JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES

OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

Global Performance Optimization of an Internal Combustion Engine Using Surrogate-Based Optimization Modeling

Otimização global do desempenho de um motor de combustão interna por meio de modelagem de otimização baseada em substitutos

Article Info:
Editor handled this article: **Ibrahim Ademola Fetuga**
Article history: Received 2026-10-01 / Accepted 2026-11-20 / Available online 2026-11-24
doi: **10.18540/jcecvl12iss1pp25237**

Pedro Ribeiro de Oliveira
ORCID: https://orcid.org/0009-0002-5993-3350
Universidade Federal de Viçosa, Brazil
Email: pedro.oliveira7@ufv.br
Estêvão Faria Duarte Freitas
ORCID: https://orcid.org/0009-0003-6007-5119
Universidade Federal de Viçosa, Brazil
Email: estevao.freitas@ufv.br
Bianca Lamas Clem de Siqueira
ORCID: https://orcid.org/0009-0002-3465-665X
Universidade Federal de Juiz de Fora, Brazil
Email: 166603593769@estudante.ufjf.br
Júlio César Costa Campos
ORCID: https://orcid.org/ 0000-0002-9488-8164
Universidade Federal de Viçosa, Brazil
Email: julio.campos@ufv.br
Álvaro Messias Bigonha Tibiriçá
ORCID: https://orcid.org/0000-0002-3300-1988
Universidade Federal de Viçosa, Brazil
Email: alvaro.tibirica@ufv.br
Antônio Marcos de Oliveira Siqueira
ORCID: https://orcid.org/0009-0008-9051-4633
Universidade Federal de Viçosa, Brazil
Email: antonio.siqueira@ufv.br
Washington Orlando Irrazabal Bohorquez
ORCID: https://orcid.org/0000-0002-9762-0665
Universidade Federal de Juiz de Fora, Brazil
Email: washington.irrazabal@ufjf.br
Rogério Fernandes Brito
ORCID: https://orcid.org/0000-0002-6833-7801
Universidade Federal de Itajubá, Brazil
Email: rogbrito@unifei.edu.br

Abstract

The automotive industry faces increasing regulatory pressure, driven by standards such as EU 2024/1257 and CONAMA No. 492. These guidelines require drastic reductions in emissions of nitrogen oxides (NOx), particulate matter, and greenhouse gases, which puts pressure on the development of cleaner engines. In this context, the calibration of the Engine Control Unit (ECU)

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

has become a complex optimization challenge. Traditionally, calibration relied on extensive experimental testing on dynamometers. However, this method has become impractical due to the high financial cost and the long development time required. As an alternative, Model-Based Calibration (MBC) has emerged, enabling virtual exploration of the design space. Nevertheless, high-fidelity simulations of diesel combustion require a computational cost that is prohibitive for iterative optimization routines. Given this scenario, this work proposes a Surrogate-Based Optimization (SBO) framework applied to a real diesel engine. The methodology focuses on the analysis of six parameters: engine speed, intake pressure, injection timing, injection duration, injected fuel quantity, and exhaust gas recirculation (EGR) mixture. The process uses the ANSYS CHEMKIN software to emulate engine behavior and generate high-fidelity data. Based on this dataset, a Neural Network is trained to act as a computationally inexpensive surrogate model. This model is integrated into an active learning cycle, where new conditions suggested by the optimization algorithm are validated by the physical simulator and reincorporated into the network's training. Convergence of the process was achieved in only 252 engine cycles. The surrogate model presented a Mean Absolute Percentage Error (MAPE) of approximately 2% and coefficients of determination (R^2) close to 1 for all variables. In the test case, the framework achieved an 11.70% reduction in fuel consumption and a 58.15% decrease in NOx emissions, at the cost of a slight reduction in power and torque.

Keywords: Diesel. NOx. Neural Network. Python. Optimization.

Resumo

A indústria automotiva enfrenta uma pressão regulatória crescente, impulsionada por normas como a UE 2024/1257 e a CONAMA nº 492. Essas diretrizes exigem reduções drásticas nas emissões de óxidos de nitrogênio (NOx), material particulado e gases de efeito estufa, o que pressiona o desenvolvimento de motores mais limpos. Nesse contexto, a calibração da Unidade de Controle do Motor (ECU) tornou-se um desafio de otimização complexo. Tradicionalmente, a calibração dependia de extensivos testes experimentais em dinamômetro. Contudo, esse método se mostra inviável devido ao elevado custo financeiro e ao longo tempo de desenvolvimento exigido. Como alternativa, surge a Calibração Baseada em Modelo (MBC), que permite a exploração virtual do projeto. Entretanto, as simulações de alta fidelidade da combustão diesel demandam um custo computacional proibitivo para rotinas de otimização iterativas. Diante desse cenário, este trabalho propõe um framework de Otimização Baseada em Substitutos (SBO) aplicado a um motor diesel real. A metodologia foca na análise de seis parâmetros: rotação, pressão de alimentação, tempo de injeção, duração da injeção, quantidade de combustível injetado e mistura de recirculação de gases de escape (EGR). O software ANSYS CHEMKIN emula o comportamento do motor e gera dados de alta fidelidade, a partir dos quais uma Rede Neural é treinada para atuar como modelo substituto computacionalmente barato. Esse modelo é integrado a um ciclo de aprendizado ativo, onde novas condições sugeridas pelo algoritmo de otimização são validadas pelo simulador físico e reincorporadas ao treinamento da rede. A convergência foi atingida em apenas 252 ciclos do motor. O modelo substituto apresentou um Erro Percentual Absoluto Médio (MAPE %) de cerca de 2% e Coeficiente de Determinação (R^2) próximos de 1 para todas as variáveis. No caso teste, o framework alcançou uma redução de 11,70% no consumo de combustível e uma queda de 58,15% nas emissões de NOx, às custas de uma pequena redução na potência e no torque.

Palavras-chave: Diesel. NOx. Rede Neural. Python. Otimização.

