Journal of Techniques, ISSN: 2708-8383, Vol. 8, No. 2, 2026, Pages 27-36
https://doi.org/10.51173/jt.v8i2.2983

JOURNAL OF TECHNIQUES

Journal homepage: http://journal.mtu.edu.iq

RESEARCH ARTICLE - ENGINEERING (MISCELLANEOUS)

# Artificial Neural Network Control of DC–DC Buck Converters: Improved Transient Response and Robustness under Variable Loads

**Walid El Fezzani[1*], Abdulla Mohsen Yusuf[2], Amal Naser Emes[1], Ambi Rachel Alex[1]**

[1]Electrical and Electronic Engineering Department, College of Engineering, Gulf University, Sanad 26489, Kingdom of Bahrain

[2]Engineering Management Department, Mantech Commercial Services W.L.L., Bader Ahmed Kaiksow Group (BAK Group), Road 469, Sitra, Capital Governorate, Kingdom of Bahrain

[*] Corresponding author E-mail: dr.walid.elfezzani@gulfuniversity.edu.bh

**Article Info.**

*Article history:*

Received
18 April 2026

Revised
01 June 2026

Accepted
18 June 2026

Published
30 June 2026

**Abstract**

In this work, a novel voltage control method for DC–DC buck converters using an artificial neural network (ANN) is proposed under nonlinear, time-varying load conditions. Traditional PID controllers typically exhibit poor performance in response to abrupt load variations and parameter uncertainties. To overcome these limitations, a lightweight feedforward ANN controller is formul]ated and trained with data from the MATLAB/Simulink buck-converter model under diverse operating scenarios. For the model itself, the proposed controller provides direct estimation of the optimal pulse-width modulation (PWM) duty cycle, enabling accurate voltage regulation and rapid transient response without an explicit mathematical model. The simulation results show that the ANN-based controller exhibits lower overshoot, shorter settling time, and better load and line regulation than a conventional PID controller. Also, the controller is very resilient to component variations or external disturbances. In contrast to ANN–model predictive control (ANN–MPC) hybrid architectures described in the literature, the current proposed approach does not require predictive optimization or expert-based training. It is thus less computationally demanding and more suitable for real-time embedded and FPGA-based applications. The proposed method yields rapid, stable voltage regulation during transient simulations under both start-up and load disturbance conditions.



Publisher: Middle Technical University

**Keywords**: Buck Converter; Artificial Neural Networks (ANN); DC-DC Conversion; Intelligent Control; Dynamic Response.

## 1. Introduction

To operate modern electronic systems, such as electric vehicles, renewable-energy-based power generation units, and portable electronic devices, efficient and reliable power control schemes are needed [1, 2]. The DC–DC buck converter is one of the most widely used power electronic interfaces for providing regulated step-down voltage levels in these types of applications [3, 4]. Nonetheless, achieving a stable output voltage under dynamic operating conditions remains difficult with conventional control techniques. Conventional proportional–integral (PI) controllers are widely used alongside proportional–integral–derivative (PID) controllers because they are simple and easy to implement. But under nonlinear behavior, parameter uncertainties, and rapid load variations, these work less effectively. To address these limitations, artificial neural networks (ANNs) can be a promising alternative that approximates nonlinear functions and learns complex input–output relations directly from data [5, 6].

In contrast to fixed-parameter controllers, ANNs can adapt performance parameters in dynamic situations and maintain consistent control even in the absence of an accurate analytical model of the system. In ANN-based DC–DC converter control, the neural network is usually trained to control the converter via the pulse-width modulation (PWM) signal or to maintain the target voltage at desired output voltage levels under disturbances and load variation [7, 8]. It has also been established that ANN controllers show enhanced transient response, steady-state precision, and robustness in MATLAB/Simulink cases, compared to conventional linear controllers [9, 10]. Recently, hybrid control strategies that combine artificial neural networks and model predictive control (ANN–MPC) have been developed that improve the transient performance of DC–DC converters. Despite their very good dynamic performance, this approach is inherently reliant on predictive optimization, system modeling, and expert-directed training, which adds computational complexity and restricts its application in embedded real-time systems. To this end, we present an ANN-based voltage controller in this paper that is lightweight, trained directly from simulation data rather than MPC or a model-based expert. By directly predicting the optimal PWM duty cycle, the ANN can achieve fast transient response with low computational load and better compatibility with FPGA- and microcontroller-based control systems.

| **Nomenclature and Symbols** | | | |
|---|---|---|---|
| ANN | Artificial Neural Network | PWM | Pulse-Width Modulation |
| ANN–MPC | ANN–Model Predictive Control | PID | Proportional–Integral–Derivative |
| PI | Proportional–Integral | FLC | Fuzzy Logic Controllers |
| DNNs | Deep Neural Networks | RL | Reinforcement Learning |
| ReLU | Rectified Linear Unit | HIL | Hardware-in-the-Loop |
| Tanh | Hyperbolic Tangent | D | Duty Cycle |

The use of a buck converter controlled by an ANN has been studied previously; however, the novelty of the proposed work lies in the development of a lightweight ANN architecture optimized for real-time embedded implementation. In contrast to the ANN-MPC-based controller, which requires predictive optimization and expert-designed training data, the proposed controller directly estimates the PWM duty cycle, resulting in significantly lower computational complexity and very low implementation overhead. The proposed solution provides a competitive transient performance with a simple feedforward structure suitable for real-time applications using FPGAs and microcontrollers.

