

Article

Novel Model Predictive Control Strategies for PMSM Drives: Reducing Computational Burden and Enhancing Real-Time Implementation

Mohamed Salah ^{1,*}, Kotb B. Tawfiq ^{2,3,4,5} , Arafa S. Mansour ⁶  and Ahmed Farhan ¹

¹ Electrical Engineering Department, Faculty of Engineering, Fayoum University, Fayoum 63514, Egypt; afm01@fayoum.edu.eg

² Department of Electrical Engineering and Computer Science, Khalifa University, Abu Dhabi 127788, United Arab Emirates; kotb.basem@ugent.be

³ Department of Electromechanical, Systems and Metal Engineering, Ghent University, 9000 Ghent, Belgium

⁴ FlandersMake@Ugent-Corelab MIRO, 3001 Leuven, Belgium

⁵ Electrical Engineering Department, Faculty of Engineering, Menoufia University, Shibin El Kom 32511, Egypt

⁶ Electrical Engineering Department, Faculty of Engineering, Beni-Suef University, Beni-Suef 62511, Egypt; arafa.sayed@eng.bsu.edu.eg

* Correspondence: msm24@fayoum.edu.eg

Abstract

Model predictive control (MPC) has emerged as a favorable control approach for PMSM drives, though its practical deployment is frequently hindered by superior computational complexity and execution burden. This paper presents four finite control set MPC (FCS-MPC) techniques applied to a two-level inverter-fed PMSM drive. Two of the approaches are conventional methods, while the other two are novel developed strategies proposed in this paper. The novel techniques focus on significantly decreasing computational burdens by employing an efficient space-vector selection mechanism that quickly selects the optimum switching vector without exhaustive evaluation. A comprehensive comparative assessment of all four control methods is conducted under various operating conditions, evaluating their dynamic and steady-state performance, computational requirements, and real-time feasibility. Simulation results demonstrate that the proposed techniques achieve a significant reduction in computational effort and faster processing, up to 39.65% faster than conventional full-state evaluation, while maintaining control performances comparable to conventional techniques. These results highlight the potential of the proposed MPC approaches to bridge the gap between advanced control theory and practical implementation in real-time PMSM drive systems, providing effective solutions for installing high-performance PMSM drives on hardware with limited resources.

Keywords: permanent magnet synchronous motor; model predictive control; current predictive control; three-phase two level inverter



Academic Editor: Ahmed Abu-Siada

Received: 8 September 2025

Revised: 26 September 2025

Accepted: 29 September 2025

Published: 2 October 2025

Citation: Salah, M.; Tawfiq, K.B.; Mansour, A.S.; Farhan, A. Novel Model Predictive Control Strategies for PMSM Drives: Reducing Computational Burden and Enhancing Real-Time Implementation. *Machines* **2025**, *13*, 908. <https://doi.org/10.3390/machines13100908>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Known for their exceptional performance characteristics, permanent magnet synchronous motors (PMSMs) are among the most commonly utilized devices in electric drive systems. The remarkable torque-to-current ratio, outstanding efficiency, and high power density of PMSMs are some of its significant benefits [1]. There is no requirement for rotor currents because PMSMs use permanent magnets in the rotor, which significantly lowers

copper losses. Furthermore, reduced rotor copper loss and the lack of brushes and slip rings result in an improved performance over traditional machines [2].

Field-oriented control (FOC) and direct torque control (DTC) are the two commonly used conventional speed control methods for PMSMs. Field-oriented control (FOC) separates the torque and flux components through an appropriate coordinate transformation, allowing PMSM control to mimic that of a direct current (DC) motor [3]. Proportional–integral (PI) controllers are commonly used in the literature when FOC is applied to PMSMs in order to control the speed and regulate the i_q and i_d components [4,5]. FOC's ability to provide superior steady-state performance and good torque quality are its main advantages. Nevertheless, it has drawbacks such as a decreased dynamic response because of PI controllers' constrained bandwidth and the difficulty of adjusting their parameters [5].

On the other hand, DTC works by choosing the right inverter switching states to directly regulate stator torque and flux without the need for current control loops. This approach uses hysteresis controllers to adjust the error between reference and estimated values of torque and flux linkage [6] and flux and torque estimators that are taken from the PMSM model. The removals of current control loops, coordinate transformations, and PI regulators are some of the main advantages of DTC. However, DTC can lead to disadvantages like a torque ripple and excessive current. A self-tuning stator flux observer was proposed in [7] to improve the overall performance of DTC in PMSM systems, while improvements were made in [8] to reduce torque ripples and enhance dynamic torque responsiveness.

In recent decades, model predictive control (MPC) has developed as a promising alternative to conventional control approaches. Its application has been widely investigated in various motor drive systems, including synchronous reluctance motors, brushless DC motors, induction machines, and both interior and surface-mounted permanent magnet synchronous motors (PMSMs) [9]. MPC operates by employing a discrete-time system model to predict future behaviors over a defined prediction horizon [10]. The optimal control action is then determined by minimizing a cost function that reflects the desired control objectives [10].

Model predictive control (MPC) has developed a highly attractive approach for power electronic applications, owing to its inherently discrete design, which aligns well with converter switching behavior [11]. In finite control set MPC (FCS-MPC), the optimal switching state is obtained from a predefined admissible set by minimizing the deviation between the reference signals and the predicted system responses. Despite its effectiveness, conventional FCS-MPC implementations need the evaluation of all possible switching states through repeated prediction of load and source currents, followed by a cost function calculation, which executes a significant computational burden.

Regardless of its potential, MPC's real-time processing demand becomes a serious issue, especially when considering the high sampling frequencies and long prediction horizons needed for the best results. In order to overcome this difficulty, one particular approach is to shift a portion of the computation offline and use an explicit solution, which is a precomputed lookup table, to apply the control law [12].

Numerous optimization procedures have been investigated to reduce the computational load of MPC. For instance, multistep FCS-MPC [13] uses a modified sphere decoding algorithm to simplify long-horizon predictions, although Zafra et al. [14] redeveloped the problem as a box-constrained integer least-squares job. Other methods reduce computational load by limiting the control set [15] through graphical [16], adjacent-vector [17], or commutation-based selections [15]. Similarly, Habibullah et al. [18] employed a DTC-based switching table, and subsequent works [19,20] additionally reduced the vector states. In [21], a simplified method that requires only a single prediction was introduced, eliminat-

ing the need to evaluate all possible vectors. Stabilizing terminal constraint-free nonlinear MPC through a sliding mode-based terminal cost significantly reduces the computational burden compared with classical approaches as in [22].

By exploiting the finite number of inverter switching states, finite control set model predictive control (FCS-MPC) has emerged as an extensively used approach that decreases computational complexity in real-time applications [23]. The effectiveness of conventional MPC for current control has been investigated across various power converters [24–26] and motor drive systems [27,28]. Nevertheless, the processing limitations of standard hardware remain a major challenge in industrial applications, as conventional MPC requires evaluating the predicted future currents for each possible voltage vector generated by the inverter [23,29,30]. In multilevel inverter systems, where there are substantially more potential voltage vectors—for example, 19 for three-level and 61 for five-level inverters—this problem becomes even more serious. As a result, each sample interval requires 19 or 61 evaluations of the cost function [31]. The short sampling times needed for traditional MPC to function well are in conflict with this high computational load [32]. Furthermore, MPC's capacity to fully utilize its theoretical benefits is constrained by the computational load, particularly in multilevel inverters [23,30], and [33]. Field programmable gate arrays (FPGAs) are a potential approach to avoid these restrictions by accelerating calculations [34]. Furthermore, especially in cascaded H-bridge inverter topologies, Patricio Cortes and colleagues proposed a technique for lowering the number of evaluated vectors by choosing a subset according to a predetermined distance criterion [26]. A technique is presented in [31] that selects a narrow subset of voltage vectors that are similar to those used in space vector pulse width modulation (SVPWM), thereby reducing computational complexity significantly. This method is exceptionally scalable and suitable for multilevel inverter topologies since it successfully restricts the number of voltage vectors taken into account during prediction and optimization to just three.

This study investigates two novel methods aimed at enhancing the efficiency of model predictive control (MPC) in multilevel inverter systems. The first approach reduces computational complexity by restricting the number of candidate voltage vectors from seven vectors to only two in each control cycle. This approach minimizes the processing burden without compromising control accuracy, thereby enabling faster implementation, improved dynamic response, and practical implementation on cost-effective hardware with limited computational capacity. The second strategy employs a direct selection method that identifies the optimal voltage vector without iterative evaluation of a cost function. By eliminating this computational step, the approach hurries decision-making, reduces latency, and enhances real-time performance, particularly in high-frequency switching applications. Both methods can be successfully implemented in higher-level inverter topologies and are naturally scalable. They help increase control accuracy, system stability, and overall hardware efficiency under a variety of operating circumstances in addition to lowering computing complexity. These techniques mark a significant step forward for the implementation of MPC in contemporary electric drive and power electronic systems in a way that is realistic, fast, and resource efficient.

The remainder of this paper is organized as follows: Section 2 presents the mathematical modeling of the system, including the PMSM and the two-level inverter. Section 3 presents the conventional model predictive control techniques applied to PMSMs. Section 4 investigates the two proposed MPC methods for the motor to reduce the computational burden. Sections 5 and 6 discuss the comparative analysis and simulation results of all techniques. Finally, Section 7 provides the conclusions of the study.

2. System Modeling

2.1. PMSM Modeling

The following are the mathematical modeling equations for the PMSM in the $dq0$ rotating reference frame [35,36],

$$\lambda_q = L_q i_q, \quad (1)$$

$$\lambda_d = L_d i_d + \lambda_m, \quad (2)$$

$$v_q = (r_s + pL_q)i_q + \omega_r L_d i_d + \omega_r \lambda_m, \quad (3)$$

$$v_d = (r_s + pL_d)i_d - \omega_r L_q i_q, \quad (4)$$

$$T_e = \frac{3}{2} \frac{P}{2} [\lambda_m i_q + (L_d - L_q) i_d], \quad (5)$$

$$T_e = T_l + B\omega_m + j \frac{d\omega_m}{dt}. \quad (6)$$

where (v_q, v_d) , (i_q, i_d) , and (λ_q, λ_d) denote the stator voltages, currents, and flux linkages in the dq -reference frame. The stator winding resistance is represented by r_s . ω_r represents the electrical rotor angular speed and ω_m stands for the mechanical rotor angular speed. λ_m denotes the magnetic rotor flux linkage; p symbolizes the time differentiator, T_e, T_l represents the electromagnetic torque and torque of the load; and b, P , and j are the viscous friction of the motor, the total poles numbers, and the inertia of the machine, respectively. In our model, we considered a round-rotor PMSM, where $L_d = L_q$. The equivalent PMSM circuit might easily be signified as shown in Figure 1.

