sensors

Article

Neural Network Signal Integration from Thermogas-Dynamic Parameter Sensors for Helicopters Turboshaft Engines at Flight Operation Conditions

Serhii Vladov [1], Lukasz Scislo [2,*], Valerii Sokurenko [3], Oleksandr Muzychuk [3], Victoria Vysotska [4,5], Serhii Osadchy [6] and Anatoliy Sachenko [7,8]

[1] Department of Scientific Work Organization and Gender Issues, Kremenchuk Flight College of Kharkiv National University of Internal Affairs, 17/6 Peremohy Street, 39605 Kremenchuk, Ukraine; serhii.vladov@univd.edu.ua

[2] Faculty of Electrical and Computer Engineering, Cracow University of Technology, Warszawska 24, 31-155 Craków, Poland

[3] Kharkiv National University of Internal Affairs, Ministry of Internal Affairs of Ukraine, 61080 Kharkiv, Ukraine; rector_hnuvs@ukr.net (V.S.); o.muzychuk23@gmail.com (O.M.)

[4] Information Systems and Networks Department, Lviv Polytechnic National University, 12 Bandera Street, 79013 Lviv, Ukraine

[5] Institute of Computer Science, Osnabrück University, 1 Friedrich-Janssen-Street, 49076 Osnabrück, Germany

[6] Flight Operation and Flight Safety Department, Flight Academy of the National Aviation University, 1 Chobanu Stepana Street, 25005 Kropyvnytskyi, Ukraine; srg2005@sfa.org.ua

[7] Research Institute for Intelligent Computer Systems, West Ukrainian National University, 11 Lvivska Street, 46009 Ternopil, Ukraine; as@wunu.edu.ua

[8] Department of Teleinformatics, Kazimierz Pulaski University of Radom, 29, Malczewskiego Street, 26-600 Radom, Poland

* Correspondence: lscislo@pk.edu.pl

Abstract: The article's main provisions are the development and application of a neural network method for helicopter turboshaft engine thermogas-dynamic parameter integrating signals. This allows you to effectively correct sensor data in real time, ensuring high accuracy and reliability of readings. A neural network has been developed that integrates closed loops for the helicopter turboshaft engine parameters, which are regulated based on the filtering method. This made achieving almost 100% (0.995 or 99.5%) accuracy possible and reduced the loss function to 0.005 (0.5%) after 280 training epochs. An algorithm has been developed for neural network training based on the errors in backpropagation for closed loops, integrating the helicopter turboshaft engine parameters regulated based on the filtering method. It combines increasing the validation set accuracy and controlling overfitting, considering error dynamics, which preserves the model generalization ability. The adaptive training rate improves adaptation to the data changes and training conditions, improving performance. It has been mathematically proven that the helicopter turboshaft engine parameters regulating neural network closed-loop integration using the filtering method, in comparison with traditional filters (median-recursive, recursive and median), significantly improve efficiency. Moreover, that enables reduction of the errors of the 1st and 2nd types: 2.11 times compared to the median-recursive filter, 2.89 times compared to the recursive filter, and 4.18 times compared to the median filter. The achieved results significantly increase the helicopter turboshaft engine sensor readings accuracy (up to 99.5%) and reliability, ensuring aircraft efficient and safe operations thanks to improved filtering methods and neural network data integration. These advances open up new prospects for the aviation industry, improving operational efficiency and overall helicopter flight safety through advanced data processing technologies.

Keywords: sensor; integration; filtration; helicopter turboshaft engine; neural network; error

check for updates

Citation: Vladov, S.; Scislo, L.; Sokurenko, V.; Muzychuk, O.; Vysotska, V.; Osadchy, S.; Sachenko, A. Neural Network Signal Integration from Thermogas-Dynamic Parameter Sensors for Helicopters Turboshaft Engines at Flight Operation Conditions. *Sensors* **2024**, *24*, 4246. https://doi.org/10.3390/s24134246

Academic Editors: Jesmin Farzana Khan and Mandoye Ndoye

Received: 3 June 2024
Revised: 25 June 2024
Accepted: 27 June 2024
Published: 29 June 2024

1. Introduction and Related Work

Helicopter turboshaft engines (TEs) are complex technical devices requiring continuous monitoring of their condition [1]. Helicopter TEs' reliable operation depends on the accurate monitoring of thermogas-dynamic parameters, such as temperature and pressure at the engine's various points, gas–generator rotor rpm and free turbine rotor speed, fuel consumption and others [2]. Modern monitoring and diagnostic systems use many sensors to obtain this data, which makes it possible to detect deviations from the norm and prevent possible malfunctions promptly. However, under flight operating conditions, sensors are exposed to various factors that affect measurement accuracy [3].

The helicopter turboshaft engine sensors must guarantee that data are received with high accuracy to ensure the helicopter's safe operation. Sensors that measure gas–generator rotor rpm, free turbine rotor speed, and gas temperature in the compressor turbine front are the engine control system's critical components [4]. In real operating conditions, situations may arise when sensors begin to provide inadequate information due to various factors, such as noise, interference, sensor malfunctions, etc. Even though the sensor is recognized as operational, the information received may be distorted. This can lead to incorrect conclusions about the engine state and, as a result, to erroneous actions by the crew, which are critically dangerous in flight conditions.

Traditional methods for processing sensor signals often face problems associated with noise, sensor failures, and imperfect mathematical models [5–8]. As a result, a situation may arise when the control system receives distorted data, which leads to incorrect conclusions about the engine state [9,10]. This is especially critical for in-flight operating conditions, where the measurement accuracy can be affected by vibrations, temperature changes and other external factors [11,12].

To solve these problems, data integration methods [13,14] are of particular importance, which makes it possible to combine information from various sensors and systems, the errors impact minimizing and the diagnostic reliability increase. Data integration is based on the algorithms used to process signals, taking into account their relations and correlations [15]. This makes obtaining a more accurate and complete picture of the engine state possible.

The relevance of data integration methods in helicopter TE control and diagnostic systems is due to several key factors. Increasing helicopter TE reliability is a priority task in aviation [16]. Data integration systems make it possible to integrate and jointly process information from multiple sensors, which significantly increases the accuracy and reliability of engine condition diagnostics [17]. Such systems become indispensable in-flight operating conditions, where failures and errors can lead to serious consequences. They provide continuous monitoring and allow you to quickly respond to any changes in engine operation, preventing potential emergencies.

Secondly, data integration methods significantly reduce the influence of external factors, such as vibrations and temperature fluctuations, which can distort the individual sensors' readings [18]. It is possible to minimize measurement errors and increase the accuracy of the obtained data by taking into account the relations between various parameters and using correction algorithms [19]. This is especially important in helicopter operations, where the environment and flight conditions can vary significantly, affecting sensor performance.

Thirdly, data integration contributes to early fault detection and deviations from the norm [20]. Combining information from various sensors allows potential problems to be detected faster and more accurately, which makes it possible to carry out timely preventive and repair work [21]. This significantly reduces the risk of sudden failure and accidents, increasing overall flight safety. Early fault detection also helps to extend equipment life and reduce operating costs.

Finally, data fusion techniques are crucial for maintenance optimization [22]. Accurately determining the engine components' conditions allows for effective maintenance planning, minimizing downtime and reducing costs. This enhances the helicopter's operational performance and mission readiness.

In [23], an aircraft and spacecraft hull structures state acoustic emission diagnostics method was developed using a distributed fibre-optic sensor system. It is shown that such systems can provide vital information for ensuring safety at aerospace facilities. Green's function for a single sensor is found, and its features under the pulsed influence of acoustic emission signals are researched. The sensor system's ability to estimate the coordinates and acoustic emission signal parameters has been determined. Experimental research has confirmed the system's ability to detect radiation signals even under specific interference conditions. A key disadvantage is the difficulty in distributing fibre optic sensors into existing aerospace designs, which may limit their widespread use and require significant modifications during implementation.

In [24], challenges focused on flexibility and scalability are explored when integrating sensors for Industry 4.0 functions into manufacturing systems. These systems use reconfigurable machines with intelligent actuators connected through a single electromechanical interface. An adaptive sensor integration unit architecture is presented that supports interleaved communication protocols and reduces the number of signal lines and the need for protocol conversion units. A prototype system using dynamic partial FPGA reconfiguration has been demonstrated to be effective in an industrial environment. The key disadvantages are the complexity and high cost of developing and implementing a dynamic partial reconfiguration FPGA system and the need for specialized knowledge to operate and maintain it.

The multi-stage supercharging process in [25] is investigated, which is an effective method for improving the engine's volumetric efficiency at high altitudes, where the intercooler must withstand high thermal loads caused by high levels of supercharging. However, the existing research's disadvantages on intercooler performance at high altitudes require additional research to optimize its performance in actual flight conditions.

The research [26] aims to modify engines' advanced knowledge to run on ammonia to promote decarbonization by demonstrating that compression ignition engines can efficiently burn ammonia using spark plugs. The disadvantage is that existing research is limited, and the available experimental data are randomly distributed without careful design, increasing the results' uncertainty.

Thus, the data integration methods used in monitoring and control systems for the helicopter TE operation are an integral part of modern technology [23–26] to increase aircraft operation safety and efficiency. The development and implementation of such methods require an interdisciplinary approach and the advanced advances used in the signal processing [27,28], mathematical modelling [29,30] and artificial intelligence [31,32] field. One of these technologies is the neural network approach [33,34], which has several unique advantages over traditional methods [35,36].

Firstly, neural network algorithms can efficiently process large amounts of data and identify complex dependencies between various parameters, which is impossible when using classical signal processing methods. Neural networks can be trained on large amounts of data, which allows them to adapt to changing operating conditions and improve the accuracy of engine prediction [37].

Secondly, neural network models are highly resistant to noise and data failures, which is especially important in flight conditions, where data may be subject to various external influences. Thanks to the ability to self-train and adapt, neural networks can effectively compensate for distortions and provide more accurate diagnostic results [38].

In addition, neural network approaches make it possible to implement more complex and accurate mathematical models that consider nonlinear dependencies between parameters and their time changes. This provides the helicopter TE operational status with a more in-depth analysis and accurate prediction, which increases the helicopter operation reliability and safety [39].

Introducing neural network technology into helicopter TE control and diagnostic systems is a promising development direction that opens up new opportunities for increasing aircraft operations efficiency and reliability.

The proposed method from helicopter TE thermogas-dynamic parameter sensor signals neural network integration represents significant improvements compared to traditional methods due to the dynamic compensation circuit (sensor signals adjustment in real time to improve control system operations) use and a neural network for closed parameter control loop integration. Dynamic compensation with an adaptive noise suppression device effectively filters noise interference while maintaining the helicopter TE parameter essential characteristics sensor signals, improving the control system's accuracy and reliability.

A goal of this work is to develop a method for the helicopter TE thermogas-dynamic parameters integrating signals coming from sensors in real-time and correcting them if they contain noise or distortion to ensure the sensor readings' accuracy and reliability, despite the possible noise and distortion, to maintain safe and efficient helicopter control. To reach this goal, it is necessary to solve the following tasks:

1. Diagram development for integrating signals from the helicopter TE thermogas-dynamic parameter sensors based on the filtering method.
2. Neural network development to implement a diagram for integrating signals from the helicopter TE thermogas-dynamic parameter sensors using the filtering method.
3. Neural network training algorithm development.
4. Helicopter TE thermogas-dynamic parameter sensor signals' analysis and preliminary processing.
5. Conduct a computational experiment to solve the filtering sensors signal task of the helicopter TE thermogas-dynamic parameters (using a gas–generator rotor rpm signal example).
6. Evaluate the results obtained effectiveness according to efficiency metrics (efficiency coefficient, quality coefficient, accuracy, recall, precision, F1-score, etc.).
7. The 1st and 2nd errors are calculated and the results obtained are compared with known analogues.

2. Materials and Methods

2.1. Diagram Development for Integrating Signals from Helicopter TE Thermogas-Dynamic Parameter Sensors Using the Filtering Method

It is known [40–42] that the on-board system for monitoring helicopter TE parameters uses the following sensors: D-2M (records the gas–generator rotor rpm), 14 dual thermocouples T-102 (records the gas temperature in front of the compressor turbine), and D-1M (records free turbine rotor speed). These sensors transmit recorded parameters to the onboard instrument panel, providing the pilot with important information about the engine's condition. However, all of these sensors are subject to noise, which reduces the accuracy and reliability of the recorded data. This requires the neural network aggregation and filtering methods used for signal processing to reduce the noise influence and increase the information reliability displayed on the dashboard.

In the first stage, data reliability is analyzed, which makes it possible to identify and correct anomalies in the data received by helicopter turboshaft engine sensors. According to the helicopter TE sensors readings at times t_1, t_2, …, t_n, a thermogas-dynamic parameter time series is formed:

$$\begin{aligned} N_{TC} &= \{n_{TC\ 1}, n_{TC\ 2}, \ldots,\ n_{TC\ n}\}, \\ T_G &= \left\{T^*_{G\ 1}, T^*_{G\ 2}, \ldots,\ T^*_{G\ n}\right\}, \\ N_{FT} &= \{n_{FT\ 1}, n_{FT\ 2}, \ldots,\ n_{FT\ n}\}, \end{aligned} \tag{1}$$

where $n_{TC\ 1}$, $n_{TC\ 2}$, …, $n_{TC\ n}$ is the gas–generator rotor rpm values at times t_1, t_2, …, t_n; $T^*_{G\ 1}$, $T^*_{G\ 2}$, …, $T^*_{G\ n}$ is the gas temperature in the compressor turbine values front at times t_1, t_2, …, t_n; and $n_{FT\ 1}$, $n_{FT\ 2}$, …, $n_{FT\ n}$ is the free turbine rotor speed values at times t_1, t_2, …, t_n.

The interquartile range (*IQR*) method is used [43] to remove outliers from the helicopter TE thermogas-dynamic parameters data:

$$\begin{gathered} IQR = Q_3 - Q_1, \\ BL = Q_1 - 1.5 \cdot IQR, \\ TL = Q_1 + 1.5 \cdot IQR, \end{gathered} \tag{2}$$

where Q_1 and Q_3 are the 1st and 3rd quartiles, respectively, *BL* is the lower limit, and *TL* is the upper limit.

IQR is a statistical tool used to identify and remove outliers from a data set, which is especially important when the helicopter TE thermogas-dynamic parameter time series are analysed. *IQR* is defined as the difference between the 3rd Q_3 and 1st Q_1 quartiles, which are the values below which the data lie at 75% and 25%, respectively. The *IQR* calculation allows us to set the *BL* lower and TL upper bounds. Data that falls outside these boundaries are considered outliers, anomalous values that can skew the analysis. Removing outliers helps ensure data reliability and accuracy, which is critical for helicopter health reliable monitoring and diagnosis, preventing errors and improving operational safety.

Q_1 and Q_3 are calculated for each data set to apply *IQR* to eliminate abnormal data preprocessing. Then, values below *BL* or above *TL* are excluded from the analysis. Once outliers are removed and cleaned, further processing and analysis can be carried out. This helps ensure data reliability and accuracy, which is critical for helicopter TE operational status reliable monitoring and diagnostics, preventing errors and increasing operational safety.

The reference range standard is based on the central tendency and variability statistical measures used to determine normal and anomalous values in helicopter TE thermogas-dynamic parameter data sets recorded by sensors in onboard implementation conditions. In particular, it applies the concepts of quartiles, which divide ordered data into four equal parts. The Q_1 1st quartile and the Q_3 3rd quartile are the basis for determining the *IQR*, which is the difference between Q_3 and Q_1. This range covers the middle 50% of the data and provides a value spread measure.

To standardize the helicopter TE thermogas-dynamic parameter values, eliminating differences in their ranges and facilitating subsequent analysis, these values are brought to a single scale using normalization:

$$\begin{gathered} n_{TC}^{norm}(t) = \frac{n_{TC}(t) - \mu(n_{TC})}{\sigma(n_{TC})}, T_G^{*\,norm}(t) = \frac{T_G^*(t) - \mu\left(T_G^*\right)}{\sigma\left(T_G^*\right)}, \\ n_{FT}^{norm}(t) = \frac{n_{FT}(t) - \mu(n_{FT})}{\sigma(n_{FT})}, \end{gathered} \tag{3}$$

where $\mu(\bullet)$ and $\sigma(\bullet)$ are the helicopters TE thermogas-dynamic parameters n_{TC}, and T_G^* and n_{FT} are the average value and standard deviation, respectively.

Data from the helicopter TE thermogas-dynamic parameters that exceed the established *BL* and *TL* are considered anomalies since they demonstrate significant deviations from the expected values. These deviations may indicate serious problems such as faulty hardware malfunctions or sensors malfunctioning. Anomalous data require special attention and analysis, as they can lead to erroneous conclusions about the system state and, as a result, to potentially dangerous situations. Such anomalies' identification and correction are critical to maintaining reliability and helicopter TE operation safety.