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

1. Introduction

The contemporary automotive industry operates under an increasing regime of regulatory pressures, motivated by the need to reduce the environmental impact of the transportation sector. The internal combustion engine (ICE), and specifically the compression ignition (diesel) engine, is at the center of an engineering effort to reconcile performance with low pollutant emissions. Regulations such as UE 2024/1257 (2024), in Europe, and CONAMA nº 492 (2018) in Brazil, impose strict limits on nitrogen oxides (NOx), particulate matter (PM), and greenhouse gases (CO and CO2). This scenario has transformed engine calibration from an empirical "fine-tuning" practice into a rigorous scientific discipline, dependent on computational modeling and complex mathematical optimization.

The calibration of the Engine Control Unit (ECU) represents a constrained, nonlinear, high-dimensional optimization problem. In a modern diesel engine with common-rail direct injection, variable geometry turbocharger, and exhaust gas recirculation (EGR) circuits, the number of control and monitoring parameters can reach the thousands. Of these, dozens are decision variables, which include injection timing (SOI), rail pressure, the multiple injection strategy (pilot, main, post), boost pressure, and EGR rates, and must be optimized simultaneously for each operating point of the engine map.

In order to achieve this optimization, it has become common engineering practice to carry out extensive dynamometer bench tests. However, these experimental rounds can be extremely long and costly, especially due to the large number of possible combinations of ECU control parameters. To explore this design space, calibration engineering began to adopt Model-Based Calibration (MBC) as a tool, which has been becoming an industry standard (Gutjahr et al., 2024; Hoffmann et al., 2015). Its premise is to use mathematical models to predict engine behavior and allow optimization algorithms to evaluate parameter combinations virtually, reducing dynamometer testing time.

Software that enables accurate modeling of engine operation is, therefore, fundamental. However, diesel combustion modeling involves transient processes of spray atomization, evaporation, turbulent mixing, and autoignition, followed by complex chemical reactions that govern the formation of emissions. A detailed reaction mechanism for diesel may include hundreds of chemical species and thousands of elementary reactions, so that the simulation of a single engine cycle at this level of detail may require hours of processing on high-performance clusters.

Given this scenario, Surrogate-Based Optimization (SBO) emerges as an alternative to this bottleneck. SBO consists of constructing a computationally efficient approximation of the high-fidelity simulation, called a surrogate model or metamodel. After its development, this model can be evaluated in milliseconds, enabling extensive exploration of the design space and significantly reducing the number of simulations required for optimization, as demonstrated by Kim and Boukouvala (2020) and Brahma and Rutland (2003). In this context, the present article proposes the development of a framework for the optimization of a real diesel engine, integrating high-fidelity engineering simulations - ANSYS CHEMKIN, with an auxiliary Neural Network model, with the objective of reducing costs and accelerating the calibration process within the SBO methodology.

2. Theoretical Background

2.1 SBO Theory

The *Surrogate Based Optimization* (SBO) methodology, also known as metamodel-assisted optimization, consists of replacing computationally expensive evaluation functions with

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

mathematical approximations of low computational cost. In engineering problems, the objective function generally depends on complex physical simulators such as CFD models, detailed combustion processes, or multiphysics simulations, whose evaluation may require minutes or hours for each point of the design space. SBO stands out by introducing an auxiliary model capable of reproducing the global behavior of the function accurately but at a lower computational cost. The theory developed below is described in Forrester et al. (2008) and Koziel and Leifsson (2013).

Consider a vector of design variables $x \in \Omega \subset \mathbb{R}^d$ and an unknown real function:

$$f(x): \mathbb{R}^d \rightarrow \mathbb{R}^m, \quad (1)$$

evaluated only by means of an expensive simulator. The classical optimization problem consists of finding:

$$x^* = argmin_{x \in \Omega} J(x, f(x)), \quad (2)$$

where J represents the objective function or performance criterion. Since the cost of evaluating $f(x)$ is high, it becomes infeasible to apply traditional methods that require thousands of evaluations.

SBO introduces a surrogate model $s(x)$, constructed from a finite set of samples:

$$D = \quad (3)$$

This model can be represented by neural networks, *Kriging* (*Gaussian Process Regression*), radial basis functions, or polynomial expansions. The objective is to obtain an approximation such that:

$$s(x) \approx f(x) \forall x \in \Omega, \quad (4)$$

minimizing the generalization error in the domain of interest. Under this formulation, the optimization is carried out over a function that is inexpensive to evaluate:

$$x_s^* = argmin_{x \in \Omega} J(x, s(x)). \quad (5)$$

The initial Design of Experiments (DoE) stage plays a central role in the development of SBO. The objective is to select points that maximize the statistical information obtained about the design space. Techniques such as *Latin Hypercube Sampling* or low-discrepancy sequences aim to minimize correlations between variables and ensure uniform coverage of the domain, reducing the variance of the surrogate model estimate. From a probabilistic point of view, it is advantageous to maximize the mutual information between the sampled data and the unknown function.

After the development of the surrogate, the optimization can be performed by deterministic or stochastic methods, such as genetic algorithms, gradient descent, or evolutionary strategies. An important aspect of SBO is the control of the approximation error. Since $s(x)$ is only an estimate, there is a risk of convergence to artificial optima introduced by the model. To mitigate this problem, many approaches, such as the one used in this study, employ adaptive strategies (*adaptive sampling* or *infill criteria*), in which new points are iteratively added to the training set in promising regions or where the model uncertainty is greatest.

Regarding engine calibration, SBO has been becoming an important tool. Pal et al. (2019) used Kriging to adjust the variable geometry turbocharger (VGT) position, the exhaust gas recirculation (EGR) valve position, and the start of injection (SOI), reporting computational savings of more than 60%. Kesgin (2004) implemented a neural network to predict efficiency values and NOx emissions and to adjust multiple operating parameters of a natural gas turbine.

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

2.2 Modeling in ANSYS CHEMKIN

Diesel combustion is a multiphase process that involves the injection of liquid fuel into the chamber, followed by jet atomization, droplet evaporation, mixing with the compressed air, and autoignition. The chemical transformation of diesel, which consists of a complex mixture of paraffinic, naphthenic, and aromatic hydrocarbons, generates final products such as CO2 and H2O, in addition to pollutants such as NOx and soot. The detailed description of this process involves thousands of intermediate species and elementary reactions.