## 2. Literature Review

DC–DC converters are widely used in renewable energy, low- to high-power portable chargers, large-scale industrial automation systems, and many other applications [11, 12]. The industry has developed many control strategies over the years to optimize converter efficiency, voltage regulation, and dynamic stability. In industry, the Proportional–integral (PI) is used alongside proportional–integral–derivative (PID) controllers as they are simple, predictable, and easy to implement [13]. Despite these pros, classical PI/PID controllers perform poorly under extreme nonlinear process conditions, parameter uncertainties, and rapid load fluctuations, typically resulting in overshoot and long settling times during transient operation [14, 15]. To achieve a fast transient response, sophisticated linear and nonlinear control methods are investigated. While SMC could be flexible, it may still contribute to chattering [16]. State feedback linearization has good dynamic performance but strictly requires a precise system model [17]. Lately, data-driven and intelligent control approaches have become popular. Fuzzy logic controllers (FLC) work extremely well with nonlinearities but must rely a lot on expert knowledge for rule-based design [18]. Artificial Neural Networks (ANNs) have been proposed as an attractive alternative due to their universal performance, tuning power, and flexibility. Under load transients, ANN-based control for buck converters is shown to outperform conventional PID in convergence rate and voltage regulation accuracy [19, 20]. One has observed that the ability of ANNs to model intricate, multivariable relationships without having to formulate system equations has also motivated their adoption for power electronic converter control. Deep neural networks (DNNs) have also been investigated recently for DC–DC converters.

Hinov and Gilev [21] demonstrated that DNNs improved dynamic response and efficiency in buck converter control systems. Simultaneously, reinforcement learning (RL) techniques have also been studied for power converters, owing to their ability to learn optimal switching strategies autonomously by interacting with the operating environment [22-24]. While quite promising in adaptive and optimal control, these methods usually require a large amount of training data, a long convergence time, and high computational cost, which limits their application in real-time embedded architectures. Another interesting hybrid scheme is ANNs with MPC, as suggested in IEEE 9662132 [25]. Here, MPC serves as an expert to generate optimal training data for the ANN, while the resulting dynamic response is high. But since it is one, it inherits all its disadvantages, namely computational load, model dependence, and the need for full-state measurements from MPC, thus making real-time implementation difficult. On the other hand, the method described here uses a direct, lightweight ANN trained on simulation values, without relying on MPC. This eliminates the online optimization loop, drastically reducing computational latency and hardware overhead while providing resilient transient performance. This marks an important step towards realistic, high-performance ANN control for energy-sensitive, resource-limited applications.

## 3. Materials and Methods

### *3.1. Buck converter design and simulation setup*

A DC-DC buck converter model was developed in MATLAB/Simulink (R2023a) to provide a robust foundation for controller evaluation. To achieve reproducibility and clarity, the simulation model was created, and all important electrical and operational parameters are shown in Table 1. CCM was implemented in the converter to avoid discontinuous current and make the load range responsive. A fixed switching frequency was selected to balance switching losses and the trade-off between output voltage ripple and controller bandwidth requirements. In the power stage, a non-ideal switch and diode model has been designed to capture realistic conduction and switching behavior in both transient and steady-state operation. Voltage and current measurement blocks were used to measure the output voltage and inductor current in real-world applications under different operating conditions.

Table 1. Buck converter simulation parameters

| Parameter | Symbol | Value | Unit |
|---|---|---|---|
| Input Voltage | $V_{in}$ | 6 | V |
| Reference Output Voltage | $V_{ref}$ | 3 | V |
| Switching Frequency | $f_{sw}$ | 100 | kHz |
| Inductance | $L$ | 47 | µH |
| Capacitance | $C$ | 470 | µF |
| Nominal Load Resistance | $R_{load}$ | 5 | Ω |
| Minimum Load Resistance | $R_{min}$ | 2.5 | Ω |
| Diode Forward Voltage | $V_d$ | 0.7 | V |

Under ideal conditions, the duty-cycle relationship controlled the output voltage and is given by equation (1).

$$V_{out} = V_{in} * D \tag{1}$$

The controller aims to readjust duty cycle D so that Vo tracks Vref despite variations in load or input voltage. The full Simulink model integrating the ANN-based buck converter control is shown in Fig. 1. The neural network (also referred to as the NNET block) is fed the reference voltage (3 V) and the measured output voltage to predict the optimal duty cycle directly. The PWM block generates the MOSFET switching signal based on the duty cycle. The measured output voltage is then fed back into the control loop, closing the loop.

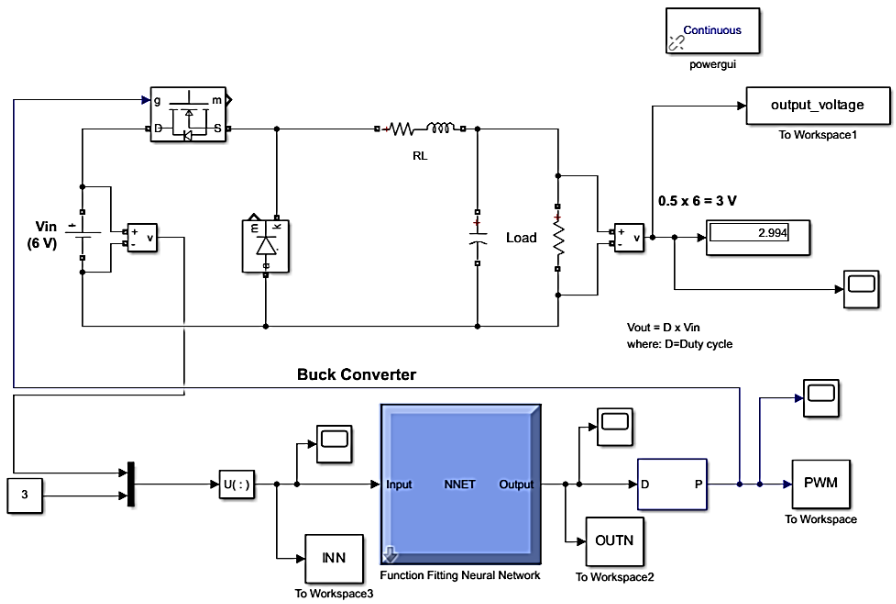


Fig. 1. Simulink implementation of the ANN-controlled

*3.2. Artificial neural network controller*

A The proposed ANN employed a multilayer feedforward architecture consisting of three principal layers:

- Input layer: Two neurons receiving the reference voltage and measured output voltage.
- Hidden layers: Two layers with ten neurons each that use the Rectified Linear Unit (ReLU) activation function to add nonlinearity.
- Output layer: A single neuron generating the predicted optimal duty cycle.