To do this, for each *i*-th helicopter TE thermogas-dynamic parameter $x_i(t)$ and each time moment *t*, the anomaly indicator at is calculated as:

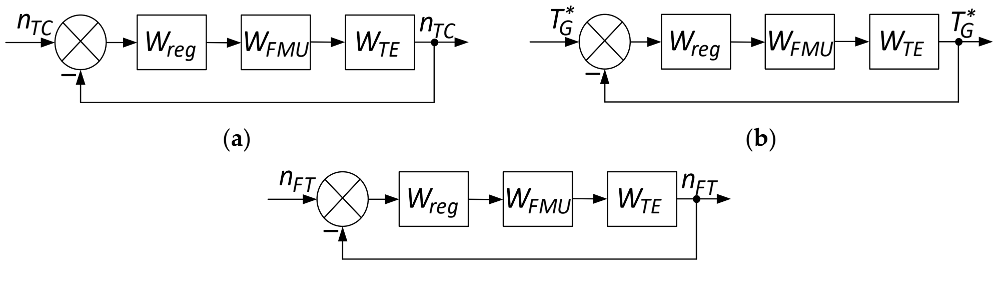
$$a_t = \begin{cases} 1, \; if \; x_i(t) < BL \;\; or \;\; x_i(t) > TL, \\ 0, \; if \qquad\qquad BL \leq x_i(t) \leq TL, \end{cases} \tag{4}$$

Next, the deviation $d_i(t)$ is calculated for each i-th helicopter TE thermogas-dynamic parameter $x_i(t)$ and each time moment t:

$$d_i(t) = \frac{x_i(t) - \mu_i}{\sigma_i}. \tag{5}$$

The value $d_i(t)$ shows how many standard deviations the measured value $x_i(t)$ differs from the average value μ_i. If the helicopter turboshaft engine thermogas-dynamic parameter (n_{TC}, T_G^*, n_{FT}) for any parameter $a_i(t)$ = 1, an alarm is generated about a possible sensor malfunction or deviation in the system operation.

The proposed mathematical model makes it possible to detect sensor data anomalies based on the parameter's statistical characteristics. Using average values and standard deviations to determine the deviations ensures anomaly detection in the reliable boundaries in the helicopter TE sensors' functioning. This, in turn, is closely related to the helicopter TE control loops (Figure 1) [44], since accurate sensor data are critical for the engine control systems' proper functioning. Reliable detection and correction of anomalies ensure the control loops' correct operation, allowing timely response to changes in engine operation and maintaining optimal operating conditions, which increases overall flight safety and efficiency.



(**a**) (**b**)

(**c**)

Figure 1. Diagram of closed loops for regulating helicopter turboshaft engine parameters (W_{reg} is regulator transfer function, W_{FMU} is fuel dispenser model transfer function, W_{TE} is helicopter turboshaft engine model transfer function): (**a**) gas–generator rotor rpm, (**b**) gas temperature in front of the compressor turbine, (**c**) free turbine rotor speed (author's research, based on [44]).

It is worth noting that feedback has been introduced into the control loop (Figure 1), which plays a key role in maintaining the helicopter TE parameters control stability and accuracy. Feedback allows the control system to correct deviations in real time using sensor data. This improves the engine operation reliability and safety, ensuring optimal operating parameters and preventing emergencies.

The helicopter TE parameter regulator in the control loop (Figure 1), connected to the fuel dispenser regulator, acts as a filter. A fuel-metering regulator is present in the control loop to ensure the fuel supply is accurate and has timely regulation, which is necessary to maintain optimal engine operation and quickly respond to changes in operating conditions (Figure 1) [44,45].

During the identification process, the closed control loop changes all elements' parameters in real time (Figure 1). To ensure the desired behaviour, the helicopter TE parameter regulators' dynamic compensation [46] is carried out by replacing them with a similar structure as the regulators configured in the desired way (Figure 2) [44,47,48].

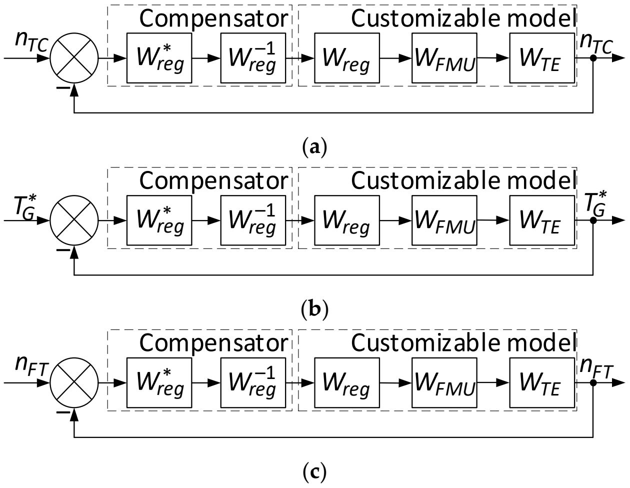


Figure 2. Dynamic compensation diagram in closed loops for regulating helicopter turboshaft engine parameters: (**a**) gas–generator rotor rpm, (**b**) gas temperature in the compressor turbine front, (**c**) free turbine rotor speed (author's research, based on [44,47,48]).

In Figure 2, the compensator consists of the helicopter TE parameter regulator (n_{TC}, T^*_G, n_{FT}) transfer function, with the desired settings W^*_{reg} and a transfer function compensating the helicopter TE parameters regulator (n_{TC}, T^*_G, n_{FT}) W^{-1}_{reg}, and the customized model consists of the transfer functions of the helicopter TE parameters regulator W_{reg}, the fuel-metering regulator W_{FMU}, and the engine model.

The dynamic compensation aim is to tune the system to provide the desired system behaviour over the entire operating range and minimize static control error [49]. Transfer functions are used to describe the system. The closed-loop transfer function with a compensator and tunable model can achieve zero static error and optimal control [50–52].

The desired system behaviour over the entire operating range is achieved by tuning the controllers. The helicopter TE customizable model structure optimises them for a symmetrical operating mode [47,53]. In this case [47,54], zero static errors are ensured. The transfer function has the form [55] for an open-loop system configured for the symmetrical mode:

$$W^{-1}(p) = \frac{4 \cdot T_\mu + 1}{8 \cdot T_\mu^2 \cdot p^2 \cdot (T_\mu \cdot p + 1)}, \tag{6}$$

where T_μ is a small uncompensated time constant, and p is the Laplace operator.

During dynamic compensation, the current controller W_{reg} is replaced by a controller with the desired settings W^*_{reg} with the inverse transfer function W^{-1}_{reg} added to compensating the current controller. The closed-loop system overall transfer function, taking into account dynamic compensation, is then determined as $W_{closet}(p) = W_{FMU}(p) \cdot W_{TE}(p) \cdot C(p)$, where $C(p) = W^*_{reg} \cdot W^{-1}_{reg}$. The Wreg and WFMU controllers are tuned to minimize static errors and provide the required system dynamics to ensure the desired behaviour over the entire operating range.

It is worth considering that various interferences are possible during the helicopter TE operation, which can significantly affect the accuracy of the data received from the sensors. This interference can occur due to external factors, such as electromagnetic fields, vibrations, or sudden changes in the environment, as well as internal factors, including sensor components wear or electronic failures. The interference makes it difficult to correctly read parameters, leading to the engine control circuits' incorrect operation, increasing the risk of errors in the system operation and reducing the overall helicopter operation reliability and safety. Therefore, the work provides filtering and data processing methods that can effectively eliminate or minimize the influence of interference.

In the helicopter TE operation context, where the helicopter TE recorded parameters (n_{TC}, T^*_G, n_{FT}) data from sensors' reliability play a decisive role, it is crucial to consider possible interference that can distort the received signals. To combat this interference, it is effective to use an adaptive interference suppression device, which uses filtering and signal processing methods (Figure 3). This device is highly adaptable to changing operating conditions and helps minimize the interference impact on data quality. Such a device operation involves passing signal components through a reference input, where the signal is compared with a reference value and then corrected or suppressed depending on the deviation degree [56]. This approach allows for effective control and data management, ensuring the helicopter TE control loops reliable operation and increasing overall flight safety.

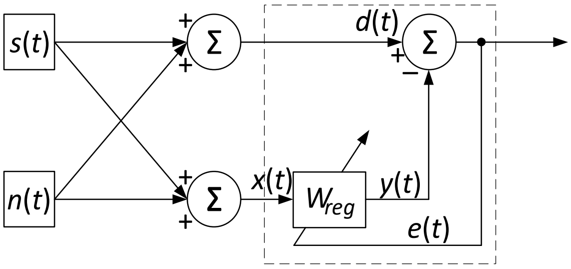


Figure 3. Adaptive device diagram for noise suppression with the helicopter turboshaft engine parameters signal components passage to the reference input (according to B. Widrow and S. Stearns) [56].

Figure 3 shows that $s(t)$ is the useful signal that needs to be restored; $n_0(t)$ is the noise added to the useful signal at the main input; $d(t) = s(t) + n_0(t)$ is the signal at the main input (contaminated with noise); $x(t)$ is the noise correlated with $n_0(t)$ at the reference input (reference input); $y(t)$ is the output of the adaptive filter, which tries to predict $n_0(t)$ based on $x(t)$; and $e(t)$ is the error signal. According to Figure 3, it is assumed that the filter has a nonlinearity, which is represented as a function $f(\bullet)$ acting on a linear filter output with a transfer function $H(p)$. Thus, the filter output $y(t)$ is expressed as:

$$y(t) = f\left(L^{-1}\{H(p)\cdot X(p)\}\right), \tag{7}$$

where $X(p)$ is the input signal $x(t)$ Laplace transform of the helicopter TE-recorded thermogas-dynamic parameters (n_{TC}, T_G^*, n_{FT}), and L^{-1} is the inverse Laplace transform.

The error signal is defined as the difference between the signal at the main input and the nonlinear filter output, that is:

$$e(t) = d(t) - y(t). \tag{8}$$

Nonlinear filter weights' adaptation is described by a generalized version of the LMS algorithm, which takes nonlinearity into account. It is assumed that $\mathbf{w}(t)$ is the nonlinear filter parameters vector, then the parameter update is defined as:

$$\mathbf{w}(t+1) = \mathbf{w}(t) + \mu\cdot\frac{\partial e(t)}{\partial \mathbf{w}(t)}, \tag{9}$$

where $\frac{\partial e(t)}{\partial \mathbf{w}(t)}$ is the error signal gradient concerning the filter parameters and μ is the training rate.

Models (7)–(9) describe the adaptive noise suppression process using a nonlinear filter and transfer functions, which makes it possible to effectively suppress noise and restore the useful signal $s(t)$ under complex nonlinear influences.

Models (7)–(9) effectively suppress interference and extract an adaptive system capable of a useful signal $s(t)$ from the helicopter TE thermogas-dynamic parameters complex nonlinear data characteristic. Equation (7) describes the nonlinear filter output, which consists of two stages: the input signal $X(p)$, transformed into Laplace space, is passed through a linear filter with a transfer function $H(p)$. The linear filtering result $f(\bullet)$ is subjected to a nonlinear transformation using the function $f(\bullet)$. Equation (8) defines the error signal as the difference between the desired signal $d(t)$ and the nonlinear filter output signal $y(t)$. Equation (9) describes the nonlinear filter parameter adaptation algorithm based on the modified LMS algorithm.

Thus, the helicopter TE parameters n_{TC}, T_G^* and n_{FT} regulator play the role of an interference suppression device. Therefore, the dynamic compensation circuit in closed loops for regulating helicopter TE parameters takes the form shown in Figure 4. Thus, the registration sensors system n_{TC}, T_G^* and n_{FT}, respectively, include the helicopter TE parameters in three identical control loops (see Figure 4).

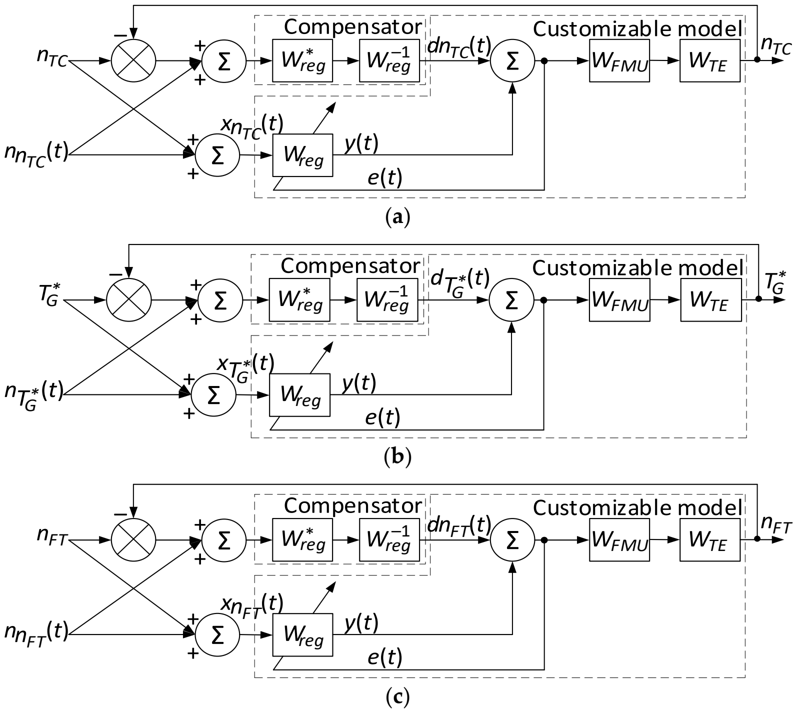


Figure 4. Dynamic compensation diagram in closed loops for regulating the helicopter turboshaft engine parameters with an adaptive noise suppression device with the component signals passage to the reference input: (**a**) gas–generator rotor rpm, (**b**) gas temperature in the compressor turbine front, (**c**) free rotor speed turbines (author's research).

The transition to a unified integration diagram using the filtering method (Figure 5) makes integrating data from three identical control loops for helicopter TE parameters possible, providing more accurate and reliable control.

This approach allows information from various sensors to be combined into one central control unit, where the data are processed using filtering techniques to eliminate noise and improve measurement accuracy.

Thus, the data integration system provides the helicopter TE parameters with more reliable and accurate control, increasing its performance and durability by optimizing the control systems' operation. This approach ensures engine stability under various flight modes and external conditions.

The key to the helicopter TE parameters resulting in the control loop (see Figure 5) is the development of helicopter TE parameter regulators that act as adaptive filters. Using a neural network controller, which acts as a filter, is advisable due to its ability to adapt to diverse and dynamic operating conditions [56]. Neural network filters have the unique ability to train from available data and automatically adjust their operation to changes in input signals [57]. This allows them to effectively consider complex nonlinear relationships between parameters and quickly adapt to new conditions without manual reconfiguring. The beneficial qualities mentioned, such as the ability to adapt to diverse and dynamic conditions and automatically adjust to changes in input signals, are characteristic of neural networks in general. However, achieving these benefits depends on the specific implementation and training of the neural network used in this context.

Thus, the key task is the neural network architecture choice, the neural network structure choice, determining the activation functions rules and the neural network hidden layers number, which ensures the engine operational status monitoring with 1st and 2nd type error probability at a minimum level.

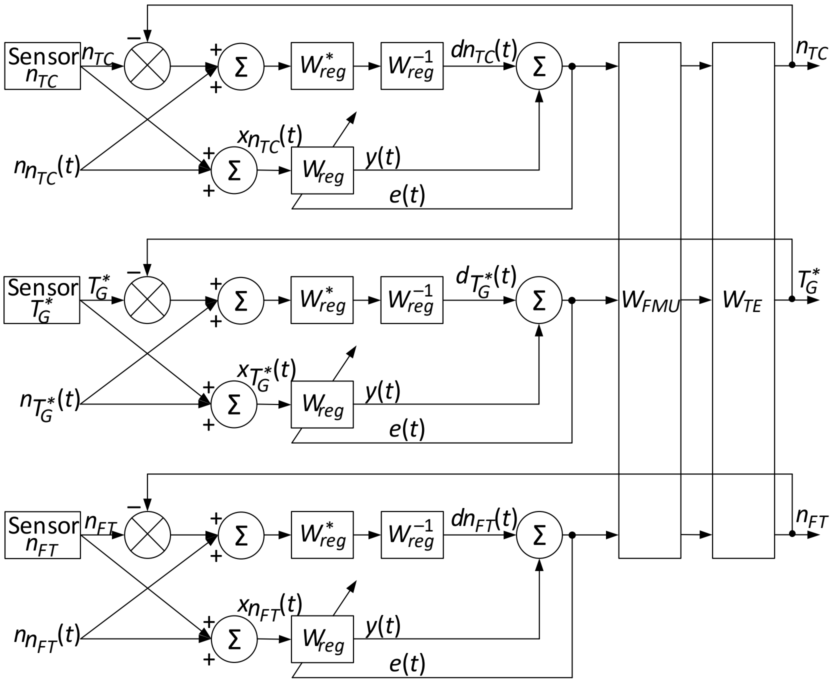


Figure 5. Diagram for integrating closed loops for regulating helicopter turboshaft engine parameters using the filtration method (author's research).

