2024 16th International Conference on Information Technology and Electrical Engineering (ICITEE) | 979-8-3503-7581-7/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICITEE62483.2024.10808195

Buck Converter Parameters Design Using Artificial Intelligence-Based Genetic Algorithm

Krishna Laksheta
Department of Electrical and Information Engineering
Universitas Gadjah Mada
Yogyakarta, Indonesia
krishnalaksheta@mail.ugm.ac.id

Yohan Fajar Sidik*
Department of Electrical and Information Engineering
Universitas Gadjah Mada
Yogyakarta, Indonesia
yohanfajarsidik@ugm.ac.id

Fransisco Danang Wijaya
Department of Electrical and Information Engineering
Universitas Gadjah Mada
Yogyakarta, Indonesia
danangwijaya@ugm.ac.id

***Abstract*—The design of circuit parameters is critical to optimize the performance and efficiency of power converters. This research focuses on designing synchronous buck converters using an AI-based approach that combines machine learning models and genetic algorithms. The genetic algorithm run on the machine learning model operates within error bounds for voltage ripple ($\leq 1\%$), current ripple ($\leq 10\%$), and total component volume ($\leq 7\text{cm}^3$). It uses these criteria to evaluate fitness values. The results show that the AI-based design achieves higher efficiency (99.383%), lower voltage ripple (0.168%), and lower current ripple (9.757%) compared to the conventional design (99.354%, 1.1543%, and 11.661%). In addition, the AI-based method is more compact (5.732 cm^3 vs. 7.716 cm^3). Thus, AI is proven to be more effective in improving power converters' efficiency and performance than conventional methods.**

***Index Terms*—buck converter, efficiency, machine learning, parameter optimization, power loss**

I. Introduction

Power electronics is a technology related to conversion and electric power control. Power electronics have some topologies used to control and convert voltage and current [1]. One of the most frequently used power converters is a buck converter that converts direct voltage (DC) from one level to another with a lower level. The implementation of power electronic converters is becoming a critical component in several applications such as electric vehicles [2], renewable energy [3], and energy storage systems [4].

Power converters require careful design of circuit parameters to meet desired performance and specifications. This performance includes size, cost, efficiency, reliability, ripple, and transient response [5], [6]. Optimal performance requires careful consideration of all these aspects, which shows the importance of analysis in converter design.

The design of circuit parameters in power converters traditionally involves two main processes: analysis and deduction and optimization [7]. Two approaches have been commonly used: the traditional human-dependent method (conventional method) and computer-aided optimization [5], [8]–[11]. Conventional methods heavily rely on human expertise, requiring engineers to perform manual calculations and draw from empirical knowledge. Complex mathematical equations must be solved for analysis and deduction process, while optimization often relies on personal experience and trial-and-error experiments. This method is limited by human analytical capabilities and susceptible to inaccuracies [7]. In addition, these methods have difficulty handling the many interrelated parameters and constraints, which often result in suboptimal solutions.

Computer-aided optimization (CAO) methods aim to overcome the limitations of conventional methods by using algorithms in the optimization process. Algorithms will be created based on the system model, and then optimization will be performed to select parameters and components [8]. However, this method relies on manual calculations for the analysis and deduction process. This hybrid approach still lacks accuracy, efficiency, and the ability to handle complex and multi-objective problems [8], [9]. The fundamental challenge facing conventional and CAO methods is their inability to simultaneously manage the complicated interactions between various design objectives and constraints. The more complex the power converter design, the more difficult it is to find an optimal solution for these methods.

On the other hand, new approaches have evolved with the advancement of artificial intelligence (AI). The application of AI in power converter circuit design offers significant improvements in the analysis and deduction and optimization process. AI can quickly analyze large amounts of data and identify complex patterns in circuit parameter design that conventional and CAO methods find difficult [12]. For instance, neural networks were employed for circuit parameter design in buck converters, achieving improved design efficiency through deep automation and predictive modeling [7]. AI was applied to automate the selection of the number and topology of power converter arrays (PCA), enabling faster performance analysis and reducing design time [13]. Similarly, reinforcement learning was used to automate circuit parameter design, significantly accelerating the finding of optimal parameters that meet specified design criteria [14].

