We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

5,600
Open access books available

137,000
International authors and editors

170M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

CLARIVATE ANALYTICS
BOOK CITATION INDEX
INDEXED

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Optimization of Model Predictive Control Weights for Control of Permanent Magnet Synchronous Motor by Using the Multi Objective Bees Algorithm

Murat Sahin

Abstract

In this study, the model predictive control (MPC) method was used within the scope of the control of the permanent magnet synchronous motor (PMSM). The strongest aspect of the MPC, the ability to control multiple components with a single function, is also one of the most difficult parts of its design. The fact that each component of the function has different effects requires assigning different weight coefficients to these components. In this study, the Bees Algorithm (BA) is used to determine the weights. Using the multi-objective function in BA, it has been tried to determine the weights that reduce the current values together with the speed error. Three different PI controllers have been designed to compare the MPC method. The coefficients of one of these are tuned with BA. Good Gain Method and Tyreus-Luyben Method were used in the other two. As a result of experimental studies, it has been observed that MPC can control PMSM more smoothly and accurately than PI controllers, with weights optimized with BA. With MPC, PMSM has been controlled with 15% settling time than other controllers and also with no overshoot.

Keywords: Model predictive control, permanent magnet synchronous motor, the Bees Algorithm, the Good Gain method, Tyreus-Luyben method

1. Introduction

Permanent Magnet Synchronous Motors (PMSM) have been used for many years due to their features such as high torque, high efficiency and fast dynamic structure. Within the scope of controlling PMSM; robust control [1], field-oriented control and direct torque control [2], fuzzy-based controllers [3, 4], sliding mode controller [5], model predictive controller (MPC) [6], and so on different control methods have been used. Especially in high speed PMSMs, driver dynamics must be controlled successfully for effective control [7]. When the applications in the literature are examined, it is seen that MPC gives successful results in this scope. Model Predictive Control (MPC), based on the optimal control theory, achieves successful results, especially in power electronics applications. MPC uses the system model equations together with the current state measurements to estimate the control

movement. With this predictive ability, it can achieve more successful results against traditional controllers [8].

Three phase inverter circuits are one of the main methods used to drive PMSM. In these circuits, there can be a limited number of switching combinations for 6 switches. This is called the Finite Control Set (FCS). FCS-MPC can be implemented even in low cost devices used today due to the optimization process performed with a limited number of iterations [9]. One of the major advantages of the FCS-MPC over other control methods is that different goals, variables, and constraints can be included in a single cost function and controlled simultaneously. [10]. Adding different categories of variables to the cost function brings great flexibility to MPC. However, the effects of these variables on the system may be different. Therefore, it is necessary to give weight coefficients for each of the variables.

Selection of weights for cost function of MPC optimization is one of the main challenge for researchers. When the MPC studies in the literature are examined, it is seen that a significant part of them are about to calculate these weights. It is seen that different methods are used in different applications. The highlights of these; empirical methods, fuzzy-based methods, evaluation algorithms, heuristic methods etc. In a sample study, a fuzzy-based calculation method was used in the PMSM current control application. Id and Iq currents were used in the cost function [11]. In another study, torque control of PMSM with MPC was performed. Torque and flux variables were used in the cost function. While weight was not used for the torque variable, a weight depending on the torque has been determined for the flux variable [12]. A different cost function that can be selected for the speed control of the PMSM may include the controller output and speed error. In order not to cause sudden effects on the system, the difference of the two control signals produced consecutively is added to the cost function [13]. A similar strategy has been used in DC motor control. With the Quadratic problem approach, along with the speed error and the difference between the two consecutive outputs are included in the cost function [14].

FCS-MPC is also used in different fields other than electric motors and similarly the weights need to be calculated. Simulated Annealing Particle Swarm Optimization with Model Predictive Control was used to control of the electric vehicles [15]. Another application that uses genetic algorithm is shunt active power filter. Link voltage, active power and reactive power are used in cost function of MPC within the scope of proper switching [16]. In a different MPC application, the multi objective genetic algorithm was used to calculate the weights [17].

In this study, the control of the PMSM used as the driving element of an actuator to be used in the aerospace area was performed with FCS-MPC. Battery life is of great importance in the aerospace area. For this reason, when creating the MPC cost function, the power variable was added along with the speed and current variables. In the PMSM control problem, apart from minimizing the speed error, minimum current consumption is also aimed. In the scope of calculating the weights, the multi objective BA is used. When the studies in the literature are analyzed, it is seen that BA gives successful results against multi objective function problems [18–20]. In order to compare the developed control system, 3 different PI controllers were designed. The second part of the study includes PMSM equations and MPC studies. The third part includes calculating weights with multi-objective BA studies. In the fourth chapter, there are studies on PI controller design. The fifth section includes experimental studies and comparisons.

2. PMSM equations and FCS-MPC design

In MPC design, first of all, it is necessary to prepare the mathematical model of the system. One of the most popular methods of controlling of PMSM with MPC is

to use machine equations in the rotor reference frame [6]. MPC is designed within the scope of speed control of PMSM. Therefore, speed and current equations were needed in the MPC cost function.

$$\frac{di_d}{dt} = -\frac{R}{L_d}i_d + \frac{L_q p}{L_d}w_r i_q + \frac{1}{L_d}v_d \tag{1}$$

$$\frac{di_q}{dt} = -\frac{R}{L_q}i_q - \frac{L_d p}{L_q}w_r i_d - \frac{\psi_m p}{L_q}w_r + \frac{1}{L_q}v_q \tag{2}$$

$$T_e = \frac{3}{2}p\left[\psi_m i_q + (L_d - L_q)i_d i_q\right] \tag{3}$$

$$\frac{dw_r}{dt} = \frac{1}{J}(T_e - T_l) - \frac{B}{J}w_r \tag{4}$$

In the currents and torque equations; ψ_m, i_q, i_d, v_q, v_d, R, L_q, L_d, w_r, and p and are the rotor magnetic flux linkage, stator currents in q and d axis, stator voltages in q and d axis, stator resistance, stator inductances in q and d axis, rotor angular speed, and pole pairs respectively. In the equation of speed; J is the inertia, T_l is the load torque, B is the viscous friction coefficient, and T_e is the electrical torque produced by the motor [11].

In the equations, resistance, inductance, pole pairs, inertia, and magnetic flux values are known. (Motor resistance and inductance can vary depending on the motor temperature [21]. This situation has been neglected in this study. It should be consideration in applications where the motor will operate under load for a long time.) In order to find Vd and Vq values, the inverter circuit must be analyzed. A typical 3 Phase Inverter circuit used in the study is given in **Figure 1**. The circuit has two driver components for each phase of the PMSM and switches for switching these components. (In this study, mosfet is used as the driver component, it is shown in green color in the figure.) In order that the components on the same phase are not switched at the same time, there is a note gate between them. Therefore, 3 switches (Switches S_a, S_b, and S_c shown in blue color in the figure.) are sufficient for switching operations. These 3 switches can be switched in 8 different combinations, each with a 0 or 1 [22]. Conversions of V_{dc} (Bus voltage) to V_d and V_q according to the switching states are found by Park-Clarke methods [23].
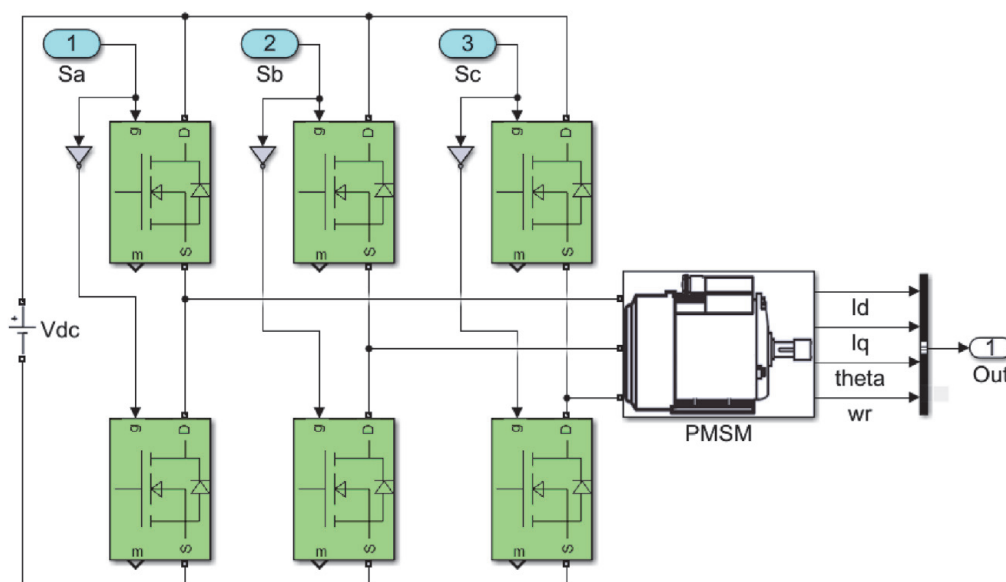

Figure 1.
PMSM drive with 3 phase inverter.

$$\begin{bmatrix} v_d \\ v_q \end{bmatrix} = \frac{2}{3} V_{dc} \begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} 1 & -1/2 & -1/2 \\ 0 & \sqrt{3}/2 & -\sqrt{3}/2 \end{bmatrix} \begin{bmatrix} S_a \\ S_b \\ S_c \end{bmatrix} \tag{5}$$

θ is the angle of rotation of the rotor in radians (In this study, it is measured by the encoder integrated into the PMSM. Also,w_r is obtained by the derivative of θ.). Other unknowns of the equations are the I_d and I_q currents. I_a, I_b, and I_c currents are measured with the current sensor. Again, Park-Clarke method is used for transition from abc phases to dq axis (In the Eq. (5), instead of switches, this time the currents are placed.).

