

Stirling engine optimization using artificial neural networks algorithm

Oumaima Taki^{1*}, Kaoutar Senhaji Rhazi¹, and Youssef Mejdoub¹

¹RITM Laboratory, Higher School of Technology, CED Science de l'ingénieur, ENSEM, Hassan II University, Casablanca, Morocco.

Abstract. Neural networks are a type of machine learning algorithm that are inspired by the structure and function of the human brain. They consist of layers of interconnected "neurons" which process and transmit information. Neural networks can learn to perform a variety of tasks by being trained on large datasets, and they have been successfully applied to a wide range of problems, including image and speech recognition, natural language processing (NLP), and predictive modeling. In this paper, we are combining Stirling engines with neural networks in order to improve the performance and efficiency of the engine by using machine learning to optimize their operation. For example, an optimizing neural network such as the Multi-Layer Perceptron (MLP) could be trained to predict the most efficient operating conditions for a Stirling engine based on design parameters such as displacer stroke, phase angle and working frequency. Additionally, neural networks could be used to diagnose and predict failures in Stirling engines, potentially improving their reliability and reducing maintenance costs.

1 Introduction

A Stirling engine is a type of heat engine that operates by cyclically compressing and expanding a gas, such as air or helium, to convert thermal energy into mechanical work. It is named after Robert Stirling, who patented the concept in 1816 [1]. The key advantage of a Stirling engine is its high efficiency, as it can convert almost all of the heat supplied to it into useful work. Stirling engines have the potential to be more efficient and have a wider range of operating temperatures than traditional heat engines, such as internal combustion engines [2].

One potential application of Stirling engines is in the field of renewable energy, as they can be powered by a variety of heat sources, including solar thermal energy, biomass, and geothermal heat. They are also well-suited for use in distributed power generation systems, as they can be used to generate electricity at small scale, such as in remote communities or on off-grid properties.

* Corresponding author: oumaima.taki@ensem.ac.ma

The basic operating principle of a Stirling engine is relatively simple. A sealed container, called the "working space," is filled with a gas, such as air. The gas is then alternately heated and cooled, causing it to expand and contract [3]. This expansion and contraction drives a piston or other mechanical element, which can be used to perform work, such as turning a generator to produce electricity.

The concept of neural networks dates back to the 1940s and 1950s, when researchers began to study the structure and function of the brain in an effort to understand how it processes and stores information. In 1943, Warren McCulloch and Walter Pitts published a paper proposing a model of the neuron, which they called a "Threshold logical unit" (TLU) that could be used to represent and manipulate logical statements [4].

The first practical neural network was developed in the 1950s by Frank Rosenblatt, who invented the perceptron, a type of single-layer neural network that could be trained to classify input data into one of two categories [5]. However, the capabilities of early neural networks were limited, and they were not widely adopted at the time.

It wasn't until the late 1980s and early 1990s, with the advent of faster computers and more powerful algorithms, that neural networks began to gain widespread attention and adoption. Today, neural networks are a key component of machine learning and are used for a wide range of applications. One of the most frequently used type of neural networks is the multilayer perceptron, which consists of an input layer, one or more hidden layers, and an output layer [6]. The input layer receives the input data and passes it through the network to the output layer, which produces the final output. The hidden layers are intermediate layers that process the input data and pass it on to the output layer. The hidden layers can be used to extract features or patterns from the input data, which can improve the accuracy and performance of the model.

MLPs are trained using a variant of the backpropagation algorithm, which involves adjusting the weights of the connections between the neurons in order to minimize the error between the predicted output and the true output, especially when handling nonlinear functions of high complexity [7]. The weights are adjusted using gradient descent, which involves calculating the gradient of the error function with respect to the weights and updating the weights in the opposite direction of the gradient.

2 Stirling engine operation

The theoretical Stirling cycle resembles, to a large extent, the Carnot cycle, with the exception that the isochoric process replaces the adiabatic cooling and heating processes of the ideal Carnot cycle.

The Stirling cycle consists of four consecutive evolutions of an ideal gas, usually air or helium, between two heat sources having constant cold and hot temperatures T_c and T_e , which are divided by a perfect exchanger that has an isochoric process. We obtain the same efficiency as the Carnot's cycle when we apply the first principle of thermodynamics [8].

