

An Artificial Neural Network-Based Model Predictive Control Of Cascaded H-Bridge Multilevel Inverter

Bao Binh Pho , Minh Hoan Tran , Trong Minh Tran , Phuong Vu 

* School of Electrical & Electronic, Hanoi University of Science and Technology, Hai Ba Trung District, Hanoi 11600, Vietnam

* Hanoi University of Civil Engineering, Hai Ba Trung District, Hanoi 11600, Vietnam

binhpb@nuce.edu.vn, hoan230999@gmail.com, minh.tran.mail@gmail.com, phuong.vuhoang@hust.edu.vn,

‡Corresponding Author: Phuong Vu, School of Electrical & Electronic, Hanoi University of Science and Technology, Hai Ba Trung District, Hanoi 11600, Vietnam, phuong.vuhoang@hust.edu.vn

Received: 30.05.2022 Accepted:25.07.2022

Abstract- In recent years, there has been an increasing interest in using Cascaded H-Bridge (CHB) for medium and high-power applications. Model predictive control (MPC) strategy has emerged as a promising option for controlling CHB with considerable advantages. However, the most significant disadvantage of MPC is the exponential increase of computational burden to solve the optimization, leading to an unacceptable amount of computing resources. Therefore, to overcome these difficulties, the ANN-MPC approach for CHB is proposed in this paper. Firstly, the multistep MPC controller is designed and operated in simulation environment to generate the data required for training. Secondly, after being successfully trained, the neural network can be used to control the system without the need for MPC to avoid the heavy-duty computing problem. The performance of ANN-MPC is evaluated and compared to that of conventional multistep MPC. Finally, a FPGA based ANN-MPC controller employing the trained ANN is designed to control the experimental system with three-phase five-level CHB with LC filter and linear loads. Both simulation and experimental results verified the excellent control performance of the proposed ANN-MPC.

Keywords: Model Predictive Control (MPC), Cascade H-Bridge (CHB), Artificial Neural Network (ANN), Field-Programmable Gate Array (FPGA)

1. Introduction

Nowadays, multilevel inverters (MLI) are widely used in high-power applications, especially for industrial motor drive, high voltage transmission, and renewable energy integration [1][2]. There are numerous MLI topologies such as Flying capacitor (FC) [3], Neutral point clamped (NPC) [4], Cascaded H-Bridge (CHB) [5][6]. According to [7], the CHB has the best performance in a lot of criteria, including the voltage and current THD quality, lower switching power losses, and electromagnetic interference (EMI). Especially, CHB has also emerged as a prominent inverter for driving IM systems due to its high degree of modularity, allowing to increase the voltage or power level easily by increasing the number of H-bridge [8][9]. Even though the CHB topology (Fig. 1) are available in the literature for several decades, many control strategies are still under research to improve the steady-state performance of CHB. With the conventional method, input reference signals are tracked by PI controllers and then transformed into high frequency switching signals for semiconductor valves using the modulation stage (PWM, SVM) [1], [3]. Among the modern

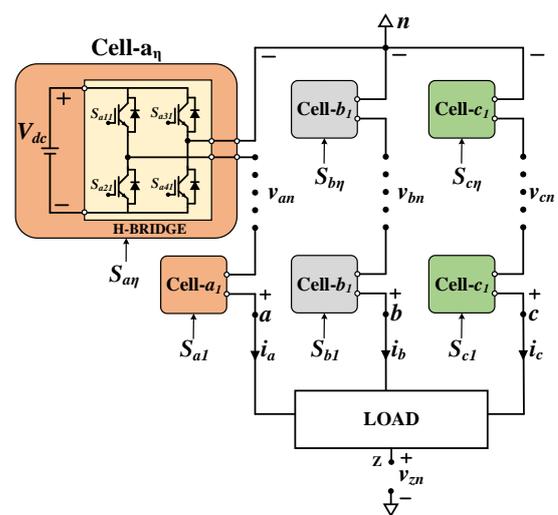


Fig. 1. CHB system structure

control methods, MPC has emerged as a promising option and has shown significant advancement over traditional

ones [10]. Due to high computation burden, MPC firstly evolved in process industry. Together with the development of microprocessor such as DPS or FPGA, the MPC has

extended its application for nonlinear system with fast dynamic behavior [11]. Via applying FPGA as a controller

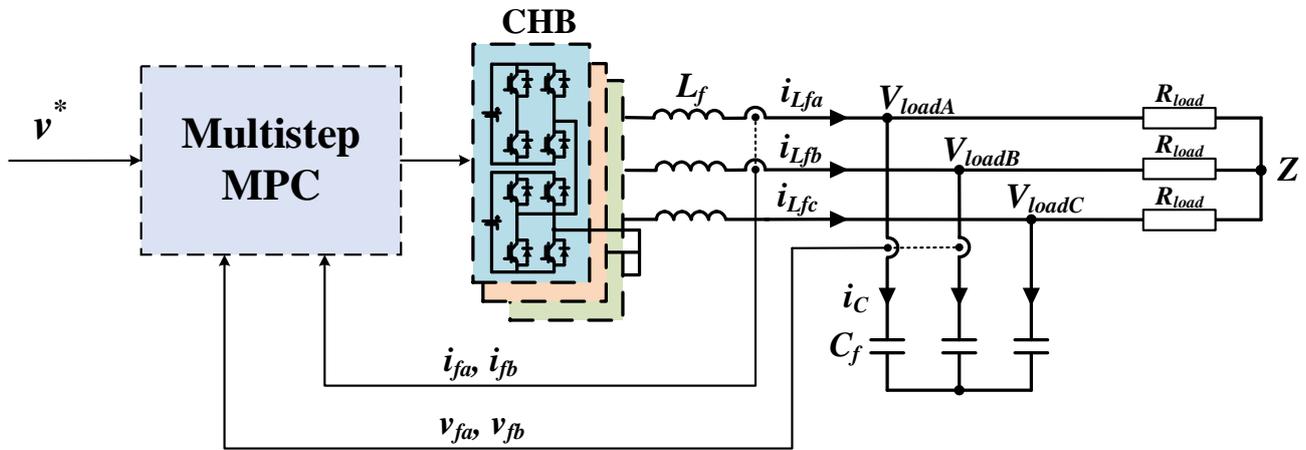


Fig. 2. Control scheme of the conventional MPC

without PWM algorithm, the FCS-MPC method can be calculated in very short time, so that the response of the inner current is considerably enhanced in comparison with the PID controller [12-14].

