & o_2^T & \cdots & o_h^T \\ o_1^T & L_2(0)^T & \cdots & o_h^T \\ \vdots & \vdots & \ddots & \vdots \\ o_1^T & o_2^T & \cdots & L_h(0)^T \end{bmatrix} \xi \ , \quad (30)$$

where $o_k^T, k = 1, 2, \cdots, h$ is a row vector with an appropriate dimension.

In linear state feedback control form, the control variable $\Delta u(k_i)$ is written as follows:

$$\Delta u(k_i) = -K_{\text{lmpc}} x(k_i) \ , \quad (31)$$

where the gain is given as follows:

$$K_{\text{lmpc}} = \begin{bmatrix} L_1(0)^T & o_2^T & \cdots & o_h^T \\ o_1^T & L_2(0)^T & \cdots & o_h^T \\ \vdots & \vdots & \ddots & \vdots \\ o_1^T & o_2^T & \cdots & L_h(0)^T \end{bmatrix} \Pi^{-1} \Psi. \quad (32)$$

3.2.2. Solution to Constrained LMPC. The optimization procedure is to minimize the cost function J_L while ensuring Equations (33) and (34):

$$\Delta u^{\min} \leq \begin{bmatrix} L_1(0)^T & o_2^T & \cdots & o_h^T \\ o_1^T & L_2(0)^T & \cdots & o_h^T \\ \vdots & \vdots & \ddots & \vdots \\ o_1^T & o_2^T & \cdots & L_h(0)^T \end{bmatrix} \xi \ , \quad (33)$$
$$\leq \Delta u^{\max}$$

$$u^{\min} \leq \begin{bmatrix} \sum_{i=0}^{k-1} L_1(i)^T & o_2^T & \cdots & o_h^T \\ o_1^T & \sum_{i=0}^{k-1} L_2(i)^T & \cdots & o_h^T \\ \vdots & \vdots & \ddots & \vdots \\ o_1^T & o_2^T & \cdots & \sum_{i=0}^{k-1} L_h(i)^T \end{bmatrix}$$
$$\xi \leq u^{\max} - u(k_i - 1), \quad (34)$$

where $\Delta u^{\min}$ and $\Delta u^{\max}$ represent the lower and upper boundaries for the change in control input (Δu), while $u^{\min}$ and $u^{\max}$ define the lower and upper constraints for the control input itself (u). Here, $u(k_i - 1)$ denotes the control input at the previous time step. Additionally, o_m^T is a zero-row vector, and its dimension corresponds to that of $L_m(0)^T$.

Therefore, given these conditions and considerations, the constraints originally described in Equations (33) and (34) can now be redefined as follows:

$$M\xi \leq \gamma, \quad (35)$$

where M and γ compatible matrices in the QP problem [24]. This compatibility signifies that they are structurally suitable for the specific requirements and constraints of the problem, ensuring a harmonious integration within the framework of the QP problem.

3.2.3. Stability. In this context, this paper is broadening the scope of stability analysis for the MPC algorithm, specifically when constraints are in effect and Laguerre polynomials are employed. To recap, in RHC, the control trajectory's first term, denoted as $\Delta u(k)$ for time step k, is typically applied. However, this work considers the case where Laguerre polynomials are used, which extends our understanding of the MPC algorithm's stability, especially when constraints are activated. This allows us to explore the broader range of control actions over multiple time steps, $\Delta u(k+h), h = 0, 1, 2, \cdots, N_p$, rather than just the immediate $\Delta u(k)$ action.

This extension of stability analysis is crucial for a deeper understanding of how the MPC algorithm performs when constraints are in play, enabling more robust and effective control in various real-world applications.

$$J_L = \sum_{h=1}^{N_P} x(k+h|k)^T Q_L x(k+h|k) + \sum_{h=0}^{N_p} \Delta u(k+h)^T R_L \Delta u(k+h). \quad (36)$$

Subject to constraints in Equations (33) and (34).

Assumption 1 (Terminal State Constraints $x(k+N_p) = 0$). *The assumption made here is that as the prediction horizon becomes sufficiently long, the control trajectory $\Delta u(k+h) = L(h)^T \xi$ tends to converge to zero. This convergence behavior is observed when the terminal state constraints are met, specifically when $x(k+N_p) = 0$.*

Assumption 2 (Minimizable Cost). *It is also assumed that there exists a solution ξ when J_L is minimized, i.e., $J_{\min} = \partial J_L / \partial \xi = 0$.*

Theorem 1 (Stability). *Under Assumptions 1 and 2, the PMSM system described in Equation (10) exhibits asymptotic stability when operated in a closed-loop configuration. This is achieved through the implementation of a receding horizon controller denoted as $\Delta u(k)$, an objective function defined in Equation (34), and the constraints outlined in Equations (33) and (34).*

Proof. Let $V(x(j), k) = J_{\min}$ be a candidate Lyapunov function:

$$V(x(k), k) = \sum_{h=1}^{N_P} x_0(k+h|k)^T Q_L x_0(k+h|k), \quad (37)$$

where $x_0(k+h|k) = A^h x(k) + \sum_{i=0}^{h-1} A^{h-i-1} \text{BL}(i)^T \xi_0$ and ξ_0 is the solution. □

It can be clearly seen that $V(x(k),k)>0$ and $V(x(k),k) \to \infty$ as $x(k) \to \infty$.

$$V(x(k+1),k+1) = \sum_{h=1}^{N_p} x_1(k+h|k+1)^T Q_L x_1(k+h|k+1) + \sum_{h=0}^{N_p-1} \Delta u_1(k+h)^T R_L \Delta u_1(k+h), \tag{38}$$

where $x_1(k+h+1|k+1) = A^h x(k+1) + \sum_{i=0}^{h-1} A^{h-i-1} \mathrm{BL}(i)^T \xi_1$ and ξ_1 is the solution at $k+1$.

Because ξ_1 is the optimal solution, $V(x(k+1),k+1) \leq \overline{V}(x(k+1),k+1)$ is valid where $\overline{V}(x(k+1),k+1)$ is same as $V(x(k+1),k+1)$ with ξ_0 replacing $\xi_1 \cdot \xi_0$ is feasible solution in the vicinity of $x(k)$.

