applied sciences

MDPI

*Article*

# Unified Evolutionary Algorithm Framework for Hybrid Power Converter

**Samira Ghorbanpour, Mingyu Seo, Jeong-Ju Park, Musu Kim, Yuwei Jin and Sekyung Han ***

School of Electronic and Electrical Engineering, Kyungpook National University, Daegu 41566, Korea
* Correspondence: skhan@knu.ac.kr

**Abstract:** A significant amount of the literature is focused on converters that supply the required voltage with low input-current ripple to electronic devices. A hybrid converter that combines boost and Cuk converters was developed recently. This hybrid converter achieved a relatively low input-current ripple based on earlier strategies. This paper proposes a new model that simulates a hybrid power converter system using a unified evolutionary algorithm (EA). As part of this paper, we present an improved framework for a hybrid power converter. Moreover, a unified EA is developed to incorporate differential evolution (DE) and genetic algorithm (GA) properties in order to analyze the proposed modified hybrid power converter. This research describes a modified hybrid power converter that optimizes the zero-ripple duty cycle ($D_Z$) through the proposed algorithm for minimizing the input-current ripple. Based on our simulation results, comparing the proposed method with the baseline algorithm reveals that the proposed approach is significantly more efficient than the baseline algorithm and achieves the minimum input-current ripple in different gain values. In addition, we observe that the proposed algorithm performs better than the DE and GA algorithms in terms of obtaining low input-current ripple results. Therefore, the proposed hybrid algorithm is becoming more efficient with hybridization.

**Keywords:** power converter; differential evolution; genetic algorithm; unified framework



**Citation:** Ghorbanpour, S.; Seo, M.; Park, J.-J.; Kim, M.; Jin, Y.; Han, S. Unified Evolutionary Algorithm Framework for Hybrid Power Converter. *Appl. Sci.* **2022**, *12*, 11236. https://doi.org/10.3390/app122111236

Academic Editor: Giovanni Petrone

Received: 8 October 2022
Accepted: 30 October 2022
Published: 5 November 2022



## 1. Introduction

Over the past several years, emphasis has been placed on optimizing advanced electronic systems, which depend on reliable and efficient power supplies. To achieve these objectives, optimization methods are used in the design of power converters [1]. The voltage ratings of the various components in electrical and electronic devices differ. DC–DC converters are electronic circuits that generate the appropriate voltage for a specific element [2]. In general, these are categorized into buck, boost, buck-boost, and Cuk converters. The DC–DC converters are power electronics-based circuits that are capable of varying and regulating the voltage level from input to output. These converters have been used in a wide variety of applications, such as automatic control systems, renewable energy systems [3,4], photovoltaic generation, fuel cell generation, and electric vehicles [2,5] because of their high efficiency, small size, and simple structure.

The converters produce waves with many harmonics. This is an important characteristic. A filter is placed at the output of the converter to reduce these harmonics. It is complex and expensive to filter the output voltage or current of a converter that delivers a square voltage or current by alternation. The pulse-width modulation (PWM) technique has been demonstrated to be the most suitable for a controlled converter. In addition, it effectively neutralizes the output wave [6]. In [6], the authors presented various techniques using controlled PWM. Three PWM strategies with different vertical and horizontal combinations were evaluated in [7]. Thereby, their output harmonics were measured.

Many scholars have recently applied and implemented metaheuristic techniques in the context of complex optimization problems [8,9]. Several metaheuristic algorithms

have been developed. These include DE [10], GA [11], and particle swarm optimization (PSO) [12]. These methods are reliable, straightforward, and simple to implement. Consequently, these are among the preferred approaches [13]. Metaheuristics have been applied in several practical applications [14], and some are used in optimizing relevant problems. These include an improved interlaced boost converter [15]. Problems with power converters have been solved using the PSO algorithm [15–17]. In [15], an interleaved boost converter (IBC) using an optimal Type-III controller is described. In order to design the controller parameters, the classical "k-factor" strategy was employed, followed by the development of an optimal Type-III controller based on PSO. In the paper [16], the authors introduced an optimization method for a proportional-integral PI controller for four-phase interleaved boost converters. The PSO algorithm was used to find the optimal gains for PI controllers [16]. By using the chaos PSO algorithm, the paper [17] simulated and analyzed a boost converter with maximum power point tracking (MPPT).

In [18], a GA was utilized to perform constrained optimization of a DC–DC converter for uninterruptible power supply applications. In [19], the design of a PID controller for controlling a DC–DC boost converter in PV systems was discussed. Here, three controller parameters were adjusted using the GA. A method that optimizes the duty cycle based on a pre-calculated duty cycle by using the GA for a non-inverting buck-boost converter was proposed in [20]. Furthermore, the duty cycles were estimated using a discrete model. An optimization method for synchronized buck converters using discrete variables was presented in [21]. The DE algorithm was used in [22] to design high-power millimeter-wave mode converters. In [23], a variation in the DE algorithm was developed to determine the switching angles of a programmed pulse-width modulation (PPWM)controlled inverter. In [24], a differential evolution strategy was used to solve the optimal switching angle problem in an harmonic elimination pulse-width modulation (HE-PWM). As described in the paper [25], a hybrid converter was presented that was capable of providing both AC and DC loads with one DC input where a voltage source inverter replaced the control switch of the boost converter. An optimized filter design for a boost converter was created using GA in [26]. By using GA, it was possible to reduce the lower-order harmonics of an inverter considerably. Due to the nonlinear characteristics and time-varying features of power converters, GA was applied to optimize PI parameters [27]. The bee colony optimization algorithm generated duty cycles for the Sheppard–Taylor and interleaved converter in [28,29]. In [30], authors constructed and developed a hybrid converter that was intended for marine applications.

An EA based on pulse-width modulation was proposed in [31] for a hybrid interleaved boost–Cuk converter. This paper introduced the pulse-width-modulation technique and described the formula to analyze the influence of independent duty cycles and an input-current ripple; however, differential evolution was employed to minimize the input-current ripple. The approach proposed therein had certain limitations. First, the zero-ripple duty cycle $(D_Z)$ was maintained at a constant. Maintaining $D_Z$ at a constant affected the performance because the input-current ripple estimation depended on $D_Z$. Second, the approach used a basic DE algorithm and failed to utilize the new advances proposed in the DE literature. Motivated by these, the main contributions of this paper are listed as follows:

1. Introduce a modified hybrid converter that simultaneously optimizes three variables (duty cycle $D$, scale factor $k$, and $D_Z$);
2. A new framework to be designed for the hybrid power converter;
3. We proposed a unified EA to minimize the input-current ripple;
4. A hybrid algorithm that combines DE and GA is suggested to enhance the current model's performance.

The remainder of this paper is organized as follows: In Section 2, we provide an overview of the existing hybrid power converters, DE, and GA. Section 3 provides further details on the proposed method. The results are presented in Section 4, and the conclusions are presented in Section 5.

## 2. Background

### 2.1. Existing Hybrid Power Converters

A wide variety of DC–DC converter topologies have been developed for power electronics over the past few years [32,33]. Interleaved converters, like the interleaved boost converter, have proven successful among the various converters. An important aspect of interleaved converters is that all parallel converters need to have an identical duty cycle or voltage gain. Parallel converters share the same input voltage. Therefore, a severe power imbalance occurs when one converter increases the voltage more than the other. A key feature of interleaved converters is that they have a zero input-current ripple at their operation points. For instance, a double-phase interleaved boost converter exhibits zero input-current ripple. Furthermore, hybrid interleaved converters such as the one described in [34] are being investigated. It is feasible to select a zero input-current ripple for this converter. This enables it to operate in the best operating range. Unlike the conventional interleaving converters with an identical duty cycle for all converters, a hybrid converter [34] employ a PWM strategy to assign a duty cycle to each converter. The duty cycles are dependent. Although hybrid converters may have independent duty cycles, an equal gain in voltage may be obtained by any set of duty cycles. This results in various input ripple currents.

