



Article

A Hybrid Degradation Evaluation Model for Aero-Engines

Likun Ren, Haiqin Qin, Na Cai, Bianjiang Li and Zhenbo Xie *

Qingdao Campus, Naval Aviation University, Qingdao 266041, China

* Correspondence: symbol2008@163.com

Abstract: The non-convergence and low efficiency of the thermodynamic model make them difficult to be used in the aero-engines degradation evaluation, while the negligence of the thermodynamics process of data-driven degradation evaluation methods makes them inaccurate and hard to analyze the actual degradation of air path components. So, we propose a thermodynamic-based and data-driven hybrid model for aero-engine degradation evaluation. Different from thermodynamic-based methods, the iteration calculation is converted to the forward flow in the proposed neural network, thus improving convergence. Moreover, a multi-objective loss function considering the components co-operation process and fusion training process fully taking advantage of simulation and degradation trajectory datasets are proposed to improve the degradation evaluation accuracy. The test case is carried out on NASA's benchmark for aero-engine degradation evaluation. The result shows that the proposed method can improve the accuracy significantly, which suggests its effectiveness.

Keywords: remaining useful life; hybrid model; aero-engine; degradation evaluation



Citation: Ren, L.; Qin, H.; Cai, N.; Li, B.; Xie, Z. A Hybrid Degradation Evaluation Model for Aero-Engines. *Sustainability* **2023**, *15*, 29. <https://doi.org/10.3390/su15010029>

Academic Editors: Feng Lu, Xin Zhou and Jing Cai

Received: 4 November 2022

Revised: 3 December 2022

Accepted: 8 December 2022

Published: 20 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Aero-engines are technology-intensive and high-precision products. Because of operating at an elevated temperature, high pressure, variable load and variable speed for a long period, the performance of air path components (such as compressors and turbines) is significantly degraded, resulting in air path failures from time to time.

As a key tool to improve the reliability and safety of the engine, remaining useful life estimation has been a hot topic in the field of aero-engine monitoring [1].

To get an accurate estimation, two main methods are used.

One is the regression method based on the thermodynamic model. Urban first proposed the concept of Gas Path Analysis (GPA) in the 1970s, arguing that the degradation of air path components is firstly reflected in the changes in the air path structure, such as blade contamination, falling blocks and increased tip clearance. The above structural changes will cause significant changes in the characteristics of the air path components (compressor efficiency, air mass flow, etc.), which in turn will lead to deviations in the air path measurements from baseline conditions. Therefore, there is a corresponding functional relationship between the degradation factors of the engine air path components (e.g., compressor efficiency degradation factor) and the measurable parameters (e.g., total compressor outlet pressure). Now, this functional relationship is widely constructed by using a component-level thermodynamic nonlinear model for aero engines.

Given the aero-engine air path measurements and operating conditions, the degradation factors can be regressed or estimated in conjunction with the thermodynamic model of the aero-engine, and then the degradation of each component can be assessed.

In the 1970s, nonlinear optimal regression of multiple degradation factors was extremely difficult due to the lack of computing power. Linear regression methods [2] transform the nonlinear relationship between the component degradation factors and the measured parameters into a linear one by introducing a linear approximation at a given operating point and thus estimate the degradation factors through a matrix inversion

process. However, large modeling errors are inevitably introduced in the linear process, which reduces the accuracy of the degradation assessment.

With the development of computational power and optimization algorithms, more nonlinear regression methods are introduced to solve the original nonlinear system of equations. For example, the Newton-Raphson method is used to solve the nonlinear system of equations iteratively directly, considering the nonlinear thermodynamic processes of the engine [3]. Moreover, genetic algorithms were introduced into solving the nonlinear equations to get a more accurate evaluation [4].

The thermodynamic model based regression method can calculate all the unmeasurable parameters through the optimization and iteration process and thus can analyze the air path degradation in detail. However, the iteration process inevitably introduces convergence problems. Once the engine works near the flight envelope boundary, the iteration process may run outside the components' map, resulting in interruptions of the thermodynamics calculation [5]. What is more, it takes a long time in the iteration process, and thus is impossible to use these methods in real-time monitoring.