Let $\Delta V(x) = V(x(k+1),k+1) - V(x(k),k)$ is bounded by the following equation:

$$\Delta V(x) \leq \overline{V}(x(k+1),k+1) - V(x(k),k). \tag{39}$$

And $\overline{V}(x(k+1),k+1) - V(x(k),k) = x(k+N_p)^T Q_L x(k+N_p) - x(k+1)^T Q_L x(k+1) - \Delta u(k)^T R_L \Delta u(k)$.

$$\overline{V}(x(k+1),k+1) - V(x(k),k). \tag{40}$$

Hence, $\Delta V(x) = V(x(k+1),k+1) - V(x(k),k)$ is negative, i.e.:

$$-x(k+1)^T Q_L x(k+1) - \Delta u(k)^T R_L \Delta u(k) \leq 0. \tag{41}$$

Therefore, the PMSM controller is asymptotically stable.

Remark 1. In MPC algorithms that contain integrators, the prediction horizon also affects numeric condition [27]. Specifically, for a large control horizon, the MPC algorithm becomes ill-conditioned. A proven technique to solve the problem is to use exponential weighted cost function. In contrast to the more common exponentially increasing weight [28, 29], an exponentially decreasing weighting is suggested in this paper. The main purpose of focusing on exponentially decreasing weighting is to enhance the numerical stability of the class of MPC algorithms that contain integrators for multivariable systems.

The exponentially weighted cost function is as follows:

$$J = \sum_{h=1}^{N_p} \sigma^{-2h} x(k+h|k)^T Q_L x(k+h|k). \tag{42}$$

Subject to the constraints:

$$M\xi \leqslant \gamma. \tag{43}$$

With state equation:

$$x(k+h+1|k) = Ax(k+h|k) + B\Delta u(k+h). \tag{44}$$

Lemma 1 (Cost Function Equivalence). *The solution of the exponentially weighted objective function (Equation (42)) subject to the inequality constraints Equation (44) and state equation constraints Equation (43) can be found by minimizing the following equation:*

$$\widehat{J} = \sum_{h=1}^{N_p} \widehat{x}(k+h|k)^T Q_L \widehat{x}(k+h|k). \tag{45}$$

Subject to the following equation:

$$M_\sigma \Delta \widehat{U} \leq \gamma. \tag{46}$$

With state-equation $\widehat{x}(m+h+1) = \frac{A}{\sigma}\widehat{x}(m+h|m) + \frac{B}{\sigma}\Delta\widehat{u}(m+h)$ *and* $A_\sigma = A/\sigma, B_\sigma = B/\sigma$, *where* M_σ *is the matrix defined by the following equation:*

$$M_\sigma = M \begin{bmatrix} I & 0 & \cdots & 0 & 0 \\ 0 & \sigma^1 I & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \sigma^{N_p-1} I & 0 \\ 0 & 0 & \cdots & 0 & \sigma^{N_p} I \end{bmatrix}. \tag{47}$$

Proof. Let $\widehat{x}(k+h|k) = \sigma^{-h} x(k+h|k), \Delta\widehat{u}(k+h) = \sigma^{-h}\Delta u(k+h)$. This leads us to the equivalence of J (Equation (42)) and $\widehat{J}$ (Equation (45)). □

Similarly, for the state Equation (44):

$$\widehat{x}(k+h+1|k) = \sigma^{-(h+1)} x(k+h+1|k). \tag{48}$$

Here, $\sigma^{-h} x(k+h|k) = \widehat{x}(k+h|k); \sigma^{-h}\Delta u(k+h) = \Delta\widehat{u}(k+h)$. Additionally, modifying the constraint Equations (43) and (45) is done by substituting $\Delta u(k) = \Delta\widehat{u}(k), \Delta\widehat{u}(k+1) = \sigma\Delta\widehat{u}(k+1), \cdots, \Delta u(k+1) = \sigma^h \Delta\widehat{u}(k+h)$. Hence, it has been proved that objective function (42) and (45) with their respective constraints are identical.

Remark 2. It should be mentioned that the result Lemma 1 shows that an objective function with exponential data weight is similar to unweighted objective function with no weights if proper scaling factor σ is used to achieve a stable model.

3.3. Model Predictive Control with an Integral Action (MPC-IA). Standard MPC algorithms usually do not have an integrator. This is because algorithms employing integral actions may not result in optimal operation. Among the many

techniques to achieve optimality when integrator is in use, apply a simple MPC algorithm in increment form proposed in Di Ruscio's [21] study for our task of controlling speed of the PMSM system and compare with the proposed algorithm.

4. Simulation Parameters and Results

4.1. Parameters. For simulation purpose, the system parameter values given as follows have been used. Take $p = 2$, $J = 0.0235\ \mathrm{Kgm}^2$, $B_v = 1.1 \times 10^{-4}\ \mathrm{Nm} \cdot \mathrm{s}$, $L = 0.007\mathrm{H}$, $R = 2.98\ \Omega$, $V_{\mathrm{dc}} = 100\ \mathrm{V}$, $\psi = 0.125\ \mathrm{Wb}$, $\omega_*^e = 83.8\ \mathrm{rad/s}$, $N_c = 20$, $N_{p_0} = 25$, $N_{p_1} = 50$, $N_{p_2} = 500$, $T_s = 200 \times 10^{-6}\mathrm{s}$. The steady-state value for the q-axis current is $i_{q0} = 1$ A.

4.2. Constraints. In order to avoid over modulation, the maximum achievable voltage is limited by the following equation:

$$\sqrt{v_d^2 + v_q^2} \leq \frac{V_{\mathrm{dc}}}{\sqrt{3}}. \tag{49}$$

The constraint in Equation (49) is nonlinear. Linearizing using rectangular area approximation [30], yields the following equation:

$$|v_q| \leq \varepsilon \frac{V_{\mathrm{dc}}}{\sqrt{3}}, |v_d| \leq \sqrt{(1 - \varepsilon^2)} \frac{V_{\mathrm{dc}}}{\sqrt{3}}. \tag{50}$$

For this simulation, choose $\varepsilon = 0.9$. The constraints $-51.96 \leq v_q \leq 51.96$, $-25.17 \leq v_d \leq 25.17$, $-10 \leq \Delta v_q \leq 10$ and $-10 \leq \Delta v_q \leq 10$.