NOx formation occurs through different routes, such as the Zeldovich thermal mechanism, which exhibits a strong exponential dependence on local temperature peaks. Soot formation also depends on multiple chemical and physical steps. The process includes the generation of gaseous precursors, such as Polycyclic Aromatic Hydrocarbons, growth by surface mass addition, particle coagulation, and subsequent oxidation. Since these steps are highly sensitive to temperature, local mixing, and residence time, the modeling requires detailed kinetic mechanisms and adequate treatment of transport properties.

ANSYS CHEMKIN handles this complexity by solving a stiff system of Ordinary Differential Equations derived from the law of mass action, coupled with the energy and species conservation equations. The software uses detailed chemical mechanisms that specify thermodynamic properties, transport coefficients, and kinetic constants for all the species involved. Since commercial diesel has a variable composition, surrogate fuels are used to represent its properties. Simple surrogates, such as n-heptane and n-dodecane, are often used for ignition and ignition delay studies.

For more realistic applications, multicomponent mixtures are used, with mechanisms that include components such as n-hexadecane and 1-methylnaphthalene to represent different classes of hydrocarbons (alkanes, aromatics) found in the real fuel, allowing a more faithful representation of the reactivity and soot formation properties of commercial diesel. In this work, all simulations are set up following good practices instructed by the Pro Theory Manual (ANSYS, 2022a) and the Chemkin Tutorials (ANSYS, 2022b)

2.3 Neural Network Theory

From a mathematical point of view, an artificial neural network can be interpreted as a parametric approximator of nonlinear functions. The Universal Approximation Theorem states that a feedforward neural network with at least one hidden layer and a finite number of neurons can approximate any continuous (complex) function with arbitrary precision, provided that the activation function is adequate (Hornik et al., 1989). The following formulation is presented based on the book by Haykin (2009):

Consider an input vector $x \in \mathbb{R}^n$, which represents the system parameters (for example, ECU variables). Each neuron initially performs a linear combination of the inputs:

$$z=w^{\top}x+b, \tag{6}$$

where w is the vector of synaptic weights and b is the *bias*. Then, a nonlinear activation function $\phi(\cdot)$ is applied, producing the neuron output:

$$a=\phi(z). \tag{7}$$

The introduction of nonlinearity is fundamental, since a composition of linear transformations alone would be equivalent to a single linear transformation, which is insufficient to represent the complex relationships present in real physical systems. Functions such as ReLU, hyperbolic tangent, and sigmoid allow the network to model highly nonlinear behaviors.

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

In a multilayer network, the outputs of one layer become the inputs of the next layer. For a network with L layers, the complete mapping can be written as a successive composition of functions:

$$y=f(x;\Theta)=\phi^{(L)}\left(W^{(L)}\phi^{(L-1)}(\dots\phi^{(1)}(W^{(1)}x+b^{(1)})\dots)+b^{(L)}\right), \quad (8)$$

where $\Theta=\{W^{(l)},b^{(l)}\}_{l=1}^{L}$ represents the total set of trainable parameters. Thus, training the network is equivalent to finding the values of θ that best approximate an unknown function relating the inputs and outputs of the physical system.

Learning is formulated as an optimization problem. Given a dataset $\{(x_i,y_i)\}_{i=1}^{N}$, a loss function $J(\Theta)$ is defined, usually the Mean Squared Error (MSE):

$$J(\Theta)=\frac{1}{N}\sum_{i=1}^{N}\|y_i-f(x_i;\Theta)\|^2. \quad (9)$$

The minimization of this function is performed by gradient-based methods. The efficient computation of the gradient $\nabla_\Theta J$ is possible through the *backpropagation* algorithm, which systematically applies the chain rule of differential calculus to propagate the error from the output layer towards the initial layers. Each weight is adjusted in proportion to its contribution to the final error:

$$\Theta_{k+1}=\Theta_k-\eta\nabla_\Theta J(\Theta_k), \quad (10)$$

where η is the learning rate. Modern optimizers, such as Adam, introduce adaptive estimates of the first- and second-order moments of the gradient, which accelerates convergence and increases numerical stability in high-dimensional problems.

From a statistical perspective, training neural networks involves a balance between fitting and generalization. A network with many parameters has high representational capacity and can perfectly fit the training data, but runs the risk of learning noise, a phenomenon known as *overfitting*. *Dropout* deactivates part of the neurons during training and prevents excessive co-adaptations, while *Early Stopping* interrupts training when the validation error begins to increase, indicating a loss of generalization capacity.

Neural networks can be interpreted as high-dimensional nonlinear regression methods capable of building hierarchical representations of the data. The first layers learn basic combinations of the input variables, while deeper layers capture complex interactions between physical parameters. This characteristic makes neural networks particularly suitable to act as surrogate models of computationally expensive simulators, enabling fast estimates while maintaining good fidelity to the physical relationships.

3. Workflow

3.1 Engine Presentation

The object of study is a single-cylinder, four-stroke, direct injection diesel engine, model TD 212 (TecQuipment, United Kingdom). This engine is installed at the Thermal Systems Laboratory (LST) of the Universidade Federal de Viçosa (UFV) and will be used as the basis for modeling in the ANSYS software. Table 1, below, presents the detailed specifications of the engine:

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

Table 1 – Engine Specifications.

Item	Specification
Fuel type	Diesel
Exhaust Diameter	1″ BSP - nominal
Ignition system	Autoignition
Absolute Maximum Power	3.5 kW (4.8 hp) at 3600 RPM
Continuous Rated Power	3.1 kW at 3000 RPM
Diameter (Bore)	69 mm
Stroke / Crank Radius	62 mm / 31 mm
Connecting Rod Length	104 mm
Engine Capacity	232 cm^3
Compression Ratio	22:1
Oil Type	Multigrade SAE 5W-40
Cylinders	1
Cooling	Air
Noise level	May exceed 70 dB

Additional information was gathered, such as internal engine dimensions and injector characteristics, required for modeling in CHEMKIN. Next, a simulation was carried out, from which performance curves were generated and compared with experimental data obtained on a dynamometer bench, provided by the manufacturer. This step is important, as it ensures that the engine behavior represented in the software makes physical sense. The result is presented in Figure 1.

