

Parameter Estimation and Predictive Speed Control of Chopper-Fed Brushed DC Motors

Original Scientific Paper

Son Nguyen Thanh

Hanoi University of Science and Technology,
School of Electrical and Electronic Engineering
1 Dai Co Viet Street, Hanoi, Vietnam
son.nguyenthanh@hust.edu.vn

Tuan Pham Van

Vinh University of Technology Education,
Faculty of Electrical Engineering
117 Nguyen Viet Xuan Street, Vinh City, Vietnam
tuanvp.bk@gmail.com

Tu Pham Minh

Hanoi University of Science and Technology,
School of Electrical and Electronic Engineering
1 Dai Co Viet Street, Hanoi, Vietnam
tu.phamminh@hust.edu.vn

Anh Hoang

Hanoi University of Science and Technology,
School of Electrical and Electronic Engineering
1 Dai Co Viet Street, Hanoi, Vietnam
anh.hoang@hust.edu.vn

Abstract – This paper presents an effective speed control method for brushed DC motors fed by a DC chopper using the concept of Finite Control Set-Model Predictive Control (FCS-MPC). As this control algorithm requires the parameters of the controlled object, the estimation of motor parameters is first performed by using two types of data. The first data includes the output speed response corresponding to the step input voltage to obtain the transfer function in the no-load regime. The second data consists of the motor speed and armature current when a load torque is applied to the motor shaft. The discrete-time equation of the motor armature circuit is used to obtain the future values of the armature circuit current and the motor speed. A cost function is defined based on the difference between the reference and predicted motor speed. The optimal switching states of the DC chopper are selected corresponding to the maximum value of the cost function. The performance of the proposed speed control algorithm is validated on an experimental system. The simulation and experimental results obtained show that the MPC controller can outperform the conventional proportional-integral (PI) controller.

Keywords: Brushed DC motor, DC chopper, motor parameter estimation, motor speed control, model predictive control

1. INTRODUCTION

Electric drive systems can be only seen as high-performance drive systems if they can accurately follow specified trajectories regardless of unknown load variations and uncertainties of parameters. An electric drive system is basically formed by an electrical motor, a power electronic converter, and a controller to perform precise mechanical manoeuvres. As brushed DC motors can be controlled over a wide speed range, they are still widely used in various industrial and commercial applications including electric vehicles, robotic manipulators, and precise mechanisms. In addition, brushed DC motors can be known as the most common controlled object for testing and evaluating different proposed control algorithms.

The accurate mathematical model and related parameters of a specific brushed DC motor are usually needed for designing an appropriate controller for the motor. In addition, the exact model of the motor can allow the designer to predict the closed-loop dynamics

of the motor control system. This work can be only facilitated if the motor parameters can be estimated. Motor parameter estimation approaches can be divided into two categories: offline estimation methods [1] and online estimation methods [2]. Offline methods require the use of test inputs and the measurement of corresponding parameters using data acquisition (DAQ) devices. Next, coefficients determined by the measurement are used to obtain unknown parameters by an offline computer-based analysis. Meanwhile, online parameter estimation techniques sometimes include power converters and high-speed DAQ devices.

The conventional design of controllers for brushed DC motors often involves the use of constant gains for controllers like proportional-integral-derivative (PID) controllers, which are only useful for the control of a narrow speed range [3, 4]. Recently, brushed DC motor drive control systems have been developed using non-linear control approaches, such as sliding mode controllers [5, 6], fuzzy logic controllers [7, 8], and artificial neural network controllers [9, 10].

As high-performance brushed DC motor drives frequently need to be adaptable, several studies on motor control have focused on adaptive control methods [11, 12], in which the coefficients of the controller can be adaptive utilising techniques for intelligent inference such as fuzzy logic and artificial neural networks. To increase the robustness of the motor control system, some studies have suggested techniques for estimating the speed of brushed DC motors [13-15] because the use of speed sensors could degrade system reliability and necessitate periodic maintenance. Most speed estimation techniques, however, have been developed using a precise mathematical model of the motor.

With the recent rapid development of powerful and fast microcontrollers, there is increasing attention to the exploration of model predictive control (MPC) for various topologies of power converters [16-18]. The MPC algorithms can eliminate the use of pulse width modulation (PWM) techniques and conventional PID controllers. In addition, the MPC control algorithms can take the nonlinearities of controlled objects into account.

This research focuses on the development of an effective controller for a chopper-fed DC motor drive using the Finite Control Set-Model Predictive Control (FCS-MPC) theory. Specifically, a motor speed predictive control strategy and its practical implementation have been presented in detail. It has been shown that the proposed control method can be used to control the motor speed very effectively. The proposed control method can result in good dynamic responses compared to the use of a conventional PI controller. Moreover, the proposed MPC-based method in this study is simple and easy to implement on low-cost hardware. In this study, the proposed control method is implemented on an STM32F4 Discovery board which is conveniently programmed using MATLAB Simulink. The analysis for obtaining the discrete model can be also used for other controlled objects.