The performance efficiency of the LMPC control scheme is validated using simulations on a PMSM system. First, begin by examining the responses of LMPC controller analyzed for load disturbance. Next, the stability is checked with those obtained from the optimal discrete linear quadratic regulator (DLQR) for large prediction horizon. Finally, the PMSM system under three control algorithms, i.e., SS-MPC, LMPC, and MPC-IA control, are compared by simulation.

4.3. Tuning the LMPC. Tuning using parameters in the weight matrix.

The cost function weights affect the closed-loop performance of the LMPC. The larger the weights, the faster the response.

The weight matrices in the cost function (28) are as follows:

$$Q_L = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & q_1 & 0 \\ 0 & 0 & 0 & 0 & q_2 \end{bmatrix}, R_L = r \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{51}$$

where q_1 and q_2 will be used as parameters for tuning the controller. Use $r_w = 0.1$. Several simulations on PMSM system show that tuning is not advisable, as the responses are less sensitive to changes in the value of r_w.

Table 1: Constant, $N = 5$.

Control horizon	N	Required a
10	5	0.6065
15	5	0.7165
20	5	0.7788
25	5	0.8187
50	5	0.9048

Table 2: Constant control horizon, $N_c = 20$.

Control horizon	N	Required a	Time (ms)
20	2	0.9048	37.17
20	4	0.8187	38.54
20	6	0.7408	38.66
20	8	0.6703	39.71
20	10	0.6065	39.86

4.3.1. Tuning Using the Laguerre Parameters. Let us look at the effects of Laguerre parameters (a and N) on the control horizon N_c, tuning process, and the overall computational burden of the LMPC algorithm. Using Equation (24), i.e., $a \approx e^{-N/N_c}$ for relating the control horizon (N_c) the Laguerre parameters a and N, consider the two cases:

(1) Constant N: Choosing a constant parameter $N = 5$ obtain Table 1.

Thus, the control horizon can be increased by manipulating the parameter a without changing the network order N.

(2) Constant control horizon: Assume the target control horizon is 20. Choose an integrator specifying the Laguerre network order N. Thereafter, Equation (24) to calculate the relevant scaling factor a as shown in Table 2 has been used.

In order to achieve a lower overshoot, the weight coefficients are selected as $q_1 = 1$ and $q_2 = 0.01$. In three d-axis current: i_d, motor speed, d-axis voltage: v_d, q-axis voltage: v_q, d-axis incremental voltage: Δv_d and q-axis incremental voltage: Δv_q cases ($N = 4$, $N = 6$, and $N = 10$), the LMPC tracks the reference speed very well (Figure 1). To check the computing resources required, times to compute the control law while varying the network order have been recorded. Table 2 shows that as N increases the, the computation time also increases. The experiments were carried out on a PC with Windows 10, 64 bits, Intel Core i7-7820X CPU with 3.60 GHz, 16 GB of RAM. Thus, to ensure a lower computation burden a Laguerre network of lower order N is chosen yet achieving the same control horizon. The computational effort is lower when smaller number of parameter is used. For $N = 4$, only four Laguerre term is used to capture the control input while $N = 10$ requires 10 Laguerre terms to capture the same control input. The other important thing to remember

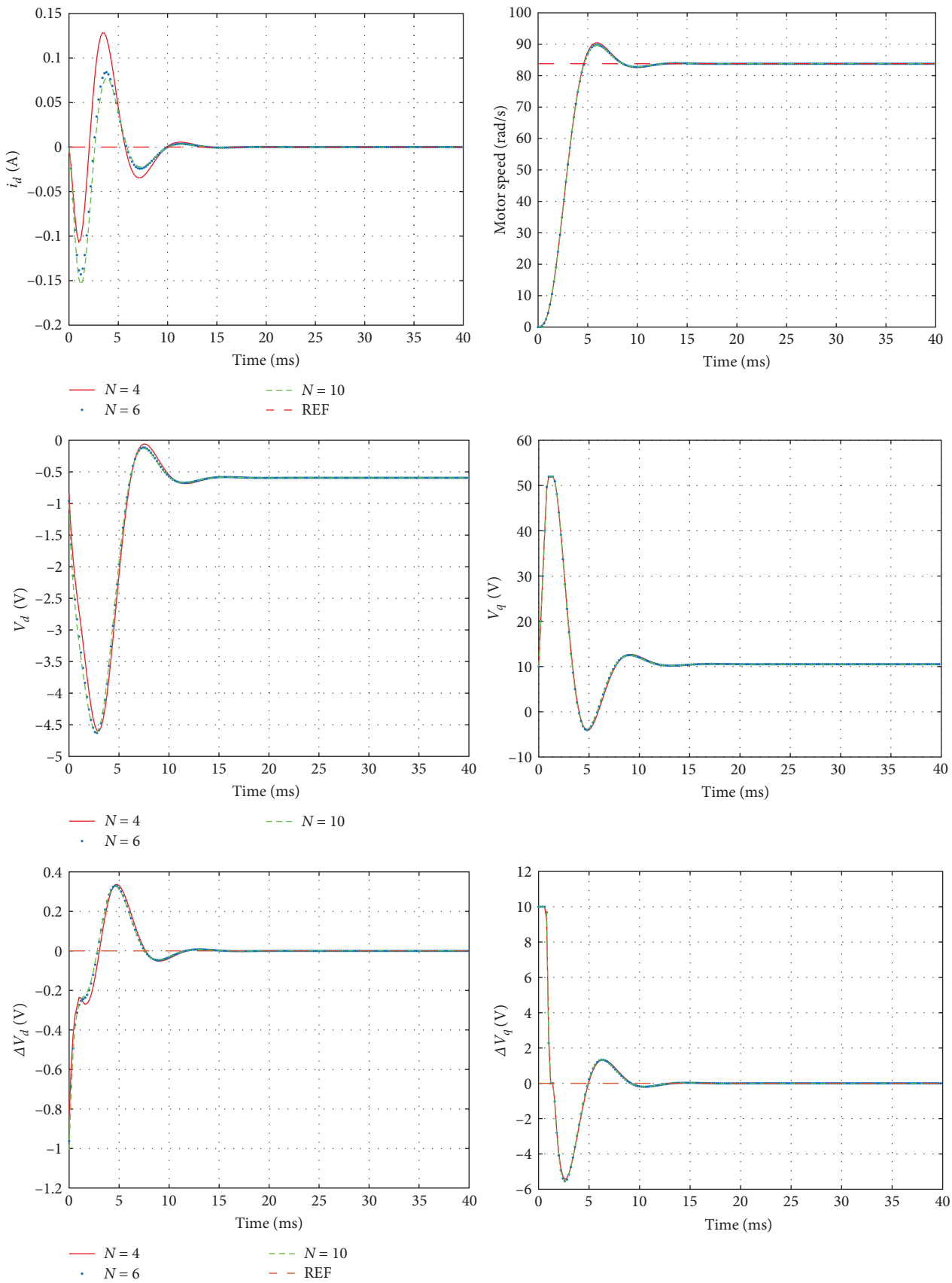

Figure 1: Tuning LMPC with N.