The hybrid converters [24,27] are illustrated in Figure 1a. It is a combination of a Cuk converter and a boost converter. Consequently, their input voltages are identical, and the load is connected differentially from the Cuk output to the boost converter output. Owing to the Cuk converter's negative output voltage, the voltage gain is more significant when the two boost converters are interleaved. This is because they have independent outputs and may have different duty cycles. The PWM strategy is illustrated in Figure 1b. Meanwhile, Figure 1c shows the current waveform in a duty cycle where the ripple in the input-current is zero.

The mathematical model of the hybrid power converter can be summarized as follows:

$$V_{C1} = V_{in}\frac{1}{1-D_1}; \quad I_{L1} = I_O\frac{1}{1-D_1} \tag{1}$$

$$V_{C2} = V_{in}\frac{1}{1-D_2}; \quad I_{L2} = I_O\frac{D_2}{1-D_2} \tag{2}$$

$$V_{C3} = V_{in}\frac{1}{1-D_2}; \quad I_{L3} = I_O \tag{3}$$

The duty cycle for each transistor $s_x$ is determined by $D_x$ (x = 1, 2). The voltage $V_{Cy}$ (y = 1, 2, 3) represents the voltage across each capacitor $C_y$. The current $I_{Lz}$ (z = 1, 2, 3) is equal to every inductor's current $L_z$. $I_o$ is the output current. A series connection of $C_1$ and $C_3$ yields the output voltage. The output results are expressed as follows, based on the load resistance $R$:

$$V_o = V_{in}\frac{1}{1-D_1} + V_{in}\frac{D_2}{1-D_2}; \quad I_o = \frac{V_O}{R} \tag{4}$$

As part of the PWM strategy of a hybrid converter [31], the duty cycle of the converter is $s_2$, $d = d_2$. Therefore, the duty cycle of $s_1$ ($d_1$) is expressed as a fraction of d multiplied by k, as follows:

$$d = d_2; \; d_1 = kd \tag{5}$$

The following equation can be obtained by applying similar properties to inductors:

$$L_1 = kL_2 \tag{6}$$

k is determined by the duty cycle wherein zero input-current ripples are preferred. The duty cycle is called $D_Z$, in which a mathematical equation is written as the following:

$$k = \frac{1 - D_Z}{D_Z} \tag{7}$$

A detailed description of the PWM strategy is available in [34]. It is generally straightforward. However, it has drawbacks because it loses a degree of freedom when a duty cycle is proportional to another over the entire range of operations. Considering this limitation in [34], a new PWM strategy was proposed in [31]. The duty cycles can be represented as in Equation (8). Here, k is a variable and not a constant. In [31], the best value was selected at each operation point. The study provided a more detailed description of the current PWM strategy used in this research. In addition, the input-current ripple was analyzed under two conditions: $D > D_Z$ and $D < D_Z$.

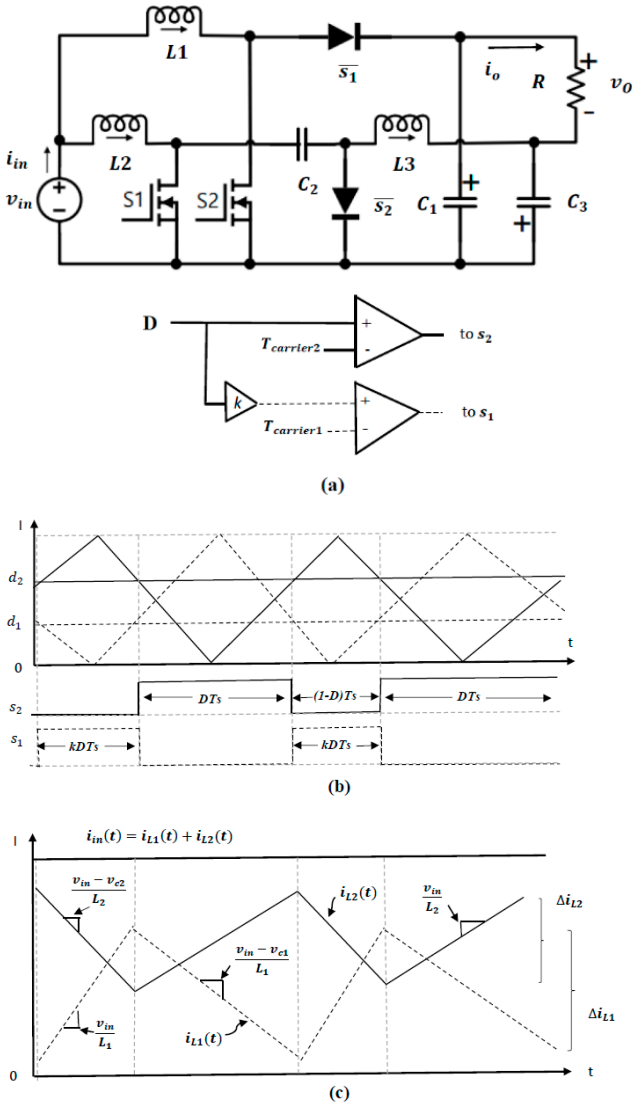


**Figure 1.** Hybrid power converter proposed in [31]. (**a**) Structure, (**b**) PWM strategy, and (**c**) Waveforms of the input current ripple.

### 2.1.1. The Case of $D > D_Z$

When $D > D_Z$ and the switching functions overlap, the maximum ripple may occur in two periods: $(1 - kD)$ TS and $(1 - D)$ $T_S$. Further information concerning this situation is provided in [31].

The input-current ripple at the time interval $(1 - kD)$ TS is defined by Equation (8):

$$\Delta_{ig} = (1 - kD)T_S(\frac{V_{in}}{L_2} + \frac{V_{in} - V_{C1}}{L_1}) \quad (8)$$

We obtain Equation (9) based on Equations (1) and (5):

$$\Delta_{igA} = \frac{V_{in}T_S}{L_2K_L}(k_L - kD - k_LkD) \quad (9)$$

The input-current ripple in the time interval $(1 - D)$ $T_S$ is derived using Equation (10):

$$\Delta_{ig} = (1 - kD)T_S(\frac{V_{in}}{L_1} + \frac{V_{in} - V_{C2}}{L_2}) \quad (10)$$

The following equation is defined based on Equations (2) and (5):

$$\Delta_{igB} = \frac{V_{in}T_S}{L_2K_L}(1 - D - k_LD) \quad (11)$$

2.1.2. The Case of $D < D_Z$

When $D < D_Z$, the switching functions have a dead time (when both transistors are off). Under such conditions, the maximum ripples can occur at two times: $DT_S$ and $kDT_S$ [31].