Various discretization methods can be used to obtain a discrete-time model for calculating predictions. One of the simplest methods is the Forward Euler method, which is based on derivatives. In this method, the prediction expression is obtained by leaving the expression x(k + 1) alone. T_s is the sampling time [23].

$$\frac{dx}{dt} \approx \frac{x(k+1) - x(k)}{T_s} \tag{6}$$

When the Forward Euler approach is applied to stator currents (1) and (2) the following MPC prediction equations are obtained [11].

$$i_d(k+1) = \left(1 - \frac{RT_s}{L_d}\right)i_d(k) + \frac{L_q}{L_d}T_s p w_r i_q(k) + \frac{T_s}{L_d}v_d \tag{7}$$

$$i_q(k+1) = \left(1 - \frac{RT_s}{L_q}\right)i_q(k) - \frac{L_d}{L_q}T_s p w_r i_d(k) - \psi_m p w_r \frac{T_s}{L_q} + \frac{T_s}{L_q}v_q \tag{8}$$

In the equations, expressions with (k) show the values measured from PMSM at the previous sampling time, while the expressions with (k + 1) show the predicted values.

For each sampling time, for the 8 different Vd and Vq values given above, currents estimates will be made. Under normal conditions, what is expected from the microprocessor is to make these calculations in the period determined for the application and to generate the necessary control signal. In real application, the microprocessor also has many different tasks. Therefore, there may be delays in estimates. In this case, a dynamic system cannot be controlled successfully. Against these possible delays, it has been suggested to predict two next steps. In this study, the prediction equations for the next two steps were updated by taking this suggestion into consideration [24].

$$i_d(k+2) = \left(1 - \frac{RT_s}{L_d}\right)i_d(k+1) + \frac{L_q}{L_d}T_s p w_r i_q(k+1) + \frac{T_s}{L_d}v_d \tag{9}$$

$$i_q(k+2) = \left(1 - \frac{RT_s}{L_q}\right)i_q(k+1) - \frac{L_d}{L_q}T_s p w_r i_d(k+1) - \psi_m p w_r \frac{T_s}{L_q} + \frac{T_s}{L_q}v_q \tag{10}$$

For velocity prediction, Eq. (4) is discretized by Forward Euler method. In the equation,$i_q(k+2)$ expression is used inT_e [6].

$$w_r(k+1) = \left(1 - \frac{B}{J}\right)w_r(k)T_s + \frac{1}{J}(T_e - T_l)T_s \tag{11}$$

For velocity control in PMSM, Eqs. (9), (10) and (11) may be sufficient in the cost function. But for a more effective cost function, it has been proposed to include

the switching losses in the inverter circuit. In the sample study, the power variable was also added in this context, and the choices were made to use low power at each step of the control system [25]. In this study, the power effect (P_f) was added to the cost function. However, this effect is added as shown in (12) in a simple form so that the processing load does not increase.

$$P_f = \sqrt{(v_d * i_d(k+2))^2 + (v_q * i_q(k+2))^2} \tag{12}$$

One of the strong features of MPC is that the constraints required for the system can be defined in cost function. The combination that gives the best result among the switching alternatives may also cause high current from the PMSM. Therefore, the current constraint given in (13) has been added to the cost function [22]. When the current values are higher than a limit value, a large value is assigned to the relevant switching option to be excluded from the options. (In the part indicated by ∞ in the equation, 1e10 is used in the application.)

$$\hat{f}(i_d(k+2), i_q(k+2)) = \begin{cases} \infty \text{ if } |i_d| > i_{max} \text{ or } |i_q| > i_{max} \\ 0 \text{ if } |i_d| \leq i_{max} \text{ and } |i_q| \leq i_{max} \end{cases} \tag{13}$$

The final version of the cost function is shown below.

$$\begin{aligned} g = w_1 * (w_{ref} - w_r(k+1))^2 + w_2 * (i_d(k+2))^2 + w_3 * (i_q(k+2))^2 + w_4 * P_f{}^2 \\ + \hat{f}(i_d(k+2), i_q(k+2)) \end{aligned} \tag{14}$$
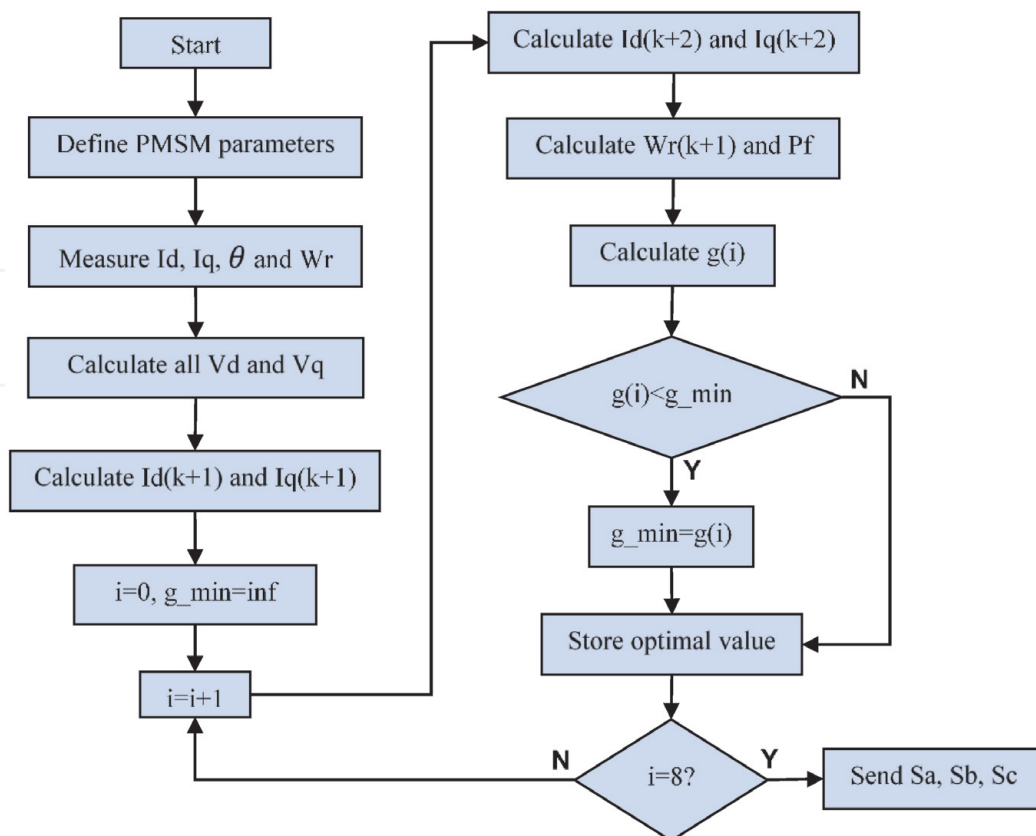

Figure 2.
Flowchart of MPC.

For each sampling time, for 8 different switching combinations, current and velocity prediction will be performed and the cost function defined in (14) will be calculated. After 8 iterations, the switching configuration that gives the minimum "g" value will be determined and sent to the inverter circuit. The flow diagram of the designed MPC is given in **Figure 2**.

3. Calculating weights with multi objective Bees Algorithm (MOBA)

BA is a population-based search algorithm. The algorithm mimics the nectar source search behavior of honey bees. Basically, it does some kind of neighbor region search along with random search and can be used for both integrated and functional optimization [26]. Detailed explanations about the algorithm were provided in Ref.s [27–29]. The pseudo code of the algorithm is given in **Figure 3**.

When designing control systems in general, the integral of the square of the error (ISE) is used as the objective function. The equation of ISE is given in (15). The equation shows the reference velocity value with r(t), the output velocity value with y(t) and the error value with e(t).

$$ISE = \int_0^\infty (r(t) - y(t))^2 dt = \int_0^\infty e(t)^2 dt \qquad (15)$$

When the error is only aimed to be minimized, it may cause high currents to be drawn from the PMSM and thus excessive energy consumption. As a solution to this issue, a multi objective (MO) optimization algorithm is suggested. The goal of MO optimization is to try to optimize all defined objective functions simultaneously. All objectives can be minimized or maximized at the same time, or some can be minimized and some maximized. General definition in the literature is given below [30].

```
1: Generate random population and evaluate fitness.
2: Sort population
3: While (stopping criterion not met)
4:      For i=1:EliteSite
5:              Recruit bees for elite site and evaluate fitness
6:              Select the best bee from each site
7:      End
8:      For i=EliteSite+1:BestSite
9:              Recruit bees for best site and evaluate fitness
10:             Select the best bee from each site
11:     End
12:     For i= BestSite+1:ScoutBee
13:             Generate random population and evaluate fitness.
14:     End
15:     Sort population
16: End While
```

Figure 3.
Pseudo code of BA.

$$Min \ (or \ Max) \left\{ f_1(x) = y_1, \ f_2(x) = y_2, \ ..., \ f_j(x) = y_j \right\} \tag{16}$$

Although there are different MO methods, the prominent method is the weighted sum method. Here, each goal has a weight coefficient. This method is also called scalarization method. In this method, basically, multiple solutions are combined into a single solution using weights [31].

$$Min(or \ Max) \sum_{j=1}^{N} w_j f_j(x) \tag{17}$$

It is necessary to limit the current for low power consumption [32]. Based on this situation, the integral of the square of the i_{bus} is included in the multi objective function (MOF) to find weights that will also minimize the i_{bus} current.

$$MOF = \int_0^\infty \left(e(t)^2 + i_{bus}(t)^2 \right) dt \tag{18}$$

The optimization process consists of two main parts. First part is Matlab M file with MOBA, second part is Simulink file with MPC, 3 Phase Inverter and PMSM models. The algorithm starts with the definition of BA parameters given in Appendix-B. Using these parameters, a random first population is created. Simulink model is run and MOF value is obtained for each member of the population. After this process is completed, the first population is sorted from small to large, according to the MOF value. Then local search section starts. In the elite and non-elite local areas, new values are generated by neighborhood search and simulations are made with these values in the Simulink model. In the global search, new sites are discovered randomly. Finally, the new population is re-sorted. These operations are repeated for all iterations and the best values are recorded when completed.

