




# THD Minimization in a Seven-Level Multilevel Inverter Using the TLBO Algorithm

Kenia Yadira Gómez Díaz \*, Susana Estefany de León Aldaco \*, Jesus Aguayo Alquicira   
and Luis Gerardo Vela Valdés

Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET), Cuernavaca 62490, Mexico; [jesus.aa@cenidet.tecnm.mx](mailto:jesus.aa@cenidet.tecnm.mx) (J.A.A.); [luis.vv@cenidet.tecnm.mx](mailto:luis.vv@cenidet.tecnm.mx) (L.G.V.V.)

\* Correspondence: [m21ce033@cenidet.tecnm.mx](mailto:m21ce033@cenidet.tecnm.mx) (K.Y.G.D.); [susana.da@cenidet.tecnm.mx](mailto:susana.da@cenidet.tecnm.mx) (S.E.d.L.A.)

**Abstract:** This paper presents the minimization of total harmonic distortion in a seven-level cascaded H-bridge multilevel inverter with resistive load using the teaching–learning-based optimization algorithm. The minimization of Total Harmonic Distortion (THD) is a challenging optimization problem due to the fact that nonlinear equations are involved. Recently, bio-inspired algorithms have become very popular approaches to solving various optimization problems in different areas of engineering. For this reason, the results obtained with the Teaching–Learning–Based Optimization (TLBO) algorithm were compared with three other popular bio-inspired algorithms, the genetic algorithm, differential evolution, and particle swarm optimization. The comparative analysis, conducted by sweeping the modulation index, made it possible to obtain graphs and data on the behavior of the four analyzed algorithms. Finally, it was concluded that the TLBO algorithm is very effective and is able to solve the THD-minimization problem. Its main advantage over the other algorithms is the fact that it does not require control parameters for its correct operation in the solution of the problem.

**Keywords:** bio-inspired computing; multilevel inverter; optimization; TLBO algorithm; power converter; power electronics



**Citation:** Gómez Díaz, K.Y.; de León Aldaco, S.E.; Aguayo Alquicira, J.; Vela Valdés, L.G. THD Minimization in a Seven-Level Multilevel Inverter Using the TLBO Algorithm. *Eng* **2023**, *4*, 1761–1786. <https://doi.org/10.3390/eng4030100>

Academic Editor: Antonio Gil Bravo

Received: 26 April 2023

Revised: 6 June 2023

Accepted: 21 June 2023

Published: 24 June 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

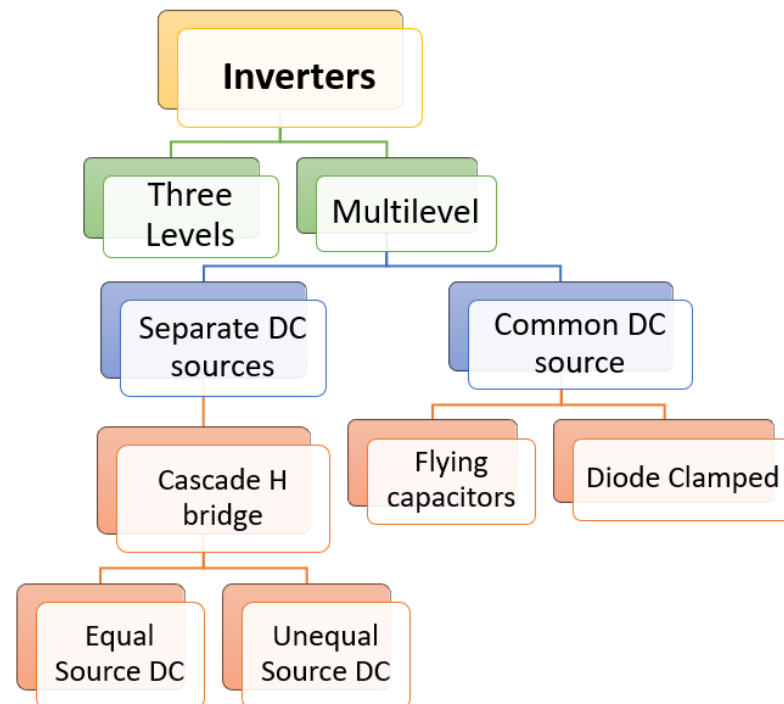
The areas of optimization span a wide range of disciplines and applications from design through business planning to industrial applications. However, some optimization problems can be challenging to solve, especially problems that are nonlinear, or problems of large magnitude. Currently, bio-inspired algorithms are becoming very popular for solving data-management and optimization problems [1].

The teaching–learning-based optimization (TLBO) algorithm was originally introduced by V. Rao in 2011 [2]. It was inspired by the philosophy of the teaching–learning process in classrooms and mimics the influence of a teacher on student outcomes. As with other swarm-intelligence algorithms, TLBO is a population-based stochastic–heuristic optimization algorithm. It has achieved great popularity due to some of its unique features, such as its concept and the fact that it does not require specific parameters, it is fast and easy to implement, and it has been widely applied to solve numerous problems in various engineering areas [3–8].

The multilevel inverter technique is based on synthesizing the forward-current voltage from several levels of DC voltage. As the number of voltage levels on the DC side increases, the output waveform adds more levels, thus producing a step wave that approximates a sine wave with minimum harmonic distortion; it was established by IEEE519 and EN50160 that the total harmonic distortion (THD) of the voltage should not be greater than 8% [9,10].

Multilevel inverters are receiving increasing attention in industry and academia, since they are among the preferred power-electronic-conversion options for high-power applications [11–13]. They have successfully entered industry and can therefore be considered a

mature and proven technology. Currently, they are marketed in standard and customized products powering a wide range of applications, such as: compressors, extruders, pumps, fans, mills, rolling mills, conveyors, crushers, blast-furnace blowers, gas-turbine starters, mixers, mine hoists, reactive power compensation, marine propulsion, high-voltage direct current (HVDC) transmission, pumped hydraulic storage, wind-energy conversion and rail traction, and the automotive industry, to name a few. A growing group of companies is commercially offering converters for these applications. Figure 1 shows the classification of the inverters, featuring only the most common topologies:



**Figure 1.** Inverter classification [14].

In this study, the cascaded H-bridge multilevel inverter was applied, since this type of inverter has different advantages compared to the other two topologies. These advantages are described below [15]:

1. Reduced volume, due to the smaller number of components.
2. The production.
3. Low distortions in the input current.
4. Operations at the two fundamental switching frequencies.
5. Very low total harmonic distortion in the output waveform, without any filtering circuitry.

However, as well as advantages, the cascaded H-bridge multilevel inverter has some disadvantages [16]:

1. DC power supplies or separate capacitors are required for each module.
2. A more complex controller is required due to the number of capacitors.

Reducing the THD of the output-voltage waveform is a vital issue in the design of useful and efficient multilevel inverters. Therefore, improving the quality of the output-voltage waveform and minimizing THD using metaheuristic optimization methods have become the subjects of many recent papers [16–19]. The performance of multilevel converters depends on their switching strategy. To improve the quality of the output-voltage waveforms of multilevel inverters, different modulation strategies have been proposed.

In the THD-minimization approach, all the harmonic components are considered for minimization, while taking into account the fundamental component. Usually, an

optimization algorithm is used to search for the best optimization algorithm in order to find the best switching angles, to achieve the lowest possible THD.

This paper focuses on providing a comparative analysis of THD minimization in a cascaded multilevel converter. Four bio-inspired algorithms, TLBO, the genetic algorithm (GA), differential evolution (DE), and particle swarm optimization (PSO) were compared.

The comparison was performed by sweeping the inverter-modulation index and comparing the obtained THD value. In addition, the execution speeds of the bio-inspired algorithms for the calculation of the optimal angles were compared. The intention of this paper is to show that one of the advantages of the TLBO algorithm is the simplicity of its implementation, since it has no initialization parameters.

## 2. Bio-Inspired Algorithms

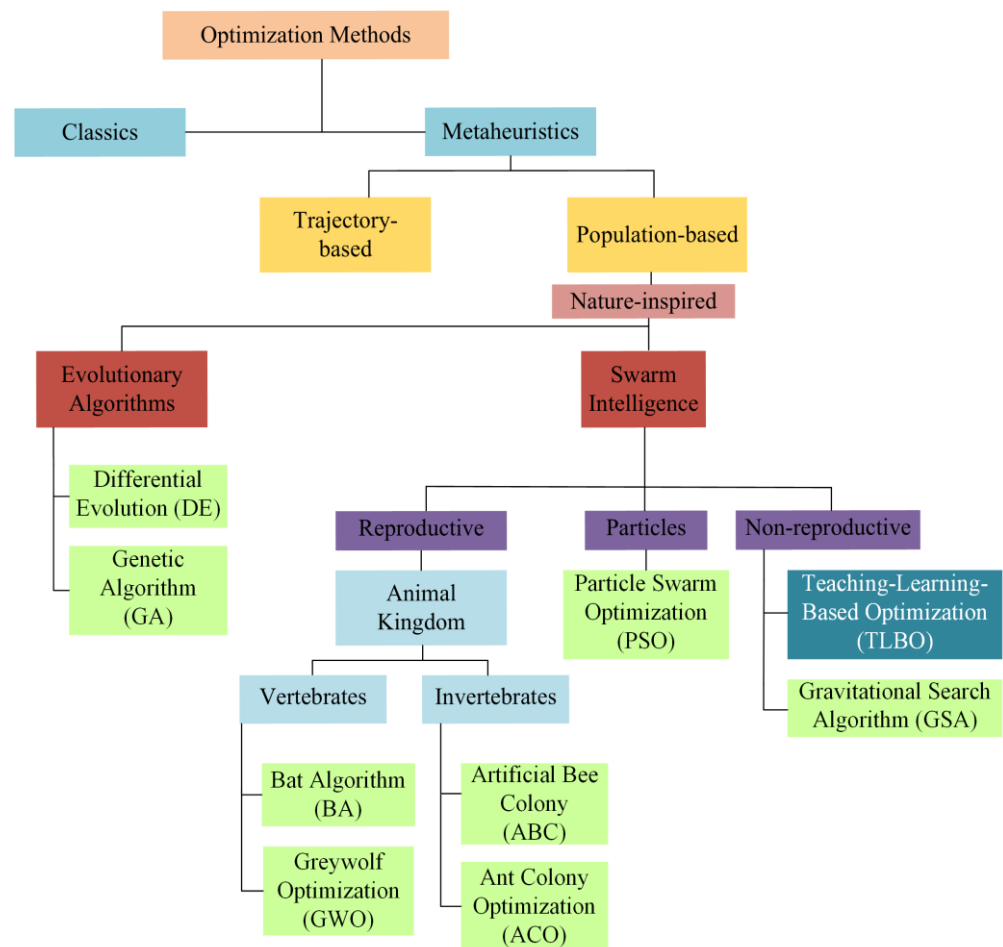
The areas of optimization cover a wide range of disciplines and applications from design through business planning to industrial applications. However, some optimization problems can be challenging to solve, especially problems that are nonlinear, or problems of large magnitude. Currently, bio-inspired algorithms are becoming very popular approaches to solving data-management and optimization problems. The differential evolution (DE) algorithm and the genetic algorithm (GA) are the best-known and most widely implemented evolution-based algorithms.

From the 1960s, the genetic algorithm, which mimics Darwin's theory of evolution, began to appear, using crossover, mutation, and selection as basic operators to perform operations. At the same time, L.J. Fogel et al. began to study artificial intelligence, which led to the development of evolutionary programming. Currently, these algorithms and many new algorithms are considered evolutionary algorithms, which are also called evolutionary computation [20].

Bio-inspired algorithms are based on using analogies with natural or social systems for problem-solving. These algorithms simulate the behavior of natural systems for the design of non-deterministic heuristic methods for search, learning, behavior, etc. Some of their main characteristics are that they are nondeterministic, they often have a parallel structure, i.e., multiple agents, and are also adaptive, since they use feedback from the environment to modify the model and parameters until an optimal solution is reached.

Since the early 1980s, research on combinatorial optimization problems has focused on the design of general strategies to guide heuristics. These have been called metaheuristics. They involve the intelligent combination of various techniques to explore the solution space. Metaheuristic procedures belong to a class of approximate methods that are designed to solve difficult combinatorial optimization problems when classical heuristics are neither effective nor efficient. Metaheuristics provide a general framework for creating new hybrid algorithms by combining different concepts derived from artificial intelligence, biological evolution, and statistical mechanics.