The rest of this paper is divided into subsequent sections. Section 2 provides the concept of MPC for controlling power electronic converters and electric drives. The motor parameter estimation process is described in Section 3. The procedure for designing an MPC controller for a chopper-fed brushed DC motor is described in Section 4. The simulation and experimental results obtained are presented in Section 5. Finally, Section 6 is the conclusion of this research.

2. MODEL PREDICTIVE CONTROL

Model predictive control (MPC) emerged as an advanced process control method in the 1960s. In the future, this control technique will be expected to be very effective for various nonlinear, multiple-input multiple-output (MIMO) systems. In MPC, the model of the controlled object is considered to predict the future behaviours of the variables over a time frame.

These predictions are then used for evaluating a cost function and the choice of the optimal action for the system based on the minimum value of the cost function. The MPC has several advantages such as the inclusion of nonlinearities and constraints of control systems. Another advantage of MPC in power converters and electric drives is to take the consideration the inherent discrete nature of power converters. As power converters only have a finite number of switching states, the MPC optimisation problem can be simplified to behaviours of possible switching states of power converters. In every sampling interval, the predicted variables are used to evaluate a cost function. This is also known as the Finite Control Set-Model Predictive Control (FCS-MPC). Therefore, this study only focuses on FCS-MPC for a DC chopper driving a brushed DC motor.

Fig. 1 shows the operating principle of FCS-MPC. The control process of power converters and electric drives can be defined as the determination of an appropriate control action $S(t)$ for the gate signals of power converter switches to obtain a desired system variable $x(t)$ as close as possible to a reference value $x^*(t)$.

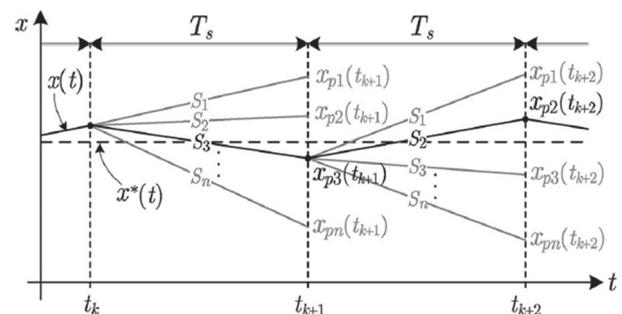


Fig. 1. The FCS-MPC operating principle

In a specific sampling period T_s , the variable $x(t)$ is sampled as $x(t_k)$ for evaluating the cost function with possible control actions of the power converter (S_1, \dots, S_n). If the action control S_3 is selected, it can be used to predict the future value $x_{p3}(t_{k+1})$. Next, the predicted value in the previous sampling period will be used to compute the cost function for selecting the control action S_2 corresponding to the minimum value of the cost function. Similarly, $x_{p2}(t_{k+2})$ can be also predicted based on $x_{p3}(t_{k+1})$ and the control action S_2 . In general, the possible predictive values of the control variable $x(t)$ can be expressed as:

$$x_{pi}(t_{k+1}) = f_p \{x(t_k), S_i\} \quad i = 1, \dots, N \quad (1)$$

where f_p is the prediction function obtained from the controlled model. Then a cost function is defined as:

$$g_i = |x^*(t_{k+1}) - x_{pi}(t_{k+1})| \quad i = 1, \dots, N \quad (2)$$

The evaluation of the cost function (2) results in N values of the cost function. The control action leading to the minimum value of the cost function is finally selected.

3. PARAMETER ESTIMATION OF BRUSHED DC MOTORS

For brushed DC motors, a DC voltage can be directly applied to the terminal of the armature circuit. Using Kirchhoff's law for the armature circuit gives:

$$V_{dc}(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + K\omega(t) \quad (3)$$

Based on Newton's law, the torque equation can be derived as follows:

$$K i_a(t) = J \frac{d\omega(t)}{dt} + D\omega(t) + T_L(t) \quad (4)$$

Where K , R_a , L_a , J and D are the back-EMF constant, the armature resistance, the armature inductance, the rotor mass moment of inertia, and the damping constant, respectively. $\omega(t)$, $i_a(t)$, $V_{dc}(t)$ and $T_L(t)$ denote the rotor angular velocity, the armature current, the terminal voltage, and the load torque, respectively.

At the steady-state, equation (3) yields:

$$\omega = \frac{V_{dc}}{K} - \frac{R_a I_a}{K} = \omega_0 - \Delta\omega \quad (5)$$

In which:

$$\omega_0 = \frac{V_{dc}}{K} \quad (6)$$

$$\Delta\omega = \frac{R_a I_a}{K} \quad (7)$$

In equations (6) and (7), ω_0 is the no-load rotor angular velocity and $\Delta\omega$ is the drop of the rotor angular velocity when a load torque is applied to the motor rotor. The back-EMF constant can be determined through the terminal voltage and the no-load rotor angular velocity as follows:

$$K = \frac{V_{dc}}{\omega_0} \quad (8)$$

When a load torque is applied to the motor shaft, the armature current is measured to calculate the armature resistance as follows:

$$R_a = \frac{V_{dc} - K\omega}{I_a} \quad (9)$$

According to equation (9), if V_{dc} is kept being a constant, ω can be measured using a tachometer or an encoder, and I_a can be measured using a DC ammeter, then the value of R_a can be computed.