The input-current ripple during the period of the $DT_S$ can be calculated using the following formula:

$$\Delta_{ig} = DT_S(\frac{V_{in}}{L_2} + \frac{V_{in} - V_{C1}}{L_1}) \quad (12)$$

We obtain Equation (13) based on Equations (1) and (5):

$$\Delta_{igA_2} = \frac{V_{in}T_S}{L_2K_L}\frac{D}{(1 - KD)}(k_L - kD - k_LkD) \quad (13)$$

For the period $kDT_S$, the ripple of the input-current can be expressed as Equation (14):

$$\Delta_{ig} = kDT_S(\frac{V_{in}}{L_1} + \frac{V_{in} - V_{C2}}{L_2}) \quad (14)$$

The following equation can be obtained based on Equations (2) and (5):

$$\Delta_{igB_2} = \frac{V_{in}T_S}{L_2K_L}kD(1 - D - k_LD) \quad (15)$$

Thus, the algorithm should comply with the gain. This can be expressed by Equation (16):

$$G = \frac{1}{1 - KD} + \frac{D}{1 - D} \quad (16)$$

*2.2. Differential Evolution (DE)*

DE is a well-known evolutionary algorithm (EA) which incorporates mutation, crossover or recombination, and selection [10]. The DE algorithm consists of a set of search agents $X = \{x_1, x_2, \ldots, x_m\}$. Here, $x_i$ represents a potential solution that improves gradually during the evolution. The DE algorithm randomly initializes the population and then, evaluates the fitness function:

$$X_i(\text{j}) = X_{iL} + \text{rand } [0, 1].\ (X_{iH} - X_{iL}) \quad (17)$$

where $X_{iL}$ is the lower bound and $X_{iH}$ is the upper bound. $X_i$(j) is selected randomly from the interval [$X_{iL}$, $X_{iH}$]. Subsequently, mutation, recombination (crossover), and selection

are used to improve individuals in an iterative process. Mutation is used as the search tool. Mutation operations create mutant vectors by adding the weighted difference between two randomly selected vectors to form a third vector. In this study, we applied the DE/current-to-best/1 mutation strategy. It is expressed as follows:

$$O_{cur} = X_{cur} + F(P_{best} - X_{cur}) + F(X_{r1} - X_{r2}) \quad (18)$$

where $X_{cur}$ and $O_{cur}$ indicate the current parent and its corresponding offspring solution, respectively. F denotes the scaling factor, and $X_{r1}$ and $X_{r2}$ denote the random solutions.

In crossover operations, the parameters of the mutant vector are mixed with those of the target vector to generate a new vector, which is commonly referred to as a trial vector. A trial vector can replace the target vector in the next generation if it is more suitable than the target vector. We used a binomial crossover in this study. It is described as follows:

$$U_{j,i,G+1} = \begin{cases} V_{j,i,G+1} & if\ a\ rand \leq CR \\ X_{j,i,G} & Otherwise \end{cases}$$
$$j = 1, 2, \ldots, D;\ i = 1, 2, \ldots, Np \quad (19)$$

where D is the number of genes. *CR* denotes the crossover factor, which is typically between zero and one.

Selection guides the search for potential regions within the search space. In the selection step, the initial step is a one-to-one competition analysis (local search). The next step is to select the best individual in the population (global search):

$$X_{i,G+1} = arg - \max\{f(X_{i,G}), f(U_{i,G+1})\}$$
$$X_{b,G+1} = arg - \max\{f(X_{i,G+1})\} \quad (20)$$
$$i = 1, 2, \ldots, Np.$$

### *2.3. Genetic Algorithm (GA)*

The GA was developed by John Holland in 1970 and has been demonstrated to be an efficient tool [35]. In contrast to other evolutionary computation techniques, such as PSO, the GA identifies the optimal solution from the solution space. In general, GAs are based on groups of points rather than an individual point. This can be used to solve optimization problems through natural selection and genetics. GAs can handle a population of many individuals to achieve the best solution by using a randomly initialized population. The initial population of the GA is generated at random. This is similar to DE (Equation (17)). After a parent population is generated, mating selection is performed to select better parent individuals to generate offspring. Variation (crossover and mutation) operators are used to produce offspring solutions. Subsequently, the parent and offspring populations are combined. The selection mechanism is then performed based on the objective function. The combined population is sorted according to the optimization criteria of the objective function, and better N solutions are preserved for the next generation. This process is repeated until the termination condition is satisfied.

## 3. Proposed Methodology

This section briefly describes the proposed model and its key components. First, we describe the modified model for a hybrid power converter. Then, we describe the proposed unified EA to solve the modified model.

### *3.1. Modified Hybrid Power Converter*

The hybrid power converter approach proposed in [31] aims to minimize input-current ripple. However, they considered two optimization parameters (*D* and *k*) for the power converter model. The power converter model proposed in [31] considers two periods. Considering these periods, the input-current ripple may attain a maximum value. Therefore,

it is necessary to formulate the objective function in order to minimize the maximum value of the input-current ripple. If $D > D_Z$ [31], this procedure minimizes a maximum ripple in the input-current. Similarly, the largest input-current ripple is minimized when $D < D_Z$ [31]. Furthermore, they considered several restrictions, including achieving the desired voltage gain; evaluating the switch time, and searching the domain, which includes valid values for $D$ and $k$. However, the objective-function estimation is highly sensitive to $D_Z$. Hence, in our proposed model, we aim to optimize three variables ($D$, $k$, and $D_Z$) simultaneously and analyze its performance. The range of the three variables is between 0 and 1, according to the below equations. The modified hybrid power converter can be expressed as:

$$\min_{D,k,Dz \in R} f(D,k,Dz) = \begin{cases} f_1(D,k,Dz), & D > Dz \\ f_2(D,k,Dz), & otherwise \end{cases} \tag{21}$$

Subject to;

$$G \leq \frac{1}{1-kD} + \frac{D}{1-D} \leq G + t \tag{22}$$

$$0 \leq D \leq 1 \tag{23}$$

$$0 \leq k \leq 1 \tag{24}$$

$$0 \leq D_Z \leq 1 \tag{25}$$

Thus, $f_1(D, k, D_Z)$ and $f_2(D, k, D_Z)$ are formulated as follows:

$$f_1(D,k,Dz) = \begin{bmatrix} |\Delta_{igA_1}|, & |\Delta_{igA_1}| > |\Delta_{igB_1}| \\ |\Delta_{igB_1}|, & Otherwise. \end{bmatrix} \tag{26}$$

$$f_2(D,k,Dz) = \begin{bmatrix} |\Delta_{igA_2}|, & |\Delta_{igA_2}| > |\Delta_{igB_2}| \\ |\Delta_{igB_2}|, & Otherwise. \end{bmatrix} \tag{27}$$

$D_Z$ was maintained at a constant of 0.6 in [31]. It plays a significant role in the calculation of the objective functions. Consequently, its optimization would yield better results. The effectiveness of optimizing $D_Z$ is verified in the simulation section.