Although there are appreciable differences between the different methods developed so far, all of them try to combine, to a greater or lesser extent, search-intensification-selection moves that improve the valuation of the objective function and diversification, accepting other solutions that, although worse, allow the evasion of local optima. The application of metaheuristic algorithms to optimization problems has been very important during the last decades. The main advantages of these techniques are their flexibility and robustness, which allow their application to a wide range of problems. Figure 2 shows a diagram with the classification of optimization methods.



**Figure 2.** Classification of metaheuristic methods.

The metaheuristics used in combinatorial optimization can be classified into three main groups. The first generalize the sequential search by environments so that, once the process has been undertaken, a path is traversed from one solution to a neighboring solution until the process is concluded. The second group includes procedures that act on populations of solutions, evolving towards higher-quality generations. The third group is made up of artificial neural networks. This form of classification is insufficient for hybrid metaheuristics that employ, to a greater or lesser extent, strategies from both groups. This eventuality generates a desirable enrichment of possibilities that are adaptable, if necessary, to different combinatorial optimization problems.

### 2.1. The Teaching–Learning–Based Optimization Algorithm (TLBO)

The TLBO algorithm was originally introduced by V. Rao in 2011 [2]. It was inspired by the philosophy of the teaching–learning process in classrooms and mimics the influence of a teacher on student outcomes. As with other swarm-intelligence algorithms, TLBO is a population-based stochastic-heuristic optimization algorithm. It has reached great popularity due to some of its unique features unique, such as its concept, the fact that it does not require specific parameters, its fast and easy implementation, and its wide application in the solution of numerous problems in various engineering areas [3].

The TLBO algorithm is based on the impact exerted by the teacher on the students in a class. As with nature-inspired algorithms, the TLBO is also a population-based approach that uses a population of explanations to refer to the global resolution; however, the path does not have a user-specified parameter. A collection of students is considered the population. Each student is considered a person. Through the TLBO algorithm, the various topics presented to the students are viewed as schema elements. The learning outcome

of a student is similar to the fitness as it occurs in different optimization algorithms. The TLBO procedure is divided into a couple of stages. The first stage consists of the “teacher phase”, and the other consists of the “learner phase”. In the “teacher phase”, the students learn through the teacher, i.e., when he/she teaches the lesson, and the “learner phase” refers to learning through the interactions between the students, all of which are executed randomly [21]. Figure 3 shows the flowchart of the TLBO algorithm.

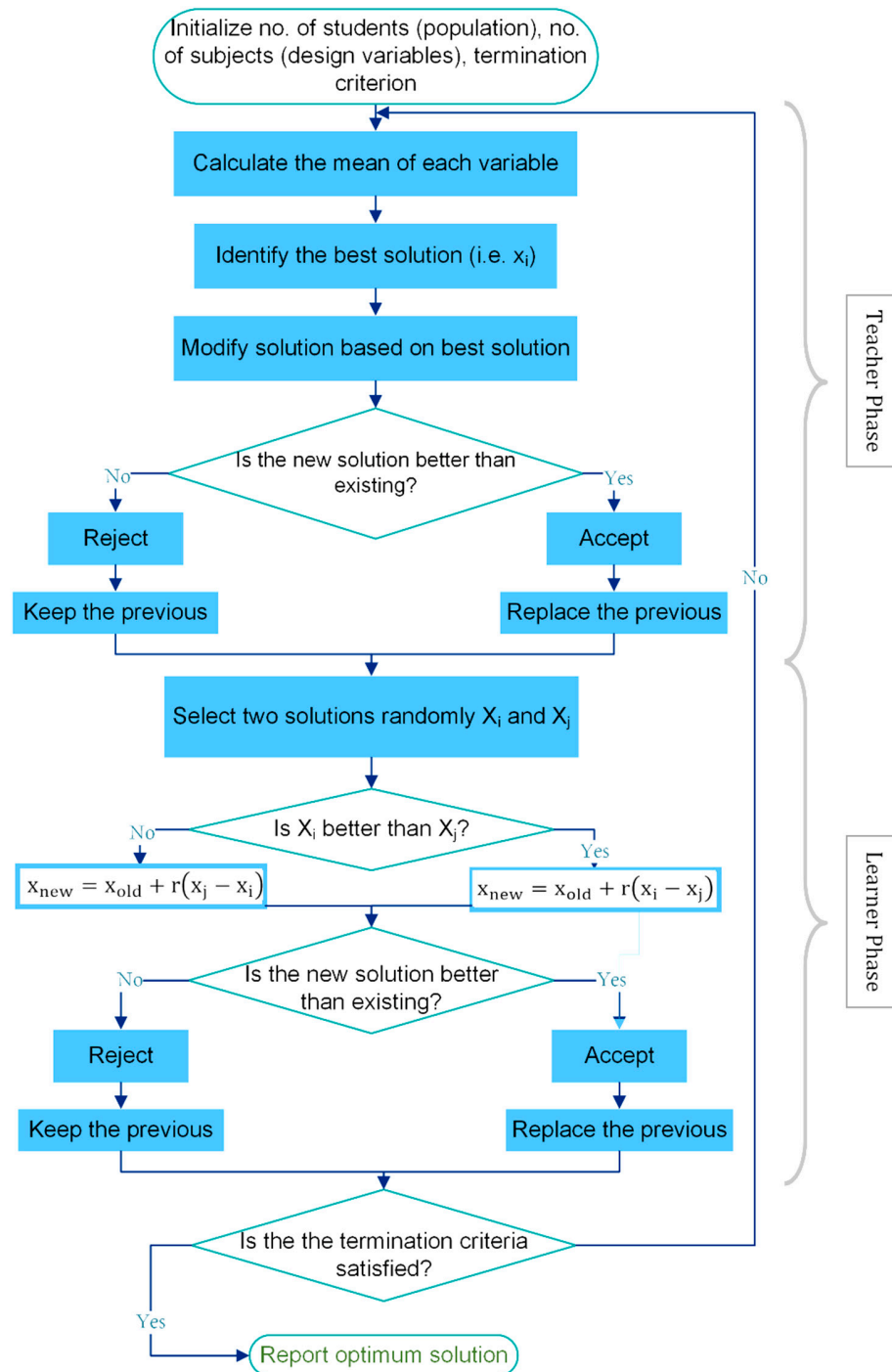


Figure 3. TLBO-algorithm flowchart.

As shown in the flowchart of the TLBO algorithm, users form a population, and the design of several topics, that is, the variables and the termination criteria, followed by the calculation of some other parameters, such as the average value of each variable,

establishes the best solution. At this point, the iteration begins, and another, better solution is calculated; if it meets the parameters and the first requirement, it is set as the new best solution, or it begins to perform another calculation until a better solution is found. Subsequently, it compares two solutions chosen randomly, the best of which is selected and compared with the solution that was previously established as the best to whether it should replace the solution that was previously chosen as a teacher until the completion criteria are met. The best solution obtained based on the iterations made is then reported.

The following is a brief description of each of the stages involved in the TLBO algorithm:

1. Initialization: the user provides a population (number of students), the decision variables or design variables (number of topics), and the termination criterion, which is the maximum number of iterations.
2. Teacher phase: this is the first part of the algorithm in which the students learn through the teacher. During this phase, the teacher tries to increase the average class score in the subject he/she teaches according to his/her ability. In any iteration  $i$ , assume there is a number of subjects “ $m$ ” (i.e., design variables), “ $n$ ” is the number of students (i.e., population size,  $k = 1, 2, \dots, n$ ), and  $M_{j,i}$  is the average result of the students in a particular subject “ $j$ ” ( $j = 1, 2, \dots, m$ ). The best overall result  $X_{total-kbest,i}$ , considering all the subjects together from the whole population of students, can be considered as the result of the best student  $k_{best}$ . However, as the teacher is usually considered to be a highly educated person who trains the students so that they can obtain better results, the algorithm considers that the best student identified is the teacher.
3. Learner phase: this is the second part of the algorithm, in which learners increase their knowledge by interacting with each other. A learner interacts randomly with other learners to improve his or her knowledge. A learner learns new concepts if the other learner has more knowledge than him/her. The learner that provides the best result to the function is chosen, and the process ends when the termination criterion that was set in the initialization stage is met.

All the observations above allow us to state that the TLBO algorithm is developed in the following steps:

- (1) The formulation of the objective function or fitness function.
- (2) The initialization of the optimization parameters and the limits of the variables. The population size, number of generations, number of design variables, and limits of the design variables are initialized. This stage establishes the objective function(s) to be maximized or minimized, while satisfying all equality and inequality constraints.
- (3) The generation of a random population. The population is expressed as:

$$\text{Population} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1D} \\ x_{21} & x_{22} & \dots & x_{2D} \\ \dots & \dots & \dots & \dots \\ x_{P_n,1} & x_{P_n,2} & \dots & x_{P_n,D} \end{bmatrix} \tag{1}$$

- (4) Teacher phase: the mean for the particular variable can be calculated as the next equation,

$$M_{*,D} = [m_1, m_2, \dots, m_D] \tag{2}$$

- (5) The best solution is considered as a teacher for that iteration:

$$X_{\text{teacher}} = X_{(f(X)=\min)} \tag{3}$$

- (6) The grade point of each variable of each student is sorted and a new mean is calculated. The difference between two means can be calculated with the following equation, where  $T_F$  may be considered as 1 or 2 and  $r$  is a random number between 0 and 1.

$$\text{Difference}_{*,D} = r(M_{\text{new},D} - T_F \times M_{*,D}) \quad (4)$$

- (7) The values are updated by adding the difference to the old solution.

$$X_{\text{new},D} = X_{\text{old},D} + \text{Difference}_{*,D} \quad (5)$$

- (8) Learner's phase; in this second phase, knowledge transfer takes place between the mutual interactions between the learners. The mathematical equations are as follows:

$$\begin{aligned} X_{\text{new}} &= X_{\text{old}} + r(X_i - X_j) \\ X_{\text{new}} &= X_{\text{old}} + r(X_j - X_i) \end{aligned} \quad (6)$$

- (9) The process is finished only if the maximum generation is reached; otherwise all the processes are repeated.

Optimization techniques applied to the feature-selection process are used to find the best possible combination of feature subsets to help reduce computational complexity. Optimization methods such as GA, PSO, ACO, and DE, to name a few, are available to solve single and multi-objective optimization problems. These algorithms are productive for solving specific problems, since their application depends on the adjustment of the specific parameters of the chosen problem. It is important that the specific parameters of the algorithm are set correctly to increase its performance. Sometimes, when the specific parameters of the algorithm are not tuned correctly, the solution may fall into a local optimum, which is a major problem in optimized feature-selection processes. In addition to this challenge, it is also necessary to adjust the common control parameters that are specific to the problem at hand to find the optimal feature subset. In order to overcome the problem of tuning particular algorithm-dependent parameters, the TLBO algorithm does not require the tuning of any algorithm-specific parameters. The TLBO algorithm does not rely on particular parameter settings, but relies instead on the usual parameters needed for its application, such as the population size (i.e., the number of learners) and the design variables (i.e., the number of topics the learner needs to learn) to derive the optimal feature subset from the original feature set.

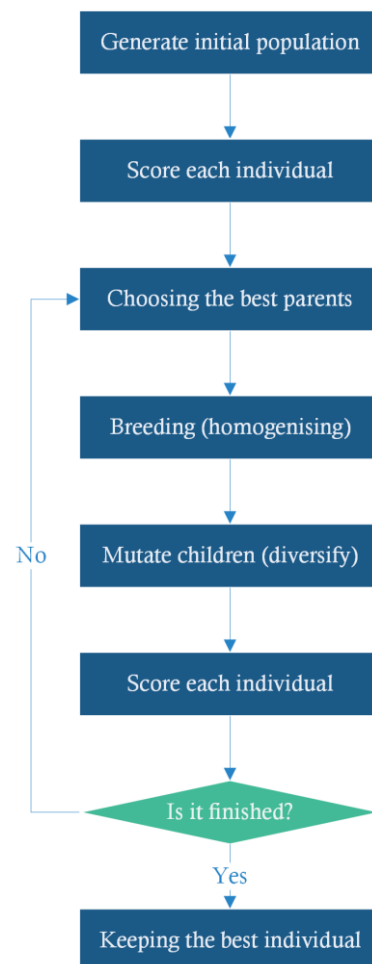
## 2.2. The Genetic Algorithm (GA)

Developed in 1975 by John Henry Holland, the genetic algorithm is an evolution-based search technique that works randomly and is based on Darwin's theory. It uses historical and current data to be analyzed in the future. This follows the rules of "survival of the fittest". This makes use of the best scheduling, through which tasks are assigned to resources according to the aptitude function for each parameter of the scheduling process. The application of the genetic algorithm to optimization consists of four main steps:

1. Population initiation;
2. Evaluation of the target function;
3. Selection;
4. The application of genetic operators.

The genetic algorithm (GA) is an optimization algorithm that is inspired by natural selection. It is a population-based search algorithm, which uses the concept of survival of the fittest [20]. The crossover (Cr) and mutation (Mr) operators play a very important role in the development of an efficient genetic algorithm. The performance of the GA depends on the configuration of its parameters. The crossover rate with a range of [0, 1] is a parameter that controls the rate at which the solutions cross. Thus, the higher the Cr, the more crossovers are performed, meaning that more diversity (in terms of solutions/chromosomes) can be

introduced into the population. Typical values of  $Cr$  are in the interval of  $[0.5, 1]$ . The mutation rate ( $Mr$ ) is a parameter with values in the range of  $[0, 1]$  that controls the rate at which individual chromosomes or their mutate. Figure 4 shows a flowchart of the genetic algorithm.