Multistep MPC and its feasibility for practical adoption have received much attention in the field of power electronics. By using the system model, multistep MPC will predict the future behavior of the system variables, and then generate the optimal switching signals by calculating and optimizing the cost function. Multistep MPC have many attractive features, including handling of multi-variable, nonlinear systems and have fast dynamic response [15]. The most distinctive feature of MPC is that system constraints and additional control objectives can be considered easily in the problem formulation [16]. Furthermore, it is recently shown that multistep MPC has better steady-state performance in comparison with single-step MPC [17]. However, the practical adoption of multistep MPC has two major challenges: the heavy-duty computational burden and the dependence on the quality of the predictive model [18][19]. When increasing the length of the prediction horizon or the number of H-bridge cells, the quantity of control-input combinations will increase exponentially, hindering the real-time implementation of MPC even to the modern powerful microcontrollers. Therefore, MPC applications are usually limited to low-order systems, low voltage levels, or low switching frequencies. Nonetheless, this goes against the modern demand for higher voltage level converters or higher switching frequencies.

In recent times, artificial neural network (ANN) has become an excellent tool to solve input-output mapping problems [20][21]. The purpose of employing ANN is to predict an output based on the network's training that

involves inputs and targets [22]. Recently, many studies were conducted to improve the computational problem of MPC mentioned above. Reference [23] used ANN to optimize the weighting parameter of MPC. Paper [24] applied ANN-MPC for a two-level voltage source inverter in the simulation environment. It has been seen that there is a limited amount of research on ANN-MPC for CHB with high-order systems. Therefore, this paper investigates the application of ANN-MPC for five-level CHB with LC filter and linear loads. The proposed control method will be validated by both simulation and experiment results. First, the multistep MPC controller is designed and operated in MATLAB/SIMULINK environment. The inputs (load voltage, load current, ...) and the output (optimal switching states) of MPC controllers are collected as the required training data for ANN-MPC. After that, a neural network is created and trained offline with the recorded data. At last, the trained ANN is then designed in FPGA model Zybo-Z7 to control the actual five-level CHB with an LC filter. By doing this, the proposed ANN-MPC controller can offer a comparable control performance to the conventional multistep MPC while using a simple mathematical expression of ANN. This paper theoretically discusses the basic concept of ANN-MPC, training phase, online operation phase, and used ANN structure.

2. Multistep MPC Controller

The simplified block diagram of the multistep MPC control scheme for five-level CHB with LC filter and R load is shown in Fig. 2.

Equation (1) represents the system model in the form of state-space representation in continuous-time:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}'\mathbf{x} + \mathbf{B}'\mathbf{u} \\ \mathbf{y} = \mathbf{C}\mathbf{x} \end{cases} \quad (1)$$

$\mathbf{x} = [i_a \ i_b \ v_a \ v_b]^T \in \mathbb{R}^{4 \times 1}$ is the system state vector with i_{ab}, v_{ab} are the instantaneous measurements of current and voltage of phase A, B filters. $\mathbf{u} = [u_a \ u_b \ u_c]^T \in \mathbb{R}^{3 \times 1}$ is the control input vector, each element $u \in \{-2, -1, 0, 1, 2\}$ as the voltage level of five-level CHB. $\mathbf{y} = [i_a \ i_b \ v_a \ v_b]^T \in \mathbb{R}^{4 \times 1}$ is the output vector. Matrix $\mathbf{C} = \mathbf{I}_4$ is the identity matrix of size four. Matrix \mathbf{A}', \mathbf{B}' are presented as:

$$\mathbf{A}' = \begin{bmatrix} 0 & 0 & -\frac{1}{L} & 0 \\ 0 & 0 & 0 & -\frac{1}{L} \\ \frac{1}{C} & 0 & -\frac{1}{RC} & 0 \\ 0 & \frac{1}{C} & 0 & -\frac{1}{RC} \end{bmatrix}, \mathbf{B}' = \frac{V_{DC}}{3L} \begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

After applying the Forward-Euler discretization with a sampling period of T_s to equation (1), the discrete-time representation model can be written as:

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \\ \mathbf{y}(k+1) = \mathbf{C}\mathbf{x}(k+1) \end{cases} \quad (2)$$

With $\mathbf{A} = \mathbf{I}_4 + T_s\mathbf{A}' \in \mathbb{R}^{4 \times 4}$ and $\mathbf{B} = T_s\mathbf{B}' \in \mathbb{R}^{4 \times 3}$, T_s is the sampling period and \mathbf{I} is the identity matrix. Thus, with each sampling instant k , the system behavior at the next sampling instant $k+1$ can be predicted by applying equation (2) for each admissible control-input $\mathbf{u}(k)$ and continue to the prediction horizon N

After that, the cost function equation is formulated to express the control objectives. By means of solving a single optimization problem, multiple control objectives are obtained such as reference tracking, common-mode voltage (CMV) minimization, and optimal switching effort. These objectives can be succeeded through adding the weighting factors, which adjust the trade-off between the control objectives. In this job, three control objectives are set as:

1. Voltage reference tracking
2. CMV minimization
3. Reduce the step voltage level change (reduce du/dt).

Targets (1) and (2) are achieved by solving the cost function, objective (3) is not presented in the cost function but is executed in the control-input vector searching process. The cost function over a prediction horizon N is written as:

$$J_N(k) = \sum_{l=k}^{k+N-1} \left(\|\mathbf{y}(\ell+1) - \mathbf{y}^*(\ell+1)\|_2^2 + \sigma_u \|\mathbf{u}(\ell) - \mathbf{u}^*(\ell)\|_2^2 \right) \quad (3)$$

Where $\mathbf{u}(\ell)$ is the control-input candidate that generates the output prediction $\mathbf{y}(\ell+1)$ with $\ell = \{k, k+1, \dots, k+N-1\}$ and N is the prediction horizon, σ_u is the weighting factor for CMV minimization objective. Vector $\mathbf{y}^*(\ell+1)$ is the reference vector and $\mathbf{u}^*(\ell)$ is the control-input reference designed to have zero CMV and can be calculated by the system model.