### 3.2. Unified Evolutionary Algorithm

In this study, we combined the advantages of two popular EAs: DE and GA. Herein, DE operators are used to produce the offspring population, which helps in its exploitation. Then, a GA selection mechanism is used to preserve elite individuals. First, an initial population of size $N$ is generated randomly according to Equation (17). Unlike in the previous approaches, three variables are considered in the proposed approach [31]. Therefore, an individual $X_i$ is defined as:

$$X_i = \{D, k, D_Z\} \tag{28}$$

The candidate solutions are evaluated based on the $X_i$ in the objective function presented in Equation (21). Thus, Equation (21) can be expressed as:

$$\min_{X_i \in R} f(X_i) = \begin{cases} f_1(D,k,Dz), & D > Dz \\ f_2(D,k,Dz), & otherwise \end{cases} \tag{29}$$

The proposed model involves a constrained optimization problem. Therefore, it is necessary to apply a penalty function to assess the population in the objective function. Feasible solutions can be determined through the penalty function. The penalty function for the power converter is formulated as follows:

$$h = \left| wc\left( G - \frac{1}{1 - X_{i1}X_{i2}} - \frac{X_{i1}}{1 - X_{i1}} \right) \right| \tag{30}$$

A constant factor $w$ is used to determine the degree to which a candidate's fitness value is penalized. $c$ represents the violated constraint indicator, with $c \in \{0, 1\}$. Meanwhile, $X_{i1} = D$, and $X_{i2} = k$. Equation (29) can be revised as follows based on the penalty function for evaluating the population:

$$\min_{X_i \in R} f(X_i) + \mathrm{h} \tag{31}$$

After evaluation, the offspring are generated using the DE mutation "*DE/current-to-best/1*". The "*DE/current-to-best/1*" to generate trail vector is performed as described in Equation (18). After the trial vector is generated, the offspring solution is produced using a binomial crossover. Binomial crossover is performed according to Equation (19). Next, the offspring and parent populations are combined. The combined population is then sorted according to the optimization criteria of the objective function (i.e., the input-current ripple was minimized in this study). Hence, the combined population is sorted in the ascending order of the objective function. Then, $N$ solutions in the sorting order are preserved for the next generations. Figure 2 presents a schematic of the general framework of the proposed method.

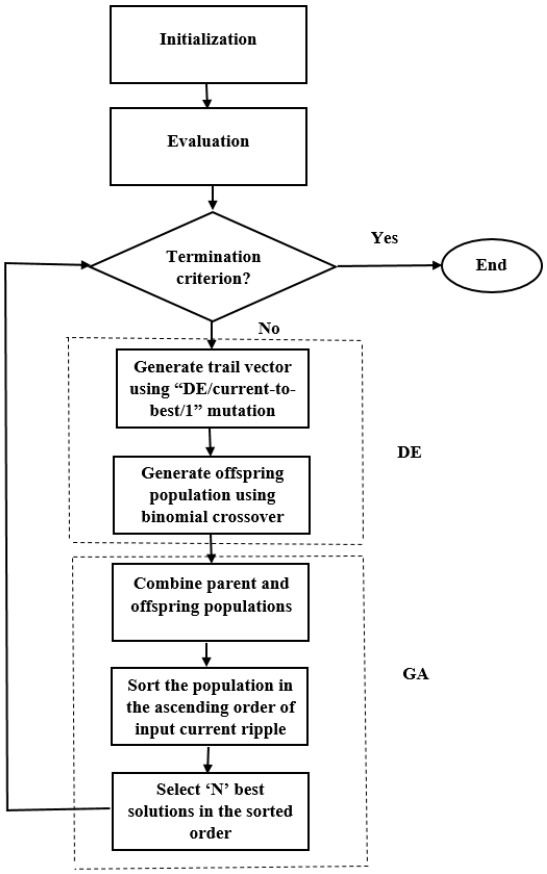


**Figure 2.** General framework of proposed method.

## 4. Simulation Setup and Results

This section describes the parameter settings for the algorithms being compared and the hybrid power converter. We conducted several experiments to evaluate the performance of our proposed method. It was compared with the baseline method [31], DE and GA algorithms. We considered various desired voltage gains as part of the evaluation to demonstrate the efficiency of our approach. Simulations were performed on a PC with an Intel Core i5-10400 CPU processor operating at 2.90 GHz and a 64-bit version of Windows 10 with MATLAB 2021b. In the experiments, the gain value G was varied from 3 to 6 at intervals of 0.1.

### 4.1. Comparison between the Proposed Method and Baseline Algorithm

The approach proposed in [31] was selected as the baseline algorithm to validate the performance of the proposed algorithm. The control parameters for the baseline algorithm were set as follows: the mutation scaling factor F was selected from a uniform distribution between 0.2 and 0.8; the crossover rate was set as 0.2 and w was maintained as 10, as recommended in [31]. F, CR, and w were set as 0.3, 0.75, and 10, respectively, for the proposed method. The population size and the number of generations were set as 20 and 50, respectively. Because the EA is a stochastic algorithm, these results are based on the average value collected from 30 independent executions. Moreover, Table 1 provides the relevant parameters for the hybrid power converter.

**Table 1.** Parameters for simulation setting.

| | |
|---|---|
| Input Voltage $V_{in}$ | 20 V |
| Inductor Factor kL | 0.6666 |
| Switching Frequency fS | 50 kHz |
| Output Resistance R | 60 Ω |
| $L_1$ | 66 µH |
| $L_2$ | 100 µH |

The mean values of the input-current ripple obtained over 30 independent runs are presented in Table 2. In addition, we have provided the hybrid power converter parameters $D$, $k$, and $D_Z$. It is evident from the results presented in this table that the proposed method achieved a lower input–current ripple value in all the cases. It achieved a significant improvement in the input-current ripple value as $G$ increased. Table 2 reveals that the optimization of $D_Z$ simultaneously optimized both $D$ and $k$. The simulation results of the proposed method are better than those of the baseline algorithm because of the optimization of $D_Z$. The baseline algorithm maintained $D_Z$ at a constant of 0.6. The proposed method performed better for each gain instance at different values of $D_Z$. For example, the proposed method optimized $D_Z$ to 0.7998 for a gain of *3.1* and 0.9519 for a gain of *4*. Thus, the optimization of $D_Z$ improved the performance of the model. In the summary of Table 2, the proposed hybrid algorithm (combines DE and GA) method performed better than a baseline (basic DE) model. Therefore, due to hybridization, the proposed method minimizes input-current ripple better than the baseline method, demonstrating its efficiency. In particular, as the G value increases, the efficiency of the proposed method increases significantly because it incorporates the properties of DE and GA.

**Table 2.** Comparison of proposed method with baseline algorithm.

| Gain | Methodology | Input-Current Ripple ($\Delta_{ig}$) | $D$ | $k$ | $D_Z$ |
|---|---|---|---|---|---|
| 3 | Proposed | **0.0758** | 0.5807 | 0.6751 | 0.9999 |
| | Baseline | 0.0844 | 0.5806 | 0.6753 | 0.6 |
| 3.1 | Proposed | **0.0203** | 0.5951 | 0.6688 | 0.7998 |
| | Baseline | 0.0520 | 0.5950 | 0.6689 | 0.6 |
| 3.166 | Proposed | $\mathbf{1.32 \times 10^{-5}}$ | 0.6000 | 0.6666 | 0.7888 |
| | Baseline | 0.0105 | 0.6000 | 0.6667 | 0.6 |
| 3.2 | Proposed | **0.0180** | 0.6045 | 0.6646 | 0.9999 |
| | Baseline | 0.0508 | 0.6037 | 0.6685 | 0.6 |
| 3.3 | Proposed | **0.0710** | 0.6175 | 0.6588 | 0.7010 |
| | Baseline | 0.1527 | 0.6144 | 0.6744 | 0.6 |

