

Article

Improving Fuel Consumption Prediction for Marine Diesel Engines Using Hierarchical Neural Networks and Pulsating Exhaust Models

Anibal Aguillon Salazar ^{1,2,*} , Georges Salameh ¹ , Pascal Chesse ¹ , Nicolas Bulot ² and Yoann Thevenoux ²

¹ LHEEA, CNRS, École Centrale Nantes, Nantes Université, UMR 6598, 44000 Nantes, France

² MAN Energy-Solutions France SAS, 44600 Saint-Nazaire, France

* Correspondence: anibalaguillons@gmail.com

Abstract: Predicting end-of-process variables in internal combustion engines, such as brake-specific fuel consumption or pollutant emissions, is crucial for engine design decisions. However, errors in common multi-layer-perceptron-based artificial neural network models often match the magnitude of the expected fuel consumption improvements, potentially leading to incorrect decisions. This study introduces a hybrid model where artificial neural networks replace engine block elements, while the 1D gas circuit and turbocharger models are retained. To enhance metamodel accuracy, two modifications are proposed: incorporating a pulsating mass flow rate in the exhaust line to capture pulsating effects missing in mean-value engine models and using a hierarchical arrangement of several multi-layer perceptrons instead of a parallel arrangement. The pulsating mass flow rate approach improves the accuracy of all tested configurations by replicating pulsating effects from a detailed 1D engine model. Meanwhile, the hierarchical arrangement refines predictions of end-of-process variables, such as fuel consumption, by increasing the total layers, with a minimal trade-off in the accuracy of other variables. These findings are validated using a metamodel derived from a calibrated 1D engine model in GT-Suite. The proposed methods are expected to enhance the accuracy of data-driven artificial neural network approaches, contributing to more reliable engine design optimization.

Keywords: diesel engines; neural networks; fuel consumption; metamodel; mean-value engine model; pulse unsteadiness



Academic Editors: Davide Di Battista, Fabio Fatigati and Marco Di Bartolomeo

Received: 29 November 2024

Revised: 19 December 2024

Accepted: 20 December 2024

Published: 24 December 2024

Citation: Salazar, A.A.; Salameh, G.; Chesse, P.; Bulot, N.; Thevenoux, Y. Improving Fuel Consumption Prediction for Marine Diesel Engines Using Hierarchical Neural Networks and Pulsating Exhaust Models. *Energies* **2025**, *18*, 17. <https://doi.org/10.3390/en18010017>

Copyright: © 2024 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

Reduced-Order Modeling (ROM) is a research field dedicated to simplifying complex systems while preserving essential features. This approach balances computational cost reduction with a trade-off in accuracy [1]. ROM techniques have been widely applied to 1D engine models to reduce computational costs [2,3]; this method allows a reduction in the number of state variables used as inputs [3], demonstrating the utility of ROM in reducing the complexity of fully characterized engine models when all explicit equations are available.

Artificial Neural Networks (ANNs) offer an alternative to ROM by creating surface-response metamodels, which establish relationships between the inputs and outputs of a physics-based model to reduce computational time. Unlike ROM methods, ANNs do not inherently retain a physics-based interpretation of the system. However, their independence from underlying equations makes them particularly valuable in scenarios where model details are inaccessible, such as when using proprietary commercial software.

ANNs in model reduction can be categorized into two approaches: data-driven modeling and metamodeling. In the data-driven approach, empirical data are used to train the ANN without requiring a physical model. A commonly used architecture for this purpose is the multi-layer perceptron (MLP), a black-box model that requires no prior knowledge of the system's governing equations. MLPs consist of fully connected layers of perceptrons relating input and output variables. Data-driven MLPs have been employed to predict various engine parameters, such as the specific fuel consumption, emissions, and exhaust temperature in internal combustion engines (ICEs) [4–8]. Additionally, they have been used to assess the potential benefits of alternative fuels, such as biodiesel [9–12] and hydrogen [13–20].

However, data-driven models are typically trained on limited datasets of 100 to 2000 data points [21], as experimental data are both costly and time-intensive. Consequently, the mean percentage error for predicted engine performance and fuel consumption often ranges from 5% to 8% [22]. These levels of error are comparable to the potential benefits of optimizing engine parameters and, as the system is modeled as a black box, the error distribution is unknown. When modeling the whole engine as a black-box to optimize the Atkinson cycle, a previous study identified the potential of reducing 8.5% fuel consumption in the metamodel; however, when evaluated in the real process this benefit is not obtained, and the engine even increased its fuel consumption by 3% [23], raising concern about the reliability of such predictions when compared to experimental results.

In contrast, metamodeling uses datasets generated by physics-based models to train ANNs. This approach allows for larger training datasets, enabling the use of more complex ANN architectures with reduced overfitting risks. Metamodeling is often employed to create lightweight models for optimization purposes [24]. For example, in one study focused on optimizing range-extender usage in electric hybrid vehicles, researchers used a dataset of 5000 points to train an ANN with an 11-30-30-1 architecture to predict engine torque. However, the study did not report detailed information about the accuracy of these predictions [25]. Another study used a high-neuron network with 500 hidden-layer neurons to create a metamodel from a 1D Diesel engine equipped with a dual-variable geometry turbocharger, achieving a prediction error of only 0.13% for fuel consumption compared to the detailed model; this high-level of accuracy is possible only when the whole system is modeled as a black-box. Even if the detailed model itself exhibited a 5% error relative to experimental measurements, the error distribution is known and could be addressed [26]. These studies relied initially on a detailed physics-based engine model, where the error distribution can be analyzed and then mimic the results with a high-accuracy metamodel.

The number of parameters required to model nonlinear phenomena grows with the number of input variables, a challenge referred to as the “curse of dimensionality”. Although adding inputs increases model complexity, it can provide additional insights if the added information outweighs the error introduced by increased model variance [27]. To address this, Hierarchical Neural Networks (HNNs) divide the system into subsystems, each with its own input parameters, reducing the overall model's susceptibility to dimensionality issues [28]. Each subsystem can represent a specific state variable, incorporating prior knowledge about the system's physical behavior [29]. This modular structure not only improves model accuracy but also enhances interpretability and scalability compared to monolithic black-box models [30–32].

HNNs have shown promise in reducing error of end-of-process variables in engine modeling. For example, in a data-driven model for a 600 hp Volvo engine, outputs from layers predicting pressure and temperature served as inputs for layers predicting emissions, significantly improving prediction accuracy compared to other fast-running models [33]. Hierarchical approaches are more used in chemical process modeling, where processes

are naturally divided into cascading steps. In these cases, ANNs are grouped into clusters representing state variables at each step, an approach termed “gray-box modeling” because it combines data-driven methods with engineering knowledge of the process [34]. Compared to black-box models like MLPs, HNNs perform better in multidimensional cascade processes by controlling inter-cluster connectivity based on system-specific knowledge, thus improving accuracy [34].

A previous study published by the researchers employed the detailed physics-based model to optimize the engine when an electric hybrid air-charging system is added to the system [35]. Due to the high computational cost of these optimizations, a model is proposed that allows solving similar optimizations with less computational resources. This study aims to develop a fast-running engine model capable of analyzing engine responses to changes in the air-charging system for optimization purposes. To achieve this, the computational cost of a 1D Diesel engine model built in the commercial software GT-Power is reduced by combining 1D elements (air and gas circuits and turbochargers) with a surface-response model. All engine block components are replaced by ANNs trained using the detailed model. The engine selected for this study is the same engine from a previous study by the team [35]; this engine is a 4 V-PA6B large four-stroke marine propulsion engine producing up to 405 kilowatts per cylinder at 1050 revolutions per minute.

The first novelty of this study is the introduction of HNNs into engine metamodeling. Two arrangements are compared: (1) a classical feed-forward MLP where all inputs are processed by individual networks and (2) a hierarchical MLP arrangement where outputs from certain networks serve as inputs for subsequent, more complex networks. The second novelty is the introduction of pulse unsteadiness into a mean-value engine model by imposing the pulsating mass flow rate of exhaust gases using a simple Gaussian function. The effect of the Gaussian pulse is analyzed in both arrangements studied.

All these novelties are introduced with the objective of improving the accuracy for predicting engine fuel consumption when there are different changes applied in the engine and the air-charging system.

2. Materials and Methods

2.1. Experimental Setup

The engine modeled in this study is a four-cylinder PA6B, four-stroke compression-ignition marine engine developed by MAN Energy Solutions France SAS (Saint-Nazaire, France). The cylinders are arranged in a V-configuration at a 60° angle. The main engine characteristics are summarized in Table 1. This engine features a fixed-timing pushrod camshaft with four valves per cylinder. Fuel injection is performed via individual jerk-type fuel pumps, where each pump mechanically pressurizes the fuel through a camshaft–piston mechanism. The start of injection (SOI) is electronically controlled during engine operation.

The experimental setup is illustrated in Figure 1. The engine’s flywheel is coupled to an AVL-Zöllner™ hydraulic dynamometer (64625, Bensheim, Germany), allowing precise control over engine speed and torque. All temperature and pressure data are acquired using an Allen-Bradley™ system. Engine functioning parameters, such as engine speed, volumetric flow of fuel, turbocharger speeds, cooling water conditions, and intake–exhaust conditions, are measured by the engine ECU. Measurements of pressure and temperature of intake–exhaust conditions are redundant between ECU and the Allen-Bradley™ (Rockwell Automation, Milwaukee, WI, USA) system. Table 2 instruments correspond to those used in the study connected to the Allen-Bradley™ system. Additional high-frequency measurements, such as injection fuel pump pressure and in-cylinder pressures, are collected by an internal data acquisition tool. All these data are transferred to the internal data processing unit of the MAN Energy Solutions, called CoCos EDS™. Cooling water and oil

are supplied by external pumps, though the oil circuit is not included in the diagram shown in Figure 1. All instrumentation and data acquisition procedures comply with the ISO 8178 standard [37], with specific details provided in Table 2. The testbench in its practical form can be appreciated in Figure 2.

Table 1. Main characteristics of the 4 V PA6B MAN engine [36].

Variable	Value
No. of cylinders	4 in V (60°)
Bore	280 mm
Stroke	330 mm
Nominal speed	1050 RPM
Nominal power	405 kW/cyl
Nominal torque	14,733 Nm at 1050 RPM
Air-charging system	2-parallel free-floating turbochargers
Start of injection	13° before TDCf
Intake valve opening (IVO)	57° before TDC
Intake valve closing (IVC)	81° after BDC
Exhaust valve opening (EVO)	85° before BDC
Exhaust valve closing (EVC)	79° after TDC

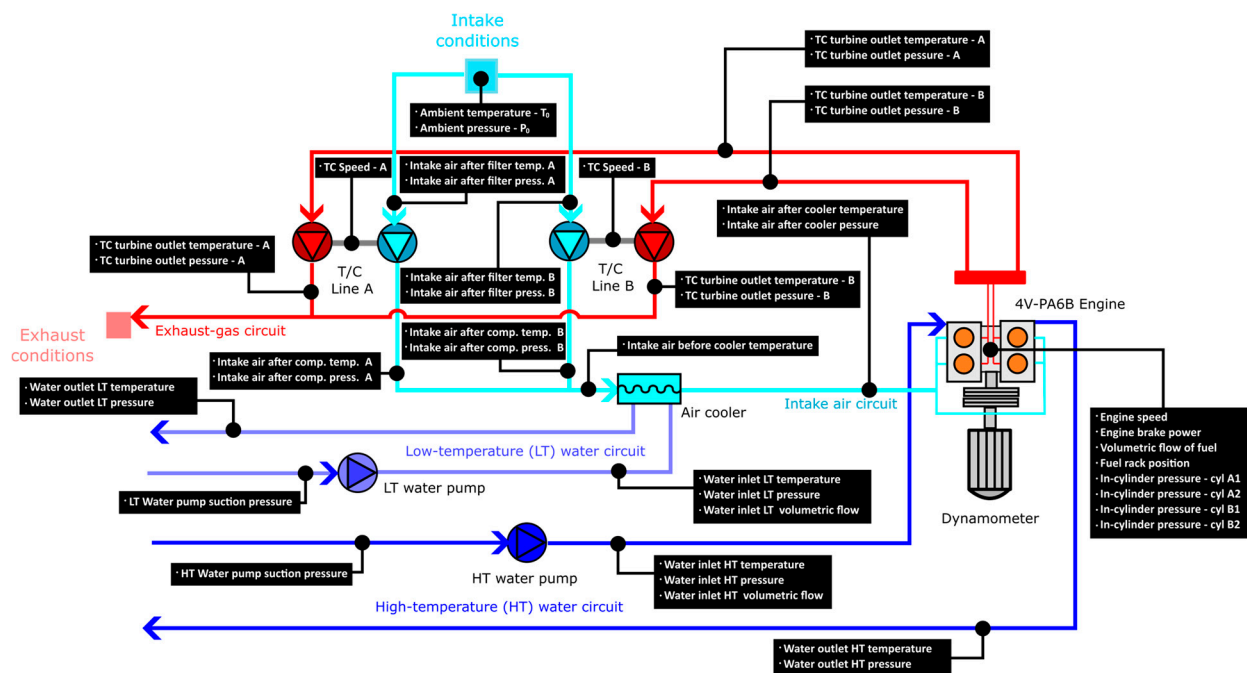


Figure 1. Testbench for the 4 V-PA6B MAN-ES engine.