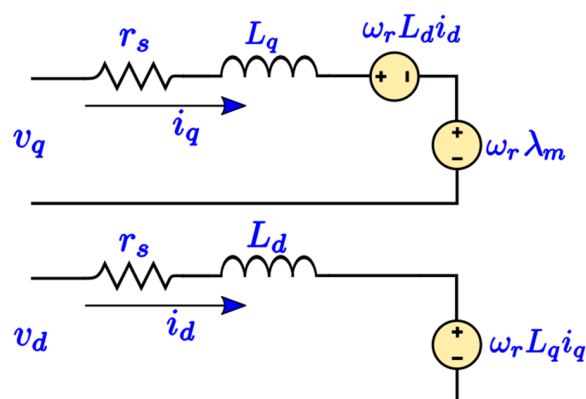


Figure 1. The equivalent PMSM circuit [35].

2.2. Inverter Model

As presented in Figure 2a, the three-phase two-level inverter has been tied to the PMSM for providing three-phase voltage since it is thought to be the most typically used inverter in power drive applications. According to Table 1 and Figure 2b, this specific type of inverter can produce eight voltage vectors from eight switching states. The dc-link voltage and switching vector in Equation (7) could be used to calculate the stator voltages v_{abc} .

$$v_{abc} = \frac{1}{3} V_{dc} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix} S_s^{abc}. \quad (7)$$

Equation (8) applies the Clarke transformation for transferring the stator voltages from the abc stationary frame to the $\alpha\beta$ stationary frame.

$$v_{\alpha\beta} = \frac{2}{3} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix}. \quad (8)$$

The output voltages of the PMSM in the dq rotating reference frame are calculated using Park transformation and expressed in terms of the switching vector [37] of the inverter in Equation (9).

$$v_{dq} = \begin{bmatrix} \cos \theta_r & \sin \theta_r \\ -\sin \theta_r & \cos \theta_r \end{bmatrix} \begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix}. \quad (9)$$

where θ_r is the rotor electrical position.

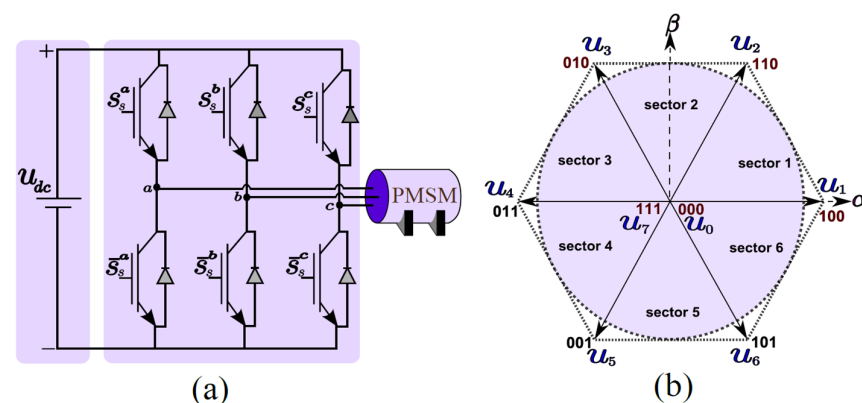


Figure 2. (a) A three-phase inverter provides power to a permanent magnet synchronous motor; (b) the space vector of three-phase two-level inverter [36].

Table 1. Two-level inverter switching state and voltage vectors.

Inverter State	S_A, S_B, S_C	$\bar{S}_A, \bar{S}_B, \bar{S}_C$	$\frac{V_A}{V_{dc}}$	$\frac{V_B}{V_{dc}}$	$\frac{V_C}{V_{dc}}$
0	000	111	0	0	0
1	100	011	2/3	-1/3	-1/3
2	110	001	1/3	1/3	-2/3
3	010	101	-1/3	2/3	-1/3
4	011	100	-2/3	1/3	1/3
5	001	110	-1/3	-1/3	2/3
6	101	010	1/3	-2/3	1/3
7	111	000	0	0	0

3. Conventional Algorithms for Control PMSM

3.1. Conventional MPC_I—Full-State Evaluation

Inverter control, a full-state assessment technique, is one of the most well-established approaches. As shown in Figure 2b, each voltage vector that corresponds to a switching state is evaluated with the objective of identifying the one that generates current responses that most closely match the reference currents.

To apply the current model predictive control and obtain the following anticipated sampling instant states, the permanent magnet synchronous motor continuous time model has to be discretized [34], using the Euler approximation technique as described in the following:

$$\frac{dx}{dt} = \frac{x(k+1) - x(k)}{T_s}, \quad (10)$$

where T_s is the sampling time.

Consequently, the following future value of the PMSM currents may be predicted:

$$i_d(k+1) = \left(1 - \frac{R_s T_s}{L_d}\right) i_d(k) + T_s \omega_r(k) i_q(k) + \frac{T_s}{L_d} v_d(k), \quad (11)$$

$$i_q(k+1) = \left(1 - \frac{R_s T_s}{L_q}\right) i_q(k) - T_s \omega_r(k) i_d(k) + \frac{T_s}{L_q} v_q(k) - \frac{T_s \lambda_m}{L_q} \omega_r(k). \quad (12)$$

Using the current stator currents $i_q(k)$ and $i_d(k)$, stator voltages $v_q(k)$ and $v_d(k)$, and speed $\omega_r(k)$ as a basis, $i_q(k+1)$ and $i_d(k+1)$ represent the stator currents at the upcoming sampling time. T_s is the sampling time.

The stator voltages v_{dq} could be calculated using Equations (7)–(9). Using Equations (11) and (12), the seven inverter voltage vectors are used to obtain seven corresponding stator current predictions. Next, applying a cost function established in [37,38], the most optimal inverter switching vector is determined. The following is the cost function utilized by this study:

$$g = \left[i_{dref}(k+1) - i_d(k+1)\right]^2 + \left[i_{qref}(k+1) - i_q(k+1)\right]^2. \quad (13)$$

The conventional MPC₁ algorithm can be presented as displayed in Figure 3 and its flow chart shown as

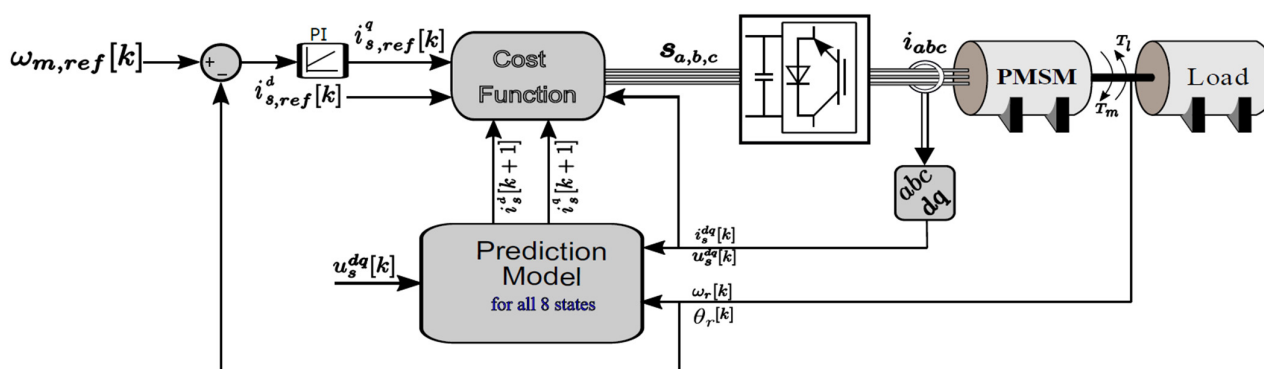


Figure 3. The conventional MPC₁ block diagram for controlling PMSM [35,36].

As Algorithm 1 illustrates, there are several calculations involved in each sample period. Some of these are independent of the number of voltage vectors and are related to system attributes like flux calculation, rotor speed, or stator resistance. On the other hand, a large percentage of the computations are directly related to the inverter voltage vectors and scale in proportion to their quantity. One of these is the need to recalculate the stator current predictions for every potential switching state at each sampling time. The main emphasis of this research is vector-dependent calculations, which comprise the majority of the computational load and are where the proposed enhancements would have the most impact. Therefore, it is clear that the controller has to achieve the following:

- Calculate the voltage vector in the dq -frame via Equations (8) and (9). [executed for all voltage vectors]
- Calculate the predicted currents in the dq -frame via Equations (11)–(12). [executed for all voltage vectors]
- Calculate the cost function using Equation (13). [executed for all voltage vectors]

Algorithm 1: Algorithm of the conventional MPC_I

-
- Step 1: Read $i_q, i_d, i_{q_ref}, i_{d_ref}$, and ω_r .
 Step 2: Calculate the voltages $v_{dq}(k)$ at each state by employing Equations (8) and (9).
 Step 3: Calculate the predicted currents for each switching state $i_d(k+1)$ and $i_q(k+1)$ using Equations (11) and (12).
 Step 4: The cost function is computed using Equation (13), which evaluates the deviation between the predicted currents corresponding to each switching state and the reference currents.
 Step 5: The switching state that yields the minimum cost function value is then selected.
-

In the case of a two-level inverter, there are seven voltage vectors, resulting in approximately 35 equation evaluations per sampling interval. However, in multilevel inverters, the number of voltage vectors increases exponentially with the number of levels—for example, 19 vectors in a three-level inverter and 61 in a five-level inverter.

This exponential growth in computational load poses a significant challenge for real-time implementation, especially in high-level inverters. It can negatively impact both control performance and system efficiency, making computational complexity a major limiting factor in the conventional MPC_I.

3.2. Conventional MPC_{II}—Three-State Evaluation

A simpler method called conventional MPC_{II} has been devised to lessen the computing burden that comes with conventional MPC_I, which evaluates every potential voltage vector at every sampling moment [31]. Conventional MPC_{II} determines the necessary voltage vector and limits the assessment to those vectors inside the same sector, as compared with examining each switching state. This approach, which draws inspiration from the space vector modulation (SVM) method, greatly reduces processing requirements without sacrificing control precision, making it more appropriate for real-time applications.

The first step is to calculate the required inverter voltages in the dq -reference frame using Equations (14) and (15) so that the predicted currents match their reference values [36–38].

$$v_d^*(k) = R_s * i_d(k) + \left(\frac{L_d}{T_s}\right) [i_{dref}(k+1) - i_d(k)] - L_d \omega_r i_q(k), \quad (14)$$

$$v_q^*(k) = R_s * i_q(k) + \left(\frac{L_q}{T_s}\right) [i_{qref}(k+1) - i_q(k)] + L_q \omega_r i_d(k) + \omega_r \lambda_m. \quad (15)$$

Secondly, using the inverse Park transformation, transform the calculated inverter voltages in the rotating dq -reference frame into the inverter voltages in the $\alpha\beta$ (alpha–beta) stationary frame using the inverse Park transformation.

$$v_{\alpha\beta}^* = \begin{bmatrix} \cos \theta_r & -\sin \theta_r \\ \sin \theta_r & \cos \theta_r \end{bmatrix} \begin{bmatrix} v_d^* \\ v_q^* \end{bmatrix}. \quad (16)$$

Next, use Equation (17) to determine the voltage vector's angle in the $\alpha\beta$ stationary reference frame.

$$\alpha(k) = \text{atan2}[v_\beta(k+1), v_\alpha(k+1)] \quad (17)$$

Based on this angle, the corresponding sector is identified (as shown in Figure 2b and Table 2), which determines the three most relevant voltage vectors for the next prediction step. For instance, if the angle lies between 0° and 60° , the vectors in sector 1 (e.g., vectors 0,

1, and 2) are selected. If the angle is between 60° and 120° , vectors in sector 2 (e.g., vectors 0, 2, and 3) are considered, and so forth.