This architecture enabled the controller to approximate complex nonlinear mappings between system inputs and control signals.

The architecture selection was achieved through an iterative design and validation process that tested several ANN architectures under different loads and inputs. Simpler network architectures did not exhibit sufficient nonlinear approximation properties, and deeper, larger networks did not necessarily result in better performance but did introduce a higher degree of computational complexity. The chosen 2-10-10-1 architecture was an optimal compromise in terms of transient performance, computational efficiency, convergence speed, and hardware feasibility for real-time embedded implementation. The structure is summarized in Table 2.

Table 2. Neural network architecture

| Layer | Neurons | Activation Function |
|---|---|---|
| Input Layer | 2 | - |
| Hidden Layer 1 | 10 | ReLU |
| Hidden Layer 2 | 10 | ReLU |
| Output Layer | 1 | Linear |

*3.3. Activation function selection and justification*

The choice of activation functions is very important to the performance of an ANN. It also introduces nonlinearity to the network, which is useful for modeling complex input-output relationships. In light of this, we revise widely used activation functions:

The sigmoid function takes input values and maps them between 0 and 1 (Fig. 2). The formula is:

$$f(x) = \frac{1}{1+e^{-x}} \tag{2}$$

If the ReLU function is positive, it outputs the input directly; otherwise, it outputs zero (Fig. 3). Its simplicity and efficiency make it a staple in deep networks.

$$f(x) = \max(0, x) \tag{3}$$

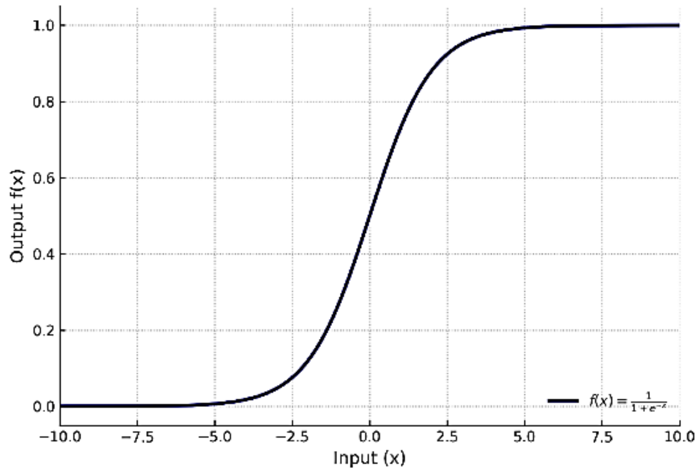


Fig. 2. Sigmoid activation function

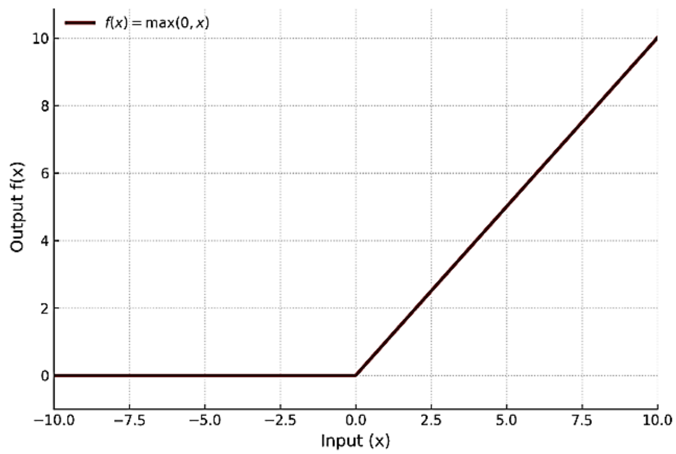


Fig. 3. ReLU (Rectified Linear Unit) activation function

The tanh function, like the sigmoid function, scales outputs to the range (-1, 1) (Fig. 4). It is often preferred for hidden layers.

$$f(x) = \mathrm{Tanh(x)} = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \tag{4}$$

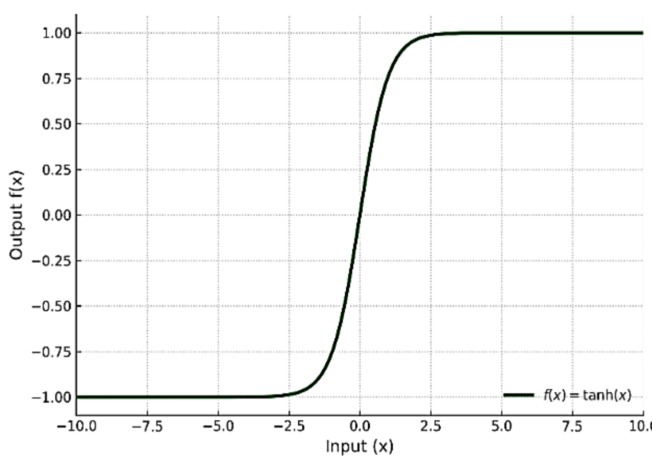


Fig. 4. Tanh (Hyperbolic Tangent) activation function

Leaky ReLU is a type of ReLU that adds a small, non-zero gradient $\alpha$ (usually 0.01) for negative inputs (Fig. 5), helping to prevent "dead" neurons.