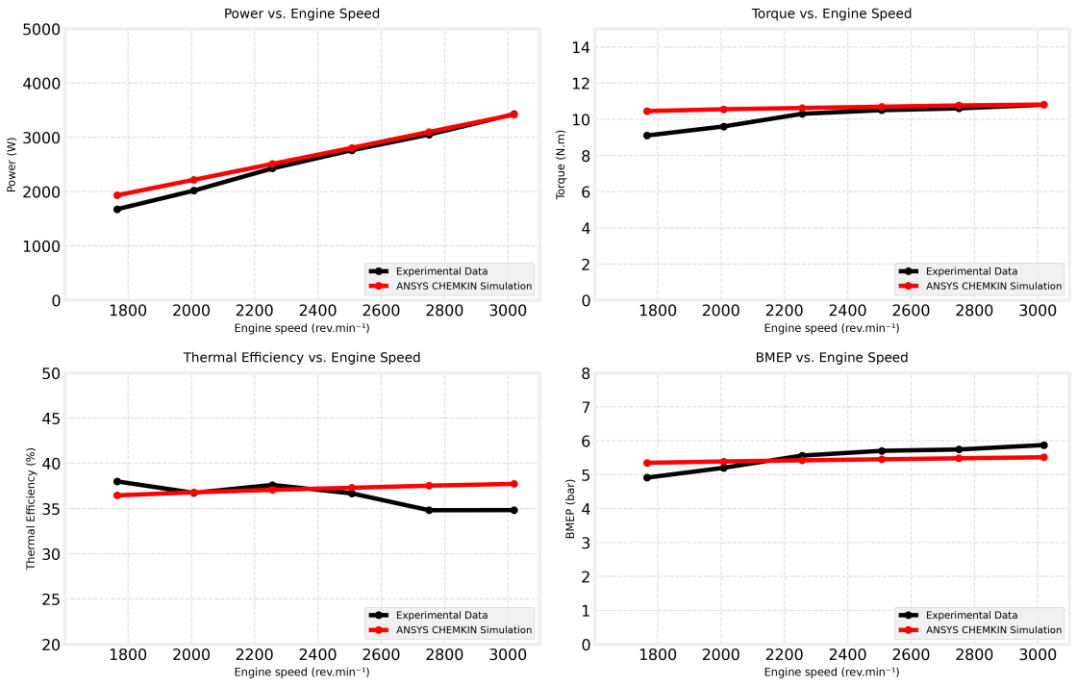

Figure 1 – Experimental Data (Manufacturer) x Numerical Simulation.

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

3.2 SBO Implementation

Each simulation performed in *CHEMKIN* represents a complete engine cycle, including air intake, diesel injection into the chamber, combustion, and exhaust of the gases. To optimize performance over the entire operating range, the simulations are executed in batches containing six cases, with engine speeds between 1800 and 3000 RPM. The software records the inputs and results in. INP and `.OUT` files. After the completion of each batch, a Python *script* extracts and processes the data, calculating an individual performance factor called Score.

The Score is defined based on Multi-Criteria Decision Analysis. The objective is to transform variables with different units and scales, such as kilowatts (kW), newton-meters (Nm), and parts per million (ppm), into a dimensionless coefficient between 0 and 1 that represents the global balance of the engine. The metrics are divided into two groups. The first contains the maximization parameters, in which high values are desirable, such as Power and Thermal Efficiency. The second contains the minimization parameters, such as Consumption and NOx.

Normalization is performed with the `MinMaxScaler` from the *Scikit-Learn* library, which applies a linear transformation to map the data into the interval [0,1], according to the equation:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{11}$$

For the minimization parameters, the scale is inverted by means of the expression $(1-x_{norm})$, ensuring that lower original values result in higher scores. Thus, all indicators have the same orientation, in which values close to 1 indicate better performance. The final Score is obtained by a weighted sum, with previously defined weights w_i whose sum equals 1 ($\sum w_i = 1$). The choice of these weights determines the calibration strategy, allowing priority to be given to consumption and emissions or to power and torque performance. An example of a possible weight distribution is: [Power: 0.2, Torque: 0.2, Thermal Efficiency: 0.15, Consumption: 0.2, NOx: 0.25].

The Score values are then used as the target for training the neural network. The model consists of a deep Artificial Neural Network (*Deep Learning*) for multivariable regression, implemented in *PyTorch*. The architecture is of the *feed-forward* type with multiple hidden layers.

The model receives six input variables and processes them through linear layers that expand the dimensionality to 256 and 512 neurons. The activation function adopted is the hyperbolic tangent ($tanh$), which introduces nonlinearity and keeps the outputs centered at zero. Between the layers, *Dropout* was implemented with a rate of 0.1, a regularization technique that randomly deactivates 10% of the neurons during each training iteration, forcing the network to learn more robust representations and significantly reducing the risk of *overfitting*.

Training uses the Adam optimizer and the Mean Squared Error loss function (*MSELoss*). A *scheduler* of the `ReduceLROnPlateau` type monitors the loss and reduces the learning rate η when there is no improvement after a certain number of epochs. The model executes the *forward pass* to generate the predictions, calculates the loss with respect to the target, and then performs the *backward pass* (*backpropagation*).

After training, an optimization module is applied with the objective of generating new sets of parameters that maximize the Score predicted by the network. The process starts with a random input vector normalized in the interval [0,1], which is iteratively adjusted by the Adam algorithm. The loss function incorporated into the optimization process seeks to maximize the Score variable, while imposing a penalty if the input vector comes excessively close to the training data. This

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

penalty is assigned to force the model to explore the variable space and prevent the predicted parameters from concentrating only around the best set already known in the database.