Table 2. Angle range and its corresponding candidate vectors for Algorithm 2 for 2-level inverters.

Angle Range	Sector	Candidate Vectors
$0^\circ \leq \alpha < 60^\circ$	1	0, 1, 2
$60^\circ \leq \alpha < 120^\circ$	2	0, 2, 3
$120^\circ \leq \alpha < 180^\circ$	3	0, 3, 4
$180^\circ \leq \alpha < 240^\circ$	4	0, 4, 5
$240^\circ \leq \alpha < 300^\circ$	5	0, 5, 6
$300^\circ \leq \alpha < 360^\circ$	6	0, 6, 1

Lastly, the three determined voltage vectors are the only ones to which the prediction and optimization stages of the conventional MPC_{II} are applied. Equations (11) and (12) are used to compute the predicted currents, and Equation (13), which evaluates the cost function, is then used. This selected method preserves efficient control performance while drastically lowering computational complexity.

As illustrated in Algorithm 2, perform several computations for each sampling interval. Specifically, the controller must achieve the following:

- Calculate v_{dq}^* , then $v_{\alpha\beta}^*$, then $\alpha(k)$ via Equations (14)–(17). [executed once]
- Identify the sector (via if condition or equivalent equation) [executed once]
- Calculate the voltage vector in the dq-frame via Equations (8) and (9). [executed for three voltage vectors]
- Calculate the predicted currents in the dq-frame via Equations (11) and (12). [executed for three voltage vectors]
- Calculates the cost function using Equation (13). [executed for three voltage vectors]

Algorithm 2: Algorithm of the conventional MPC_{II}

- Step 1: Read $i_q, i_d, i_{q_ref}, i_{d_ref}$, and ω_r .
- Step 2: Calculate the required inverter voltage v_{dq}^* using Equations (14) and (15).
- Step 3: Calculate the voltage in $\alpha\beta$ stationary frame $v_{\alpha\beta}^*$ using Equation (16).
- Step 4: Determine the angle of this voltage using Equation (17).
- Step 5: Determine sector from $\alpha(k)$ based on Table 2.
- Step 6: Calculate the voltage vectors v_{dq} for the 3 candidate vectors in the dq-reference frame using Equations (8) and (9).
- Step 7: Calculate predicted currents for the 3 candidate vectors $i_d(k+1)$ and $i_q(k+1)$ using Equations (11) and (12).
- Step 8: Compute the cost function for the candidate vectors using Equation (13), which typically evaluates the error between the predicted currents and reference currents.
- Step 9: Select the switching state that results in the minimum cost function value.
-

For a two-level inverter, the algorithm requires approximately 20 equation evaluations per sampling interval. Notably, for higher-level inverters, the number of equation evaluations remains constant at 20; only the conditional function—or its equivalent formulation—changes. The block diagram of the conventional MPC_{II} is shown in Figure 4.

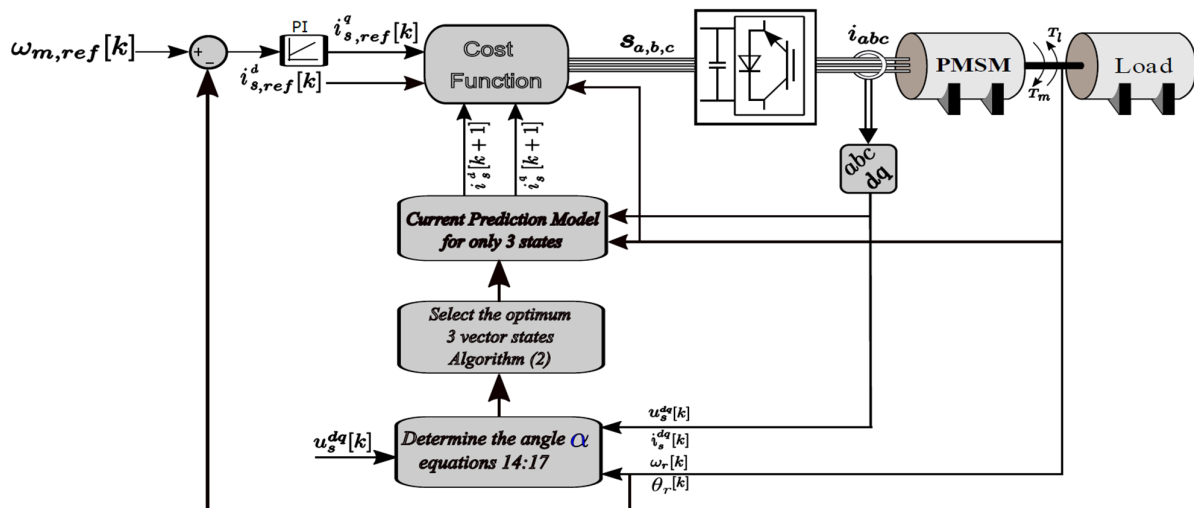


Figure 4. The conventional MPC_{II} block diagram for controlling PMSM.

4. Proposed Algorithms for Control PMSM

4.1. Proposed MPC_I—Two-State Evaluation

For the purpose of achieving improvements, this study presents a new method that only uses two voltage vectors. The fundamental concept derives from examining the three voltage vectors utilized in conventional MPC_{II}, which could be understood as those nearest to the reference vector (i.e., the required inverter voltage). Building on this realization, a little change is made to the vector space sectors: they are moved by 30° rather than being defined at 0° with a 60° span each. As a result, the first sector now spans from −30° to 30° (instead of 0° to 60°) while the second sector runs from 30° to 90° (instead of 60° to 120°), as shown in Figure 5.

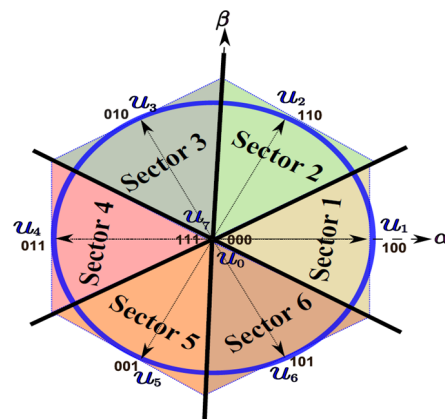


Figure 5. Space vector for the two-state MPC algorithm.

With this modification, the algorithm only has the option of selecting two voltage vectors (e.g., vectors 0 and 1) when the reference voltage vector is located within a certain sector, such as sector 1. This simplifies the control strategy. The process is identical to that utilized by conventional MPC_{II}. First, Equations (14) and (15) are used to determine the necessary inverter voltage. Equation (16) is then used to derive the reference voltages in the $\alpha\beta$ reference frame ($v_{\alpha\beta}^*$). Equation (17) is then used to calculate this voltage vector's angle.

The primary distinction is seen in the angular ranges and the candidate vectors that correspond to them. In order to identify the two appropriate voltage vectors for the following prediction step, the corresponding sector is determined based on this angle (as seen in Figure 5 and Table 3). Vectors 0 and 1 in sector 1 are determined, for example, if

the angle falls between -30° and 30° but consideration is given to sector 2 vectors (such as 0 and 2) if the angle falls between 30° and 90° , and so on. The conventional MPC's prediction and optimization stages are then applied, but just to the two voltage vectors that were chosen. Equations (11) and (12) are used to compute the expected currents, and Equation (13), which evaluates the cost function, is then used.

Table 3. Angle range and its corresponding candidate vectors for Algorithm 3 for 2-level inverters.

Angle Range	Sector	Candidate Vectors
$330^\circ \leq \alpha < 30^\circ$	1	0, 1
$30^\circ \leq \alpha < 90^\circ$	2	0, 2
$90^\circ \leq \alpha < 150^\circ$	3	0, 3
$150^\circ \leq \alpha < 210^\circ$	4	0, 4
$210^\circ \leq \alpha < 270^\circ$	5	0, 5
$270^\circ \leq \alpha < 330^\circ$	6	0, 6

As illustrated in Algorithm 3, perform several computations for each sampling interval. Specifically, the controller must achieve the following:

- Calculate v_{dq}^* , then $v_{\alpha\beta}^*$, then $\alpha(k)$ via Equations (14)–(17). [executed once]
- Identify the sector (via *if* condition or equivalent equation) [executed once]
- Calculate the voltage vector in the dq -frame via Equations (8) and (9). [executed for 2 voltage vectors]
- Calculate the predicted currents in the dq -frame via Equations (11) and (12). [executed for 2 voltage vectors]
- Calculate the cost function using Equation (13). [executed for 2 voltage vectors]

Algorithm 3: Algorithm of the proposed MPC_I

Step 1: Read $i_q, i_d, i_{q_ref}, i_{d_ref}$, and ω_r .

Step 2: Calculate the required inverter voltage v_{dq}^* using Equations (14) and (15).

Step 3: Calculate the voltage in $\alpha\beta$ stationary frame $v_{\alpha\beta}^*$ using Equation (16).

Step 4: Determine the angel of this voltage using Equation (17).

Step 5: Determine the sector from $\alpha(k)$ based on Table 3.

Step 6: Calculate the voltage vectors v_{dq} for the two candidate vectors in the dq -reference frame using Equations (8) and (9).

Step 7: Calculate predicted currents for the two candidate vectors $i_d(k+1)$ and $i_q(k+1)$ using Equations (11) and (12).

Step 8: Compute the cost function for the candidate vectors using Equation (13), which typically evaluates the error between the predicted currents and reference currents.

Step 9: Select the switching state that results in the minimum cost function value.

For a two-level inverter, the algorithm requires approximately 15 equation evaluations per sampling interval. Notably, for higher-level inverters, the number of equation evaluations remains constant at 15; only the conditional function—or equivalent expression—changes. The overall block diagram of the proposed MPC_I is shown in Figure 6.