The optimal control-input vector \mathbf{U}_{opt} is the control-input vector that minimizes $J_N(k)$ can be expressed as:

$$\mathbf{U}_{opt}(k) = [\mathbf{u}(k)^T \ \mathbf{u}(k+1)^T \ \dots \ \mathbf{u}(k+N-1)^T]^T \in \mathbb{R}^{3N \times 1}$$

After that, only the control-input $\mathbf{u}(k)^T$ is applied to control the converter.

Note that, to find the minimum $J_N(k)$ in equation (3), all possible control-input combinations must be evaluated. The computational burden of each sampling period is expressed as:

$$N_{MPC} = (2\eta + 1)^{3N} (n_p + n_c) \quad (4)$$

In equation (4), η is the number of H-bridge cells in each phase, N is the prediction horizon, n_p is the number of calculations of the prediction system model, n_c is the number of calculations of the cost function. Thus, the computational burden increases exponentially when increasing the number of cells or the prediction horizon. Besides, n_p and n_c will increase when using a more complex system model, resulting in more calculation problems. Therefore, multistep MPC applications are usually limited to low-order systems, low voltage levels, or short prediction horizons.

There has been some undergoing research to overcome this difficulty, aiming to improve a smarter searching algorithm to reduce unnecessary computation. For instance, one of these algorithms is Sphere Decoding Algorithm (SDA), which is discussed in paper in detail [25][26]. Nevertheless, this approach does not extensively solve the problem. So

that, this study investigates the ANN-MPC approach, using the ANN trained by the data collected from the multistep MPC. By doing this, the proposed ANN-MPC not only offers excellent control performance but also has more simply mathematical representation of ANN. This approach can potentially become the solution for the practical adoption of this sophisticated multistep MPC.

3. Proposed ANN-MPC Controller

With the MPC method, in each sampling cycle, the system state feedback parameters and the corresponding optimal voltage control vectors would be obtained to meet the voltage requirements. These parameters are collected to make standard input and output data to train the neural network.

3.1. Structure of ANN

ANN includes an input layer, a hidden layer, and an output layer; each layer consists of several neurons [27]. ANN utilizes composed computational neurons in layers, which

is interconnected to every hub dependent on weight factors [28]. The used structure of ANN is depicted in Fig.3. The structure has been shown to be accurate for MPC applications using two-layer ANN [29].

- Input Layer: Using the same input as MPC, to reduce the computational burden, these variables are transformed into $\alpha\beta$ -framework (six input elements including v^*, v, i in $\alpha\beta$ -framework)

$$\mathbf{I} = [I_1 \quad I_2 \quad \dots \quad I_6]^T \in \mathbb{R}^{6 \times 1} \tag{5}$$

The ANN-MPC controller, in practice, first normalizes each input element into a suitable range. [-1;1] assisting in the distribution of data in a more efficient manner using equation (6):

$$I_i = 2 \left(\frac{I_i - \min(I_i)}{\max(I_i) - \min(I_i)} \right) - 1 \tag{6}$$

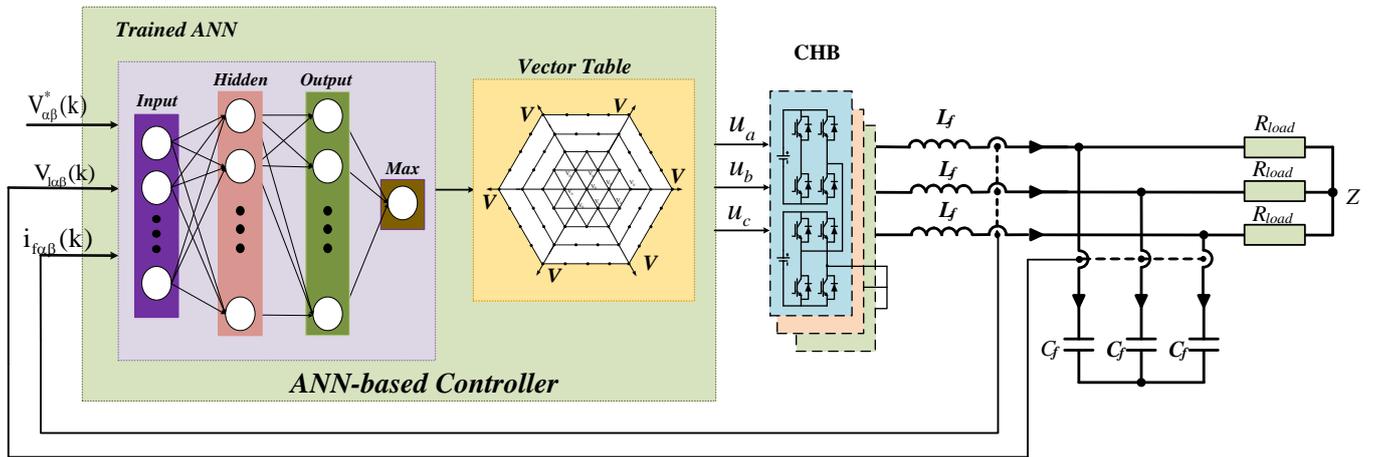


Fig. 3. The control scheme of the proposed ANN-MPC controller

- Hidden Layer: The number of nodes in the hidden layer can be self-selected. Currently, there is no research or regulation to determine the exact number of hidden layers. In this study, it is proposed to use the least number of hidden layer nodes as possible to minimize unnecessary computation volume while maintaining system accuracy. Thereby choosing the number of hidden layers to be 11.

$$\mathbf{H} = [H_1 \quad H_2 \quad \dots \quad H_{11}]^T \in \mathbb{R}^{11 \times 1} \tag{7}$$

- Output Layer: The circuit topology to be controlled is a 3-phase 5-level CHB, so there will be 61 distinct voltage vectors. The ANN architecture used is a one-vs-rest architecture, which means each voltage vector will correspond to 1 output node. Therefore, the output layer vector is expressed as:

$$\mathbf{O} = [O_1 \quad O_2 \quad \dots \quad O_{61}]^T \in \mathbb{R}^{61 \times 1} \tag{8}$$

O_n corresponds to the probability of selecting that vector. Finally, the O_n with the largest value is selected to control the system.