The newly proposed conditions are then sent to *CHEMKIN* for evaluation. This procedure characterizes an *active learning* cycle, in which the computationally expensive simulations are directed to regions with greater potential for improvement. In this way, the model is progressively refined around the most promising calibrations, reducing the total number of runs required to reach an optimal solution. All neural network calculations are executed using the NVIDIA CUDA acceleration feature.

4. Problem Application

The process begins with the definition of the operating ranges of each variable, presented in Table 2. These ranges delimit the search space and define the parameters that can be adjusted by the neural network. After this space is defined, the parameters of the initial iteration are defined. At this stage, the values are chosen so as to adequately represent the entire domain of the variables, prioritizing combinations close to the lower and upper limits of each range. This strategy provides greater data diversity to the model. The neural network is configured to automatically constrain any predicted value that exceeds the established limits.

Table 2 – Limits and ranges of the experiment - DoE.

Parameter	Lower Limit	Upper Limit
Engine Speed (RPM)	1766	3019
Pressure (bar)	1.4	2.2
Injection Timing BTDC (degrees)	12	20
Injection Duration (degrees)	6	14
Injected Fuel Quantity (mg)	4	12
EGR Rate (%)	0%	40%

The first iteration aims to provide the network with a global understanding of engine behavior. It consists of twelve sets of batches, each containing six simulations, keeping the engine speed variation between 1800 and 3000 RPM constant to ensure evaluation over the entire operating regime. The results obtained from the initial training base.

After this stage, the following iterations use parameters suggested by the neural network itself. Each new iteration adds six sets of batches to the database, progressively directing the simulations towards regions with greater performance potential. This procedure increases the density of samples near the most promising solutions and improves the accuracy of the model in these regions.

The iterative process continues until the Score value converges for each engine speed analyzed. Convergence is assumed when the Score does not vary by more than 0.05 per iteration, in absolute value. The final solution is determined by the parameter combination that achieves the highest Score within the accumulated database. This combination defines the optimal calibration set, corresponding to the final ECU map of the engine.

5. Results

After the start of the optimization process, convergence was achieved in six iterations, totaling 26 batches of simulations. A weight distribution that strongly prioritizes the reduction of consumption and NOx emissions was adopted, defined as: Power = 0.05; Torque = 0.05; Thermal Efficiency = 0.10; Consumption = 0.4; NOx = 0.4. This distribution was chosen to simulate a

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

scenario of strict compliance with the new emission standards, prioritizing the environmental aspect over pure performance.

With the model training completed, a statistical analysis is performed to evaluate the quality of the fit to the data. The model is then subjected to a test set containing independent simulations, whose parameters are within the established limits of the experiment, but which are not part of the database used to train it. Figure 2 presents the distribution of the network predictions compared with the test values. It can be observed that the predicted values do not deviate much from what is expected, with all variables obtaining a coefficient of determination close to 1.

Figure 3 presents the residual analysis of the regression model. Each histogram shows the distribution of the difference between the actual and predicted values for different variables, and makes it possible to identify whether the error behaves randomly or presents a systematic bias. In a well-fitted model, the errors should ideally follow a normal (Gaussian) distribution centered at zero (indicated by the red dashed line); if the data are shifted or show asymmetry, this indicates that the model is consistently underestimating or overestimating the results. All variables showed satisfactory statistical behavior; although the distributions exhibit a slight bias, the relative error remained low. This indicates that the trained model represents the engine operation well.

Table 3 shows the main statistical metrics of the network. All variables presented a high coefficient of determination (R^2) and a low mean absolute percentage error (MAPE), which indicates excellent generalization capacity. However, NOx presented high values of MSE, RMSE, and MAE. This result is explained by the greater order of magnitude of the absolute values of this variable, expressed in ppm, compared with the others. Even so, the R^2 and MAPE values for NOx are consistent with those of the other variables.

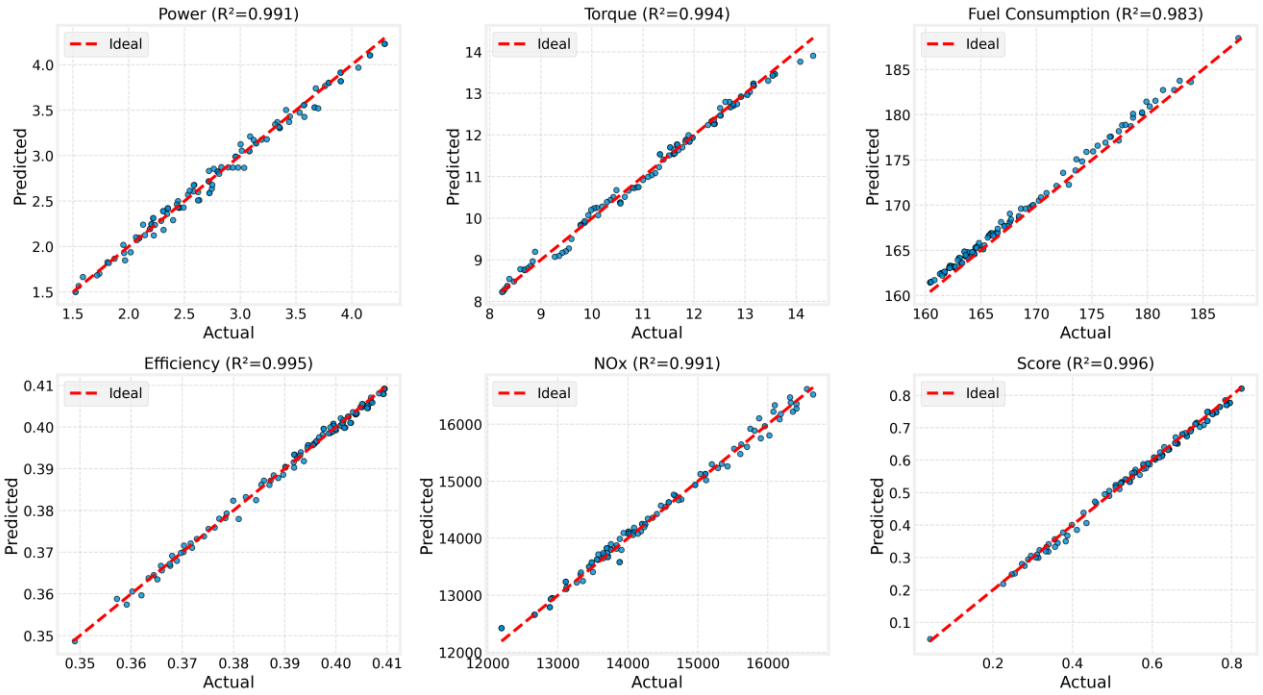

Figure 2 – Actual x Predicted Values.