According to [10], the efficiency and output power can be formulated as :

$$\eta_{th} = 1 - \frac{T_C}{T_H} \tag{1}$$

$$P = \frac{Work}{Time} = \frac{Q_H - Q_L}{t_{cycle}} \tag{17}$$

Where

- η_{th} : Theoretical efficiency.
- T_C : Temperature at which the engine expels waste heat (in Kelvin)
- T_H : Temperature at which the engine takes in heat from the source (in Kelvin)
- Q_H : Heat absorbed by the engine from the high-temperature reservoir
- Q_L : Heat expelled by the engine to the low-temperature reservoir
- t_{cycle} : Time taken for one complete cycle of the engine

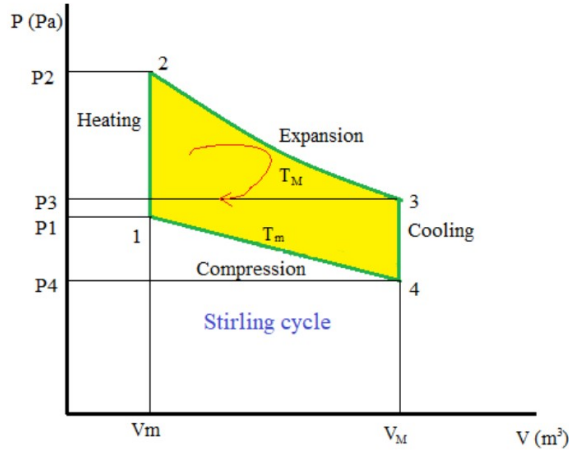


Fig. 1. Stirling engine cycle [9].

Our principal work will consist of optimizing the efficiency and the output power. Multiple design parameters of the engine have impact on those two objective functions, which makes it difficult to consider all possible combinations of values of the design parameters to eventually select an optimal set of variables that optimizes both of the objective functions η_{th} and P . We are selecting three design parameters displacer stroke d , phase angle ϕ and engine working frequency f .

3 Selection of design parameters

Any design parameter of the Stirling engine can have an impact on both its efficiency and its output power, but as we selected only three design parameters, we will describe how each of these parameters impact the objective functions that we desire to optimize.

The displacer in a Stirling engine is a component that moves back and forth inside the engine's cylinder, transferring heat from the hot end to the cold end of the engine. The movement of the displacer creates a pressure differential within the cylinder, which drives the pistons and generates power.

The efficiency of a Stirling engine is affected by the design and operation of the displacer. If the displacer is poorly designed or not properly maintained, it can reduce the efficiency of the engine. Some factors that can impact the efficiency of the displacer include the stroke length d , the stroke frequency, heat leakage and wear of the stroke [11].

However, in our paper we will consider that our engine is newly manufactured, and we will only consider the stroke length.

The phase angle refers to the timing of the movement of the displacer and the pistons. The phase angle can affect the efficiency of the engine by determining how much heat is transferred between the hot and cold ends of the engine and how much work is done by the pistons.

A phase angle ϕ that is too large can result in less heat being transferred between the hot and cold ends of the engine, which can reduce the efficiency of the engine. On the other hand, a phase angle that is too small can result in the pistons doing less work, which can also reduce the efficiency of the engine [12].

Optimizing the phase angle of a Stirling engine can be challenging, as it depends on a variety of factors such as the size and shape of the engine, the materials used, and the operating conditions. In general, a phase angle that is carefully chosen and maintained can help to maximize the efficiency of the engine.

The working frequency (f) of the Stirling engine refers to the number of cycles per unit of time that the engine goes through as it converts thermal energy into mechanical work. In general, increasing the working frequency of a Stirling engine can improve its efficiency, as it allows the engine to complete more cycles and convert more thermal energy into work in a given period of time [13]. The working frequency of a Stirling engine can also have an impact on the power output of the engine. In general, increasing the working frequency of a Stirling engine can result in an increase in the power output of the engine, as it allows the engine to complete more cycles and convert more thermal energy into mechanical work in a given period of time. However, there are limits to how much the working frequency can be increased, as the engine can only operate at frequencies that are within its design limits and that do not cause undue wear and tear on its components.

It is important to note that the efficiency and the power output of a Stirling engine are also affected by other factors, such as the temperature difference between the hot and cold ends of the engine, the design of the engine, and the materials used to construct it. Additionally, the power output of a Stirling engine can be affected by the load placed on the engine, as the power output of the engine will decrease as the load increases.

The type of the Stirling engine we will take as a default configuration in this paper is the gamma-type engine ST500 (Fig. 2).