Table 2. Data acquisition instruments.

Location	Unit	Instrument
Ambient, temperature	[°C]	Meteorological station Manumasure™ 1
Ambient, pressure	[mbar]	Kimo™ CP110
Intake air after filter, temperature (A and B lines)	[°C]	PT100 Sensor
Intake air after filter, pressure (A and B lines)	[kPa]	Kistler™ Type 4011A (0–10 bar)
Intake air after comp., temperature (A and B lines)	[°C]	PT1000 Sensor
Intake air after comp., pressure (A and B lines)	[kPa]	Kistler™ Type 4011A (0–10 bar)
Intake air before cooler, temperature	[°C]	PT1000 Sensor
Intake air after cooler, temperature	[°C]	PT100 Sensor
Intake air after cooler, pressure	[kPa]	Kistler™ Type 4011A (0–10 bar)

Table 2. Cont.

Location	Unit	Instrument
Water pump suction pressure (LT and WT)	[kPa]	Danfoss™ MBS 5150 (0–6 bar)
Water inlet temperature (LT and WT)	[°C]	PT100 Sensor
Water inlet pressure (LT and WT)	[kPa]	Danfoss™ MBS 5150 (0–10 bar)
Water inlet volumetric flow (LT and WT)	[m ³ /h]	Endress-Hausser™ Promag P100
Water outlet temperature (LT and WT)	[°C]	PT100 Sensor
Water outlet pressure (LT and WT)	[kPa]	Danfoss™ MBS 5150 (0–10 bar)
TC turbine inlet temperature (A and B lines)	[°C]	K Type Thermocouple
TC turbine inlet pressure (A and B lines)	[kPa]	Kistler™ Type 4011A (0–10 bar)
TC turbine outlet temperature (A and B lines)	[°C]	K Type Thermocouple
TC turbine outlet pressure (A and B lines)	[mbar]	Kistler™ Type 4011A (0–10 bar)
Turbocharger speed (A and B lines)	[RPM]	Hall-effect sensor
Engine speed	[RPM]	Baumer™ CH-8501/Hall-effect sensor
In-cylinder pressure (cyl.: A1, A2, B1, and B2)	[bar]	Kistler™ 6061C (0–250 bar)
In-cylinder temperature (cyl.: A1, A2, B1, and B2)	[°C]	K Type Thermocouple

1 Manumasure is a company of metrology instruments from the Chauvin Arnoux Group. Located at Asnières-sur-Seine, 92600, France

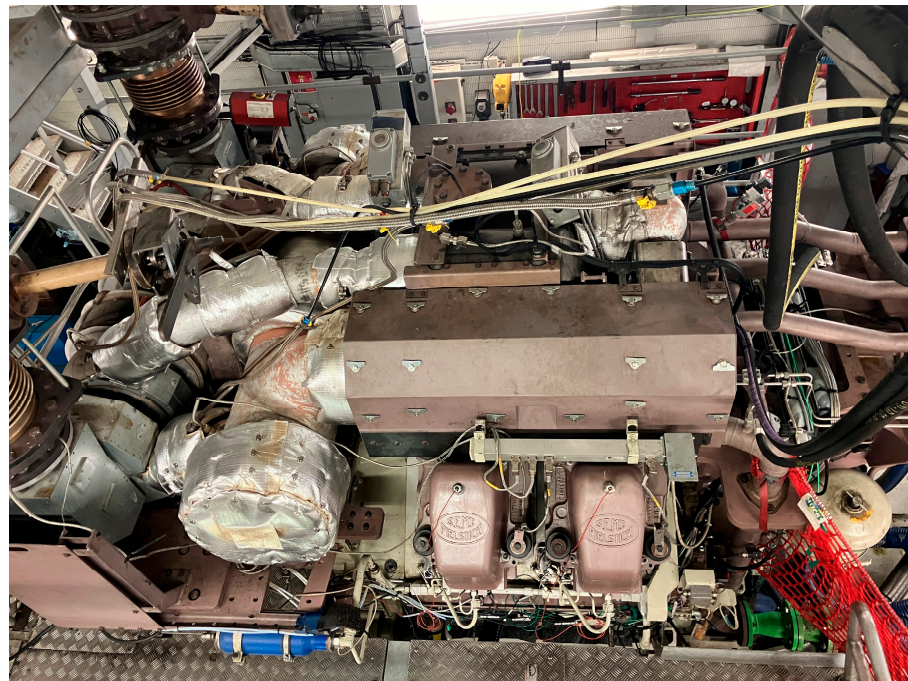


Figure 2. Practical photo of the 4 V PA6B MAN-ES engine in the testbench.

2.2. Experiment Protocol

This study focuses on marine propulsion applications, and the engine tests are conducted following the E3 cycle described in ISO 8178-4 [37]. In this cycle, the engine's brake power is related to its speed by a cubic relationship known as the light-propeller law, represented by Equation (1):

$$P_{brake} [\text{kW}] = k * N^3 \quad (1)$$

Here, P_{brake} represents the brake power, N is the engine speed (in revolutions per minute), and k is a proportional constant. In this study, k is set to 1.39942×10^6 , assuming calm water conditions to represent light propeller loads. The resulting relationship is illustrated in Figure 3.

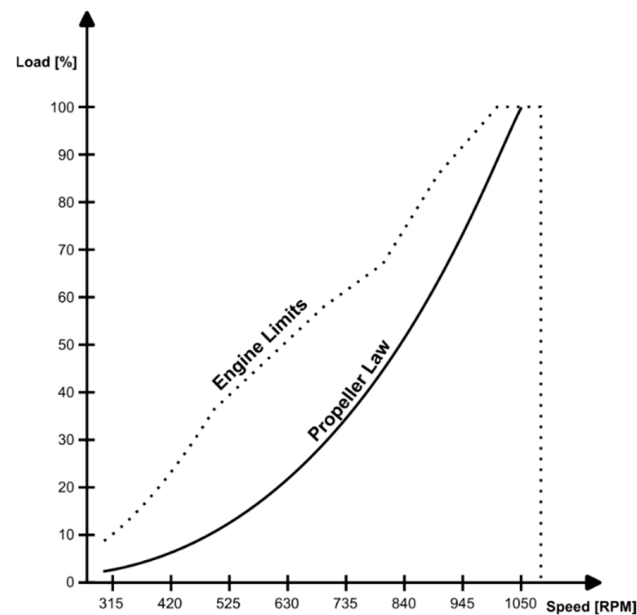


Figure 3. Propeller law for the 4 V-PA6B engine.

In Figure 3, the engine's operational limits are indicated, as defined by MAN Energy Solutions. These limits are based on mechanical constraints such as maximum in-cylinder pressure, oil film thickness, and turbocharger surge, among others. Tests conducted along the propeller law involve varying engine loads (as a percentage of nominal power), with the associated engine speeds determined by Equation (1).

2.3. 1D Detailed Engine Model

The engine was modeled using GT-Suite™, a commercial software developed by Gamma Technologies® (Illinois, United States). The diagram of Figure 4 represents the 1D model, highlighting the frontier between the intake and exhaust line with the engine block elements. The engine in the testbench operates with two turbochargers arranged in parallel, where the second turbocharger is activated only when engine loads exceed 50%. This sequential operation is not considered for the rest of the study to maintain a low complexity. In further sections, both turbochargers will be functioning for all engine operation points.

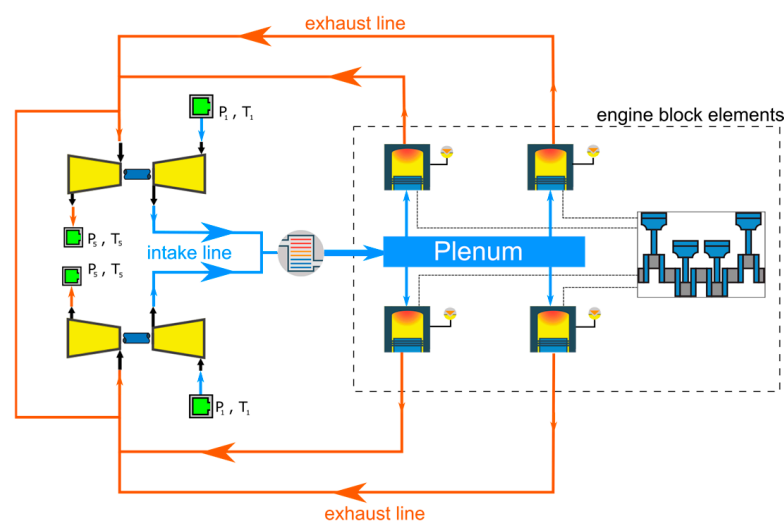


Figure 4. Diagram of the 1D engine model. Exhaust and intake lines modeled as 1D elements.

The combustion process was simulated using the DI-Pulse model [38], which is a predictive approach capable of accounting for variations in combustion rates due to changes in the thermodynamic conditions, such as modifications in the air-charging circuit [39]. Calibration of the combustion model was performed using experimental data from a 12-cylinder V-configuration PA6B engine, with details provided in a previous publication by the research team [36]. This earlier work also describes the sub-models implemented, including the Woschni model for heat transfer at the cylinder walls, a linear variation model for friction based on piston speed, and the Zeldovich model for emissions prediction. For the present study, additional sub-models, such as turbocharger SAE maps, heat transfer in the exhaust and intake line, and pressure losses, were calibrated specifically for the 4-cylinder PA6B engine, following well-established marine engine modeling practices in GT-Suite™ [40]. The procedure involves adding multipliers to heat transfer and pressure losses in the exhaust and intake line and finding the optimal value that minimizes the error among 14 operating points. Similarly, for the turbochargers' maps there are added multipliers of mass and pressure ratio. This whole process was produced according to the recommended procedure by Gamma-Technologies [41].

The accuracy of the model was validated by comparing the simulation results with experimental data for key parameters at several engine operating points along the propeller law. Validation results are presented in Figures 5–8. Figure 5 compares the measured and predicted temperatures at three different locations within the engine system. Figure 6 provides a comparison of the measured and predicted pressures, specifically after the compressor and before the turbine. Similarly, Figure 7 shows the agreement between the measured and predicted turbocharger speeds.

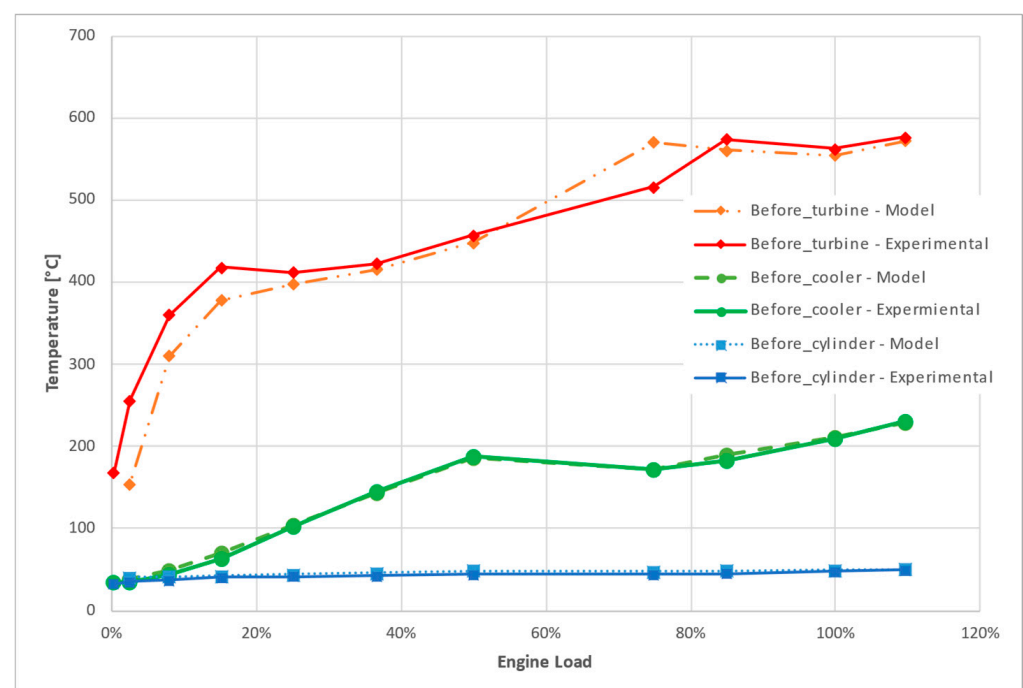


Figure 5. Temperature comparison between measurements and model results along the propeller law (kN^3).

The validation results demonstrate that the model achieves maximum errors in acceptable ranges. For turbocharger speed, the largest discrepancy of 9.7% occurred at low engine loads, where turbocharger speed is low. In terms of pressure predictions, the maximum deviation in boost pressure was 3.5%, while the error in turbine inlet pressure reached 3.4%. Temperature predictions were also highly accurate, with a maximum error of 6.9%

observed at the turbine inlet. Intake line temperatures, both before and after the intercooler, were predicted with maximum errors of 2% and 1.4%, respectively.