The Laplace transforms of equations (3) and (4) have the following forms:

$$V_{dc}(s) = R_a I_a(s) + L_a s I_a(s) + K\omega(s) \quad (10)$$

$$J s \omega(s) = K I_a(s) - D\omega(s) - T_L(s) \quad (11)$$

Where s is the Laplace operator. $V_{dc}(s)$, $I_a(s)$, $\omega(s)$ and $T_L(s)$ are the Laplace transforms of the terminal voltage, the armature current, the rotor angular velocity, and the load torque, respectively.

From equations (10) and (11), the relationship between the rotor angular velocity (the system output) and the terminal voltage, and the load torque (the system inputs) can be expressed as follows:

$$\omega(s) = \frac{K V_{dc}(s) - T_L(s)(s L_a + R_a)}{s^2 L_a J + s(R_a J + L_a D) + (R_a D + K^2)} \quad (12)$$

In the no-load regime ($T_L(s) = 0$), equation (12) yields:

$$\omega(s) = \frac{K}{s^2 L_a J + s(R_a J + L_a D) + (R_a D + K^2)} V_{dc}(s) \quad (13)$$

Equation (13) indicates a second-order system and can be shorten as follows:

$$\frac{\omega(s)}{V_{dc}(s)} = \frac{a}{s^2 + sb + c} \quad (14)$$

in which:

$$a = \frac{K}{L_a J} \quad b = \frac{R_a J + L_a D}{L_a J} \quad c = \frac{R_a D + K^2}{L_a J} \quad (15)$$

The coefficients a , b , and c of equation (14) can be obtained by acquiring the start-up motor speed with respect to the step terminal voltage and the MATLAB System Identification Toolbox [19]. According to equation (15), K and R_a can be determined using equations (8) and (9). Three remaining parameters of the motor can be computed as follows:

$$D = \frac{1}{R_a} \left(\frac{cK}{a} - K^2 \right) \quad (16)$$

$$L_a = \frac{bK - \sqrt{(bK)^2 - 4aDKR_a}}{2aD} \quad (17)$$

$$J = \frac{K}{aL_a} \quad (18)$$

4. MODEL PREDICTIVE CONTROL OF CHOPPER-FED BRUSHED DC MOTOR

If the sampling interval T_s is small enough, the derivative of the armature current can be approximated as follows:

$$\frac{di_a(t)}{dt} = \frac{i_a(t_k) - i_a(t_{k-1})}{T_s} \quad (19)$$

Fig. 2 shows the principle circuit of a brushed DC motor driven by a DC chopper. Fig. 3. shows the armature circuit when the switch S is closed. The armature circuit of the motor is depicted in Fig. 4.

Substituting (19) into (3) gives:

$$V_{dc} = \left(R_a + \frac{L_a}{T_s} \right) i_a(t_k) - \frac{L_a}{T_s} i_a(t_{k-1}) + K\omega(t_k) \quad (20)$$

Equation (20) corresponds to the switch S is closed. When the switch S is opened, the equation (20) yields:

$$0 = \left(R_a + \frac{L_a}{T_s} \right) i_a(t_k) - \frac{L_a}{T_s} i_a(t_{k-1}) + K\omega(t_k) \quad (21)$$

Equation (20) and (21) can be combined to give:

$$V_{dc}u = \left(R_a + \frac{L_a}{T_s} \right) i_a(t_k) - \frac{L_a}{T_s} i_a(t_{k-1}) + K\omega(t_k) \quad (22)$$

where u is the binary control signal for the switch S

- if $u = 1$, S is closed.
- if $u = 0$, S is opened.

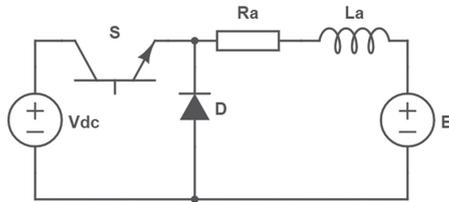


Fig. 2. The principle circuit of DC motor fed by a DC chopper

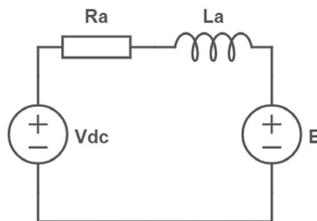


Fig. 3. The armature circuit when the switch S is closed

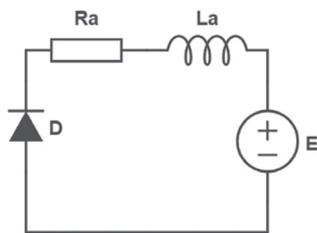


Fig. 4. The armature circuit when the switch S is opened

Equation (22) can be re-written as follows:

$$i_a(k) = \frac{uT_s V_{dc}}{R_a T_s + L_a} - \frac{KT_s \omega(t_k)}{R_a T_s + L_a} + \frac{L_a i_a(t_{k-1})}{R_a T_s + L_a} \quad (23)$$