Table 3: Comparison performance indices.

N	Performance metrics	Values
4	Feedback gain	$\begin{bmatrix} 10.1464 & 0.0413 & -0.1461 & 2.7743 & -0.0111 \\ 0.0397 & 8.8157 & 2.8364 & 0.1282 & 0.2639 \end{bmatrix}$
	Eigenvalues	$0.7824 \pm j0.1697,\ 0.9065 \pm j0.1378,\ 0.8337$
	Overshoot (%)	8.1522
	Settling time (ms)	7.8769
6	Feedback gain	$\begin{bmatrix} 9.9049 & 0.1127 & -0.1326 & 2.6542 & -0.0120 \\ 0.0084 & 9.0440 & 2.9496 & 0.1212 & 0.2735 \end{bmatrix}$
	Eigenvalues	$0.7824 \pm j0.1661,\ 0.9028 \pm j0.1390,\ 0.8337$
	Overshoot (%)	6.9892
	Settling time (ms)	7.7807
10	Feedback gain	$\begin{bmatrix} 9.8924 & 0.1186 & -0.1333 & 2.6522 & -0.0125 \\ 0.0049 & 9.0870 & 2.9686 & 0.1210 & 0.2748 \end{bmatrix}$
	Eigenvalues	$0.7875 \pm j0.1661,\ 0.9022 \pm j0.1391,\ 0.8337$
	Overshoot (%)	6.9892
	Settling time (ms)	7.7681

here is that the computational rate is slower than the sampling rate causing computational delay. This computational delay is related to using single MPC controller replacing the cascade control structure and can be solved using encoders [31, 32].

To check whether the trajectories in Figure 1 are optimal, the LMPC is compared with optimal controller, i.e., a DLQR [33, 34]. This optimal controller minimizes the same cost function of Equation (36) particularly when $N_P \to \infty$. Using the same parameters used for the LMPCs, the eigenvalues of DLQR are $0.7875 \pm \mathrm{j}0.1661, 0.9021 \pm \mathrm{j}0.1390,\ 0.8337$ with feedback gain:

$$K_{\mathrm{lqr}} = \begin{bmatrix} 9.8924 & 0.1185 & -0.1334 & 2.6522 & -0.0125 \\ 0.0047 & 9.0901 & 2.9696 & 0.1210 & 0.2748 \end{bmatrix}. \tag{52}$$

As it can be seen from Table 3 as N increases, the control trajectory is trying to converge to the optimal solution generated by using DLQR (for N as large as 10 the LMPC controller is not optimal). To achieve optimality, the exponential data weighting will be used in later sections.

4.4. Response to Load Disturbances. Figure 2 shows a simulation of the torque disturbance in Equation (1). Choose $q_1 = 1$ and $q_2 = 1$ with a Laguerre order of $N = 6$ to ensure a faster dynamic response. Table 3 shows a comparison of the performance indices on the three Laguerre orders. It is observed that the LMPC is able to reject the disturbance and continue to track the reference speed in 5 ms (Figure 2).

4.5. Stability Analysis. In this section, an exponentially decreasing data weighting suggested in Lemma 1 in order to improve the numerical condition while ensuring the closed-loop stability of the LMPC algorithm in the presence of a large prediction horizon has been used. With $Q_L \geqslant 0$, $R_L > 0$, and $N_p \to \infty$, minimizing the cost function in Equation (53) is equivalent to solving the DLQR problem using the algebraic Riccati Equation (53). It is assumed that the pair (A_σ, B_σ) is controllable and (A_σ, D) is observable with $Q_L = D^T D$. The state feedback control gain for the stabilization $\widehat{K}$ is as follows:

$$\widehat{K} = \left(R_L + \sigma^{-2} B^T \widehat{P} B\right) \sigma^{-2} B^T \widehat{P} A. \tag{53}$$

This makes the closed-loop system stable with all poles inside the unit circle and the closed-loop system being described by the following equation:

$$\widehat{x}(m+h+1|m) = \sigma^{-1}\left(A - B\widehat{K}\right)\widehat{x}(m+h|m). \tag{54}$$

From Equation (54), the modified system has all its eigenvalues inside the unit circle taking $N_p \to \infty$. So:

$$\sigma^{-1}|\lambda_{\max}(A - B\widehat{K}| < 1. \tag{55}$$

Thus, by choosing $\sigma > 1$, it is possible to make stable. Several simulations on the PMSM system indicate that a choice of σ greater than unity stabilizes the system. To examine the effects of prediction horizon on numeric condition, compute the Hessian matrix, i.e., $\Pi = \sum_{h=1}^{N_p} \Theta(h) Q_L \Theta(h)^T + R_L$ as the N_p vary. Choose the Laguerre order of $N = 6$, a control horizon of $N_c = 20$ and varying prediction horizon, i.e., $N_{p_0} = 25, N_{p_1} = 50$, and $N_{p_2} = 200$. Table 4 shows the effect of scaling on Hessian matrix as the prediction horizon varies. Specifically, when weights are used on the objective function the conditioning number converges to small finite value. On the other hand, for the cost functions without weighting, the conditioning number becomes too big as N_p