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

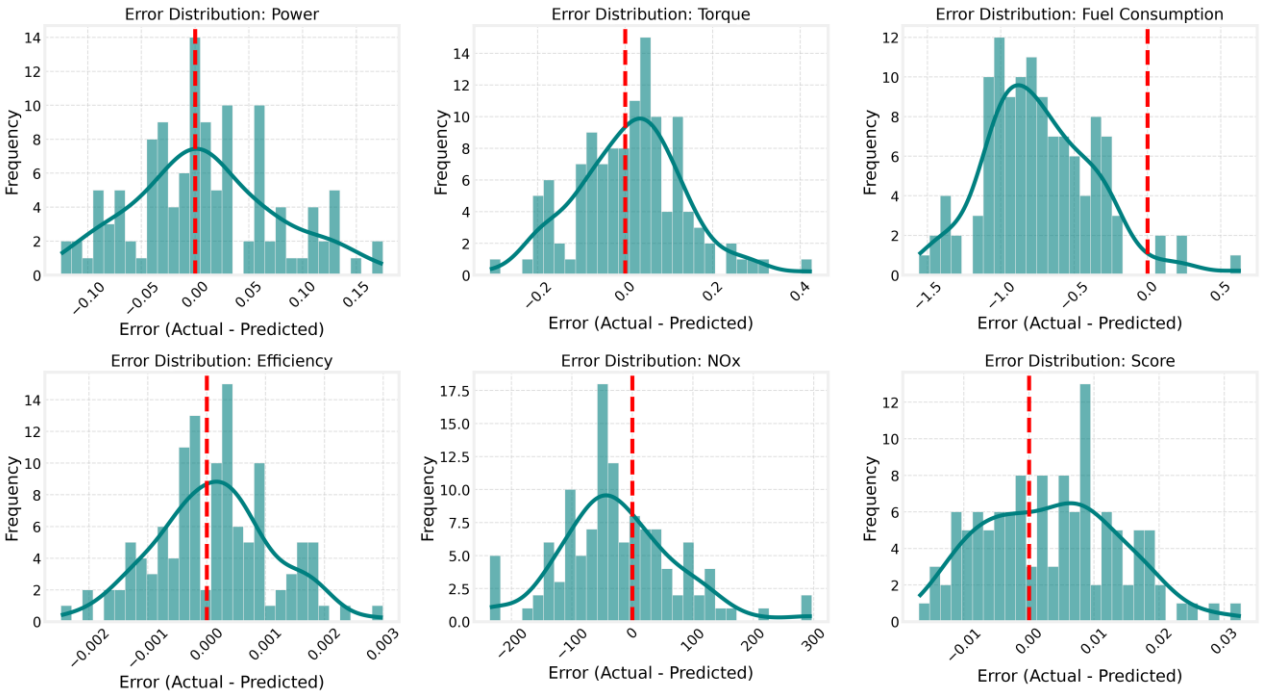

Figure 3 – Residual Analysis.

Table 3 – Performance metrics of the regression model.

Variable	MSE	RMSE	MAE	R^2	MAPE (%)
Power	0.0045	0.0669	0.0519	0.9909	1.89
Torque	0.0152	0.1235	0.0961	0.9938	0.88
Consumption	0.6826	0.8262	0.7565	0.9835	0.45
Efficiency	0.0000	0.0010	0.0008	0.9952	0.20
NOx	9978.1025	99.8905	78.3454	0.9913	0.55
Score	0.0001	0.0108	0.0089	0.9957	1.81

To facilitate visualization, graphs illustrating the development of the process were generated. Figure 4 presents the evolution of the curves of each performance variable as a function of engine speed (RPM), with each curve corresponding to one iteration. It can be observed that the Score curves show a growth trend over the iterations, as expected, since this metric guides the optimization. The other variables are adjusted by the neural network so as to balance the global performance and maximize the Score according to the adopted weight distribution.

Figure 5 presents the variables as a function of the iterations, with each curve representing a distinct RPM range. This representation shows how the neural network directs the engine operation throughout the optimization process. In this figure, some instability is observed in the initial iterations. This behavior occurs due to the penalty imposed on the network, which encourages it to explore new points. As a consequence, some selected parameters may deviate from the ideal path; however, this tends to be corrected in the following iterations.