**Figure 4.** Genetic algorithm flowchart.

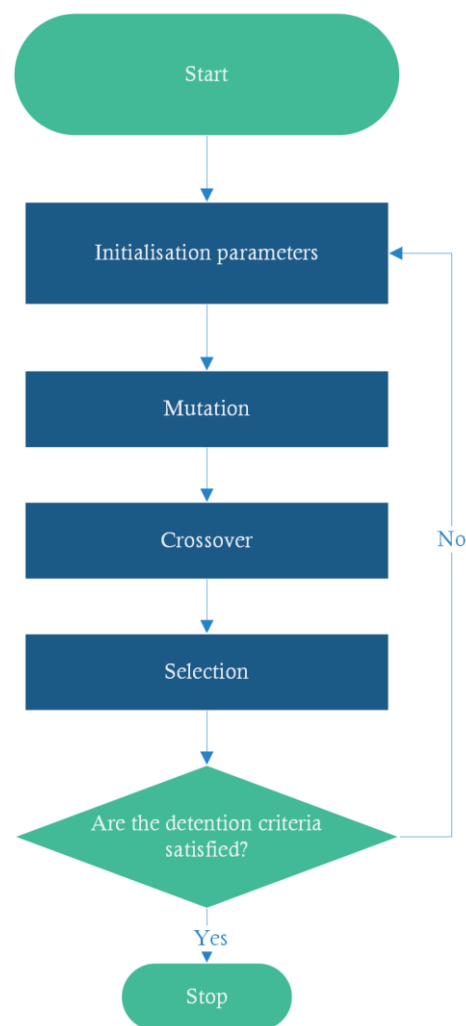
### 2.3. The Differential Evolution (DE) Algorithm

Differential evolution (DE) in the evolution section and belongs to the class of evolutionary algorithms. It was developed by Kenneth Price and published in 1994. This algorithm is a mathematical method of optimization for multidimensional functions. It is a type evolutionary algorithm. Its main procedures are initialization, mutation, crossover or recombination, and selection [22]. The DE algorithm consists of the following stages:

1. Configuration of the parameters, which involves the control parameters as well as the initialization parameters.
2. Initialization, in which the initial population is generated.
3. Mutation, in which the mutation process carried out after the first generation of the population obtained influences the control parameter, which is called the mutation factor. The mutation process consists of using a pair of vectors, which are randomly selected. This pair of vectors is subtracted in order to give a search direction (in other words, it is the orientation or direction that the algorithm takes to generate a new and different solution). In other words, for each vector in the population generated, a mutated vector is created, where  $F$  is the scaling factor and the value of  $F$  varies from 0 to 1,



4. Crossover or recombination, in which a new vector, called the test vector, is generated. Crossover is performed with the target vector, using a recombination constant, also called crossover probability, or a crossover factor,  $C_r$ , whose value is between 0 and 1. The  $C_r$  value defines how similar the test vector is to the mutant vector or the parent vector. If the  $C_r$  is close to 1, the test vector is quite similar to the mutant vector, while if it is close to 0, the test vector is similar to the parent vector.
5. Selection: a comparison is made between the target vector and the test vector according to its fitness value. That is, the parents are compared with the offspring, and the vector with the best fitness remains in the next generation, while the vector with the lowest level of fitness is eliminated. Figure 5 shows the diagram of the differential evolution algorithm.

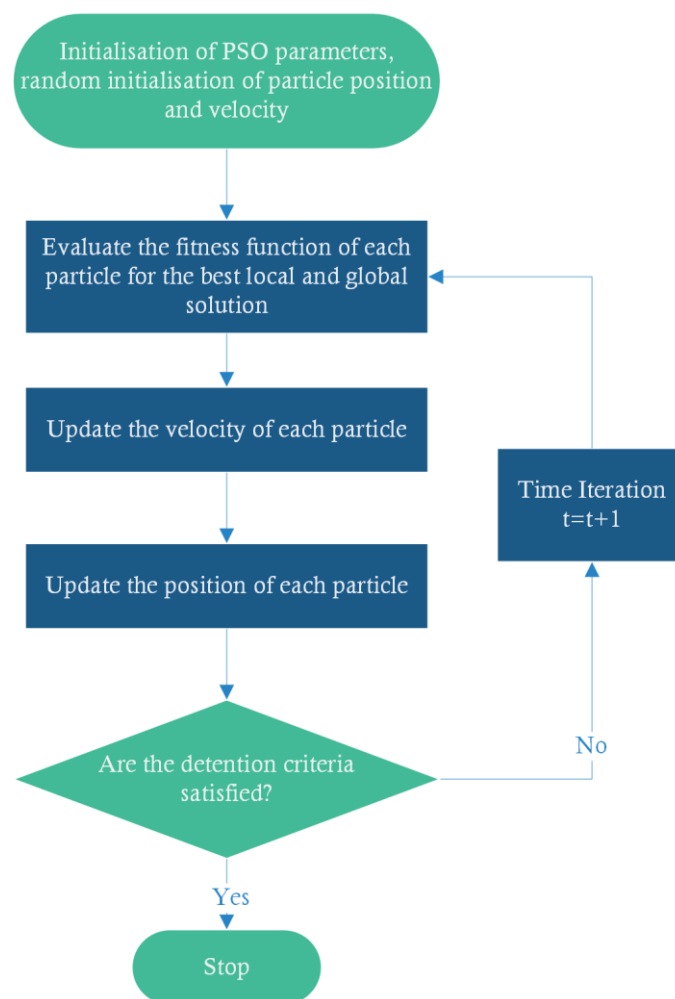


**Figure 5.** DE-algorithm flowchart.

#### 2.4. The Particle Swarm Optimization (PSO) Algorithm

Developed in 1995 by J. Kennedy and R. Eberhart, the PSO methodology is a very powerful tool for the optimization of nonlinear functions. Physically, this mimics a flock of birds communicating together as they fly. Each bird looks in a specific direction, and then, by communicating with each other, they identify the bird that is in the best location. Consequently, each bird accelerates towards the bird in the best location using a speed that depends on its current position. Each bird then investigates the search space from its new local position, and the process is repeated until the flock reaches its desired destination. It is important to note that the process involves both social interaction and intelligence, so

that birds learn from both their own experience (local search) and from the experiences of others. The convergence behavior of PSO is sensitive to the values of the inertia weight ( $W_{\max}$ ,  $W_{\min}$ ) and the acceleration coefficients ( $c_1$ ,  $c_2$ ). The parameter  $W$  is the inertia weight, and it is a positive constant, with a range of  $[0, 1]$ . This parameter is important for balancing global search, also known as exploration (when higher values are set), and local search, known as exploitation (when lower values are set). The constants  $c_1$  and  $c_2$  are also called confidence parameters;  $c_1$  expresses the degree of confidence of a particle in itself, while  $c_2$  expresses the degree of confidence of a particle in its neighbors. Low values of  $c_1$  and  $c_2$  result in smooth particle trajectories, allowing particles to explore away from good regions before being dragged back towards these regions. Higher values result in higher acceleration, with abrupt movements toward or away from good regions. Typically,  $c_1$  and  $c_2$  are static, and their optimal values are determined empirically. Figure 6 shows the basic flowchart of the PSO algorithm.



**Figure 6.** PSO-algorithm flowchart.

### 2.5. Comparison of Algorithm-Control Parameters

One of the most important properties identified by the author of the TLBO algorithm is that it has no parameters; that is, it does not require control parameters, and it only requires the initialization parameters that encompass the population, the maximum number of iterations, and the evaluation number of the function in case it is necessary to reduce to obtain a shorter time. However, these parameters are required for the use of any algorithm, as shown in Table 1, since in these parameters, the same value was set for all the chosen methods; however, it is clear that apart from these data, in the cases of the GA, DE, and

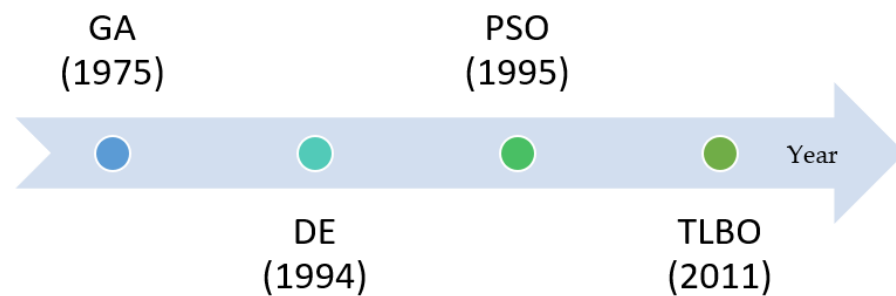
PSO, other parameters called control, are required, since these determine the behavior of each of the algorithms, which are fundamental parts in the execution of the equations that make up these methods.

**Table 1.** Comparison of the parameters of each of the algorithms.

Algorithm	TLBO	GA		DE			PSO	
Population	20	20	20	20	20	20	20	20
Iterations	50	50	50	50	50	50	50	50
Function Evaluation	50	50	50	50	50	50	50	50
Number of Control Parameters	Does not have Value	Cr	Mr	Cr	F	c <sub>1</sub> , c <sub>2</sub>	W <sub>max</sub>	W <sub>min</sub>
	-	0.8	0.2	1	0.8	2	0.9	0.4

The calculations of the angles with the TLBO, GA, DE, and PSO algorithms were performed in MATLAB (Las Vegas, NV, USA), using the initialization parameters shown in Table 1. In this case, the algorithm that requires the greatest number of control parameters is the PSO algorithm, which has four control parameters. In this case, these parameters were taken from the literature for the purpose of comparison with the algorithm analyzed only, but if new optimization problems were to arise, these parameters would need to be evaluated individually in order to determine the correct combination of values for the optimal behavior expected in each of the problems [15].

The main algorithm of interest was the TLBO, and it was decided that it would be compared with the other algorithms, which had longer implementation times in various areas, especially multilevel inverters, to obtain greater access to the literature and generate relevant statistics, in order to validate the results. Figure 7 shows a timeline to suggest the time differences in the creation of the algorithms.



**Figure 7.** Chronological order of the development of the algorithms used in this work.

### 3. Topology Selection

For this work, a multilevel inverter with separate DC sources, i.e., cascaded H-bridge, and with equal DC sources was chosen. In addition, a survey of recent research involving the use of the TLBO algorithm to minimize THD in inverters was previously conducted [18]. In that review, the results showed that single-phase topologies with resistive loads are mostly used in such studies. In three-phase topologies, the third harmonic and its multiples are naturally eliminated, considerably decreasing the THD of the output-voltage waveform. For this reason, the case study was defined as a single-phase seven-level cascaded H-bridge multilevel inverter. In Figure 8, the path taken in the choice of topology is shown with a red indicator.

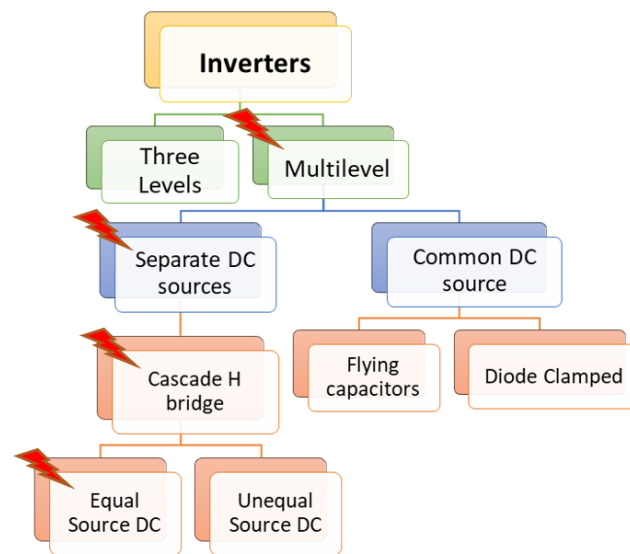


Figure 8. Selection of the topology to use as a case study.