Finally, the evolution of error in brake-specific fuel consumption (BSFC) across different operating conditions is shown in Figure 8. The error distribution is a constant overestimation of BSFC of around 10%.

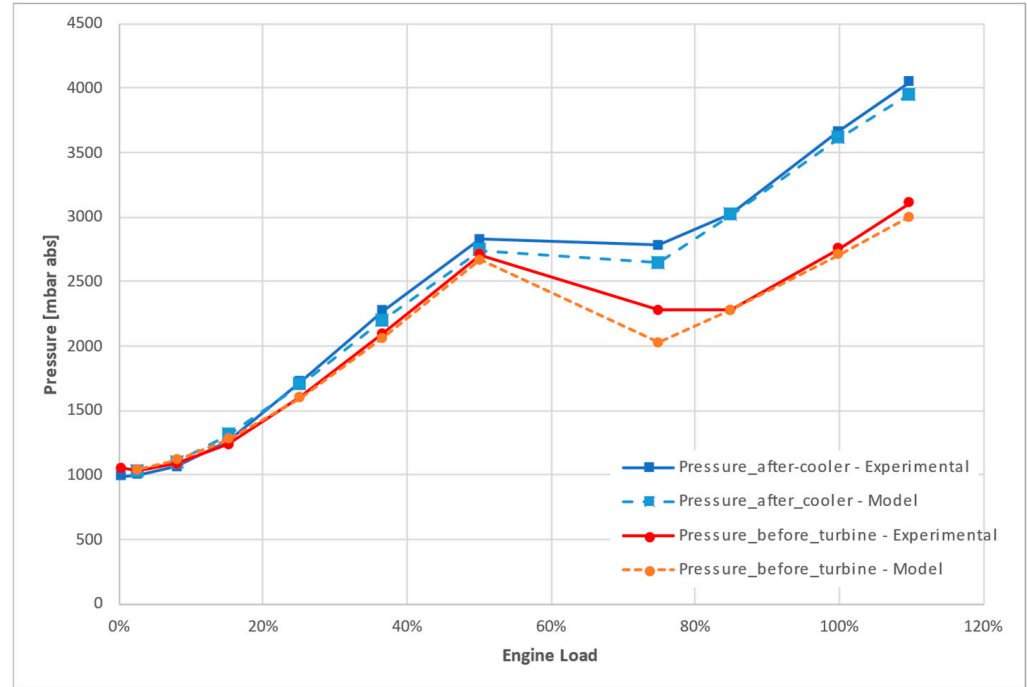


Figure 6. Pressure comparison between measurements and model results along the propeller law (kN^3).

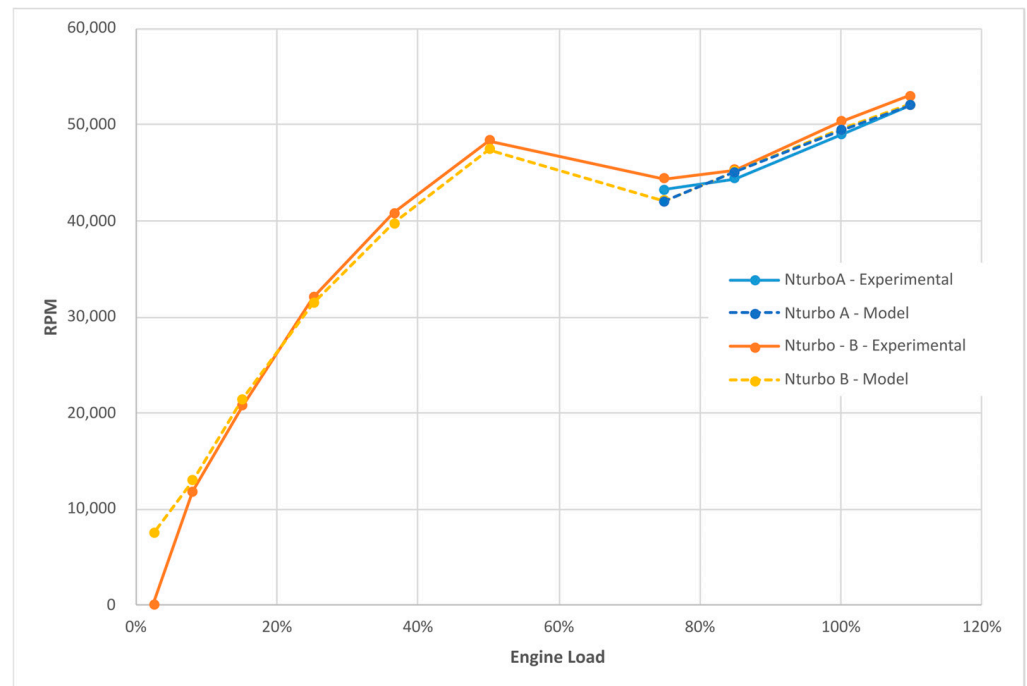


Figure 7. Turbocharger speed comparison between measurements and model results along the propeller law (kN^3). Sequential operation deactivates turbocharger-A below 50%.

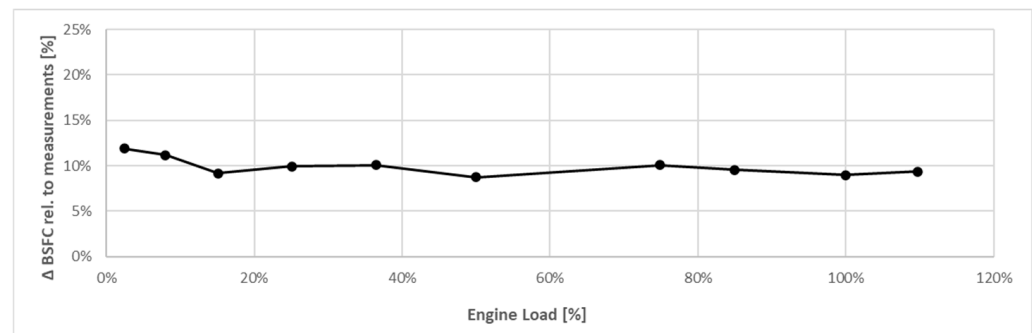


Figure 8. Relative error of brake-specific fuel consumption between the model and measurements of the 4 V engine along the propeller law (kN^3).

2.4. Engine Metamodel Layout Description

The metamodel of the engine is implemented inside the GT-Suite software (version 2023) using purely built-in tools; this ensures portability of the model without requiring third-party software and licenses. The surface-response model in this study was implemented to replicate the behavior of the engine block elements using a set of defined inputs. The metamodel replaces the cylinders, injectors, valves, crankshaft, and intake–exhaust runners, while retaining the one-dimensional (1D) elements that form the air and exhaust gas circuits, as seen in Figure 9. This modification transforms the originally closed gas circuit into two separate circuits: the air intake and exhaust gas circuits. These circuits are mathematically linked through the surface-response model, which computes the mass flow rate of air, the mass flow rate of exhaust gases, and the exhaust gas temperature necessary to complete the system. In addition to these essential outputs, the metamodel generates two critical variables: brake-specific fuel consumption (BSFC), used for optimization purposes, and the maximum in-cylinder pressure, which serves as a constraint. Two different metamodel arrangements are explored for computing these variables, as discussed in subsequent sections.

The retained 1D gas circuits are critical for this study as the model is intended to optimize engine performance under varying configurations of the air-charging circuit and turbomachinery. Inputs to the surface-response model consist of constant values and time-averaged variables, with averaging performed over two engine cycles. Consequently, the metamodel outputs are mean-value outputs.

A challenge in mean-value engine models arises from the pulsating nature of exhaust gas flow. When the exhaust valve opens, high-pressure hot gases are expelled rapidly due to the pressure differential between the cylinder and the exhaust manifold. This mass flow creates a pressure pulse influenced by manifold dynamics, turbine restrictions, and exhaust line acoustics [42]. Accounting for these pulsations is essential to maintain the metamodel's predictive accuracy. Previous studies have used correction factors on the turbine side to approximate the effects of pulses, reducing errors in mean-value models from 12% to 4% [43]. However, these factors require calibration and lack predictive capability. An alternative approach imposes pulsating unsteadiness directly on the exhaust line pressure, though the pressure wave shape is known to depend heavily on the geometry and characteristics of the exhaust line [42], making it invalid for configurations different from the calibrated setup.

The flow solver of GT-Suite can predict the pulses and the effects of acoustics in the line with high precision. To utilize these capabilities, the mass flow rate of exhaust gases is modeled as a pulse that is designed to mimic the behavior of the detailed model. GT-Suite's flow solver then resolves the resulting pressure pulse. Observations from a comparison between single-cylinder and multi-cylinder models revealed that while the mass flow rate pulse shape shows minor variations, the pressure waves change significantly due to

acoustic interactions, as illustrated in Figure 10. This finding motivated the use of a mass flow rate pulse rather than pressure to model pulsations.

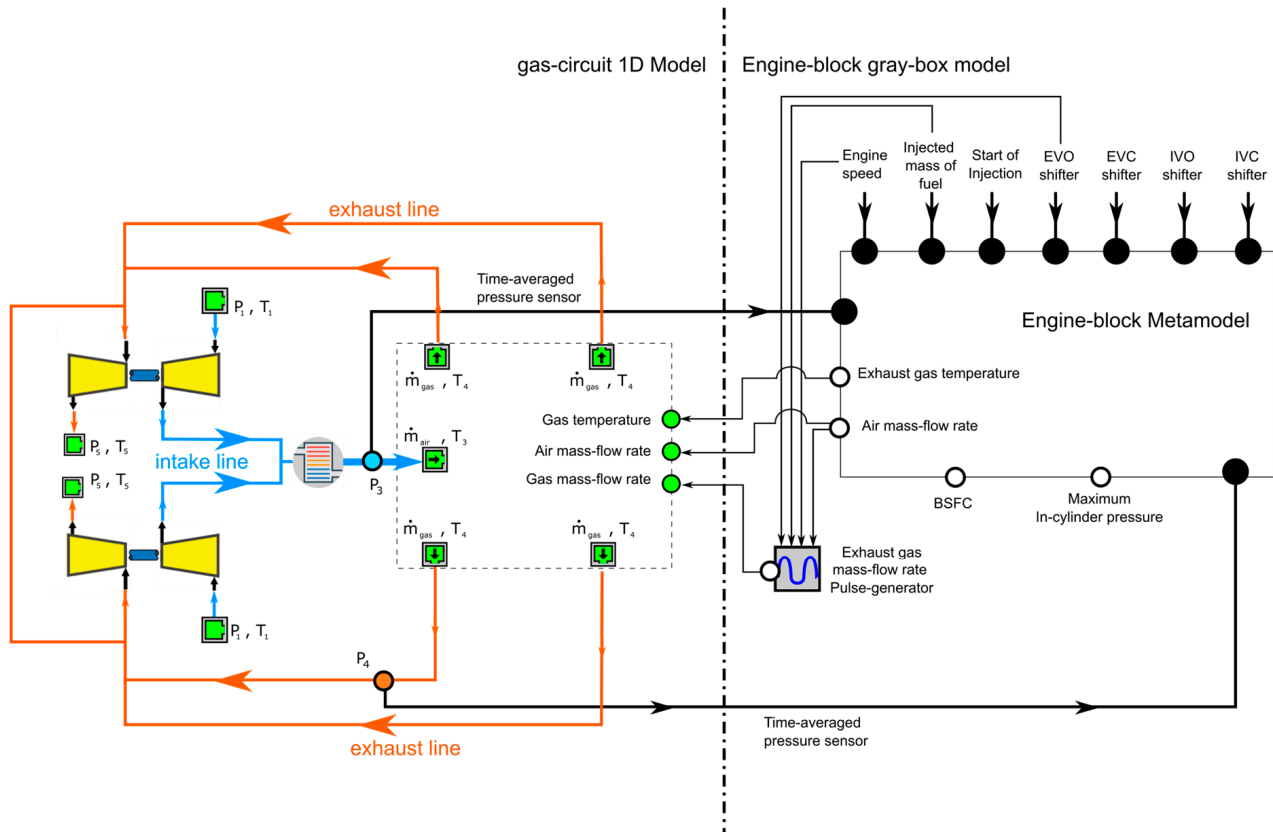


Figure 9. Diagram of the engine model using the surface-response metamodel to replace the engine block elements from Figure 4.