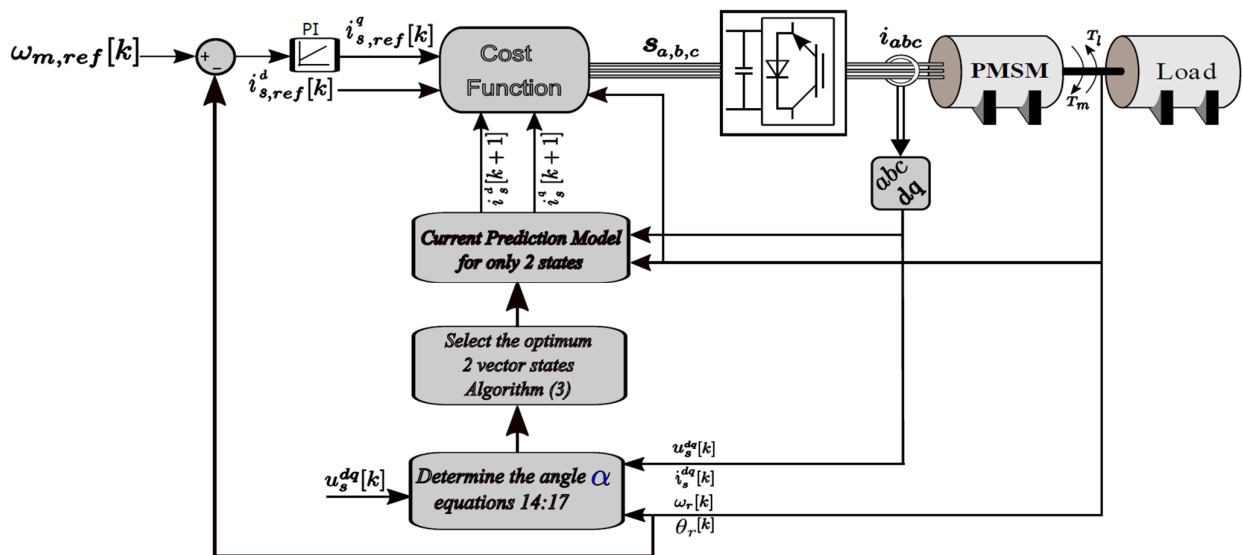


Figure 6. The proposed MPC_I block diagram for controlling PMSM.

4.2. Proposed MPC_{II}—Direct Evaluation

To continue enhancing performance even further, this study presents a different, novel method that finds the optimum voltage vector immediately, removing the need to calculate all of the voltage vectors in the dq -frame, predicted currents, and related cost function. These computations are typically required in order to assess the cost function and choose the best switching state. Nevertheless, the procedure can be greatly accelerated if the optimal state can be identified beforehand, avoiding the full set of calculations.

Using the anticipated current expressions to simplify the cost function after a thorough analysis is the core idea of the proposed MPC_{II} approach. Equations (11)–(13), in particular, are combined to provide a simplified cost function:

$$g = \left[i_{dref}(k+1) - i_d(k+1) \right]^2 + \left[i_{qref}(k+1) - i_q(k+1) \right]^2 \quad (18)$$

By substituting form Equations (11) and (12), we obtain

$$g = \left[i_{dref}(k+1) - \left(1 - \frac{R_s T_s}{L_d} \right) i_d(k) - T_s \omega_r(k) i_q(k) - \frac{T_s}{L_d} v_d(k) \right]^2 + \left[i_{qref}(k+1) - \left(1 - \frac{R_s T_s}{L_q} \right) i_q(k) + T_s \omega_r(k) i_d(k) - \frac{T_s}{L_q} v_q(k) + \frac{T_s \lambda_m}{L_q} \omega_r(k) \right]^2 \quad (19)$$

By performing some simplifications, we obtain

$$g = \left[\left(\frac{T_s}{L_d} \right) \left[\frac{i_{dref}(k+1) - i_d(k)}{T_s} + R_s i_d(k) - L_d \omega_r(k) i_q(k) - v_d(k) \right] \right]^2 + \left[\left(\frac{T_s}{L_q} \right) \left[\frac{i_{qref}(k+1) - i_q(k)}{T_s} + R_s i_q(k) + L_q \omega_r(k) i_d(k) + \lambda_m \omega_r(k) - v_q(k) \right] \right]^2 \quad (20)$$

Using Equations (14) and (15), we obtain

$$g = \left[\left(\frac{T_s}{L_d} \right) [v_d^*(k) - v_d(k)] \right]^2 + \left[\left(\frac{T_s}{L_q} \right) [v_q^*(k) - v_q(k)] \right]^2 \quad (21)$$

$$g = \left(\frac{T_s}{L_d} \right)^2 [v_d^*(k) - v_d(k)]^2 + \left(\frac{T_s}{L_q} \right)^2 [v_q^*(k) - v_q(k)]^2 \quad (22)$$

In case of a round rotor, the $L_d = L_q$, so the cost function can be written as

$$g = F * \left[[v_d^*(k) - v_d(k)]^2 + [v_q^*(k) - v_q(k)]^2 \right] \quad (23)$$

where $F = \left(\frac{T_s}{L_d}\right)^2$

We could eliminate F to obtain the final cost function since minimizing a function scaled by a constant that is positive is the same as minimizing the function itself:

$$g = [v_d^*(k) - v_d(k)]^2 + [v_q^*(k) - v_q(k)]^2. \quad (24)$$

Figure 7a illustrates how this cost function depicts the distance between the reference voltage vector $[v_d^* \ v_q^*]$ and each candidate voltage vector $[v_d \ v_q]$. As a result, the cost is reduced, which is the same as reducing the separation between these two vectors (Figure 7b). Using this geometric interpretation, we may identify places in the $\alpha\beta$ -plane where each voltage vector is nearest to the reference vector, as a result, the vector space is divided into regions as shown in Figure 7c.

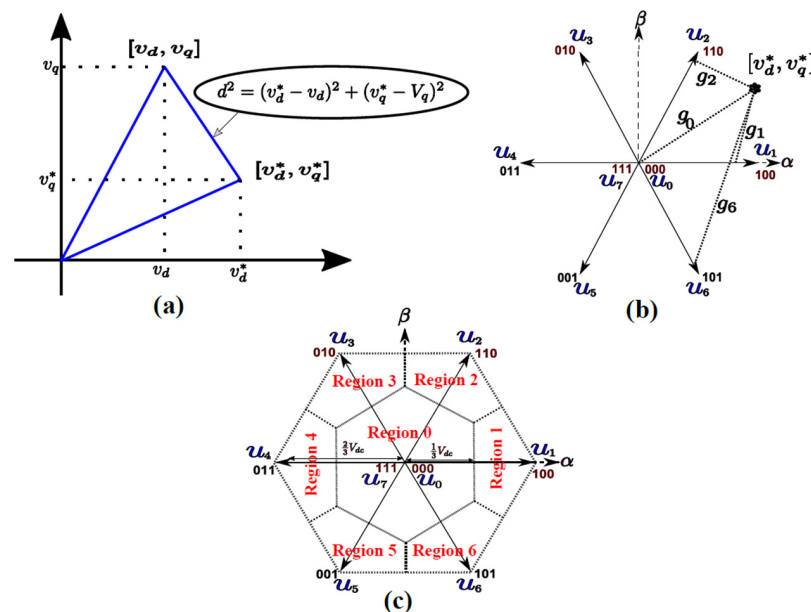


Figure 7. (a,b) The geometric interpretation of the cost function for 2-level inverters; (c) the space vector diagram rejoins for the proposed MPC_{II} algorithm.

Two concentric hexagons having the same center are depicted in Figure 7c. The active voltage vectors are represented by the outside hexagon. The inner hexagon is arranged so that each of its sides is perpendicular to a state vector drawn from the shared center, having just half the side length. The inner hexagon's boundary can be written as

$$|x| = \frac{v_{dc}}{3} \text{ and } \sqrt{3} |y| + |x| = \frac{2v_{dc}}{3} \quad (25)$$

For more illustration, the zero vector is the nearest if the reference is in the middle hexagon; voltage vector 1 is the best if it is in region 1, and so on. The best switching state could be directly chosen without computing all intermediate equations thanks to regions, which are illustrated in Figure 7c.

The proposed MPC_{II} method starts with the same stages as the MPC_I algorithm: Equations (14) and (15) are used to generate the reference voltage vector, and Equation (16)

is used to convert it into the $\alpha\beta$ stationary frame. In contrast to MPC_I, the angle is not calculated instantly. Rather, it first determines if the reference vector is inside the middle hexagon. If so, it is categorized as region 0, and the zero vector is the optimal switching state. If the vector is located outside the hexagon, the angle is determined using Equation (17). As shown in Figure 7c. and Table 4, the appropriate region is identified based on this angle. For example, the best switching state is state 1 if the vector falls inside region 1 and the angle is between -30 and 30 .

Table 4. Angle range and its corresponding candidate vectors for Algorithm 4 for 2-level inverters.

Selection Criteria		Region	Optimal Vectors (State)
Inside the middle hexagon	-	0	0
Outside the middle hexagon	$330^\circ \leq \alpha < 30^\circ$	1	1
Outside the middle hexagon	$30^\circ \leq \alpha < 90^\circ$	2	2
Outside the middle hexagon	$90^\circ \leq \alpha < 150^\circ$	3	3
Outside the middle hexagon	$150^\circ \leq \alpha < 210^\circ$	4	4
Outside the middle hexagon	$210^\circ \leq \alpha < 270^\circ$	5	5
Outside the middle hexagon	$270^\circ \leq \alpha < 330^\circ$	6	6

For checking whether a vector lies inside the hexagon, this requires using the built-in function (inpolygon) or applying a simple if statement consisting of two inequalities:

$$|v_\alpha^*| \leq \frac{v_{dc}}{3} \text{ and } \sqrt{3} |v_\beta^*| + |v_\alpha^*| \leq \frac{2v_{dc}}{3} \quad (26)$$

If both inequalities are true, the reference vector $v_{\alpha\beta}^*$ lies inside the inner hexagon. This is computationally efficient due to avoiding the additional transformations and cost function calculations required when converting future currents into the dq-frame for each vector.

Algorithm 4: Algorithm of the Proposed MPC_{II}

Step 1: Read $i_q, i_d, i_{q_ref}, i_{d_ref}$, and ω_r .

Step 2: Calculate the required inverter voltage v_{dq}^* using Equations (14) and (15).

Step 3: Calculate the voltage in $\alpha\beta$ stationary frame $v_{\alpha\beta}^*$ using Equation (16).

Step 4: Determine whether $v_{\alpha\beta}^*$ is inside the central hexagon using the geometric inclusion check (e.g., inpolygon in MATLAB)

If inpolygon = 1

 Return the zero-vector state as the optimal switching state.

else

 Determine the angle of the $v_{\alpha\beta}^*$ voltage using Equation (17).

 Based on the value of $\alpha(k)$, determine the optimal switching state according to the following angle intervals:

 end

For a two-level inverter, the algorithm requires approximately five or seven equation evaluations per sampling interval. Notably, for higher-level inverters, the number of equation evaluations remains constant at five or seven; only the conditional function—or equivalent equation—changes. The final block diagram of the proposed MPC_{II} is displayed in Figure 8.