3.1. Application of Multilevel Inverters

There are several applications of multilevel inverters, some of which are shown in Figure 9 and divided into utilities, traction, renewable energy, and uninterruptible power supply (UPS). The area of utilities is divided into four subareas: the static synchronous compensator (STATCOM), which serves to compensate for reactive power; flexible alternating-current-transmission systems (FACTS), which are effective means with which to transmit large amounts of energy; and high-voltage direct current (HVDC) transmission solutions. The traction sub-area comprises mining, electric or hybrid cars, and speed drives that can be focused on direct torque control (DTC), which allows users to directly vary the torque of the motor and field-oriented control, which in turn controls static currents, that is, the speed of the motor. The renewable-energy sub-area is divided into wind energy and photovoltaic applications, which can range from household appliances to outdoor lighting [23]. Figure 9 shows a diagram of the diverse application areas of multilevel inverters.

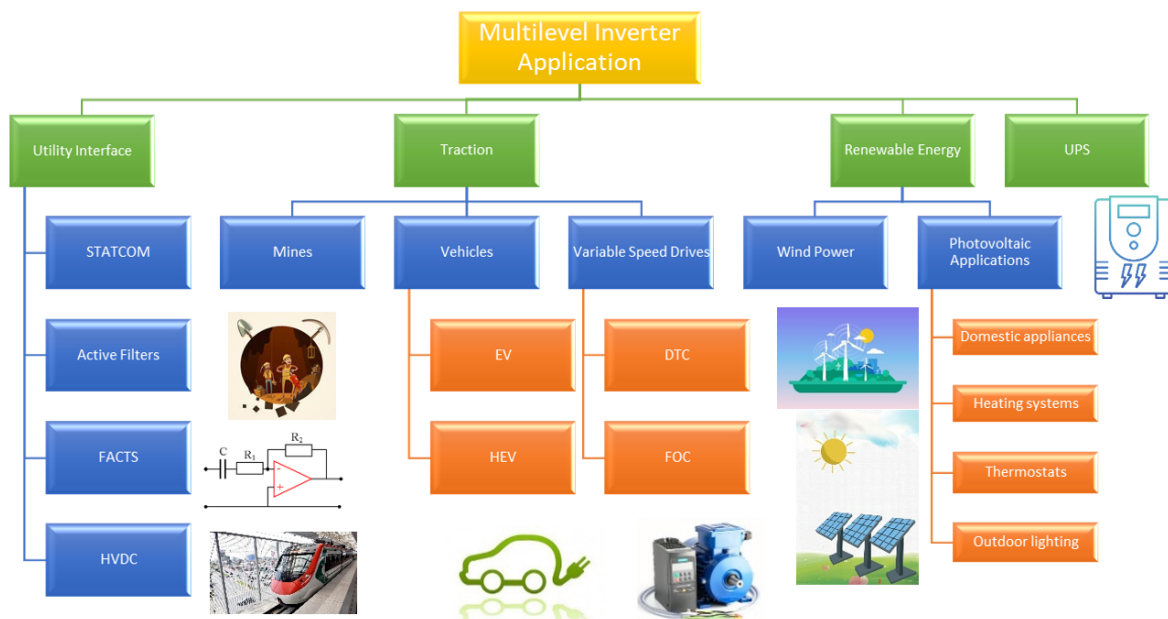
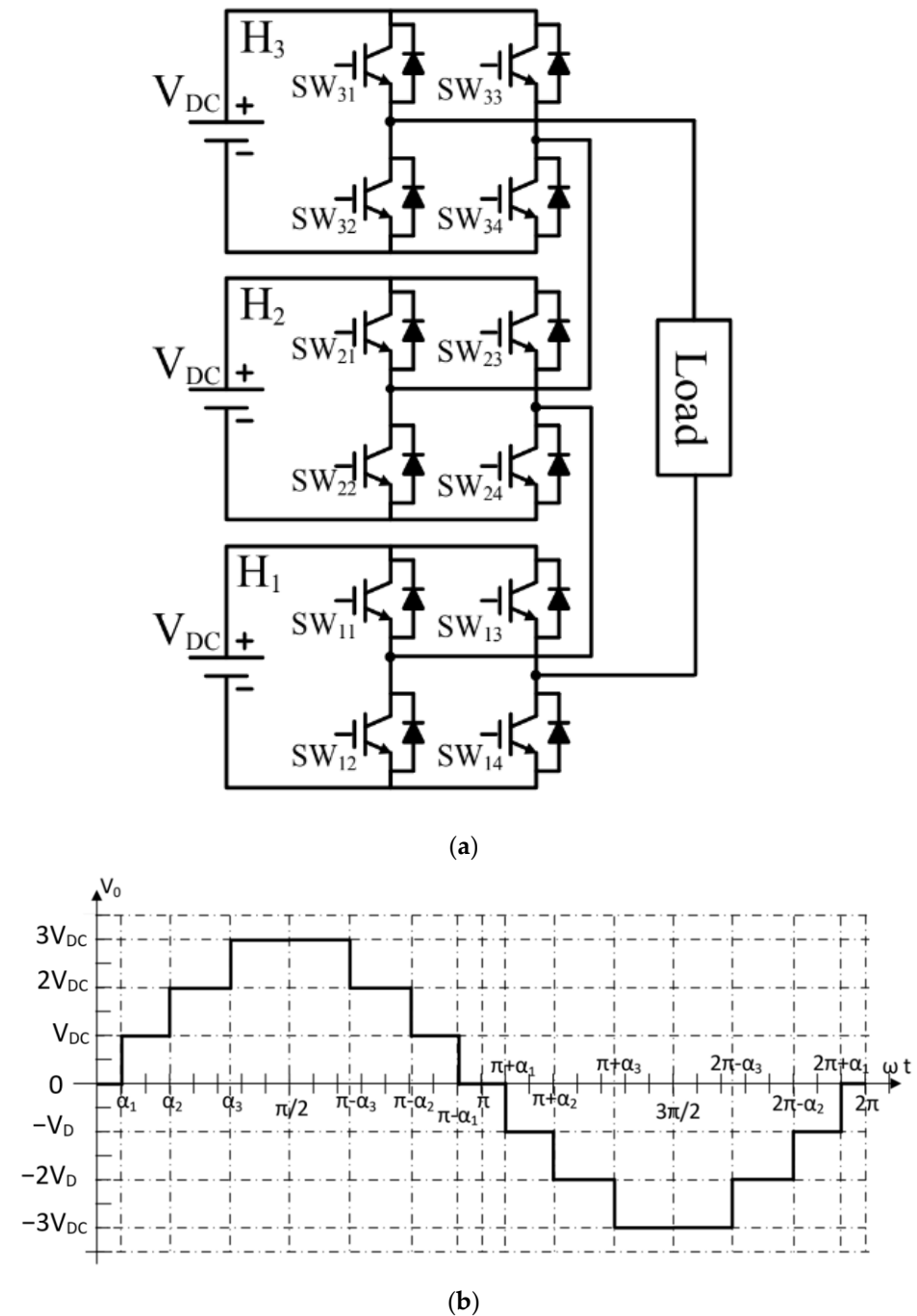


Figure 9. Diagram of the application areas of multilevel inverters.

After analyzing the range of applications for multilevel inverters, for our case study, the application was defined as UPS for outdoor lighting, so its load would be a resistive load. In addition, as previously mentioned, a review of the applications of the TLBO algorithm for THD minimization was carried out, and it was found that most of the works were performed using a resistive load [18]. This is beneficial for the analysis of the algorithm since there is no alteration in the output signal, which makes it possible to analyze the behavior of the TLBO algorithm in a natural way with the calculated angles. Figure 10 shows the diagram of the seven-level cascaded multilevel inverter and the stepped waveform of the output voltage.



**Figure 10.** Seven-level cascaded multilevel inverter. (a) Schematic diagram; (b) output voltage waveform.

### 3.2. Case Study: Objective Function

Using a Fourier analysis, the signal of the output voltage of a stepped wave can be expressed by the following equation [24]:

$$v_o = \sum_{n=1}^{\infty} \frac{4V_{DC}}{n\pi} \left[ \sum_{k=1}^s \cos(n\theta_k) \right] \sin(n\omega t) \quad (7)$$

As a signal with the symmetry of a quarter wave,  $n$  is defined as the order of the odd harmonic (1, 3, 5, 7, ...  $n$ ),  $s$  is defined as the number of stages of the multilevel inverter,  $k$  is a positive integer (1, 2, 3, ...  $s$ ), and  $\theta_k$  is the  $k$ -th switching angle, which must satisfy the following constraints:

$$\theta_1 < \theta_2 < \dots < \theta_s < \frac{\pi}{2} \quad (8)$$

When performing an expansion of Equation (1), we have the following equation:

$$b_n = \frac{4V_{DC}}{n\pi} [\cos(n\theta_1) + \cos(n\theta_2) + \dots + \cos(n\theta_s)] \quad (9)$$

According to the IEEE-519 standard [9], at least 50 harmonics must be evaluated, and the percentage of THD is calculated with Equation (4), in order to determine the quality of the output signal of an inverter.

$$\%THD = \left( \frac{\sqrt{\sum_{n=1,3,5,\dots}^{50} V_n^2}}{V_1} \right) \times 100 \quad (10)$$

where  $n$  is defined as the order of the odd harmonic (1, 3, 5, 7, ... 49),  $V_1$  is the value of the first harmonic, i.e., the fundamental voltage, and  $V_n^2$  refers to  $\left( \frac{4V_{DC}}{n\pi} [\cos(n\theta_1) + \cos(n\theta_2) + \dots + \cos(n\theta_s)] \right)^2$ , fulfilling the constraints represented in Equation (2).

Finally, the objective function is established, involving the modulation index (MI) and the percentage of THD, since the aim of the solution to the optimization problem was to find the optimal switching angles for a certain modulation index and to be able to perform a sweep from 0.44 to 1.16. The MI function term is given by the absolute value of the error required to adjust the fundamental harmonic. Therefore, the equation is as follows:

$$FO = |V_1^* - V_1| + \%THD \quad (11)$$

where  $V_1^*$  is the modulation index of the fundamental component and varies from zero to one,  $V_1$  is the modulation index that was used for the elimination of one of the harmonics, and  $|V_1^* - V_1|$  is the absolute value of the error, which is required to adjust the fundamental harmonic.

## 4. Simulation and Results

To solve the optimization problem posed (THD minimization), the algorithms were required to calculate the optimal switching angles ( $\theta_1$ ,  $\theta_2$ ,  $\theta_3$ ), satisfying the constraints posed in the previous section. The calculations of the angles with the TLBO, GA, DE, and PSO algorithms were performed in MATLAB, using the initialization parameters shown in Table 1. Once the angles were calculated, they were passed to the PSpice environment, where the angles were input as switching signals for the multilevel inverter.

The simulation software PSpice was used to evaluate the behavior and performance of the different optimization algorithms analyzed in this work. A comparison was made between the values obtained theoretically and the simulation values.

To evaluate the harmonic content of the inverter's output-voltage signal with the different optimization methods analyzed in this work, the IEEE-519 standard was used [9].

For comparison purposes, the value of the modulation index at 1.06 was used, since at this value, all the optimization methods achieved the lowest THD value.

4.1. Multilevel-Inverter Schematic

Figure 11 shows the schematic of the seven-level cascaded H-bridge multilevel inverter that was used to obtain the graphs of the output-voltage signal of the inverter, as well as the fast Fourier transformation (FFT), which is a tool that makes it possible to decompose a signal into its individual spectral components and, thus, provide information about its composition.