The other one is the data-driven method that reflects the air path measurements to the remaining useful life (RUL for short) directly. The data-driven approach uses machine learning algorithms, such as support vector regression [6], and neural networks [7], to directly establish the mapping relationship between the measurements and the remaining useful life by learning from the training data. In the training phase, the data-driven model is trained with a large number of samples with "measurements-remaining useful life" mapping. In the evaluation phase, the remaining useful life can be calculated by the mapping relationship obtained in the training phase.

With the development of deep learning theory, many related methods are proposed and assessed on the widely used aero engine remaining useful life evaluation benchmark called C-MAPSS released by NASA. Deep Belief Networks including a stack of Restricted Boltzmann Machines are first introduced to evaluate the remaining useful life [8]. Li et al. [9] proposed a deep branched network where failure mode classification and RUL prediction are jointly learned sequentially and got a good result in the C-MAPSS dataset with multiple failure modes. Convolution Neural Network (CNN) is also specified to evaluate RUL in the aero engine field [10]. Li et al. [11] introduced a time window using the convolution neural network to extract time information and get a better result. Recently, Recurrent Neural Network (RNN) and its variation the Long Short-Term Memory (LSTM) networks gained great attention in remaining useful life evaluation for their time information extraction ability. Wu et al. [12] implemented vanilla LSTM networks to achieve good RUL prediction accuracy in the cases of complicated operations, working conditions, model degradations and strong noises. Sayah et al. [13] introduced an enhanced deep LSTM approach, for which Gaussian mixture clustering is performed for all collected sensor data and operational monitoring information. A detailed description of deep learning methods applied in the C-MAPSS dataset can be found in [14,15].

Almost all the proposed methods applied to evaluate RUL in the C-MAPSS dataset are data-driven or deep learning methods. Their advantage is obvious: these methods do not consider the thermodynamic process of the aero-engine, nor do they involve the iterative process of the nonlinear equations, so the computational efficiency is high and can be used for online monitoring. However, the accuracy cannot be high because no thermodynamic constraints are considered in the evaluation and these methods are easy to run into overfitting to the training sets. Moreover, only the remaining useful life can be evaluated using these data-driven methods and the component degradation factors cannot be accessed, the evaluation cannot give a detailed analysis of the engine's operating condition [16].

Aiming to give a more accurate evaluation and detailed analysis of the engine's condition with the help of thermodynamic constraints, this paper proposed a novel hybrid model considering both the convergence and thermodynamic process. The main innovation points are as follows:

- A novel hybrid framework converting the thermodynamic or physical based iterative calculation to a forward flow of network is introduced to improve convergence.
- A novel disequilibrium loss function related to the engine components co-working and fusion training process is proposed to improve the degree of matching to the engine working process. So, the accuracy of the degradation evaluation can be improved.

The rest of the content is organized as follows. First, the traditional thermodynamic models are briefly reviewed. Next, the proposed model is detailed presented. Finally, the method proposed is experimentally validated.

2. Related Works

This section gives a brief review of the thermodynamic model in aero engine modeling.

In the thermodynamic-based model [17,18], major components of aero-engines such as fan, combustion chamber and turbine are modeled by thermodynamic calculations. Given the upstream and working parameters of one component, its exit parameters can be calculated with the component characteristics.

The state parameters at the entrance and exit of the component are called station parameters.

All station parameters can be inferred by solving the equilibrium equation followed during the engine operation.

- The work equation on one shaft.

$$\begin{aligned} N_{LPT}\eta_{mL} &= N_{Fan} + N_{LPC}, \\ N_{HPT}\eta_{mH} &= N_{HPC}, \end{aligned} \quad (1)$$

where N_{LPT} and N_{HPT} are turbine output work on the low and high pressure rotor respectively, N_{Fan} , N_{LPC} and N_{HPC} are the responding demanded work of compressors and η_{mL} and η_{mH} are the responding transferring efficiency. The turbine output work should match the compressor demanded work on the same shaft in a steady state.