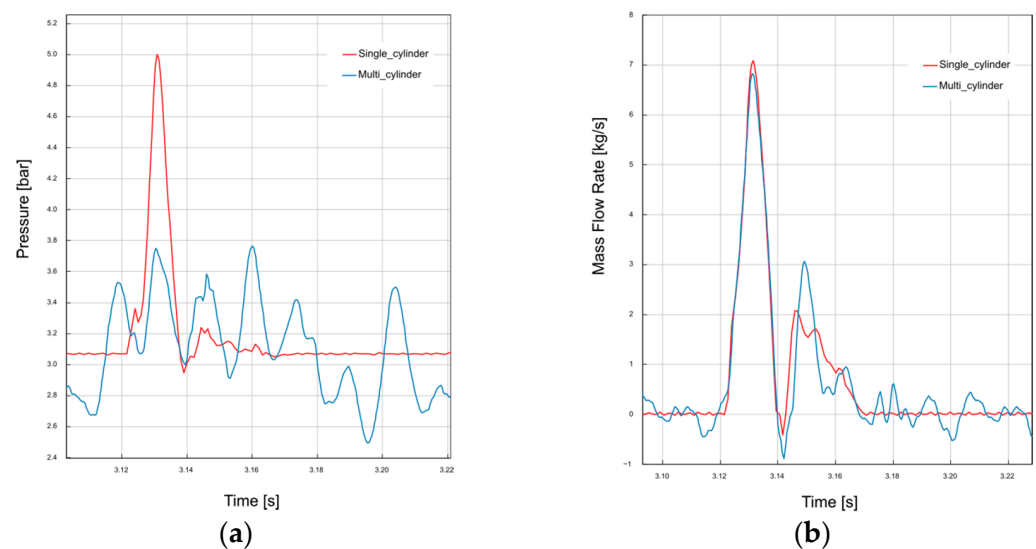


Figure 10. Comparison of the pulsating operation in the exhaust-runner of a single-cylinder model and in a multi-cylinder model. (a) Pressure waves change due to the acoustics. (b) Mass flow rate. Similar pulses can be identified in both cases. (100% Load–1050 RPM).

The pulse is represented using a double-Gaussian function to match the detailed model's mass flow rate. Several requirements guided the development of this approach. First, the time integration of the pulse must equal the mean value predicted by the meta-

model. Second, the distribution between the two Gaussian peaks was calibrated using the detailed pulse as reference, resulting in a relationship between pulses of 70% and 30% of the total mass flow rate. Third, the pulse's timing must account for variations in exhaust valve opening. Since the metamodel does not include a crankshaft, simulated time is converted into crank-angle degrees using Equation (2). The total mass flow rate as a function of crank angle is expressed as the sum of the two pulses, as shown in Equation (3).

$$\theta [\text{deg}] = \left(\frac{t [\text{s}]}{(120/\text{RPM})} - \left\| \frac{t [\text{s}]}{(120/\text{RPM})} \right\| \right) * 720 \quad (2)$$

$$\dot{m}_{gas}(\theta) = f_1(\theta) \cdot \left[\frac{\dot{m}_{gas, MV} \cdot PR_1}{\left(\frac{\int_{k=0}^{720} f_1(k) \cdot dk}{720} \right)} \right] + f_2(\theta) \cdot \left[\frac{\dot{m}_{gas, MV} \cdot (1 - PR_1)}{\left(\frac{\int_{k=0}^{720} f_2(k) \cdot dk}{720} \right)} \right] \quad (3)$$

In Equation (3), the defined integrals are independent of the current crank-angle degree, allowing them to be treated as constants. These constants are used to scale the total integrated mass flow rate of the pulse to the mean-value mass flow rate from the metamodel. Inside the metamodel, the mass flow rate of exhaust gases is calculated using the total air mass flow rate and the injected mass per cycle per cylinder, as expressed in Equation (4). The Gaussian pulse functions are expressed in Equation (5),

$$\dot{m}_{gas, MV} = \frac{\dot{m}_{fuel} [\text{mg/cycle}] \cdot \text{RPM}}{120 \cdot 1000 \cdot 1000} + \frac{\dot{m}_{air}}{N_{cyl}} \quad (4)$$

$$f_i(\theta) = e^{\left(-\frac{(\theta - (a_i + EVO))^2}{2 \cdot b_i^2} \right)} \quad (5)$$

In Equation (5), the parameter a_i represents the timing of the pulse start of each cylinder in crank-angle degrees (this value differs from the moment of exhaust valve opening due to delays between the valve and the runners), its value has been calibrated using the detailed model pulse of mass at the end of the exhaust runner. As different valve timings are explored, the pulse is adapted using an exhaust valve opening shifter timing (EVO). In same equation, b_i represents the pulse duration. The pulse start-timing and its duration are calibrated to match the detailed model and the values used are provided in Table 3.

Table 3. Factors for the double-Gaussian mass flow rate pulse.

	Cylinder A1	Cylinder A2	Cylinder B1	Cylinder B2
a_1	295	475	715	175
b_1	25	25	25	25
a_2	405	585	105	285
b_2	25	25	25	25

Figures 11 and 12 illustrate the validation of this approach. Figure 11 compares the instantaneous mass flow rate at the exhaust runner between the detailed model with a single-Gaussian pulse and a double-Gaussian pulse generator, showing that a double pulse better represents the detailed model. Figure 12 shows the resulting pressure waves obtained using the detailed model and the double-Gaussian pulse generator coupled with GT-Suite's 1D flow solver. The double-Gaussian pulse effectively reproduces the detailed model's pressure wave shapes, confirming its suitability for maintaining the metamodel's predictive capability.

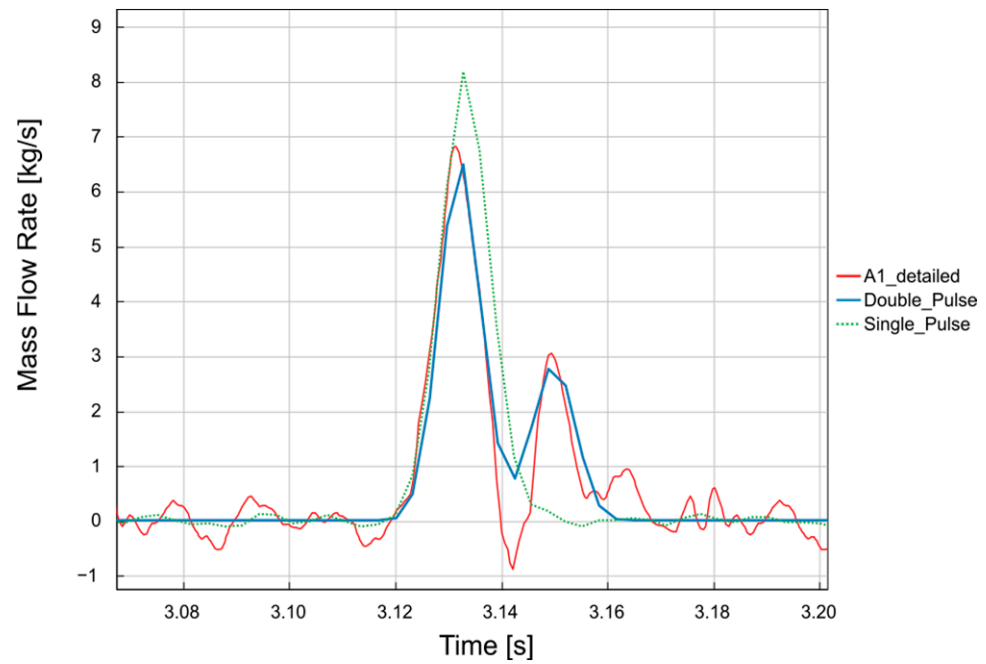


Figure 11. Instantaneous mass flow rate observed in the exhaust-runner of cylinder A1 in different engine models. Difference between using a simple pulse and a double pulse compared with a detailed model. (100% Load–1050 RPM).

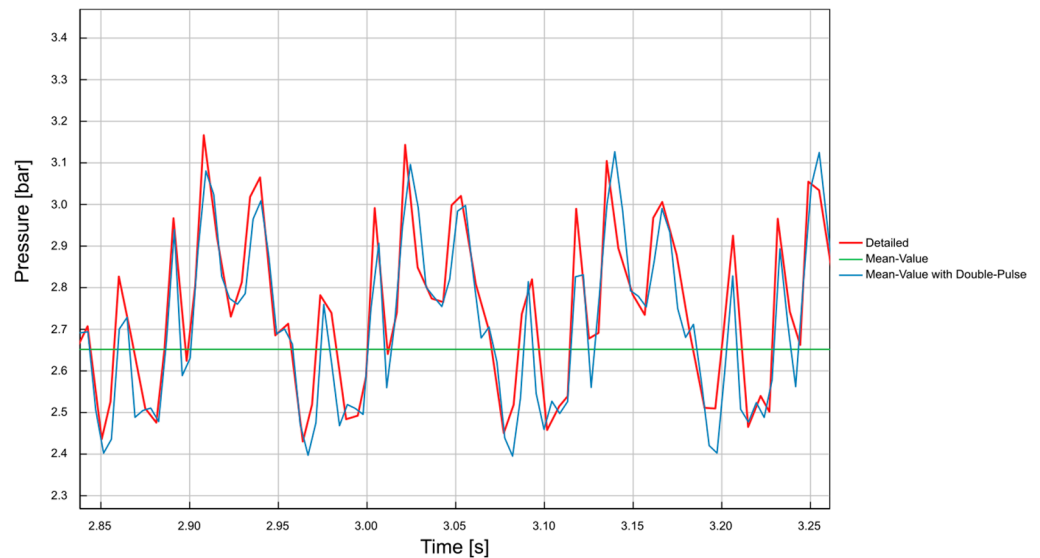


Figure 12. Pressure waves observed at turbine inlet. Comparison between a detailed model, mean-value, and mean-value with double-gaussian pulses. (100% Load–1050 RPM).

2.5. Surface-Response Method

The surface-response methodology involves using a dataset derived from the real process to train a metamodel that approximates the system's behavior. In this study, the dataset was generated using the calibrated 1D physics-based model described in the previous section, which acts as the basis for the surface-response model (Figure 13). The total squared error of the surface-response metamodel relative to the real process is expressed in Equation (6):

$$E\{e^2\} = E\{(y_u - y_m)^2\} + E\{(n^2)\} + E\{(y_m - \hat{y})^2\} \quad (6)$$

$$(error)^2 = (prior\ error)^2 + noise\ variance + (metamodel\ error)^2$$

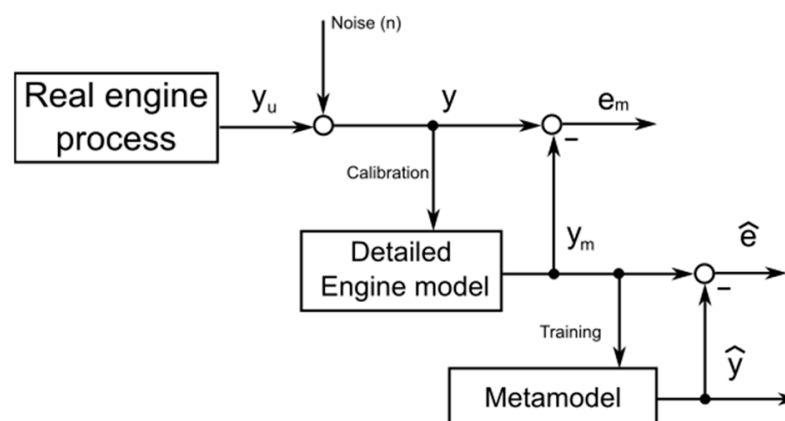


Figure 13. Diagram of the error calculation along the process of the creation of a metamodel.

This total error comprises multiple components, including the noise variance, which arises from the measurement process (instrument precision, procedure consistency, and potential human interaction) and the prior error, which reflects the deviation of the 1D model from the real process. The focus of this study is exclusively on minimizing the metamodel error, defined as the difference between the outputs of the 1D model and the surface-response model. The metamodel error can be expanded according to Equation (7),

$$E\{(y_m - \hat{y})^2\} = [y_m - E\{\hat{y}\}]^2 + E\{[\hat{y} - E\{\hat{y}\}]^2\} \quad (7)$$

$$(\text{metamodel error})^2 = (\text{bias error})^2 + \text{variance error}$$

The error of the detailed model with respect to the real engine process is discussed in Section 2.3. This error remains constant because the primary objective of this publication is not to refine the calibration of the 1D model but to propose a more accurate metamodel.

The total error of the metamodel is the sum of the detailed model error and the metamodel error relative to the detailed model. Since the detailed model error is fixed, minimizing the metamodel error relative to the detailed model effectively minimizes the total error of the metamodel.

The variance error occurs when a model's predictions deviate from the optimal values due to limitations in the available data. For instance, a cubic function approximated by a third-degree polynomial could achieve nearly zero variance error if sufficient data are provided to estimate the model parameters accurately. Conversely, when a simpler structure, such as a linear polynomial, is used to approximate a cubic function, a persistent discrepancy arises independently of the amount of data used. This difference is known as bias error. This type of error reflects the inherent limitations of an overly simplistic model structure [27].

For nonlinear systems, reducing bias error typically involves increasing the model's complexity, for example, by adding more neurons to a neural network or incorporating higher-order terms in a polynomial regression. However, this increase in complexity introduces a trade-off. More complex models require larger datasets to ensure parameter estimates are robust. With finite datasets, often constrained by computational power, adding complexity can inadvertently increase variance error, resulting in overfitting. Overfitting occurs when the model memorizes the training data instead of generalizing to unseen data, leading to poor performance in practical applications.