Next, the nodes in the neural network are related to each other through the following formula:

$$\mathbf{H} = f^{(1)}(\mathbf{W}^{(1)T} \mathbf{I} + \mathbf{b}^{(1)}) \in \mathbb{R}^{11 \times 1} \tag{9}$$

$$\mathbf{O} = f^{(2)}(\mathbf{W}^{(2)T} \mathbf{H} + \mathbf{b}^{(2)}) \in \mathbb{R}^{61 \times 1}$$

$\mathbf{W}^{(1)} \in \mathbb{R}^{6 \times 11}$, $\mathbf{b}^{(1)} \in \mathbb{R}^{11 \times 1}$ are the weight matrix and bias vector of the Hidden Layer and $\mathbf{W}^{(2)} \in \mathbb{R}^{11 \times 61}$, $\mathbf{b}^{(2)} \in \mathbb{R}^{61 \times 1}$ are the weight matrix and bias vector of the Output Layer.

Functions $f^{(1)}$ and $f^{(2)}$ are the activation functions for the Hidden Layer and Output Layer respectively. Generally, the commonly used function $f^{(1)}$ is a zero-centered function. In this study, the function *tansig* is selected and expressed as equation (8):

$$f(z) = \frac{2}{1 + e^{-2z}} - 1 \quad f(z) \in [-1; 1] \quad (10)$$

The activation function of the output layer $f^{(2)}$ is one of the most crucial aspects to consider, setting the accuracy of the neural network. For multiclass classification problems, *Softmax function* is commonly used [17], which is shown as equation (11):

$$f(z_i) = \frac{e^{z_i}}{\sum_{j=1}^n e^{z_j}} \quad f(z_i) \in [0, 1] \quad (11)$$

In which, n is the number of nodes in the Output Layer. After creating the architecture, the neural network will be trained offline from the data collected from the MPC

algorithm. The goal is to find weight matrices $\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, \mathbf{b}^{(1)}, \mathbf{b}^{(2)}$ such that the optimal state vector found from the neural network is most similar to that of MPC.

Thus, assume that the number of used nodes in input layer, hidden layer and output layers are l, m, n . Consequently, the dimension of the matrixes and vectors of the neural network are demonstrated in Table 1.

Table 1. Dimension of vectors and matrixes

Vector/ Matrix	Dimensionality (row x column)
\mathbf{I}	$1 \times l$
\mathbf{H}	$1 \times m$
\mathbf{O}	$1 \times n$
$\mathbf{W}^{[2]}$	$m \times l$
$\mathbf{W}^{[3]}$	$n \times m$

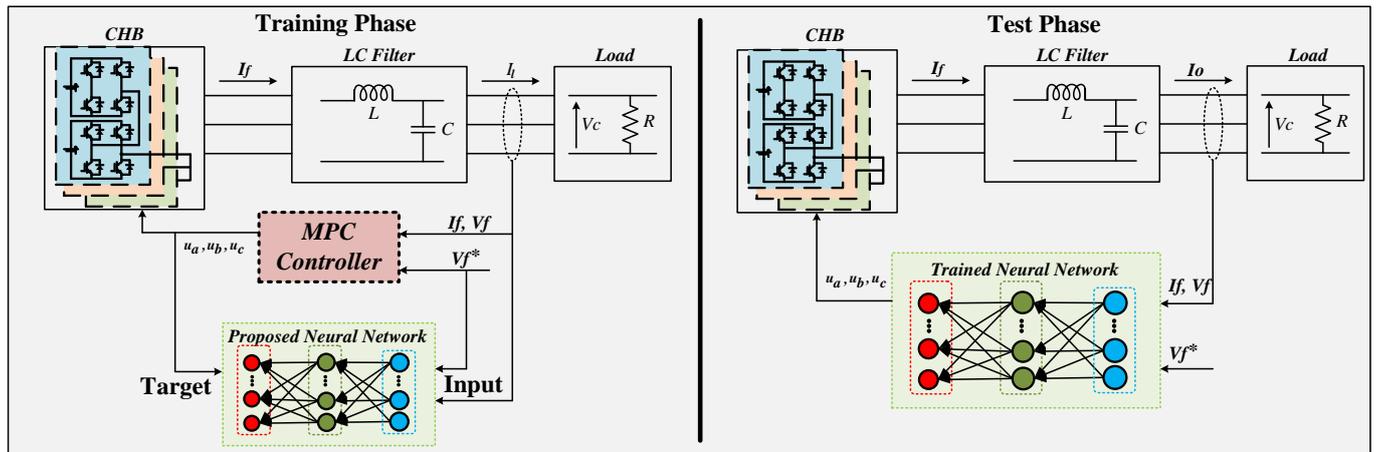


Fig. 4. Collecting data and training process of the proposed ANN-MPC controller

Then, the computational volume of the neural network can be calculated by equation (12):

$$N_{ANN-MPC} = 2lm + 2mn + \varepsilon_1 m + \varepsilon_2 n \quad (12)$$

ε_1 is the number of calculation of activation function $f^{(1)}$, ε_2 is the number of calculation of activation function $f^{(2)}$. It can be seen that the computational volume of ANN with equation (12) is significantly lower than that of MPC according to equation (4). Furthermore, unlike the conventional MPC's exponential growth in computational burden, the ANN-MPC approach's computational cost increases only minimally with growing system complexity.