- The flow equations through the air path.

$$\begin{aligned} W_{LPT} &= W_{LPTmap}, \\ W_{HPT} &= W_{HPTmap}, \end{aligned} \quad (2)$$

where W_{LPT} and W_{HPT} are the mass flow from upstream through the turbine and W_{LPTmap} and W_{HPTmap} are the mass flow calculated by the characteristics of turbines.

- The throat static pressure equation.

$$PS_{nozzle} = PS_{nozzleQ}, \quad (3)$$

where PS_{nozzle} and $PS_{nozzleQ}$ are the throat static pressure calculated by upstream parameters and downstream or ambient parameters.

Newton-Raphson method is usually used to solve the above Equations (1)–(3) iteratively. However, when the engine works near the flight envelop boundary, the iteration process is easy to run out of the characteristics map, leading to non-convergence.

3. Proposed Degradation Evaluation Method

The proposed degradation evaluation method comprises the following four processes. First, the hybrid network framework is proposed to convert the thermodynamic-based iterative calculation to a forward flow of the network. Then, to rebuild the equilibrium equations in thermodynamic models for engine components collaboratively working, a disequilibrium loss function describing the incompatibility between component modules is introduced. Then, to ensure the accuracy of the hybrid model, the integration training process with multi-source data is proposed. At last, the remaining useful life is estimated

through a regression process. The remaining part of this section will separately introduce the processes mentioned above.

3.1. Hybrid Model Framework

The iterative calculations in the thermodynamic-based method usually run out of the components' envelope, resulting in non-convergence. The main reason lies in that the solution of the equilibrium equation is a nonconvex optimization problem, and unreasonable initial value settings tend to make the iterative process fall into local optimal solutions. To address this issue, a hybrid network framework converting the iterative calculation to the forward process is proposed. The component nets are built to model the component characteristics. So, through the training process, the component characteristics can be embedded in the weights of the component nets and the forward calculation through the component nets can replace the iterative process in the thermodynamic model. The framework is shown in Figure 1.

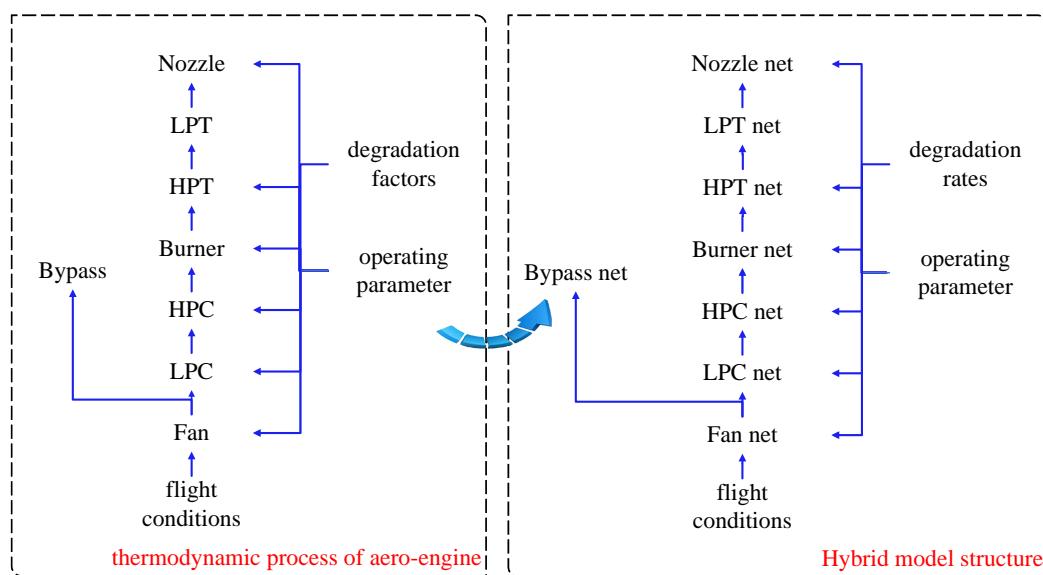


Figure 1. The structure of the proposed hybrid model.