Fig. 2. ST500 : Gamma-type Stirling engine (Designed and manufactured by IP-CO)

The ST500 Stirling engine is a small, portable engine designed for use in a variety of applications. Some key characteristics of the ST500 Stirling engine include :

- External combustion: The ST500 Stirling engine is an external combustion engine, which means that it generates power by heating a working fluid (usually air or hydrogen) using an external heat source.
- High efficiency: Stirling engines are known for their high efficiency, and the ST500 is no exception. It is capable of converting up to 70% of the heat energy it consumes into mechanical work, making it an efficient choice for a wide range of applications.
- Quiet operation: The ST500 Stirling engine operates quietly, making it well-suited for use in environments where noise is a concern.
- Flexibility: The ST500 Stirling engine is versatile and can be used in a variety of applications, including power generation, cooling, and heating.
- Compact size: The ST500 Stirling engine is compact, making it easy to transport and install in a variety of settings.
- Low emissions: The ST500 Stirling engine produces low levels of emissions, making it an environmentally friendly option for power generation.
- Low maintenance: The ST500 Stirling engine requires minimal maintenance, making it a reliable and cost-effective choice for long-term use.

In table 1, we can find the design parameters values of the ST500 Stirling engines, that our work will be based on.

Table 1. ST500 characteristics.

Specification	Values
Type	Gamma
Electrical output power	461 W
Total efficiency	8.2 %
Standard charge pressure	9 bar
Working fluid	He, CO2
Working frequency	19.1 Hz
Fuel	Natural gas
Cooling substance	Water
Power piston stroke	75 mm
Power piston diameter	84 mm
Displacer stroke	75 mm
Displacer diameter	97 mm
Clearence volume	804 000 mm ³
Phase angle	90°
Heater type	Tublar (6mm. dia) *20
Regenerator material	Steel matrix (0.96 porosity)
Heat absorbtion temperature	350-420 °C
Heat rejection temperature	30-50 °C

Based on multiple simulations performed on MATLAB® Simulink, we chose a practical range of our three design variables as such :

$$0.018m < d < 0.16m \quad (3)$$

$$60^\circ < \phi < 120^\circ \quad (4)$$

$$10\text{ Hz} < f < 20\text{ Hz} \quad (5)$$

4 Selection of design parameters

4.1 Definition of a perceptron

A perceptron is a simple type of artificial neural network that can be used for binary classification. It is considered as the most basic form of neural network architecture as it has no hidden layer in between the input and output layer [14], it is composed of a single layer of neurons, each with a binary output (0 or 1). The inputs to the perceptron are multiplied by a set of weights, which are then passed through a activation function to produce the output. This output is calculated by the equation (6) :

$$y = g\left(\sum w_i * x_i - \theta\right) \quad (6)$$

The components of a perceptron can be defined and detailed as follows [14] :

- Input Signals x_i : the input elements given to the perceptron for computation. These input values are usually provided to the perceptron for an optimization reason, in our case they represent the design variables that we attempt to optimize.
- Weights w_i : these are arbitrary values to be used as a scalar multiplier to the respective inputs in order to quantify the relevance of each input on the output. At each step of our algorithm, we try to modify the weights to see the change on the output, which in our case is the efficiency or the power output. The weights may vary at each step to have better output results.
- Threshold θ : at each step, we compare the the sum of the weighted inputs with this threshold to accept or not the output.
- Activation Function g : is a mathematical way to decided if the output is relatively positive or not. The activation function is usually either a step function, linear function, sigmoid function, tanh function or a rectified linear unit [14].
- Output value y : is the final value produced by the perceptron.

As we can see, the perceptron algorithm is used to adjust the weights of the inputs in order to correctly classify a set of training data. The process of adjusting the weights is known as training the perceptron. The goal of training is to find the set of weights that will correctly classify all of the training data. Perceptrons are limited in their ability to learn, as they can only learn linearly separable functions. This means that they are not able to classify data that is not linearly separable [15]. However, they are still a useful tool for understanding the basics of neural networks and machine learning.