To balance bias and variance errors effectively, the surface-response models in this study were evaluated using both training and cross-validation datasets. Cross-validation with a separate dataset ensures the model generalizes well to new data by providing a

realistic measure of its predictive capability. Shown below are the detailed arrangements of both neuron networks tested in this study.

2.5.1. MLP-Metamodel

In this configuration, all the inputs to the metamodel are directly utilized as inputs to multiple individual multi-layer perceptron (MLP) networks (see Appendix A). Each MLP is responsible for predicting a specific output variable required to drive the 1D gas circuit, as illustrated in Figure 9. This approach follows a traditional black-box metamodeling strategy, where each output variable is modeled independently using a dedicated network. The architecture of these networks is based on a two-hidden-layer MLP structure with layer configurations of 11-20-30-1 neurons (input-hidden1-hidden2-output), as depicted in Figure 14. By isolating the modeling of individual outputs, this approach simplifies the training process but does not exploit potential interdependencies between the output variables, which is characteristic of the black-box approach.

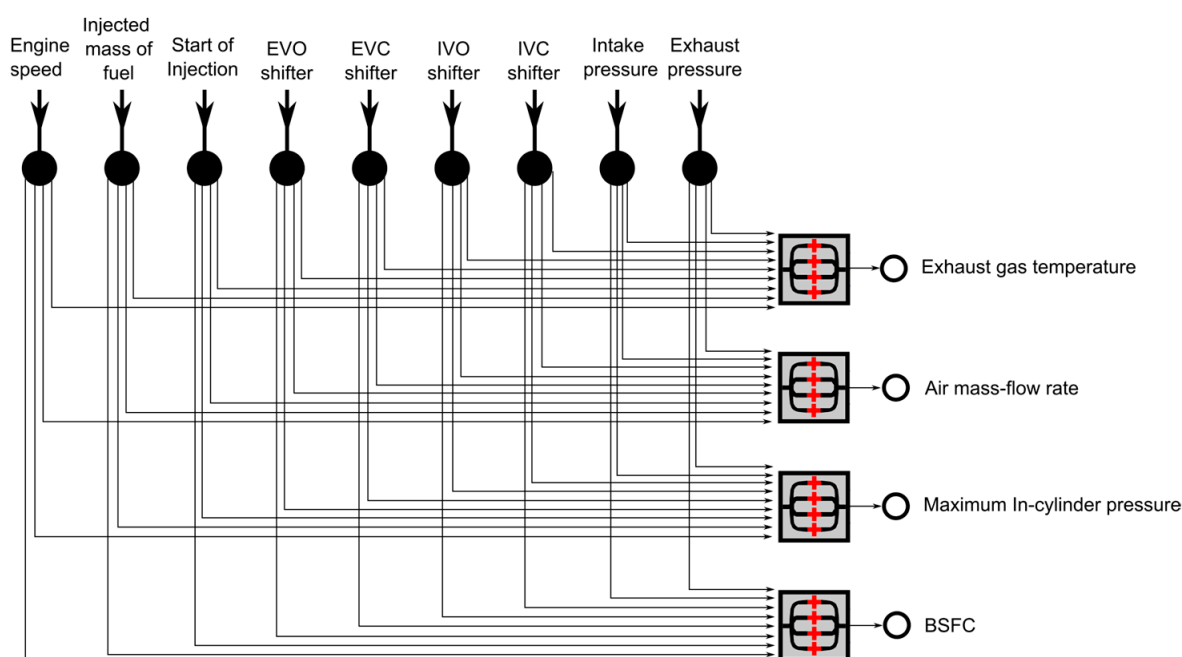


Figure 14. MLP metamodel of the elements from the engine block section.

2.5.2. Hierarchical Artificial Neuron Networks (HNNs) Arrangement Metamodel

The HNN metamodel represents a more sophisticated structure that integrates physics-based relationships with hierarchical artificial neural networks. Unlike the purely MLP-based approach, this gray-box metamodel arranges the available inputs to generate intermediate variables, which are subsequently used to calculate more complex output variables. This hierarchical arrangement provides an additional layer of interpretability and can potentially enhance prediction accuracy by incorporating domain knowledge into the modeling process.

The structure of the HNN arrangement is shown in Figure 15. The air mass flow rate is obtained using the physical relationship between the air–fuel ratio (AFR) trapped in the cylinder and the trapping ratio of air compared with the total engine air mass flow rate, as defined by Equation (8),

$$\dot{m}_{air} = \left(\frac{N_{cyl} \cdot \dot{m}_{fuel} [\text{mg/cycle}] \cdot \text{RPM}}{120 \cdot 1000 \cdot 1000} \right) * \frac{AFR_{trapped}}{trap_ratio} \quad (8)$$

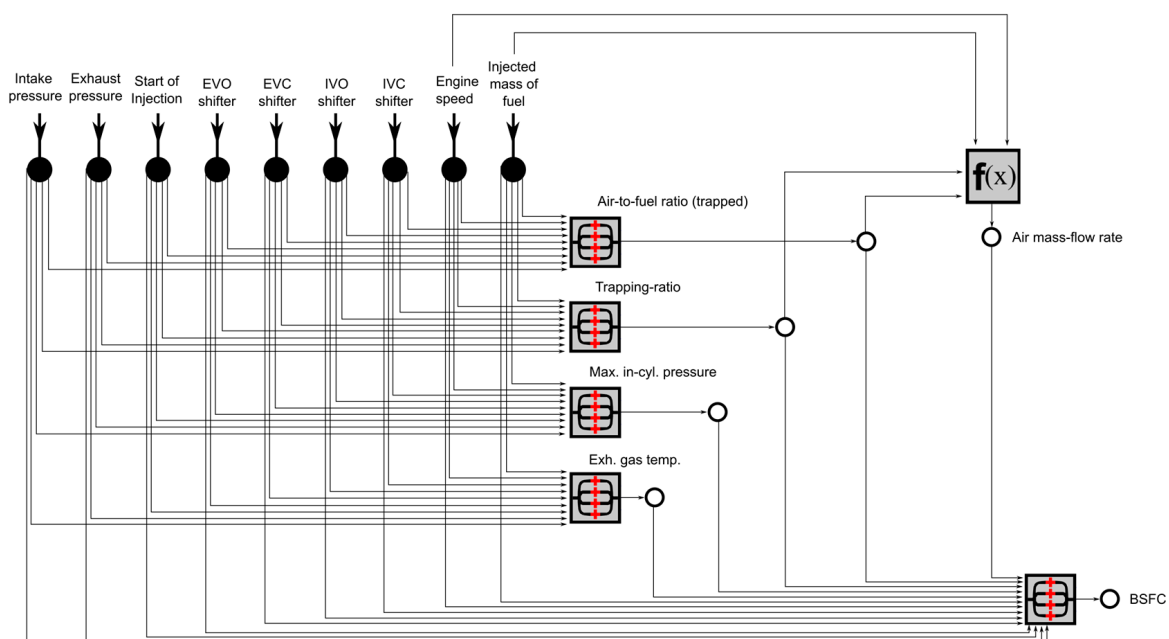


Figure 15. Hierarchical neuron network (HNN) metamodel of the engine block elements.

The air–fuel ratio and trapping ratio are obtained by individual MLPs, while the injected mass of fuel and engine speed are obtained from the model inputs. The BSFC represents an end-of-process variable since it relates the total consumed mass of fuel (grams) to the total power produced (kilowatts) in a period (hour). Since all available variables influence engine power, all of them are used to predict the BSFC, as indicated in Figure 15.

3. Results

3.1. Training Results

To develop the metamodel, the engine block elements from the detailed 1D model, shown in Figure 4, were isolated into a separate model. In this isolated model, intake and exhaust conditions—such as pressure and temperature at the cylinder head runners—were imposed using boundary elements. A design-of-experiments (DOE) approach was applied to distribute 20,000 operating points across the design space defined in Table 4.

Table 4. Range of variables explored with 20,000 points distributed with a Latin-Hypercube DOE.

Variable	Minimum	Maximum	Variable Type
Engine speed [RPM]	300	1050	Decimal
Engine power [kW]	25	1620	Decimal
SOI [CA before TDC]	−28	−12	Integer
Boost pressure P3 [bar]	1	10	Decimal
Backpressure P4 [bar]	1	10	Decimal
IVO Shifter [CA]	−35	35	Integer
IVC Shifter [CA]	−35	35	Integer
EVO Shifter [CA]	−35	35	Integer
EVC Shifter [CA]	−35	35	Integer

From these 20,000 operating points, the conditions leading to combustion efficiencies below 50% or brake-specific fuel consumption (BSFC) exceeding 600 g/kWh were excluded. This filtering process yielded a dataset of 6140 valid operating points. Of this dataset, 70% was used for training the multi-layer perceptron (MLP) networks, while the remaining 30% was reserved for validation to assess overfitting.

Training involved optimizing the network's weights and biases to minimize a linear combination of the sum of squared errors and the sum of squared weights. The Levenberg–Marquardt optimization method was employed for this purpose, ensuring robust convergence. Details of the network architecture and parameter optimization are provided in the Appendix A.

In Table 5, there are shown the root-mean squared errors obtained during the training and validation process of the networks required for the MLP arrangement. The largest difference between training and validation is observed in exhaust temperature prediction, with a validation error of 4.85 K. However, the exhaust temperatures range between ~600 K and 950 K, an error of 4.85 K at the lowest temperature of 600 K represents a relative error of 0.85%.

Table 5. Root-mean-squared-error obtained during training and validation of the networks used in the MLP architecture.

Variable Output	Training	Validation
Air mass flow rate [kg/s]	0.006831	0.007645
Exhaust gas temperature T4 [K]	3.4854	4.85883
Maximum in-cylinder pressure [bar]	0.661399	0.717855
BSFC [g/kWh]	5.26789	6.01682

Table 6 shows the root-mean squared errors observed during the training process of the networks required for the HNN arrangement. Trapping ratio and air-to-fuel ratio are new non-dimensional variables not trained in the MLP arrangement. Exhaust temperature and in-cylinder pressure are the same networks used in the previous arrangement, as they use the same input variables. Brake-specific fuel consumption is obtained differently with the HNN arrangement, reducing the validation error by half compared with the MLP arrangement.

Table 6. Root-mean-squared-error obtained during training and validation of the networks used in the HNN architecture.

Variable Output	Training	Validation
Trapping ratio	0.000603	0.000866
Air-to-fuel ratio (trapped)	0.060616	0.098622
Exhaust gas temperature T4 [K]	3.4854	4.85883
Maximum in-cylinder pressure [bar]	0.661399	0.717855
BSFC [g/kWh]	1.00038	3.36016

More advanced training methods can optimize the training duration with early stops to minimize the difference between training and validation errors. However, these methods are not available in the built-in tools of the software, and their implementation requires third-party software and licenses.

3.2. Cross-Validation Results and Comparison of Metamodels

Cross-validation was conducted using a new dataset independent of the training data distribution. This process evaluated the metamodels within a 1D gas-circuit model setup, as illustrated in Figure 9. A separate DOE generated 445 operating points across the design space defined in Table 7. To simulate variations in turbocharger size, scaling factors were introduced to modify the SAE turbine and compressor maps, specifically in mass flow rate and pressure ratio.

Table 7. Range of variables explored with 445 points distributed with a Latin-Hypercube DOE.

Variable	Minimum	Maximum	Variable Type
Engine speed [RPM]	300	1050	Decimal
Engine power [kW]	25	1620	Decimal
SOI [CA before TDC]	−28	−12	Integer
IVO Shifter [CA]	−35	35	Integer
IVC Shifter [CA]	−35	35	Integer
EVO Shifter [CA]	−35	35	Integer
EVC Shifter [CA]	−35	35	Integer
Pressure ratio multiplier (compressor)	0.75	1.25	Decimal
Mass flow multiplier (compressor)	0.75	1.25	Decimal
Pressure ratio multiplier (turbine)	0.75	1.25	Decimal
Mass flow multiplier (turbine)	0.75	1.25	Decimal

The same set of 445 operating points was evaluated with the detailed model and the four metamodels: MLP with and without pulses and HLP with and without pulses. The performance of each metamodel was benchmarked against the detailed model using the coefficient of determination (R^2) defined by Equation (9) and the root mean square error (RMSE) defined by Equation (10).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_{m,i} - \hat{y}_i)^2}{\sum_{i=1}^n (y_{m,i} - \bar{y}_m)^2} \quad (9)$$

$$RSME = \sqrt{\frac{1}{n} \sum_{i=1}^n |y_{m,i} - \hat{y}_i|^2} \quad (10)$$

Figure 16 compares the predictions of the detailed model and the metamodels for key outputs, including the air mass flow rate, turbocharger power and speed, and BSFC. Incorporating the gas pulse generator significantly improved the performance of both the MLP and HNN metamodels. As shown in Tables 8 and 9, the RMSE decreased by 16.8% for the MLP arrangement and by 12.7% for the HNN arrangement when pulses were included.

Table 8. Coefficient of determination (R^2) obtained during cross-over validation by evaluating in the complete engine model. Comparison between metamodel methods.