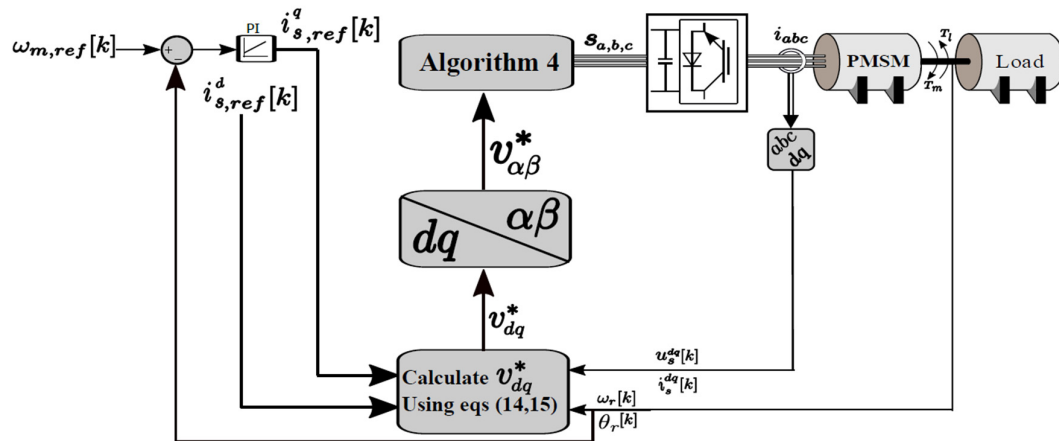


Figure 8. The proposed MPC_{II} block diagram for controlling PMSM.

As illustrated in Algorithm 4, perform several computations for each sampling interval. Specifically, the controller must achieve the following:

- Calculate v_{dq}^* then $v_{\alpha\beta}^*$ via Equations (14)–(16). [executed once]
- Determine whether $v_{\alpha\beta}^*$ inside the hexagon. [executed once]
- Use a conditional statement (or equivalent expression) to determine inclusion. [executed once]
- If it lies outside the hexagon, compute the angle $\alpha(k)$ using Equation (17). [conditionally executed]
- Obtain the region and optimal state via conditional logic or an expression. [conditionally executed]

5. The Differences Between the Algorithms

The fundamental difference between the four MPC algorithms is their computational expense as well as execution effectiveness:

- Conventional MPC_I (Algorithm 1): This algorithm evaluates every potential switching state (seven for a two-level inverter), which makes real-time implementation difficult due to its large computing burden (~35 equations each period).
- Conventional MPC_{II} (Algorithm 2): This algorithm improves efficiency while preserving high accuracy by choosing three candidate vectors based on sector identification (around 20 equations each period).
- Proposed MPC_I (Algorithm 3): By redefining sectors (around 15 equations each period), it further restricts assessments to two candidate vectors, improving computing efficiency.
- The most efficient method for real-time systems is the proposed MPC_{II} (Algorithm 4), which reduces calculations to 5–7 equations per period by using a geometric integration test to either directly choose the zero state or identify an optimal state depending on the angle.

Figure 9 illustrates the differences in the number of candidates switching states across the algorithms, while Table 5 provides a general comparison among them.

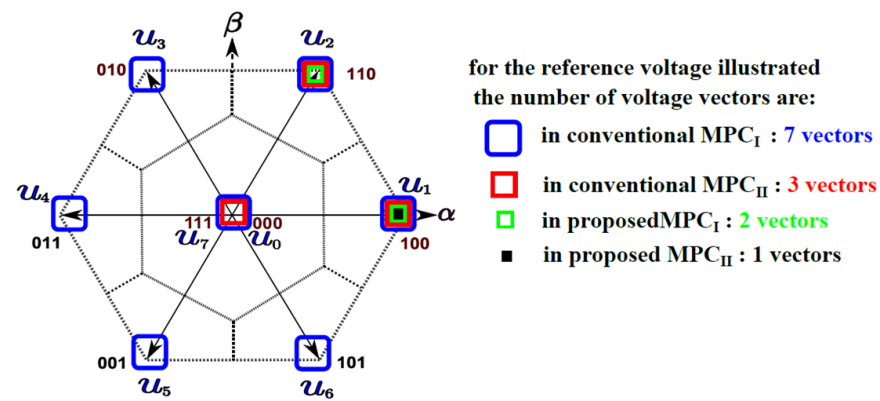


Figure 9. Comparison between number of candidate vectors for each algorithm for the 2-level inverter.

Table 5. Comprehensive comparison between the computational burden for each algorithm.

Feature	Convention MPC _I Algorithm 1	Convention MPC _{II} Algorithm 2	Proposed MPC _I Algorithm 3	Proposed MPC _{II} Algorithm 4
Full Vector Evaluation	yes	no	no	no
Number of candidates vectors	Increase exponentially	constant	constant	constant
for 2-level inverter	7	3	2	1
for 3-level inverter	19	3	2	1
for 5-level inverter	61	3	2	1
Calculation of vector voltages in dq, predicted currents and the cost function	yes	yes	yes	no
Calculation of angle of the reference voltage	no	yes	yes	conditionally
Conditional logic	No	yes	yes	yes
Approximated number of equations per cycle	Increase exponentially	Constant	constant	constant
for 2-level inverter	$5 \times 7 = 35$	20	15	5–7
for 3-level inverter	$5 \times 19 = 95$	20	15	5–7
for 5-level inverter	$5 \times 61 = 305$	20	15	5–7

6. Results and Discussion

A comprehensive analysis of the proposed model predictive control (MPC) techniques for a non-salient pole permanent magnet synchronous motor (PMSM) is provided in this section with parameters as Table 6. Three primary components comprise the results:

- (a) Analysis of the proposed MPC_I algorithm under various operating conditions.
- (b) Analysis of the proposed MPC_{II} algorithm under various operating conditions.
- (c) Comparative analysis between the proposed and conventional approaches.

Table 6. PMSM parameters.

Parameter	Nomenclature/Unit	Value
flux linkage	λ_m	0.41
Nominal d-axis inductance	L_d [H]	0.01
Nominal q-axis inductance	L_q [H]	0.01
Nominal Stator resistance	R_s [Ω]	1.3
Number of Pole pairs	P	3
Rotor inertia	J [kg.m ²]	0.0012

6.1. Analysis of the Proposed MPC_I Algorithm Under Various Operating Conditions

Three popular operating situations are used to assess the effectiveness of the first proposed MPC algorithm: (a) step-up in speed, (b) speed step-down to standstill, and (c) torque variation at constant speed. The simulation results are shown in Figure 10a–c, which illustrates the system's response to torque, rotor speed, dq -axis currents, and three-phase stator currents.

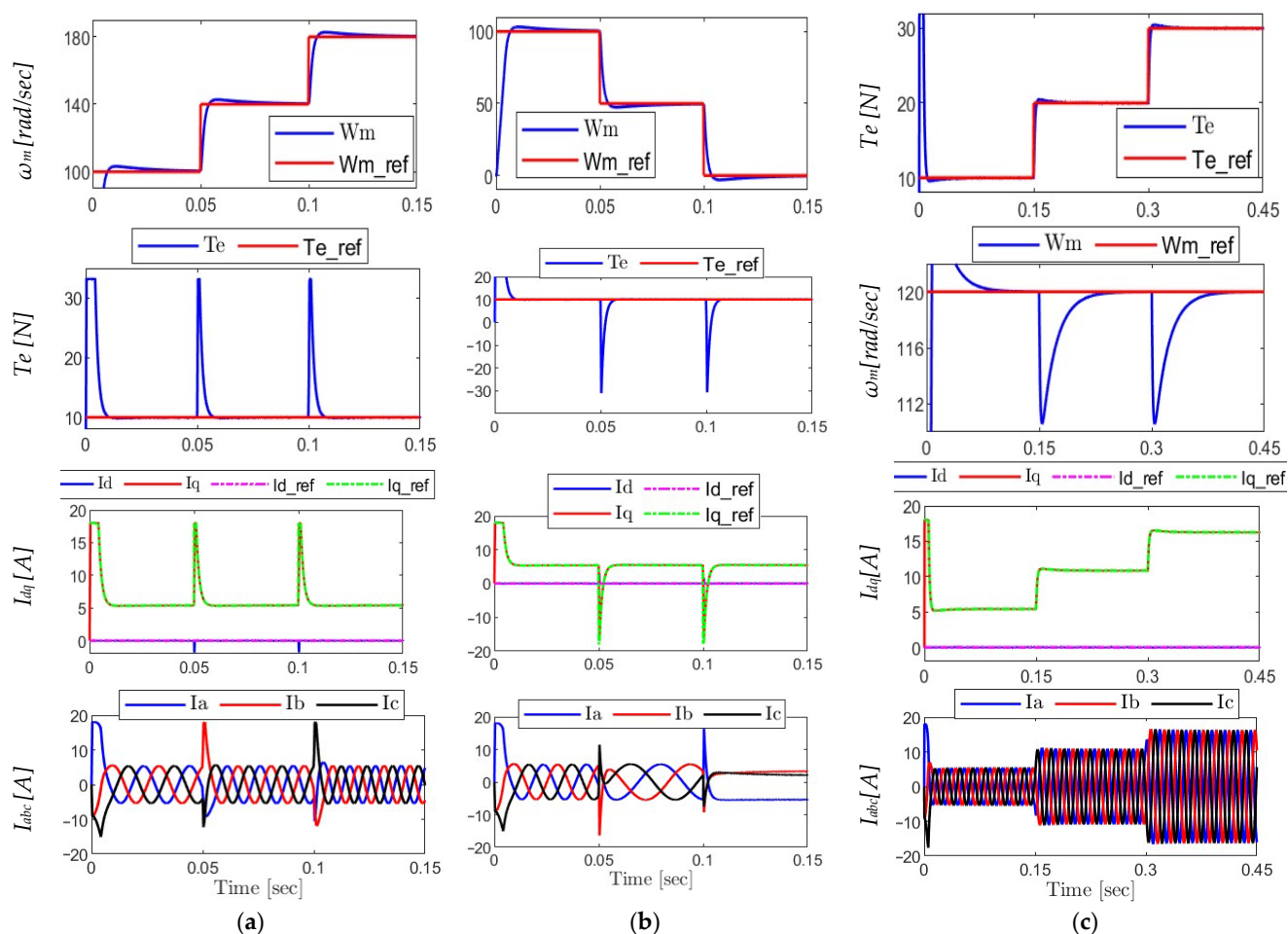


Figure 10. Analysis of the proposed MPC_I algorithm under various operating conditions: (a) step-up in speed; (b) speed step-down to standstill (c) torque variation at constant speed.

(a) Step-Up in Speed

- The speed reference is raised by two steps for the motor.
- There are minor overshoot and steady-state errors, demonstrating fast dynamic behavior, as actual rotor speed ω_m rapidly follows the reference $\omega_{m,ref}$.