Figure 2 gives an example of a fan net module which is a single-entrance and double-exits air path structure. One of the double-exits is connected to the low-pressure compressor and the other one is connected to the bypass. To calculate the exit parameters of the fan net, the entrance parameters (such as the total temperature and pressure), the working parameters (such as the rotor speed) and its degradation rate should be given.

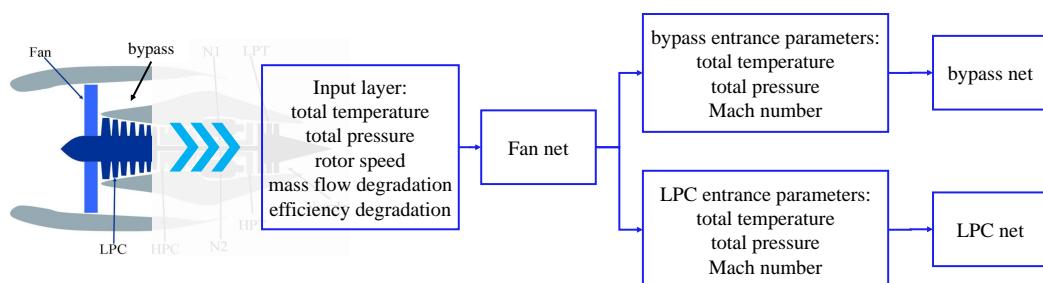


Figure 2. The structure of the fan net.

Once the entrance parameters, the exit parameters and the connection relationship between components are determined, the airflow process inside the fan (the left in Figure 2) can be converted to a network module (the right in Figure 2) whose forward calculation is the same as the thermodynamic flow.

As shown in Figure 3, the net modules for the components have four layers. The state parameters of the component entrance such as the total temperature and pressure, the operating parameters such as the rotor speed and the degradation rate are taken as the inputs. The state parameters of the component exit are calculated through the forward process of the component net. The calculated outputs are in turn taken as the entrance parameters of the downstream component.

Outlet station parameters

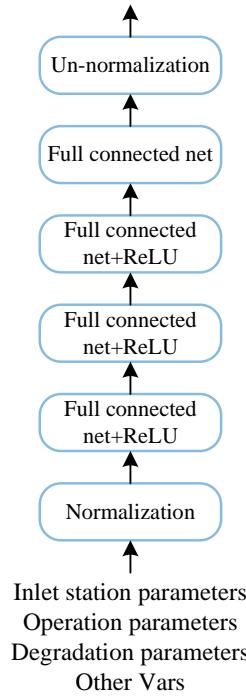


Figure 3. The structure of the component net.

3.2. Disequilibrium Loss

The thermodynamic-based model establishes a set of equilibrium equations for engine components collaboratively working. These equations can be divided into three parts: the mass flow equilibrium describes that the mass flow of the component entrance and exit should be equal; the pressure equilibrium describes that the static pressure of certain components such as nozzle should be equal; the power equilibrium describes that the turbine output power is equal to the compressor power requirement.

Newton-Raphson method is used to solve the equilibrium equations iteratively. However, after transforming into the neural network model, the feed-forward process of the network replaces the iterative solution process of the equilibrium equations. If the mean square loss of the traditional neural network is used as the optimization objective, the equilibrium state for components cooperating is no longer considered and thus is difficult to be satisfied.

To address this issue, a disequilibrium loss based on component co-operating is proposed. The equilibrium equations when components are co-operating at the steady state are converted to corresponding disequilibrium losses as follows to train the hybrid network.

The work Equation (1) are converted to the work disequilibrium loss.

$$\text{loss}_1 = \frac{|\eta_m N_T - N_C - P_{ext}|^2}{N_C}, \quad (4)$$

where η_m is the mechanical drive efficiency of the turbine, N_T is the turbine output work calculated through the state parameters, N_C is the compressor demanding work and P_{ext} is the work extraction from the turbine by aircraft and engine accessories.