**Table 2.** *Cont.*

| Gain | Methodology | Input-Current Ripple ($\Delta_{ig}$) | $D$ | $k$ | $D_Z$ |
|---|---|---|---|---|---|
| 3.4 | Proposed | **0.1265** | 0.6298 | 0.6533 | 0.8508 |
| | Baseline | 0.2548 | 0.6245 | 0.6794 | 0.6 |
| 3.5 | Proposed | **0.1721** | 0.6414 | 0.6480 | 0.9180 |
| | Baseline | 0.3562 | 0.6342 | 0.6847 | 0.6 |
| 3.6 | Proposed | **0.2271** | 0.6524 | 0.6430 | 0.8303 |
| | Baseline | 0.4504 | 0.6435 | 0.6886 | 0.6 |
| 3.7 | Proposed | **0.2662** | 0.6629 | 0.6383 | 0.9999 |
| | Baseline | 0.5339 | 0.6524 | 0.6923 | 0.6 |
| 3.8 | Proposed | **0.3204** | 0.6728 | 0.6338 | 0.9334 |
| | Baseline | 0.6255 | 0.6607 | 0.6971 | 0.6 |
| 3.9 | Proposed | **0.3533** | 0.6823 | 0.6295 | 0.9759 |
| | Baseline | 0.7045 | 0.6687 | 0.7007 | 0.6 |
| 4 | Proposed | **0.3944** | 0.7081 | 0.5149 | 0.9519 |
| | Baseline | 0.7873 | 0.6768 | 0.7031 | 0.6 |
| 4.1 | Proposed | **0.4234** | 0.7169 | 0.5054 | 0.8247 |
| | Baseline | 0.8646 | 0.6840 | 0.7067 | 0.6 |
| 4.2 | Proposed | **0.4657** | 0.7250 | 0.4968 | 0.9966 |
| | Baseline | 0.9416 | 0.6912 | 0.7101 | 0.6 |
| 4.3 | Proposed | **0.4758** | 0.7327 | 0.4890 | 0.9602 |
| | Baseline | 1.0116 | 0.6980 | 0.7130 | 0.6 |
| 4.4 | Proposed | **0.5354** | 0.7400 | 0.4819 | 0.9141 |
| | Baseline | 1.0635 | 0.7045 | 0.7156 | 0.6 |
| 4.5 | Proposed | **0.5407** | 0.7468 | 0.4753 | 0.8284 |
| | Baseline | 1.1519 | 0.7108 | 0.7185 | 0.6 |
| 4.6 | Proposed | **0.5418** | 0.7533 | 0.4693 | 0.8076 |
| | Baseline | 1.1947 | 0.7169 | 0.7205 | 0.6 |
| 4.7 | Proposed | **0.5613** | 0.7594 | 0.4636 | 0.8536 |
| | Baseline | 1.2526 | 0.7229 | 0.7218 | 0.6 |
| 4.8 | Proposed | **0.5797** | 0.7652 | 0.4584 | 0.9999 |
| | Baseline | 1.3181 | 0.7285 | 0.7253 | 0.6 |
| 4.9 | Proposed | **0.6217** | 0.7708 | 0.4536 | 0.9999 |
| | Baseline | 1.364 | 0.7339 | 0.7273 | 0.6 |
| 5 | Proposed | **0.6652** | 0.7760 | 0.4490 | 0.9999 |
| | Baseline | 1.4482 | 0.7392 | 0.7281 | 0.6 |
| 5.1 | Proposed | **0.7102** | 0.7811 | 0.4448 | 0.9999 |
| | Baseline | 1.4702 | 0.7444 | 0.7296 | 0.6 |
| 5.2 | Proposed | **0.7848** | 0.7859 | 0.4408 | 0.9999 |
| | Baseline | 1.5316 | 0.7492 | 0.7331 | 0.6 |
| 5.3 | Proposed | **0.7166** | 0.7904 | 0.4371 | 0.9999 |
| | Baseline | 1.5812 | 0.7541 | 0.7329 | 0.6 |
| 5.4 | Proposed | **0.6714** | 0.7948 | 0.4336 | 0.9999 |
| | Baseline | 1.6132 | 0.7587 | 0.7342 | 0.6 |
| 5.5 | Proposed | **0.7472** | 0.7990 | 0.4303 | 0.9999 |
| | Baseline | 1.6585 | 0.7628 | 0.7376 | 0.6 |
| 5.6 | Proposed | **0.7614** | 0.8031 | 0.4271 | 0.9426 |
| | Baseline | 1.7035 | 0.7673 | 0.7380 | 0.6 |
| 5.7 | Proposed | **0.7751** | 0.8069 | 0.4241 | 0.9999 |
| | Baseline | 1.7547 | 0.7716 | 0.7403 | 0.6 |

**Table 2.** *Cont.*

| Gain | Methodology | Input-Current Ripple ($\Delta_{ig}$) | $D$ | $k$ | $D_Z$ |
|---|---|---|---|---|---|
| 5.8 | Proposed | **0.8227** | 0.8107 | 0.4213 | 0.9999 |
| | Baseline | 1.778 | 0.7752 | 0.7416 | 0.6 |
| 5.9 | Proposed | **0.8362** | 0.8142 | 0.4186 | 0.9995 |
| | Baseline | 1.8344 | 0.7793 | 0.7419 | 0.6 |
| 6 | Proposed | **0.8493** | 0.8177 | 0.4161 | 0.9999 |
| | Baseline | 1.8721 | 0.7831 | 0.7442 | 0.6 |

Furthermore, we compared the baseline and proposed methods as shown in Figure 3 to demonstrate the effectiveness of the proposed method. Figure 3a shows the input-current ripple values of the compared algorithms. It indicates that the proposed technique can attain smaller input-current ripple values over its operating area. Moreover, in most cases, the models differ significantly, except in one case, which demonstrates an almost similar performance (gain = 3).

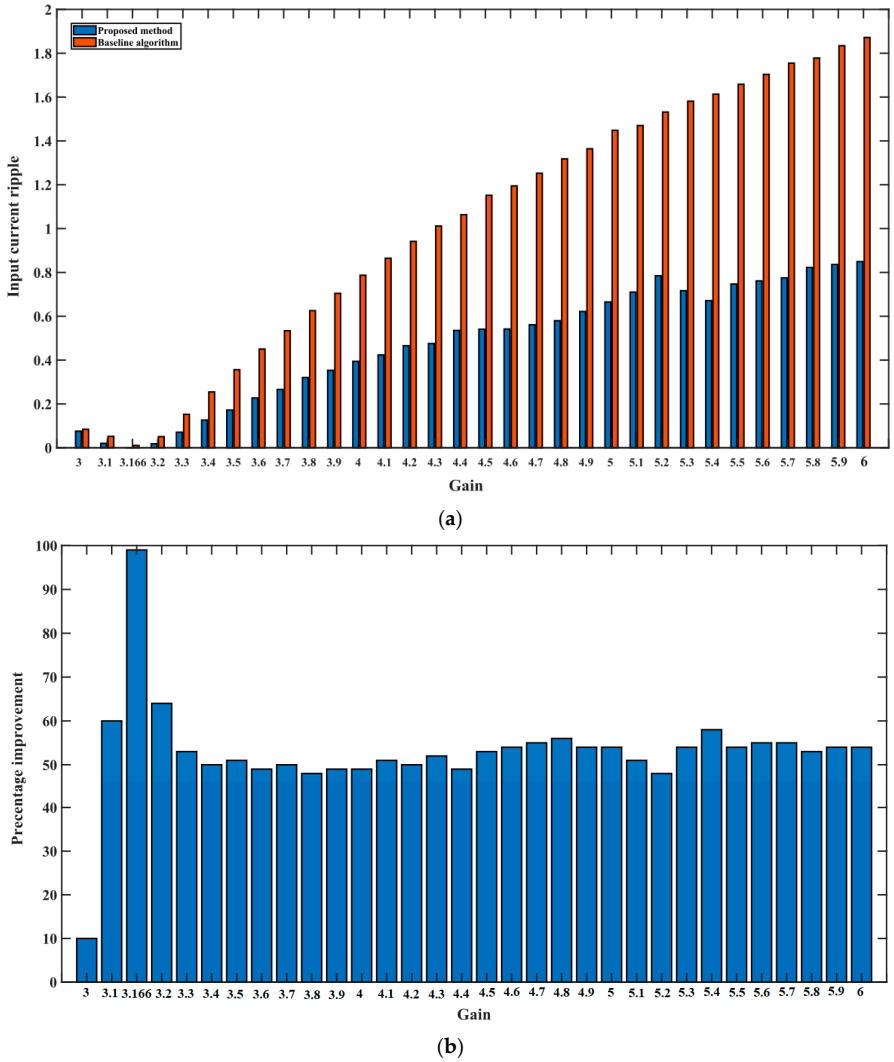


**Figure 3.** (**a**) Comparison between proposed method and baseline algorithm based on input-current ripple, (**b**) Percentage improvement of proposed method compared with baseline algorithm.