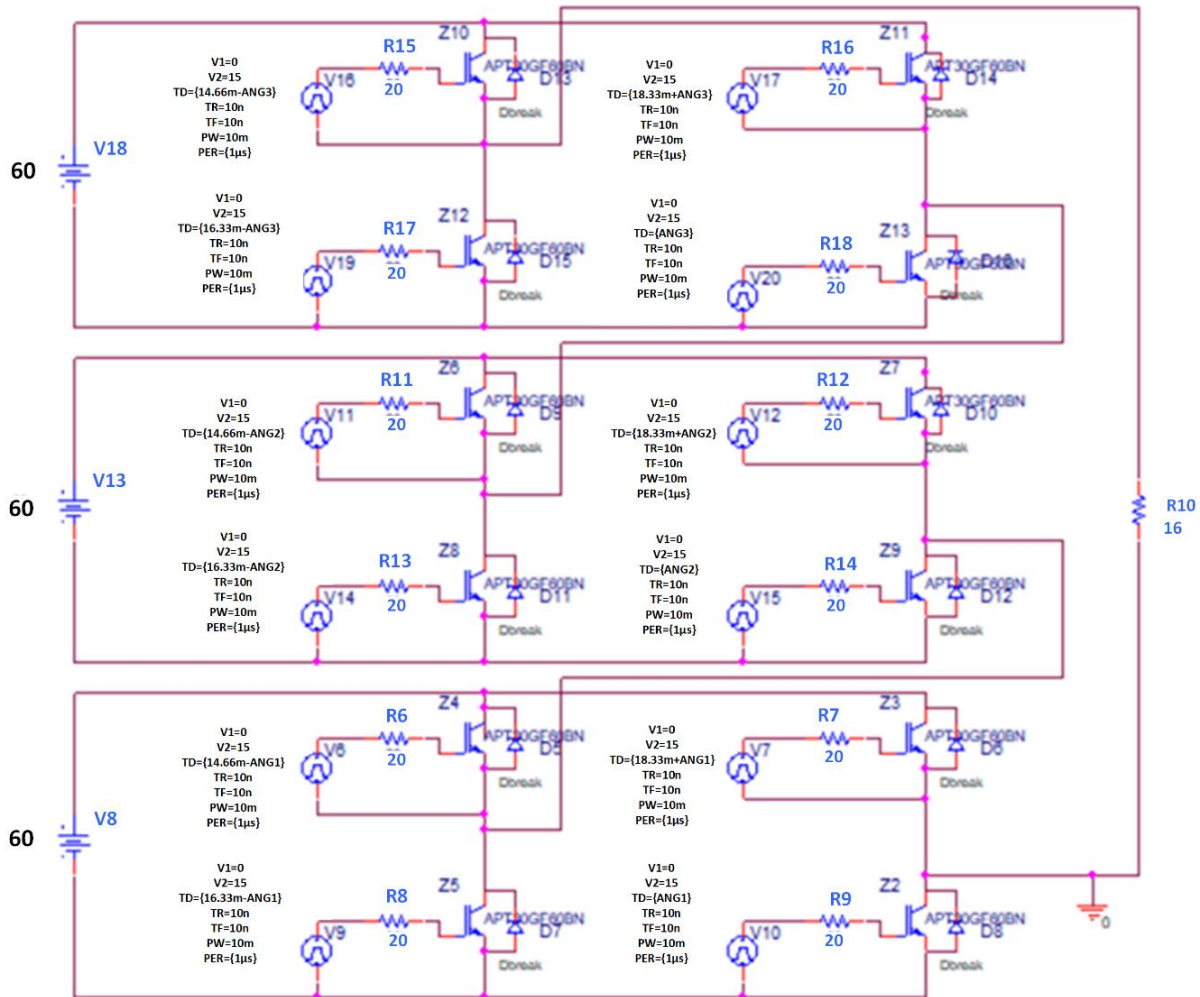


Figure 11. Schematic of the seven-level multilevel inverter H-bridge in cascade.

Subsequently, the simulation parameters were defined, as well as the IGBT to be used, which was APT30GF60BN, with the characteristics of 30 A, 600 V. In the case of the DC sources, 60 V and the base resistors of the IGBT of 20Ω and the load resistance of 16 Ω were used, with a frequency of 60 Hz. Table 2 shows the simulation parameters used to carry out this work.

Table 2. Simulation parameters.

Name	V1 (V)	V2 (V)	TD (ms)	TR (ns)	TF (ns)	PW (ms)	PER (ms)
V8, V13, V18 (DC)	60	-	-	-	-	-	-
V6 (pulse)	0	15	14.66-Ang1	10	10	10	1/fs = 16.66
V7 (pulse)	0	15	8.33-Ang1	10	10	10	16.66
V9 (pulse)	0	15	6.33-Ang1	10	10	10	16.66
V10 (pulse)	0	15	Ang1	10	10	10	16.66

#### 4.2. Output-Voltage Signal

In Figure 12, the waveform of the output voltage of the multilevel inverter is shown with the switching angles obtained by the TLBO algorithm. Figure 13 shows the evaluation of the fast Fourier transformation (FFT), to compare the harmonics that were presented with these angles and, finally, for comparison with the calculated harmonics.

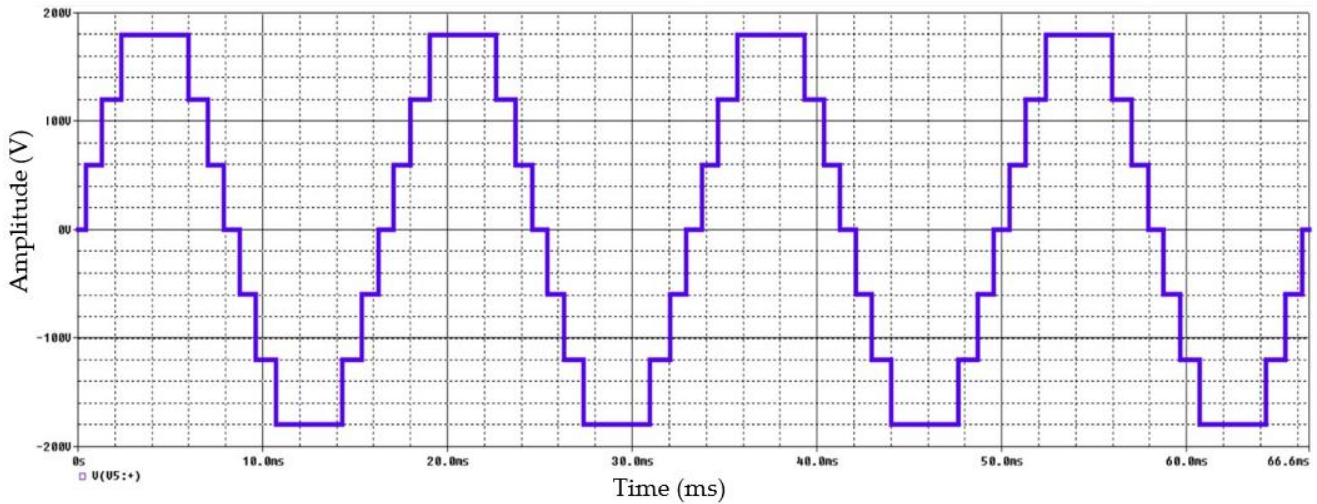


Figure 12. Signal of the output voltage, with the switching angles obtained with the TLBO algorithm.

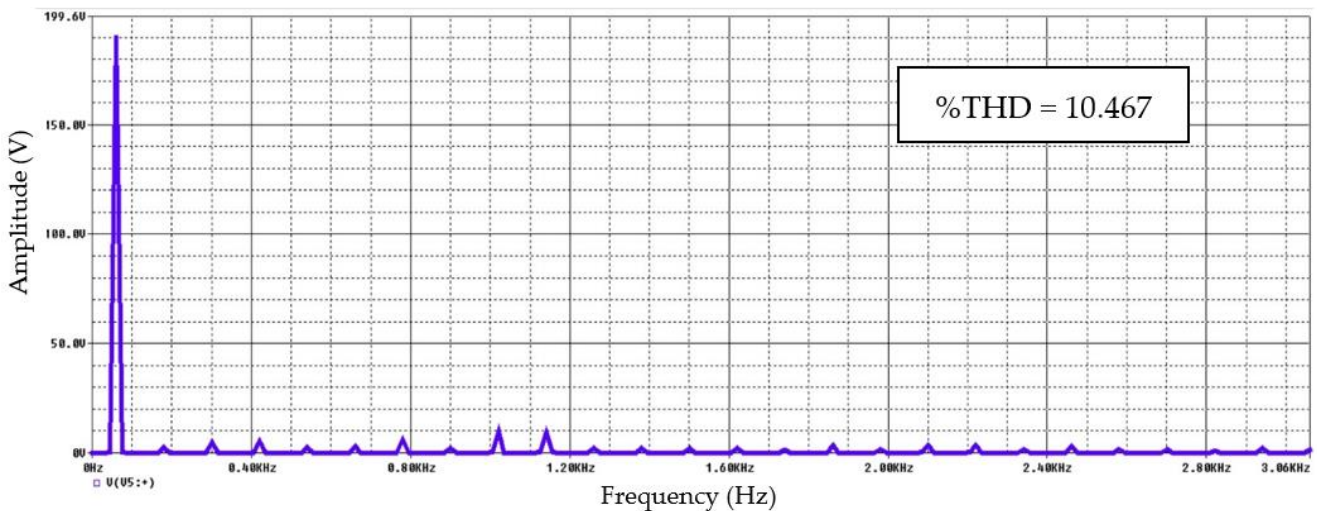


Figure 13. FFT with switching angles obtained with the TLBO algorithm.

In Figure 14, the waveform of the output voltage of the multilevel inverter is shown with the switching angles obtained by the GA algorithm. Figure 15 presents the evaluation of the fast Fourier transformation (FFT), to compare the harmonics that were presented with these angles and, finally, for comparison with the calculated harmonics.



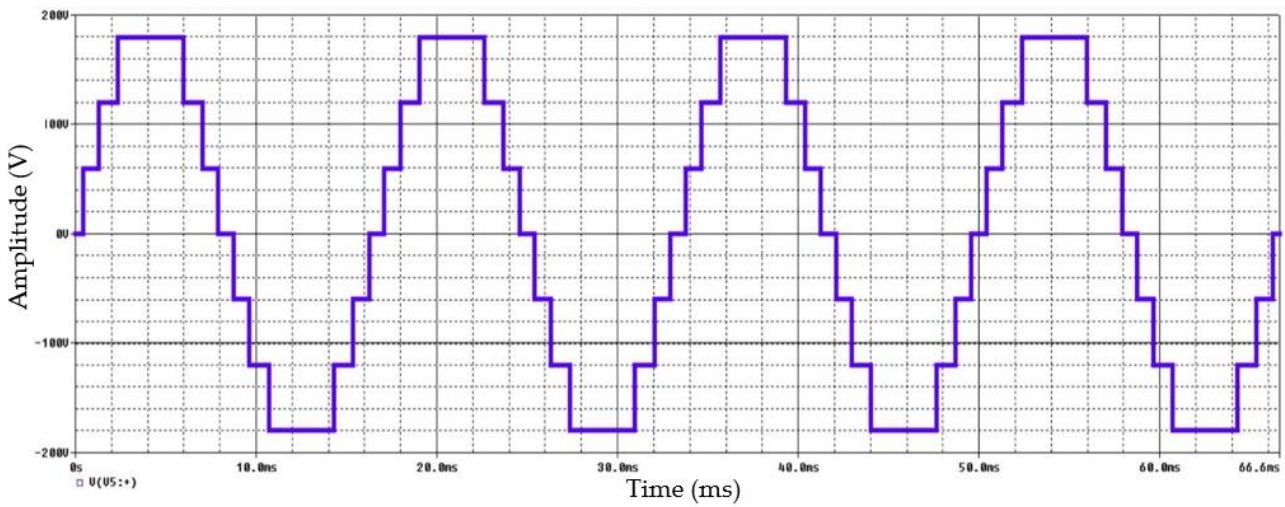


Figure 14. Output-voltage signal, with switching angles obtained with the GA algorithm.

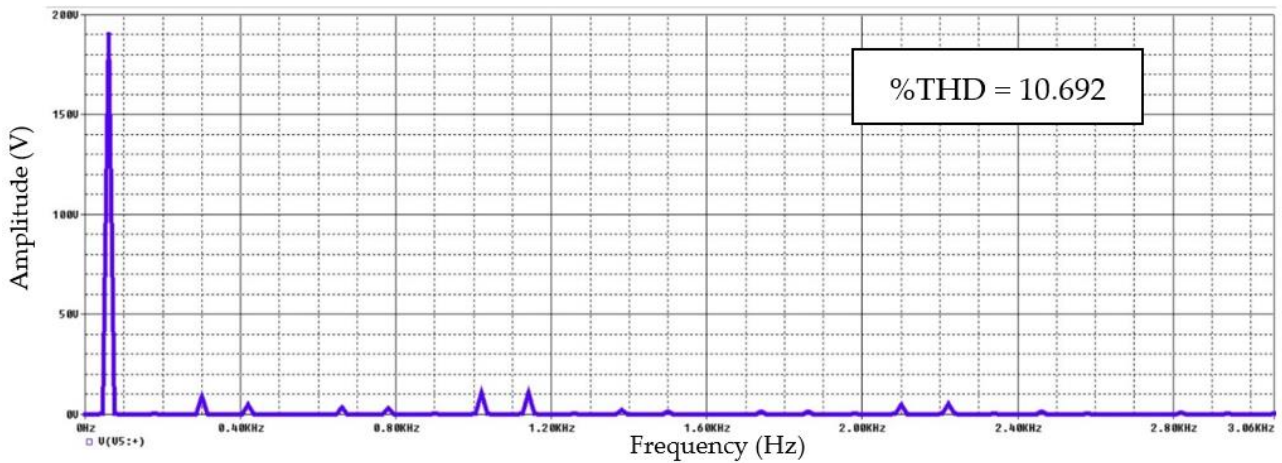


Figure 15. FFT with switching angles obtained with the GA algorithm.