The flow Equation (2) are converted to the flow disequilibrium loss.

$$\text{loss}_2 = \frac{1}{M} \sum_{i=1}^M \frac{(W_i - Q(T_i, P_i, Ma_i, A_i))^2}{W_i^2}, \quad (5)$$

where T_i, P_i, Ma_i and A_i denote the total temperature, total pressure, Mach number and area at the i th station. $Q(T_i, P_i, Ma_i, A_i)$ is the mass flow function:

$$Q(T_i, P_i, Ma_i, A_i) = K \frac{P_i A_i}{\sqrt{T_i}} q(Ma_i). \quad (6)$$

The throat static pressure Equation (3) is converted to the pressure disequilibrium loss.

$$\text{loss}_3 = \frac{|PS(T_i, P_i, Ma_i, A_i) - PS_Q(T_i, P_i, Ma_i, A_i)|^2}{PS^2(T_i, P_i, Ma_i, A_i)}, \quad (7)$$

where $PS(T_i, P_i, Ma_i, A_i)$ and $PS_Q(T_i, P_i, Ma_i, A_i)$ are the throat static pressure calculated by the upstream parameters and the downstream/ambient pressure respectively.

Traditional mean squared loss is then added to lead all station parameters converging to values calculated by the thermodynamic model in the simulation dataset or actual values in the flight data.

$$\text{loss}_m = \sqrt{\frac{1}{M} \sum_{i=1}^M (y_i - \hat{y}_i)^2}. \quad (8)$$

where y_i is the station parameters from the thermodynamic model or the flight data and \hat{y}_i the evaluated values from the hybrid network.

Then, the total loss is as follows.

$$\text{loss} = \lambda_0 \text{loss}_m + \lambda_1 \text{loss}_1 + \lambda_2 \text{loss}_2 + \lambda_3 \text{loss}_3, \quad (9)$$

where λ_i is the loss weight.

In (9), loss_1 is the work disequilibrium loss measuring the compatibility of the compressor and turbine in the aero engine, loss_2 is the flow disequilibrium loss measuring the mass flow compatibility of the inlet and the exit of a certain component in the aero engine and loss_3 is the pressure disequilibrium loss measuring the pressure compatibility of the inlet and the exit of certain component in the aero engine. The loss weights in (9) are determined through a hyper-parameter validation experiment where 16 groups of different loss weight settings ($(1, 2, 3, 4) \times (2, 5, 7, 9)$) are compared. Finally, the loss weights group $\lambda_0 = 1, \lambda_1 = \lambda_2 = \lambda_3 = 5$ is chosen for its best performance.

Though the loss function in (9) has as many as four items, there are only two groups. One is the traditional mean squared loss denoted as loss_m . The other group is the thermodynamic loss denoted as $\lambda_1 \text{loss}_1 + \lambda_2 \text{loss}_2 + \lambda_3 \text{loss}_3$. The nature of these three losses is the same (measuring the compatibility of the engine components) and can converge along the same direction once given suitable λ .

3.3. Fusion Training

Based on the establishment of the network framework and disequilibrium loss, the hybrid model needs to be trained so that the model can converge toward the actual working process of the path components under the guidance of the optimization objective. The modeling accuracy of the hybrid model is closely related to the quality of the data set used for training. Two datasets are used to train the hybrid model: the simulation dataset Ω_s and the degradation trajectories dataset Ω_d .

The degradation trajectories dataset includes “measurement—remaining useful life” reflection samples along with certain degradation processes of engines, which is the ideal training set. However, its volume is too small to train the hybrid model. So, the simulation dataset generated by the traditional thermodynamic model with all station parameters and components degradation factors is introduced. The simulation dataset has a large sample volume and can encompass most of the flight envelope. However, there is a certain degree of deviation from the actual working process of the air path components.

Inspired by the mixed batches for the GAN training [19] where the true and fake samples are mixed to input into GANs, a fusion training process with multiple data sources is proposed. The fusion training of the hybrid model within the full flight envelope is performed by combining the data sets above according to the following process, as shown in Figure 4.