Figure 3b illustrates the percentage improvement of the proposed method compared with the baseline algorithm. It is evident from Figure 3b that the proposed method achieved

an improvement of almost 50% in most cases. The performance of the proposed algorithm improved by almost 100% for a gain of 3.166.

In Figure 4, the proposed method is evaluated using a convergence graph that illustrates its speed performance. We consider ten operation points in the test to estimate the speed with which the optimization algorithm attains the optimal value. Figure 4 exhibits the input-current ripple obtained at each generation for gain values of 3.4, 3.6, 3.8, 4, 4.3, 4.7, 5, 5.3, 5.7, and 6 for the baseline and proposed methods. The gain values were selected based on the most significant and relevant points. It is evident from Figure 4 that the proposed method performed significantly better than the baseline method. In addition, it exhibited a speed performance higher than that of the previous method.

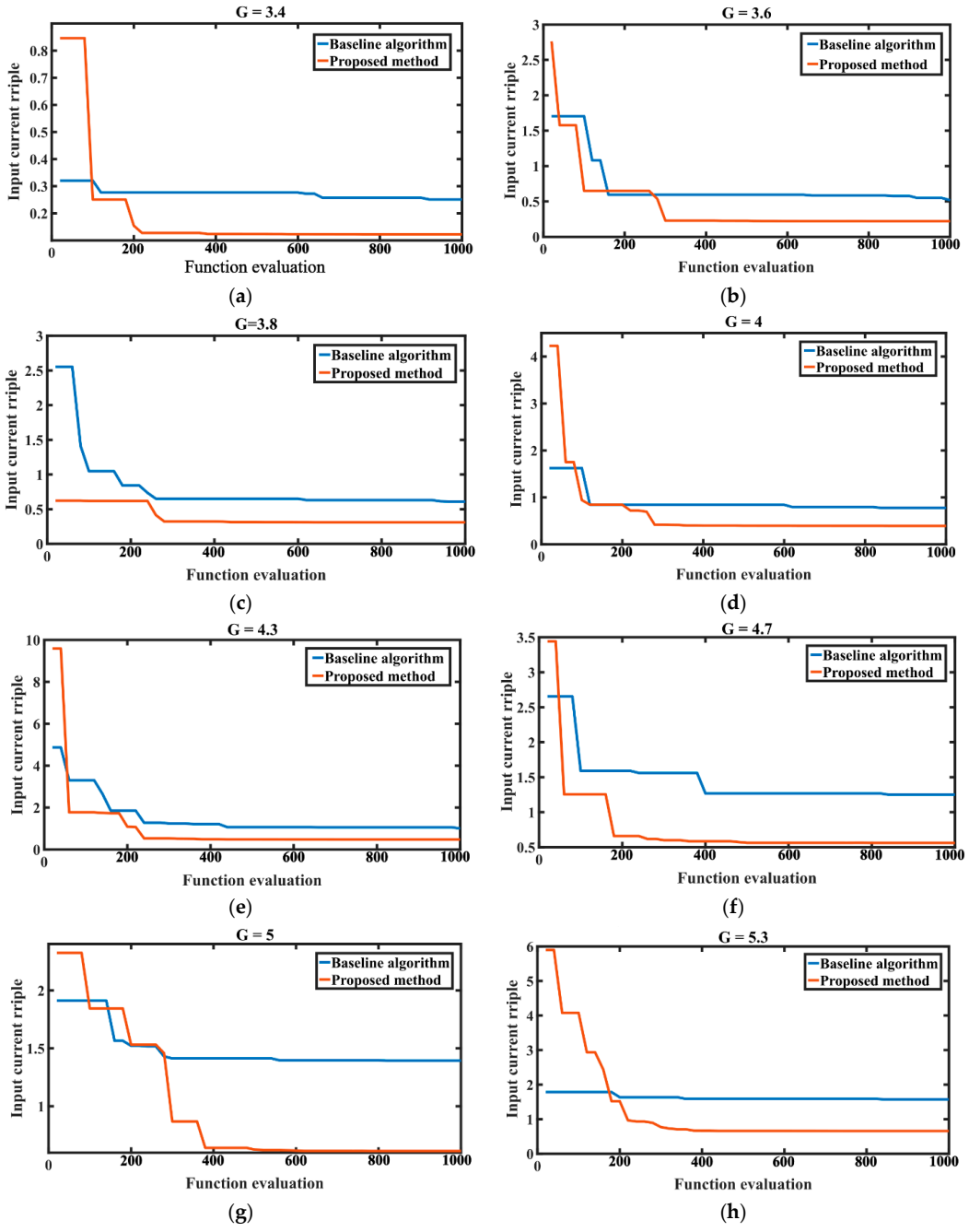


**Figure 4.** *Cont.*

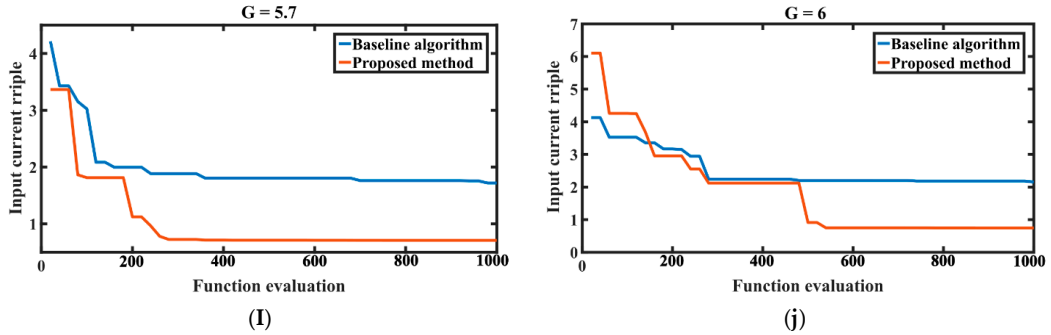


**Figure 4.** Convergence graphs of proposed (red) and baseline (blue) methods with different gain values. (**a**) G = 3.4, (**b**) G = 3.6, (**c**) G = 3.8, (**d**) G = 4, (**e**) G = 4.3, (**f**) G = 4.7, (**g**) G = 5, (**h**) G = 5.3, (**I**) G = 5.7, and (**j**) G = 6.