Variable Output	MLP with Pulses	HNN with Pulses	MLP w/o Pulses	HNN w/o Pulses
Injected mass per cycle	0.9930	0.9974	0.9876	0.9944
Average of maximum cylinder pressures	0.9956	0.9954	0.9906	0.9889
Boost pressure P3	0.9949	0.9955	0.9894	0.9877
Backpressure P4	0.9939	0.9921	0.9843	0.9787
Turbocharger speed—line B	0.9947	0.9931	0.9870	0.9829
Turbine power gas—line B	0.9945	0.9949	0.9887	0.9843
Compressor power—line B	0.9946	0.9950	0.9887	0.9843
Turbine efficiency—line B	0.9914	0.9752	0.8879	0.9012
Compressor efficiency—line B	0.9721	0.9623	0.9260	0.9338
Turbocharger speed—line A	0.9947	0.9930	0.9870	0.9830
Turbine power gas—line A	0.9941	0.9939	0.9896	0.9857
Compressor power—line A	0.9942	0.9941	0.9896	0.9858
Turbine efficiency—line A	0.9924	0.9770	0.8937	0.9075
Compressor efficiency—line A	0.9694	0.9593	0.9281	0.9350
Air mass flow rate	0.9955	0.9957	0.9921	0.9907
BSFC	0.9589	0.9815	0.9381	0.9765

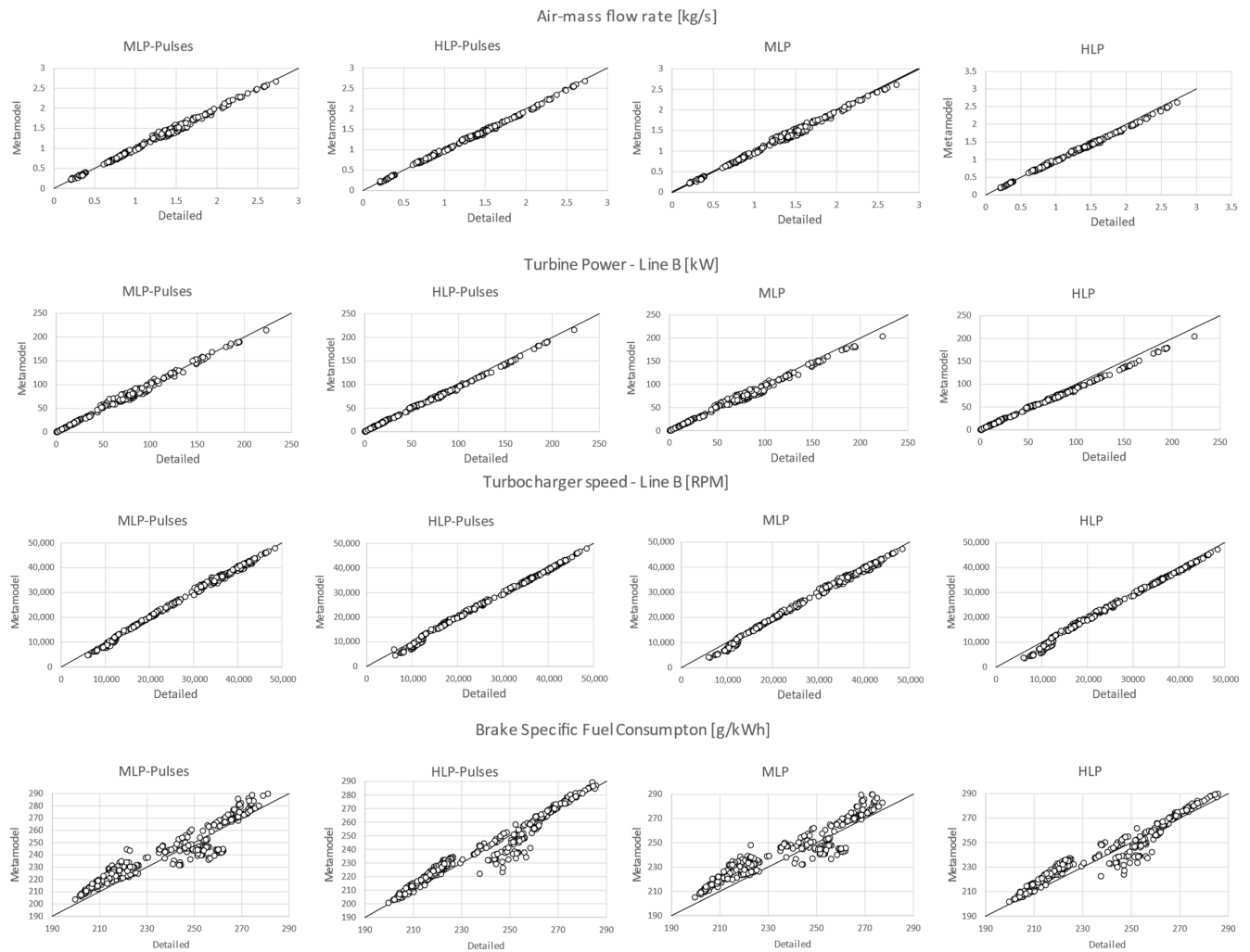


Figure 16. Comparison of metamodel prediction with the detailed model prediction in terms of air-mass flow rate, turbocharger power, turbocharger speed, and brake-specific fuel consumption.

Table 9. Root-mean-squared-error obtained during cross-over validation by evaluating in the complete engine model. Comparison between metamodel methods.

Variable Output	Unit	MLP with Pulses	HNN with Pulses	MLP w/o Pulses	HNN w/o Pulses
Injected mass per cycle	[mg/cycle]	60.24	36.72	80.27	53.40
Average of maximum cylinder pressures	[bar]	2.40	2.45	3.51	3.81
Boost pressure P3	[bar]	0.05	0.04	0.07	0.07
Backpressure P4	[bar]	0.03	0.04	0.05	0.06
Turbocharger speed—line B	[RPM]	834.99	956.44	1296.50	1502.18
Turbine power gas—line B	[kW]	3.52	3.41	5.09	6.02
Compressor power—line B	[kW]	3.51	3.37	5.11	6.03
Turbine efficiency—line B	[%]	0.48	0.82	1.70	1.66
Compressor efficiency—line B	[%]	1.12	1.31	1.80	1.75
Turbocharger speed—line A	[RPM]	840.48	963.43	1295.31	1498.94
Turbine power gas—line A	[kW]	3.61	3.68	4.82	5.66
Compressor power—line A	[kW]	3.58	3.63	4.84	5.67
Turbine efficiency—line A	[%]	0.45	0.79	1.65	1.60
Compressor efficiency—line A	[%]	1.17	1.35	1.76	1.72
Air mass flow rate	[kg/s]	0.04	0.04	0.05	0.05
BSFC	[g/kWh]	8.80	5.90	10.58	6.76

Despite these improvements, turbocharger speed was consistently underestimated at low speeds across all metamodels. This underestimation introduced a noticeable dispersion in BSFC predictions. However, the hierarchical network arrangement mitigated this issue to some extent. The HNN with pulses achieved the greatest reduction in RMSE, improving by 32.9% compared to the MLP with pulses.

Figure 17 illustrates the error distribution for BSFC predictions across the different metamodel configurations. The HLP with pulses demonstrated the most accurate performance, as evidenced by a higher concentration of points near the zero-error region, indicating better agreement with the detailed model.

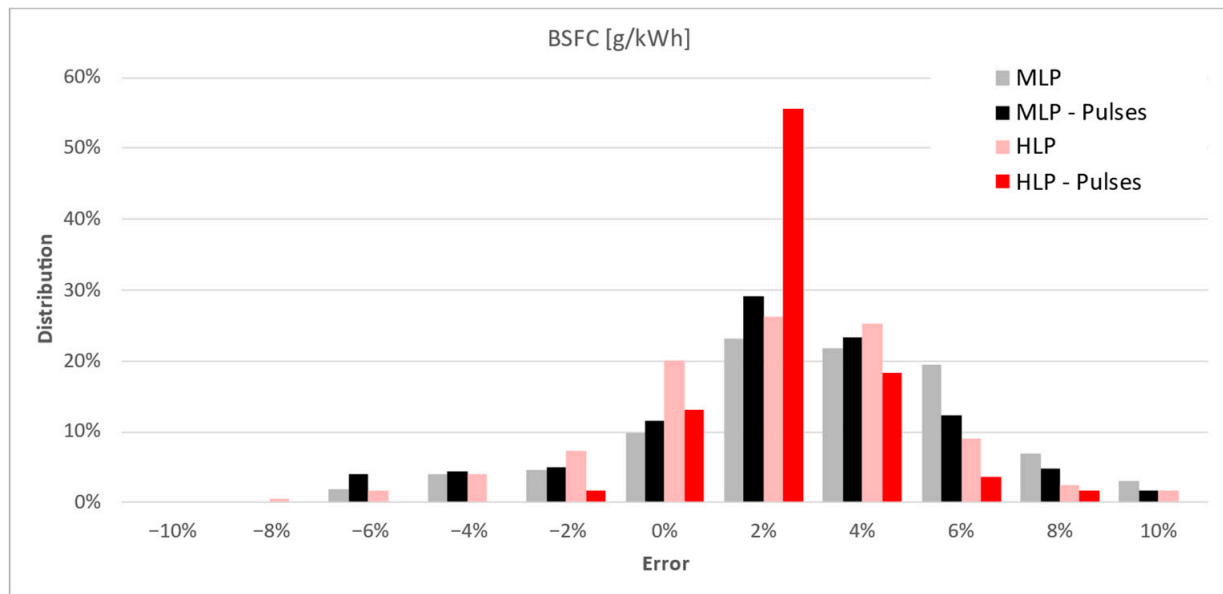


Figure 17. Histogram of error distribution of BSFC during cross-validation of the metamodel. Comparison between each metamodel architecture associated with the 1D air-charging circuit.

4. Discussion

The application of metamodels has significantly reduced computational time compared to the detailed engine model. Computational performance is compared in terms of average elapsed time per case. All cases are evaluated in a computer equipped with an Intel® Core™ i7-7800X–3.5 GHz processor and 32 Gb of RAM. The detailed model required an average of 165.5 s per case; real time is on average 30 times the simulated time. In contrast, metamodels with gas pulses required averages of 39.2 s and 36.4 s per case for MLP and HNN, respectively, with real-time factors of 9.8 and 9.2. Metamodels without pulses were even faster, requiring 22 and 24.6 s per case for MLP and HNN, respectively, with real-time factors averaging 7.4 and 7.2.

In total, the metamodels with pulses were approximately 4.5 times faster than the detailed model in terms of real elapsed time, while those without pulses achieved a speedup of 7 times. This improvement highlights the metamodels' computational efficiency, particularly in scenarios where pulse effects are neglected.

The primary factor limiting further reductions in computational time across all models is the dynamic behavior of the turbocharger. In GT-Suite, the steady-state solver operates as a variation of the transient solver, with variable time-step sizes. Turbocharger inertia delays the system's convergence, as the turbocharger shaft must accelerate or decelerate to its final steady-state speed. To address this, turbocharger inertia was dynamically adjusted during all simulations.

Higher inertia values ($\times 100$) were applied up to 0.4 s of simulated time, as the stabilization of the different initializations of all volumes creates large oscillations in turbocharger speed in the very beginning of the simulation. Then, at 0.4 s of simulated time, the strategy is followed by lower inertia values ($\times 0.01$) until 1.7 s to speed up the evolution of speed from the initial to the final value stabilization, a side effect of the lower inertia is an increase of the amplitude of the pulse-oscillations. Finally, realistic inertia values ($\times 1$) were restored once passed 1.7 s. All simulations were forced to run for at least 2.1 s to ensure stabilization.

Figure 18 illustrates the evolution of turbocharger speed (Line A) across all models for a common operating point (100% load, 1050 RPM, baseline turbocharger size, and SOI at -17° TDC). It shows that pulses in the metamodels are activated after one second of simulated time, allowing the turbocharger speed to stabilize with a larger time-step value, reducing the overall computation time.