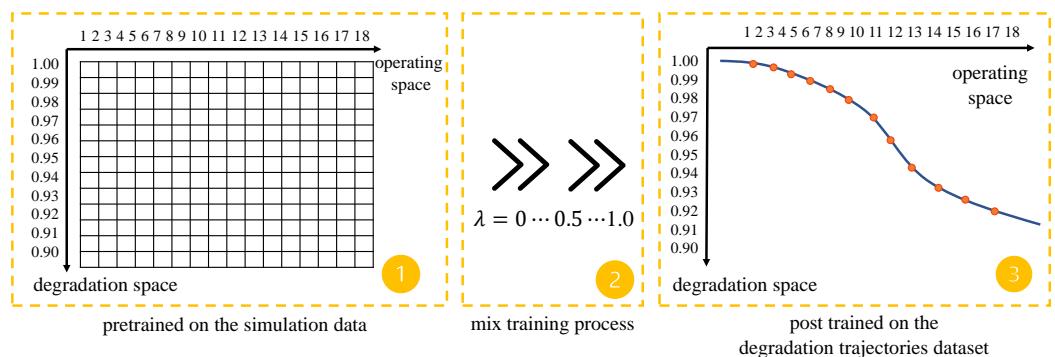


Figure 4. The fusion training process.

First, the hybrid model is pre-trained on the simulation dataset with a fine grid using the diverse working and degradation samples in the simulation dataset that is generated from the rebuilt thermodynamic model. The measurements corresponding to NASA’s dataset are extracted from the parameters of the results output by the thermodynamic model. To give a comprehensive description of the degradation situation, a total of ten degradation parameters including the mass flow degradation and the efficiency degradation of the fan, low pressure compressor, high pressure compressor, high pressure turbine and low pressure turbine are injected into the thermodynamic model to calculate the targeted measurements.

As illustrated in the left part of Figure 4, the degradation space and the operating space are divided into a grid to cover all the working states of aero-engines. Each intersection in the grid is input into the thermodynamic model to calculate the parameters of the air path. So, the hybrid model can roughly learn the thermodynamic processes in the working and degradation space of the air path components. For consistency with the subsequent degradation trajectories dataset, the degradation rate in the degradation trajectories dataset is defined as the maximum of the ratios of the current stall and exhaust gas temperature (EGT) margins to these of the undegraded sample.

$$\theta_i = \max\left(\frac{SM_i}{SM_n}, \frac{EGTm_i}{EGTm_n}\right), \quad (10)$$

where θ_i is the degradation rate of the i th sample in the simulation set Ω_s , SM_i and SM_n are the stall margins of the i th sample and the undegraded sample respectively while $EGTm_i$ and $EGTm_n$ are the exhaust gas temperature margin of the i th sample and the undegraded sample respectively.

Then, the hybrid model is mixed trained using samples from both the simulation dataset and the degradation trajectories dataset. Since only remaining useful life values are accessed related to the degradation information, the degradation rate is defined as the ratio of the remaining useful life of the current sample to the total useful life.

$$\theta_{ij} = \frac{rul_{ij}}{l_j}, \quad (11)$$

where θ_{ij} is the degradation rate of the i th sample in the j th degradation trajectory, rul_{ij} is the remaining useful life of the i th sample in the j th degradation trajectory that can be accessed in the dataset and l_j is the total life of the j th trajectory.

The mixing strategy is to randomly select certain percent training samples from two datasets to generate the training set for one epoch.

$$\begin{aligned} \Omega_k &= \{ss_1, ss_2, \dots, ss_m, sd_1, sd_2, \dots, sd_n\}, \\ ss_i &\in \Omega_s, \\ sd_i &\in \Omega_d, \\ \frac{n}{m+n} &= \lambda_k, \end{aligned} \quad (12)$$

where Ω_k is the training set in the k th epoch, Ω_{ks} is the subset whose samples are randomly selected from the simulation dataset, Ω_{kd} is the subset whose samples are randomly selected from the degradation trajectories dataset and λ_k is the percent of samples from the degradation trajectories dataset in the k th epoch:

$$\lambda_k = \frac{1}{1 + e^{1 - \frac{ak-b}{k_{max}}}}, \quad (13)$$

where k_{max} is the total training epochs in the mixed training process, and a and b are the parameters defining the mixing process in the fusion training. This equation comes from the logistic regression function that can add samples from another dataset gradually.