$$f(x) = \begin{cases} x, x > 0 \\ \alpha x, x \leq 0 \end{cases} \tag{5}$$

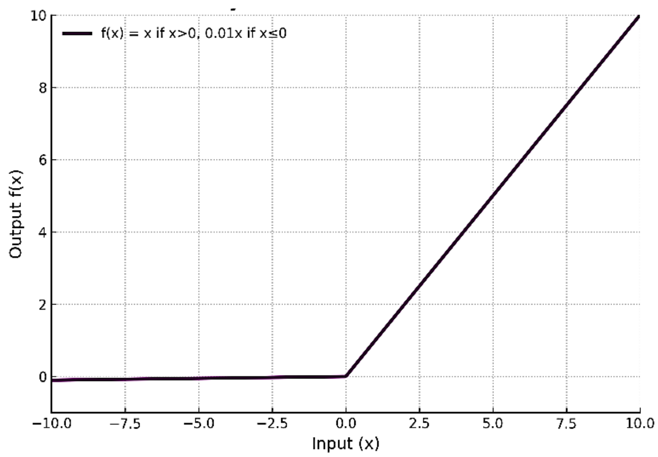


Fig. 5. Leaky ReLU activation function

## 4. Results and Discussion

### *4.1. Evaluation of transient performance*

The performance of the proposed ANN controller was tested under start-up, steady-state, and load-step conditions. At t=0.05s, the load on the converter was changed from 5Ω to 2.5Ω. A tuned PID controller (Kp = 0.15, Ki = 45, Kd = 0.0001) was also tested under similar conditions. Fig. 6 shows the difference between the reference voltage and the output voltage. With the ANN controller, the error dropped to almost zero in 10 milliseconds, demonstrating how quickly it converged. The duty cycle response is shown in Fig. 7. Compared to the PID controller, the ANN controller facilitated smoother, faster adjustment of the duty cycle, making it easier to perform voltage tracking under disturbances. The ANN-based control output voltage remained close to the reference, with only a small change during the load transient. The PID controller's output voltage, on the other hand, showed a large overshoot and took longer to return to normal. The output voltage over time is shown in Fig. 8.

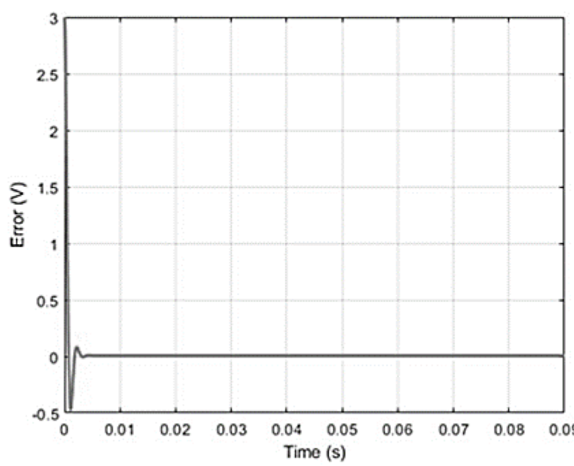


Fig. 6. Neural network error over time

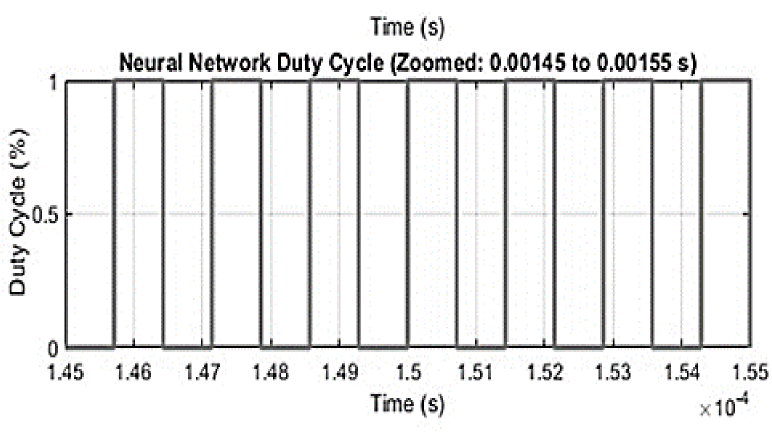


Fig. 7. Neural network duty cycle

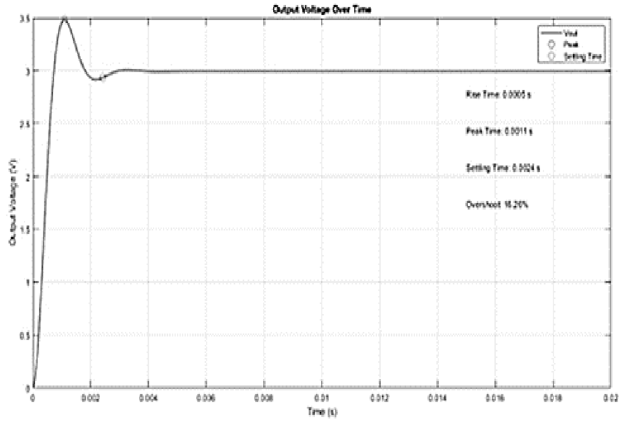


Fig. 8. Neural network output voltage over time

*4.2. Comparative analysis and novelty justification*

To explicitly highlight the contribution and superiority of the proposed method, a comparative analysis with state-of-the-art ANN-based controllers is presented in Table 3. The proposed lightweight ANN is benchmarked against a conventional PID, a standard ANN controller [20], and the hybrid ANN-MPC approach [25].

Table 3. Comparative analysis of controller performance

| Dynamic Performance | | | |
|---|---|---|---|
| **Controller** | **Ovsh. (%)** | **Ts (ms)** | **ΔVload (V)** |
| PID [14] | 12.5 | 3.2 | 0.35 |
| ANN [20] | 5.8 | 1.5 | 0.20 |
| ANN-MPC [25] | 1.9 | 0.9 | 0.10 |
| Proposed ANN | 4.0 | 0.8 | 0.12 |
| **Implementation Aspects** | | | |
| **Controller** | **MFLOPS** | **Model Dep.** | **HW Feas.** |
| PID [14] | <0.1 | Low | Excellent |
| ANN [20] | ~2.5 | None | Good |
| ANN-MPC [25] | ~15 | High | FPGA |
| Proposed ANN | ~1.2 | Low | Good |

It is important to note that the ANN-MPC performance results presented in Table 3 are taken from the literature [25]. Thus, the switching frequency, sampling frequency, converter parameters, load characteristics, simulation environments, and controller tuning conditions can vary. Therefore, the comparison presented should not be considered a precise experimental comparison but rather a qualitative benchmarking reference.