The system model used in the simulation is given in **Figure 4** and the flow chart of the optimization algorithm is given in **Figure 5**. The simulation model has been prepared in discrete-time to be close to the real application. The sampling frequency is 50KHz. (Information about the model is given in Appendix-A.) The w1, w2, w3, and w4 weights produced by the optimization algorithm are transmitted to the Simulink model and the simulation is performed. At the end of the simulation, the outputs are taken with the "Error" and "Current" blocks and sent to the MOF in the optimization algorithm.

One of the critical parameters in the optimization algorithm is determining the value ranges of weights. BA focuses on the areas with the best values with the first iterations. Therefore, even if large ranges are specified for variables, it quickly shrinks the solution set to include the parts with the best. For this reason, when
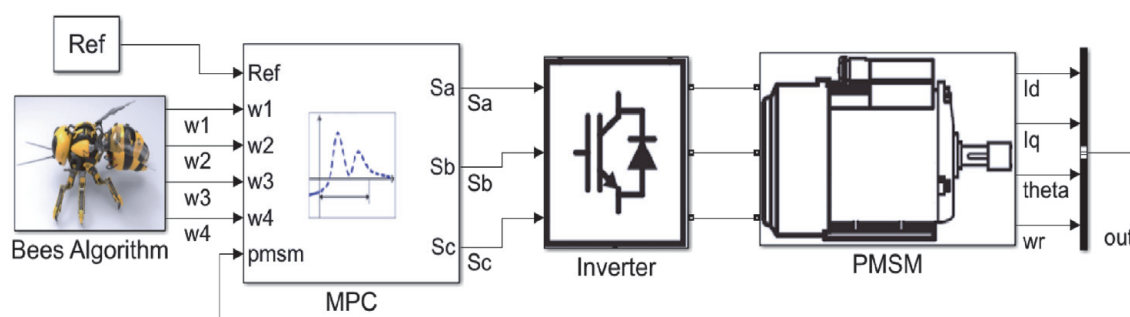

Figure 4.
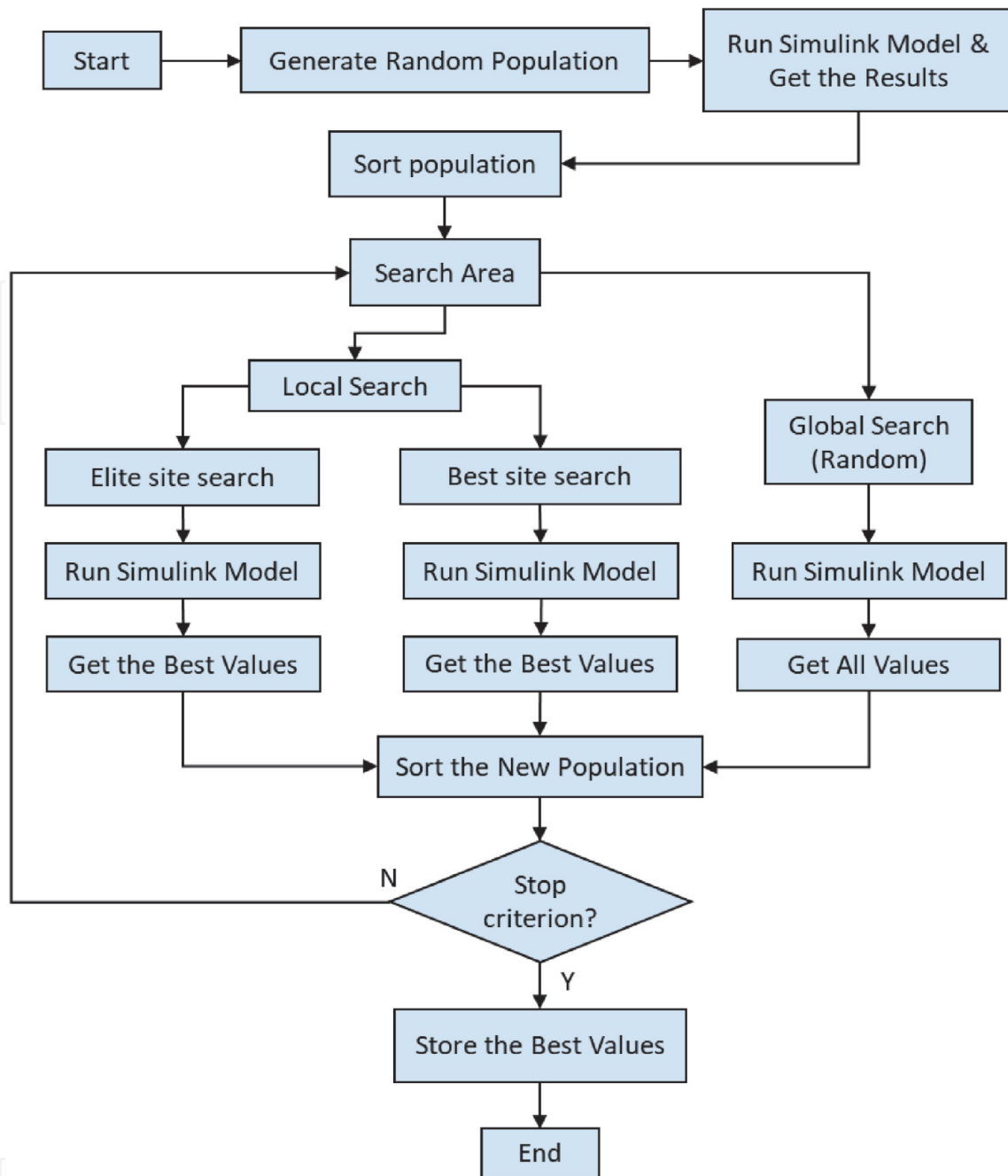*Model of PMSM control with MPC.*

Figure 5.
Flowchart of optimization algorithm.

determining the value ranges, the range 0–1000 was initially chosen to have a wide solution range. After the optimization study these values were calculated; w1 = 251.5, w2 = 6.9, w3 = 5.1, and w4 = 1.05. Consideration of these values, 300 for w1 and 10 for the others were selected and a second optimization study was carried out. The results of the both studies are given in **Figure 6**. As can be seen, in the first optimization where wide ranges are defined, there are higher error values in the first iterations. But along with other iterations, cost functions are minimized quickly. In the second optimization, since the limit values are chosen in a narrower ranges, the cost function change is also in a narrow area. (Sufficient number of searches must be made for escaping the local minimums. The important parameters in this regard are the number of foraging bees and the number of iterations. As can be seen from the figures, the minimization process has been fixed in the last iterations. This indicates that the current algorithm parameters are sufficient. If the decline continues in the last iterations, it is necessary to update the parameters.)

| It	w1	w2	w3	w4	Cost
1	938,6396	734,2122	22,7597	106,3132	20,8203
2	917,7667	665,6631	24,3344	82,1057	17,4343
3	834,7710	518,1121	20,0664	64,2322	15,5069
4	888,4555	325,1831	19,4840	45,0660	11,3746
5	714,9332	161,1120	21,3138	29,3554	10,1064
6	544,4434	104,6522	14,8776	21,8344	9,9240
7	359,9001	77,4331	15,5587	12,3356	9,3730
8	228,0098	9,9960	5,3908	2,0361	7,6221
9	233,2833	4,9776	5,5422	1,4421	7,6142
10	251,5511	6,9205	5,1322	1,0520	7,5858
11	251,5511	6,9205	5,1322	1,0520	7,5858
12	251,5511	6,9205	5,1322	1,0520	7,5858
13	251,5511	6,9205	5,1322	1,0520	7,5858
14	251,5511	6,9205	5,1322	1,0520	7,5858
15	251,5511	6,9205	5,1322	1,0520	7,5858
16	251,5511	6,9205	5,1322	1,0520	7,5858
17	251,5511	6,9205	5,1322	1,0520	7,5858
18	251,5511	6,9205	5,1322	1,0520	7,5858
19	251,5511	6,9205	5,1322	1,0520	7,5858
20	251,5511	6,9205	5,1322	1,0520	7,5858
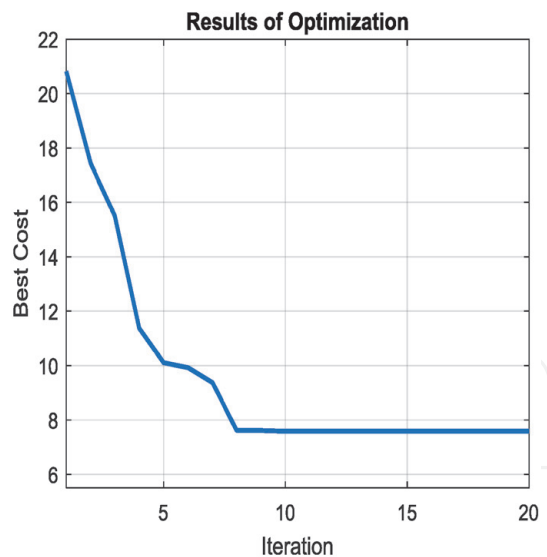
Figure 6.
Results of optimization (MPC).