2.2. Neural Network Architecture Development

To solve this task, a multilayer neural network has been developed that processes input data containing useful signals and noise, selects useful signals and provides feedback (Figure 6). The developed neural network input layer contains 6 neurons: 1 is the helicopter TE gas–generator rotor rpm n_{TC} signal, 2 is the helicopter TE gas–generator rotor rpm n_{TC} signal interference $n_{n_{TC}}$, 3 is the helicopter TE gas temperature in front of the compressor turbine T^{*}_{G} signal, 4 is the helicopter TE gas temperature in front of the compressor turbine T^{*}_{G} signal interference $n_{T^{*}_{G}}$, 5 is the helicopter TE free turbine rotor speed n_{FT} signal, and 6 is the helicopter TE free turbine rotor speed n_{FT} signal interference $n_{n_{FT}}$. In the 1st hidden layer, the parameter signals (inputs 1, 3, 5) are summed with their noise (inputs 2, 4, 6). The 2nd hidden layer means dynamic compensation. The following hidden layers filter the resulting signal, which extracts the useful signal. The output layer contains feedback.

The developed neural network input layer does not perform any data transformation but is intended to receive initial data (helicopter TE parameters n_{TC}, T^{*}_{G}, n_{FT}) from sensors, which the network's subsequent layers will process. In this case, 6 input neurons are received, each of input signals, one of which receives x_1 is the helicopter TE gas–generator rotor rpm n_{TC} signal, x_2 is the helicopter TE gas–generator rotor rpm n_{TC} signal interference $n_{n_{TC}}$, x_3 is the helicopter TE gas temperature in the compressor turbine T^{*}_{G} signal front, x_4 is the helicopter TE gas temperature in the compressor turbine T^{*}_{G} signal interference front $n_{T^{*}_{G}}$, x_5 is the helicopter TE free turbine rotor speed n_{FT} signal, and x_6 is the helicopter TE free turbine rotor speed n_{FT} signal interference $n_{n_{FT}}$. Each input neuron receives one of these signals unchanged, that is, $y_i = x_i$.

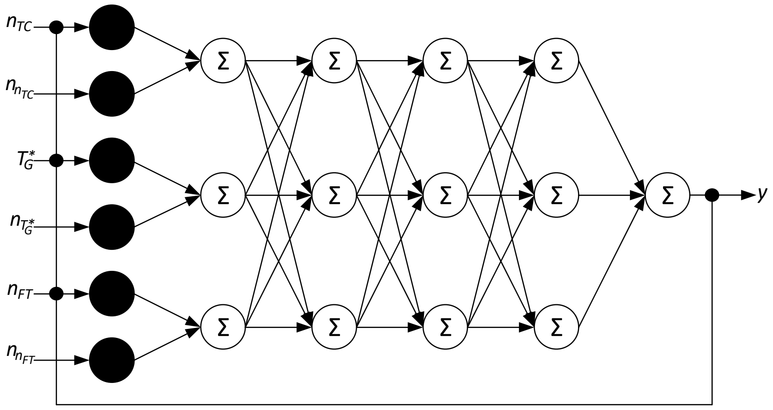


Figure 6. The developed neural network architecture, which implements the closed-loop integration for regulating the helicopter turboshaft engines' parameters using the filtering method (author's research).

The neural network's 1st hidden layer operates by summing the signals with their corresponding noise. Each neuron in this layer processes a signal–noise pair (the signal-noise is formed by adding random noise to the pure signal, which corresponds to the actual helicopter flight conditions). This is necessary to prepare the data for subsequent dynamic compensation and filtering. The combined signals allow the model to be trained more efficiently to remove noise and highlight useful data. Thus,

$$h_1 = x_1 + x_2,\ h_2 = x_3 + x_4,\ h_3 = x_5 + x_6, \tag{10}$$

where h_1, h_2, h_3 are the first hidden layer neurons outputs.

Based on the above, the neural network's 1st hidden layer has three neurons. This layer passes through the neuron without activation since simple summation is required.

The 2nd hidden layer performs the dynamic compensation task. This layer corrects signals to compensate for noise and dynamic changes to improve the data quality before filtering it in subsequent layers. In this layer, neurons are trained to adjust signals taking into account their dynamics and noise. This is achieved by applying trainable weights and activation functions to the summed signals, that is:

$$\begin{aligned} z_1 &= f(w_{11} \cdot h_1 + w_{12} \cdot h_2 + w_{13} \cdot h_3 + b_1), \\ z_2 &= f(w_{21} \cdot h_1 + w_{22} \cdot h_2 + w_{23} \cdot h_3 + b_2), \\ z_3 &= f(w_{31} \cdot h_1 + w_{32} \cdot h_2 + w_{33} \cdot h_3 + b_3), \end{aligned} \tag{11}$$

where z_1, z_2, and z_3 are the second hidden layer neuron outputs, w_{ij} are the weights trained during the network training process, b_i are the biases trained during the network training process, and $f(\bullet)$ is the activation function.

For the 2nd hidden layer, selecting the ReLU (Rectified Linear Unit) activation function is advisable because ReLU effectively copes with dynamic changes in signals and noise due to its ability to pass positive values unchanged and to null out negative ones. This allows neurons to adapt to different levels of input signals and quickly adjust for dynamic changes, which is important for effective noise compensation. In addition, ReLU helps avoid the gradient fading problem, improving the deep networks training and providing more stable and faster model convergence, which is critical for tasks that require accurate dynamic compensation.

Note 1. A critical drawback of the ReLU activation function is the problem of "dying" neurons when input values are negative and the outputs become zero. In this case, neurons stop participating in training since their weight gradient becomes zero. This can cause a significant number of neurons in the network to remain inactive, reducing the model's overall training ability and degrading its performance. Later in the work, this problem will be solved by modifying the ReLU function.

The 3rd hidden layer (Filtering Layer 1) performs the resulting signals after dynamic compensation filtering in the 1st stage. This layer is designed to reduce noise further and improve the desired signal quality. The 3rd hidden layer applies trainable weights and activation functions to the input signals to filter them and extract useful information. Mathematically, the neural network's 3rd hidden layer is represented as follows:

$$\begin{aligned} f_1 &= g(w_{11}{\cdot}z_1 + w_{12}{\cdot}z_2 + w_{13}{\cdot}z_3 + b_1), \\ f_2 &= g(w_{21}{\cdot}z_1 + w_{22}{\cdot}z_2 + w_{23}{\cdot}z_3 + b_2), \\ f_3 &= g(w_{31}{\cdot}z_1 + w_{32}{\cdot}z_2 + w_{33}{\cdot}z_3 + b_3), \end{aligned} \tag{12}$$

where f_1, f_2, and f_3 are the 3rd hidden layer neuron outputs, w_{ij} is the trainable weights applied to the input signals, b_i is the trainable biases, and $g(\bullet)$ is the activation function that helps neurons process input signals nonlinearly, which improves the ability models to isolate useful signals and eliminate noise.

For the 3rd hidden layer, choosing the ReLU activation function is advisable because it can handle noise and extract useful signals efficiently. ReLU allows neurons to only pass positive values through while nulling out negative ones, which helps eliminate unnecessary noise and improves overall signal quality. In addition, ReLU speeds up training by eliminating the gradient decay issues associated with other activation functions and allows the network to better model complex dependencies in data. This makes ReLU ideal for the 1st stage of filtering, effectively extracting useful information from the corrected signals.

The 4th hidden layer represents the resulting signals filtering the 2nd stage, which follows the filtering 1st stage in the 3rd hidden layer. This layer further improves the wanted signal quality by suppressing the remaining noise and making the signal more distinguishable from the background. The 4th hidden layer also applies trainable weights and activation functions to the input signals to further filter the data and improve the desired signal quality. Mathematically, the fourth neural network hidden layer is represented as follows:

$$\begin{aligned} g_1 &= h(w_{11}{\cdot}f_1 + w_{12}{\cdot}f_2 + w_{13}{\cdot}f_3 + b_1), \\ g_2 &= h(w_{21}{\cdot}f_1 + w_{22}{\cdot}f_2 + w_{23}{\cdot}f_3 + b_2), \\ g_3 &= h(w_{31}{\cdot}f_1 + w_{32}{\cdot}f_2 + w_{33}{\cdot}f_3 + b_3), \end{aligned} \tag{13}$$

where g_1, g_2, and g_3 are the 4th hidden layer neuron outputs, w_{ij} is the trainable weights applied to the input signals, b_i is the trainable biases, and $h(\bullet)$ is the activation function.

For the 4th hidden layer, choosing the ReLU activation function is also advisable due to its ability to effectively suppress unnecessary negative values, thereby reducing the noise impact and preserving the signal's positive aspects. This allows the network to more efficiently identify and store useful features present in the data, which is important for the task of integrating and improving signal quality.

Thus, the second, third and fourth hidden layers each have three neurons.

The neural network output layer predicts or classifies the input data according to the given task. In this context, the output layer will predict the system parameters or characteristics based on the input signals after they have been processed and filtered through hidden layers. Given the feedback presence, this layer must also take into account the error received in the previous stages and adjust the network outputs according to this error. For the output neuron, the expression is valid:

$$y = r(v_1{\cdot}g_1 + v_2{\cdot}g_2 + v_3{\cdot}g_3 + c), \tag{14}$$

where y is the helicopter TE parameter (n_{TC}, T_G^* and n_{FT}) predicted output, v_i is the output layer trainable weights, c is the trainable bias (bias), and $r(\bullet)$ is the activation function.

Taking into account feedback, the error E at the network output can be defined as the difference between the predicted output y and the expected output d, that is:

$$E = y - d. \tag{15}$$

The backpropagation algorithm is applied to update the weights v_i and bias c of the output layer. The weights and bias are adjusted in the opposite direction to the gradient of the loss function to these parameters. Thus, updating the weights v_i and bias c of the output layer will occur according to gradient descent as:

$$\Delta v_i = \alpha \cdot \frac{\partial E}{\partial v_i}, \; \Delta c = \alpha \cdot \frac{\partial E}{\partial c}, \tag{16}$$

where α is the adaptive training rate, which includes an adaptive change in the training rate depending on the current gradient and the weight updates history, which, according to the AdaGrad algorithm (Adaptive Gradient Algorithm), is defined as:

$$\alpha_t = \frac{\eta}{\sqrt{G_t + \epsilon}}, \tag{17}$$

where ϵ is a small constant added for numerical stability ($\epsilon \approx 10^{-8}$ is assumed), α_t is the training rate at time t, η is the initial training rate, and G_t is the accumulated squared gradients up to time t, which are updated at each training iteration in the following way:

$$G_t = G_{t-1} + (\nabla E)^2, \tag{18}$$

where ∇E is the loss function gradient over the model parameters.

In a developed neural network where the main task is to predict the parameters or characteristics of the system based on the processed data, the use of a linear activation function at the output layer may be appropriate. This is especially relevant since the output values will be helicopter TE parameters n_{TC}, T_G^*, and n_{FT} continuous numerical values. A linear activation function will allow the neural network to flexibly adapt to different ranges of output parameter values without limiting them to any specific range.

2.3. The ReLU Activation Function Modification

The work proposes to use the innovative activation function Smooth ReLU, developed by this author's team, which is a ReLU function derivative. The main aim of the ReLU function modification is to create a smoother and more continuous activation function to improve the convergence process and training stability. The proposed modification can significantly affect the neural network efficiency, especially in deep training problems where stability and convergence speed play a key role. The expression describes the Smooth ReLU activation function:

$$f(x) = \begin{cases} x, & \text{if } x > 0, \\ \frac{1}{1+e^{-\gamma \cdot x}}, & \text{if } x \leq 0, \end{cases} \tag{19}$$

where γ is a parameter that determines the function "level of smoothness". For $x > 0$, the function behaves similarly to a regular ReLU, and for $x \leq 0$ it smoothly transitions to negative values using a sigmoid function. This avoids sudden gradient changes and can speed up neural network training. The proposed Smooth ReLU activation function retains the ReLU benefits, such as no gradient for positive values, while adding smoothness for negative values. This can sometimes improve training, allowing for more stable and faster convergence.

Theorem 1. *The Smooth ReLU function is continuous over the entire domain of definition.*

Proof of Theorem 1. Let $f(x)$ be the Smooth ReLU activation function defined according to (18). The Smooth ReLU activation function is continuous for $x > 0$ and $x < 0$, since for $x > 0$ the function $f(x) = x$ is a linear function that is continuous over the definition of the entire domain, and for $x < 0$ $f(x) = \frac{1}{1+e^{-\gamma \cdot x}}$, which means a modified sigmoid function is continuous throughout the entire domain of the definition.

To prove continuity at $x = 0$, it is necessary to show that the limit of $f(x)$ as x tends to zero from the left is equal to the limit of $f(x)$ as x tends to zero from the right, and that this limit is equal to the value of the function at $x = 0$. Consider the limit:

$$\lim_{x\to 0^-} f(x) = \lim_{x\to 0^-} \frac{1}{1+e^{-\gamma\cdot x}}, \tag{20}$$

in which as x $\to 0^-$, $\gamma\cdot x \to 0$, $e^{-\gamma\cdot x} = 1$, thus:

$$\lim_{x\to 0^-} \frac{1}{1+e^{-\gamma\cdot x}} = \frac{1}{1+1} = \frac{1}{2}. \tag{21}$$

Consider the limit:

$$\lim_{x\to 0^+} f(x) = \lim_{x\to 0^+} x = 0. \tag{22}$$

Since the different sides' limits do not coincide, checking the function value at point x = 0 is necessary. Let us take the value $x \neq 0$ at point $x = 0$, for example, $f(0) = \frac{1}{2}$. Next, continuity is revised with the set value $f(0) = \frac{1}{2}$, that is:

$$\lim_{x\to 0^-} f(x) = \frac{1}{2}, \lim_{x\to 0^+} f(x) = 0 \neq \frac{1}{2}. \tag{23}$$

Thus, the original function was not continuous at $x = 0$ with the given condition, and an adjustment is required to determine continuity correctly. Then, the Smooth ReLU function is described by the expression:

$$f(x) = \begin{cases} x, & \text{if } x > 0, \\ 0 & \text{if } x = 0, \\ \frac{1}{1+e^{-\gamma\cdot x}}, & \text{if } x < 0. \end{cases} \tag{24}$$

This function, similar to (19), is continuous for $x > 0$ and $x < 0$, and continuity at point $x = 0$ is defined as:

$$\lim_{x\to 0^-} f(x) = \lim_{x\to 0^-} \frac{1}{1+e^{-\gamma\cdot x}} = \frac{1}{2}, \lim_{x\to 0^+} f(x) = 0. \tag{25}$$

And its value at point $x = 0$ is zero.

Thus, the adjustment assumes that the function is not smooth at point 0, but remains continuous throughout the definition of the entire domain. The theorem is proven: the function $f(x)$, defined with correction in the form (24), is continuous over the entire domain of the definition. □

To research the neuron's activation functions, it is imperative to analyze their derivatives. The activation function derivative allows us to estimate this function change rate in response to changes in input data. In turn, the updating neuron weights process helps optimize during neural network training. The traditional neuron activation function ReLU $f(x) = \max(0, x)$ derivative (Figure 7a) has the form:

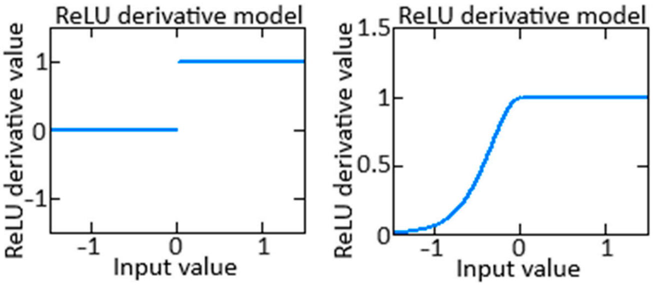
$$f\prime(x) = \begin{cases} 1, \text{ if } x > 0, \\ 0, \text{ if } x \leq 0, \end{cases} \tag{26}$$

The proposed Smooth ReLU neuron activation function with correction (24) derivative (Figure 7b) has the form:

$$f\prime(x) = \begin{cases} 1, & \text{if } x > 0, \\ 0, & \text{if } x = 0, \\ \frac{\gamma\cdot e^{-\gamma\cdot x}}{(1+e^{-\gamma\cdot x})^2}, & \text{if } x < 0. \end{cases} \tag{27}$$

As can be seen from (26), (27), and Figure 7, the problem with the traditional ReLU function $f(x)$ = max(0, x) is that its derivative is zero for all $x \leq 0$. This can lead to “dead neurons” in the neural network when neurons stop updating due to the lack of gradient. The advantage of Smooth ReLU is that it always has a non-zero gradient for all values of x, including negative ones (except for x = 0). This avoids the problem of “dead neurons” and ensures more stable neural network training.