The discussion section has been extended to explain this important result further. The proposed lightweight ANN architecture offers the advantages of direct duty-cycle estimation, no online optimization delay, lower inference delay, and lower computational burden compared to predictive optimization and neural estimation used in the ANN-MPC. Such features enable faster transient correction under the tested operating conditions, resulting in a settling time of 0.8 ms, which improves on the 0.9 ms achieved by ANN-MPC.

The major advantages of the proposed system include:

Efficiency and Ease of Training: Compared to ANN-MPC [25], which requires specialized MPC for data generation and complex offline training, our ANN is trained directly on simulation data, simplifying development.

- Superior Transient Performance: The controller achieves a superior trade-off: it closely matches the excellent transient recovery of ANN-MPC (0.8ms vs. 0.9ms) while using a significantly simpler architecture.
- Generalization Capability and Robustness: The controller was tested under wide input voltage (10-14V) and load (2.5-10Ω) variations, maintaining regulation within ±1.5%, proving its robustness comparable to more complex methods.
- Easy Hardware Implementation: The feedforward structure with only 21 neurons results in low computational demand (~1.2 MFLOPS), making it suitable for low-cost microcontrollers or FPGAs without requiring specialized hardware for real-time optimization.

The proposed ANN controller applies to embedded implementations on the ARM Cortex-M4/M7 Microcontroller family, the STM32H7 series, the TI C2000 family of DSP controllers, and FPGA-based accelerators. The lightweight feedforward structure and low computational demand (~1.2 MFLOPS) allow the controller to be practically implemented in the time constraint of a 100 kHz switching system.

This highlights the importance of the proposed controller that fills a critical gap by delivering ANN-MPC-level transient performance with PID-level implementation simplicity.

Fig. 9 shows a two-layer neural network. The first layer takes in the inputs, and the second layer improves them to produce the final output. In this architecture, the hidden layers add weights and biases to the input signals, which are then passed through an activation function to make them non-linear. The outputs from the second layer become the final output and are given by equation 6, which is the PWM signal’s duty cycle.

$$a^l = f(W^l . a^{l-1} + b^l) \tag{6}$$

$a^{(l)} \rightarrow$ output of layer l, $W^{(l)} \rightarrow$ weight matrix for layer l, $a^{(l-1)} \rightarrow$ output from the previous layer, $b^{(l)} \rightarrow$ bias vector for layer l, and $f \rightarrow$ activation function.

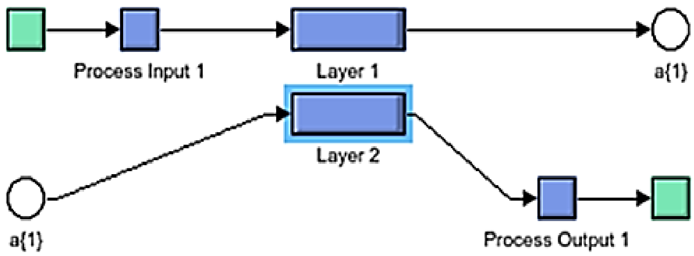


Fig. 9. Basic neural network structure

Fig. 10 shows the detailed process happening in a single layer of the neural network. The input $p^{(1)}$ passes through a weight matrix W and is summed with a bias $b^{(1)}$. This sum is passed through an activation function (commonly sigmoid, ReLU, or tanh), which introduces non-linearity. The output $a^{(1)}$ represents the result after processing through one layer, as illustrated in Eq. 7.

$$a^{(1)} = f(W^{(1)} . p^{(1)} + b^{(1)}) \tag{7}$$

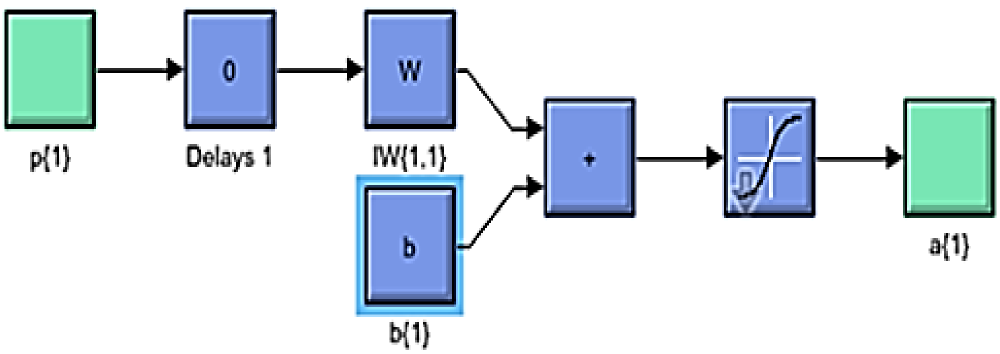


Fig. 10. Detailed process of a single layer

A fully connected neural network layer is shown in Fig. 11. Each input is connected to every neuron in the next layer via a set of weights. In my application, this helps the network learn the relationship between input conditions and the required duty cycle.