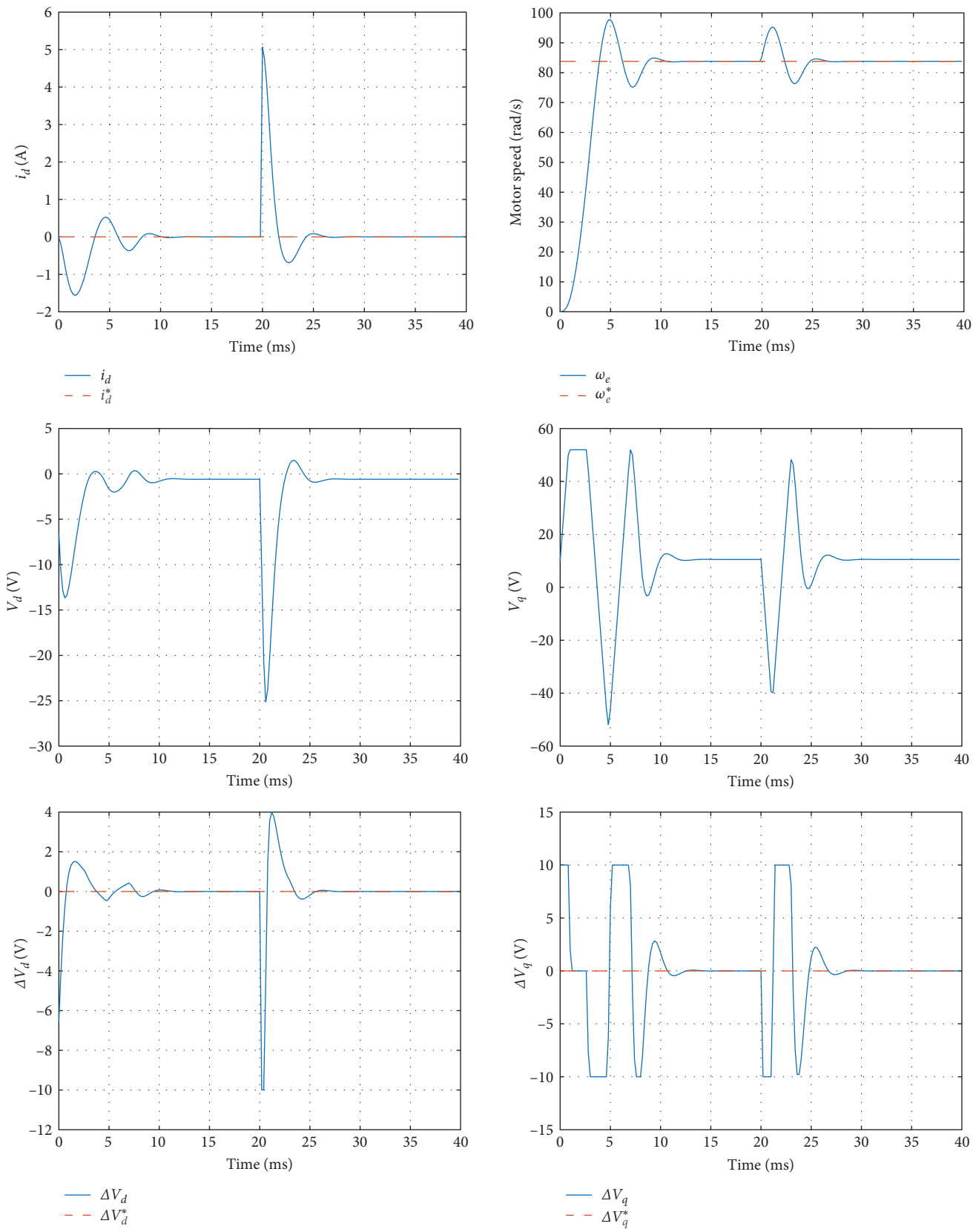

FIGURE 2: Rejection of torque disturbance. d-axis current i_d, motor speed, d-axis voltage: v_d, q-axis voltage: v_q, d-axis incremental voltage: Δv_d, and q-axis incremental voltage: Δv_q. * Operating values.

Table 4: Conditioning number, $\kappa(\Pi)$.

σ	N_p	$\kappa(\Pi)$
	25	97.4225
1	50	701.6131
	200	2.2430×10^4
	25	11.1588
1.2	50	13.6601
	200	10.8304

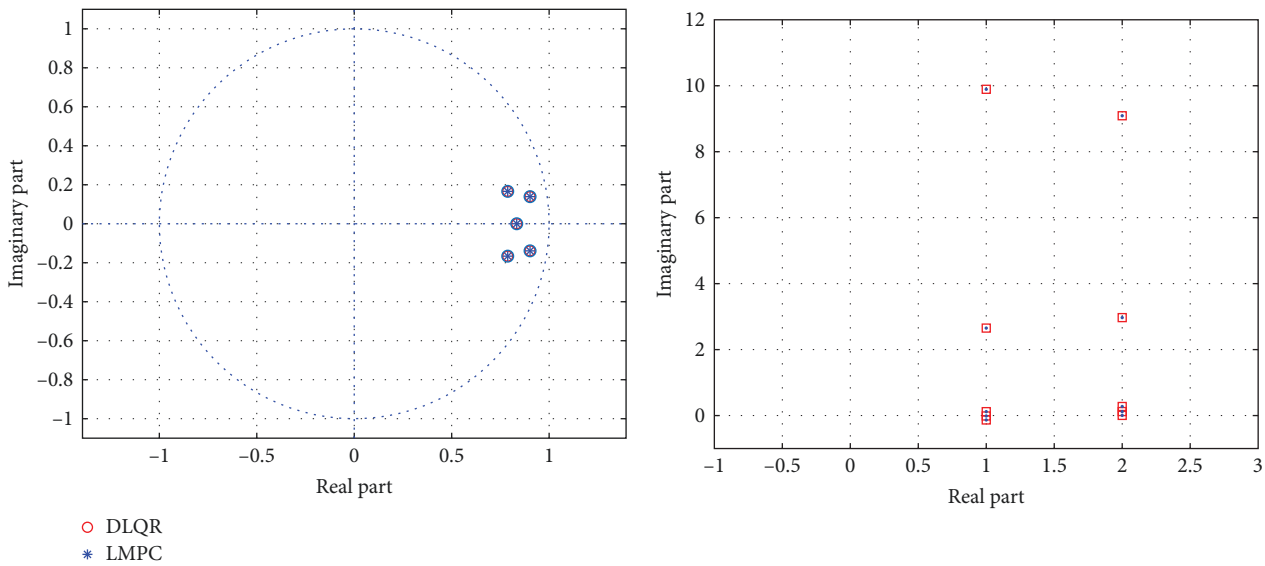

Figure 3: Stability analysis with (a) closed-loop eigenvalues, and (b) gain elements.