In Figure 16, the waveform of the output voltage of the multilevel inverter is shown with the switching angles obtained by the DE algorithm. In Figure 17, the evaluation of the fast Fourier transformation (FFT) is presented, to compare the harmonics that were presented with these angles and, finally, for comparison with the calculated harmonics.

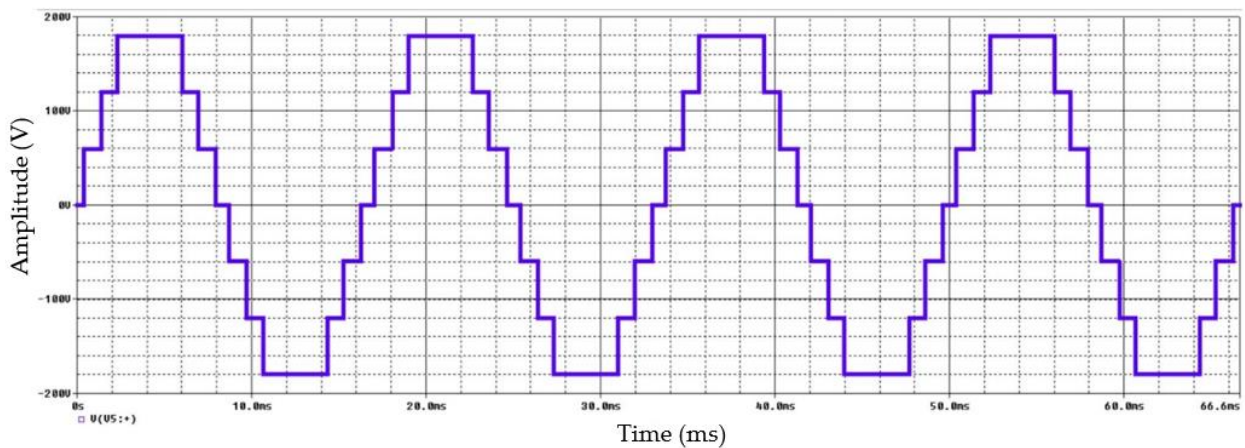


Figure 16. Output-voltage signal, with switching angles obtained with the DE algorithm.

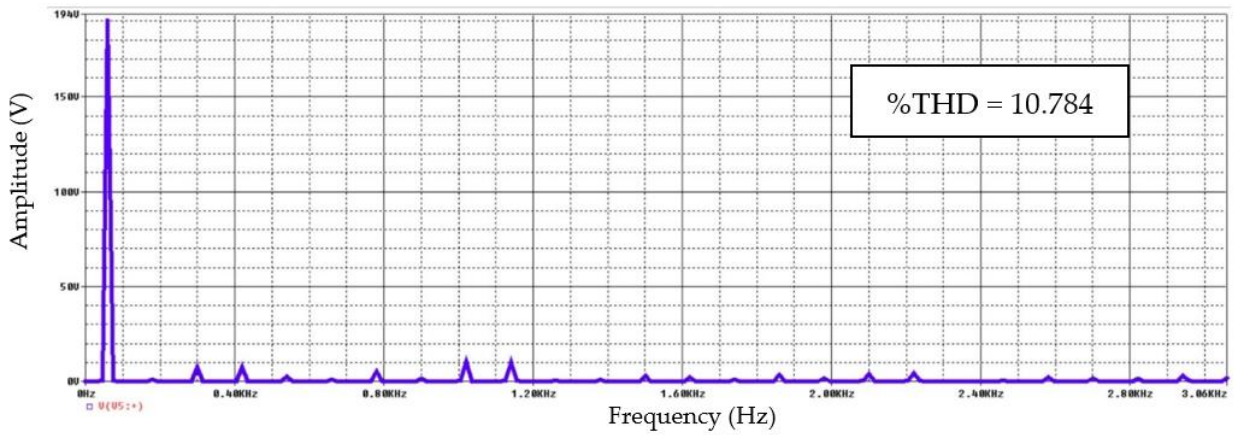


Figure 17. FFT with switching angles obtained with the DE algorithm.

In Figure 18, the waveform of the output voltage of the multilevel inverter is shown with the switching angles obtained by the PSO algorithm. Figure 19 presents the evaluation of the fast Fourier transformation (FFT), to compare the harmonics that were presented with these angles and, finally, for comparison with the calculated harmonics.

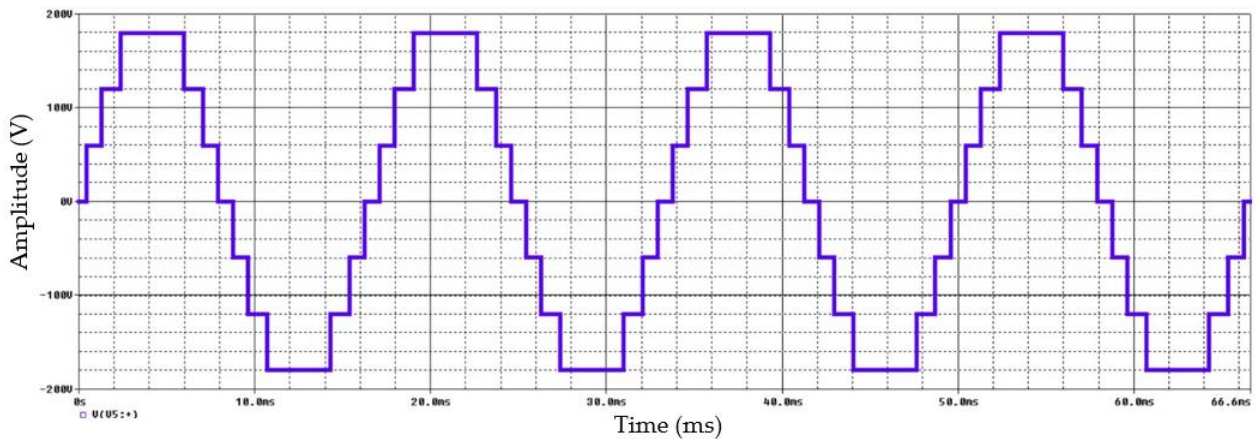


Figure 18. Output-voltage signal, with switching angles obtained with PSO algorithm.

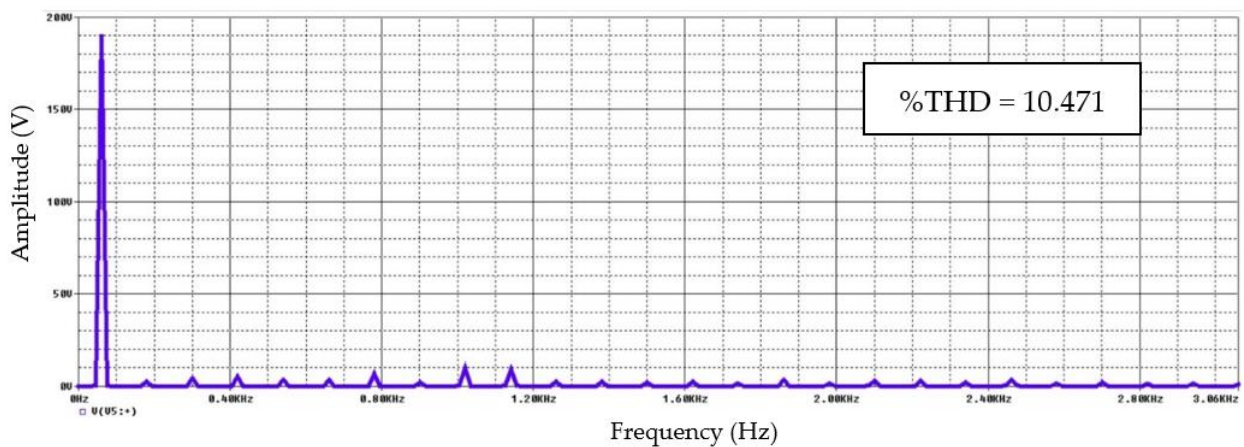


Figure 19. FFT with switching angles obtained with PSO algorithm.

Table 3 shows a comparison of the values of each of the harmonics that were obtained by the theoretical calculations of each of the algorithms.

**Table 3.** Comparison of the magnitude (whose unit is volts) of the harmonics obtained theoretically.

Harmonic	TLBO	GA	DE	PSO
1	190.8013	190.9158	190.7983	190.7996
3	2.3874	0.4359	0.8887	2.6275
5	4.9582	8.8901	7.1476	4.5787
7	5.2816	4.8059	7.0876	5.0872
9	2.7207	0.1066	2.3617	3.4034
11	3.275	3.4124	0.9684	3.6031
13	6.3406	3.0558	5.1422	6.4466
15	2.2283	0.4204	1.3488	2.3324
17	10.0067	10.771	10.3616	9.8098
19	9.3331	10.425	9.7513	9.0847
21	2.272	0.4505	0.3411	2.5134
23	2.2125	2.1179	0.857	2.7462
25	2.0546	1.1746	3.12	2.0411
27	2.4178	0.1058	2.1677	2.9217
29	1.4602	1.4149	0.9948	1.8055
31	3.6211	1.1156	3.3645	3.4277
33	1.8202	0.4042	1.6608	1.6835
35	3.5564	4.9916	4.1559	3.2802
37	3.4948	5.528	4.3454	3.0482
39	1.9135	0.4642	0.2198	2.0492
41	3.1251	1.2797	0.5727	3.6415
43	1.8184	0.575	2.0608	1.9468
45	1.8712	0.1043	1.8106	2.078
47	0.9616	0.9386	1.2488	1.1848
49	2.3175	0.5934	2.8089	1.8989

Table 4 shows a comparison of the values of each of the harmonics obtained through Fourier analysis, which is one of the tools in the Spice simulator, which evaluates the number of harmonics that the user defines, as mentioned above. The value was defined as 49; however, all the even-numbered harmonics were dismissed, according to the type of waveform and the definition in the section of the target function.

**Table 4.** Comparison of the magnitude (whose unit is volts) of the harmonics obtained by simulation.

Harmonic	TLBO	GA	DE	PSO
1	190.79	190.92	190.8	190.8
3	2.3747	0.43945	0.89097	2.6248
5	4.9594	8.9004	7.1317	4.5857
7	5.277	4.8001	7.1043	5.0867
9	2.7305	0.1103	2.3745	3.398
11	3.2749	3.4175	9.4923	3.6032
13	6.3402	3.0453	5.1629	6.4422
15	2.2123	0.42332	1.3578	2.331
17	1.0019	10.764	10.356	9.812
19	9.327	10.431	9.7393	9.089
21	2.265	4.5469	0.33477	2.5108
23	2.2113	2.1278	0.84915	2.7382
25	2.0451	1.168	3.1304	2.0397
27	2.428	0.10946	2.1761	2.9182
29	1.4565	1.4193	1.0118	1.8061
31	3.6213	1.1048	3.3856	3.4265
33	1.8037	0.40638	1.6729	1.6846
35	3.5687	4.9841	4.1454	3.2842
37	3.4933	5.5332	4.3263	3.0552

**Table 4.** *Cont.*

Harmonic	TLBO	GA	DE	PSO
39	1.9141	0.46898	0.23508	2.0479
41	3.1253	1.2892	0.57375	3.6348
43	1.8053	0.56782	2.0626	1.9452
45	1.8817	0.1078	1.8114	2.0777
47	0.9564	0.94224	1.2596	1.1869
49	2.3176	0.58244	2.826	1.9002

Table 5 shows a comparison of the calculated values in terms of power, power factor, effective value, and THD. The purpose of this comparison was to assess which of the algorithms performed best by minimizing the target function.