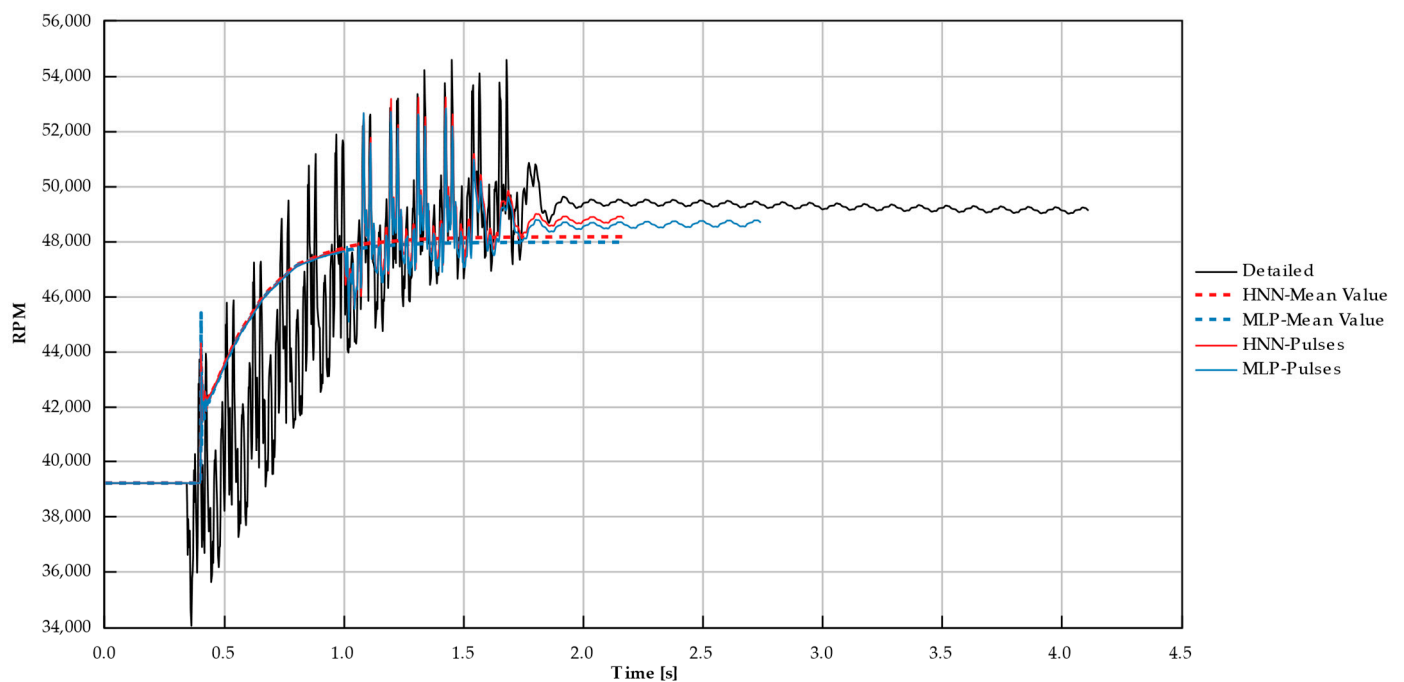


Figure 18. Evolution of turbocharger speed versus simulated time (in seconds) evaluated on all models presented. The same initial conditions and same engine operating point (100% load–1050 RPM, baseline size of turbocharger, SOI at -17° of TDC).

4.1. Error Reduction Due to Pulsating Mass Flow Rate

The adoption of pulses improves the prediction of all variables in both arrangements (MLP and HLP). Pulsating flow becomes significant when the pulse traveling time is comparable to the pulse period. A previous study [44] has identified that a pulsating flow can be considered as a mean-value only when the product of pressure fluctuation factor (Π) and traveling time (St) approaches zero. This relationship is determined by Equation (11) [44],

$$\frac{2 \cdot \Delta P}{\gamma \cdot P_0} \cdot \frac{L_0}{v_0 \cdot t_0} = \Pi \cdot St = \Lambda \quad (11)$$

At high engine speeds and loads, the Λ factor of Equation (11) evaluated in the exhaust line (1.65 m in length) achieves values near 0.5, with speeds in the order of magnitude of 40 m/s and periods of oscillation of 0.11 s for the pulses shown in Figure 11.

This order of magnitude indicates a significant influence of pulsation on flow behavior in the exhaust line at high engine speeds and loads. Neglecting this effect introduces larger errors, particularly at those operating points. Figure 16 supports this observation,

showing that turbine power is underestimated at high power levels in those metamodels not considering pulses.

4.2. Error Reduction Due to Hierarchical Layer Arrangement

The hierarchical neural network (HNN) arrangement improved prediction of the end-of-process variable brake-specific fuel consumption (BSFC), reducing the RMSE by 32.9% compared to the MLP arrangement (both with pulses). This error reduction is attributed to a more complex network for the end-of-process variable. The connectivity of this network is controlled using prior knowledge of the variable relationship, obtaining a deeper equivalent network than the one used with MLP.

However, this approach introduces a trade-off. While end-of-process variables are better predicted, intermediate variables—such as turbocharger speed—experience reduced accuracy. For instance, the turbocharger speed RMSE (Line B) increases by 14.5% in the HNN arrangement relative to the MLP arrangement (both with pulses). The proposed HNN connectivity arrangement is based in physics knowledge of the engine process. More advanced methods, such as semi-supervised optimum-path forest classifier [45], can determine the optimal connectivity of the networks involved, potentially improving the performance of the metamodel.

Previous studies have demonstrated that key variables like volumetric efficiency, in-cylinder pressure, and exhaust gas temperature can be effectively modeled using individual data-driven MLP networks with low RMSE values. Volumetric efficiency can be predicted using data-driven MLP with an RMSE below 0.01% [46], maximum in-cylinder pressure can be predicted with a data-driven approach, obtaining an RMSE below 0.05 bar [47], and exhaust gas temperature can be predicted with an RMSE below 10 °C [48]. These findings suggest the potential for integrating data-driven MLPs in the proposed hierarchical arrangement to reduce the total error between the model and real engine process.

5. Conclusions

This study demonstrates the advantages of using gray-box metamodels for engine system analysis, combining ANN-based approximations of engine block elements with a detailed 1D air-charging circuit model. Two network arrangements, including a novel hierarchical neural network (HNN) structure, were evaluated. The HNN arrangement significantly improves end-of-process variable predictions, reducing the BSFC RMSE by 32.9% compared to conventional MLP arrangements. Additionally, incorporating a double-Gaussian pulse for exhaust gas mass flow accurately captures unsteady flow dynamics, enhancing overall metamodel performance.

Future work will extend this framework to investigate electric hybrid air-charging systems in marine diesel engines, including the adoption of electric compressors and electric recovery turbines. Previous work by the researchers replaced the turbochargers of the 4 V-PA6B engine with electrically divided turbochargers, modeled using detailed 1D simulations. This early study evaluated the impact of various parameters on fuel consumption optimization. The next phase will compare different configurations, such as electric compressors in series or in parallel working in tandem with the turbocharger and electric recovery turbines working in tandem with turbocharger, in series or in parallel. The presented metamodel's ability to reduce computational time facilitates the multiple optimizations required for this comparative analysis within the available resources.

Author Contributions: Conceptualization, A.A.S., G.S. and P.C.; methodology, A.A.S. and G.S.; software, A.A.S.; investigation, A.A.S.; resources, Y.T.; writing—original draft preparation, A.A.S.; writing—review and editing, G.S. and Y.T.; supervision, N.B.; project administration, P.C. and N.B.

funding acquisition, P.C. and N.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Association Nationale de la Recherche et de la Technologie (CIFRE 2020/1035) and MAN Energy Solutions France SAS.

Data Availability Statement: The original contributions presented in the study are included in the article; further inquiries can be directed to the corresponding author.

Acknowledgments: The authors wish to thank Olivier Giroux and Vivien Leroy for their technical support during the study.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

AFR _{trapped}	Air-to-fuel ratio trapped inside the closed cylinder
ANN	Artificial Neural Network
BSFC	Brake-specific fuel consumption [g/kWh]
CA	Crank angle
DOE	Design of Experiments
ECU	Electronic Control Unit
EVC	Exhaust valve closing shifter value
EVO	Exhaust valve opening shifter value
HNN	Hierarchical Neural Network
ICE	Internal combustion engine
ISO	International Standardization Office
IVC	Intake valve closing shifter value
IVO	Intake valve opening shifter value
L_0	Reference length of the element
MLP	Multi-layer perceptron
N_{cyl}	Turbine speed [rpm]
P_{brake}	Brake power [kW]
P_0	Mean-value of pressure waves
RMSE	Root-mean squared error
ROM	Reduced-Order Modeling
RPM	Revolutions per minute
SAE	Society of Automotive Engineers
SOI	Start of injection
St	Strouhal number
TDC	Top dead center
ΔP	Amplitude of pressure waves
$\dot{m}_{air}$	Mass flow rate of air
$\dot{m}_{fuel}$	Mass flow rate of fuel
$\dot{m}_{gas}$	Mass flow rate of exhaust gases, instantaneous value
$\dot{m}_{gas, MV}$	Mass flow rate of exhaust gases, mean value
γ	Heat capacity ratio
Λ	Unsteadiness criterion
Π	Pressure amplitude weighting factor
t	Instantaneous simulated time
t_0	Period of the pressure wave inside the element
θ	Instantaneous crank angle
v_0	Average speed of the pressure wave along the element

Appendix A

MLP Artificial neural networks

Almost every function can be expressed as a weighted sum of M basis functions (ϕ_i). On Equation (A1), the expression of a general basis function formulation is shown,

$$\hat{y} = \sum_{i=0}^M \theta_i \Phi_i(u, \theta_i^{(nl)}) \quad (A1)$$

where the linear weights are expressed as θ_i and a set of non-linear parameters expressed as $\theta_i^{(nl)}$. This decomposition on basis functions is the base of every ROM method.

The specific case where all basis functions are the same, differing by their non-linear parameters, is called an artificial neural network. Figure A1 represents a network of basic functions. In the case of an ANN, each node is called a neuron.

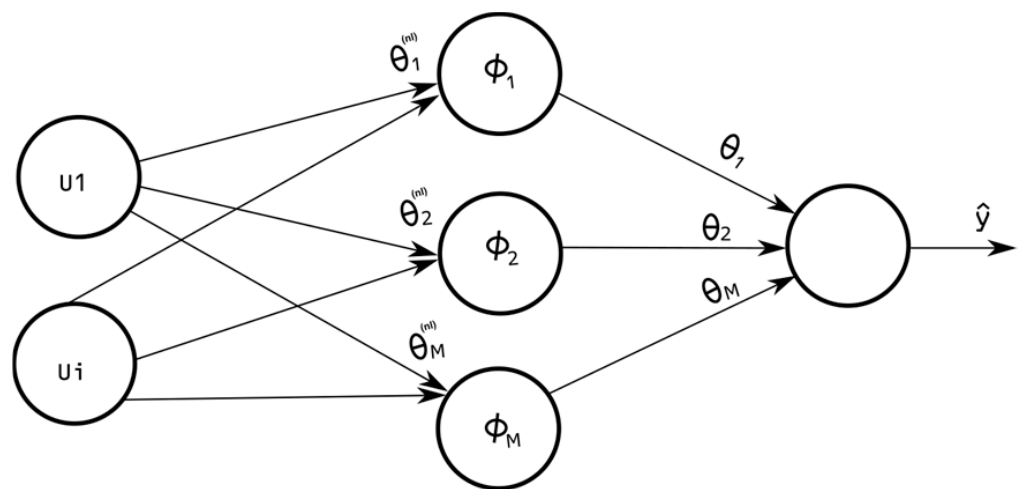


Figure A1. Diagram of a neural network.

MLP is the most common ANN structure. Each neuron is called a perceptron, and its basis function is called an activation function. The activation function used in this study is a logistic sigmoid function scaled in the range $[-1,1]$ in the form expressed in Equation (A2). This nonlinear activation function allows the MLP to fit nonlinear processes.

$$\begin{cases} g_i(x) = \frac{2}{1+\exp(-2x)} - 1, & \text{for } i > 0 \\ g_0(x) = 1, & \text{for } i = 0 \end{cases} \quad (A2)$$

For a two-hidden layer MLP, as seen in Figure A2, the output signal from Equation (5) is rewritten as Equation (A3). The weights of the first layer $w^{(1)}$ are linearly combined with the p inputs u_l to compute the argument of the activation function $g(x)$ of the M_1 neurons of the first layer. The M_1 outputs of the first layer are then linearly combined with the M_2 weights $w^{(2)}$ of the second layer in the same manner as the first layer. Finally, these M_2 outputs are linearly combined with the weights of the output layer w_i to obtain the output signal $\hat{y}$.

$$\hat{y} = \sum_{i=0}^{M_2} w_i \cdot g_i \left(\sum_{j=0}^{M_1} w_{ij}^{(2)} \cdot g_j \left(\sum_{l=0}^p w_{jl}^{(1)} \cdot u_l \right) \right) \quad (A3)$$

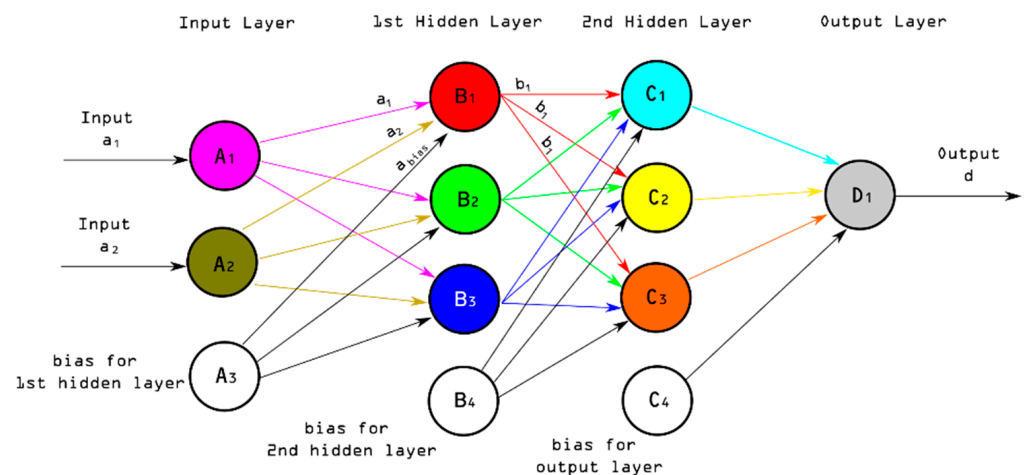


Figure A2. Diagram of the two-hidden-layer multi-layer perceptron architecture of a neural network.