Applying the two-step-ahead prediction for equation (23) gives:

$$i_a(t_{k+1}) = \frac{uT_s V_{dc}}{R_a T_s + L_a} - \frac{KT_s \omega(t_{k+1})}{R_a T_s + L_a} + \frac{L_a i_a(t_k)}{R_a T_s + L_a} \quad (24)$$

Re-arranging (24) results in:

$$\omega(t_{k+1}) = \frac{V_{dc}u}{K} - \frac{(R_a T_s + L_a)}{KT_s} i_a(t_{k+1}) - \frac{L_a i_a(t_k)}{KT_s} \quad (25)$$

Equation (24) and (25) are used to design a MPC controller for the speed control of the motor. A cost function is defined according to the difference of the reference and future speeds as follows:

$$g = |\omega^* - \omega(t_{k+1})|$$

In every sampling interval, there are two possible switching states of the switch S : the switch S is closed, and the switch S is opened. The optimal state of the switch S will be selected corresponding to the minimum value of the cost function. Fig. 5 is a flowchart of the algorithm of the MPC controller.

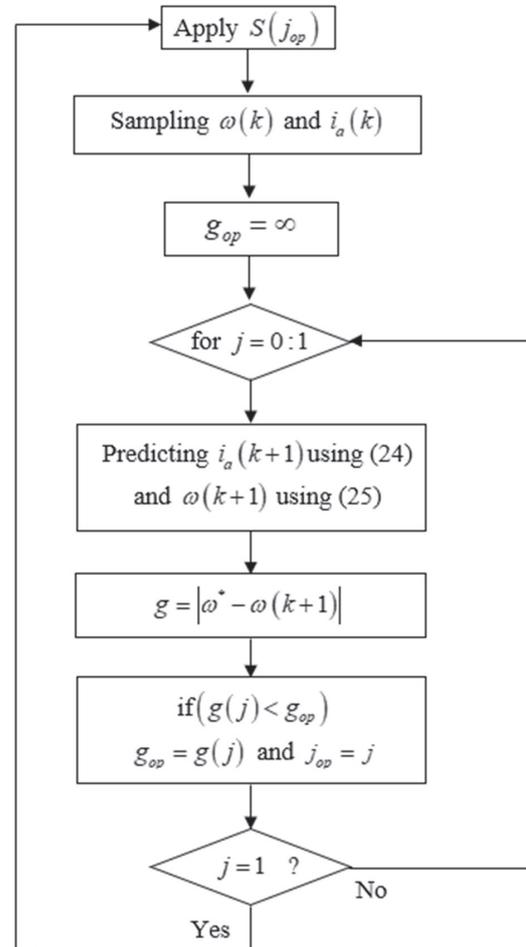


Fig. 5. Flowchart of the MPC control algorithm.

5. SIMULATION AND EXPERIMENTAL RESULTS

This section describes the procedure for deploying an experimental system for the MPC-based speed control of a brushed DC motor. The whole system is shown in Fig. 6. Table 1 is the specification of the experimental system.

Table 1. Specification of the experimental system.

| No | Items |
|----|--|
| 1 | An excited-separately 175W DC motor |
| 2 | An IGBT-based DC chopper can be controlled by an external signal |
| 3 | A dynamometer is used to adjust the load torque |
| 4 | An isolated current measurement module is used to measure the armature current |
| 5 | Speed measurement and load torque modules are integrated with the dynamometer |
| 6 | A laptop is used to monitor the performance of the system |
| 7 | An Arduino MEGA 2560 board |
| 8 | A STM32F4 Discovery board |

A. Motor Parameter Estimation

Fig. 7 is a Simulink diagram for acquiring the output speed response according to the step input voltage in the no-load regime. The motor speed and armature current were acquired by using a Simulink diagram as shown in Fig. 8. The output speed response according to the step input voltage in the no-load regime is illustrated in Fig. 9.

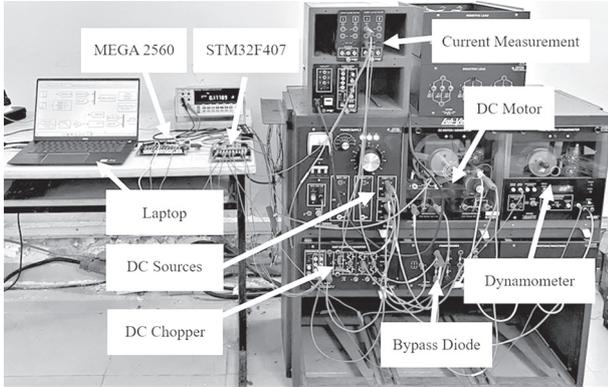


Fig. 6. The whole experimental system

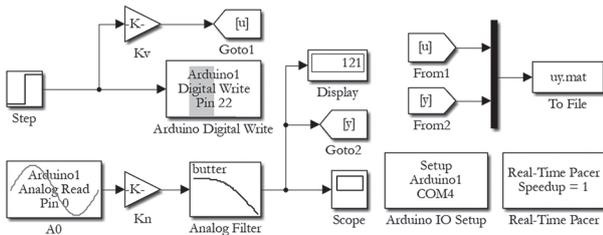


Fig. 7. Simulink diagram for acquiring the output speed response according to the step input voltage