- The electromagnetic torque T_e sudden transition in reaction to changes in speed demonstrates the responsiveness of the control system.
 - The d-axis current i_d is roughly equal to zero, demonstrating adequate PMSM operation, whereas the q-axis current i_q precisely matches the torque profile.
 - The sinusoidal and balanced i_a , i_b , and i_c stator currents show proper inverter switching.
- (b) Speed Step-Down to Standstill
- To simulate slowing down to a stop, the speed reference is gradually reduced until it reaches zero.
 - Throughout all phases, the rotor speed ω_m keeps tracking the reference precisely and smoothly.
 - During braking, there occurs a momentary reversal in torque T_e , which is accurately reflected in the i_q waveform.
 - The i_d component stays close to zero, guaranteeing appropriate flux direction.
 - The stator currents maintain their balance and sinusoidal during transitional phases, confirming stability during deceleration.
- (c) Torque Variation at Constant Speed
- The torque reference $T_{e,ref}$ is changed in three separate phases while keeping the speed constant.
 - Each torque adjustment is followed by a quick and steady reaction in T_e with little settling time
 - There is no change in the rotor speed ω_m , indicating resilience and efficient decoupling.
 - While i_d is relatively constant, the torque change is accurately reflected by the i_q current.
 - Although the waveforms maintain their sinusoidal shape, the amplitude of stator currents varies with torque levels.
 - Overall, the findings show that under a variety of operating situations, the first proposed MPC algorithm provides high dynamic stability, precise torque and speed tracking, and outstanding current quality.

6.2. Analysis of the Proposed MPC_{II} Algorithm Under Various Operating Conditions

As demonstrated in Figure 11a–c, the second proposed MPC technique was tested under the same dynamic operating conditions as the first: torque changes at constant speed, speed step-up, and speed step-down to standstill. The findings acquired show that the behavior of the drive system is the same as that obtained with the first proposed approach. In particular,

- The rotor speed closely matches its reference
- When the load varies, the electromagnetic torque reacts rapidly.
- The three-phase stator currents stay balanced and sinusoidal during all transitions.
- The dq -axis currents preserve predicted profiles, with i_d staying near zero and i_q exactly matching torque demands.

These outcomes validate the efficiency of the second proposed approach in high-performance PMSM drive applications by confirming that they guarantee the same degree of dynamic performance and control precision as the first.

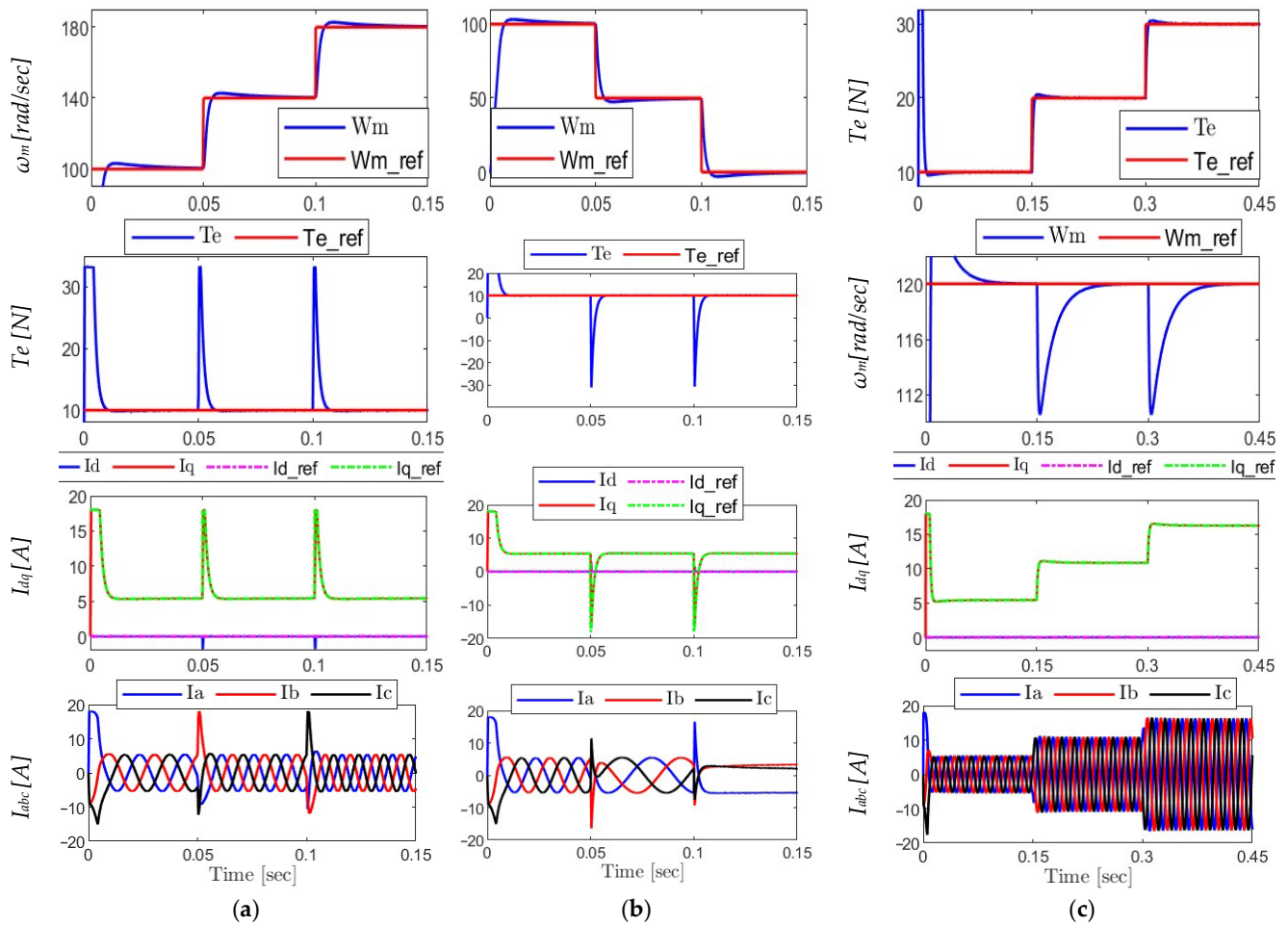


Figure 11. Analysis of the proposed MPC_{II} algorithm under various operating conditions: (a) step-up in speed; (b) speed step-down to standstill; (c) torque variation at constant speed.

6.3. Comparative Analysis Between the Proposed and Conventional Approaches

In order to evaluate the controllers' performance, they are put through three distinct dynamic scenarios: a step-up in speed, a speed step-down to standstill, a torque variation at constant speed, and a comparison of the computational times for each.

Plots demonstrating the rotor speed, electromagnetic torque, three-phase currents, dq -axis currents, and absolute differences between the currents derived from full-state and reduced-state evaluations are shown for each example. By demonstrating the proposed MPCs' capacity for real-time implementation due to their lower computing load, these visual comparisons seek to confirm that they are effective in terms of dynamic responsiveness, waveform quality, and control accuracy.

6.3.1. Step Increase in Reference Speed

Figure 12 assesses how well the controllers work when the reference speed is increased step-by-step.

- Rotor Speed (ω_m) and Torque (T_e): All controllers exhibit efficient speed control by precisely tracking the reference speed and torque with little overshoot and quick reaction.
- dq Currents (i_d, i_q): All controllers exhibit well-regulated current actions that have a relationship with torque demand.
- Three-Phase Currents (i_{abc}): The sinusoidal and balanced waveforms show that the inverter is operating and regulating current properly.

- The speed tracking root mean square error (RMSE) was identical for all algorithms; its value was 6.37 for the entire simulation period.
- The d- and q-axis current tracking errors (RMSEs) were the same among the algorithms; the values were 0.086 and 0.706, respectively, for the full simulation time.
- The torque ripple was calculated during the steady-state period (0.065–0.1 s); it was the same for all algorithms, with 3.72%.
- Current Difference (Row 5): Although certain approaches use fewer states, differences between full-state and reduced-state (one-, two-, and three-state) assessments are none, demonstrating that all algorithms provide the same three-phase current outputs. Consequently, the total harmonic distortion (THD) is also the same for all of them validating the reduced-complexity model.

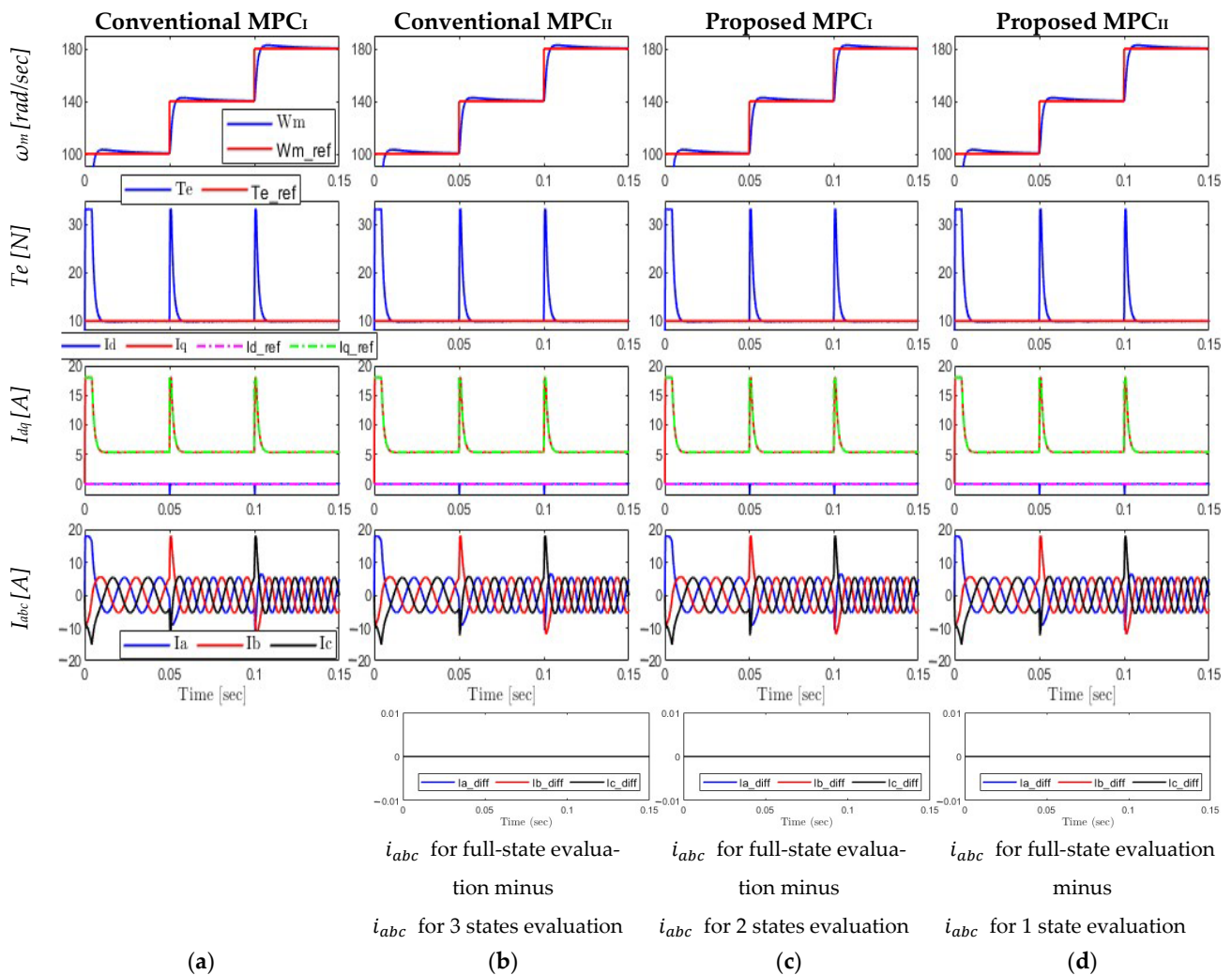


Figure 12. Comparative analysis between all approaches under step increase in reference speed: (a) conventional MPC_I; (b) conventional MPC_{II}; (c) proposed MPC_I; (d) proposed MPC_{II}.