Machine learning models, in particular, have shown great potential in predicting system performance by considering multiple variables and their interactions. Unlike traditional approaches, AI is highly effective at managing multi-objective challenges, allowing for the simultaneous optimization of various constraints and goals. [15]. This capability is critical to the design of modern power converters, where efficiency, size, cost, and reliability must be carefully

Authorized licensed use limited to: UNIVERSITAS GADJAH MADA. Downloaded on March 17,2025 at 13:32:01 UTC from IEEE Xplore. Restrictions apply.

balanced [5], [6]. Furthermore, AI-based approaches can continuously learn and improve from new data and designs, potentially resulting in new optimized solutions. AI can explore an ample design space more quickly than conventional methods, thus significantly reducing the time required for the design process. By utilizing AI capabilities, designers can overcome the limitations of conventional and CAO methods to achieve more accurate, efficient, and innovative power converter designs.

This research uses an AI-based approach to design the buck converter parameters. AI technology is applied in the process of analysis and deduction and optimization. Implementing AI enables the system to identify complex patterns in the circuit. The AI technology will be the basis for running optimization algorithms to obtain optimal parameter values.

The rest of this paper is organized as follows. The implementation of AI-based circuit design is detailed in Section II. Section III covers parameter calculations using conventional methods. A comparison between the conventional and AI-based methods is presented in Section IV, followed by conclusions and future works in Section V.

II. AI-Based Circuit Design Implementation

AI-based circuit design implementation for a synchronous buck converter finds five main parameters: inductor value (L), capacitor value (C), switching frequency (f_{sw}), dead time (t_{d}), and duty cycle (d_{cycle}). The design process involves three main components: the data set, machine learning model, and optimization method. Table I shows the buck converter circuit's specifications. On the other hand, Table II shows the range of limits for each parameter that will be used in this study.

A. Dataset Creation

The dataset used in this study consists of simulation data generated using PLECS software, capacitance data, and inductance data. Figure 1 shows the synchronous buck converter circuit simulated using PLECS software.

TABLE I
Specifications of the Synchronous Buck Converter Circuit

Parameter	Value
Input Voltage (V_{in})	48 V
Output Voltage (V_{out})	12 V
Output Power (P_{o})	100 W
Current Ripple (ΔI_{L})	$\leq$ 10 %
Voltage Ripple (ΔV_{out})	$\leq$ 1 %
Volume Limit (V_{lim})	≤ 7 cm^3
Target Efficiency	90 %

TABLE II
Range of Limits for Each Parameter

Parameter	Range of Values
Inductance (L)	30 μH - 2000 μH
Capacitance (C)	20 μF - 1000 μF
Switching Frequency (f_{sw})	20 kHz - 200 kHz
Dead Time (t_{d})	175.2 ns - 262.8 ns
Duty Cycle (d_{cycle})	0.2 - 0.35