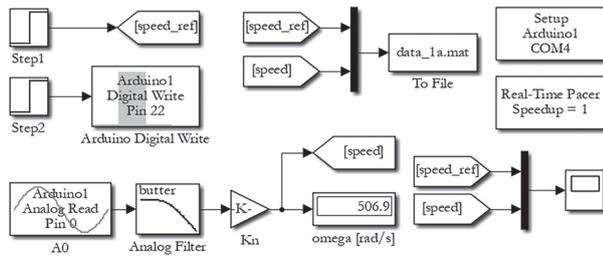


Fig. 8. Simulink diagram for acquiring the motor speed and the armature current

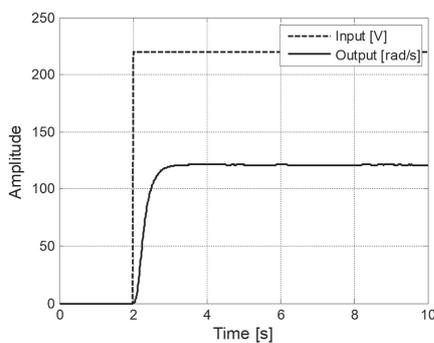


Fig. 9. The output speed response according to the step input voltage of the motor in the no-load regime

The data acquired from the no-load experiment including the step input voltage and the rotor angular velocity as shown in Fig. 9 was then used to estimate the transfer function of the motor. By using the MATLAB System Identification Toolbox [19], the transfer function of the motor in the no-load regime has the following form:

$$\frac{\omega(s)}{V_a(s)} = \frac{18.34}{s^2 + 10.36s + 33.62} \quad (27)$$

Therefore

$$a=18.34; b=10.36; c=33.62 \quad (28)$$

The back-EMF coefficient is calculated as:

$$K = \frac{V_a}{\omega_0} = \frac{228}{126} = 1.8095 [V / ras / s] \quad (29)$$

If applying $T_L = 2.5$ [N.m] to the rotor shaft and the measurements are $\omega 1 = 108$ [rad/s] and $I_a = 1.397$ [A], then the resistance of the armature circuit is computed as:

$$R_a = \frac{V_a - K\omega}{I_a} = \frac{220 - 1.8095 \times 108}{1.397} = 17.5887 [\Omega] \quad (30)$$

The damping constant is determined as:

$$D = \frac{1}{R_a} \left(\frac{cK}{a} - K^2 \right) = \frac{1}{17.5887} \left(\frac{33.62 \times 1.8095}{18.34} - 1.8095^2 \right) = 24 \times 10^{-4} [N.m.s] \quad (31)$$

The armature inductance is computed as:

$$L_a = \frac{bK - \sqrt{(bK)^2 - 4aDKR_a}}{2aD} = 1.7047 [H] \quad (32)$$

Finally, the moment of inertia is given by:

$$J = \frac{K}{aL_a} = \frac{1.8095}{18.34 \times 1.7047} = 0.0579 [kg.m^2] \quad (33)$$

B. Simulation Results

Fig. 10 shows a Simulink diagram for investigating the performance of the PI controller. The simulated speed step increment with the PI controller is illustrated in Fig. 11 and the simulated speed step decrement with the PI controller is depicted in Fig. 12.

Fig. 13 displays a Simulink diagram for investigating the performance of the MPC controller. The MPC algorithm is developed using a user-defined function as follows:

function $S = \text{MPC}(\text{speed_ref}, \text{speed}, ia)$

$R_a = 17.5887;$

$L_a = 1.7047;$

$K = 1.8095;$

$V_{dc} = 230;$

$T_s = 0.2;$

$\omega_{ref} = \text{speed}_{ref} * \text{pi}/30;$

$\omega = \text{speed} * \text{pi}/30;$

$g_{opt} = \text{inf};$

```

 $S_{opt} = 0;$ 
for  $S = 0:1$ 
     $ia_1 = T_s^*V_{dc}/(R_a^*T_s + L_a)^*S + L_a/(R_a^*T_s + L_a)^*ia -$ 
         $K^*T_s/(R_a^*T_s + L_a)^*\omega;$ 
     $\omega_{a1} = V_{dc}/K^*S - (R_a^*T_s + L_a)/K/T_s^*ia_1 + L_a/K/T_s^*ia;$ 
     $g = \text{abs}(\omega_{a1} - \omega_{ref});$ 
    if ( $g < g_{opt}$ )
         $g_{opt} = g;$ 
         $S_{opt} = S;$ 
    end
end
 $S = S_{opt};$ 

```

The simulated speed step increment with the MPC controller is shown in Fig. 14. Lastly, the simulated speed step decrement with the MPC controller is illustrated in Fig. 15.