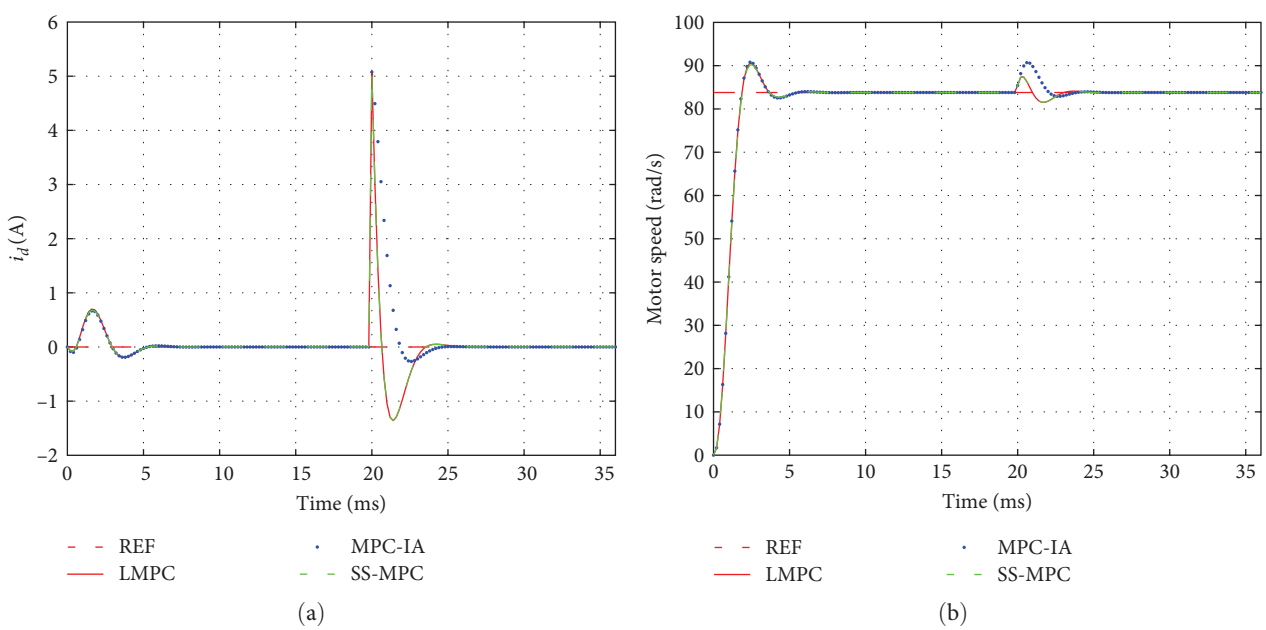

Figure 4: Continued.

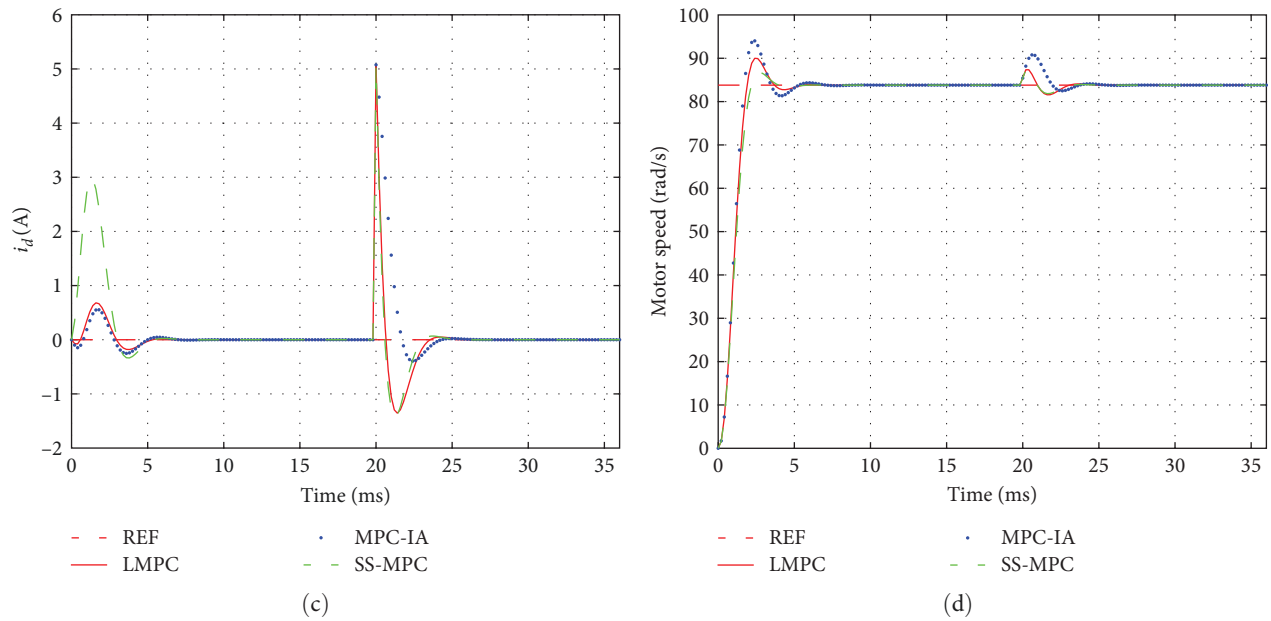

FIGURE 4: Disturbance rejection performance (a) current (i_d) in case of $N_c = 15$, (b) speed ω_e in case of $N_c = 15$, (c) current (i_d) in case of $N_c = 7$, (d) speed ω_e in case of $N_c = 7$.