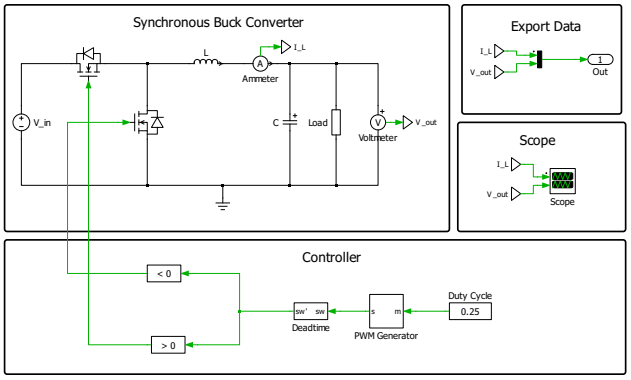


Fig. 1. Synchronous Buck Converter Schematic in PLECS

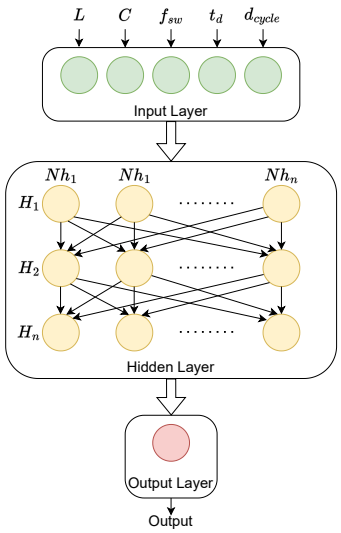


Fig. 2. Neural Network Architecture Diagram

With Python-PLECS integration, simulations can be performed automatically, and data can be processed immediately. Several simulations were performed with different parameters L, C, f_{sw}, t_{d}, and d_{cycle} for each iteration. Parameter combinations were made with ten variations for each parameter, resulting in 100,000 data points (10^5). The data set includes values for output voltage (V_{out}), output voltage ripple (ΔV_{out}), inductor current (I_{L}), inductor current ripple (ΔI_{L}), conduction loss high-side and low-side ($P_{\text{ON-H}} - P_{\text{ON-L}}$), output capacitance loss (P_{COSS}), dead time loss (P_{D}), and gate charge loss (P_{G}). In addition, the capacitance data set includes various combinations of capacitor values that can be used.

B. Machine Learning Model

The type of machine learning used is supervised learning with a neural network model. The neural network model used in this project was developed using the Keras library.

Figure 2 illustrates the machine learning model with three layers: input, hidden, and output. In the input layer, the Rectified Linear Unit (ReLU) activation function is used due to its ability to speed up training convergence by reducing

Authorized licensed use limited to: UNIVERSITAS GADJAH MADA. Downloaded on March 17,2025 at 13:32:01 UTC from IEEE Xplore. Restrictions apply.

the effect of vanishing gradients and modeling non-linear relationships in the data. The ReLU activation function is used at the hidden layer, with the number of layers (H) and neurons (Nh) determined by a grid search for each model. A Softplus activation function is used on the output layer to keep all outputs positive with small values. The number of neurons in the output layer is adapted to the number of predicted features. Equations (1) and (2) are the Mean Absolute Error (MAE) and Huber Loss formula, respectively, which are used for the loss function [16]. MAE is defined as:

$$MAE = \frac{1}{m}\sum_{l=1}^{m}|z_l - f(x_l)| \tag{1}$$

where m is the number of samples, i is the i th sample. Zi represents the true value, and $f(x_i)$ represents the predicted value.

Huber Loss is defined as:

$$L_\delta(z, f(x)) = \begin{cases} \frac{1}{2}(z - f(x))^2, & |z - f(x)| \le \delta, \\ \delta|z - f(x)| - \frac{1}{2}\delta^2, & |z - f(x)| > \delta, \end{cases} \tag{2}$$

where z represents the true value, $f(x)$ indicates the predicted value, and δ is the set hyper-parameter.

MAE is suitable for data with small values and high variability because it can tolerate extreme value fluctuations, while Huber Loss overcomes the insensitivity of MAE to error variations. The Adam optimizer is used to speed up model convergence and optimize learning. Table III summarizes the results of hyperparameter tuning with the loss functions used.

C. Optimization Method

Figure 3 depicts the genetic algorithm block with the integration of the machine-learning model. The machine learning model serves as the foundation for the optimization method, predicting the fitness value (total power loss) and verifying that the value conforms to the specified constraints (ΔI_{out}, ΔV_{out}, V_{out}). The algorithm optimizes the selection of parameter values (L, C, f_{sw}, t_{d}, and d_{cycle}) by the constraints outlined in Table I and the specified range of values presented in Table II. The implementation of the genetic algorithm consists of several main processes, starting with the initialization of the population and ending with the generation of a new population through selection, crossover, and mutation.

TABLE III
HYPERPARAMETER TUNING RESULT AND LOSS FUNCTION

Machine Learning Model	H and Nh	Loss Function
average_current	2 and 256	*Huber*
delta_current_percentage	5 and 64	*Huber*
average_voltage	1 and 64	*Huber*
delta_voltage_percentage	5 and 64	MAE
P_ON_H_P_ON_L	1 and 32	MAE
P_COSS	1 and 16	MAE
P_D	4 and 64	MAE
P_G	2 and 16	MAE