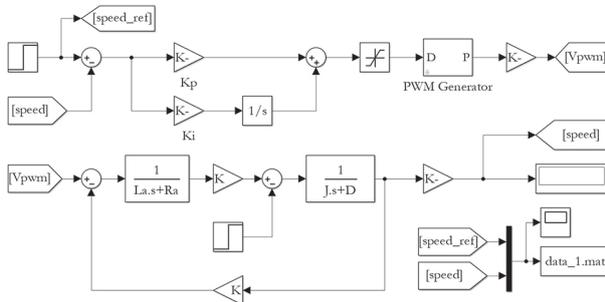


Fig. 10. Simulink diagram for investigating the performance of the PI controller

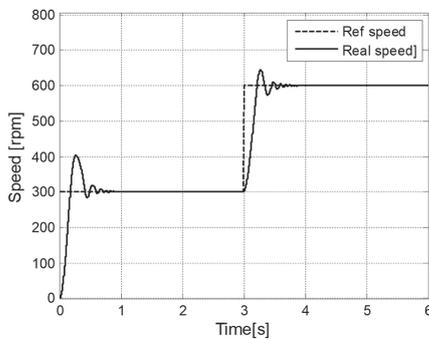


Fig. 11. Simulated speed step increment with the PI controller

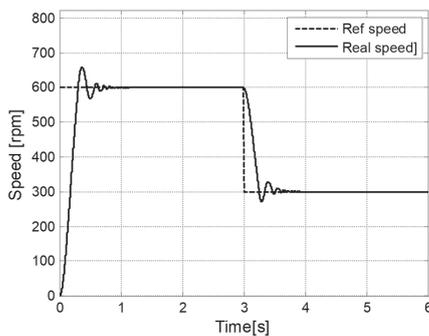


Fig. 12. Simulated speed step decrement with the PI controller

The simulated speed step increment with the MPC controller is shown in Fig. 14. Lastly, the simulated speed step decrement with the MPC controller is illustrated in Fig. 15.

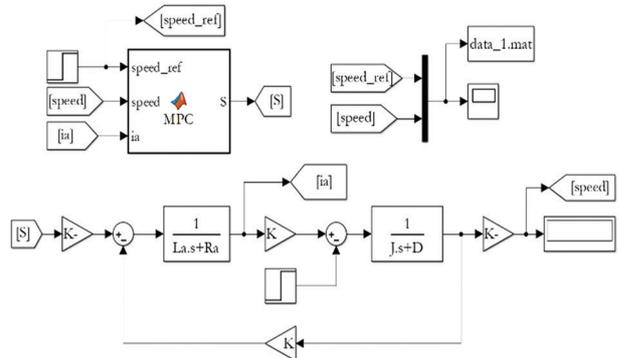


Fig. 13. Simulink diagram for investigating the performance of the MPC controller

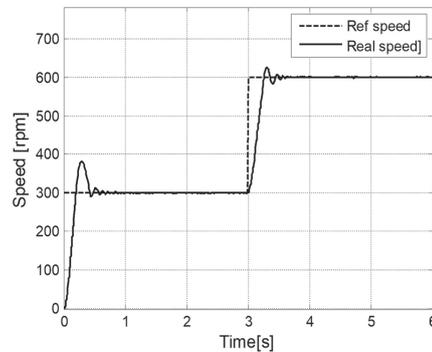


Fig. 14. Simulated speed step increment with the MPC controller

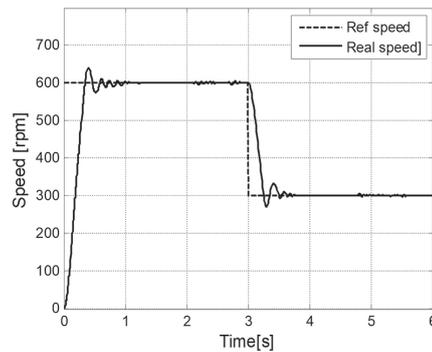


Fig. 15. Simulated speed step decrement with the MPC controller

C. Experimental Results

Fig. 16 is a Simulink diagram using the Waijung Blockset for deploying a discrete-time PI controller on the STM32F4 board [20]. The change of the reference speed is controlled by an external signal at pin PA1 of the board. The PI controller has the proportional constant $K_p = 100$ and the integral constant $K_i = 0,1$. The sampling period is $0,0001[s]$. Fig. 17 is a Simulink diagram using the Waijung Blockset for deploying the MPC controller on the STM32F4 Discovery board.