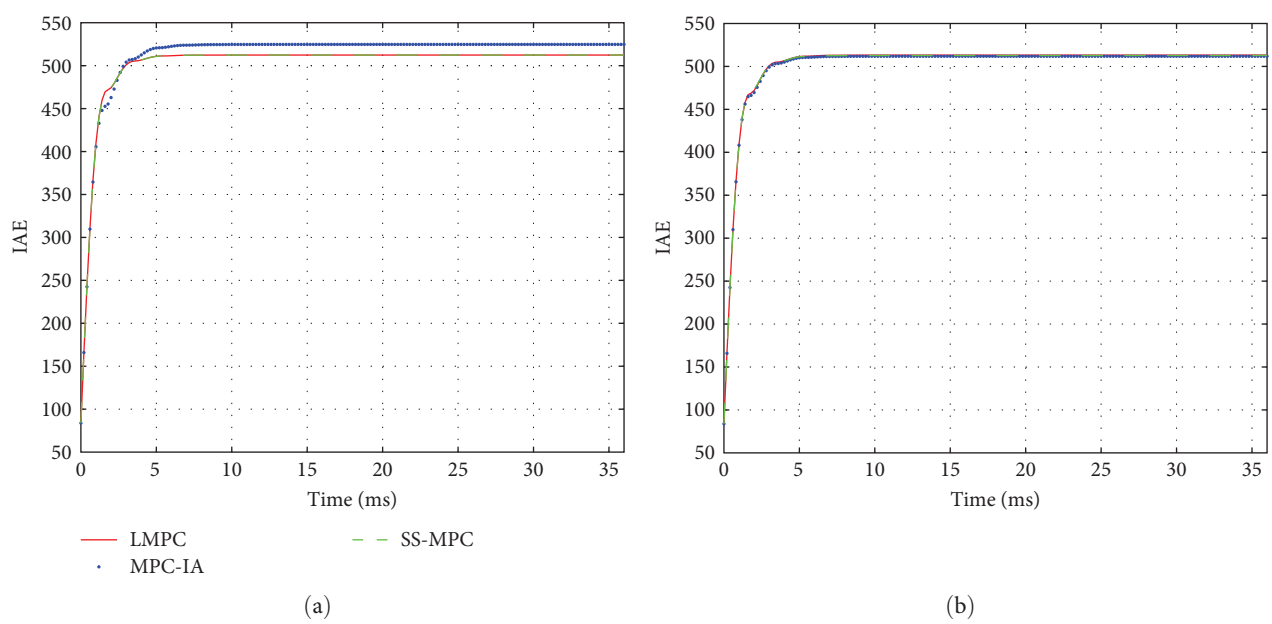

FIGURE 5: Comparison of Integral of Absolute Error (IAE) (a) IAE in case of $N_c = 7$, (b) IAE in case of $N_c = 15$.

inicreases, making the systems unstable and restricting our choose of N_P.

In order to examine the stability of the LMPC controller, check if all the eigenvalues appear inside a unit circle on two-plane. To grant such stability, $\sigma = 1.2$, $N_p = 200$ and $N = 10$ are used. Figure 3 illustrates the eigenvalues of the LMPC (blue stars) coinciding with the DLQR (red circles) and all appearing inside unit circle. Figure 3 confirms that both the control gain matrices of LMPC (blue dots) and DLQR (red squares) are identical to each other with conditioning number 14.7619.

4.6. Performance of LMPC as Compared to SSMPC and MPC-IA. This section focuses on the performance of LMPC by comparing to MPC-IA and SS-MPC when used to regulate PMSM speed (ω_ε) to reference signal (ω_ε^*). Here, the integral of absolute error (IAE) is introduced as performance metric in addition to scheduling time and overshoot

to measure the performance of the controllers. It is defined as follows:

$$\mathrm{IAE} = \sum_{m=1}^{M} |\omega^*(m) - \omega(m)|. \tag{56}$$

Next, examine the response of the three control scheme to step load disturbance. Assume a control horizon of 15 and prediction horizon of 50 to be achieved.

As shown in Figure 4, all three control schemes regulate the speed of PMSM to the reference (83.8 rad/s), though MPC-IA and SS-MPC regulation has achieved at the cost of higher computational complexity, because the LMPC requires only seven parameters ($N = 7$) compared the minimum of 15 parameters in the MPC-IA and SS-MPC (the LMPC uses $N = 7$ with $a = 0.6271$ achieving $N_c = 15$). The settling time and overshoot of the three controllers are the same. It can also be seen that when a load torque is applied, both the LMPC and SS-MPC remove the disturbance with shorter recovery time than MPC-IA (Figure 5). However, the LMPC controller removes the disturbance with a lower computational complexity, requiring only seven control parameters compared to 15 in the case of the SS-MPC.

5. Conclusion

In this paper, the design of Laguerre-based model predictive speed controller for the PMSM system has been extensively investigated. The paper has presented a solution to stability and speed regulation of PMSM systems. Also, exponential data weighing is used to decrease numerical issue, particularly with large prediction horizon. For stability analysis, LMPC has been compared to the optimal DLQR system, whereas for speed tracking LMPC has been compared to a popular state-space MPC and a simple MPC with an integrating action (MPC-IA). Therefore, the proposed controller significantly has better performance compared to MPC-IA controller. The developed LMPC method also exhibited improved steady-state characteristics by successfully eliminating the steady-state error and stability issues. The simulation results also showed that when reference and disturbance changes are encountered, the developed method operates with minimum settling time and overshoot. Therefore, the dynamic properties of the proposed control scheme are similar to SS-MPC controller but with less computational effort while providing more flexibility. By selecting appropriate values of a and N, LMPC reduces the number of parameters required for accurate prediction. Hence, the results obtained in this paper indicate that the control scheme performs very well and is considered feasible.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest.

References

[1] J. Gao, J. Zhang, M. Fan, Z. Peng, Q. Chen, and H. Zhang, "Model predictive control of permanent magnet synchronous motor based on state transition constraint method," *Mathematical Problems in Engineering*, vol. 2021, Article ID 3171417, 11 pages, 2021.

[2] G. Zhang, L. Gao, H. Yang, and L. Mei, "A novel method of model predictive control on permanent magnet synchronous machine with Laguerre functions," *Alexandria Engineering Journal*, vol. 60, no. 6, pp. 5485–5494, 2021.

[3] M. Karabacak and H. I. Eskikurt, "Speed and current regulation of a permanent magnet synchronous motor via nonlinear and adaptive backstepping control," *Mathematical and Computer Modelling*, vol. 53, no. 9-10, pp. 2015–2030, 2011.

[4] R. Krishnan, *Permanent Magnet Synchronous and Brushless DC Motor Drives*, CRC press, 2017.