As a result, ANN-MPC is well suited to managing complex systems, with control performance comparable to traditional MPC but a significantly lower computing burden.

3.2. Collecting Data and Training Process

Fig. 4 describes the collecting data and offline training process of the ANN-MPC. Training data is collected with 8 different instances of output voltage reference ($V_{ref} = 80, 100, 120, 140, 160, 180, 200, 220$).

The data is collected with the same circuit values, load values, and sampling times. With the 8 cases above, the total number of data sets collected are 31983. In this study, the

data set collected from MPC is divided into 3 parts: the Training Set occupies 70%, the Validation Set and Test Set occupy 15% each. The accuracy of the Neural network will be evaluated by the error value of the Test set.

In this case, the training function “*trainscg*” is used with the Scaled Conjugate Gradient (SCG) update algorithm. This is a very popular algorithm, capable of fast training and widely used in neural network applications [30].

After offline training, the system's neural network achieved 81.6% accuracy on Test Set. The error value of Training Set and Validation Set is very similar at the end of training, and training ends early at epochs 863 with a Validation Error value of 0.00784 indicating that the Neural network has been updated to the minimum point.

Finally, to control the valve, the state vector found is converted into switching state according to the switching state selection strategy in Table 2.

Table 2. Switching state selection strategy

S_x	$S_{x1}(S_{x1,1}; S_{x1,3})$	$S_{x2}(S_{x2,1}; S_{x2,3})$
+2	1(1; 0)	1(1; 0)
+1	1(1; 0)	0(0; 0)
0	0(0; 0)	0(0; 0)
-1	0(0; 0)	0(0; 0)
-2	0(0; 0)	0(0; 0)

In Table 2, variable $x \in \{A, B, C\}$ represents the phases, S_x represents the voltage level of phase x and the index notation is the position-number of the H-bridge, $(S_{x1,1}; S_{x1,3})$ are the switching states of valve 1 and 3 of the H-bridge.

4. Results

4.1. Simulation Results

The control performance of the proposed ANN-MPC will be tested on MATLAB/Simulink environment with the following parameters in Table 3.

Table 3. System parameters

Sampling frequency: f_s	10kHz
Weighting Factor: σ_u	1e-5
Prediction Horizon MPC	2
H-Bride Cells per phase η	2

DC voltage V_{dc}	100V
Voltage Frequency	50Hz
Filter Capacitance	30uF
Filter Inductance	3mH
Load	50Ω

❖ **Simulation Scenario:**

- First, the controller is tested with the output voltage reference available in the data set.
 - 0-0.03 (s): $V_{ref} = 200V$
 - 0.03-0.06 (s): $V_{ref} = 160V$

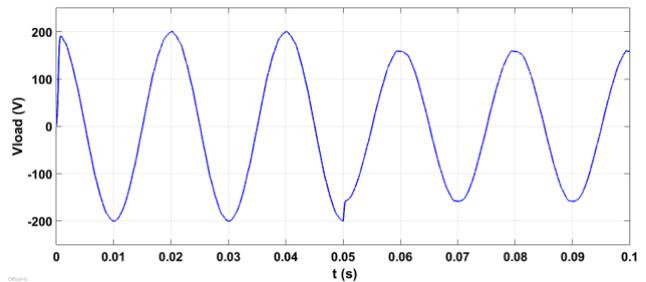


Fig. 5. Load voltage with proposed ANN-MPC controller

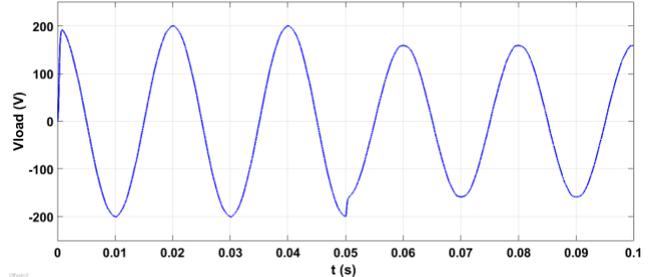


Fig. 6. Load voltage with conventional MPC

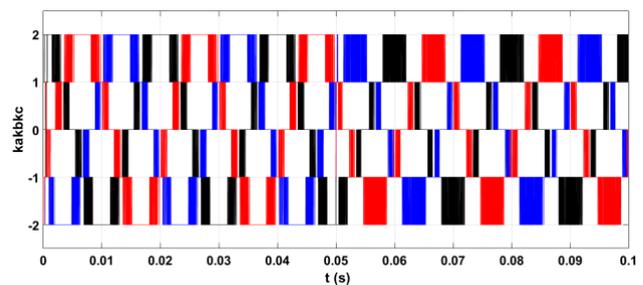


Fig. 7. Switching state with proposed ANN-MPC controller

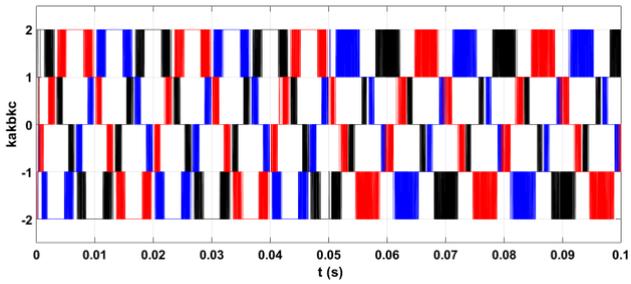


Fig. 8. Switching state with conventional MPC

Fig. 5 shows the load voltage with proposed ANN-MPC controller and Fig. 6 is that with conventional MPC. Switching state with proposed ANN-MPC and conventional MPC are described in Fig. 7 and Fig. 8. The control performance of the ANN-MPC controller is approximated and comparable with the MPC controller, with THD of ANN-MPC and MPC being 0.66% and 0.41%, respectively. The proposed ANN-MPC controller can effectively track the reference value under steady-state condition. Belonging the dynamic condition, the response time under step-up change in the amplitude of voltage reference value is similar between two controllers, approximately 0.1ms.