Thus, the proposed Smooth ReLU use with adjustment (24) is mathematically justified since it provides a smooth and continuous gradient throughout the definition domain, which can help improve the convergence and training efficiency of the model.



(**a**) (**b**)

Figure 7. Derivative ReLU functions diagrams: (**a**) traditional ReLU max(0, x); (**b**) proposed Smooth ReLU with adjustment (22) (author’s research).

2.4. A Neural Network Training Algorithm Development

The work proposes an algorithm for training a neural network (Figure 6), consisting of the following stages:

1. Weights initialization is initially setting the neural network weight values before starting training. Proper weight initialization is important for efficient and fast network training, as it helps avoid convergence problems and helps achieve more accurate results.

For the developed neural network that uses Smooth ReLU in the hidden layers, it is appropriate to use the He initialization method [58]. The 1st hidden layer weight initialization is not required since this layer performs simple summation. The neural network remaining layers’ weight initialization is carried out as follows:

$$W \sim N\left(0, \frac{2}{n_i}\right), \tag{28}$$

where $N(\mu, \sigma^2)$ is a normal distribution with mean μ and variance σ^2, and n_i is the neuron number in the neural network i-th layer.

2. Forward propagation is passing the neural network input data through all layers to the output to obtain predictions, in which the input signals’ weighted sums are calculated at each layer, to which an activation function is then applied. This process ensures that the input data are transformed into network output values. Direct propagation is carried out according to (10)–(15).

3. Backpropagation is a neural network training to minimize prediction error by propagating the network output back error calculated through the layers, updating the network’s weights and biases. The main purpose of backpropagation is to adjust the weights in such a way as to reduce the predicted and actual values difference between them. For each layer, the error function gradient concerning its weights and biases is calculated as:

$$\delta_{out} = \frac{\partial E}{\partial Y} = y - d,\ \frac{\partial E}{\partial v_i} = \delta_{out} \cdot g_i,\ \frac{\partial E}{\partial c} = \delta_{out}. \tag{29}$$

For the remaining layers (1st–4th hidden layers) of the neural network, similar calculations are given in Table 1.

Table 1. Analytical expressions for the developed neural network hidden layers parameters calculating (author's research).

Neural Network Layer	Parameter	Analytical Expression
4th hidden layer	Neuron weight error	$\delta_{g_i} = \delta_{out} \cdot w_{ij}^{(4)}$, where $w_{ij}^{(4)}$ mean the 4th hidden layer of neurons' weights (see expression (13)).
	Gradient error	$\delta_{g_i} = \delta_{g_i} \cdot \frac{\partial(\text{Smooth ReLU})}{\partial x}$, where $\frac{\partial(\text{Smooth ReLU})}{\partial x}$ is defined according to (27).
	Gradients by weights	$\frac{\partial E}{\partial w_{ij}^{(4)}} = \delta_{g_i} \cdot f_i$
3rd hidden layer	Neuron weight error	$\delta_{f_i} = \sum \delta_{f_i} \cdot w_{ij}^{(3)}$, where $w_{ij}^{(3)}$ mean the 3rd hidden layer neurons weights (see expression (12)).
	Gradient error	$\delta_{f_i} = \delta_{f_i} \cdot \frac{\partial(\text{Smooth ReLU})}{\partial x}$, where $\frac{\partial(\text{Smooth ReLU})}{\partial x}$ is defined according to (27).
	Gradients by weights	$\frac{\partial E}{\partial w_{ij}^{(3)}} = \delta_{f_i} \cdot z_i$
2nd hidden layer	Neuron weight error	$\delta_{z_i} = \sum \delta_{z_i} \cdot w_{ij}^{(2)}$, where $w_{ij}^{(2)}$ mean the 3rd hidden layer of neurons' weights (see expression (11)).
	Gradient error	$\delta_{z_i} = \delta_{z_i} \cdot \frac{\partial(\text{Smooth ReLU})}{\partial x}$, where $\frac{\partial(\text{Smooth ReLU})}{\partial x}$ is defined according to (27).
	Gradients by weights	$\frac{\partial E}{\partial w_{ij}^{(2)}} = \delta_{z_i} \cdot h_i$.

4. Updating weights and biases is the neural network's weights and biases changing process over time to the loss function minimization. The squared gradients accumulation is defined as:

$$G_{w_{ij}^{(m)}} = G_{w_{ij}^{(m)}} + \left(\frac{\partial E}{\partial w_{ij}^{(m)}}\right)^2, \tag{30}$$

where m = 2. . .4 is the neural network hidden layers number (2nd, 3rd and 4th hidden layers, respectively).

Weights and biases are updated according to the expressions:

$$w_{ij}^{(m)} = w_{ij}^{(m)} - \frac{\eta}{\sqrt{G_{w_{ij}^{(m)}} + \epsilon}} \cdot \frac{\partial E}{\partial w_{ij}^{(m)}}, \tag{31}$$

$$c = c - \frac{\eta}{\sqrt{G_c + \epsilon}} \cdot \frac{\partial E}{\partial c}. \tag{32}$$

The weights and biases forward propagation, backpropagation, and updating are repeated for all training examples and throughout all training epochs until the convergence criterion is reached.

For the task posed in the work of filtering interference and integrating signals from the helicopter TE thermogas-dynamic parameter sensors using the developed neural network (Figure 6), which implements a diagram for the regulating helicopter TE parameters integrating closed loops using the filtering method (Figure 5), the stopping criterion is a balanced criterion for improving accuracy on the validation set and controlling the neural network retraining.

To create a balanced convergence criterion that takes into account both the improvement in accuracy on the validation set and control for overfitting, a weighted average between the two criteria can be used:

$$C = \alpha \cdot \varepsilon + (1 - \alpha) \cdot O, \tag{33}$$

where ε is the accuracy (increase in model accuracy on the validation set (for example, the percentage of correct predictions), O is the measure of model retraining control (ratio of error on the validation set to error on the training set), and $0 \leq \alpha \leq 1$ is the coefficient reflecting the increasing accuracy importance compared to the overfitting control.

Note 2. An appropriate value for α can be chosen depending on the specific task requirements and preferences. For example, if increasing accuracy is more important than controlling for overfitting, then α might be chosen closer to 1. If controlling for overfitting is more important, then α might be chosen closer to 0.

The scientific novelty of the proposed neural network training algorithm lies in the integration of two convergence criteria: increasing accuracy on the validation set and controlling overfitting into a single criterion that allows for balancing, improving model accuracy, and preventing overfitting. The difference from the traditional backpropagation algorithm is that the proposed method not only minimizes the loss function but also takes into account the error dynamics on the validation sample, which allows for maintaining the model generalization ability when achieving a certain level of accuracy. Additionally, the adaptive training rate complements this approach by allowing the model to efficiently and quickly adapt to changes in data and training conditions, which improves its performance in practice.

3. Case Study

3.1. Analysis and Preliminary Processing Results for Initial Signals from Helicopter TE Thermogas-Dynamic Parameter Sensors

To conduct a computational experiment, data were obtained on the TV3-117 TE thermogas-dynamic parameters [59–61], recorded on board the Mi-8MTV helicopter during flight: gas–generator rotor rpm n_{TC} at times $t_1 \ldots t_n$; gas temperature in front of the compressor turbine T_G^* at times $t_1 \ldots t_n$; and free turbine rotor speed n_{FT} at times $t_1 \ldots t_n$. The n_{TC}, T_G^*, and n_{FT} values are given in absolute values (Tables 2–4).

Table 2. The training sample fragment for n_{TC} gas–generator rotor rpm (author's research).

Number	**1**	**2**	…	**37**	…	**84**	…	**115**	…	**172**	…	**202**	…	**256**
Value	0.943	0.982	…	0.948	…	0.957	…	0.962	…	0.974	…	0.935	…	0.981

Table 3. The training sample fragment for T_G^* gas temperature in the compressor turbine front (author's research).

Number	**1**	**2**	…	**29**	…	**73**	…	**109**	…	**164**	…	**200**	…	**256**
Value	0.932	0.964	…	0.975	…	0.926	…	0.918	…	0.905	…	0.902	…	0.953

Table 4. The training sample fragment for n_{FT} free turbine rotor speed (author's research).

Number	**1**	**2**	…	**32**	…	**80**	…	**105**	…	**181**	…	**207**	…	**256**
Value	0.929	0.933	…	0.909	…	0.932	…	0.941	…	0.955	…	0.926	…	0.973

At the helicopter TE thermogas-dynamic parameter (n_{TC}, T_G^* and n_{FT}) values preliminary processing in the 1st stage, the homogeneity of the training samples is assessed

(Tables 2–4). According to [59–61], the criterion by which the training sample homogeneity is determined is the Fisher–Pearson criterion, which is defined as [62]:

$$\chi^2 = \frac{\sqrt{N \cdot (N-1)}}{N-2} \cdot \frac{\frac{1}{N} \cdot \sum_{i=1}^{N} (x_i - \overline{x})^3}{\left(\frac{1}{N} \cdot \sum_{i=1}^{N} (x_i - \overline{x})^2\right)^{\frac{3}{2}}}, \tag{34}$$

where N = 256 is the training sample size, x_i is the training sample (Tables 2–4) i-th element value, and $\overline{x} = \frac{1}{n} \cdot \sum_{i=1}^{n} x_i$ is the training sample average value.

The significance level adopted in the work is 0.01, which means the probability of introducing a type I error (erroneously rejecting the true null hypothesis) is 1%. That is, while a statistical test indicates a significant result, there is only a 1% chance that the result is due to chance and is due to noise or random variations in the data. This significance indicates strict requirements for the results' reliability, which is especially important in the context of helicopter TE thermogas-dynamic parameter sensor readings' accuracy and reliability. The freedom degrees number is 1 (one type of parameter in each training sample: n_{TC}, T_G^* or n_{FT}). Thus, the Fisher–Pearson test's critical value for one degree of freedom at a significance level of 0.01 was 6.6. The Fisher–Pearson criterion obtained values are $\chi^2_{n_{TC}} = 4.727$, $\chi^2_{T_G^*} = 4.645$, and $\chi^2_{n_{FT}} = 5.619$ are less than the critical value $\chi^2_{critical} = 6.6$, which indicates the helicopter TE thermogas-dynamic parameter training sample homogeneity.

To confirm the training sample (see Tables 2–4) homogeneity assessing results using the Fisher–Pearson criterion according to [59–61], an identical experiment was carried out using the Fisher–Snedecor criterion, which is defined as [63]:

$$F = \frac{S_1^2}{S_2^2} = \frac{\frac{1}{n_1 - 1} \cdot \sum_{i=1}^{n_1} \left(x_i^{(1)} - \overline{x}^{(1)}\right)^2}{\frac{1}{n_2 - 1} \cdot \sum_{i=1}^{n_2} \left(x_i^{(2)} - \overline{x}^{(2)}\right)^2}, \tag{35}$$

where n_1 and n_2 are the 1st and 2nd training sample sizes, $x_i^{(1)}$ and $x_i^{(2)}$ are the i-th element of the 1st and 2nd training samples, respectively, and $\overline{x}^{(1)}$ and $\overline{x}^{(2)}$ are the average values of the 1st and 2nd training samples, respectively.

To calculate the Fisher–Snedecor criterion according to (35), the helicopter TE thermogas-dynamic parameter (see Tables 2–4) value training samples, consisting of 256 elements each, are randomly divided into two equal samples of 128 elements each, that is, n_1 = n_2 = 128. The significance level for the Fisher–Snedecor criterion is also accepted as 0.01, and the freedom degrees number is 1, which indicates the helicopter TE parameters one type in each training sample: n_{TC}, T_G^* or n_{FT}. Thus, the Fisher–Snedecor test critical value for one freedom degree at a significance level of 0.01 was 6.6. The Fisher–Snedecor criterion obtained values are $F_{n_{TC}} = 4.727$, $F_{T_G^*} = 4.645$, and $F_{n_{FT}} = 5.619$ are less than the critical value $F_{critical} = 6.6$, which confirms the helicopter TE thermogas-dynamic parameter training sample homogeneity.

At the helicopter TE thermogas-dynamic parameter (n_{TC}, T_G^* and n_{FT}) values preliminary processing in the 1st stage according to [59–61], the training and test samples representativeness is assessed using cluster analysis, the which aim is to separate the input data set X = {x_1, x_2, …, x_n} (see Tables 2–4) into k disjoint clusters, where k is a predetermined number of clusters (k = 8 is assumed based on [59–61]). Each cluster is an object group that is considered more similar than objects from other clusters. Taking into account the results of [64], the k-means cluster analysis method was applied, based on minimizing the squared distances sum between cluster objects and their centroids (the j-th cluster centre) C = {μ_1, μ_2, …, μ_n}, where $\mu_j \in R^d$.

For each x_i value of helicopter thermogas-dynamic parameters TE n_{TC}, T_G^* and n_{FT}, the distance to all centroids is calculated and the object is assigned to the cluster with the nearest centroid according to the expression:

$$r_{ij} = \begin{cases} 1, \text{ if } j = \text{argmin}_l \|x_i - \mu_l\|^2, \\ \quad 0, \text{ otherwise} \end{cases} \tag{36}$$

where $j = \text{argmin}_l \|x_i - \mu_l\|^2$ means that object x_i belongs to the j-th cluster, with $r_{ij} \in \{0, 1\}$.

The k-means method minimizes the squared distance sum between features and their corresponding centroids. The objective function is presented as:

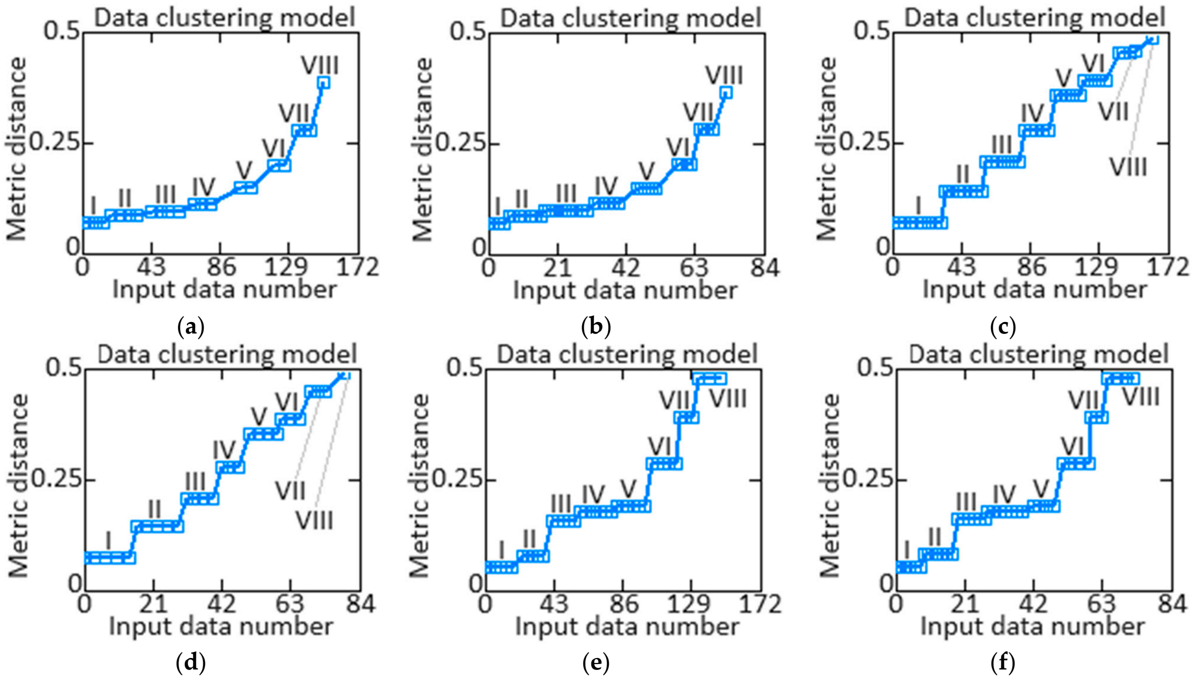
$$J = \sum_{j=1}^{k} \sum_{i=1}^{N} r_{ij} \cdot \|x_i - \mu_j\|^2. \tag{37}$$

For each j-th cluster, the centroid μ_j is updated as all objects assigned to that cluster average as:

$$\mu_j = \frac{\sum_{i=1}^{N} r_{ij} \cdot x_i}{\sum_{i=1}^{N} r_{ij}}. \tag{38}$$

Calculations according to (36) and (38) are repeated until the object assignments to clusters stop changing (until convergence).