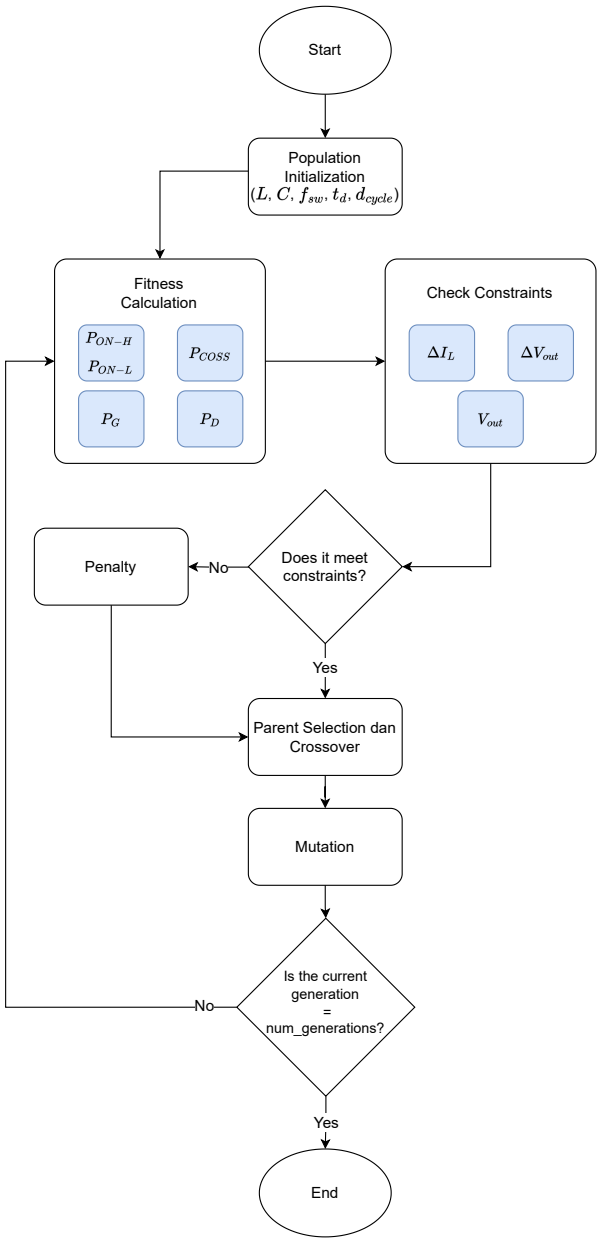


Fig. 3. Genetic Algorithm Block Based on a Machine Learning Model

The first stage of the optimization method is population initialization. This section provides the initial population according to the number of *num_values*. This population is formed by creating an array of discrete values for the parameters to be optimized, namely L, C, f_{sw}, t_{d}, and d_{cycle}. These values are taken from predefined ranges (Table II). This population initialization process is critical to ensure the algorithm has a solid foundation for selection, crossover, and mutation steps.

The next process is to find the fitness value based on the prediction of the machine learning model. Figure 3 shows that the blue-colored block in the Fitness Calculation section

Authorized licensed use limited to: UNIVERSITAS GADJAH MADA. Downloaded on March 17,2025 at 13:32:01 UTC from IEEE Xplore. Restrictions apply.

is a machine learning model that will predict the power loss values to calculate the fitness value based on Equation 3.

$$\min(\text{object_value}) = P_{\text{ON_H}} + P_{\text{ON_L}} + P_{\text{COSS}} + P_{\text{D}} + P_{\text{G}} \quad (3)$$

The value of this objective function (Equation 3) is minimized to get the best solution. Then, each chromosome in the population is checked to ensure it satisfies the predefined constraints (Table I). A large penalty is applied to its fitness value if a chromosome does not satisfy the constraints. This process begins by loading the machine learning model to predict the values of ΔI_{L}, ΔV_{out}, and V_{out}. Each chromosome in the population is then evaluated. A penalty is applied to its fitness value if it does not meet the constraints.

The parent selection and crossover process identifies the optimal parent from the population based on the minimum fitness value and generates new offspring through crossover, ensuring optimal genes are transmitted, and genetic diversity is maintained. Mutations are then introduced to the offspring to preserve genetic diversity further and prevent premature convergence to local optima. New offspring are incorporated into the population, and the process continues until the specified number of generations (*num_generations*) is reached. At this point, the genetic algorithm terminates, saves the optimal result, and replaces the population to continue the evolution and optimization.

III. Conventional Circuit Design Implementation

This section will calculate the parameter values for the synchronous buck converter based on the design in Table I. The parameters to be calculated are L, C, f_{sw}, t_{d}, and d_{cycle}.