[5] I. Boldea, "Control issues in adjustable speed drives," *IEEE Industrial Electronics Magazine*, vol. 2, no. 3, pp. 32–50, 2008.

[6] M. Preindl and S. Bolognani, "Model predictive direct speed control with finite control set of PMSM drive systems," *IEEE Transactions on Power Electronics*, vol. 28, no. 2, pp. 1007–1015, 2013.

[7] A. M. Kassem and A. A. Hassan, "Performance improvements of a permanent magnet synchronous machine via functional model predictive control," *Journal of Control Science and Engineering*, vol. 2012, Article ID 319708, 8 pages, 2012.

[8] F. F. M. El-Sousy, "High-performance neural-network model-following speed controller for vector-controlled PMSM drive system," in *2004 IEEE International Conference on Industrial Technology, 2004. IEEE ICIT '04*, vol. 1, pp. 418–424, IEEE, Hammamet, Tunisia, December 2004.

[9] C.-H. Fang, C.-M. Huang, and S.-K. Lin, "Adaptive sliding-mode torque control of a PM synchronous motor," *IEE Proceedings—Electric Power Applications*, vol. 149, no. 3, pp. 228–236, 2002.

[10] J.-W. Jung, V. Q. Leu, T. D. Do, E.-K. Kim, and H. H. Choi, "Adaptive PID speed control design for permanent magnet synchronous motor drives," *IEEE Transactions on Power Electronics*, vol. 30, no. 2, pp. 900–908, 2014.

[11] Y.-S. Kung and M.-H. Tsai, "FPGA-based speed control IC for PMSM drive with adaptive fuzzy control," *IEEE Transactions on Power Electronics*, vol. 22, no. 6, pp. 2476–2486, 2007.

[12] B. Nahid-Mobarakeh, F. Meibody-Tabar, and F.-M. Sargos, "Mechanical sensorless control of PMSM with online estimation of stator resistance," *IEEE Transactions on Industry Applications*, vol. 40, no. 2, pp. 457–471, 2004.

[13] Y. A.-R. I. Mohamed, "Adaptive self-tuning speed control for permanent-magnet synchronous motor drive with dead time," *IEEE Transactions on Energy Conversion*, vol. 21, no. 4, pp. 855–862, 2006.

[14] A. Wang, X. Jia, and S. Dong, "A new exponential reaching law of sliding mode control to improve performance of permanent magnet synchronous motor," *IEEE Transactions on Magnetics*, vol. 49, no. 5, pp. 2409–2412, 2013.

[15] J. M. Maciejowski and M. Huzmezan, "Predictive control," in *Robust Flight Control. Lecture Notes in Control and Information Sciences*, vol. 70, pp. 125–134, Springer, Berlin, Heidelberg, 1997.

[16] T. T. Nguyen, H. N. Tran, T. H. Nguyen, and J. W. Jeon, "Recurrent neural network-based robust adaptive model predictive speed control for PMSM with parameter mismatch," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 6, pp. 6219–6228, 2022.

[17] Z. Li, F. Wang, D. Ke, J. Li, and W. Zhang, "Robust continuous model predictive speed and current control for PMSM with adaptive integral sliding-mode approach," *IEEE Transactions on Power Electronics*, vol. 36, no. 12, pp. 14398–14408, 2021.

[18] S. Bolognani, S. Bolognani, L. Peretti, and M. Zigliotto, "Design and implementation of model predictive control for electrical motor drives," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 6, pp. 1925–1936, 2009.

[19] P. Cortes, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodriguez, "Predictive control in power electronics and drives," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 12, pp. 4312–4324, 2008.

[20] C. M. Abdissa and K. T. Chong, "Stabilization and voltage regulation of the buck DC–DC converter using model predictive of Laguerre functions," *Studies in Informatics and Control*, vol. 26, no. 3, pp. 315–324, 2017.

[21] D. Di Ruscio, "Model predictive control with integral action: a simple MPC algorithm," *Modeling, Identification and Control: A Norwegian Research Bulletin*, vol. 34, no. 3, pp. 119–129, 2013.

[22] A. Brosch, O. Wallscheid, and J. Bocker, "Model predictive control of permanent magnet synchronous motors in the overmodulation region including six-step operation," *IEEE Open Journal of Industry Applications*, vol. 2, pp. 47–63, 2021.

[23] J. F. Gieras, *Permanent Magnet Motor Technology: Design and Applications*, CRC Press, 2009.

[24] A. Akpunar and S. Iplikci, "Runge–Kutta model predictive speed control for permanent magnet synchronous motors," *Energies*, vol. 13, no. 5, Article ID 1216, 2020.

[25] S. P. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[26] S. Vazquez, J. Rodriguez, M. Rivera, L. G. Franquelo, and M. Norambuena, "Model predictive control for power converters and drives: advances and trends," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 2, pp. 935–947, 2017.

[27] L. Wang, "Discrete time model predictive control design using Laguerre functions," in *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, vol. 3, pp. 2430–2435, IEEE, Arlington, VA, USA, June 2001.

[28] L. Wang, "Use of exponential data weighting in model predictive control design," in *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No.01CH37228)*, vol. 5, pp. 4857–4862, IEEE, Orlando, FL, USA, December 2001.

[29] B. D. O. Anderson and J. B. Moore, *Optimal Control: Linear Quadratic Methods*, Courier Corpo-Ration, 2007.

[30] A. Mohammadi, H. Asadi, S. Mohamed, K. Nelson, and S. Nahavandi, "MPC-based motion cueing algorithm with short prediction horizon using exponential weighting," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 000521–000526, IEEE, Budapest, Hungary, October 2016.

[31] X. Guo, M. He, and Y. Yang, "Over modulation strategy of power converters: a review," *IEEE Access*, vol. 6, pp. 69528–69544, 2018.

[32] L. Shen, "DSP-solution for high-resolution position with sin/cos-encoders," in *2010 International Conference on System Science, Engineering Design and Manufacturing Informatization*, vol. 2, pp. 285–288, IEEE, Yichang, China, November 2010.

[33] M. Staebler, "Tms320f240 DSP-solution for high-resolution position with sin/cos-encoders," *Technical Report*, 2000.

[34] J. Hespanha, *Linear Systems Theory*, Vol. 156, Prentice-Hall Englewood Cliffs, NJ, 1980.