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

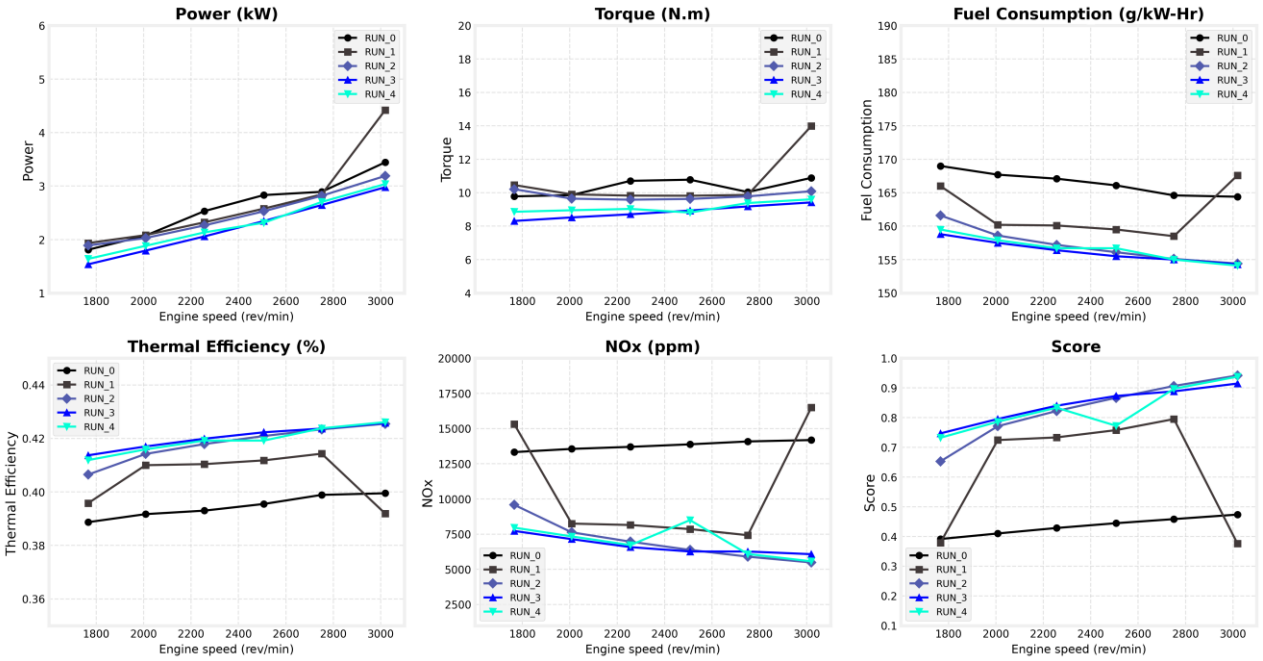

Figure 4 – Comparison of the RUNs.

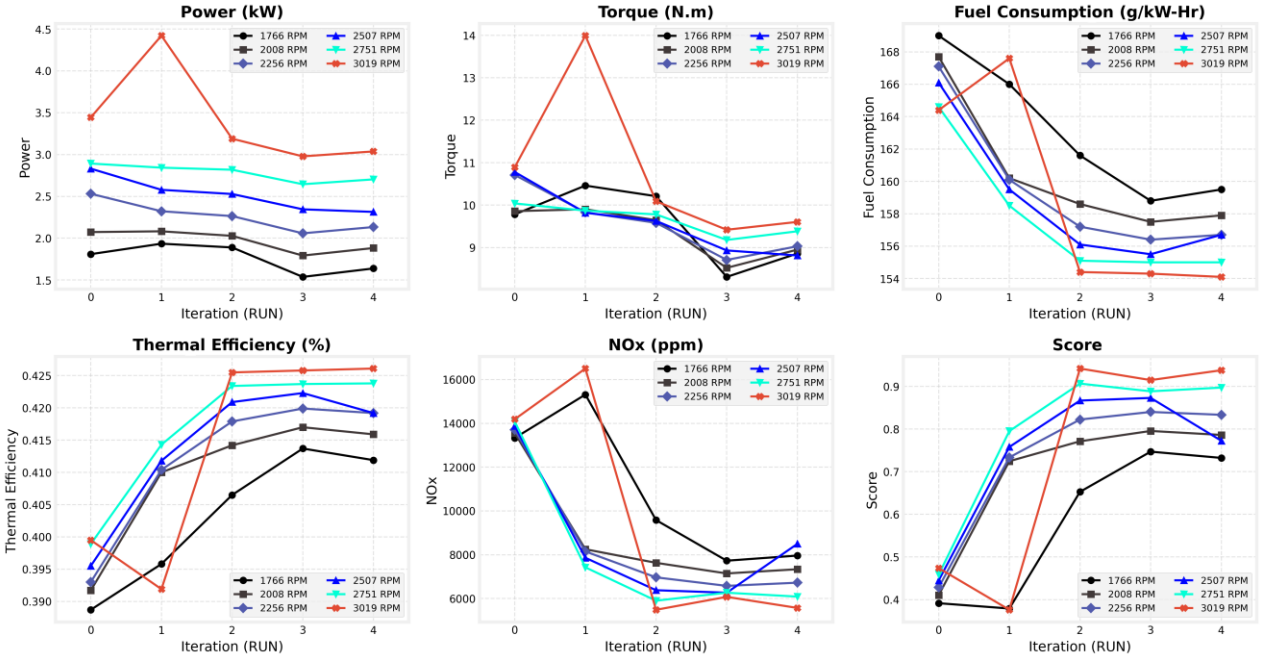

Figure 5 – Evolution of the variables per iteration.

Finally, Table 4 presents the final solution of the optimization problem. This table corresponds to the ECU calibration map, containing the parameters that provide the best results according to the weights defined for the Score. Naturally, a different choice of weights results in a different map, which allows the development of different calibration strategies for different objectives.

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
OPEN ACCESS
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

Table 4 – Matrix of Variables and Engine Speeds.

-	-	Engine Speed	-	Engine Speed	-	Engine Speed
Variable	1766.0	2008	2260	2507	2751	3019
Pressure	1.95	2.1	2.12	2.2	2.2	2.2
EGR	32	35	34	30	22	12
Injection Timing	13.1	14	15.3	16.6	17.5	18.4
Injection Duration	7	8.2	9.5	11.1	12.7	13.4
Fuel Quantity	5.2	6.4	8.5	9.4	10.8	11.8

6. Conclusions

The process reached convergence and resulted in a satisfactory solution. The intermediate steps, such as the configuration of the simulations and the data processing, were automated by means of Python scripts. The model analysis indicated a sufficiently accurate fit, even with a reduced amount of data. All performance variables analyzed presented a Mean Absolute Percentage Error (MAPE %) of about 2% and Coefficients of Determination (R^2) close to 1. Compared with the original engine calibration, the solution found resulted in a reduction in fuel consumption of 11.70% and a decrease in NOx emissions of 58.15%, but at the cost of a decrease in power and torque of 16.43% and 17.75%, respectively. This reduction can be explained by the weight assignment: [Power: 0.05, Torque: 0.05, Thermal Efficiency: 0.10, Consumption: 0.4, NOx: 0.4], which strongly prioritizes the reduction of consumption and NOx emissions to the detriment of the other parameters. A different assignment decision could present more balanced results.