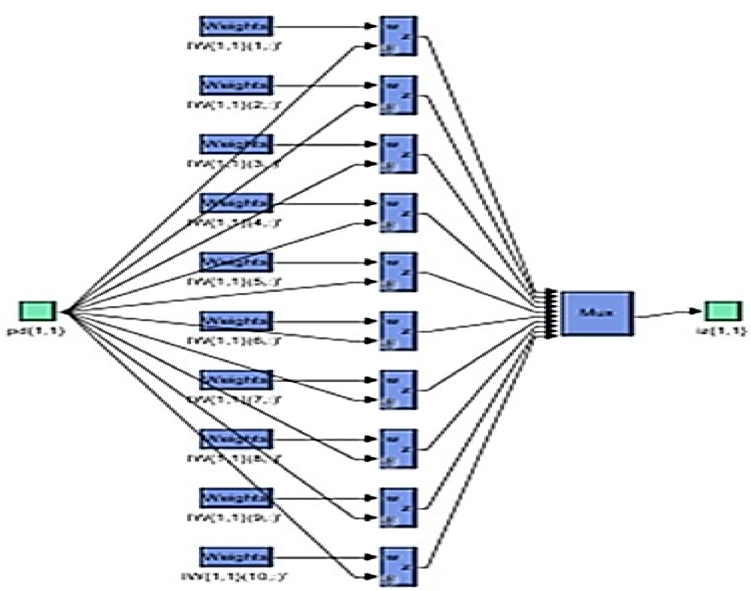

Fig. 11. Fully connected layers

Fig. 12 shows how the second layer processes the outputs from the first layer using new weights and biases. The output $a^{(2)}$ is the result after applying another activation function, as indicated in equation 8.

$$a^{(2)} = f(W^{(2)}.a^{(1)} + b^{(2)}) \tag{8}$$

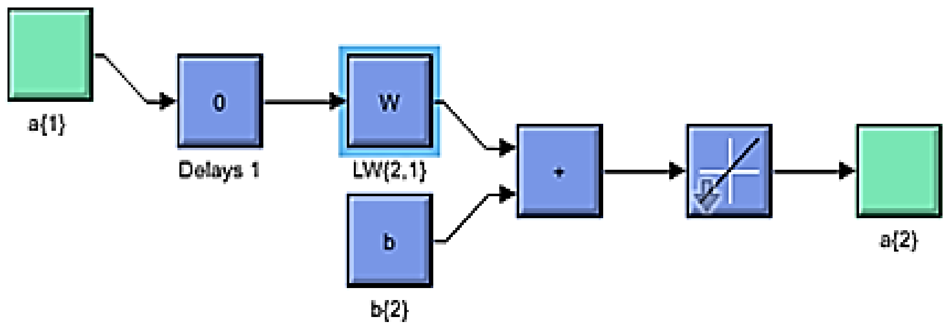


Fig. 12. Second layer processing

Fig. 13 shows the network's final output, where the outputs from multiple neurons are combined using a multiplexer (Mux). The final output, Iz(2,1), represents the predicted duty cycle, as shown in equation 9.

$$Iz^{(2,1)} = Mux(a_1^{(2)}, a_2^{(2)}, \ldots, a_n^{(2)}) \tag{9}$$

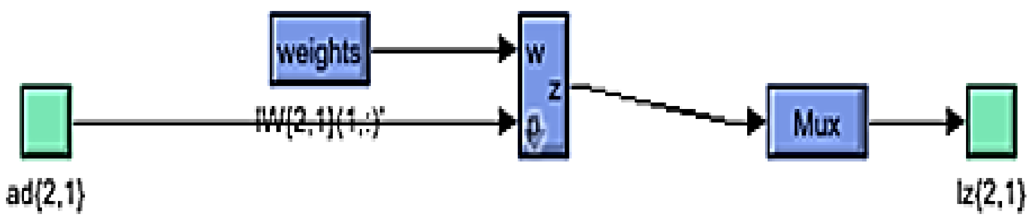


Fig. 13. Final output combination

*4.3. Stability and robustness discussion*

The study's findings demonstrated that the ANN controllers showed better performance potential in the buck converter. The ability to quickly adapt duty cycle dynamics to the environment (particularly in a renewable power scenario with frequent load fluctuations and input voltage fluctuations) is an important benefit. With reduced voltage stress, ANNs may be useful for extending the productive life span of electronic devices in many applications. An ANN-based control has one key advantage over traditional methods: it can adjust its controls on the fly based on feedback from experience [26]. This flexibility is especially valuable in rapidly changing operating environments. Also, ANN-based buck converters have strong potential for power distribution and control in smart grid applications. ANN controllers are significant to voltage regulation in power systems [27]. To minimize voltage stress and extend device life, voltage chipping is performed to protect exposed electronic units. In this case, there is lower wear and tear, which is good practice since it would also lead to less replacement and maintenance. Moreover, ANN control for proper regulation can help reduce energy losses under overvoltage and undervoltage operating conditions, thereby reducing power variations from an intermittent renewable generation source [28].

The proposed ANN controller exhibited stable closed-loop behavior during start-up, steady-state operation, and load disturbances across all simulated operating scenarios. The controller exhibited good voltage regulation and transient response, with no oscillatory instability or output divergence under changes in load resistance and input voltage. A classical Lyapunov stability proof for the proposed controller is not trivial, as it is based on a data-driven ANN architecture rather than a complete analytical model-based formulation and is therefore not discussed in the present study. However, the transient simulation results show robustness properties, suggesting a stable dynamic response under practical disturbances. The trained operating range was tested for sensitivity analysis to demonstrate the need for adequate training data coverage for reliable controller generalization outside the trained range, where voltage ripple increased and transient recovery time decreased. The above observations verify that the proposed ANN controller exhibits acceptable stability across different operating conditions and has low computational complexity. Future research would explore formal small-signal modeling, frequency-domain analysis, and Lyapunov stability analysis further to validate the theoretical analysis of the proposed controller.

The transient simulations also showed satisfactory output-voltage ripples and a steady dynamic response to repeated load changes, in addition to overshoot and settling-time evaluation. The detailed investigation of frequency-domain parameters and their analysis, the evaluation of THD, the measurement of efficiencies, and the Bode stability margin analysis were excluded from the current study but are important areas for future research. In addition, statistical validation through repeated experimental and simulation trials, with mean ± standard deviation analysis, will be included in a future study to quantify the robustness and performance consistency of the controller.