In the test case, a and b are set to $0.05k_{max}$ and $5k_{max}$ respectively to form λ_k in Figure 5 controlling the percentage of the samples from the degradation trajectories dataset during the training process.

At last, the hybrid model is post-trained on the degradation trajectories dataset for shorter epochs.

Different from the traditional pre-training process where the model is first trained in the pre-training dataset and further trained in the targeted dataset [20], the fusion training process mixes the simulation dataset and degradation trajectories dataset in one training epoch. This fusion method can avoid sudden changes in the network weight optimization process caused by the switch of the training set so that the network training process is much smoother and easier to get a better training result.

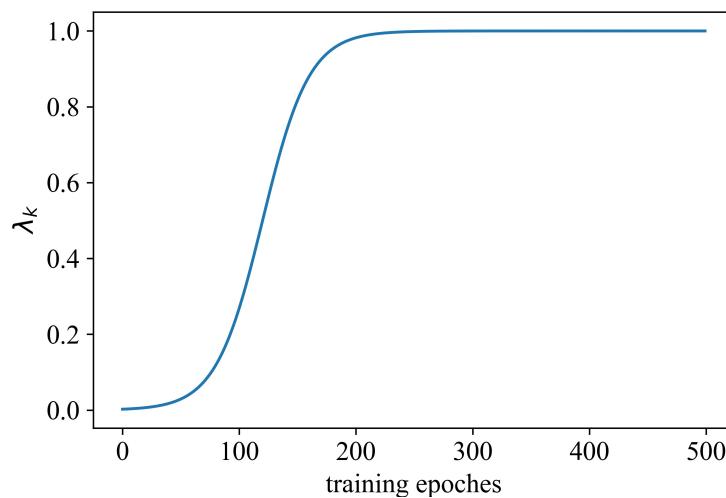


Figure 5. λ_k during the training process.

3.4. Model Testing

When trained by the simulation data and degradation trajectory data, the hybrid model can reflect the degradation rates and operating settings to the measurement parameters.

$$z_j = \mathbf{h}(\omega_j, \theta_j), \quad (14)$$

where ω_j is the engine operating vector, θ_j is the degradation rate of the last cycle in the j th testing sample, z_j is the corresponding measurement parameters of the air path vector and \mathbf{h} is the hybrid model mapping the degradation rates to the measurement parameters.

When given the measurement parameters and engine operating parameters, the degradation rates of the last cycle in the j th testing sample can be regressed using the reverse operation of (14).

$$\theta_j = \mathbf{h}^{-1}(\omega_j, z_j), \quad (15)$$

Then, the remaining useful life can be computed using the following reverse calculation.

$$\hat{r}_j = \frac{c_j}{\theta_j}, \quad (16)$$

where \hat{r}_j is the evaluated remaining useful life value and c_j is the cycle numbers that the j th sample has operated.

At last, the root mean square error (RMSE) metric is used to measure the accuracy of the proposed methods:

$$RMSE = \sqrt{\frac{1}{N} \sum_1^N (\hat{r}_j - r_j)^2}, \quad (17)$$

where N is the total sample number of the testing set, \hat{r}_j is the evaluated remaining useful lift and r_j the actual one.

4. Test Case

4.1. Dataset Description

The proposed method is tested under NASA's benchmark [21,22]. Different operating states and fault types are included in the four datasets of this benchmark. Table 1 shows the detail of the benchmark whose four datasets are FD001, FD002, FD003 and FD004 respectively.

Table 1. Benchmark details.

Datasets	FD001	FD002	FD003	FD004
the number of training units	100	260	100	249
the number of testing units	100	259	100	248
the number of operation states	1	6	1	6
the number of fault modes	1	1	2	2
the number of training samples	17,731	48,819	21,820	57,522
the number of testing samples	100	259	100	248

Three operating states including the flight altitude, Mach number and inlet total temperature and twenty-one measurements are recorded in one sample. Table 2 shows the meaning of the operation states and measurements. Certain reasonable initial wear and individual variation are injected into every aero-engine which is taken to be healthy at the beginning. With the continuous use of the aero-engine, it begins to degenerate at a certain time point and then degrades to failures, when the remaining useful life of this aero-engine is denoted as zero.