4.2 Definition of a multi-layer perceptron

A multi-layer perceptron (MLP) is a type of artificial neural network that is composed of multiple layers of perceptrons. It is called a "multi-layer" perceptron because it is composed

of at least three layers: an input layer, one or more hidden layers, and an output layer (Fig. 3)

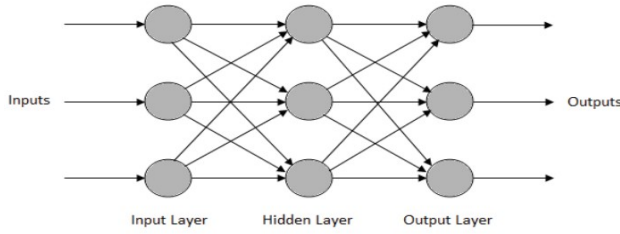


Fig. 3. Basic architecture of a multi layer perceptron with one hidden layer [14]

Each layer of the MLP consists of a set of perceptrons, where each perceptron in a layer receives input from all of the perceptrons in the previous layer, and produces an output that is passed on to the perceptrons in the next layer.

The output of the MLP is calculated by propagating the input through the layers of perceptrons and applying the activation function at each layer. The propagation is performed in a single direction, that is called a feedforward neural network [16]. The final output of the MLP is produced by the output layer, which may have one or more perceptrons depending on the number of classes in the classification problem, and in our situation we have two classes that are the efficiency and power output.

4.3 Backpropagation algorithm

Backpropagation is a training method, also called generalized delta rule, is a gradient descent method that minimizes the total squared error of the output computed by each perceptron [17]. Practically, the backpropagation algorithm works by propagating the error from the output layer of a network to its input layer, tweaking the weights of the network along the way to minimize the error.

The backpropagation algorithm consists of two phases: the forward pass and the backward pass. In the forward pass, the input data is fed through the network, and the output is calculated. In the backward pass, the error between the predicted output and the true output is calculated, and the weights of the network are adjusted accordingly. This process is repeated for each training example in the dataset, and the weights are updated after each iteration. Note that in the backpropagation algorithm, the activation function is sigmoid; it varies smoothly between 0 and 1 (Fig. 4) [15] :

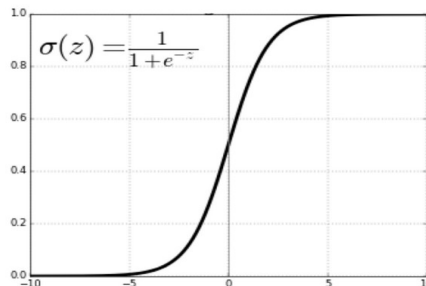


Fig. 4. Sigmoid activation function

5 Results and discussion

Before we give inputs to our MLP algorithm program, let us have a look on the output given by the lower and upper bounds of the three design parameters, in Eq. 3, Eq. 4 and Eq. 5.

For lower bounds we have :

$$d=0.018\text{ m}, \phi=60, f=10\text{ Hz} \tag{7}$$

Using the Schmidt analysis that can be found in a script in MATLAB® Simulink, we obtain the following output results :

$$W=3.535\text{ J}, P=30.535\text{ W}, \eta=0.675 \tag{8}$$

The corresponding P-V diagram is shown in Fig. 5.

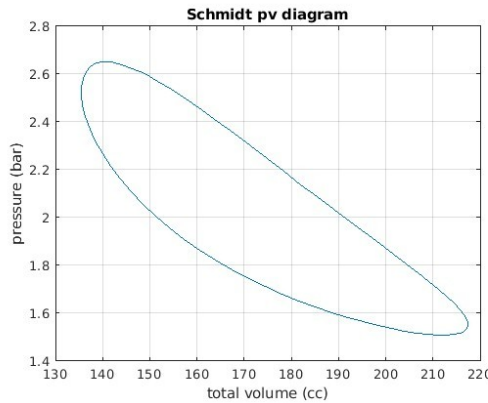


Fig. 5. P-V diagram output using the preset lower bounds values of the selected design parameters.

Now for the upper bounds :

$$d=0.16\text{ m}, \phi=120, f=20\text{ Hz} \tag{9}$$

We get the following output results :

$$W=2.385\text{ J}, P=40.771\text{ W}, \eta=0.526 \tag{10}$$

The corresponding P-V diagram is shown in Fig. 6.

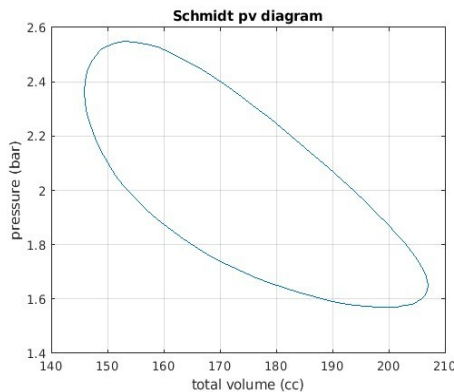


Fig. 6. P-V diagram of the optimal point cycle.