A. Duty Cycle

Based on Table I regarding circuit specifications, the values $V_{\text{out}} = 12\,\text{V}$, $V_{\text{in}} = 48\,\text{V}$, and $\eta = 0.90$ are obtained. The duty cycle value can be calculated as follows:

$$D = \frac{12\,\text{V}}{48\,\text{V} \times 0.90} = 0.225 \,\text{or}\, 22.5\% \quad (4)$$

Therefore, the obtained duty cycle value is 0.225 or 22.5%.

B. Dead Time

The dead time value can be determined using the turn-off delay time ($t_{\text{d(off)}}$), the turn-on delay time ($t_{\text{d(on)}}$), and the fall time (t_{f}). For this design, $t_{\text{d(off)}}$ is specified as 68 ns, $t_{\text{d(on)}}$ is specified as 26 ns, and t_{f} is specified as 78 ns. With a safety margin of 1.2, the dead time value is obtained as follows:

$$t_{\text{dead_min}} = \max(68\,\text{ns} + 78\,\text{ns}, 26\,\text{ns}) \times 1.2 = 175.2\,\text{ns} \quad (5)$$

Thus, the minimum dead time ($t_{\text{dead_min}}$) is 175.2 ns.

C. Inductance Value

Before calculating the inductance value, the inductor current ripple (ΔI_{L}) can be calculated as follows.

With $P_{\text{out}} = 100$ W, $V_{\text{out}} = 12$ V, and ripple = 10% or 0.1:

$$\Delta I_{\text{L}} = 0.1 \times \frac{100\ \text{W}}{12\ \text{V}} = 0.833\text{A} \quad (6)$$

TABLE IV
Comparison of Conventional Circuit Parameter Design Results with AI-Based

Parameter	Conventional	AI-Based Approach
L	243.097 μH	285.8 μH
C	21.692 μF	127 μF
f_{sw}	40 kHz	39.841 kHz
t_{d}	175.2 ns	180.90 ns
d_{cycle}	22.5 %	24 %

TABLE V
Simulation Results of Circuit Parameter Design

Parameter	Conventional	AI-Based Approach
I_{L} (A)	7.228	7.719
ΔI_{L} (A)	0.843	0.753
ΔI_{L} (%)	11.661	9.757
V_{out} (V)	10.395	11.127
ΔV_{out} (V)	0.120	0.0187
ΔV_{out} (%)	1.154	0.168
$P_{\text{ON-H}} - P_{\text{ON-L}}$ (W)	0.0634 and 0.229	0.0774 and 0.256
P_{COSS} (W)	0.0249	0.0248
P_{D} (W)	0.0355	0.0389
P_{G} (W)	0.136	0.136
Total Power Loss (W)	0.489	0.533
Volume (cm^3)	7.176	5.732
Efficiency (%)	99.354	99.383

Then, the inductance value can be calculated with $V_{\text{in}} = 48$ V, $V_{\text{out}} = 12$ V, $\Delta I_{\text{L}} = 0.833$ A, $f_{\text{sw}} = 40$ kHz, and efficiency (η) of 90% or 0.9:

$$L = \frac{12\,\text{V} \times (48\,\text{V} - 12\,\text{V})}{0.833\,\text{A} \times 40\,\text{kHz} \times 48\text{V}} \times 0.9 = 243.097\,\mu\text{H} \quad (7)$$

Thus, the required inductance value (L) is 243.097 µH.

D. Capacitance Value

The output voltage ripple (ΔV_{out}) can be obtained based on the set ripple of 1%. Meanwhile, the output current ripple and switching frequency values have been previously obtained.

$$\Delta V_{\text{out}} = 0.01 \times 12\,\text{V} = 0.12\,\text{V} \quad (8)$$

Consequently, the value of C_{out} can be obtained as follows:

$$C_{\text{out}} = \frac{0.833\,\text{A}}{8 \times 40\,\text{kHz} \times 0.12\,\text{V}} = 21.692\,\mu\text{F} \quad (9)$$

Thus, the output capacitance (C_{out}) used is 21.692 µF.

IV. Comparison and Discussion of Circuit Parameter Design

This section will present a comparison and analysis of the results from both methods. Table IV compares the parameter results for both methods.