4. PI controller design

One of the advantages of controlling PMSM with MPC that it does not need an external section for commutation. When controlling PMSM with PID control, it is necessary to prepare a commutation section as well. One of the methods used in this context is PWM (Pulse width modulation). The PWM signal provides the signal in a certain order with the duty cycle changes so that the DC signal becomes an AC signal. (If this signal is passed through a low-pass filter, a pure sine wave is obtained.) In SPWM (Sinusoidal Pulse Width Modulation), two signals are compared. The reference signal is sinusoidal and the carrier signal is triangular. Pulses are produced by comparing two signals, and the width of each pulse varies in proportion to the amplitude of the sine signal. The frequency of the reference signal determines the inverter output frequency and controls the reference peak amplitude, the modulation index of the output voltage, and the RMS value [33]. SPWM model used in the simulation is given in **Figure 7**.

Simulink model prepared for PI controller design is given in **Figure 8**. Id and Iq currents are used in the model as in MPC simulation. As can be seen from Eq. (3),
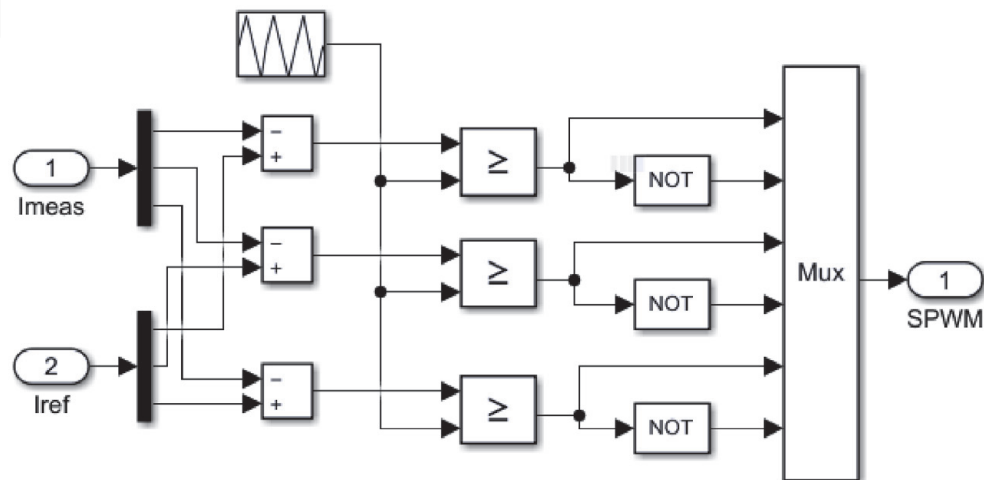

Figure 7.
SPWM model.

the PMSM speed depends on the electrical torque generated and hence the Iq and Id currents. If the inductances Lq and Ld are the same or very close, the electrical torque depends only on the Iq current. As can be seen from the motor parameters given in the Appendix-A, the Lq and Ld values are equal. Therefore, Iq current is used to perform speed control. The output of the speed controller drives the current Iq. SPWM is created for the calculated current value and PMSM is controlled through the Inverter. The sampling time and solver type of the model are the same as the MPC model.

The coefficients of the PI controller are first tuned with BA. The same structure in **Figure 5** is prepared for the PI model. Similarly, the error value is used in the objective function together with the currents. For Kp and Ki, values from 0 to 10000 were set as the limit for a wide range. BA parameters are the same as in the MPC study. **Figure 9** shows the optimization results. Costs decrease with the first iteration. The minimum value was reached with the 13th iteration. The changes in Kp and Ki are also shown in the table. The PI controller tuned with BA is given in (19).

$$G_{BEES} = 3.67 + 1601.4\frac{1}{s} \tag{19}$$

Also, two different conventional methods were used to determine the coefficients of the PI controller. The first of these is the Tyreus-Luyben method. Tyreus
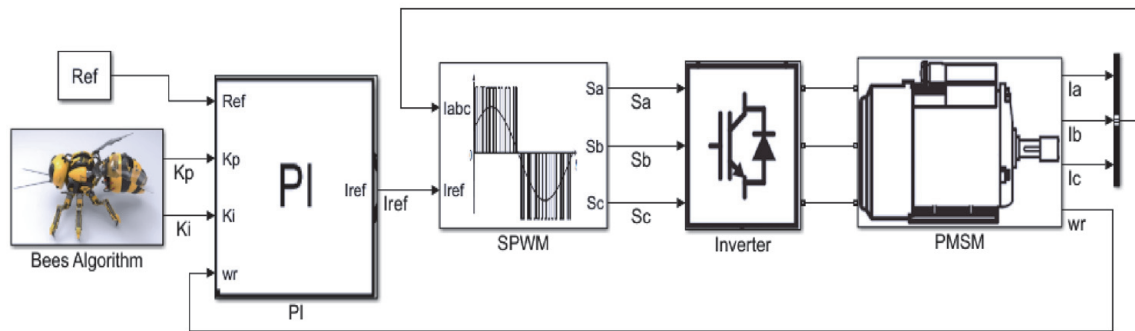

Figure 8.
Model of PMSM control with PI&SPWM.

| It	Kp	Ki	Cost
1	1830,1931	9612,1901	8,47
2	1830,1931	9612,1901	8,47
3	1830,1931	9612,1901	8,47
4	63,7641	9199,0136	8,3991
5	63,7641	9199,0136	8,3991
6	17,6877	9119,7832	8,1792
7	17,6877	9119,7832	8,1792
8	17,6877	9119,7832	8,1792
9	16,6152	8932,0877	8,1632
10	16,6152	8932,0877	8,1632
11	16,6152	8932,0877	8,1632
12	7,59767	8919,6934	8,0425
13	3,6669	1601,3678	7,9262
14	3,6669	1601,3678	7,9262
15	3,6669	1601,3678	7,9262
16	3,6669	1601,3678	7,9262
17	3,6669	1601,3678	7,9262
18	3,6669	1601,3678	7,9262
19	3,6669	1601,3678	7,9262
20	3,6669	1601,3678	7,9262
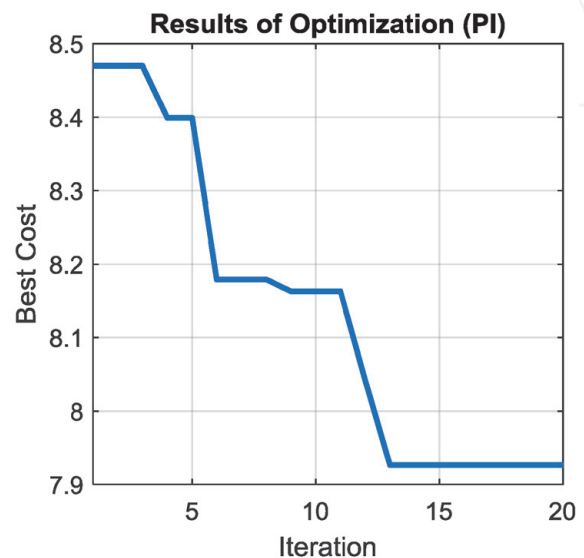
Figure 9.
Results of optimization (PI).

and Luyben's adjustment method is based on oscillations as in the Ziegler-Nichols method but has been modified for the controller parameters to achieve better stability in the control loop compared to the Ziegler-Nichols method. First, only P control is used and all gains are set to zero. The proportional gain K_p is increased until there are oscillations in the system response. K_p is increased until the oscillations are symmetrical. The final K_p value is recorded as Ku. The oscillation period of the signal is taken as Pu. K_p and T_i values are found according to the equation given in (20) [34].

$$K_p = 0.31 K_u, T_i = 2.2 T_u \tag{20}$$

When the K_p value is 27, the oscillations approached the symmetrical state (**Figure 10**). K_p and T_i values are calculated using (20).

$$K_p = 0.31 * 27 = \textbf{8.37}, T_i = 2.2 * (0.01284 - 0.0122) = \textbf{0.0014}$$

$$G_{TL} = Kp \left(1 + \frac{1}{T_{is}} \right) = 8.37 + 5944 \frac{1}{s} \tag{21}$$

As the second conventional method, The Good Gain (TGG) method was used, which is more stable than the ZN method and therefore can obtain fewer oscillation values. In the TGG method, K_i value is chosen as 0 first and K_p value is increased starting from 0. This increase is continued until the answers close to the desired reference value are obtained. As shown in the graph below, when the peak value of the system response approaches the reference value, the K_p increase is stopped and the T_{ou} value is calculated. T_{ou} value is the time between the overshoot and undershoot values in the system response. From the T_{ou} value, the T_i and K_p values are calculated as shown below [35].

$$T_i = 1.5 T_{ou}, K_p = 0.8 K_p \tag{22}$$

When the K_p value is 4, the system response approaches the reference value (**Figure 10**). K_p and T_i values are calculated using (22).

$$K_p = 4 * 0.8 = \textbf{3.2}, T_i = 1.5 * (0.00156 - 0.00116) = \textbf{0.0006}$$

$$G_{TGG} = Kp \left(1 + \frac{1}{T_{is}} \right) = 3.2 + 5333 \frac{1}{s} \tag{23}$$
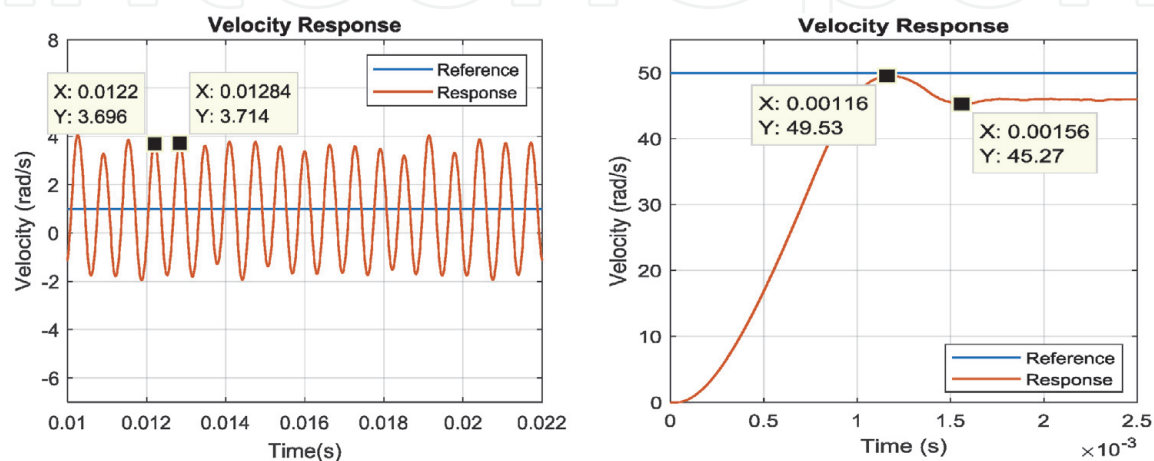

Figure 10.
Velocity response for Tyreus-Luyben method and the good gain method.