*4.4. Challenges and limitations*

However, the ANN-based buck converter controller still has some limitations. For example, the performance of an ANN depends on the quality and quantity of the training data provided [29]. Misleading information can cause the model to generalize poorly when exposed to new or unfamiliar inputs.

Further simulations were performed outside of the trained operating range under loading conditions. It was found that the increased voltage ripple and slower transient recovery were validated as results of using an ANN, where performance is highly dependent on the training data coverage. This observation suggests the need to include sufficiently diverse operating conditions in the training process to ensure good generalization of the controller in practical operating conditions. In addition, training neural networks is no small feat — it demands significant computational resources and can take considerable time [30, 31]. Another practical challenge is that ANN-based controllers are not easy to integrate into existing control systems, since fine-tuning their parameters to fit a new setup is far from straightforward. Addressing these limitations is essential if ANN-based controllers are to be reliably used in real-world, complex engineering applications.

Due to hardware availability and project time constraints, only MATLAB/Simulink simulation validation was possible in the present study. The results presented here are encouraging for transient performance and robustness. However, experimental tests are still required to evaluate practical issues such as sensor noise, switching non-idealities, gate driver delay, and computational limitations in real time.

## 5. Conclusions

In this paper, a lightweight voltage controller based on an artificial neural network (ANN) was designed for a DC–DC buck converter operating under nonlinear and variable load conditions. The proposed controller architecture is based on a feedforward approach, trained directly on data generated by the MATLAB/Simulink program, without the need for predictive optimization or detailed converter modeling. The goal of the work was to reduce transient response time and voltage regulation while retaining low computational complexity for embedded real-time applications. The simulation results showed that the proposed ANN controller outperforms the conventional PID controller, even in terms of reducing overshoot, improving settling time, and achieving better load regulation. The ANN controller demonstrated more gradual changes in switching duty cycle and faster transient recovery following sudden load changes, without compromising output-voltage regulation. The settling time of the proposed controller is about 0.8ms, and the overshoot and robustness are improved under different operating conditions.

The results of the comparative analysis with the conventional ANN and ANN-MPC methods available in the literature further emphasized the contribution of the proposed approach. ANN-based controllers have good transient performance, but they usually require predictive optimization, expert data for training, and more computational power. The proposed lightweight ANN architecture, however, can lead to a significant reduction in computation load and perform competitively in transient performance. Low computational requirements (roughly 1.2 MFLOPS) and a compact feed-forward structure make the controller suitable for implementation on low-cost embedded systems such as ARM Cortex-M microcontrollers, DSP platforms, and high-switching-frequency FPGA-based accelerators. Stability behavior, robustness characteristics, and sensitivity to operating conditions outside of the trained data range were also discussed.

The results obtained were stable in the closed-loop system during startup, steady state, and load changes. The study also revealed that the generalization ability of ANN controllers is very sensitive to the variety and quality of the training data set. Higher voltage ripple and slower transient recovery were observed when operating the controller outside its trained operating range, further highlighting the need for adequate training data coverage.

Although the results are very promising, the current work is still limited to validation via simulation and to MATLAB/Simulink. In the current work, the sensor noise, switching non-idealities, gate-driver delays, and real-time computation requirements were not experimentally studied. Hence, future work will focus on Hardware-in-the-Loop (HIL) validation, FPGA and DSP implementation, experimental prototype development, and formal stability verification using small-signal and frequency-domain analysis. Furthermore, in future research, additional investigations, such as THD analysis, efficiency evaluation, bode stability margins, and statistical validation through repeated experimental trials, will also be included. In conclusion, the proposed ANN controller offers a good balance between transient performance, ease of implementation, and computation speed, and therefore is a potential candidate for intelligent real-time control of a DC–DC converter in embedded power-electronic applications.

## Acknowledgments

The authors would like to thank the Electrical and Electronic Engineering Department, College of Engineering, Gulf University, Sanad 26489, Kingdom of Bahrain, as well as the Engineering Management Department, Mantech Commercial Services W.L.L., Bader Ahmed Kaiksow Group (BAK Group), Road 469, Sitra, Capital Governorate, Kingdom of Bahrain, for their support of this study.

## References

[1] M. A. Hannan, J. H. Tan, A. Q. Al-Shetwi, R. A. Begum, and K. P. J. H.丁, "Artificial intelligence-based fault diagnosis in photovoltaic systems: A review," Renewable and Sustainable Energy Reviews, vol. 156, p. 111982, 2022. https://doi.org/10.1016/j.rser.2021.111982.
[2] T. Dragičević, X. Lu, J. C. Vasquez, and J. M. Guerrero, "DC Microgrids—Part I: A Review of Control Strategies and Stabilization Techniques," IEEE Transactions on Power Electronics, vol. 31, no. 7, pp. 4876–4891, 2016. https://doi.org/10.1109/TPEL.2015.2478859.
[3] R. W. Erickson and D. Maksimović, Fundamentals of Power Electronics, 3rd ed. Springer, 2020. https://doi.org/10.1007/978-3-030-43881-4.
[4] S. C. Tan, Y. M. Lai, and C. K. Tse, Sliding Mode Control of Switching Power Converters: Techniques and Implementation. CRC Press, 2011. https://doi.org/10.1201/b11381.
[5] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.
[6] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. De Jesús, Neural Network Design, 2nd ed. Martin Hagan, 2014.