**Table 5.** Theoretical comparison of some of the characteristics of the inverter.

Concept	TLBO	GA	DE	PSO
Active Power (p)	1190.3 W	1193.8 W	1192.1 W	1190.1 W
Apparent Power (s)	1190.5 W	1194 W	1192.3 W	1190.3 W
Power Factor	0.9998	0.9998	0.9998	0.9998
Effective value	135.6493 V <sub>RMS</sub>	135.7775 V <sub>RMS</sub>	135.6935 V <sub>RMS</sub>	135.6478 V <sub>RMS</sub>
%THD	10.466	10.6907	10.7819	10.4705

Table 6 shows the comparison of the percentage of THD obtained in a calculated and simulated manner. In fact, in the four algorithms, some of the decimals varied; however, this variation was minimal, so there was coherence between the firing angles obtained theoretically and through the simulation using the MATLAB software.

**Table 6.** Comparison of the percentage of THD by theoretical calculations and by simulation with the switching angles of each of the algorithms.

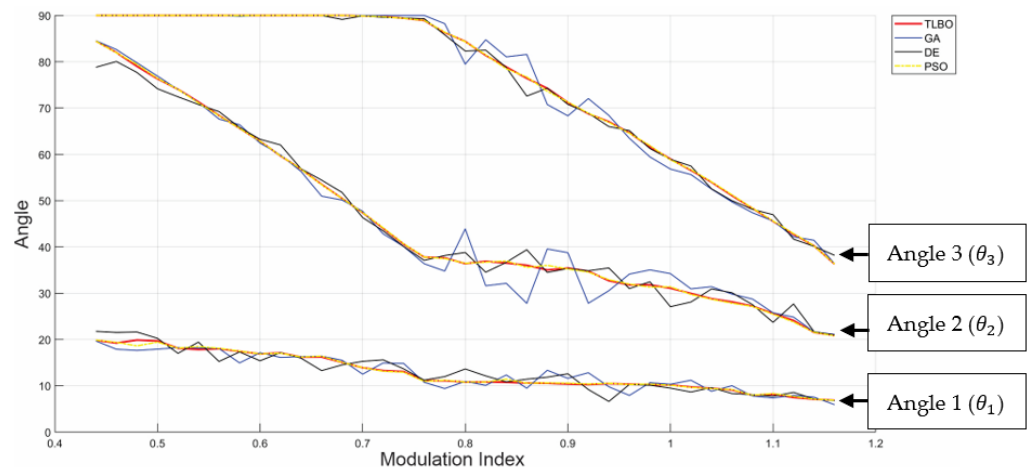
Algorithm	Calculated %THD	Simulated %THD
TLBO	10.466	10.467
GA	10.6907	10.692
DE	10.7819	10.784
PSO	10.4705	10.471

## 5. Comparison and Discussion of Results

This section presents the results obtained in a comparative way between each of the algorithms, in order to provide graphic information that allows the observation of the behavior of each of the algorithms chosen in terms of the optimization of the percentage of THD, execution times, and harmonics of the output signal.

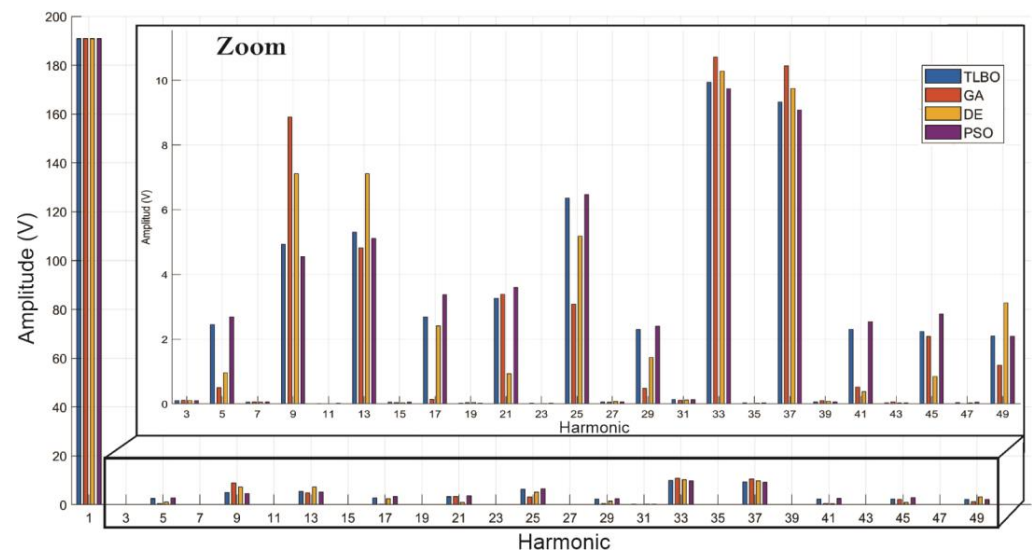
### 5.1. Behavior of THD Percentage and Harmonics

Figure 20 shows a comparison of the firing angles of the four algorithms as the modulation index varied. It can be observed that angle 3 ranged between 35 and 89.99 degrees, angle 2 ranged between 20 and 85 degrees, and angle 1 ranged between 5 and 25 degrees. The expected behavior was observed, since there was a relationship between the modulation index and the firing angles, as when the modulation index was close to 0, the angles were greater, and when the modulation index approached 1.27, the angles presented lower values. In this case, the sweep of the modulation index ranged from 0.44 to 1.16 due to the objective function that was established.



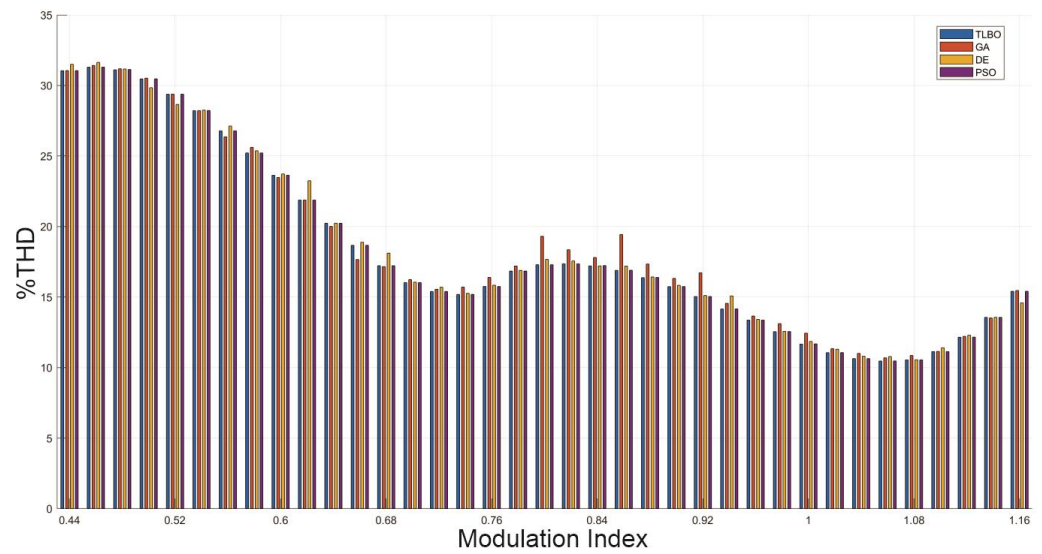
**Figure 20.** Percentage of THD with respect to the desired modulation index.

In Figure 21, the comparison of each of the harmonics is shown, with a focus from the third harmonic to the forty-ninth harmonic number; since the fundamental value was high, the other harmonics were not easy to observe. As shown in the following graph, the TLBO and PSO algorithms had lower values for many of the harmonics, so it can be concluded that these two algorithms optimized the target function better than the GA and DE algorithms. To obtain these harmonics, the MI was set as 1.06, which was the value with the lowest percentage of THD in most of the algorithms.

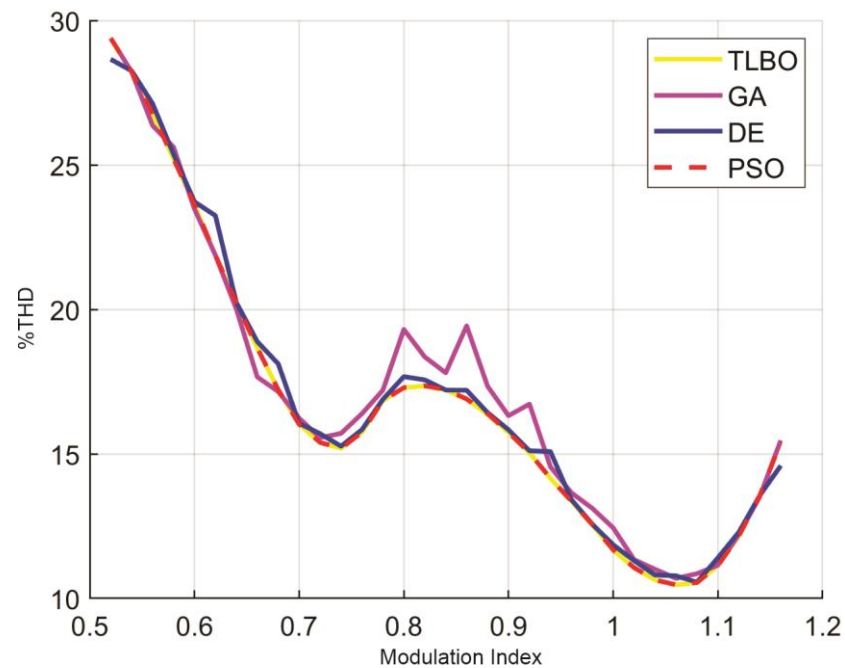


**Figure 21.** Harmonic comparison of various metaheuristic methods for a seven-level multilevel inverter.

In Figures 22 and 23, the percentage of THD with respect to the modulation index is shown. This was the core of the work, since we sought to minimize the percentage of THD when calculating with the TLBO algorithm with the firing angles of the integrated devices. A sweep of the modulation index (MI) was also performed to analyze the behavior of the THD when the MI increased or decreased. In the first figure, is the behavior not very noticeable, since the modulation index was evaluated from 0.44 to 1.16, with a step of 0.02 and, therefore, there were more than 30 evaluations. This is why the second figure, which only compares the maximum points that were obtained for each modulation index evaluated, is presented.



**Figure 22.** Histogram of the behavior of each of the algorithms in minimizing the THD with respect to the modulation index.



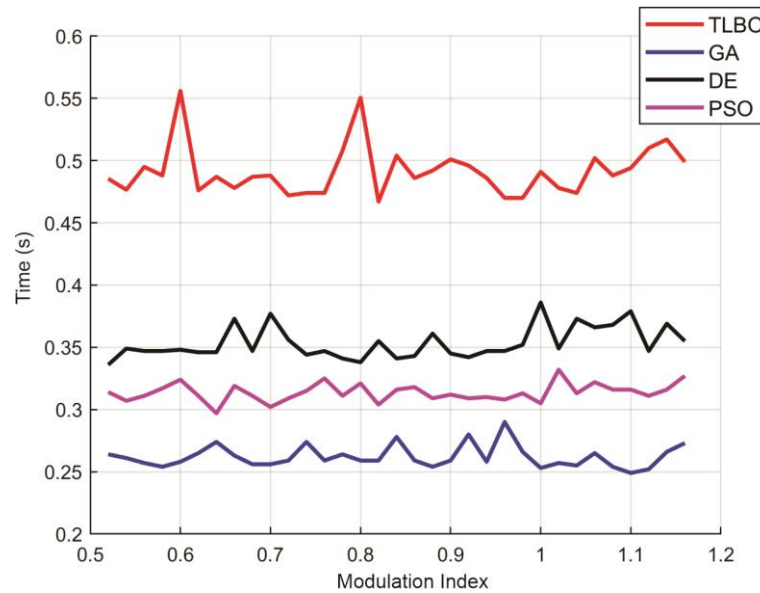
**Figure 23.** Comparative graph of the percentage of THD with respect to the modulation index for each of the chosen algorithms.

*5.2. Comparison of Algorithm Efficiency*

In Figure 23, it is possible to observe more clearly which was the least efficient algorithm in minimizing the THD. This was the GA algorithm and, later, the DE algorithm, and it can be seen that the PSO and TLBO algorithms had very similar minimization behavior, with a difference of hundredths; this is why practically the same line can be observed. This allows us to define the TLBO algorithm as a novel algorithm capable of being used in this type of optimization problem, since it has the same behavior as algorithms on which more research has been conducted and that have been applied for a long time in various fields, such as the PSO algorithm.

Another relevant factor to address is the issue of computation time, since one of the main challenges of multilevel inverters is the calculation of the online switching angles for when the inverter is interconnected to the grid. During the comparative analysis performed

in this work, the computation times recorded were less than 1 s. The values ranged from 0.2 s to 0.56 s. Figure 24 shows the execution time of each optimization algorithm when varying the modulation index.