A random selection procedure was used to select training and test samples in a 2:1 ratio (67 and 33%, respectively—172 and 84 elements) based on the helicopter TE (see Tables 2–4) thermogas-dynamic parameter training samples. The cluster data analysis results from the helicopter TE thermogas-dynamic parameter (Tables 2–4) training samples revealed 8 classes (classes I. . .VIII), that is, eight groups are present in them, which indicates the composition similarity of both training and test samples (Figure 8).



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 8. Cluster analysis results: (**a**) training sample of the parameter n_{TC}, (**b**) test sample of the parameter n_{TC}, (**c**) training sample of the parameter T_G^*, (**d**) test sample of the parameter T_G^*, (**e**) training sample of the parameter n_{FT}, (**f**) test sample of the n_{FT} parameter (author's research).

The results obtained made it possible to determine the helicopter TE thermogas-dynamic parameters' optimal sample size: the training sample is 256 elements (100%), the control sample is 172 elements (67% of the training sample), and the test sample is 84 elements (33% of the training sample).

3.2. The Developed Neural Network Training Results

At the developed neural network (see Figure 6) training in the 1st stage, the epochs passed number (Figure 9, Table 5) which influence the final standard deviation that is the training (loss function) quality assessing criterion and is defined as [59–61]:

$$E_{epoch} = \frac{1}{N} \cdot \sum_{i=1}^{N} \left(\frac{1}{2} \cdot \sum_{k=1}^{n} (y_k - \hat{y}_k)^2 \right). \tag{39}$$

Table 5. Determining the influence results for the number of epochs passed on the resulting error (author's research).

Number	1	2	3	4	5	6	7	8	9
Epoch	0	40	80	120	160	200	240	280	320
E_{epoch}	17.352	14.018	10.342	8.665	5.229	4.315	3.399	2.005	3.767

The results obtained indicate that 280 training epochs are sufficient to achieve the minimum value of E_{epoch} = 2.005. (Figure 9a). It is worth noting that after 280 epochs, the neural network training error increases. The neural network training convergence demonstrated it was trained for 1000 epochs (Figure 9b). It can be seen that almost immediately after 320 training epochs, the loss function decreases to its minimum value E_{epoch} = 2.005 and remains stable over 1000 training epochs. The slight increase in training error after epoch 280 is due to the overfitting phenomenon, where the model begins to overfit the training data rather than generalize to the new data. However, after epoch 330, the error is reduced again due to hyperparameter adjustments using regularization techniques that stabilize training.

The situation described above, where a slight increase in the training error is observed after the *i*-th epoch due to model overtraining, is discussed in detail in [65,66]. These sources provide various methods for reaching this state, including adjusting hyperparameters and using regularization methods. These methods aim to stabilize the training process and improve the model's generalisation ability. It is important to note that the temporary increase in training error is not critical and can be effectively managed using appropriate techniques, allowing the model to continue training and achieve optimal results.

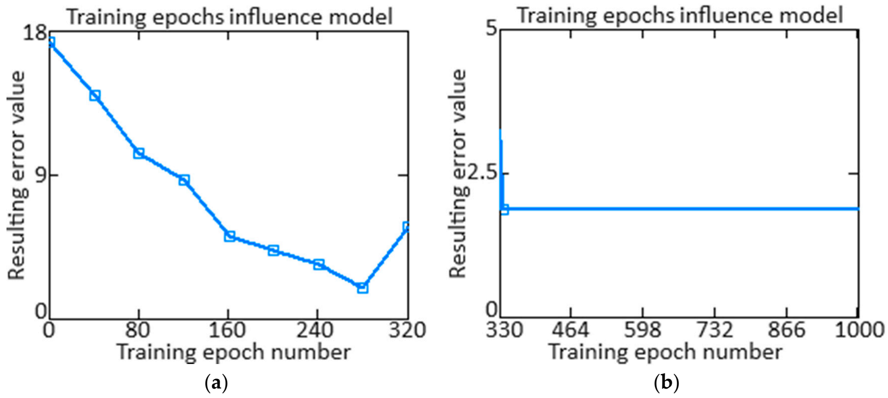


Figure 9. The influence diagram for the number of epochs passed on the resulting error (author's research). (**a**) Training for the 320 epochs (**b**) Training from 320 to 1000 epochs.

At the next stage of training of the developed neural network (see Figure 6), its performance, accuracy (Figure 10) and loss (Figure 11) are determined for 280 training epochs. Accuracy provides information about how well the model classifies the data, while loss reflects how well the model minimises the difference between predicted and actual values.

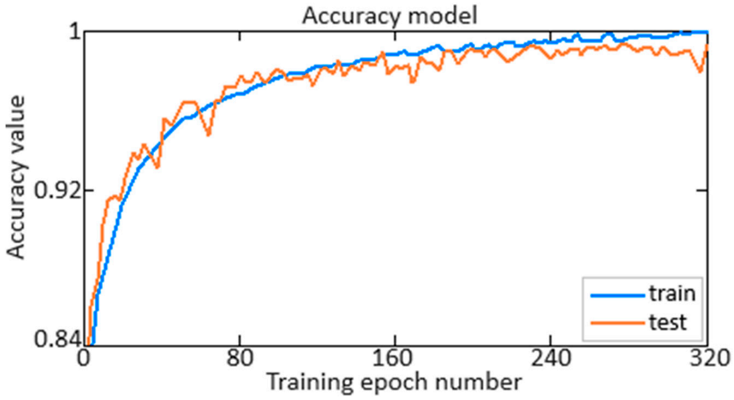


Figure 10. Accuracy metric diagram (author's research).

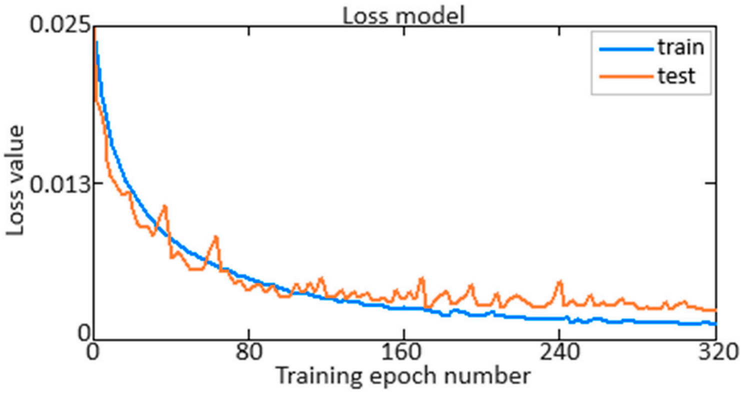


Figure 11. Loss function diagram (author's research).

As can be seen from Figure 10, accuracy reaches a value of 0.995 (almost 100%). Moreover, as can be seen from Figure 11, the loss function does not exceed 0.025 (2.5%) at the beginning of training and decreases to 0.005 (0.5%) at 280 training epochs. This indicates the high efficiency and accuracy of the developed neural network in solving the problem. Achieving an accuracy value of 0.995 (almost 100%) indicates that the model successfully classifies the data with high accuracy. In parallel, the observed decrease in the loss function from 0.025 (2.5%) to 0.005 (0.5%) indicates that the model successfully reduces the difference between the actual and predicted values, which is a key indicator of the effectiveness of neural network training. These results highlight the high-quality performance of the model and its ability to make accurate and reliable predictions.

It is worth noting that the work compares the accuracy and loss definition. As mentioned above, accuracy reaches a value of 0.995, and the loss function decreases from 0.025 (2.5%) to 0.005 (0.5%) after 280 training epochs when applying the Smooth ReLU activation function. At the same time, when using the ReLU activation function, accuracy reaches a 0.995 value when the loss function is reduced from 0.025 (2.5%) to 0.005 (0.5%) and is achieved with 490 training epochs, which is almost several times further compared to using Smooth ReLU. Moreover, with 280 training epochs, when using the ReLU activation function, accuracy only reaches a 0.972 value, and the loss function decreases from 0.025 (2.5%) to 0.018 (1.8%).

3.3. Helicopter Turboshaft Engines Thermogas-Dynamic Parameter Sensors Signal Neural Network Integration Results

To conduct a computational experiment based on helicopter TE thermogas-dynamic parameter (see Tables 2–4) training samples, as an example, the n_{TC} parameter original signal, contaminated with noise, that was received from the corresponding sensor (Figure 12) was restored.

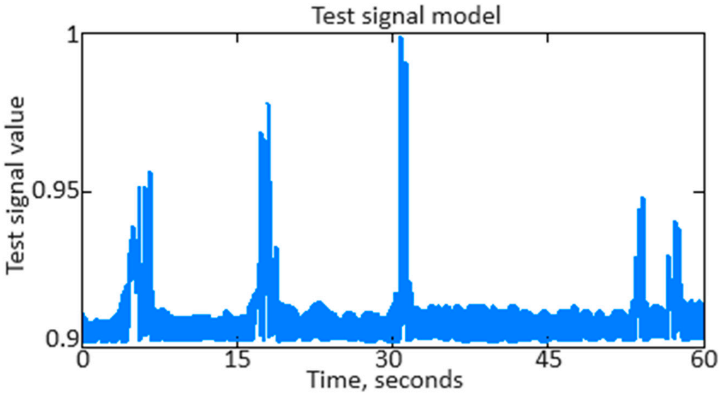


Figure 12. Initial diagram of the n_{TC} gas–generator rotor rpm signal (author's research).

As can be seen from Figure 12, the original nTC parameter signal contains various distortions, interference, and noise, which can affect its accuracy in analysis and interpretation. Figure 13 shows the n_{TC} parameter filtered signal after applying the developed neural network (see Figure 6).

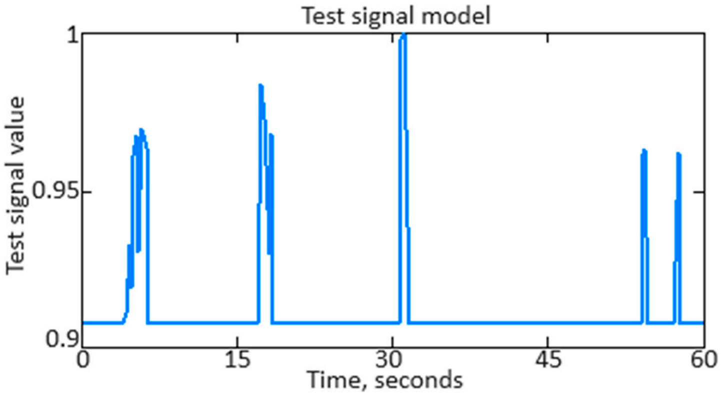


Figure 13. Resulting diagram of the n_{TC} gas–generator rotor rpm signal (author's research).

As shown in Figure 13, the n_{TC} parameter signal filtering effectively removed noise and distortion while the signal's main characteristics were preserved. The filtered n_{TC} signal appears cleaner and smoother, making it more suitable for further analysis and use.

To analyze the signal frequency composition, the n_{TC} parameter is used to isolate or suppress certain frequency components using the direct Fourier transform $F(\omega) = \int_{-\infty}^{\infty} f(t) \cdot e^{-j \cdot \omega \cdot t} dt$ the transition from the time domain to the frequency domain has been completed. This allows you to determine which frequencies are present in the signal and at what amplitude, which in turn helps determine which frequencies should be retained or excluded to achieve the desired filtering result. Moving into the frequency domain allows you to understand the signal structure better and make informed decisions about the necessary filtering actions, such as noise reduction, extraction of frequency components of interest, or interference suppression. Figure 14a shows the n_{TC} parameter's original signal spectrum, and Figure 14b shows the n_{TC} parameter's filtered signal spectrum.

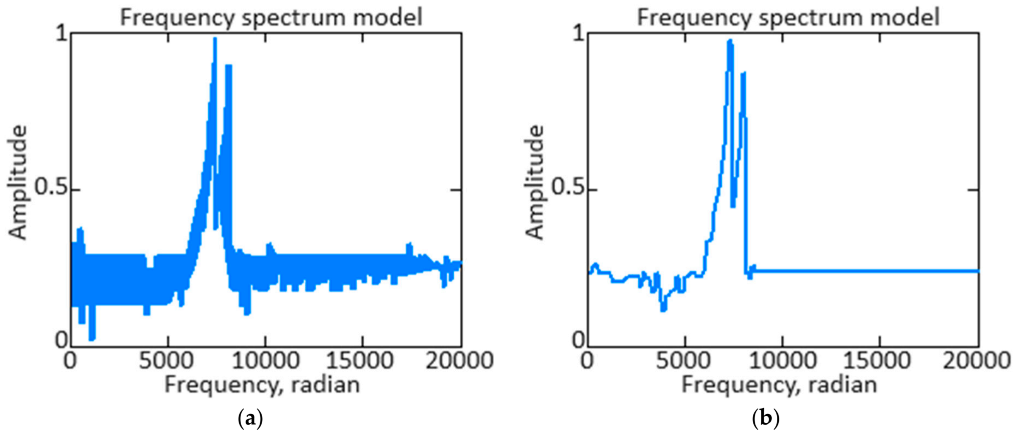


Figure 14. The n_{TC} gas–generator rotor rpm signal spectrum diagram: (**a**) Original signal (**b**) Filtered signal (author's research).

The n_{TC} parameter is the original and filtered signals spectrum resulting diagram, allowing us to compare their frequency characteristics visually. Analyzing the differences between the spectra allows you to determine the filtration effectiveness and evaluate how successfully your aims were achieved. If the filtered signal spectrum shows a significant reduction in amplitude at frequencies that need to be suppressed and retention or enhanced that are important to the signal, this indicates the designed neural network is (Figure 6) functioning well.

Further in the work, the parameter n_{TC} signal repetition period is determined as $T = \frac{1}{f}$, where $f = \frac{2\pi}{\omega}$ is the frequency. This allows you to understand the n_{TC} parameter signal time's main characteristics, such as its frequency and frequency spectrum. Knowing the signal period helps analyze its dynamics and detect regular patterns. Figure 15a shows the parameter n_{TC} initial signal repetition period, and Figure 15b shows the n_{TC} parameter filtered signal repetition period.

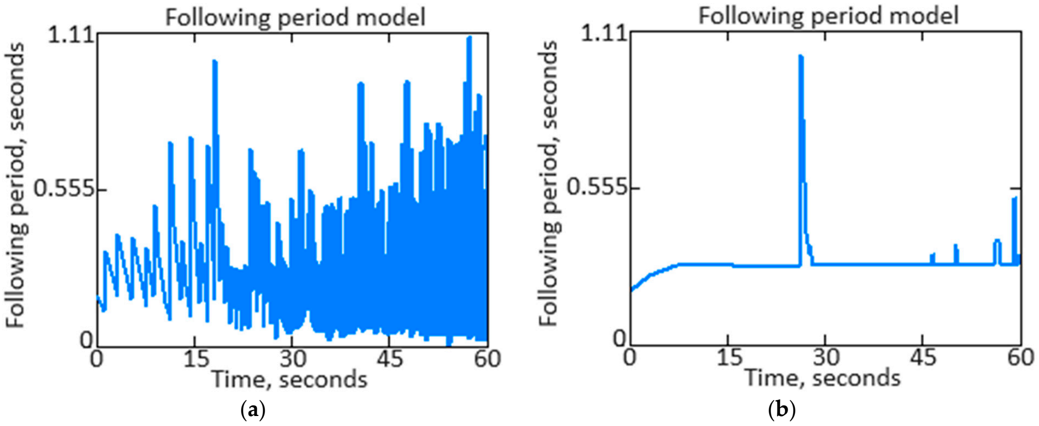


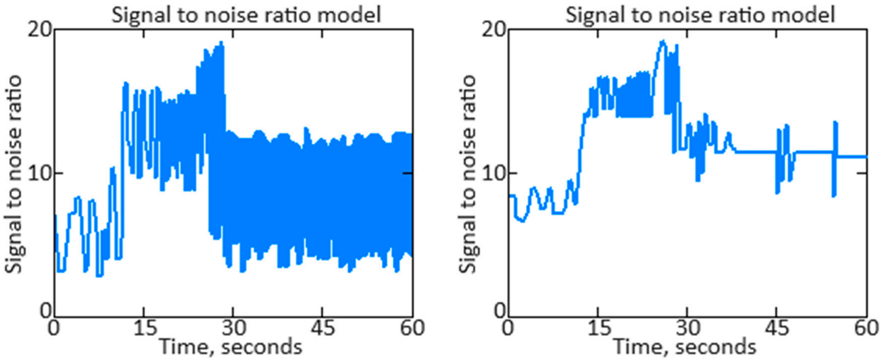
Figure 15. The n_{TC} gas–generator rotor rpm signal repetition period diagram: (**a**) Original signal (**b**) Filtered signal (author's research).