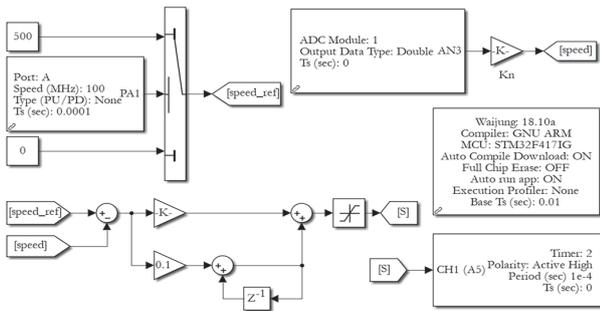


Fig. 16. Simulink diagram of the PI controller using the Waijung Blockset

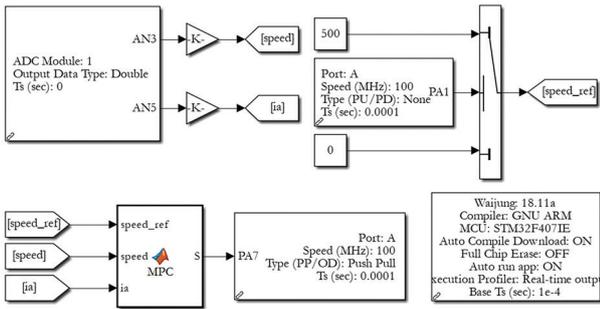


Fig. 17. Simulink diagram of the MPC controller using the Waijung Blockset.

Fig. 18 is the real speed step increment with the PI controller. Fig. 19 is the real speed step decrement with the PI controller. The real speed step increment with the MPC controller is depicted in Fig. 20. Finally, the real speed step decrement with the MPC controller is illustrated in Fig. 21.

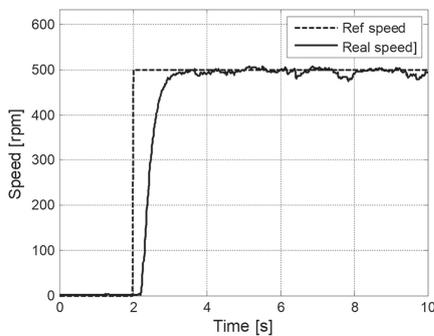


Fig. 18. Real speed step increment with the PI controller

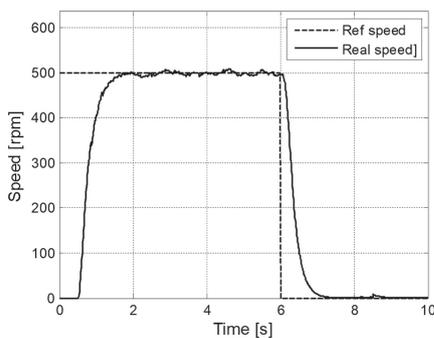


Fig. 19. Real speed step decrement with the PI controller

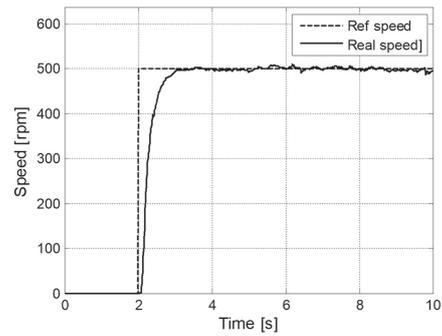


Fig. 20. Real speed step increment with the MPC controller

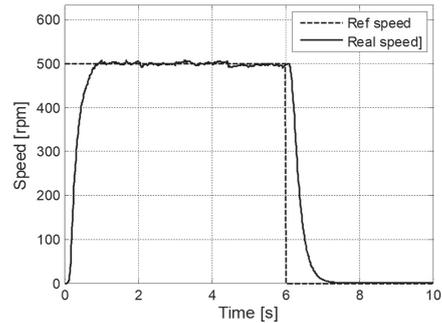


Fig. 21. Real speed step decrement with the MPC controller

The data of the motor speed steps in Figs. 19 and 21 are used to investigate the performances of the PI and MPC controllers. Table 2 is a comparison between the performances of the real PI and MPC controllers. The rise time, settling time and overshoot are considered. Compared to the PI controller, the use of the MPC controller can result in all the reduction of the rise time, settling time and overshoot. Therefore, the MPC controller can outperform the PI controller.

Table 2. Comparison between the performances of the PI and MPC controllers

| Controller | Rise Time (s) | Settling Time (s) | Overshoot (%) |
|----------------|---------------|-------------------|---------------|
| PI Controller | 0.5599 | 2.2675 | 1.4714 |
| MPC Controller | 0.4340 | 2.1093 | 1.2005 |

6. CONCLUSION

This study shows that FCS-MPC can be successfully applied to the speed control of brushed DC motors. This control algorithm does not require the use of PWM techniques and conventional PI controllers. The procedure for developing the MPC controller needs the model of the motor. The motor parameters can be conveniently estimated using the measured data and the MATLAB System Identification Toolbox. Finally, with the high performance of the whole system has been given by the proposed control method, the FSC-MPC method can be seen as the most powerful model-based control approach. The future work of this research is to deploy effective MPC controllers for other types of controlled objects such as popular power electronic converters and electric drives.