**Figure 24.** Comparison of the execution time of each optimization algorithm when varying the modulation index.

These run times were obtained using a Dell G7 machine with an Intel(R) Core(TM) i7-8750H CPU at 2.20 GHz and 2.21 GHz, with 2 × 8 GB RAM at 2667 MHz, and a Windows 11 operating system.

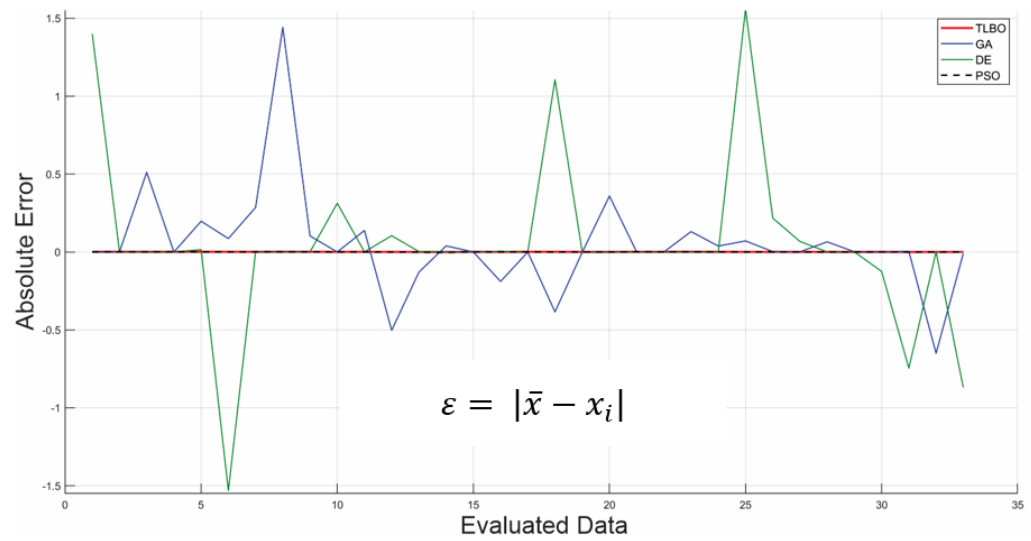
Table 7 shows the minimum and maximum values reached by the execution times of each algorithm analyzed. As can be seen in Figure 24, as well as Table 7, the algorithm with the shortest execution time was the GA.

**Table 7.** Minimum and maximum execution times of each of the algorithms.

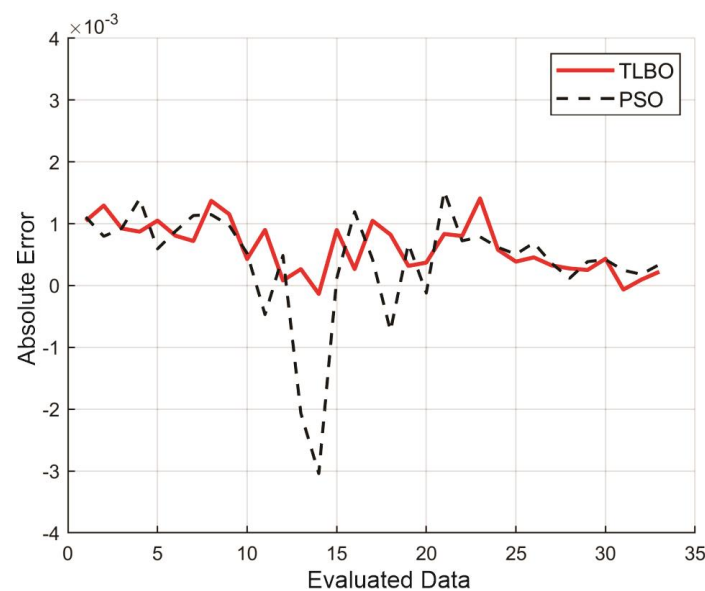
Algorithm	Minimum Time (s)	Maximum Time (s)
TLBO	0.467	0.556
DE	0.336	0.386
PSO	0.296	0.426
GA	0.249	0.290

Figure 25 shows the absolute error between the desired modulation index and that obtained by the switching angles calculated by each of the algorithms. Again, those that had more intermittent behavior, in compliance with the expected and obtained modulation indices, were the GA algorithm and the DE algorithm, while the TLBO and PSO algorithms had absolute errors that were very close to zero. The equation with which the absolute errors were obtained is also shown.

In Figure 26, a zoom-in on Figure 25 is shown to establish which of the two algorithms mentioned had the lowest absolute error, which was the TLBO algorithm.



**Figure 25.** Graph of the absolute errors of the algorithms between the desired modulation index and the obtained index.



**Figure 26.** Zoom-in on the graph with the absolute errors of the TLBO and PSO algorithms between the desired modulation index and the index obtained.

**6. Conclusions**

The main objectives, which were to analyze the TLBO algorithm and optimize the percentage of THD in a seven-level cascaded H-bridge multilevel inverter, were satisfactorily reached, and the simulation and analysis of the results were conducted successfully.

In order to compare the results, it was decided to add other algorithms to observe the behavior of the algorithm of interest, as well as to perform a sweep of the modulation index. All this was carried out, and the expected results were obtained, and they were analyzed through graphs and statistics on the behaviors of the algorithms compared to the others.

Through the findings of this study, it can be concluded that the TLBO algorithm is a metaheuristic method that allows the solution of optimization problems, It can be considered a recent algorithm, since it was created 11 years prior to the writing of this paper.

One of the most important features of this algorithm is that it has no parameters. The only information that users must enter are the population, the design variables, the



maximum number of iterations, and the target function. The analysis of the TLBO algorithm allowed us to conclude that it is an efficient optimization method, capable of solving and optimizing problems with one or several objectives (and one or several objective functions), with various constraints, or even without them. It is a practical algorithm, since it does not require any adjustments to the control parameters, which facilitates its application to various types of problem. However, it was also shown to be a slow algorithm compared to others, albeit capable implementation even in real time for the optimization of problems.

**Author Contributions:** K.Y.G.D. Data curation, Formal analysis, Investigation, Writing; S.E.d.L.A. Conceptualization, Project administration, Supervision, Writing; J.A.A. Formal analysis, Methodology, Supervision, Validation, Writing; L.G.V.V. Funding acquisition, Resources, Visualization, Software, Writing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

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

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

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

## References

1. Abdel-Basset, M.; Abdel-Fatah, L.; Sangaiah, A.K. Chapter 10—Metaheuristic Algorithms: A Comprehensive Review. In *Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications*; Sangaiah, A.K., Sheng, M., Zhang, Z., Eds.; Academic Press: Cambridge, MA, USA, 2018; pp. 185–231.
2. Rao, R.V.; Savsani, V.J.; Vakharia, D.P. Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Comput. Aided Des.* **2011**, *43*, 303–315. [[CrossRef](#)]
3. Zou, F.; Chen, D.; Xu, Q. A survey of teaching–learning-based optimization. *Neurocomputing* **2018**, *335*, 366–383. [[CrossRef](#)]
4. Wu, D.; Wang, S.; Liu, Q.; Abualigah, L.; Jia, H. An Improved Teaching-Learning-Based Optimization Algorithm with Reinforcement Learning Strategy for Solving Optimization Problems. *Comput. Intell. Neurosci.* **2022**, *2022*, 1535957. [[CrossRef](#)] [[PubMed](#)]
5. Nayak, J.; Naik, B.; Chandrasekhar, G.T.; Behera, H.S. A Survey on Teaching–Learning-Based Optimization Algorithm: Short Journey from 2011 to 2017. In *Computational Intelligence in Data Mining*; Springer: Berlin/Heidelberg, Germany, 2019; pp. 739–758.
6. Hamed, E.A. An improved teaching-learning-based optimization: Briefly survey. *Turk. J. Comput. Math. Educ.* **2021**, *12*, 2315–2319.
7. Xue, R.; Wu, Z. A Survey of Application and Classification on Teaching-Learning-Based Optimization Algorithm. *IEEE Access* **2019**, *8*, 1062–1079. [[CrossRef](#)]
8. Chen, Y.-K.; Weng, S.-X.; Liu, T.-P. Teaching–Learning Based Optimization (TLBO) with Variable Neighborhood Search to Retail Shelf-Space Allocation. *Mathematics* **2020**, *8*, 1296. [[CrossRef](#)]
9. *IEEE Std 519-2014*; IEEE Recommended Practice and Requirements for Harmonic Control in Electric Power Systems. Institute of Electrical and Electronics Engineers: Piscataway, NJ, USA, 2014; pp. 1–29.
10. *EN 50160*; Voltage Disturbances (Power Quality Application Guide). European Copper Institute, Wrocław University of Technology and Copper Development Association: Wrocław, Poland, 2004.
11. Bughneda, A.; Salem, M.; Richelli, A.; Ishak, D.; Alatai, S. Review of Multilevel Inverters for PV Energy System Applications. *Energies* **2021**, *14*, 1585. [[CrossRef](#)]
12. Gupta, K.K.; Jain, S. Comprehensive review of a recently proposed multilevel inverter. *IET Power Electron.* **2014**, *7*, 467–479. [[CrossRef](#)]
13. Alam, B.; Nusrat, M.; Sarwer, Z.; Zaid, M.; Sarwar, A. A General Review of the Recently Proposed Asymmetrical Multilevel Inverter Topologies. In *Innovations in Cyber Physical Systems*; Lecture Notes in Electrical Engineering; Singh, J., Kumar, S., Choudhury, U., Eds.; Springer: Singapore, 2021; Volume 788. [[CrossRef](#)]
14. Memon, M.A.; Mekhilef, S.; Mubin, M.; Aamir, M. Selective harmonic elimination in inverters using bio-inspired intelligent algorithms for renewable energy conversion applications: A review. *Renew. Sustain. Energy Rev.* **2018**, *82*, 2235–2253. [[CrossRef](#)]
15. Krishna, R.A.; Suresh, L.P. A brief review on multilevel inverter topologies. In Proceedings of the 2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Nagercoil, India, 18–19 March 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 1–6.
16. Sánchez-Vargas, O.; De León-Aldaco, S.E.; Aguayo-Alquicira, J.; López-Núñez, A.R. Evolutionary Metaheuristic Methods Applied to Minimize the THD in Inverters: A Systematic Review. *Eur. J. Electr. Eng.* **2021**, *23*, 237–245. [[CrossRef](#)]
17. Ürgün, S.; Yiğit, H.; Mirjalili, S. Investigation of Recent Metaheuristics Based Selective Harmonic Elimination Problem for Different Levels of Multilevel Inverters. *Electronics* **2023**, *12*, 1058. [[CrossRef](#)]

18. Gómez Díaz, K.Y.; De León Aldaco, S.E.; Aguayo Alquicira, J.; Ponce-Silva, M.; Olivares Peregrino, V.H. Teaching–Learning-Based Optimization Algorithm Applied in Electronic Engineering: A Survey. *Electronics* **2022**, *11*, 3451. [[CrossRef](#)]
19. Sánchez Vargas, O.; De León Aldaco, S.E.; Aguayo Alquicira, J.; Vela Valdés, L.G.; Mina Antonio, J.D. Differential Evolution Applied to a Multilevel Inverter—A Case Study. *Appl. Sci.* **2022**, *12*, 9910. [[CrossRef](#)]
20. Katoch, S.; Chauhan, S.S.; Kumar, V. A review on genetic algorithm: Past, present, and future. *Multimed. Tools Appl.* **2021**, *80*, 8091–8126. [[CrossRef](#)] [[PubMed](#)]
21. Rao, R.V. Teaching-learning-based optimization algorithm. In *Teaching Learning Based Optimization Algorithm*; Springer Nature: Berlin/Heidelberg, Germany, 2016; pp. 9–39.
22. Firmansyah, E.F.; Qudsi, O.A.; Habibi, M.N.; Windarko, N.A. Optimized Modified PWM based on Differential Evolution for Reducing THD on Multilevel Inverter. In Proceedings of the 2020 International Seminar on Intelligent Technology and Its Applications (ISITIA), Surabaya, Indonesia, 22–23 July 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 113–118.
23. Franquelo, L.G.; Rodriguez, J.; Leon, J.I.; Kouro, S.; Portillo, R.; Prats, M.A. The age of multilevel converters arrives. *IEEE Ind. Electron. Mag.* **2008**, *2*, 28–39. [[CrossRef](#)]
24. Hwei, P.H. *Fourier Analysis*; Addison-Wesley Iberoamericana: Boston, MA, USA, 1987; pp. 9–32, ISBN 0-201-02942-1.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.