At the computational experiment next stage, a transition was made from the parameter signal n_{TC}, presented in the time dependence $f(t)$ form (see Figure 12), to the signal-to-noise ratio (SNR) according to the expression:

$$SNR = \frac{\int_0^T (f(t))^2}{\int_0^T (n(t))^2}, \tag{40}$$

where $n(t)$ is the noise signal (in this work it is taken as a random variable).

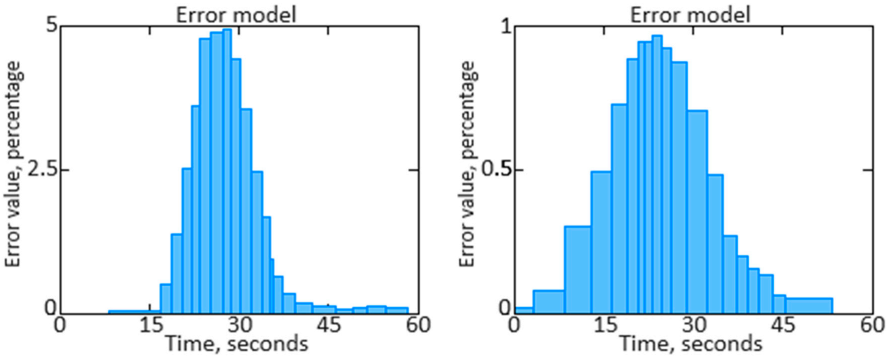
SNR provides a clear numerical value that shows how much the signal stands out from the background noise. A high *SNR* indicates that the signal dominates the noise, which means better data transmission and processing. A low *SNR* indicates that noise is having a significant impact, which can result in distortion or loss of important information. Figure 16a shows the *SNR* based on the original signal of the n_{TC} parameter, and Figure 16b shows the *SNR* based on the filtered signal of the n_{TC} parameter.



(**a**) (**b**)

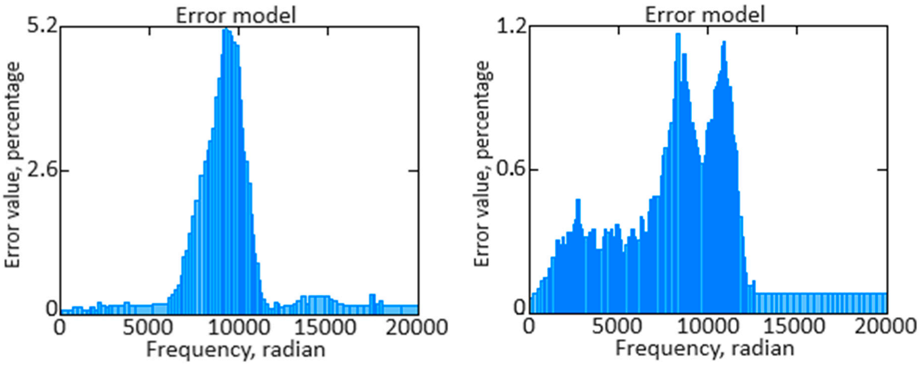
Figure 16. The n_{TC} gas–generator rotor rpm signal signal-to-noise ratio diagram: (**a**) Original signal (**b**) Filtered signal (author's research).

We have performed a comparison of filtered and unfiltered results histograms along with the n_{TC} parameter signal filtered estimates dynamics qualitative analysis (Figures 17–20).



(**a**) (**b**)

Figure 17. Signal histogram for the n_{TC} gas–generator rotor rpm estimates: (**a**) Original signal (**b**) Filtered signal (author's research).



(**a**) (**b**)

Figure 18. The spectrum histogram for the n_{TC} gas–generator rotor rpm signal estimates: (**a**) Original signal (**b**) Filtered signal (author's research).

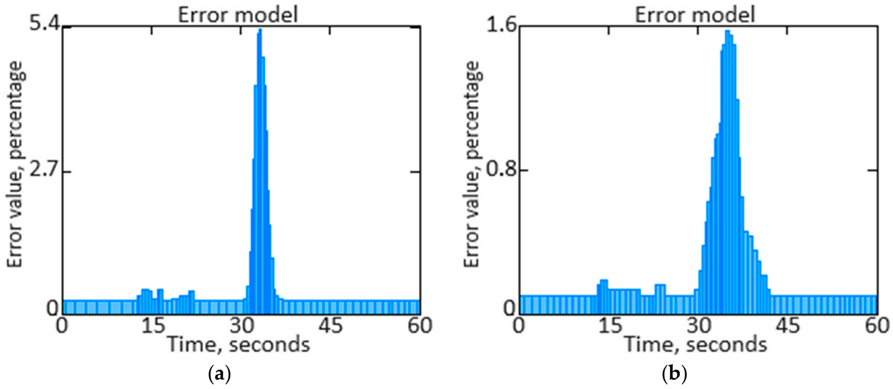


Figure 19. The sequence histogram for the gas–generator rotor rpm signal n_{TC} estimates: (**a**) Original signal (**b**) Filtered signal (author's research).

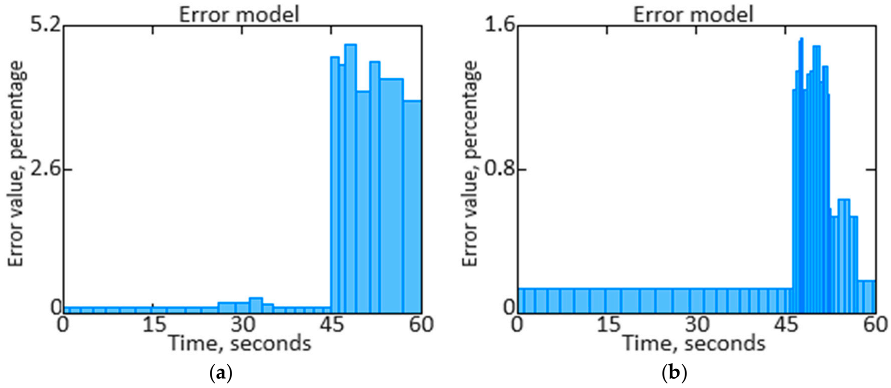


Figure 20. The n_{TC} gas–generator rotor rpm signal signal/noise estimates histogram: (**a**) Original signal (**b**) Filtered signal (author's research).

The obtained data analysis (see Figures 17–20) allows us to state that combining signals using the filtering method leads to narrowing the unfiltered estimate histogram, which, in turn, helps to increase accuracy. The presented histogram research shows that the histogram maximum narrowing, and therefore accuracy improvement, is achieved for the pulse repetition period.

4. Discussion

4.1. Noise Variance Estimation

The n_{TC} parameter signal (see Figures 12 and 13) noise dispersion $f(t)$ estimation is an important aspect of time series analysis and signal processing since dispersion characterizes the noise values spread degree around its average value. In the signal $f(t)$ analyzing process, which can be represented as the useful signal and noise sum, isolating and estimating the noise variance makes it possible for the filtering methods to signal and the effectiveness quality evaluated. A statistical method is used to calculate noise variance: first, the average noise value is determined, and then all noise values' standard deviation from this average is calculated. Noise dispersion is defined as:

$$\sigma^2 = \frac{1}{N} \cdot \sum_{i=1}^{N} (n(t) - \overline{n})^2, \tag{41}$$

where $n(t)$ are the noise values at time t, $\overline{n}$ is the average noise value, and N is the element number in the training set.

Figure 21 shows a diagram of the noise dispersion resulting estimate depending on the number of the elements in the training set, from which it can be seen that if there are 156 elements in the training set (58% of the total volume), the noise dispersion value becomes almost equal to zero. This indicates that increasing the training set size significantly improves the model's accuracy in estimating noise, and once a certain amount of data are reached, the model almost eliminates the uncertainties associated with noise. Thus, we can conclude that to achieve minimum noise variance, it is necessary to use at least 156 elements in the training set, which ensures that all possible signal variations have adequate representation and allows the model to effectively take them into account.

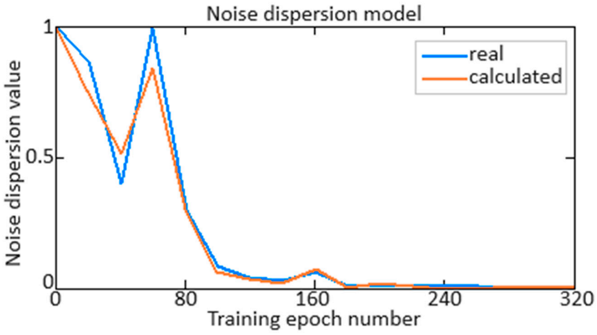


Figure 21. Noise dispersion diagram of the n_{TC} gas–generator rotor rpm signal (author's research).

4.2. Comparative Analysis of Neural Network Signal Integration Based on the Filtering Method with Traditional Filters

When performing the neural network signal integration comparative analysis using the filtering method and traditional filters, the following metrics are used:

1. Mean square error characterizes the differences in squares arithmetic mean between the observed and predicted values defined as:

$$MSE = \frac{1}{N} \cdot \sum_{i=1}^{N} \left(f_i(t) - \hat{f}_i(t) \right)^2. \tag{42}$$

2. The average absolute error characterizes the estimated values' average absolute deviation from the true signal values and is defined as:

$$MAE = \frac{1}{N} \cdot \sum_{i=1}^{N} \left| f_i(t) - \hat{f}_i(t) \right|. \tag{43}$$

3. The determination coefficient is a measure showing the variation proportion in the dependent variable explained by the independent variables in the model and is defined as:

$$R^2 = 1 - \frac{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(f_i(t) - \hat{f}_i(t) \right)^2}{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(f_i(t) - \overline{f} \right)^2}. \tag{44}$$

4. Peak signal to noise is used to the reconstructed signal quality measure concerning the maximum signal and is defined as:

$$PNSR = 10 \cdot \log_{10} \left(\frac{MAX^2}{MSE} \right), \tag{45}$$

where MAX is the maximum possible signal value.

5. Signal-to-noise ratio measures the ratio between signal power and noise power and is defined as:

$$SNR = 10 \cdot \log_{10} \left(\frac{\sum_{i=1}^{N} (f_i(t))^2}{\sum_{i=1}^{N} \left(f_i(t) - \hat{f}_i(t) \right)^2} \right). \tag{46}$$

6. Correlation measures the linear relations between the true and filtered values and is defined as:

$$r = \frac{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(f_i(t) - \overline{f} \right) \cdot \left(\hat{f}_i(t) - \overline{\hat{f}} \right)}{\sqrt{\sum_{i=1}^{N} \left(f_i(t) - \overline{f} \right)^2 \cdot \sum_{i=1}^{N} \left(f_i(t) - \overline{\hat{f}} \right)^2}}, \tag{47}$$

where $\overline{f}$ and $\overline{\hat{f}}$ are the average values of the true and filtered signals, respectively.

7. The root mean square error is the square root of the *MSE* gives an idea of magnitude errors and is defined as:

$$RMSE = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(f_i(t) - \hat{f}_i(t) \right)^2}. \tag{48}$$

8. Average absolute percent error measures the average absolute error as the percentage of the true value and is defined as:

$$MAPE = \frac{100\%}{N} \cdot \sum_{i=1}^{N} \left| \frac{f_i(t) - \hat{f}_i(t)}{f_i(t)} \right|. \tag{49}$$

9. Average relative error estimates the predicted values average relative error relative to the true values and is defined as:

$$MRE = \frac{1}{N} \cdot \sum_{i=1}^{N} \left| \frac{f_i(t) - \hat{f}_i(t)}{f_i(t)} \right|. \tag{50}$$

10. The goodness-of-fit index measures the agreement between the true and predicted values, taking into account both precision and deviation and is defined as:

$$CCC = \frac{2 \cdot r \cdot \sigma_f \cdot \sigma_{\hat{f}}}{\sigma_f^2 + \sigma_{\hat{f}}^2 + \left(\overline{f} - \overline{\hat{f}} \right)^2}, \tag{51}$$

where σ_f and $\sigma_{\hat{f}}$ are the true and predicted values standard deviations, respectively.

11. Normalized mean squared error normalizes the true values variance *MSE* concerning, allowing comparison of models at different data scales and is defined as:

$$NMSE = \frac{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(f_i(t) - \hat{f}_i(t) \right)^2}{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(f_i(t) - \overline{f} \right)^2}. \tag{52}$$

12. The signal reconstruction quality function evaluates the signal reconstruction quality, taking into account the minimum values between the true and predicted values and is defined as:

$$SQR = \frac{\frac{1}{N} \cdot \sum_{i=1}^{N} \min\left(f_i(t), \hat{f}_i(t) \right)}{\frac{1}{N} \cdot \sum_{i=1}^{N} f_i(t)}. \tag{53}$$

Table 6 shows the neural network signal integration based on the filtering method with recursive [67], median [68] and median-recursive [69] filters comparative analysis results according to metrics (42)–(53).

Table 6. The neural network signal integration based on the filtering method with traditional filters comparative analysis results (author's research).

Metrics	Neural Network Integration	Median Recursive Filter	Recursive Filter	Median Filter
MSE	0.000992	0.00255	0.00912	0.0116
MAE	0.0079	0.0403	0.0763	0.1622
R^2	0.9495	0.7358	0.6892	0.5171
PNSR	40.02 dB	25.91 dB	20.38 dB	13.80 dB
SNR	39.35 dB	25.25 dB	19.72 dB	13.13 dB
r	0.9761	0.6519	0.4219	0.2740
RMSE	0.0315	0.0505	0.0955	0.1077
MAPE	0.8615%	1.365%	4.250%	17.50%
MRE	0.00861	0.0436	0.0825	0.1750
CCC	0.9756	0.6009	0.4998	0.1097
NMSE	0.505	1.299	2.641	4.121
SQR	0.9960	0.9766	0.9656	0.9207

The results obtained (see Table 6) confirm that neural network signal integration based on the filtering method is the best in all metrics, demonstrating minimal errors and maximum correspondence to the true signal. Meanwhile, applying the median filter is the worst, showing maximum errors and minimum compliance. Table 7 shows the improvement in the neural network quality metrics signal integration results based on the filtering method with traditional filters.

Table 7. The neural network signal integration improvement degree calculating results based on the filtering method with traditional filters (author's research).

Metrics	The Improvement Compared to the Median Recursive Filter	The Improvement Compared to the Recursive Filter	The Improvement Compared to the Median Filter
MSE	2.571	9.194	11.694
MAE	5.101	9.658	20.532
R^2	1.290	1.378	1.836
PNSR	1.545	1.964	2.900
SNR	1.558	1.995	2.997
r	1.497	2.314	3.562
RMSE	1.603	3.032	3.419
MAPE	1.584	4.933	20.313
MRE	5.064	9.582	20.325
CCC	1.624	1.952	8.893
NMSE	2.572	5.230	8.160
SQR	1.020	1.031	1.082

4.3. Results of a Trained Neural Network with Traditional Filtering Methods Comparison

To compare the trained neural network model (Figure 6) for each layer data with the traditional filtering method (for example, using a median-recursive filter), each step results, followed by checking each hidden layer design's actual effectiveness. The following algorithm is used in the work.

The training samples' data (Tables 2–4) are divided into signals and noise to apply traditional filtering methods, followed by the results' analysis at each step. In this case, $S_{clean} = \{x_1, x_3, x_5\}$ is a pure signals' set (x_1 is the gas–generator rotor rpm n_{TC} signal; x_3 is the gas temperature in the compressor turbine T_G^* signal front; and x_5 is the free turbine rotor speed n_{FT} signal), and $N_{noice} = \{x_2, x_4, x_6\}$ is the noise's set (x_2 is the gas–generator rotor rpm n_{TC} noice; x_4 is the gas temperature in the compressor turbine T_G^* noise front; and x_6 is the free turbine rotor speed n_{FT} noise). As a result, two sets are obtained: one containing pure signals and the other containing their corresponding noise.

Next, the traditional filtering method performs each step on the prepared data. In this case, signals are sequentially filtered from interference, dynamic changes are compensated, and sequential filtering is carried out to reduce noise and highlight useful signals.

After this, the prepared data are passed through a trained neural network, followed by saving each hidden layer output: the 1st hidden layer outputs (summing signals with noise) are the h_1, h_2, h_3; the 2nd hidden layer outputs (dynamic compensation) are the z_1, z_2, and z_3; the 3rd hidden layer outputs (filtering first stage) are the f_1, f_2, and f_3; the 4th hidden layer outputs (filtering second stage) are the g_1, g_2, and g_3; and the output layer is the y.