- Second, the controller is tested with the output voltage reference which is not available in the data set.
 - 0-0.03 (s): $V_{ref} = 150V$
 - 0.03-0.06 (s): $V_{ref} = 170V$

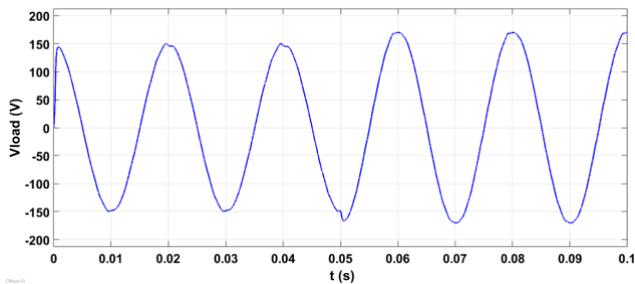


Fig. 9. Load voltage with proposed ANN-MPC

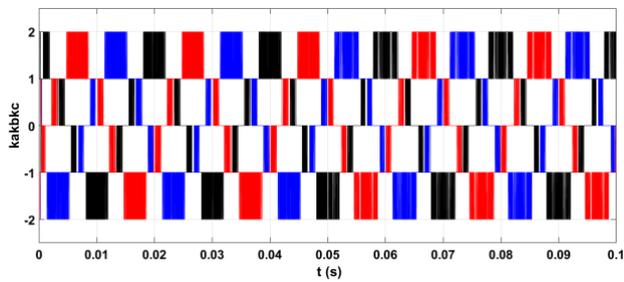


Fig. 10. Switching state with proposed ANN-MPC

Fig. 9 and Fig. 10 demonstrate the control performance of the proposed ANN-MPC controller with the voltage reference not available in the data set (150 V, 170 V). The proposed controller, as observed, can effectively track the voltage reference in both steady-state and dynamic

condition with the THD value is 0.91% and fast dynamic response.

4.2. Experiment Results

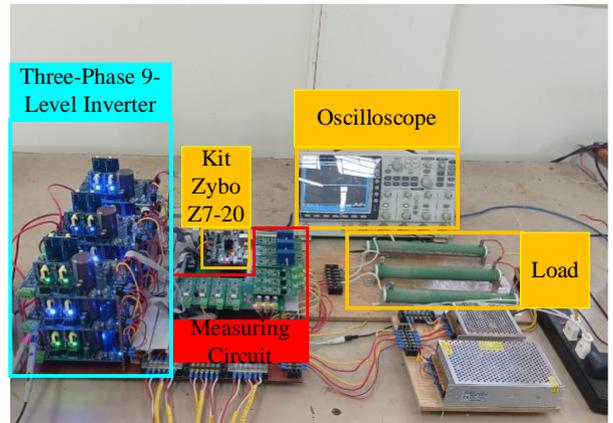


Fig. 11. The experimental setup

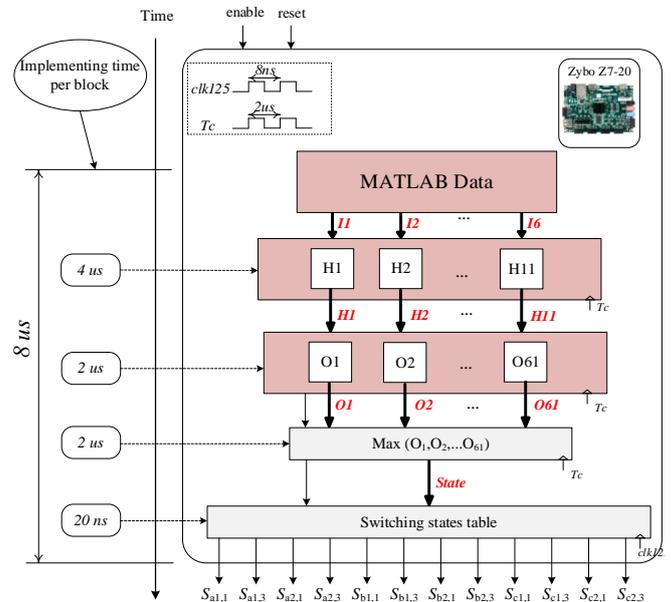


Fig. 12. Block diagram of FPGA-based controller

The proposed ANN-MPC controller is designed on a Zybo Z7-20 FPGA kit with a 5-level CHB system and R load as expressed in Fig. 11.

Fig. 12 illustrates the block diagram of the proposed FPGA-based ANN-MPC controller. The controller is designed with pipeline-based architecture by a calculation frequency of 500kHz and a sampling period of 100us (switching frequency is 10kHz). The total calculation time from the input layer to the output switching state is relatively small, evaluated at 8ms.

Table 4. FPGA resource utilization

	Used	Available	Percentage
LUT	28799	53200	54%
Flip-Flop	1243	106400	1.2%
DSPs	162	220	74%

The FPGA resource utilization of kit Zybo Z720 is represented in Table 3. The proposed controller used 54% of the available LUT, 74% of the available DSP slices and

a small number of Flip-flop. This is an acceptable amount of calculation resource considering the excellent control performance.

The experiment results of the experimental setup are shown in Fig.13 and Fig.14. With the output voltage reference $V_{ref} = 200V$, it is observed that the voltage level is similar to the simulation results in Fig. 5. The load current measured at 4A has a sinusoidal waveform, indicating the robustness of the steady-state performance.