5. Experimental studies & results

To drive the PMSM, a motor driver card with a dsPIC33f model MCU is used. In addition to the MCU, the driver board includes the communication interface, the MOSFET H Bridge and drivers, and sensor reading interfaces. The driver board and other hardware used in the tests can be seen in **Figure 11**. Target Language Compiler in Simulink was used to convert the model into a machine code that dsPIC MCU can run directly. Firstly, ANSI C code was created and then machine code was generated by using the C30 C compiler provided by Microchip. Current sensors are located on the driver circuit in series with the motor phases. It is collected from each phase separately. Position data are taken from the digital encoder connected to the back of the motor shaft.

In experimental studies firstly, "100 rad/s" step command was applied to the PMSM for all controllers. All velocity results are given in **Figure 12**. and current results are given in **Figure 13**. It is seen that all four controllers are able to give enough response to the velocity command. But as can be seen, there is an overshoot (MPO) of 7% and 11.8% in PI controllers which were tuned by conventional methods. On the other hand, an overshoot of 4.2% was observed in PI tuned with BA. There is no overshoot in MPC. Because of the prediction realized by MPC using the PMSM model, controller output is produced in a more controlled manner after each step, thus creating a smooth effect on the system. In PI controllers, rise time (RT) is 0.1 ms smaller than MPC. The aggressiveness seen in overshoot is also seen here. On the other hand, in the settling time (ST) smaller value was obtained in MPC. The most important factor here is the overshoot and oscillations in PI controller responses. The resistance and inductance values of PMSM used in this study are very low. For this reason, high currents can be seen especially during take-off. Because of the optimum switching with MPC, instantaneous accelerations and currents can be suppressed. The low-value fluctuations seen in the velocity responses in PI control, after the settling time, cause continuous current to be drawn from the battery. **Table 1**. shows the performance values of the controllers.

As the second test, a sinusoidal signal with an amplitude of 100 rad/s and a frequency of 2 Hz was applied to the PMSM. Test results are shown in **Figures 14** and **15**. MPC also follows the reference value in this test with a steady state error of
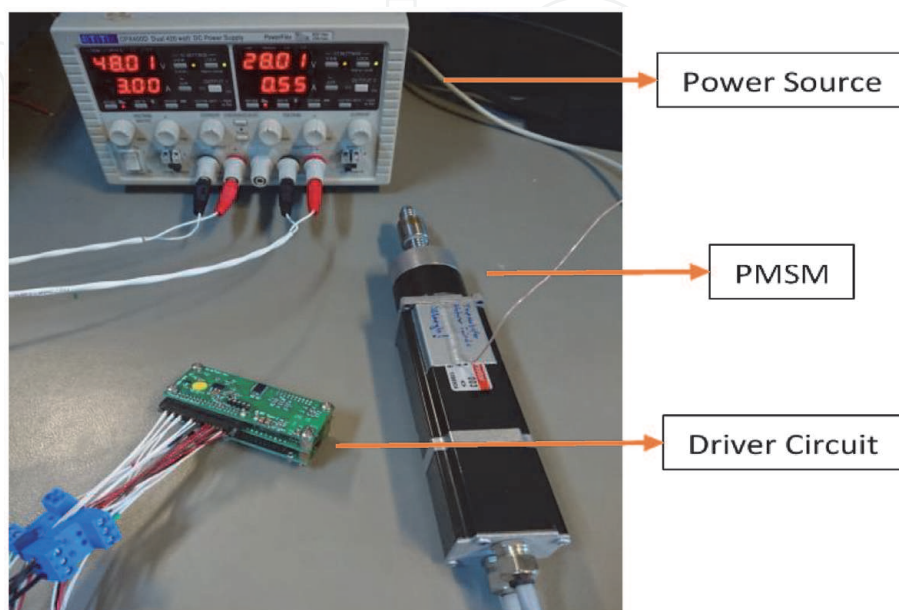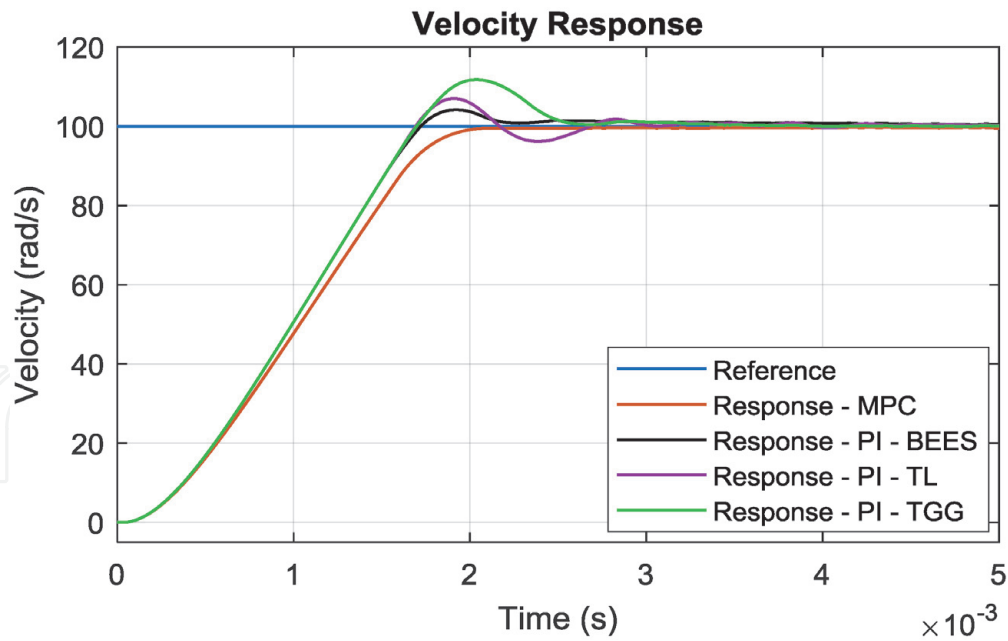

Figure 11.
Experimental environment.

Figure 12.
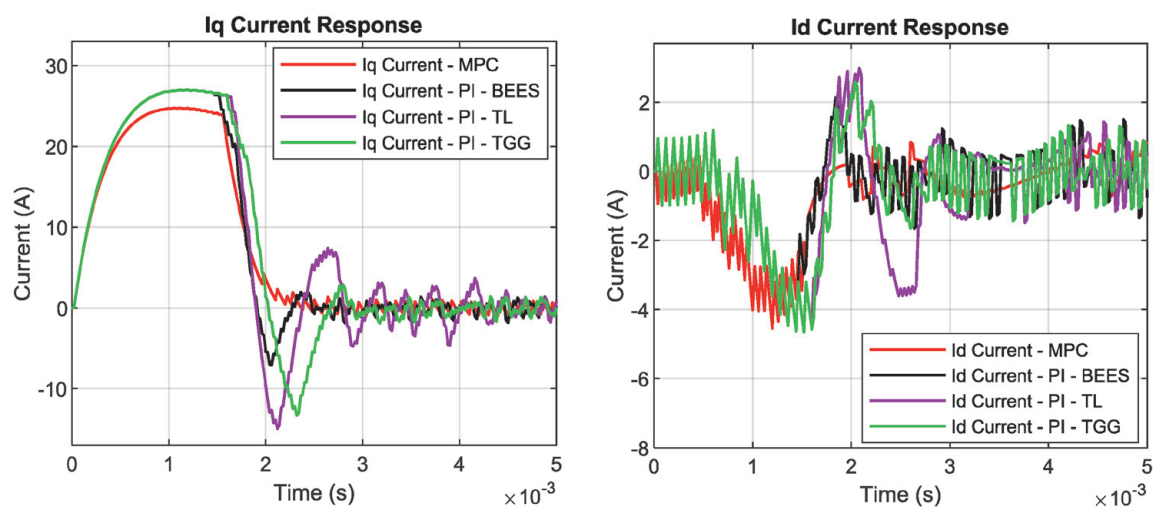*Velocity responses of controllers (step response).*

Figure 13.
Iq & id Current Responses of controllers (step response).