6 Conclusion and perspective

In this work, we used neural networks in a real life application, which is the optimization of the Stirling engine efficiency and output power, and thus by finding the optimal point that has three preselected design parameters as coordinates.

Using neural networks to optimize Stirling engines has the potential to greatly improve the efficiency and performance of these engines. By learning from large amounts of data and using pattern recognition to identify trends and predict outcomes, neural networks can help to optimize the design and operation of Stirling engines in a way that is not possible with traditional methods.

Other machine learning algorithms can also help find optimal parameterization, such as Genetic Algorithm or Conjugate Gradient method. However, it is important to note that the use of neural networks or other machine learning algorithms to optimize Stirling engines is still an active area of research, and there are many challenges and limitations that need to be addressed. These include the need for large amounts of labeled data to train the network, the time and resources required to train and fine-tune the network, the interpretability and explainability of the results and the practical use of the results to meet the needs of real world problems.

Despite these challenges, the use of neural networks to optimize Stirling engines holds great promise and has the potential to revolutionize the way these engines are designed and operated. As research and development in this area continues, it is likely that we will see significant progress in the use of neural networks to optimize Stirling engines, leading to more efficient and effective devices that can help to meet the increasing demand for clean and renewable energy. As a perspective, we will attempt to plot results for different hidden layers numbers, compare the results and also plot hyperplanes that are represented by the output power and efficiency which are expressed by each couple of the three preselected parameters; stroke length, phase angle and engine frequency.

References

1. P. Gras, *Le moteur Stirling et autres moteurs à air chaud*, Decoopman Sciences & Techniques, 2010.
2. S. Kundu, Technologies for utilization of biogas as energy source: a review, *International Journal of Engineering Research and Technology* 5 (3) 775-780, (2016).
3. P. Stouffs, "Machine thermique non conventionnelles: état de l'art, application, problème à résoudre," *Journée d'étude de SFT (France)*, octobre 1999.
4. McCulloch, W.S., Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics* 5, 115–133 (1943).
5. F. Rosenblatt, The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408, (1958).
6. F. Murtagh, Multilayer perceptrons for classification and regression, *Neurocomputing*, Volume 2, Issues 5–6, 183-197, 1991.
7. M. Buscema, Back Propagation Neural Networks. *Substance use & misuse*. 33. 233-70, 1998.
8. B. Kongtragool, S. Wongwises, "Investigation on power output of the gamma-configuration low temperature differential Stirling engines". *Renewable Energy*, 30(3), 465–476, 2005.
9. O. Taki, K.S. Rhazi, Y. Mejdoub "A Study of Stirling Engine Efficiency Combined with Solar Energy", *Advances in Science, Technology and Engineering Systems Journal*, vol. 6, no. 2, 837-845 (2021).
10. M. Ahmadi, A. Mohammadi, S. Dehghani, M. Barranco-Jiménez. Multiobjective thermodynamic-based optimization of output power of Solar Dish Stirling engine by implementing an evolutionary algorithm. *Energy Convers Manage*, 75, 438–45, 2013.

11. H. Karabulut, C. ÇINAR, F. Aksoy, H. Yücesu, Hüseyin, Improved Stirling engine performance through displacer surface treatment. *International Journal of Energy Research*. 34. 275 – 283, 2010.
12. T. Raymond, M.F. Zainudin, T. Zanariah, S. Ali, R. Bakar, Y. Ali, Study on the phase angle effect for alpha type stirling engine thermodynamics behavior, 2014.
13. Sh. Zare, A.R. Tavakolpour-Saleh, Frequency-based design of a free piston Stirling engine using genetic algorithm, *Energy*, Volume 109, 466-480, 2016.
14. J. Singh, R. Banerjee, A Study on Single and Multi-layer Perceptron Neural Network. 3rd ICCMC, 2019.
15. H. Mühlenbein, Limitations of multi-layer perceptron networks - steps towards genetic neural networks, *Parallel Computing*, Volume 14, Issue 3, 249-260, 1990.
16. G. Bebis, M.Georgiopoulos, Feed-forward neural networks. *IEEE Potentials*, 13(4), 27–31, 1994.
17. L. Fausett, *Fundamentals of Neural Network*. Prentice Hall, Hoboken, 289-300, 1994.