The following is a direct comparison:

1. The summing signals with noise results by the traditional method are compared with the neural network's 1st hidden layer results.
2. The dynamic compensation results by the traditional method are compared with the neural network's 2nd hidden layer results.
3. The filtering 1st stage results by the traditional method are compared with the neural network's 3rd hidden layer results.
4. The filtering 2nd stage results by the traditional method are compared with the neural network's 4th hidden layer results.

In a computational experiment, clear signals and noise interference were identified from the training samples data (Tables 2–4), which are shown in Tables 8–10.

Table 8. The training sample fragment for n_{TC} gas–generator rotor rpm with the separation of clean signal and noise interference (author's research).

Number	**1**	**2**	...	**37**	...	**84**	...	**115**	...	**172**	...	**202**	...	**256**
S_{clean}	0.922	0.962	...	0.923	...	0.935	...	0.942	...	0.944	...	0.912	...	0.956
N_{noice}	0.021	0.020	...	0.025	...	0.022	...	0.020	...	0.030	...	0.023	...	0.025

Table 9. The training sample fragment for T_G^* gas temperature in the compressor turbine front with the separation of clean signal and noise interference (author's research).

Number	**1**	**2**	...	**29**	...	**73**	...	**109**	...	**164**	...	**200**	...	**256**
S_{clean}	0.903	0.933	...	0.955	...	0.895	...	0.898	...	0.875	...	0.880	...	0.922
N_{noice}	0.029	0.031	...	0.020	...	0.031	...	0.020	...	0.030	...	0.022	...	0.031

Table 10. The training sample fragment for n_{FT}-free turbine rotor speed with the separation of clean signal and noise interference (author's research).

Number	**1**	**2**	...	**32**	...	**80**	...	**105**	...	**181**	...	**207**	...	**256**
S_{clean}	0.907	0.913	...	0.888	...	0.911	...	0.921	...	0.936	...	0.903	...	0.952
N_{noice}	0.022	0.020	...	0.021	...	0.021	...	0.020	...	0.019	...	0.023	...	0.021

Table 11 shows the data processing results (Tables 8–10) using a neural network. Table 12 uses traditional filtering methods according to stages.

Table 11. Data processing results from training samples using a neural network (author's research).

Stage Number	Stage Name	Results
1	1st hidden layer	Parameters h_1, h_2, h_3 are calculated according to (10). The final values are h_1 = 0.898, h_2 = 0.875, h_3 = 0.880.
2	2nd hidden layer	The accepted weight and bias matrices are: $W = \begin{pmatrix} 0.5 & 0.3 & 0.2 \\ 0.4 & 0.6 & 0.1 \\ 0.3 & 0.3 & 0.4 \end{pmatrix}$, $b = \begin{pmatrix} 0.1 \\ 0.2 \\ 0.1 \end{pmatrix}$. Using the Smooth ReLU activation function, the parameters z_1, z_2, z_3 are calculated according to (11). The final values are z_1 = 0.988, z_2 = 1.172, z_3 = 0.984.
3	3rd hidden layer	By applying the Smooth ReLU activation function to the linear combinations z_i, the parameters f_1, f_2, f_3 are calculated according to (12). The final values are f_1 = 1.143, f_2 = 1.396, f_3 = 1.142.
4	4th hidden layer	By applying the Smooth ReLU activation function to the linear combinations f_i, the parameters g_1, g_2, g_3 are calculated according to (13). The final values are g_1 = 1.319, g_2 = 1.609, g_3 = 1.319.
5	Output layer	The accepted weight and bias matrices are $v = \begin{pmatrix} 0.4 \\ 0.3 \\ 0.3 \end{pmatrix}$,$c$ = 0.5. The neural network output signal is calculated according to (14). The final value is y = 1.907.

Table 12. Data processing results from training samples using a traditional filtration method (author's research).

Stage Number	Stage Name	Results
1	Summation of signals with their noise	The summation of signals with their noise is carried out in the same way as in the neural network method (Table 11). The final values are $n_{TC}^{(1)} = 0.898$, $T_G^{*(1)} = 0.875$, and $n_{FT}^{(1)} = 0.880$, similar to h_1 = 0.898, h_2 = 0.875, and h_3 = 0.880.
2	Dynamic compensation	Signals and interference are adjusted using coefficients for each parameter. For a median-recursive filter, according to [70], it is advisable to use the following coefficients: 0.8 for the n_{TC} parameter, 1.2 for the T_G^* parameter, and 0.9 for the n_{FT} parameter. Then: $Corrected(n_{TC}) = 0.8 \cdot Clear(n_{TC}) + Noice(n_{TC})$, $Corrected(T_G^*) = 1.2 \cdot Clear(T_G^*) + Noice(T_G^*)$, $Corrected(n_{FT}) = 0.9 \cdot Clear(n_{FT}) + Noice(n_{FT})$, Total values $n_{TC}^{(2)} = 0.932$, $T_G^{*(2)} = 1.007$, and $n_{FT}^{(2)} = 0.918$, similar to z_1, z_2, and z_3.
3	Filtration 1st stage	Signals and interference are adjusted using coefficients for each parameter. For a median-recursive filter, according to [71], it is advisable to use the following coefficients: 0.75 for the n_{TC} parameter, 1.05 for the T_G^* parameter, and 0.85 for the n_{FT} parameter. Then: $Filtered(n_{TC}) = 0.75 \cdot Corrected(n_{TC})$, $Filtered(T_G^*) = 1.05 \cdot Corrected(T_G^*)$, $Filtered(n_{FT}) = 0.85 \cdot Corrected(n_{FT})$, Total values $n_{TC}^{(3)} = 0.699$, $T_G^{*(3)} = 1.057$, and $n_{FT}^{(3)} = 0.780$, similar to f_1, f_2, and f_3.
4	Filtration 2nd stage	Signals and interference are adjusted using coefficients for each parameter. For a median-recursive filter, according to [72], it is advisable to use the following coefficients: 0.65 for the n_{TC} parameter, 0.70 for the T_G^* parameter, and 0.60 for the n_{FT} parameter. Then: $Filtered(n_{TC}) = 0.65 \cdot Corrected(n_{TC})$, $Filtered(T_G^*) = 0.70 \cdot Corrected(T_G^*)$, $Filtered(n_{FT}) = 0.60 \cdot Corrected(n_{FT})$, Total values $n_{TC}^{(3)} = 0.454$, $T_G^{*(3)} = 0.740$, and $n_{FT}^{(3)} = 0.507$, similar to f_1, f_2, and f_3.
5	Final result	The output signal is calculated as: $y = n_{TC}^{(4)} + T_G^{*(4)} + n_{FT}^{(4)} = 1.701$.

The output parameter value obtained in the neural network for each layer is significantly higher than the values obtained using traditional filtering methods, consisting of summing signals with their noise, dynamic compensation, and the first and second filtering stages (see Table 13). This is because neural networks can train and adapt to data complex and nonlinear relations, allowing them to compensate for interference and improve signal quality more effectively. Having the Smooth ReLU activation function allows the neural network to ignore negative values that may represent noise and emphasize positive values, thereby improving the output signal quality. However, higher output values indicate a cleaner and more amplified signal, which is important for improving the overall system accuracy and reliability.

Table 13. The obtained data comparison results (author's research).

Stage Number	Method Type	Stage Name	Output Variable	Value	Comparison Results
1	Neural network	1st hidden layer	h_1 h_2 h_3	0.898 0.875 0.880	The results obtained in the neural network's 1st hidden layer are identical to the results obtained using traditional filtering methods.
	Traditional filtration method	Summation of signals with their noise	$n_{TC}^{(1)}$ $T_G^{*(1)}$ $n_{FT}^{(1)}$	0.898 0.875 0.880	
2	Neural network	2nd hidden layer	z_1 z_2 z_3	0.988 1.172 0.984	The results obtained in the neural network's 2nd hidden layer are up to 44.1% higher than the results obtained using traditional filtering methods.
	Traditional filtration method	Dynamic compensation	$n_{TC}^{(2)}$ $T_G^{*(2)}$ $n_{FT}^{(2)}$	0.932 1.007 0.918	
3	Neural network	3rd hidden layer	f_1 f_2 f_3	1.143 1.396 1.142	The results obtained in the neural network's 3rd hidden layer are up to 44.1% higher than the results obtained using traditional filtering methods.
	Traditional filtration method	Filtration 1st stage	$n_{TC}^{(3)}$ $T_G^{*(3)}$ $n_{FT}^{(3)}$	0.699 1.057 0.780	
4	Neural network	4th hidden layer	g_1 g_2 g_3	1.319 1.609 1.319	The results obtained in the neural network's 4th hidden layer are up to 68.5% higher than the results obtained using traditional filtering methods.
	Traditional filtration method	Filtration 2nd stage	$n_{TC}^{(4)}$ $T_G^{*(4)}$ $n_{FT}^{(4)}$	0.454 0.740 0.507	
5	Neural network	Final result	y	1.907	The output signal value obtained in the neural network's output layer is 10.8% higher than its value obtained using traditional filtering methods.
	Traditional filtration method	Final result	y	1.701	

4.4. The I and II Type Errors Calculation

A type I error occurs when the null hypothesis H_0 is rejected even though it is true, and is defined at a given significance level as:

$$\alpha = P(\text{Reject } H_0 | H_0 \text{ is true}). \tag{54}$$

A type II error occurs when the null hypothesis H_0 is not rejected even though the alternative hypothesis H_1 is true, and is defined as:

$$\beta = P\left(\text{Don't reject } H_0 \middle| H_1 \text{ is true}\right). \tag{55}$$

As mentioned above, the paper set the significance level to 0.01, which means the type I error probability (erroneously rejecting a true null hypothesis) is 1%; that is, if a statistical test shows a significant result, there is only a 1% chance that this result is due to chance and caused by noise or random variations in the data. The significance indicates a given level of high requirements for the reliability of the results, which is especially important for ensuring the helicopter TE thermogas-dynamic parameters sensor readings' accuracy and reliability.

For the given task, the null hypothesis is "The developed neural network (see Figure 6) does not improve the helicopter TE thermogas-dynamic parameters from sensor signals integration accuracy in comparison with traditional filters (median-recursive, recursive and median filters)", and the alternative hypothesis is "The developed neural network (see Figure 6) significantly improves the helicopter TE thermogas-dynamic parameter sensor signals integrating accuracy compared to traditional filters (median-recursive, recursive and median filters)".

Table 8 shows the 1st and 2nd types' errors calculating results for neural network integration of signals based on the filtering method with recursive [67], median [68] and median-recursive [69] filters according to metrics (42)–(53).

The results (see Table 14) showed that the neural network signal integration based on the filtering method used made it possible to reduce the 1st and 2nd types' errors by 2.11 times compared with the use of a median-recursive filter, by 2.89 times compared with the recursive filter use, and by 4.18 times compared with the recursive filter use. using a median filter.

Table 14. The neural network signal integration based on the filtering method 1st and 2nd types errors calculating results with traditional filters (author's research).

Error Type	Neural Network Integration	Median Recursive Filter	Recursive Filter	Median Filter
Type I error, %	0.86	1.82	2.49	3.60
Type II error, %	0.38	0.80	1.10	1.59

The neural network implementation approach in real helicopter operating conditions faces challenges and advantages number. The main challenges include the need for significant computing resources to train and operate the neural network, difficulties adapting to rapidly changing operating conditions, and interference of various types. In addition, careful model tuning and validation of large amounts of data are required to avoid overfitting and ensure that the system operates stably. However, this approach's advantages are significant: it provides helicopter TE parameters signal filtering with higher accuracy and reliability through adaptive noise suppression and integration of dynamic compensation methods. A neural network trained using a backpropagation algorithm with an adaptive training rate allows you to balance between model accuracy and generalization ability, preventing overfitting. As a result, this method improves the signal filtering efficiency compared to traditional methods and reduces errors of the first and second types several times, significantly increasing the helicopter TE performance and reliability control in real operating conditions.

Thus, the research did not focus on creating a new helicopter or engine. At the same time, it is focused on the sensors data analysis from a particular class of existing helicopter TE thermogas-dynamic parameters (the TV3-117 engine was used in the work) and its parameters (Tables 2–4 [40–42,59–61]), obtained in the Mi-8MTV helicopters flight operation. This aim is achieved by collecting data from sensors during engine operation, data analysis obtained to identify deviations in engine operation, comparison of deviations with benchmarks for a specific engine type, and determining the resulting deviations' causes.

The prospect for further research is the recommendations for eliminating the identified deviations of helicopter TE parameters from the reference values.

5. Conclusions

The article develops a helicopter turboshaft engine's thermogas-dynamic parameter signals neural network integration method, which allows data to be effectively corrected from sensors in real time, ensuring high accuracy and reading reliability:

1. The helicopter turboshaft engines' thermogas-dynamic parameter signals neural network integration method relevance is substantiated since this method provides effective noise filtering, which makes it possible to increase the engine condition monitoring accuracy.
2. An integrating signals scheme from helicopter turboshaft engine thermogas-dynamic parameter sensors has been developed using a filtering method, which achieves almost 100% (0.995 or 99.5%) accuracy and reduces the loss function to 0.005 (0.5%) with 280 training epochs.
3. Based on the backpropagation algorithm, a neural network training method has been developed for the helicopter turboshaft engine parameters integrating control loops, which combines increasing accuracy on the validation sample and controlling overtraining into a single criterion. This method minimizes the loss function and considers the error dynamics on the validation set, preserving the model's ability to generalize. The adaptive training rate helps quickly adapt to data changes and improves performance. In this case, to achieve the loss function minimum value of 2.005, 280 training epochs are enough, after which the error begins to increase; however, the loss function stabilizes immediately after 320 epochs and remains stable for 1000 epochs.
4. It is proposed that a modified Smooth ReLU activation function be used, in which accuracy reaches 0.995, and the loss function decreases from 0.025 to 0.005 in 280 epochs, while with ReLU it takes 490 epochs to achieve the same accuracy and loss, and in 280 epochs the accuracy reaches only 0.972. Furthermore, losses are reduced to 0.018.
5. It is mathematically substantiated that the neural network integration closed loops used for regulating the helicopter turboshaft engine parameters using the filtering method compared with traditional filters (median-recursive, recursive, median filter) improves efficiency by 1.020. . .5.101 times compared to the median-recursive filter, 1.031. . .9.658 times compared to the recursive filter, and 1.082. . .20.325 times compared to the median filter.
6. It is mathematically substantiated that the neural network signal integration use based on the filtering method made it possible for the first and second types to reduce errors by 2.11 times compared with the median-recursive filter use, by 2.89 times compared with the recursive filter use, and by 4.18 times compared with the median filter use.

Author Contributions: Conceptualization, S.V. and V.V.; methodology, S.V., V.V. and S.O.; software, V.V.; validation, V.V., S.O. and A.S.; formal analysis, S.V.; investigation, L.S. and A.S.; resources, L.S., V.S., O.M. and A.S.; data curation, V.V. and S.O.; writing—original draft preparation, V.V.; writing—review and editing, S.V.; visualization, V.V.; supervision, V.S. and O.M.; project administration, V.S. and O.M.; funding acquisition, L.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Ministry of Internal Affairs of Ukraine "Theoretical and applied aspects of the development of the aviation sphere" under Project No. 0123U104884.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data are contained within the article.