Controller	Results of step response				
	SS error	**RT**	**ST**	**MPO**	**Max. Iq current**
MPC	0.3%	1.3 ms	1.86 ms	0%	24.7 A
PI - BEES	0.5%	1.2 ms	2.2 ms	4.2%	26.9 A
PI - TGG	0.1%	1.2 ms	2.42 ms	11.8%	26.9 A
PI - TL	0.3%	1.2 ms	2.52 ms	7%	26.9 A

Table 1.
Results of step response.

approximately 0.4%. In the MPC cost function design, especially low power consumption was emphasized and power constraints were added. Similarly, in the calculation of the weights with BA, the bus current is also used with the speed error in MOF. Thus, when calculating the weights, the values with low speed error and
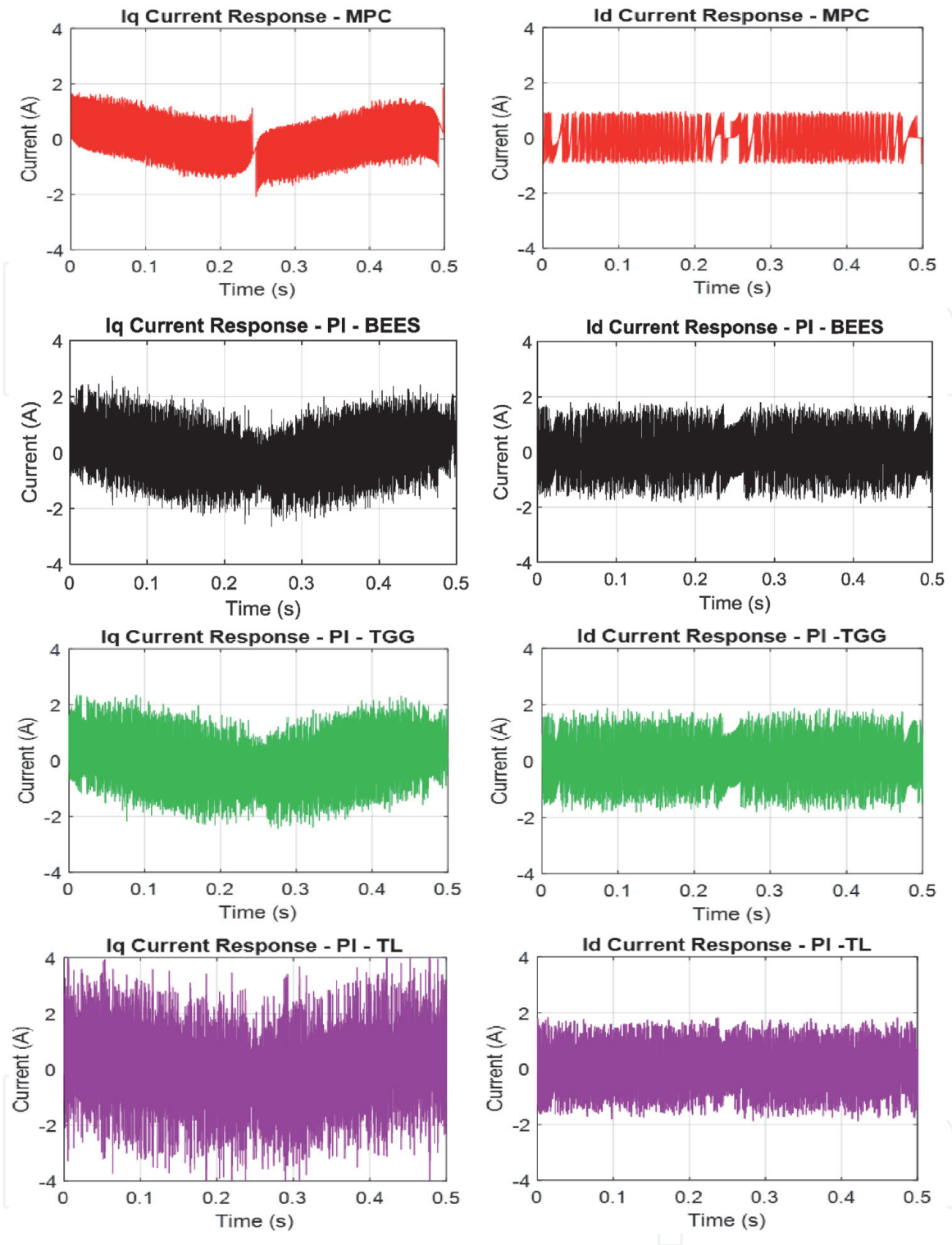
Figure 14.
Iq & id Current Responses of controllers.

low currents at the same time came to the fore. Therefore, a very low steady state error occurs in MPC control.

In PI controllers, although there is no steady state error, higher oscillations are seen compared to MPC. These oscillations can cause significant damage to systems, especially in applications requiring precise control. The effect of oscillations in the velocity response is clearly seen in the flow results. Amplitudes less than 2 A in MPC is up to 4 A in PI controllers.

Finally, a position control application was implemented to test the controllers. Only a P controller with a gain value of 100 has been added to existing controllers. As a test signal, a profile with many changes of direction was used to test its
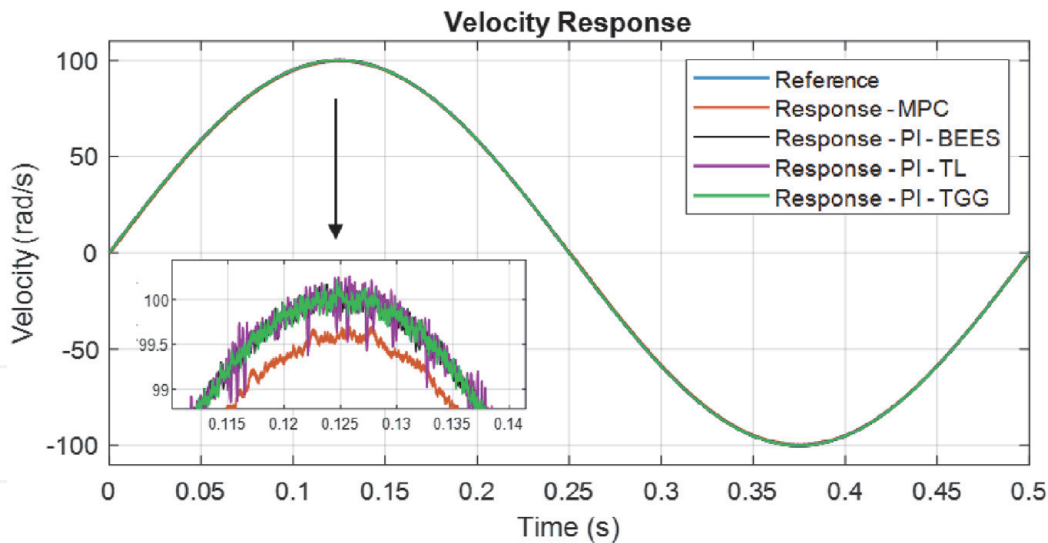
Figure 15.
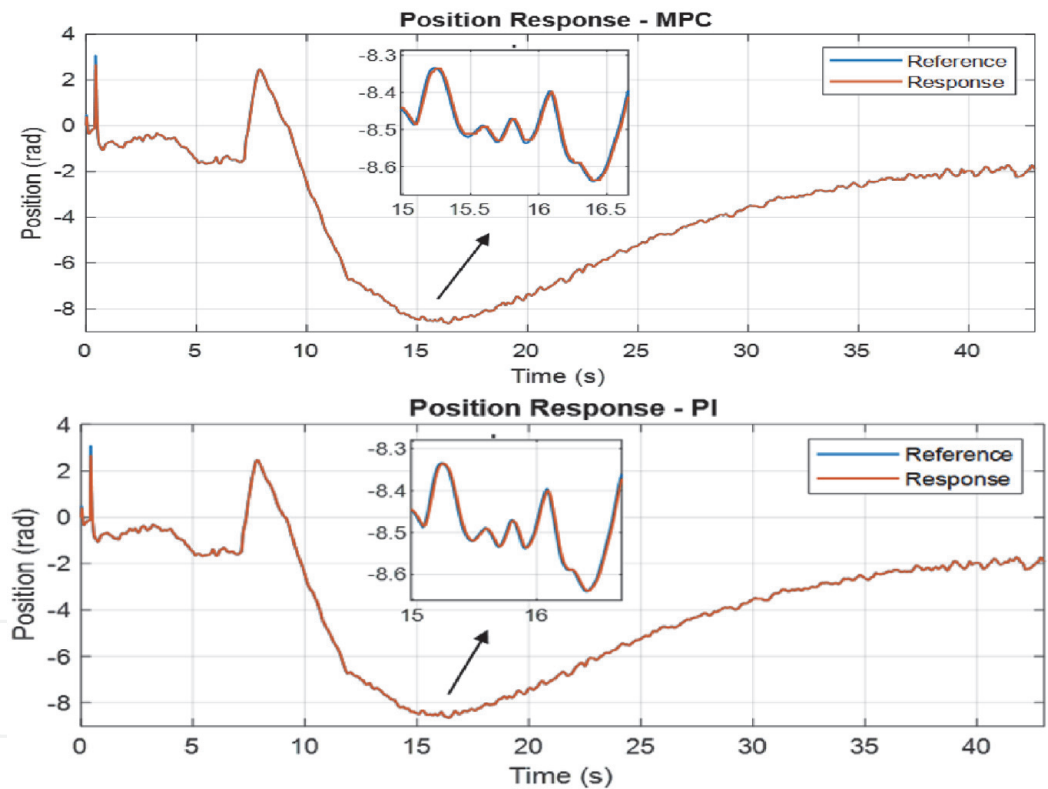*Velocity responses of controllers (sinus response).*

Figure 16.
Position responses of controllers.

performance with the switch for the PI coefficients, those tuned with BA were used. Position results are shown in **Figure 16** and position errors are shown in **Figure 17**.