Based on these parameter values, PLECS simulations can be run to obtain the circuit's performance regarding output voltage, voltage ripple, current ripple, and power loss. Table V summarizes the simulation results for both the conventional design method and the AI-based approach.

The conventional circuit design and the AI-based approach show similar performance in some parameters. The power loss parameters, such as conduction loss, output capacitance

Authorized licensed use limited to: UNIVERSITAS GADJAH MADA. Downloaded on March 17,2025 at 13:32:01 UTC from IEEE Xplore. Restrictions apply.

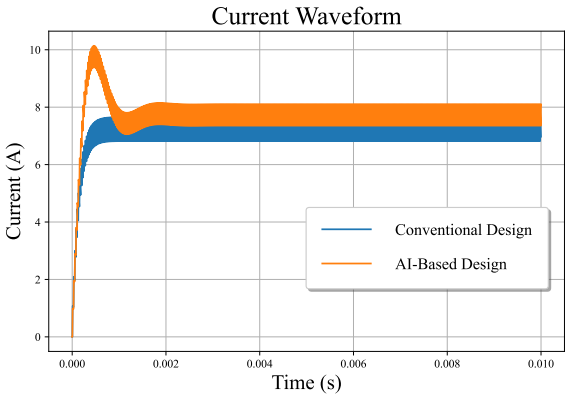


Fig. 4. Current Waveform - Conventional Design and AI-Based Design

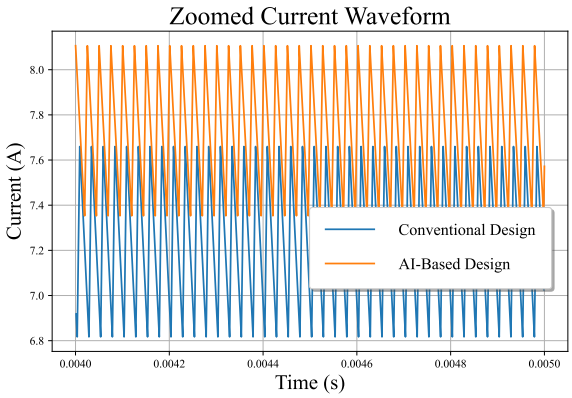


Fig. 5. Current Plot - Conventional Design and AI-Based Design (4 ms - 5 ms)

loss, dead time loss, and gate charge loss, indicate that both methods have comparable efficiency levels (99.354% compared to 99.383%). This minimal difference in efficiency occurs because the inductor and capacitor components used in the PLECS circuit simulation are ideal components, which do not fully capture real-world losses and imperfections. In practical implementations, the difference in efficiency might be more pronounced.

Despite the similar efficiency level, the AI-based approach has several other advantages. The inductor current generated by the AI-based approach is slightly higher with a much smaller current ripple (9.757% compared to 11.661%). Meanwhile, the output voltage of the conventional method has a more significant error (13.37% compared to 7.28%) from the desired output voltage of 12 V (See Table I). Additionally, the volume of the inductor and capacitor from the AI-based approach results (5.732 cm^3) meets the circuit specifications (≤ 7 cm^3), while the conventional design exceeds this volume limit at 7.176 cm^3.

The increase in current ripple is mainly due to the larger inductor (285.8 μH compared to 243.097 μH) selected by the AI-based approach, which more effectively smooths the current fluctuations. Voltage ripple also substantially improves, reducing from 1.154% to 0.168%. This reduction stems from the AI-based approach's selection of a much larger output capacitor (127 μF compared to 21.692 μF), which provides a more stable voltage output by better absorbing current fluctuations.

Then, the PLECS simulation was run to obtain Figure 4, which shows the current plot for both methods. This plot helps observe the performance differences between the conventional method (blue) and the AI-based approach (orange). The average current values of both methods are slightly different, at 7.228 A and 7.719 A (See Table V).