6.3.2. Step Decrease in Reference Speed

The scenario shown in Figure 13 investigates the reaction to a step decrease in speed.

- Rotor Speed and Torque: All controllers once more demonstrate accurate tracking, rapid deceleration, and steady transitions, demonstrating resilience in decelerating circumstances.
- Currents: With equally clear and steady profiles, the *dq* and *abc* currents behave similarly to the previous case.

- Speed tracking (RMSE) for all algorithms was equal to 6.68 for full interval.
- The d- and q-axis current (RMSE) were again identical for algorithms. It is measured at 0.112 and 1.081, respectively.
- The torque ripple is evaluated in the steady-state region (0.065–0.1 s) and it was uniform for the algorithms, with a value of 3.74%.
- Current Difference: Once more, the current difference graphs (bottom row) demonstrate the computational effectiveness of the reduced-state algorithms without compromising performance, which means that the total harmonic distortion (THD) is identical for all methods, confirming that all reduced-state assessments match the full-state findings.

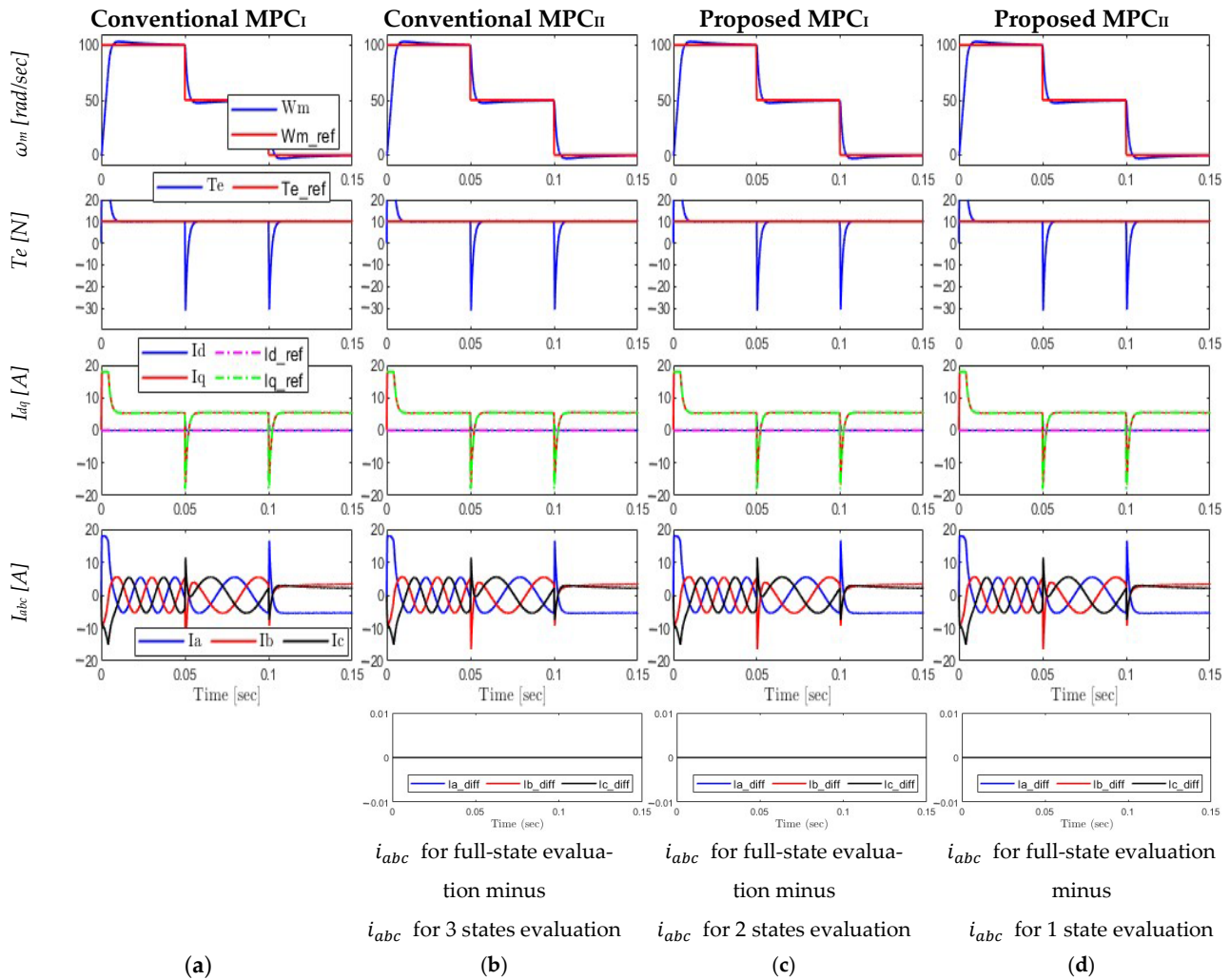


Figure 13. Comparative analysis between all approaches under step decrease in reference speed: (a) conventional MPC_I; (b) conventional MPC_{II}; (c) proposed MPC_I; (d) proposed MPC_{II}.

6.3.3. Torque Variation at Constant Speed

Now the torque responds to step changes while the speed is constant, as in Figure 14.

- Speed and Torque: The controllers exhibit decoupling control efficacy by maintaining a steady speed even when torque steps are applied. The torque response stays precise and follows the standard.
- Currents: abc currents maintain high-quality waveforms, but dq currents adapt suitably to torque demands.
- The speed tracking (RMSE) was again uniform for all algorithms; it was 5.019.

-

6.3.4. Comparing the Computational Time

The following are the total execution times needed to complete 1,800,211 control loop iterations for each method, as seen in Figure 15:

- Conventional MPC_I: 1.867 s \rightarrow (baseline)
- Conventional MPC_{II}: 1.593 s \rightarrow 14.68% improvement

- Proposed MPC_I: 1.417 s → 24.10% improvement
- Proposed MPC_{II}: 1.127 s → 39.64% improvement





Path	Time Plot (Dark Band = Self Time)	Total Time (s)	Self Time (s)	Number of Calls
MPC_Full_States		1.867	1.867	1,800,211
(a)				
Path	Time Plot (Dark Band = Self Time)	Total Time (s)	Self Time (s)	Number of Calls
MPC_3_States		1.593	1.593	1,800,211
(b)				
Path	Time Plot (Dark Band = Self Time)	Total Time (s)	Self Time (s)	Number of Calls
MPC_2_States		1.417	1.417	1,800,211
(c)				
Path	Time Plot (Dark Band = Self Time)	Total Time (s)	Self Time (s)	Number of Calls
MPC_1_State		1.127	1.127	1,800,211
(d)				

Figure 15. Comparative analysis between all approach computational times: (a) conventional MPC_I; (b) conventional MPC_{II}; (c) proposed MPC_I; (d) proposed MPC_{II}.

This graph illustrates how the number of switching states assessed by the proposed approaches has decreased. With a 39.64% enhancement over the baseline conventional MPC_I, the proposed MPC_{II} exhibits the lowest computational burden. For real-time applications, where less computing work immediately improves hardware simplicity, energy economy, and system responsiveness, this enhancement is especially beneficial.

7. Conclusions

This paper investigates four finite control set model predictive control (FCS-MPC) approaches for a two-level inverter-fed PMSM drive. Two of these methods are newly developed and presented in this paper, while the other two are derived from determined conventional approaches. The proposed techniques decrease the selection process among space vectors from seven candidates to only two or even a single vector. According to simulation findings, the four assessed MPC algorithms show comparable control accuracy in all dynamic conditions, including torque variation at constant speed, speed raise, and speed decrease. This features excellent waveforms, efficient current control, and accurate tracking of speed and torque. Interestingly, the phase current variations between the full-state and reduced-state implementations are represented in the simulation results, which are essentially null (equal zero). This shows that even though they evaluate a lot fewer switching states, the reduced-state predictive controllers perfectly mimic the behavior of their full-state counterparts.

In addition to control performance, each technique's computational effectiveness was evaluated quantitatively across 1,800,211 cycles of the control loop. Conventional MPC_I, conventional MPC_{II}, proposed MPC_I, and proposed MPC_{II} all had times of execution of 1.867 s, 1.593 s, 1.417 s, and 1.127 s, respectively. These translate into decreases of around 14.65%, 24.10%, and 39.65% in computing time compared with the baseline conventional MPC_I.

These results highlight how beneficial the proposed MPC techniques are for applications that operate in real time. The reduced-state techniques, especially the MPC_{II}, provide useful benefits for implementation in real-time control hardware, where comput-

ing economy is crucial, by significantly reducing computational needs while maintaining control quality.