Both controllers were able to respond to the references successfully. Error values are around 0.02 rad, except for the instantaneous step reference applied initially. Current data can also be seen in **Figure 18**. As with velocity application, MPC control uses less current in position control. It is seen that the optimum inverter switching method used in this study gives successful results. The success of the weights calculated with BA in PMSM control has also been confirmed by the position control application.
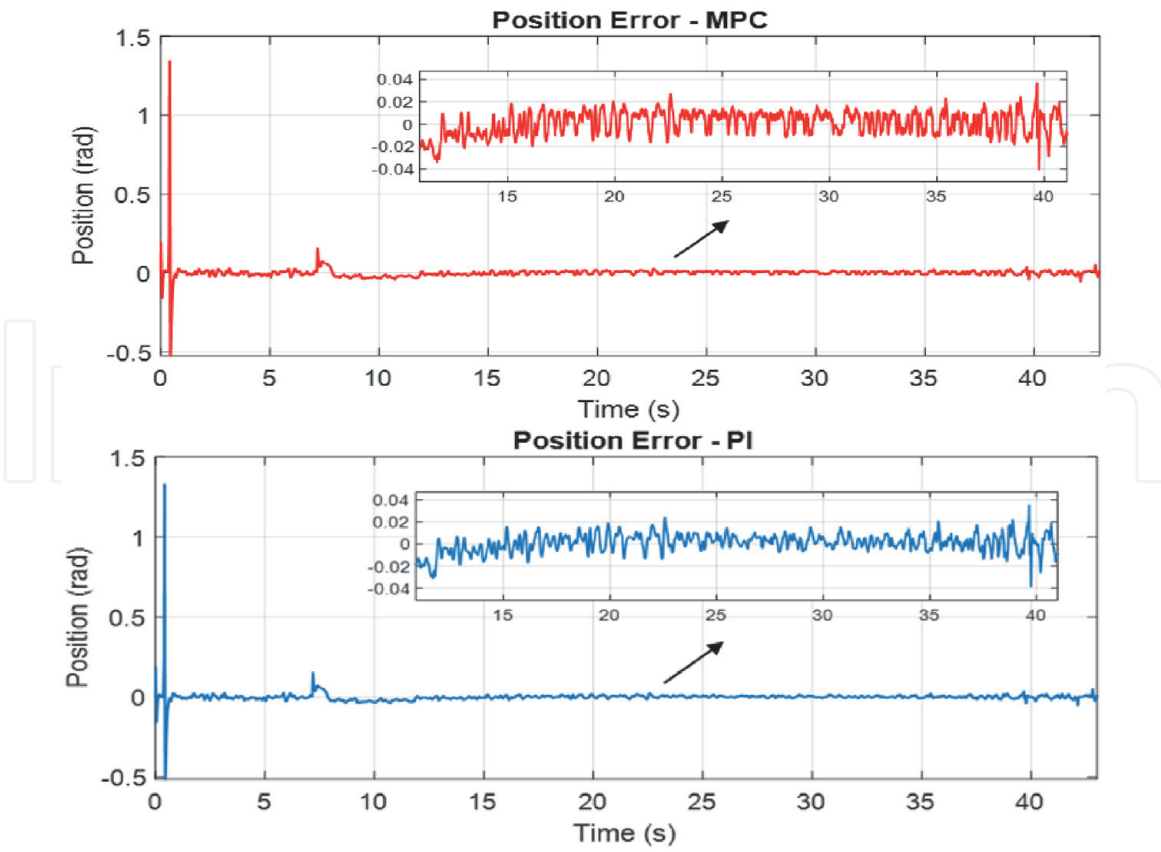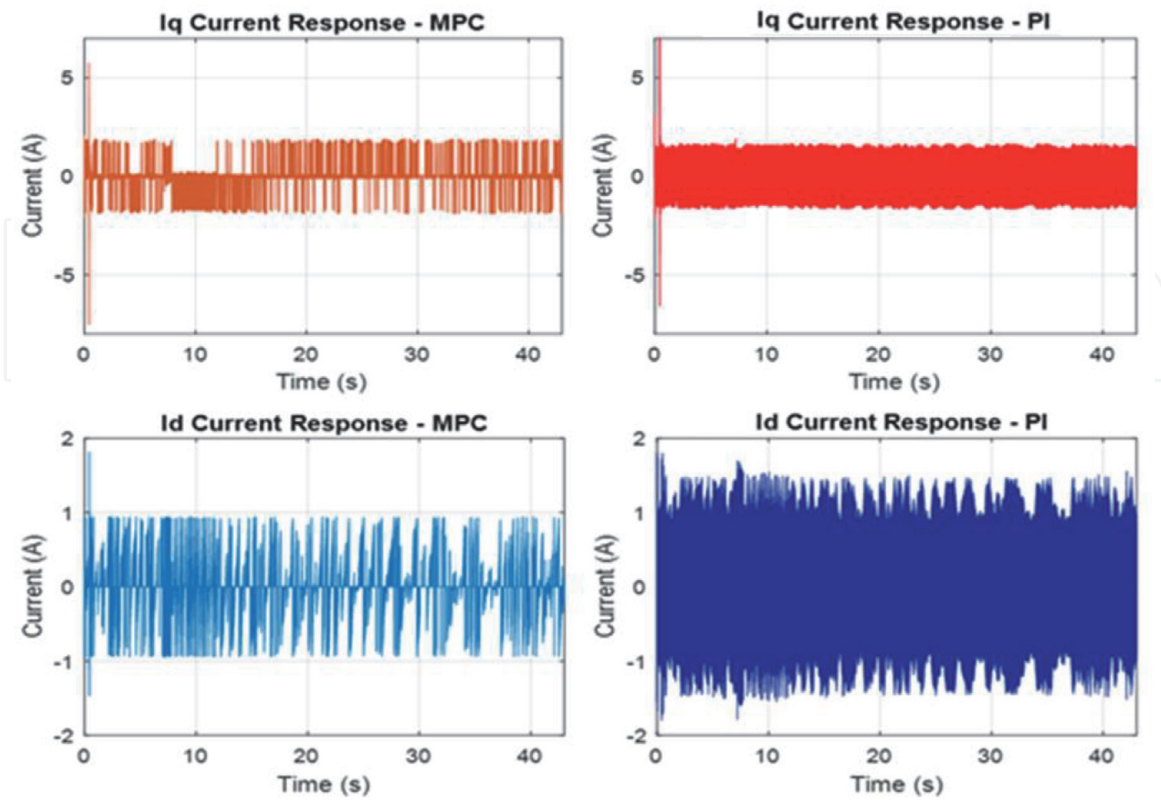
Figure 17.
Position errors of controllers.

Figure 18.
Current responses of position controllers.

6. Conclusion

FCS-MPC has a limited number of optimization and calculation processes. Therefore, the delay compensation method was used to prevent timing errors in MPC. Thus, in this application, MPC was used with a relatively low-level processor without any problem. In addition to speed error and current predictions, power prediction has been added to the standard cost function used for speed control. In this way, in each switching selection, the possibilities that the lowest power consumption may occur along with other factors are evaluated.

For the Bees Algorithm, which is used to determine the weight coefficients, an infrastructure has been established to minimize the speed error and bus current. With the fast search capability of BA, optimum weight coefficients were calculated in approximately 15 iterations. Because of the low current and low energy preferences used in both the MPC and BA, MPC has achieved a more effective and less oscillatory control by using much lower currents than PI methods. With MPC, PMSM has been controlled with 15% settling time than other controllers and also with no overshoot. There is no exact method for determining the weight coefficients. It seems that manual adjustment is still preferred in many applications. With the efficient neighborhood search structure of BA, weights can be calculated with a small number of iterations. It provides great convenience for researchers. By designing a multi objective function, the number of variables that can be optimized can be increased if desired.

Energy consumption in autonomous vehicles and robots is one of great importance. MPC is used in this context with its smooth control structure. As in this application, BA can be preferred for autonomous control applications that require weight optimization. One of the advantages of BA over other meta-heuristic algorithms is that there is no mathematical equation in its structure. In this way, it can be used easily on different platforms and software languages.

In future works, the parameters of the CARIMA method, which is one of the popular MPC methods, will be optimized using this algorithm. A comparison of different MPC methods will be carried out together with the same test system.

Acknowledgements

Conflict of interest

I declare that this manuscript is original, has not been published before, and is not currently being considered for publication elsewhere. I know of no conflicts of interest associated with this publication, and there has been no significant financial support for this work that could have influenced its outcome.

Appendix - A

Model Configuration
Solver type: Fixed step

Solver: ode 5 (Dormand–Prince)
Fixed-step size: 2e-5 [s]
Tasking mode: Singletasking
PMSM Parameters:
R = 0.894;% Resistance [Ohm]
Ts = 2e-5; % Sampling time [s]
L = 0.338e-3;% Inductance [H]
Fl = 0.0329; % Flux linkage [Wb]
Vdc = 48;% DC-link voltage [V]
J = 368e-7; % Inertia [kg.m2]
p = 2; % Pole pairs

Appendix - B

Bees Algorithm Parameters.
MaxIt = 20;% Maximum Number of Iterations
nScoutBee = 20;% Number of Scout Bees
nBestSite = 4;% Number of Best Sites
nEliteSite = 2;% Number of Elite Sites
nBestSiteBee = 5;% Number of Recruited Bees for Best Sites
nEliteSiteBee = 10;% Number of Recruited Bees for Elite Sites

Author details

Murat Sahin
Control Systems Department, Roketsan A.S., Ankara, Turkey

*Address all correspondence to: msahin@roketsan.com.tr

IntechOpen

References

[1] Cai, R., Zheng, R., Liu, M. and Li, M. Robust Control of PMSM Using Geometric Model Reduction and u-Synthesis. IEEE Transactions on Industrial Electronics, vol. 65, no. 1, pp. 498-509, Jan. 2018, doi: 10.1109/TIE.2017.2714140..

[2] Wang, Z., Chen, J., Cheng, M., Chau, K. T. Field-Oriented Control and Direct Torque Control for Paralleled VSIs Fed PMSM Drives With Variable Switching Frequencies. IEEE Transactions on Power Electronics, Volume: 31, Issue: 3, 2016; pp. 2417 – 2428. Doi: 10.1109/TPEL.2015.2437893.