### *4.2. Further Analysis on the Proposed Method*

In this section, we analyze the performance of the proposed unified EA with differential evolution [10] (DE and DE*) and the genetic algorithm [35] to demonstrate its superiority. We have presented the obtained input-current ripple ($\Delta_{ig}$) for the proposed hybrid approach, DE, DE*, and GA, in Table 3. As shown in Table 3, DE represents the original differential algorithm with the "DE/rand/bin" and binomial crossover. As mentioned earlier, in the proposed unified EA, the "DE/current-to-best/1" and binomial crossover are employed. For better analysis, we have included DE* in the simulation setup that replaces the "DE/rand/bin" mutation with the "DE/current-to-best/1" mutation. For a fair comparison, the crossover rate (CR) and mutation scaling factor (F) are maintained identically for DE, DE*, and unified EA as CR = 0.75 and F = 0.3. The GA employs a real-encoding crossover and a polynomial mutation. The mutation factor F is chosen from real random numbers as uniformly distributed between 0 and 1, and CR ∈ [0, 1], in GA. We set the population size and number of generations to 20 and 50, respectively.

**Table 3.** Comparison of proposed method with DE and GA algorithms.

| Gain | Input-Current Ripple ($\Delta_{ig}$) | | | |
|---|---|---|---|---|
| | **Proposed** | **DE** | **DE*** | **GA** |
| 3 | **0.0758** | 0.0903 | 0.8000 | 0.2745 |
| 3.1 | **0.0203** | 0.0368 | 0.0338 | 0.2478 |
| 3.166 | $\mathbf{1.32 \times 10^{-5}}$ | 0.0098 | 0.0031 | 0.2214 |
| 3.2 | **0.0180** | 0.0303 | 0.0213 | 0.1413 |
| 3.3 | **0.0710** | 0.0811 | 0.0892 | 0.3102 |
| 3.4 | **0.1265** | 0.1369 | 0.1353 | 0.4496 |
| 3.5 | **0.1721** | 0.1920 | 0.1857 | 0.4079 |
| 3.6 | **0.2271** | 0.2304 | 0.2348 | 0.4199 |
| 3.7 | **0.2662** | 0.2906 | 0.2774 | 0.5628 |
| 3.8 | **0.3204** | 0.3318 | 0.3345 | 0.5955 |
| 3.9 | **0.3533** | 0.3654 | 0.3699 | 0.7023 |
| 4 | **0.3944** | 0.4157 | 0.4170 | 0.6418 |
| 4.1 | **0.4234** | 0.4461 | 0.4475 | 0.6955 |
| 4.2 | **0.4657** | 0.4751 | 0.4756 | 0.8467 |
| 4.3 | **0.4758** | 0.5040 | 0.4921 | 0.9972 |
| 4.4 | **0.5354** | 0.5485 | 0.5557 | 1.0329 |
| 4.5 | **0.5407** | 0.5801 | 0.5592 | 1.0595 |
| 4.6 | **0.5418** | 0.6622 | 0.5698 | 1.0270 |
| 4.7 | **0.5613** | 0.5925 | 0.5795 | 1.2939 |

**Table 3.** *Cont.*

| Gain | Input-Current Ripple ($\Delta_{ig}$) | | | |
|---|---|---|---|---|
| | **Proposed** | **DE** | **DE*** | **GA** |
| 4.8 | **0.5797** | 0.6535 | 0.5910 | 1.0793 |
| 4.9 | **0.6217** | 0.7050 | 0.6450 | 1.2583 |
| 5 | **0.6652** | 0.6843 | 0.6792 | 1.3074 |
| 5.1 | **0.7102** | 0.7431 | 0.7441 | 1.3449 |
| 5.2 | **0.7848** | 0.7932 | 0.7936 | 1.3159 |
| 5.3 | **0.7166** | 0.7579 | 0.7214 | 1.3645 |
| 5.4 | **0.6714** | 0.8528 | 0.6949 | 1.3718 |
| 5.5 | **0.7472** | 0.7864 | 0.7645 | 1.5165 |
| 5.6 | **0.7614** | 0.9056 | 0.7806 | 1.5898 |
| 5.7 | **0.7751** | 0.7966 | 0.7888 | 1.4792 |
| 5.8 | **0.8227** | 0.8491 | 0.8499 | 1.7313 |
| 5.9 | **0.8362** | 0.9011 | 0.8652 | 1.6052 |
| 6 | **0.8493** | 0.9514 | 0.8783 | 1.4860 |

As shown in Table 3, the mean values of the input-current ripple over 30 runs are given. Based on the results in Table 3, the proposed method resulted in lower input–current ripple values than DE, DE*, and GA in all cases. The proposed method has better simulation results than DE, DE*, and GA algorithms because our proposed hybrid algorithm incorporates the properties of DE and GA. Additionally, the efficiency of the proposed hybrid algorithm is increasing with hybridization. The hybridization of the proposed method can potentially improve the current model's performance and minimizes input-current ripple. To conclude, a hybrid algorithm can improve the method's performance (convergence speed) and the quality of the solutions.

## 5. Conclusions

In this study, we developed a hybrid power converter model based on a unified EA. Three power converter variables were optimized simultaneously to increase the efficiency of the model. Furthermore, we proposed a unified EA that integrated the properties of DE and genetic GA to evaluate the modified hybrid power converter. This study employed the unified EA to optimize the zero-ripple duty cycle ($D_Z$) in order to minimize the input-current ripple. Based on the simulation results obtained in this study, the proposed method achieved a lower input-current ripple value across the operating range when compared to the baseline method. By analyzing the percentage improvement results that were found, the proposed method improved the majority of cases by nearly 50%. In addition, the proposed algorithm was compared with DE and GA, and the simulation results indicated that the proposed algorithm obtained a low input-current ripple when compared to those with different gain values. Due to this, the efficiency of the proposed method and the hybridization of the current model was proven. Consequently, the proposed algorithm outperformed the baseline technique, DE, and the GA algorithms. In addition, it performed better than the baseline algorithm in terms of speed.

In the future, we intend to optimize other parameters of the power converter (such as the indicator factor, switching frequency, and output resistance) to enhance the performance. Additionally, we would like to apply the hybrid algorithm to other hybrid power converters.