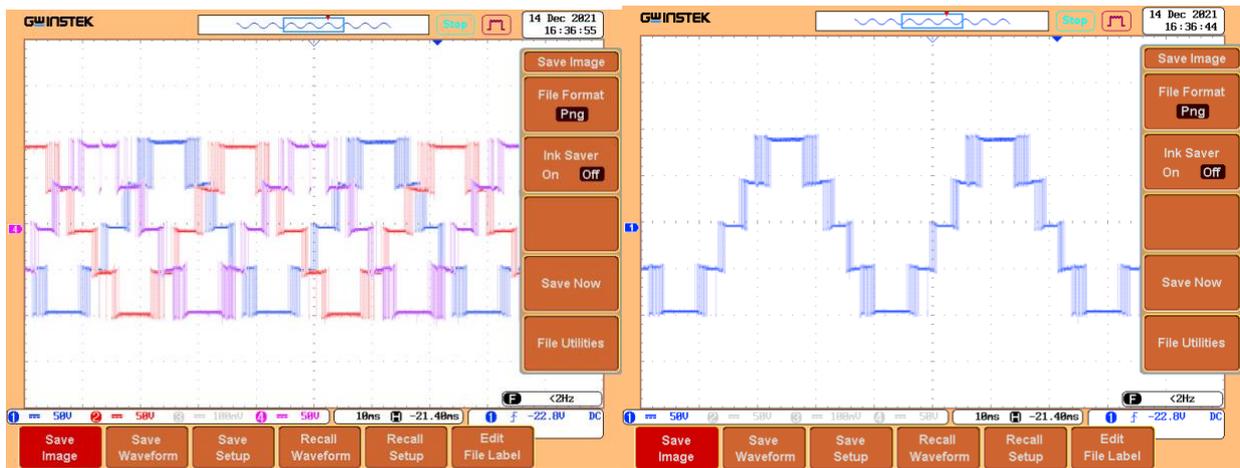


Fig. 13. Voltage level in 3 phases and 1 phase

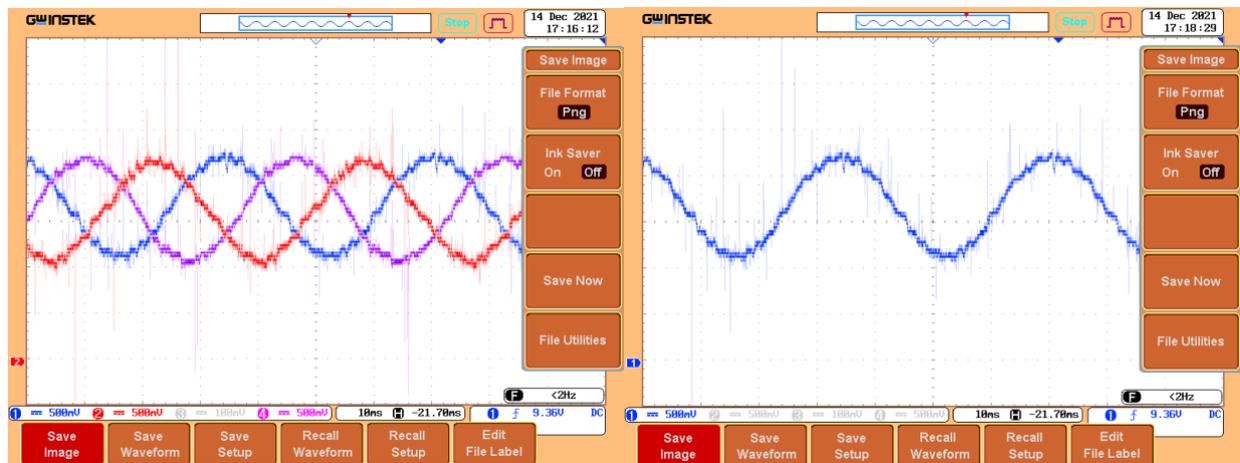


Fig. 14. Load current in 3 phases and 1 phase

5. Conclusion

In this paper, an artificial neural network-based model predictive control (ANN-MPC) strategy for 5-level CHB is proposed. Both simulation and experimental results not only have proved the robustness of control performance but also obtained the feasibility and applicability of the control method in practice. The controller can retain the superior control characteristics of the MPC method while employing an acceptable amount of hardware resources and simplifying the computation with a trained ANN. Therefore, this controller can be applied for many

applications with higher-order systems and different types of loads.

Acknowledgment

This research is funded by Hanoi University of Civil Engineering (HUCE) under grant number 21-2022/KHXD-TĐ.