Figure 5 is a zoomed-in current plot for 4 ms - 5 ms. This time range was chosen because the system is in a steady state. The current ripples are pretty significant for both methods, with 11.661% for the conventional method and 9.757% for the AI-based approach. These results indicate that although the AI-based approach is more effective in reducing the current ripple, both methods still produce

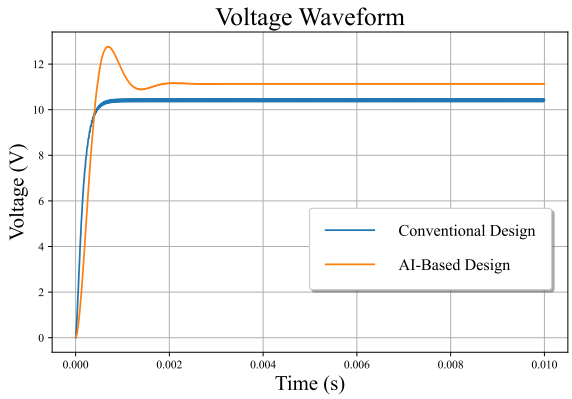


Fig. 6. Voltage Waveform - Conventional Design and AI-Based Design

significant fluctuations in stable conditions.

In addition to the current, the voltage plot is also displayed to analyze the system's state. Figure 6 shows the voltage response for both the conventional and AI-based methods, while Figure 7 presents a zoomed-in version of the voltage plot to focus on the transient state behavior. The blue plot represents the conventional design results, while the orange plot represents the AI-based approach. Both plots have a voltage difference of about 1 V, specifically 11.127 V and 10.395 V. The system's response has a similar pattern to the previous current plot, where a voltage surge occurs during the transient state (AI-based approach).

Although the main focus of this research is on optimizing design parameters in a steady state, it is essential to highlight that AI-based designs exhibit higher current and voltage surges during the transient phase compared to conventional methods, which maintain a more stable and flat response. These surges in AI-based methods may cause increased stress on components such as MOSFETs and inductors, potentially leading to faster component degradation in practical applications. Significant voltage spikes can also pose challenges for sensitive loads that require stable voltage levels. Although these transient characteristics are not the main focus of this research, the observed surges indicate the need for further research on surge mitigation techniques or

Authorized licensed use limited to: UNIVERSITAS GADJAH MADA. Downloaded on March 17,2025 at 13:32:01 UTC from IEEE Xplore. Restrictions apply.

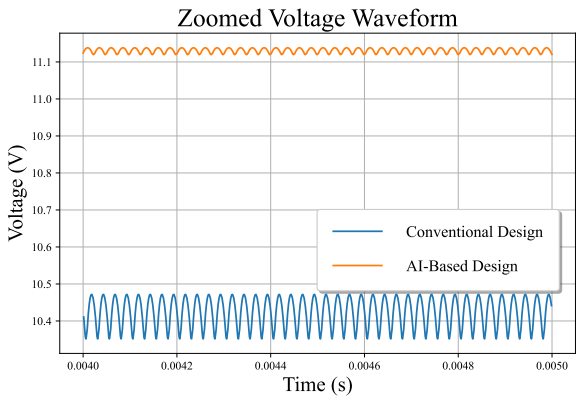


Fig. 7. Voltage Waveform - Conventional Design and AI-Based Design (4 ms - 5 ms)

more effective control strategies to reduce potential negative impacts in real-world applications.

V. Conclusion and Future Works

This research aimed to optimize the circuit parameters of a synchronous buck converter using an AI-based approach. The results of this approach were compared in terms of efficiency, output voltage, output current, and total power loss with the conventional method.

Based on the simulations and analysis conducted, it can be concluded that the AI-based approach for circuit parameter design has a higher efficiency (99.383% vs. 99.354%). Additionally, the output voltage error produced by the AI-based approach is lower (7.28% vs. 13.37%). The output voltage ripple is also significantly lower (0.0187 V vs. 0.120 V). Furthermore, the AI-based approach results in a lower current ripple (0.753 A vs. 0.843 A). Importantly, the volume of the inductor and capacitor from the AI-based approach (5.732 cm^3) meets the circuit specifications (≤ 7 cm^3), while the conventional design exceeds this volume limit at 7.176 cm^3.

Thus, the research objective of optimizing circuit parameters using an AI-based approach has been achieved. These findings indicate that the AI-based circuit parameter design improves the buck converter circuit's efficiency and performance more effectively than the conventional circuit parameter design.

Future work could focus on mitigating the higher transient surges observed in the AI-based method to enhance its suitability for sensitive applications. Additionally, considering parasitic inductance and capacitance effects in the circuit components would provide a more realistic evaluation, as the current research assumes ideal conditions.