The calibration process was carried out with only 252 individual CHEMKIN cycles, a reduced number compared with the total search space. If the 5 experimental parameters were varied independently at 4 distinct points for each of the 6 engine speeds, 6×4^5=6144 simulations would be required, without the guarantee of convergence to the optimal solution. This saving of resources is corroborated by the literature, which indicates that the use of surrogate models in optimization processes can reduce the computational cost by more than 50%.

The research addressed a reduced-scale calibration problem. In real applications, diesel engine ECUs operate with a significantly larger number of variables and with a broader set of engine speed samples, exceeding the six ranges between 1800 and 3000 RPM considered in this study. Even so, the methodology presented is easily scalable and can be applied to highly complex industrial problems.

Although the results were satisfactory, there are still possibilities for improving the structure of the process. A more in-depth analysis of the number of simulations in the initial batch and of the number of additional batches may indicate strategies to further reduce the computational cost.

Acknowledgements

The authors thank the Universidade Federal de Viçosa (UFV)-Universidade Federal de Juiz de Fora (UFJF) e a Universidade Federal de Itajubá (UNIFEI). This work was carried out with the support of the Fundação de Apoio à Pesquisa do Estado de Minas Gerais (FAPEMIG) and the Fundação Arthur Bernardes (FUNARBE) - Process PPE-00023-21, Governo de Minas Gerais - Seinfra-MG, Núcleo de Desenvolvimento Tecnológico Ferroviário do Estado de Minas Gerais - NDF, and CNPq.

JCEC
THE JOURNAL OF ENGINEERING AND EXACT SCIENCES
The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

References

ANSYS. (2022a). *Ansys Chemkin-Pro Theory Manual* (Release 2022 R1). ANSYS, Inc.

ANSYS. (2022b). *Ansys Chemkin-Pro Tutorials* (Release 2022 R1). ANSYS, Inc.

Brahma, I., & Rutland, C. (2003). Optimization of Diesel Engine Operating Parameters Using Neural Networks. https://doi.org/10.4271/2003-01-3228

CONAMA nº 492. (2018). *Resolução CONAMA nº 492, de 20 de dezembro de 2018: Estabelece a Fase L8 do Programa de Controle da Poluição do Ar por Veículos Automotores (PROCONVE)* (Resolução nº 492/2018). Ministério do Meio Ambiente, Brasília, DF, Brasil.

Forrester, A. I. J., Sobester, A., & Keane, A. J. (2008). *Engineering Design via Surrogate Modelling: A Practical Guide*. Wiley.

Gutjahr, T., Kruse, T., & Huber, T. (2024). Advanced Modeling and Optimization for Virtual Calibration of Internal Combustion Engines. https://doi.org/10.4271/2024-01-3630

Haykin, S. (2009). *Neural Networks and Learning Machines* (3rd ed.). Pearson.

Hoffmann, Schrott, Huber, & Kruse. (2015). Model-based Methods for the Calibration of Modern Internal Combustion Engines. *MTZ worldwide, 76*. https://doi.org/10.1007/s38313-014-1024-9

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer Feedforward Networks Are Universal Approximators. *Neural Networks, 2*(5), 359–366.

Kesgin, U. (2004). Genetic algorithm and artificial neural network for engine optimisation of efficiency and NOx emission. *Fuel, 83*, 885–895. https://doi.org/10.1016/j.fuel.2003.10.025

Kim, S., & Boukouvala, F. (2020). Surrogate-Based Optimization for Mixed-Integer Nonlinear Problems. *Computers & Chemical Engineering, 140*, Article 106847. https://doi.org/10.1016/j.compchemeng.2020.106847

Koziel, S., & Leifsson, L. (2013). *Surrogate-Based Modeling and Optimization: Applications in Engineering*. Springer.

Pal, A., Wang, Y., Zhu, L., & Zhu, G. (2019). Engine Calibration Optimization Based on its Surrogate Models. https://doi.org/10.1115/DSCC2019-8984

UE 2024/1257. (2024). *Regulamento (UE) 2024/1257 do Parlamento Europeu e do Conselho, relativo à homologação de veículos a motor e aos requisitos de emissões de CO2*. Jornal Oficial da União Europeia, Bruxelas, Bélgica.

CC BY OPEN ACCESS

The Journal of Engineering and Exact Sciences – jCEC
journal homepage: https://periodicos.ufv.br/jcec
eISSN: 2527-1075 - ISSN: 2446-9416

Declarations

Authors' contributions

Julio C. C. Campos, Pedro Ribeiro de Oliveira, Estêvão Faria Duarte Freitas, Bianca Lamas Clem de Siqueira: Conceptualization; Data curation; Formal analysis; Investigation; Methodology; Resources; Validation; Visualization; Writing – original draft; Writing – review & editing.

Washington Orlando Irrazabal Bohorquez, Álvaro M. Bigonha Tibiriçá, Antônio Marcos de Oliveira Siqueira, Rogério Fernandes Brito: Conceptualization; Data curation; Formal analysis; Investigation; Methodology; Project administration; Resources; Supervision; Validation; Visualization; Writing – original draft; Writing – review & editing.

All authors reviewed and approved the final manuscript and agree to be accountable for their contributions and for addressing questions concerning the integrity of the work.

Funding information

The authors received financial support from FAPEMIG - NDF - Project PPE-00023-21 to carry out the aforementioned research of the NDF group.

Conflict of interests

The authors declare that they have no competing financial or non-financial interests relevant to this work.

Data availability

The study data will be available after publication in the journal, starting from the DOI.

Declaration of artificial intelligence use

The authors declare that no generative artificial intelligence tools were used to prepare this manuscript.

Handling editor

Editor handled this article: **Ibrahim Ademola Fetuga.**