References

- Schilders, W. Introduction to model order reduction. In *Model Order Reduction: Theory, Research Aspects and Applications*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 3–32.
- Choroszuca, R.B.; Sun, J.; Butts, K. Closed-loop model order reduction and MPC for diesel engine airpath control. In Proceedings of the 2015 American Control Conference (ACC), Chicago, IL, USA, 1–3 July 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 3279–3284.
- Stockar, S.; Canova, M.; Guezennec, Y. Model-order reduction for prediction of pressure wave propagation dynamics in the IC engine air path system. In Proceedings of the 2013 American Control Conference, Washington, DC, USA, 17–19 June 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 5201–5206.
- Parlak, A.; Islamoglu, Y.; Yasar, H.; Egrisogut, A. Application of artificial neural network to predict specific fuel consumption and exhaust temperature for a diesel engine. *Appl. Therm. Eng.* **2006**, *26*, 824–828. [\[CrossRef\]](#)
- Togun, N.K.; Baysec, S. Prediction of torque and specific fuel consumption of a gasoline engine by using artificial neural networks. *Appl. Energy* **2010**, *87*, 349–355. [\[CrossRef\]](#)
- Boruah, D.; Thakur, P.K.; Baruah, D. Artificial neural network based modelling of internal combustion engine performance. *Int. J. Eng. Res. Technol.* **2016**, *5*, 568–576.
- Boz, T.; Unel, M.; Aran, V.; Yilmaz, M.; Gurel, C.; Bayburtlu, C.; Koprubasi, K. Diesel engine NO_x emission modeling with airpath input channels. In Proceedings of the IECON 2015—41st Annual Conference of the IEEE Industrial Electronics Society, Yokohama, Japan, 9–12 November 2015; IEEE: Piscataway, NJ, USA, 2015; pp. 003382–003387.
- Mohammadhassani, J.; Khalilarya, S.; Solimanpur, M.; Dadvand, A. Prediction of NO_x emissions from a direct injection diesel engine using artificial neural network. *Model. Simul. Eng.* **2012**, *2012*, 830365.
- Salam, S.; Verma, T.N. Appending empirical modelling to numerical solution for behaviour characterisation of microalgae biodiesel. *Energy Convers. Manag.* **2019**, *180*, 496–510. [\[CrossRef\]](#)
- Ghobadian, B.; Rahimi, H.; Nikbakht, A.; Najafi, G.; Yusaf, T. Diesel engine performance and exhaust emission analysis using waste cooking biodiesel fuel with an artificial neural network. *Renew. Energy* **2009**, *34*, 976–982. [\[CrossRef\]](#)
- Kumar, S.; Pai, P.S.; Rao, B.S. Artificial neural network based prediction of performance and emission characteristics of a variable compression ratio CI engine using WCO as a biodiesel at different injection timings. *Appl. Energy* **2011**, *88*, 2344–2354.
- Arumugam, S.; Sriram, G.; Subramanian, P.S. Application of artificial intelligence to predict the performance and exhaust emissions of diesel engine using rapeseed oil methyl ester. *Procedia Eng.* **2012**, *38*, 853–860. [\[CrossRef\]](#)
- Rezaei, J.; Shahbakhti, M.; Bahri, B.; Aziz, A.A. Performance prediction of HCCI engines with oxygenated fuels using artificial neural networks. *Appl. Energy* **2015**, *138*, 460–473. [\[CrossRef\]](#)
- Cho, J.; Song, S. Prediction of hydrogen-added combustion process in T-GDI engine using artificial neural network. *Appl. Therm. Eng.* **2020**, *181*, 115974. [\[CrossRef\]](#)
- Cervantes-Bobadilla, M.; Escobar-Jiménez, R.F.; Gómez-Aguilar, J.F.; García-Morales, J.; Olivares-Peregrino, V.H. Experimental study on the performance of controllers for the hydrogen gas production demanded by an internal combustion engine. *Energies* **2018**, *11*, 2157. [\[CrossRef\]](#)
- Cervantes-Bobadilla, M.; García-Morales, J.; Escobar-Jiménez, R.; Hernández-Pérez, J.; Gómez-Aguilar, J.; Olivares-Peregrino, V. Experimental implementation of a new control approach using an inverse neural network to on-demand hydrogen production. *Control Eng. Pract.* **2020**, *105*, 104631. [\[CrossRef\]](#)
- Yap, W.K.; Ho, T.; Karri, V. Exhaust emissions control and engine parameters optimization using artificial neural network virtual sensors for a hydrogen-powered vehicle. *Int. J. Hydrogen Energy* **2012**, *37*, 8704–8715. [\[CrossRef\]](#)

18. Mehra, R.K.; Duan, H.; Luo, S.; Rao, A.; Ma, F. Experimental and artificial neural network (ANN) study of hydrogen enriched compressed natural gas (HCNG) engine under various ignition timings and excess air ratios. *Appl. Energy* **2018**, *228*, 736–754. [\[CrossRef\]](#)
19. Ho, T.; Karri, V. Basic tuning of hydrogen powered car and artificial intelligent prediction of hydrogen engine characteristics. *Int. J. Hydrogen Energy* **2010**, *35*, 10004–10012. [\[CrossRef\]](#)
20. Syed, J.; Baig, R.U.; Algarni, S.; Murthy, Y.S.; Masood, M.; Inamurrahman, M. Artificial Neural Network modeling of a hydrogen dual fueled diesel engine characteristics: An experiment approach. *Int. J. Hydrogen Energy* **2017**, *42*, 14750–14774. [\[CrossRef\]](#)
21. Yusri, I.; Majeed, A.A.; Mamat, R.; Ghazali, M.; Awad, O.I.; Azmi, W. A review on the application of response surface method and artificial neural network in engine performance and exhaust emissions characteristics in alternative fuel. *Renew. Sustain. Energy Rev.* **2018**, *90*, 665–686. [\[CrossRef\]](#)
22. Hoang, A.T.; Nižetić, S.; Ong, H.C.; Tarelko, W.; Le, T.H.; Chau, M.Q.; Nguyen, X.P. A review on application of artificial neural network (ANN) for performance and emission characteristics of diesel engine fueled with biodiesel-based fuels. *Sustain. Energy Technol. Assess.* **2021**, *47*, 101416.
23. Zhao, J.; Xu, M.; Li, M.; Wang, B.; Liu, S. Design and optimization of an Atkinson cycle engine with the Artificial Neural Network Method. *Appl. Energy* **2012**, *92*, 492–502. [\[CrossRef\]](#)
24. Cavazzuti, M. *Optimization Methods: From Theory to Design Scientific and Technological Aspects in Mechanics*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012.
25. Zhao, J.; Ma, Y.; Zhang, Z.; Wang, S.; Wang, S. Optimization and matching for range-extendors of electric vehicles with artificial neural network and genetic algorithm. *Energy Convers. Manag.* **2019**, *184*, 709–725. [\[CrossRef\]](#)
26. Zhou, G.; Liu, R.; Zhang, Z.; Yang, C.; Ding, H. Optimization of diesel engine dual-variable geometry turbocharger regulated two-stage turbocharging system based on radial basis function neural network-quantum genetic algorithm. *Energy Sources Part A Recovery Util. Environ. Eff.* **2021**, 1–17. [\[CrossRef\]](#)
27. Nelles, O. *Nonlinear System Identification*; IOP Publishing: Bristol, UK, 2002.
28. Mavrovouniotis, M.L.; Chang, S. Hierarchical neural networks. *Comput. Chem. Eng.* **1992**, *16*, 347–369. [\[CrossRef\]](#)
29. Wang, F.; Devabhaktuni, V.K.; Zhang, Q.-J. A hierarchical neural network approach to the development of a library of neural models for microwave design. *IEEE Trans. Microw. Theory Tech.* **1998**, *46*, 2391–2403. [\[CrossRef\]](#)
30. Sharkey, A.J.C. On combining artificial neural nets. *Connect. Sci.* **1996**, *8*, 299–314. [\[CrossRef\]](#)
31. Jansen, W.J.; Diepenhorst, M.; Nijhuis, J.; Spaanenburg, L. Assembling engineering knowledge in a modular multi-layer perceptron neural network. In Proceedings of the International Conference on Neural Networks (ICNN'97), Houston, TX, USA, 12 June 1997; IEEE: Piscataway, NJ, USA, 1997; Volume 1, pp. 232–237.
32. Lendaris, G.G.; Rest, A.; Misley, T.R. Improving ANN generalization using a priori knowledge to pre-structure ANNs. In Proceedings of the International Conference on Neural Networks (ICNN'97), Houston, TX, USA, 12 June 1997; IEEE: Piscataway, NJ, USA, 1997; Volume 1, pp. 248–253.
33. Sediako, A.D.; Andric, J.; Sjoblom, J.; Faghani, E. *Heavy Duty Diesel Engine Modeling with Layered Artificial Neural Network Structures*; SAE Technical Paper; SAE International: Warrendale, PA, USA, 2018.
34. Cong, Q.; Yu, W.; Chai, T. Cascade process modeling with mechanism-based hierarchical neural networks. *Int. J. Neural Syst.* **2010**, *20*, 1–11. [\[CrossRef\]](#)
35. Salazar, A.A.; Salameh, G.; Chesse, P.; Bulot, N.; Thevenoux, Y. Impact of Optimization Variables on Fuel Consumption in Large Four-Stroke Diesel Marine Engines with Electrically Divided Turbochargers. *Machines* **2024**, *12*, 926. [\[CrossRef\]](#)
36. Xavier, T.; Cyril, R.; Simon, P.; Pascal, C.; Georges, S. A New Simulation-Based Methodology to Improve Medium-Speed Diesel Engines BSFC/NOx Tradeoff With an Optimized Air Charging System. *J. Eng. Gas Turbines Power* **2023**, *145*, 011009. [\[CrossRef\]](#)
37. ISO 8178-4; Reciprocating Internal Combustion Engines—Exhaust Emission Measurement—Part 4: Steady-State and Transient Test Cycles for Different Engine Applications. ISO: Geneva, Switzerland, 2020.
38. Jung, D.; Assanis, D.N. Multi-zone DI diesel spray combustion model for cycle simulation studies of engine performance and emissions. *SAE Trans.* **2001**, *110*, 1510–1532.
39. Hatz, R.; Zhou, H.; Diez, C.; Wachtmeister, G.; Jaensch, M. *Investigation and Comparison of the Prediction Capabilities of Multiple 0D/1D Combustion Calibration Strategies Using Different Turbocharger Systems as Calibration Basis*; SAE Technical Paper; SAE International: Warrendale, PA, USA, 2022.
40. Stoumpos, S.; Theotokatos, G.; Boulougouris, E.; Vassalos, D.; Lazakis, I.; Livanos, G. Marine dual fuel engine modelling and parametric investigation of engine settings effect on performance-emissions trade-offs. *Ocean Eng.* **2018**, *157*, 376–386. [\[CrossRef\]](#)
41. GT-Suite. *Engine Performance Application Manual*; Gamma Technologies: Westmont, IL, USA, 2023.
42. Watson, N.; Janota, M. *Turbocharging the Internal Combustion Engine*; Macmillan International Higher Education: London, UK, 1982.
43. Li, R.; Huang, Y.; Li, G.; Han, K.; Song, H. Calibration and validation of a mean value model for turbocharged diesel engine. *Adv. Mech. Eng.* **2013**, *5*, 579503. [\[CrossRef\]](#)

44. Copeland, C.; Newton, P.; Martinez-Botas, R.; Seiler, M. A comparison of timescales within a pulsed flow turbocharger turbine. In Proceedings of the 10th International Conference on Turbochargers and Turbocharging, London, UK, 15–16 May 2012; pp. 389–404.
45. Amorim, W.P.; Falcão, A.X.; Papa, J.P.; Carvalho, M.H. Improving semi-supervised learning through optimum connectivity. *Pattern Recognit.* **2016**, *60*, 72–85. [[CrossRef](#)]
46. de Nola, F.; Giardiello, G.; Gimelli, A.; Molteni, A.; Muccillo, M.; Picariello, R. Volumetric efficiency estimation based on neural networks to reduce the experimental effort in engine base calibration. *Fuel* **2019**, *244*, 31–39. [[CrossRef](#)]
47. Ristow Hadlich, R.; Loprete, J.; Assanis, D. A Deep Learning Approach to Predict In-Cylinder Pressure of a Compression Ignition Engine. *J. Eng. Gas Turbines Power* **2024**, 1–12. [[CrossRef](#)]
48. Liu, J.; Huang, Q.; Ulishney, C.; Dumitrescu, C.E. Machine learning assisted prediction of exhaust gas temperature of a heavy-duty natural gas spark ignition engine. *Appl. Energy* **2021**, *300*, 117413. [[CrossRef](#)]

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.