[3] Wang, Z., Yu, A., Li, X., Zhang, G., and Xia, C. A Novel Current Predictive Control Based on Fuzzy Algorithm for PMSM. IEEE Journal of Emerging and Selected Topics in Power Electronics, vol. 7, no. 2, pp. 990-1001, June 2019, doi: 10.1109/JESTPE.2019.2902634.

[4] Feng, G., Lai, C., Kar, N. C. A Closed-Loop Fuzzy-Logic-Based Current Controller for PMSM Torque Ripple Minimization Using the Magnitude of Speed Harmonic as the Feedback Control Signal. IEEE Transactions on Industrial Electronics Volume: 64, Issue: 4, 2017; pp. 2642 – 2653. Doi: 10.1109/TIE.2016.2631524.

[5] Mani, P., Rajan, R., Shanmugam, L., and Joo, Y. H. Adaptive Fractional Fuzzy Integral Sliding Mode Control for PMSM Model. IEEE Transactions on Fuzzy Systems, vol. 27, no. 8, pp. 1674-1686, Aug. 2019, doi: 10.1109/TFUZZ.2018.2886169.

[6] Slapak, V., Kyslan, K., Durovsky, F. Position Controller for PMSM Based on Finite Control Set Model Predictive Control. Elektronika Ir Elektrotechnika, ISSN 1392-1215, VOL. 22, NO. 6, 2016; pp. 17-21. Doi: 10.5755/j01. eie.22.6.17217.

[7] Šlapák, V., Kyslan, K., Lacko, M., Fedák, V., Durovský, F. Finite Control Set Model Predictive Speed Control of a DC Motor. Hindawi Publishing Corporation Mathematical Problems in Engineering, Volume 2016, Article ID 9571972, 2016; 10 pages. Doi: 10.1155/2016/9571972.

[8] Metry, M., Balog, R. S. A Parameter Mismatch Study on Model Predictive Control Based Sensorless Current Mode. IEEE Texas Power and Energy Conference (TPEC). 2018, Doi: 10.1109/TPEC.2018.8312065.

[9] Espinoza, J., Buele, J., Castellanos, E. X., Pilatásig, M., Ayala, P., García, M. V. Real-Time Implementation of Model Predictive Control in a Low-Cost Embedded Device. Systemics, Cybernetics and Informatics Volume 16 - Number 2, 2018; pp. 72-77.

[10] Bartsch, A. G., Negri, G. H., Cavalca, M. S. M., Oliveira, J., and Nied, A. Cost function tuning methodology for FCS-MPC applied to PMSM drives. 2017 Brazilian Power Electronics Conference (COBEP), 2017, pp. 1-6, doi: 10.1109/COBEP.2017.8257320.

[11] Mahmoudi, H., Aleenejad, M., Moamaei, P., Ahmadi, R. Fuzzy Adjustment of Weighting Factor in Model Predictive Control of Permanent Magnet Synchronous Machines Using Current Membership Functions. IEEE Power and Energy Conference at Illinois (PECI), 2016; Doi: 10.1109/PECI.2016.7459225.

[12] Rodríguez, J., Kennel, R. M., Espinoza, J. R., Trincado, M., Silva, C. A., Rojas, C. A. High-Performance Control Strategies for Electrical Drives: An Experimental Assessment. IEEE Transactions on Industrial Electronics, Vol. 59, No. 2, 2012; pp. 812-820. Doi: 10.1109/TIE.2011.2158778.

[13] Codreş, B., Găiceanu, M., Şolea, R., Eni, C. Model Predictive Speed Control of Permanent Magnet Synchronous Motor. International Conference on Optimization of Electrical and Electronic Equipment (OPTIM), 2014; pp. 477-482. Doi: 10.1109/ OPTIM.2014.6850946.

[14] Dani, S., Sonawane, D., Ingole, D., Patil, S. Performance Evaluation of PID, LQR and MPC for DC Motor Speed Control. 2nd International Conference for Convergence in Technology (I2CT), 2017; pp. 348-354. Doi: 10.1109/ I2CT.2017.8226149

[15] Ma, H., Chu, L., Guo, J., Wang, J., and Guo, C. Cooperative Adaptive Cruise Control Strategy Optimization for Electric Vehicles Based on SA-PSO With Model Predictive Control. IEEE Access, vol. 8, pp. 225745-225756, 2020, doi: 10.1109/ACCESS.2020.3043370.

[16] Zanchetta, P. Heuristic Multi-Objective Optimization for Cost Function Weights Selection in Finite States Model Predictive Control. Workshop on Predictive Control of Electrical Drives and Power Electronics, 2011; pp. 70-75. Doi: 10.1109/ PRECEDE.2011.6078690.

[17] Mohammadi, A., Asadi, H., Mohamed, S., Nelson, K., Nahavandi, S. Multiobjective and Interactive Genetic Algorithms for Weight Tuning of a Model Predictive Control-Based Motion Cueing Algorithm. IEEE Transactions on Cybernetics, Vol. 49, No. 9, 2019; pp. 3471-3481. Doi: 10.1109/ TCYB.2018.2845661.

[18] Zhou, Z.D., Xie, Y. Q., Pham, D. T., Kamsani, S., and Castellani, M. Bees Algorithm for multimodal function optimization. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 2016, Volume: 230 issue: 5, page(s): 867-884. doi.org/ 10.1177/0954406215576063. .

[19] Zarchi, M. & Attaran, B. Performance improvement of an active vibration absorber subsystem for an aircraft model using a bees algorithm based on multi-objective intelligent optimization. 2017, Engineering Optimization, 49:11, 1905-1921, DOI: 10.1080/0305215X.2017.1278757. .

[20] Moradia, A., Nafchib, A. M., Ghanbarzadeh, A. Multi-objective optimization of truss structures using the bee algorithm. Scientia Iranica B (2015) 22(5), 2015; pp. 1789-1800.

[21] Kizir, G., Demirbas, S., Sahin, M. Real-time resistance estimation of permanent magnet synchronous motor. 2018 IEEE 12th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG 2018); pp. 1-5. Doi: 10.1109/ CPE.2018.8372532.

[22] Rodriguez, J., and Cortes, P. Predictive Control of Power Converters and Electrical Drives. John Wiley & Sons, Ltd., Publication, 2012; pp. 133-143. ISBN: 978-1-119-94264-1.

[23] Wang, F. Model Predictive Torque Control for Electrical Drive Systems with and without an Encoder. PhD Thesis, Technical University of Munich, 2014; pp. 7-10.

[24] Jin, T., Shen, X., Su, T., Flesch, R. C. C. Model Predictive Voltage Control Based on Finite Control Set With Computation Time Delay Compensation for PV Systems. IEEE Transactions on Energy Conversion (Volume: 34, Issue: 1, March 2019); pp. 330-338. Doi: 10.1109/TEC.2018.2876619.

[25] Khosravi, M., Amirbande, M., Khaburi, D. A., Rivera, M., Riveros, J., Rodriguez, J., Vahedi, A., and Wheeler, P. Review of model predictive control strategies for matrix converters. IET Power Electronics, 2019, Volume 12, Issue 12 p. 3021-3032, doi.org/ 10.1049/iet-pel.2019.0212. .

[26] Baronti, L., Castellani M., Pham D. T. An analysis of the search mechanisms of the bees algorithm. Swarm and Evolutionary Computation 59, 2020; 100746. Doi: 10.1016/j. swevo.2020.100746.

[27] Liu, J., Zhou, Z., Pham, D. T., Xu, W., Yan, J., Liu, A., Ji, C, Liu, Q. An improved multi-objective discrete bees algorithm for robotic disassembly line balancing problem in remanufacturing. The International Journal of Advanced Manufacturing Technology, 2018; 97: 3937–3962.

[28] Ismail, A. H., Hartono, N., Zeybek, S., Pham, D. T. Using the Bees Algorithm to solve combinatorial optimisation problems for TSPLIB. IOP Conf. Series: Materials Science and Engineering 847, 2020; 012027.

[29] Castellani, M, Otri, S., and Pham, D. T. Printed circuit board assembly time minimisation using a novel Bees Algorithm. Computers & Industrial Engineering, Volume 133, 2019, Pages 186-194, ISSN 0360-8352, https://doi.org/10.1016/j.cie. 2019.05.015.

[30] Zheng, J. H., Kou, Y. N., Jing, Z. X., and Wu, Q. H. Towards Many-Objective Optimization: Objective Analysis, Multi-Objective Optimization and Decision-Making. IEEE Access, vol. 7, pp. 93742-93751, 2019, doi: 10.1109/ ACCESS.2019.2926493..

[31] Gunantara, N. A review of multi-objective optimization: Methods and its applications. Journal Cogent Engineering, Volume 5, 2018 - Issue 1, 2018; pp. 1-16. Doi: 10.1080/ 23311916.2018.1502242.

[32] Hruska, K.; Dvorak, P. Optimization of a PMSM Design for Control with Zero Direct Axis Current Component. 2016 ELEKTRO, 10.1109/ ELEKTRO.2016.7512057. 2016; Doi: 10.1109/ELEKTRO.2016.7512057.

[33] Namboodiri, A., Wani, H. S. Unipolar and Bipolar PWM Inverter. IJIRST –International Journal for Innovative Research in Science & Technology| Volume 1, Issue 7. 2014; ISSN (online): 2349-6010.

[34] Haugen, F. Comparing PI Tuning Methods in a Real Benchmark Temperature Control System. Modeling, Identification and Control, Vol. 31, No. 3, 2010; pp. 79-91, ISSN 1890-1328. Doi: 10.4173/mic.2010.3.1.

[35] Haugen, F. The Good Gain method for simple experimental tuning of PI controllers. Modeling, Identification and Control, Vol. 33, No. 4, 2012; pp. 141–152, ISSN 1890–1328. Doi: 10.4173/ mic.2012.4.3.