References

- [1] S. Kouro, M. Malinowski, K. Gopakumar, Josep Pou, Leopoldo G. Franquelo, Bin Wu, José Rodríguez, Marcelo A. Pérez, Jose I. Leon (2010), "Recent advances and industrial applications of multilevel converters," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 8, pp. 2553–2580.
- [2] M. Roknuzzaman and S.I. Hamasaki (2019), "Control of Full Bridge type Modular Multilevel Converter for AC/AC Conversion with Grid Connection," 6th IEEE International Conference on Smart Grid, icSmartGrids 2018, pp. 64–69, IEEE.
- [3] T.A. Meynard, H. Foch, P. Thomas, J. Courault, R. Jakob, M. Nahrstaedt (2002), "Multicell converters: Basic concepts and industry applications," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 5, pp. 955–964.
- [4] J. Rodriguez, S. Bernet, P.K. Steimer, I.E. Lizama (2010), "A survey on neutral-point-clamped inverters," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 7, pp. 2219–2230.
- [5] M. Malinowski, K. Gopakumar, J. Rodriguez, M.A. Pérez. (2010), "A survey on cascaded multilevel inverters," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 7, pp. 2197–2206.
- [6] H. Benbouhenni and O. Maurice (2019), "Application of Five-Level NPC inverter in DPC- ANN of Doubly Fed Induction Generator for Wind Power Generation Systems," *International Journal of Smart Grid - ijSmartGrid*, vol. 3, no. 3.
- [7] R. Mali, N. Adam, A. Satpaise, A.P. Vaidya (2019), "Performance Comparison of Two Level Inverter with Classical Multilevel Inverter Topologies," *Proceedings of 2019 3rd IEEE International Conference on Electrical, Computer and Communication Technologies, ICECCT 2019*, pp. 1–7, IEEE.
- [8] H. Truong, C. Mai, C. Nguyen, P. Vu (2021), "Modified space vector modulation for cascaded H-bridge multilevel inverter with open-circuit power cells," *Journal of Electrical and Computer Engineering*, vol. 2021.
- [9] C.M. Van, T.N. Xuan, P.V. Hoang, M.T. Trong, S.P. Cong, L.N. Van (2019), "A Generalized space vector modulation for cascaded h-bridge multi-level inverter," *Proceedings of 2019 International Conference on System Science and Engineering, ICSSE 2019*, pp. 18–24, IEEE.
- [10] V.C. Mai, M.L. Nguyen, T.H. Vo, P.V. Hoang, T.M. Tran (2019), "Hardware in the loop simulation of predictive current control for im fed by multi-level cascaded h-bridge inverters," *2019 IEEE Vehicle Power and Propulsion Conference, VPPC 2019 - Proceedings*.
- [11] M. Van Chung, D.T. Anh, and P. Vu (2021), "A finite set-model predictive control based on FPGA platform for eleven-level cascaded H-Bridge inverter fed induction motor drive," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 12, no. 2, p. 845.
- [12] M. Perez, M. Vasquez, J. Rodriguez, and J. Pontt, "FPGA-based predictive current control of a three-phase active front end rectifier", in *Industrial Technology, 2009. ICIT 2009. IEEE International Conference on*, pp. 1–6, 2009.
- [13] J. Alvarez, Ó. Lopez, F. D. Freijedo and J. Doval-Gandoy, "Digital Parameterizable VHDL Module for Multilevel Multiphase Space Vector PWM", *IEEE Transactions on Industrial Electronics*, vol. 58, no. 9, pp. 3946–3957, 2011.
- [14] M. V. Chung, D.T. Anh, P. Vu, M.L. Nguyen (2020), "Hardware in the loop co-simulation of finite set-model predictive control using fpga for a three level CHB inverter," *International Journal of Power Electronics and Drive Systems*, vol. 11, no. 4, pp. 1719–1730.
- [15] C. Mai-Van, S. Duong-Minh, D. Tran-Huu, B. Pho-Bao, P. Vu (2021), "An improved method of model predictive current control for multilevel cascaded H-bridge inverters," *Journal of Electrical Engineering*, vol. 72, no. 1, pp. 1–11.
- [16] S. Kouro, P. Cortes, R. Vargas, U. Ammann, J. Rodriguez (2009), "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] T. Geyer and D.E. Quevedo (2014), "Performance of multistep finite control set model predictive control for power electronics," *IEEE Transactions on Power Electronics*, vol. 30, no. 3, pp. 1633–1644.
- [18] S. Vazquez, J. Rodriguez, M. Rivera, L.G. Franquelo, M. Norambuena (2017), "Model Predictive Control for Power Converters and Drives: Advances and Trends," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 2, pp. 935–947.
- [19] Jose Rodriguez and P. Cortes (2012), "Predictive control of power converters and electrical drives," *John Wiley & Sons, Ltd.*
- [20] K.J. Hunt, D. Sbarbaro, R. Zbikowski, P.J. Gawthrop (1992), "Neural networks for control systems-A survey," *Automatica*, vol. 28, no. 6, pp. 1083–1112.
- [21] D. Gueye, A. Ndiaye, and A. Diao (2020), "Adaptive Controller Based on Neural Network Artificial to Improve Three-phase Inverter Connected to the Grid," *9th International Conference on Renewable Energy Research and Applications, ICRERA 2020*, pp. 72–77.
- [22] A. Elhor and O. Soares (2022), "Grid-connected PV System with a Modified-Neural Network Control," *International Journal of Renewable Energy Research-IJRER*, vol. 12, no. 2, pp. 1013–1022.
- [23] T. Dragičević and M. Novak (2019), "Weighting Factor Design in Model Predictive Control of Power

- Electronic Converters: An Artificial Neural Network Approach,” *IEEE Transactions on Industrial Electronics*, vol. 66, no. 11, pp. 8870–8880, IEEE.
- [24] I.S. Mohamed, S. Rovetta, T.D. Do, T. Dragicevic, A.A.Z Diab (2019), “A neural-network-based model predictive control of three-phase inverter with an output LC Filter,” *IEEE Access*, vol. 7, pp. 124737–124749.
- [25] B.B. Pho, N.V. Cao, T.M. Hoan, P. Vu (2021), “Modified multistep model predictive control for three-phase induction motor drive system considering the common-mode voltage minimization,” *International Journal of Power Electronics and Drive Systems*, vol. 12, no. 4, pp. 2251–2260.
- [26] P. Vu, T.A. Do, and L. Nguyen (2021), “A Novel Multi-Step Model Predictive Control Design for Three-phase T-Type Inverter in Grid-Connected Mode,” *International Journal of Renewable Energy Research*, vol. 11, no. 4, pp. 1968–1976.
- [27] G.E. Jung, J.K. Baek, J. Liu, V.Q. Dao, M.C. Dinh, C.S. Kim, M.K. Lee, J.H. Bae (2021), “The Precision SOC Estimation for Fire Prevention of the EES Using ANN,” *10th IEEE International Conference on Renewable Energy Research and Applications, ICRERA 2021*, pp. 231–234, IEEE.
- [28] A. Saidi, A. Harrouz, I. Colak, K. Kayisli, R. Bayindir (2019), “Performance Enhancement of Hybrid Solar PV-Wind System Based on Fuzzy Power Management Strategy: A Case Study,” *7th International Conference on Smart Grid, icSmartGrid 2019*, pp. 126–131.
- [29] X. Yuan, G. Huang, and K. Shi (2020), “Improved Adaptive Path following Control System for Autonomous Vehicle in Different Velocities,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 8, pp. 3247–3256, IEEE.
- [30] M.F. Møller (1993), “A scaled conjugate gradient algorithm for fast supervised learning,” *Neural Networks*, vol. 6, no. 4, pp. 525–533.