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

References

1. Zhang, S.; Ma, A.; Zhang, T.; Ge, N.; Huang, X. A Performance Simulation Methodology for a Whole Turboshaft Engine Based on Throughflow Modelling. *Energies* **2024**, *17*, 494. [CrossRef]
2. Gu, Z.; Pang, S.; Zhou, W.; Li, Y.; Li, Q. An Online Data-Driven LPV Modeling Method for Turbo-Shaft Engines. *Energies* **2022**, *15*, 1255. [CrossRef]
3. Kim, S.; Im, J.H.; Kim, M.; Kim, J.; Kim, Y.I. Diagnostics using a physics-based engine model in aero gas turbine engine verification tests. *Aerosp. Sci. Technol.* **2023**, *133*, 108102. [CrossRef]
4. Zhang, J.; Wang, Z.; Li, S.; Wei, P. A digital twin approach for gas turbine performance based on deep multi-model fusion. *Appl. Therm. Eng.* **2024**, *246*, 122954. [CrossRef]
5. Catana, R.M.; Badea, G.P. Experimental Analysis on the Operating Line of Two Gas Turbine Engines by Testing with Different Exhaust Nozzle Geometries. *Energies* **2023**, *16*, 5627. [CrossRef]
6. Aygun, H.; Turan, O. Application of genetic algorithm in exergy and sustainability: A case of aero-gas turbine engine at cruise phase. *Energy* **2022**, *238 Pt A*, 121644. [CrossRef]
7. Liu, X.; Chen, Y.; Xiong, L.; Wang, J.; Luo, C.; Zhang, L.; Wang, K. Intelligent fault diagnosis methods toward gas turbine: A review. *Chin. J. Aeronaut.* **2024**, *37*, 93–120. [CrossRef]
8. Li, B.; Zhao, Y.-P. Group reduced kernel extreme learning machine for fault diagnosis of aircraft engine. *Eng. Appl. Artif. Intell.* **2020**, *96*, 103968. [CrossRef]
9. Balli, O. Exergetic, sustainability and environmental assessments of a turboshaft engine used on helicopter. *Energy* **2023**, *276*, 127593. [CrossRef]
10. Abdalla, M.S.M.; Balli, O.; Adali, O.H.; Korba, P.; Kale, U. Thermodynamic, sustainability, environmental and damage cost analyses of jet fuel starter gas turbine engine. *Energy* **2023**, *267*, 126487. [CrossRef]
11. Castiglione, T.; Perrone, D.; Strafella, L.; Ficarella, A.; Bova, S. Linear Model of a Turboshaft Aero-Engine Including Components Degradation for Control-Oriented Applications. *Energies* **2023**, *16*, 2634. [CrossRef]
12. Liu, J. Gas path fault diagnosis of aircraft engine using HELM and transfer learning. *Eng. Appl. Artif. Intell.* **2022**, *114*, 105149. [CrossRef]
13. Song, J.; Li, S.; Zhu, S.; Wang, Y.; Zhang, H. Low-emission optimization control method for coaxial compound helicopter/engine based on variable geometry adjustment. *Aerosp. Sci. Technol.* **2024**, *151*, 109263. [CrossRef]
14. Liu, X.; Song, E.; Zhang, L.; Luan, Y.; Wang, J.; Luo, C.; Xiong, L.; Pan, Q. Design and implementation for the state time-delay and input saturation compensator of gas turbine aero-engine control system. *Energy* **2024**, *288*, 129934. [CrossRef]
15. Yang, Y.; Nikolaidis, T.; Jafari, S.; Pilidis, P. Gas turbine engine transient performance and heat transfer effect modelling: A comprehensive review, research challenges, and exploring the future. *Appl. Therm. Eng.* **2024**, *236 Pt A*, 121523. [CrossRef]
16. Gong, W.; Lei, Z.; Nie, S.; Liu, G.; Lin, A.; Feng, Q.; Wang, Z. A novel combined model for energy consumption performance prediction in the secondary air system of gas turbine engines based on flow resistance network. *Energy* **2023**, *280*, 127951. [CrossRef]
17. Kim, S. A new performance adaptation method for aero gas turbine engines based on large amounts of measured data. *Energy* **2021**, *221*, 119863. [CrossRef]
18. Hanachi, H.; Liu, J.; Mechefske, C. Multi-mode diagnosis of a gas turbine engine using an adaptive neuro-fuzzy system. *Chin. J. Aeronaut.* **2018**, *31*, 1–9. [CrossRef]
19. Singh, R.; Maity, A.; Nataraj, P.S.V. Modeling, Simulation and Validation of Mini SR-30 Gas Turbine Engine. *IFAC-Pap.* **2018**, *51*, 554–559. [CrossRef]
20. Ntantis, E.L.; Botsaris, P. Diagnostic methods for an aircraft engine performance. *J. Eng. Sci. Technol.* **2015**, *8*, 64–72. [CrossRef]
21. Pang, S.; Li, Q.; Ni, B. Improved nonlinear MPC for aircraft gas turbine engine based on semi-alternative optimization strategy. *Aerosp. Sci. Technol.* **2021**, *118*, 106983. [CrossRef]
22. Zeng, J.; Cheng, Y. An Ensemble Learning-Based Remaining Useful Life Prediction Method for Aircraft Turbine Engine. *IFAC-Pap.* **2020**, *53*, 48–53. [CrossRef]
23. Zaletin, V.V.; Savitsky, O.A.; Silnikov, M.V.; Sorokovikov, V.N.; Yakushenko, E.I. Acoustic emission diagnostics of a hull structures by a system of integrating fiber-optic sensors for the aircraft and spacecraft safe operation. *Acta Astronaut.* **2024**, *in press*. [CrossRef]
24. Schade, F.; Karle, C.; Mühlbeier, E.; Gönnheimer, P.; Fleischer, J.; Becker, J. Dynamic Partial Reconfiguration for Adaptive Sensor Integration in Highly Flexible Manufacturing Systems. *Procedia CIRP* **2022**, *107*, 1311–1316. [CrossRef]
25. Sun, M.; Liu, Z.; Liu, J. Numerical Investigation of the Intercooler Performance of Aircraft Piston Engines Under the Influence of High Altitude and Cruise Mode. *ASME J. Heat Mass Transf.* **2023**, *145*, 062901. [CrossRef]
26. Liu, Z.; Liu, J. Machine Learning Assisted Analysis of an Ammonia Engine Performance. *J. Energy Resour. Technol.* **2022**, *144*, 112307. [CrossRef]
27. Avrunin, O.G.; Nosova, Y.V.; Abdelhamid, I.Y.; Pavlov, S.V.; Shushliapina, N.O.; Wójcik, W.; Kisała, P.; Kalizhanova, A. Possibilities of Automated Diagnostics of Odontogenic Sinusitis According to the Computer Tomography Data. Possibilities of Automated Diagnostics of Odontogenic Sinusitis According to the Computer Tomography Data. *Sensors* **2021**, *21*, 1198. [CrossRef] [PubMed]
28. Baranovskyi, D.; Bulakh, M.; Michajłyszyn, A.; Myamlin, S.; Muradian, L. Determination of the Risk of Failures of Locomotive Diesel Engines in Maintenance. *Energies* **2023**, *16*, 4995. [CrossRef]

29. Li, B.; Zhao, Y.-P.; Chen, Y.-B. Unilateral alignment transfer neural network for fault diagnosis of aircraft engine. *Aerosp. Sci. Technol.* **2021**, *118*, 107031. [CrossRef]
30. Xu, M.; Wang, J.; Liu, J.; Li, M.; Geng, J.; Wu, Y.; Song, Z. An improved hybrid modeling method based on extreme learning machine for gas turbine engine. *Aerosp. Sci. Technol.* **2020**, *107*, 106333. [CrossRef]
31. Zhu, X.; Li, M.; Liu, X.; Zhang, Y. A backpropagation neural network-based hybrid energy recognition and management system. *Energy* **2024**, *297*, 131264. [CrossRef]
32. Hu, Z.; Kashyap, E.; Tyshchenko, O.K. GEOCLUS: A Fuzzy-Based Learning Algorithm for Clustering Expression Datasets. *Lect. Notes Data Eng. Commun. Technol.* **2022**, *134*, 337–349. [CrossRef]
33. Talebi, S.S.; Madadi, A.; Tousi, A.M.; Kiaee, M. Micro Gas Turbine fault detection and isolation with a combination of Artificial Neural Network and off-design performance analysis. *Eng. Appl. Artif. Intell.* **2022**, *113*, 104900. [CrossRef]
34. Lytvynenko, V.; Nikytenko, D.; Voronenko, M.; Savina, N.; Naumov, O. Assessing the Possibility of a Country's Economic Growth Using Dynamic Bayesian Network Models. In Proceedings of the 2020 IEEE 15th International Conference on Computer Sciences and Information Technologies (CSIT), Zbarazh, Ukraine, 23–26 September 2020; pp. 36–39. [CrossRef]
35. Rusyn, B.; Lutsyk, O.; Kosarevych, R.; Maksymyuk, T.; Gazda, J. Features extraction from multi-spectral remote sensing images based on multi-threshold binarization. *Sci. Rep.* **2023**, *13*, 19655. [CrossRef]
36. Baranovskyi, D.; Myamlin, S. The criterion of development of processes of the self organization of subsystems of the second level in tribosystems of diesel engine. *Sci. Rep.* **2023**, *13*, 5736. [CrossRef]
37. Sachenko, A.; Kochan, V.; Turchenko, V.; Tymchyshyn, V.; Vasylkiv, N. Intelligent nodes for distributed sensor network. In Proceedings of the 16th IEEE Instrumentation and Measurement Technology Conference (IMTC/99), Venice, Italy, 24–26 May 1999; pp. 1479–1484. [CrossRef]
38. Babichev, S.; Korobchynskyi, M.; Lahodynskyi, O.; Korchomnyi, O.; Basanets, V.; Borynskyi, V. Development of a technique for the reconstruction and validation of gene network models based on gene expression. *East. -Eur. J. Enterp. Technol.* **2018**, *1*, 19–32. [CrossRef]
39. Shen, Y.; Khorasani, K. Hybrid multi-mode machine learning-based fault diagnosis strategies with application to aircraft gas turbine engines. *Neural Netw.* **2020**, *130*, 126–142. [CrossRef]
40. Gebrehiwet, L.; Nigussei, Y.; Teklehaymanot, T. A Review-Differentiating TV2 and TV3 Series Turbo Shaft Engines. *Int. J. Res. Publ. Rev.* **2022**, *3*, 1822–1838. [CrossRef]
41. Catana, R.M.; Dediu, G. Analytical Calculation Model of the TV3-117 Turboshaft Working Regimes Based on Experimental Data. *Appl. Sci.* **2023**, *13*, 10720. [CrossRef]
42. Vladov, S.; Shmelov, Y.; Yakovliev, R. Control and Diagnostics of TV3-117 Aircraft Engine Technical State in Flight Modes Using the Matrix Method for Calculating Dynamic Recurrent Neural Networks. *CEUR Workshop Proc.* **2021**, *2864*, 97–109. [CrossRef]
43. Bernardet, U.; Blanchard, M.; Verschure, P.F.M.J. IQR: A distributed system for real-time real-world neuronal simulation. *Neurocomputing* **2002**, *44–46*, 1043–1048. [CrossRef]
44. Bahirev, I.; Basargin, S.; Kavalerov, B. Adaptive control of a gas turbine plant with a reference model and signal tuning. *Control Syst. Inf. Technol.* **2015**, *2*, 71–76.
45. Vladov, S.; Shmelov, Y.; Yakovliev, R. Helicopters Aircraft Engines Self-Organizing Neural Network Automatic Control System. *CEUR Workshop Proc.* **2022**, *3137*, 28–47. [CrossRef]
46. Vasiliev, S.; Valeev, S. Design of intelligent control systems based on the principle of minimum complexity. *Bull. USATU* **2007**, *9*, 32–41.
47. Bahirev, I.; Kavalerov, B. Adaptive control of a gas turbine plant with a reference model and a sigmoid function. *Control Syst. Inf. Technol.* **2015**, *3*, 118–123.
48. Bahirev, I. Application of radial basis function networks for interpolating the equation factors of a gas turbine unit model. *Innov. Process. Res. Educ. Act.* **2014**, *1*, 40–41.
49. Wang, X.; Xu, B.; Han, T.; Wang, Y. Sensor dynamic compensation method based on GAN and its application in shockwave measurement. *Mech. Syst. Signal Process.* **2023**, *190*, 110157. [CrossRef]
50. Wang, Y.; Shi, Y.; Cai, M.; Xu, W.; Yu, Q. Efficiency optimized fuel supply strategy of aircraft engine based on air-fuel ratio control. *Chin. J. Aeronaut.* **2019**, *19*, 489–498. [CrossRef]
51. Lutsenko, I.; Mykhailenko, O.; Dmytriieva, O.; Rudkovskyi, O.; Kukharenko, D.; Kolomits, H.; Kuzmenko, A. Development of a method for structural optimization of a neural network based on the criterion of resource utilization efficiency. *East. -Eur. J. Enterp. Technol.* **2019**, *2*, 57–65. [CrossRef]
52. Wu, D.; Sun, Y.; Xia, R.; Lu, S. Improved Adaptive Fuzzy Control for Non-Strict Feedback Nonlinear Systems: A Dynamic Compensation System Approach. *Appl. Math. Comput.* **2022**, *435*, 127470. [CrossRef]
53. Vijaya Kumar, M.; Suresh, S.; Omkar, S.N.; Ganguli, R.; Sampath, P. A direct adaptive neural command controller design for an unstable helicopter. *Eng. Appl. Artif. Intell.* **2009**, *22*, 181–191. [CrossRef]
54. Hernandez-Gonzalez, M.; Hernandez-Vargas, E.A. Discrete-time super-twisting controller using neural networks. *Neurocomputing* **2021**, *447*, 235–243. [CrossRef]
55. Vladov, S.; Shmelov, Y.; Yakovliev, R.; Petchenko, M. Neural Network Method for Parametric Adaptation Helicopters Turboshaft Engines On-Board Automatic Control. *CEUR Workshop Proc.* **2023**, *3403*, 179–195. Available online: https://ceur-ws.org/Vol-3403/paper15.pdf (accessed on 17 February 2024).

56. Widrow, B.; Stearns, D.S. *Adaptive Signal Processing*; Prentice-Hall Inc.: New York, NY, USA, 1985; pp. 302–367.
57. Karatzinis, G.D.; Boutalis, Y.S.; Van Vaerenbergh, S. Aircraft engine remaining useful life prediction: A comparison study of Kernel Adaptive Filtering architectures. *Mech. Syst. Signal Process.* **2024**, *218*, 111551. [CrossRef]
58. Dumka, P.; Pawar, P.S.; Sauda, A.; Shukla, G.; Mishra, D.R. Application of He's homotopy and perturbation method to solve heat transfer equations: A python approach. *Adv. Eng. Softw.* **2022**, *170*, 103160. [CrossRef]
59. Vladov, S.; Yakovliev, R.; Bulakh, M.; Vysotska, V. Neural Network Approximation of Helicopter Turboshaft Engine Parameters for Improved Efficiency. *Energies* **2024**, *17*, 2233. [CrossRef]
60. Vladov, S.; Yakovliev, R.; Hubachov, O.; Rud, J.; Stushchanskyi, Y. Neural Network Modeling of Helicopters Turboshaft Engines at Flight Modes Using an Approach Based on "Black Box" Models. *CEUR Workshop Proc.* **2024**, *3624*, 116–135. Available online: https://ceur-ws.org/Vol-3624/Paper_11.pdf (accessed on 10 March 2024).
61. Vladov, S.; Shmelov, Y.; Yakovliev, R. Modified Method of Identification Potential Defects in Helicopters Turboshaft Engines Units Based on Prediction its Operational Status. In Proceedings of the 2022 IEEE 4th International Conference on Modern Electrical and Energy System (MEES), Kremenchuk, Ukraine, 20–22 October 2022; pp. 556–561. [CrossRef]
62. Corotto, F.S. Appendix C—The method attributed to Neyman and Pearson. In *Wise Use Null Hypothesis Tests*; Corotto, F.S., Ed.; Academic Press: Cambridge, MA, USA, 2023; pp. 179–188. [CrossRef]
63. Motsnyi, F.V. Analysis of Nonparametric and Parametric Criteria for Statistical Hypotheses Testing. Chapter 1. Agreement Criteria of Pearson and Kolmogorov. *Stat. Ukr.* **2018**, *4*, 14–24. [CrossRef]
64. Babichev, S.; Krejci, J.; Bicanek, J.; Lytvynenko, V. Gene expression sequences clustering based on the internal and external clustering quality criteria. In Proceedings of the 2017 12th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 5–8 September 2017. [CrossRef]
65. Anfilets, S.; Bezobrazov, S.; Golovko, V.; Sachenko, A.; Komar, M.; Dolny, R.; Kasyanik, V.; Bykovyy, P.; Mikhno, E.; Osolinskyi, O. Deep multilayer neural network for predicting the winner of football matches. *Int. J. Comput.* **2020**, *19*, 70–77. [CrossRef]
66. Pasieka, M.; Grzesik, N.; Kuźma, K. Simulation modeling of fuzzy logic controller for aircraft engines. *Int. J. Comput.* **2017**, *16*, 27–33. [CrossRef]
67. Ferreira, H.H.; Gastal, E.S.L. Efficient 2D Tikhonov smoothness regularization with recursive filtering. *Pattern Recognit. Lett.* **2023**, *175*, 95–103. [CrossRef]
68. Tay, D.B. Sensor network data denoising via recursive graph median filters. *Signal Process.* **2021**, *189*, 108302. [CrossRef]
69. Yang, X.; Mu, Y.; Cao, K.; Lv, M.; Peng, B.; Zhang, Y.; Wang, G. Robust kernel recursive adaptive filtering algorithms based on M-estimate. *Signal Process.* **2023**, *207*, 108952. [CrossRef]
70. Poirier, C.; Descoteaux, M. A unified filtering method for estimating asymmetric orientation distribution functions. *NeuroImage* **2024**, *287*, 120516. [CrossRef] [PubMed]
71. Zhao, Y. An inverse Q filtering method with adjustable amplitude compensation operator. *J. Appl. Geophys.* **2023**, *215*, 105111. [CrossRef]
72. Pellegrino, S.F. A filtered Chebyshev spectral method for conservation laws on network. *Comput. Math. Appl.* **2023**, *151*, 418–433. [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.