Author Contributions: M.S. and A.F. conceived, designed, and implemented the control strategies and wrote the manuscript; K.B.T. and A.S.M. were responsible for the guidance and number of key suggestions. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data is contained within the article. The original contributions presented in this study are included in the article. Further inquiries can be directed at the corresponding authors.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Adel, M.M.; Ahmed, W.A.; Taha, M.; Saleh, A.A. Enhanced Sensorless Field Oriented Controlled PMSM Drive Using Fractional Calculus and PSO Technique. In Proceedings of the 2020 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), Sorrento, Italy, 24–26 June 2020; pp. 5–10. [CrossRef]
2. El Maguid Ahmed, W.A.; Adel, M.M.; Taha, M.; Saleh, A.A. PSO technique applied to sensorless field-oriented control PMSM drive with discretized RL-fractional integral. *Alex. Eng. J.* **2021**, *60*, 4029–4040. [CrossRef]
3. FPGA-Based High Performance Sensorless Control for PMSM Drives (Berichte aus der Elektrotechnik)—Ma, Zhixun: 9783844028553—AbeBooks. Available online: <https://www.abebooks.com/9783844028553/FPGA-Based-High-Performance-Sensorless-Control-3844028552/plp> (accessed on 5 September 2025).
4. Maji, P.; Kpanda, G.; Ksaha, P. Field oriented control of permanent magnet synchronous motor using PID controller. *Adv. Res. Electr. Electron. Instrum. Eng.* **2015**, *4*, 632–639.
5. Tawfiq, K.B.; Mansour, A.S.; Sergeant, P. Mathematical Design and Analysis of Three-Phase Inverters: Different Wide Bandgap Semiconductor Technologies and DC-Link Capacitor Selection. *Mathematics* **2023**, *11*, 2137. [CrossRef]
6. Zhang, Y.; Zhu, J. Direct Torque Control of Permanent Magnet Synchronous Motor with Reduced Torque Ripple and Commutation Frequency. *IEEE Trans. Power Electron.* **2011**, *26*, 235–248. [CrossRef]
7. Lin, X.; Huang, W.; Jiang, W.; Zhao, Y.; Zhu, S. A Stator Flux Observer With Phase Self-Tuning for Direct Torque Control of Permanent Magnet Synchronous Motor. *IEEE Trans. Power Electron.* **2020**, *35*, 6140–6152. [CrossRef]
8. Foo, G.H.B.; Zhang, X. Constant Switching Frequency Based Direct Torque Control of Interior Permanent Magnet Synchronous Motors With Reduced Ripples and Fast Torque Dynamics. *IEEE Trans. Power Electron.* **2016**, *31*, 6485–6493. [CrossRef]
9. Farhan, A.; Abdelrahman, M.; Saleh, A.; Shaltout, A.; Kennel, R. Simplified Sensorless Current Predictive Control of Synchronous Reluctance Motor Using Online Parameter Estimation. *Energies* **2020**, *13*, 492. [CrossRef]
10. Trubenbach, R.A.; Mackay, A.T.; Kamper, M.J. Performance of a reluctance synchronous machine under vector control. In Proceedings of the IEEE Power Electronics Specialist Conference—PESC '93, Seattle, WA, USA, 20–24 June 1993; pp. 803–808. [CrossRef]
11. Siami, M.; Khaburi, D.A.; Rodriguez, J. Simplified finite control set-model predictive control for matrix converter-fed PMSM drives. *IEEE Trans. Power Electron.* **2017**, *33*, 2438–2446. [CrossRef]
12. Abdelrahman, M.; Hackl, C.M.; Kahia, B.; Kennel, R. Predictive Direct Torque Control Strategy for Surface-Mounted Permanent-Magnet Synchronous Generators. In Proceedings of the NEIS 2017; Conference on Sustainable Energy Supply and Energy Storage Systems, Hamburg, Germany, 21–22 September 2017; pp. 1–6. Available online: <https://ieeexplore.ieee.org/document/8421831> (accessed on 6 September 2025).
13. Geyer, T.; Quevedo, D.E. Multistep finite control set model predictive control for power electronics. *IEEE Trans. Power Electron.* **2014**, *29*, 6836–6846. [CrossRef]
14. Zafra, E.; Vazquez, S.; Geyer, T.; Aguilera, R.P.; Freire, E.; Franquelo, L.G. Computational analysis of the long horizon FCS-MPC problem for power converters. *IEEE Trans. Power Electron.* **2024**, *39*, 12762–12773. [CrossRef]
15. Iqbal, A.; Abu-Rub, H.; Ahmed, S.K.M.; Cortes, P.; Rodriguez, J. Model predictive current control of a three-level five-phase NPC VSI using simplified computational approach. In Proceedings of the 2014 IEEE Applied Power Electronics Conference and Exposition—APEC 2014, Fort Worth, TX, USA, 16–20 March 2014; pp. 2323–2330.
16. Hu, J.; Zhu, J.; Lei, G.; Platt, G.; Dorrell, D.G. Multi-objective model-predictive control for high-power converters. *IEEE Trans. Energy Convers.* **2013**, *28*, 652–663.
17. Cortés, P.; Wilson, A.; Kouro, S.; Rodriguez, J.; Abu-Rub, H. Model predictive control of multilevel cascaded H-bridge inverters. *IEEE Trans. Ind. Electron.* **2010**, *57*, 2691–2699. [CrossRef]

18. Habibullah, M.; Lu, D.D.-C.; Xiao, D.; Rahman, M.F. A simplified finite-state predictive direct torque control for induction motor drive. *IEEE Trans. Ind. Electron.* **2016**, *63*, 3964–3975. [\[CrossRef\]](#)
19. Siami, M.; Khaburi, D.A.; Rivera, M.; Rodriguez, J. A computationally efficient lookup table based FCS-MPC for PMSM drives fed by matrix converters. *IEEE Trans. Ind. Electron.* **2017**, *64*, 7645–7654. [\[CrossRef\]](#)
20. Mohsen, S.; Amiri, M.; Savadkoohi, H.K.; Rezavandi, R.; Valipour, S. Simplified predictive torque control for a PMSM drive fed by a matrix converter with imposed input current. *IEEE J. Emerg. Sel. Top. Power Electron.* **2018**, *6*, 1641–1649. [\[CrossRef\]](#)
21. Zhang, Y.; Xie, W.; Li, Z.; Zhang, Y. Low-complexity model predictive power control: Double-vector-based approach. *IEEE Trans. Ind. Electron.* **2014**, *61*, 5871–5880. [\[CrossRef\]](#)
22. Ji, D.; Ren, J.; Liu, C.; Shi, Y. Stabilizing terminal constraint-free nonlinear MPC via sliding mode-based terminal cost. *Automatica* **2021**, *134*, 109898. [\[CrossRef\]](#)
23. Hadla, H.; Cruz, S. Predictive Stator Flux and Load Angle Control of Synchronous Reluctance Motor Drives Operating in a Wide Speed Range. *IEEE Trans. Ind. Electron.* **2017**, *64*, 6950–6959. [\[CrossRef\]](#)
24. Zhang, X.; Foo, G.H.B.; Vilathgamuwa, D.M.; Maskell, D.L. An Improved Robust Field-Weakening Algorithm for Direct-Torque-Controlled Synchronous-Reluctance-Motor Drives. *IEEE Trans. Ind. Electron.* **2015**, *62*, 3255–3264. [\[CrossRef\]](#)
25. Tang, L.; Zhong, L.; Rahman, M.F.; Hu, Y. A novel direct torque controlled interior permanent magnet synchronous machine drive with low ripple in flux and torque and fixed switching frequency. *IEEE Trans. Power Electron.* **2004**, *19*, 346–354. [\[CrossRef\]](#)
26. Xia, C.; Wang, Y.; Shi, T. Implementation of Finite-State Model Predictive Control for Commutation Torque Ripple Minimization of Permanent-Magnet Brushless DC Motor. *IEEE Trans. Ind. Electron.* **2013**, *60*, 896–905. [\[CrossRef\]](#)
27. Carlet, P.G.; Tinazzi, F.; Bolognani, S.; Zigliotto, M. An Effective Model-Free Predictive Current Control for Synchronous Reluctance Motor Drives. *IEEE Trans. Ind. Appl.* **2019**, *55*, 3781–3790. [\[CrossRef\]](#)
28. Abdelrahem, M.; Hackl, C.; Kennel, R. Simplified model predictive current control without mechanical sensors for variable-speed wind energy conversion systems. *Electr. Eng.* **2017**, *99*, 367–377. [\[CrossRef\]](#)
29. Farhan, A.; Saleh, A.; Shaltout, A. High performance Reluctance Synchronous Motor drive using Field Oriented Control. In Proceedings of the 2013 5th International Conference on Modelling, Identification and Control (ICMIC), Cairo, Egypt, 31 August–2 September 2013; pp. 181–186. Available online: <https://ieeexplore.ieee.org/document/6642215> (accessed on 6 September 2025).
30. Consoli, A.; Russo, F.; Scarcella, G.; Testa, A. Low and zero speed sensorless control of synchronous reluctance motors. In Proceedings of the Conference Record of 1998 IEEE Industry Applications Conference. Thirty-Third IAS Annual Meeting (Cat. No.98CH36242), St. Louis, MO, USA, 12–15 October 1998; Volume 1, pp. 685–692. [\[CrossRef\]](#)
31. Zhang, Y.; Lin, H. Simplified model predictive current control method of voltage-source inverter. In Proceedings of the 8th International Conference on Power Electronics—ECCE Asia, Jeju, Republic of Korea, 30 May–3 June 2011; pp. 1726–1733. [\[CrossRef\]](#)
32. Kouro, S.; Cortes, P.; Vargas, R.; Ammann, U.; Rodriguez, J. Model Predictive Control—A Simple and Powerful Method to Control Power Converters. *IEEE Trans. Ind. Electron.* **2009**, *56*, 1826–1838. [\[CrossRef\]](#)
33. Hong, J.; Pan, D.; Zong, Z. Comparison of the Two Current Predictive-Control Methods for a Segment-Winding Permanent-Magnet Linear Synchronous Motor. *IEEE Trans. Plasma Sci.* **2013**, *41*, 1167–1173. [\[CrossRef\]](#)
34. Gao, X.; Abdelrahem, M.; Hackl, C.M.; Zhang, Z.; Kennel, R. Direct Predictive Speed Control With a Sliding Manifold Term for PMSM Drives. *IEEE J. Emerg. Sel. Top. Power Electron.* **2020**, *8*, 1258–1267. [\[CrossRef\]](#)
35. Filho, C.J.V.; Xiao, D.; Vieira, R.P.; Emadi, A. Observers for High-Speed Sensorless PMSM Drives: Design Methods, Tuning Challenges and Future Trends. *IEEE Access* **2021**, *9*, 56397–56415. [\[CrossRef\]](#)
36. Hassan, M.A.; Adel, M.M.; Farhan, A.; Saleh, A.A. Finite speed-set model reference adaptive system based on sensorless control of permanent magnet synchronous generators for wind turbines. *Machines* **2024**, *12*, 429. [\[CrossRef\]](#)
37. Farhan, A.; Abdelrahem, M.; Hackl, C.M.; Kennel, R.; Shaltout, A.; Saleh, A. Advanced Strategy of Speed Predictive Control for Nonlinear Synchronous Reluctance Motors. *Machines* **2020**, *8*, 44. [\[CrossRef\]](#)
38. Farhan, A.; Saleh, A.; Abdelrahem, M.; Kennel, R.; Shaltout, A. High-precision sensorless predictive control of salient-pole permanent magnet synchronous motor based-on extended Kalman filter. In Proceedings of the 2019 21st International Middle East Power Systems Conference (MEPCON), Cairo, Egypt, 17–19 December 2019; pp. 226–231. Available online: <https://ieeexplore.ieee.org/abstract/document/9008188/> (accessed on 4 May 2025).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.