7. REFERENCES

- [1] M. Li, Y. Ma, "Parameter Identification of DC Motor based on Compound Least Square Method", Proceedings of the IEEE 5th Information Technology and Mechatronics Engineering Conference, Chongqing, China, 12-14 June 2020, pp. 1108-1111.
- [2] Rubaai, R. Kotaru, "Online identification and control of a DC motor using learning adaptation of neural networks", IEEE Transactions on Industry Applications, Vol. 36, No. 3, 2000, pp. 935-942.
- [3] H. Zhou, "DC Servo Motor PID Control in Mobile Robots with Embedded DSP", Proceedings of the International Conference on Intelligent Computation Technology and Automation, Changsha, China, 20-22 October 2008, pp. 332-336.
- [4] J. J. Muoz-Cesar, E. A. Merchan-Cruz, L. H. Hernandez-Gomez, E. Guerrero-Guadarrama, A. Jimenez-Ledes, "Speed Control of a DC Brush Motor with Conventional PID and Fuzzy PI Controllers", Proceedings of the Electronics, Robotics and Automotive Mechanics Conference, Cuernavaca, Mexico, 2008, pp. 344-349.
- [5] M. S. Qureshi, P. Swarnkar, S. Gupta, "Assessment of DC servo motor with sliding mode control approach", Proceedings of the IEEE First International Conference on Control, Measurement and Instrumentation, Kolkata, India, 8-10 January 2016, pp. 351-355.
- [6] H.-P. Ren, R. Zhou, J. Li, "Adaptive backstepping sliding mode tracking control for DC motor servo system", Proceedings of the Chinese Automation Congress, Jinan, China, 20-22 October 2017, pp. 5090-5095.
- [7] Munadi, M. A. Akbar, "Simulation of fuzzy logic control for DC servo motor using Arduino based on MATLAB/Simulink", Proceedings of the International Conference on Intelligent Autonomous Agents, Networks and Systems, 2014, pp. 42-46.
- [8] J. O. Jang, "A deadzone compensator of a DC motor system using fuzzy logic control", IEEE Transactions on Systems, Man, and Cybernetics, Part C, Vol. 31, no. 1, 2001, pp. 42-48.
- [9] M. A. Rahman, M. A. Hoque, "On-Line Self-Tuning ANN-Based Speed Control of a PM DC Motor", IEEE Transactions on Mechatronics, Vol. 2, No. 3, 1997, pp. 169-178.
- [10] Z. Runjing, Y. Waiting, Zhangfei, "Application of NN-PI Controller in Direct Current Motor Servo System", Proceedings of the 8th International Conference on Electronic Measurement and Instruments, 2007, pp. 468-471.
- [11] J. Pongfai, W. Assawinchaichote, "Self-tuning PID parameters using NN-GA for brush DC motor control system", Proceedings of the 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, Phuket, Thailand, 27-30 June 2017, pp. 111-114.
- [12] S. D. Sahputro, F. Fadilah, N. A. Wicaksono, F. Yuisivar, "Design and implementation of adaptive PID controller for speed control of DC motor", Proceedings of the 15th International Conference on Quality in Research (QiR): International Symposium on Electrical and Computer Engineering, Nusa Dua, Bali, Indonesia, 24-27 July 2017, pp. 179-183.
- [13] E. Afjei, A. Nadian Ghomsheh, A. Karami, "Sensorless speed/position control of brushed DC motor", Proceedings of the International Aegean Conference on Electrical Machines and Power Electronics, Bodrum, Turkey, 10-12 September 2007.
- [14] R. M. Ramli, N. Mikami, H. Takahashi, "Adaptive filters for rotational speed estimation of a sensorless DC motor with brushes", Proceedings of the 10th International Conference on Information Science, Signal Processing, and their Applications, Kuala Lumpur, Malaysia, 10-13 May 2010, pp. 562-565.
- [15] D. Ertl, L. Weber, "A Method for Real-Time Sensorless Speed Control of Brushed DC Motors in Cost Constrained Systems", Proceedings of the 2nd Global Power, Energy and Communication Conference, Izmir, Turkey, 20-23 October 2020, pp. 152-157.
- [16] S. Kouro, P. Cortes, R. Vargas, U. Ammann, J. Rodriguez, "Model Predictive Control—A Simple and Powerful Method to Control Power Converters", IEEE Transactions on Industrial Electronics, Vol. 56, No. 6, pp. 1826-1838.

- [17] J. Baek, S. Kim, S. Kwak, "Predictive control method for load current of single-phase voltage source inverters", Proceedings of the IEEE Applied Power Electronics Conference and Exposition, Charlotte, NC, USA, 15-19 March 2015.
- [18] R. E. Pérez-Guzmán, M. Rivera, N. Vicencio, P. Wheeler, "Model-based Predictive Control in Three-Phase Inverters", Proceedings of the IEEE International Conference on Industrial Technology, Buenos Aires, Argentina, 26-28 February 2020.
- [19] MATLAB System Identification Toolbox, <https://www.mathworks.com/products/sysid.html> (accessed: 2023)
- [20] Waijung Blockset, <https://waijung1.aimagin.com> (accessed: 2023)