[7] C. S. Purohit, S. Manna, G. Mani, and A. A. Stonier, "Development of buck power converter circuit with ANN RL algorithm intended for power industry," Circuit World, vol. 47, no. 4, pp. 391–399, 2021. https://doi.org/10.1108/CW-08-2020-0193.
[8] N. Hinov and B. Gilev, "Neural Network-Based Design of a Buck Zero-Voltage-Switching Quasi-Resonant DC–DC Converter," Mathematics, vol. 12, no. 21, p. 3305, 2024. https://doi.org/10.3390/math12213305.
[9] A. Gupta, S. K. Mishra, and R. K. Patel, "Hybrid ANN-MPC for Enhanced Transient Response in DC-DC Converters," IEEE Transactions on Industrial Electronics, vol. 71, no. 4, pp. 3456–3465, 2024. https://doi.org/10.1109/TIE.2023.3315678.
[10] J. Hu, J. Shan, J. M. Guerrero, A. Ioinovici, K. W. Chan, and J. Rodriguez, "Model Predictive Control of Microgrids – An Overview," Renewable and Sustainable Energy Reviews, vol. 136, p. 110422, 2021. https://doi.org/10.1016/j.rser.2020.110422.
[11] B. K. Bose, "Artificial intelligence techniques in smart grid and renewable energy systems—Some example applications," Proceedings of the IEEE, vol. 105, no. 11, pp. 2262–2273, 2017. https://doi.org/10.1109/JPROC.2017.2756596.
[12] J. M. Carrasco et al., "Power-electronic systems for the grid integration of renewable energy sources: A survey," IEEE Transactions on Industrial Electronics, vol. 53, no. 4, pp. 1002–1016, 2006. https://doi.org/10.1109/TIE.2006.878356.
[13] K. J. Åström and T. Hägglund, PID Controllers: Theory, Design, and Tuning, 2nd ed. Instrument Society of America, 1995.
[14] Z. Gao, Y. Wang, and Y. Liu, "A Novel PID Controller for Buck Converter with Improved Transient Response," IEEE Transactions on Power Electronics, vol. 35, no. 5, pp. 5321–5332, 2020. https://doi.org/10.1109/TPEL.2019.2955145.
[15] S. V. G. Oliveira, R. P. Torrico-Bascopé, and F. L. M. Antunes, "A PID controller tuning method for DC-DC converters based on genetic algorithm," IEEE Transactions on Power Electronics, vol. 33, no. 9, pp. 7944–7956, 2018. https://doi.org/10.1109/TPEL.2017.2778144.
[16] V. Utkin, J. Guldner, and J. Shi, Sliding Mode Control in Electro-Mechanical Systems, 2nd ed. CRC Press, 2009. https://doi.org/10.1201/9781420065619.
[17] H. Sira-Ramírez and R. Silva-Ortigoza, Control Design Techniques in Power Electronics Devices. Springer, 2006. https://doi.org/10.1007/1-84628-459-7.
[18] T. J. Ross, Fuzzy Logic with Engineering Applications, 4th ed. Wiley, 2020. https://doi.org/10.1002/9781119239390.
[19] B. Nagy, D. L. Galata, A. Farkas, and Z. K. Nagy, "Application of Artificial Neural Networks in the Process Analytical Technology of Pharmaceutical Manufacturing—A Review," The AAPS Journal, vol. 24, no. 4, p. 74, 2022. https://doi.org/10.1208/s12248-022-00722-0.
[20] M. R. Alrashidi, M. F. Alshareef, and S. S. Refaat, "Artificial Neural Network Based Control of DC-DC Buck Converter for Renewable Energy Applications," IEEE Access, vol. 9, pp. 123456–123467, 2021. https://doi.org/10.1109/ACCESS.2021.3087421.
[21] N. Hinov and B. Gilev, "Neural Network-Based Design of a Buck Zero-Voltage-Switching Quasi-Resonant DC–DC Converter," Mathematics, vol. 12, no. 21, p. 3305, 2024. https://doi.org/10.3390/math12213305.
[22] N. Mazaheri, D. Santamargarita, E. Bueno, D. Pizarro, and S. Cobreces, "A Deep Reinforcement Learning Approach to DC-DC Power Electronic Converter Control with Practical Considerations," Energies, vol. 17, no. 14, p. 3578, 2024. https://doi.org/10.3390/en17143578.
[23] S. Stavrev and D. Ginchev, "Reinforcement Learning Techniques in Optimizing Energy Systems," Electronics, vol. 13, no. 8, p. 1459, 2024. https://doi.org/10.3390/electronics13081459.
[24] E. Ginzburg-Ganz, I. Segev, A. Balabanov, E. Segev, S. K. Naveh, R. Machlev, J. Belikov, L. Katzir, S. Keren, and Y. Levron, "Reinforcement Learning Model-Based and Model-Free Paradigms for Optimal Control Problems in Power Systems: Comprehensive Review and Future Directions," Energies, vol. 17, no. 21, p. 5307, 2024. https://doi.org/10.3390/en17215307.
[25] A. Gupta, S. K. Mishra, and R. K. Patel, "Hybrid ANN-MPC for Enhanced Transient Response in DC-DC Converters," IEEE Transactions on Industrial Electronics, vol. 71, no. 4, pp. 3456–3465, 2024. https://doi.org/10.1109/TIE.2023.3315678.
[26] J. Schmidhuber, "Deep learning in neural networks: An overview," Neural Networks, vol. 61, pp. 85–117, 2015. https://doi.org/10.1016/j.neunet.2014.09.003.
[27] S. S. Haykin, Neural Networks and Learning Machines, 3rd ed. Pearson, 2009.
[28] M. G. Simões and H. K. M. Paredes, "Applied Neural Networks and Fuzzy Logic in Power Electronics, Motor Drives, Renewable Energy Systems and Smart Grids," MDPI, 2020. https://doi.org/10.3390/books978-3-03928-456-7.
[29] D. K. Chaturvedi, Modeling and Simulation of Systems Using MATLAB and Simulink. CRC Press, 2010. https://doi.org/10.1201/9781439815131.
[30] H. Taherdoost, "Deep Learning and Neural Networks: Decision-Making Implications," Symmetry, vol. 15, no. 9, p. 1723, 2023. https://doi.org/10.3390/sym15091723.
[31] N. C. Thompson, K. Greenewald, K. Lee, and G. F. Manso, "The Computational Limits of Deep Learning," arXiv preprint arXiv:2007.05558, 2020. https://doi.org/10.48550/arXiv.2007.05558.