**Author Contributions:** Conceptualization, S.G. and S.H.; methodology, S.G. and S.H.; software, S.G.; validation, S.G.; formal analysis, S.G., M.S., J.-J.P., M.K. and Y.J.; investigation, S.G., M.S., J.-J.P. and Y.J.; resources, S.G.; data curation, S.G., J.-J.P. and M.K.; writing—original draft preparation, S.G., M.S. and Y.J.; writing—review and editing, S.G.; visualization, S.G. and M.K.; supervision, S.H.; project administration, S.H.; funding acquisition, S.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Trade, Industry & Energy (MOTIE) of the Republic of Korea (No. 20202010600010).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable, the study does not report any data.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Rahal-Arabi, T.; Park, H.-J.; Hahn, J. Power delivery for the next generation mobile platform. In Proceedings of the 2008 Electrical Design of Advanced Packaging and Systems Symposium, Seoul, Korea, 10–12 December 2008.
2. Williams, B.W. Unified Synthesis of Tapped-Inductor DC-to-DC Converters. *IEEE Trans. Power Electron.* **2013**, *29*, 5370–5383. [CrossRef]
3. Pavlovic, T.; Bjazic, T.; Ban, Z. Simplified Averaged Models of DC–DC Power Converters Suitable for Controller Design and Microgrid Simulation. *IEEE Trans. Power Electron.* **2012**, *28*, 3266–3275. [CrossRef]
4. Hegazy, O.; Van Mierlo, J.; Lataire, P. Analysis, Modeling, and Implementation of a Multidevice Interleaved DC/DC Converter for Fuel Cell Hybrid Electric Vehicles. *IEEE Trans. Power Electron.* **2012**, *27*, 4445–4458. [CrossRef]
5. Rosas-Caro, J.C.; Ramirez, J.M.; Peng, F.Z.; Valderrabano, A. A DC–DC multilevel boost converter. *IET Power Electron.* **2010**, *3*, 129–137. [CrossRef]
6. Berrezzek, F. A study of new techniques of controlled PWM inverters. *Eur. J. Sci. Res.* **2009**, *32*, 77–87.
7. Carrara, G.; Gardella, S.; Marchesoni, M.; Salutari, R.; Sciutto, G. A new multilevel PWM method: A theoretical analysis. *IEEE Trans. Power Electron.* **1992**, *7*, 497–505. [CrossRef]
8. Palakonda, V.; Awad, N.H.; Mallipeddi, R.; Ali, M.; Veluvolu, K.C.; Suganthan, P.N. Differential evolution with stochastic selection for uncertain environments: A smart grid application. In Proceedings of the 2018 IEEE Congress on Evolutionary Computation (CEC), Rio de Janeiro, Brazil, 8–13 July 2018. [CrossRef]
9. Ramlan, F.W.; Palakonda, V.; Mallipeddi, R. Differential Evolutionary (DE) Based Interactive Recoloring Based on YUV Based Edge Detection for Interior Design. In Proceedings of the 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 16–18 October 2019. [CrossRef]
10. Storn, R.; Price, K. Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **1997**, *11*, 341–359. [CrossRef]
11. Holland, J.H. *Adaptation in Natural and Artificial Systems*, 1st ed.; University of Michigan Press: Ann Arbor, MI, USA, 1975.
12. Eberhart, R.; Kennedy, J. A new optimizer using particle swarm theory. In Proceedings of the MHS'95, Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, 4–6 October 1995.
13. Ghorbanpour, S.; Pamulapati, T.; Mallipeddi, R. Swarm and evolutionary algorithms for energy disaggregation: Challenges and prospects. *Int. J. Bio-Inspired Comput.* **2021**, *17*, 215–226. [CrossRef]
14. Ghorbanpour, S.; Pamulapati, T.; Mallipeddi, R.; Lee, M. Energy disaggregation considering least square error and temporal sparsity: A multi-objective evolutionary approach. *Swarm Evol. Comput.* **2021**, *64*, 100909. [CrossRef]
15. Banerjee, S.; Ghosh, A.; Rana, N. An Improved Interleaved Boost Converter With PSO-Based Optimal Type-III Controller. *IEEE J. Emerg. Sel. Top. Power Electron.* **2016**, *5*, 323–337. [CrossRef]
16. Laoprom, I.; Tunyasrirut, S. Design of PI Controller for Voltage Controller of Four-Phase Interleaved Boost Converter Using Particle Swarm Optimization. *J. Control Sci. Eng.* **2020**, *2020*, 9515160. [CrossRef]
17. Suganya, R.; Rajkumar, M.V.; Pushparani, P. Simulation and Analysis of Boost Converter with MPPT for PV System using Chaos PSO Algorithm. *Int. J. Emerg. Technol. Eng. Res.* **2017**, *5*, 97–105.
18. Chew, C.-K.; Kondapalli, S.R.R. Modeling, Analysis, Simulation and Design Optimization (Genetic Algorithm) of dc-dc Converter for Uninterruptible Power Supply Applications. In Proceedings of the 2005 International Conference on Power Electronics and Drives Systems, Kuala Lumpur, Malaysia, 28 November–1 December 2005.
19. Badis, A.; Mansouri, M.N.; Boujmil, M.H. A genetic algorithm optimized MPPT controller for a PV system with DC-DC boost converter. In Proceedings of the 2017 International Conference on Engineering & MIS (ICEMIS), Monastir, Tunisia, 8–10 May 2017.
20. Ortatepe, Z.; Karaarslan, A. Pre-calculated duty cycle optimization method based on genetic algorithm implemented in DSP for a non-inverting buck-boost converter. *J. Power Electron.* **2019**, *20*, 34–42. [CrossRef]
21. Chang, W.; Wang, J.; Chen, Q. Multi-objective optimization of synchronous buck converter based on NSGA-U algorithm. In Proceedings of the 2016 IEEE International Conference on Mechatronics and Automation, Harbin, China, 7–10 August 2016.
22. Yang, S.; Qing, A. Design of high-power millimeter-wave TM/sub 01/-TE/sub 11/mode converters by the differential evolution algorithm. *IEEE Trans. Plasma Sci.* **2005**, *33*, 1372–1376. [CrossRef]
23. Yahia, H.; Liouane, N.; Dhifaoui, R. Weighted differential evolution based PWM optimization for single phase voltage source inverter. *Int. Rev. Electr. Eng.* **2010**, *9*, 125–130.

24. Rashid, M.I.M.; Hiendro, A.; Anwari, M. Optimal HE-PWM inverter switching patterns using differential evolution algorithm. In Proceedings of the 2012 IEEE International Conference on Power and Energy (PECon), Kota Kinabalu, Malaysia, 2–5 December 2012.
25. Raghavendran, P.; Surendar, V. Design and Implementation of Hybrid Converter for PV System for Both AC and DC Load. In Proceedings of the 2019 International Conference on Recent Advances in Energy-efficient Computing and Communication (ICRAECC), Nagercoil, India, 7–8 March 2019.
26. Prabakar, K.; Li, F. Application of genetic algorithm for the improved performance of boost converters. *IFAC Proc. Vol.* **2012**, *45*, 85–90. [CrossRef]
27. Achiammal, B.; Kayalvizhi, D. Optimal tuning of PI controller using genetic algorithm for power electronic converter. *Int. J. Eng. Res.* **2013**, *2*, 6.
28. Karaarslan, A. The Implementation of Bee Colony Optimization Algorithm to Sheppard–Taylor PFC Converter. *IEEE Trans. Ind. Electron.* **2012**, *60*, 3711–3719. [CrossRef]
29. Karaarslan, A. The implementation of bee colony optimization control method for interleaved converter. *Electr. Eng.* **2015**, *98*, 109–119. [CrossRef]
30. Sasilatha, T.; Lakshmi, D.; Rajasree, R. Design and Development of Hybrid Converter for Marine Applications. *Eur. J. Med. Nat. Sci.* **2022**, *5*, 1–8. [CrossRef]
31. Rodríguez, A.; Alejo-Reyes, A.; Cuevas, E.; Beltran-Carbajal, F.; Rosas-Caro, J.C. An Evolutionary Algorithm-Based PWM Strategy for a Hybrid Power Converter. *Mathematics* **2020**, *8*, 1247. [CrossRef]
32. Wang, C. Investigation on Interleaved Boost Converters and Applications. Ph.D. Thesis, Virginia Tech, Blacksburg, VA, USA, 2009.
33. Jang, Y.; Jovanovic, M.M. Interleaved Boost Converter With Intrinsic Voltage-Doubler Characteristic for Universal-Line PFC Front End. *IEEE Trans. Power Electron.* **2007**, *22*, 1394–1401. [CrossRef]
34. Pedro, A.-A.J.; Cesar, R.-C.J.; Francisco, B.-C.; Antonio, V.-G.; Eduardo, H.-S.; Salvador, G.-A.; Avelina, A.-R.; Martin, G.-V.P. Power quality improvement by interleaving unequal switching converters. *IEICE Electron. Express* **2016**, *13*, 20160558. [CrossRef]
35. Holland, J.H. Genetic algorithms: Computer programs that "evolve" in ways that resemble natural selection can solve complex problems even their creators do not fully understand. *Sci. Am.* **2005**, *267*, 1992.