References

[1] Z. Gong, B. A. V. D. Ven, K. M. Gupta, C. D. Silva, C. H. Amon, H. J. Bergveld, M. C. Donkers, and O. Trescases, "Distributed control of active cell balancing and low-voltage bus regulation in electric vehicles using hierarchical model-predictive control," *IEEE Transactions on Industrial Electronics*, vol. 67, 2020.

[2] J. A. Sanguesa, V. Torres-Sanz, P. Garrido, F. J. Martinez, and J. M. Marquez-Barja, "A review on electric vehicles: Technologies and challenges," 2021.

[3] S. Habib, M. M. Khan, F. Abbas, A. Ali, M. T. Faiz, F. Ehsan, and H. Tang, "Contemporary trends in power electronics converters for charging solutions of electric vehicles," *CSEE Journal of Power and Energy Systems*, vol. 6, 2020.

[4] S. Thangavel, D. Mohanraj, T. Girijaprasanna, S. Raju, C. Dhanamjayulu, and S. M. Muyeen, "A comprehensive review on electric vehicle: Battery management system, charging station, traction motors," 2023.

[5] X. Li, X. Zhang, and F. Lin, "Multi-objective design of output lc filter for buck converter via the coevolving-amosa algorithm," *IEEE Access*, vol. 9, 2021.

[6] F. Blaabjerg, H. Wang, I. Vernica, B. Liu, and P. Davari, "Reliability of power electronic systems for ev/hev applications," *Proceedings of the IEEE*, vol. 109, 2021.

[7] X. Li, X. Zhang, F. Lin, and F. Blaabjerg, "Artificial-intelligence-based design for circuit parameters of power converters," *IEEE Transactions on Industrial Electronics*, vol. 69, 2022.

[8] Z. Xiao, Z. Li, Z. He, H. Wang, Y. Tang, and A. Luo, "Computer-aided time-domain operation and design optimization of resonant converters," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 11, 2023.

[9] L. F. Costa, Y. Ko, and M. Liserre, "Design and experimental analysis of a modular smart transformer architecture," *2023 IEEE Energy Conversion Congress and Exposition (ECCE)*, pp. 861–868, 10 2023.

[10] Z. Lv, J. Zhuang, J. Wu, L. Liu, Z. Yuan, Z. Jiang, and X. Wang, "Parameter optimization design of the commutation circuit of a hybrid dc-current-limiting circuit breaker," *Energies*, vol. 16, 2023.

[11] Y. Cui, Y. Wang, and X. Ma, "Parameters design and optimization of a high frequency, interleaved, dual-buck, bidirectional, grid-connected converter," *Electronics (Switzerland)*, vol. 8, 2019.

[12] A. Rahmani, E. Azhir, S. Ali, M. Mohammadi, O. H. Ahmed, M. Y. Ghafour, S. H. Ahmed, and M. Hosseinzadeh, "Artificial intelligence approaches and mechanisms for big data analytics: a systematic study," *PeerJ Computer Science*, vol. 7, 2021.

[13] A. Andreta, L. F. L. Villa, Y. Lembeye, and J. C. Crebier, "A novel automated design methodology for power electronics converters," *Electronics (Switzerland)*, vol. 10, 2021.

[14] S. Fan, N. Cao, S. Zhang, J. Li, X. Guo, and X. Zhang, "From specification to topology: Automatic power converter design via reinforcement learning," *IEEE/ACM International Conference on Computer-Aided Design, Digest of Technical Papers, ICCAD*, vol. 2021-November, 2021.

[15] D. Li and Z. Li, "Performance optimization analysis of artificial intelligence system based on multi-objective genetic algorithm," *2023 International Conference on Networking, Informatics and Computing (ICNETIC)*, pp. 535–539, 2023.

[16] S. Ma, D. Li, T. Hu, Y. Xing, Z. Yang, and W. Nai, "Huber loss function based on variable step beetle antennae search algorithm with gaussian direction," in *2020 12th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 1, 2020, pp. 248–251.

Authorized licensed use limited to: UNIVERSITAS GADJAH MADA. Downloaded on March 17,2025 at 13:32:01 UTC from IEEE Xplore. Restrictions apply.