Table 2. Measurements in the NASA benchmark.

Indices	Description
1	Altitude
2	Flight Mach number
3	Throttle resolver angle
4	Temperature at fan entrance
5	Temperature at the low-pressure compressor exit
6	Temperature at the high-pressure compressor exit
7	Temperature at the low-pressure turbine exit
8	Pressure at the fan entrance
9	Pressure at the bypass
10	Pressure at the high-pressure compressor exit
11	Fan speed
12	Core speed
13	Pressure ratio
14	Static pressure at the high-pressure compressor exit
15	Ratio of fuel flow
16	Corrected fan speed
17	Corrected core speed
18	Bypass ratio
19	Fuel-air ratio
20	Bleed enthalpy
21	Demanded fan speed
22	Demanded corrected fan speed
23	High-pressure turbine coolant bleed
24	Low-pressure turbine coolant bleed

To one sample of the training set, the remaining useful life can be inferred through the following equation for each cycle of the aero-engine before failure.

$$r_i = l - c_i, \quad (18)$$

where r_i is the i th sample's remaining useful life, c_i is the cycle number this sample experienced and l is the total cycle number of this sample.

An episode from the beginning to some point before failure is extracted from the entire running cycle to form the testing sample. So its remaining useful life can be inferred by Equation (18) and provided in the datasets. The proposed method is tested to evaluate the remaining useful life in the four testing datasets.

During the training phase, all the provided measurements associated with its remaining useful life values are used as the training samples. During the testing phase, the data point recorded at the last cycle for each aero-engine is used as the testing sample.

4.2. Data Correction and Augmentation

The degeneration is usually not occurred in the early phase of one engine while the remaining useful life values given by Equation (18) show a decreasing trend. So, a constant degradation correction [23] in the early phase is applied as follows:

$$rc_i = \min(r_i, r_{max}), \quad (19)$$

where r_i is the calculated RUL value from Equation (18) and r_{max} is the pre-set maximum RUL that is set to 130 in our experiment. This correction assumes that the aero-engine will not degenerate until 130 cycles from the bottom.

Because the proposed model is pre-trained by simulation data, the inputs and the outputs of the model are the real values without normalization. So, there are no data pre-processing.

Another important issue in the aero-engine degradation evaluation is the measurements or sensors selection. In the subsets FD001 and FD003 of NASA's dataset, some measurements remain constant all the engine's lifetime. In this condition, these measurements are useless to the RUL estimation. Therefore, sensors 2, 3, 4, 7, 8, 9, 11, 12, 13, 14, 15, 17, 20, and 21 are selected as the input features for subsets FD001 and FD003. As for subsets FD002 and FD004, no constant measurements exist and all the measurements are selected as the input features.

The volume of the training set in the benchmark datasets is too small to fully train the hybrid model. Only 100 units and an average of 150 samples per unit are accessed. To enlarge the training sample, the rebuilt thermodynamic model is used to generate more training data according to the setting of the benchmark datasets.

Four augmentation datasets with 20,000 samples each corresponding to the four subsets in the benchmark datasets (Table 1) are generated from the rebuilt thermodynamics model, which is called FD001AUG to FD004AUG respectively. In each augmentation dataset, the fault modes are the same as the original subset, the degradation factors are randomly generated over the scale of the original subset and the degradation rates are computed according to (10).

4.3. Model Training and Testing

As proposed in Section 3.3, the hybrid model is pre-trained in the augmentation dataset using the multi-objective loss function. The training epochs are set to 500. The learning rate is set to $1e - 3$ with 10% decay every 100 epochs. The batch size is 256.

Here, hyper-parameters such as the batch size and the initial learning rate are essential to get a better result. To get a relatively reasonable set of these hyper-parameters, the final hyper-parameters are the result of a grid-search process: the size of mini-batch [128, 256, 512] and the learning rate [$1e - 2, 5e - 3, 1e - 3, 5e - 4$]. The hyper-parameters with the best RMSE in the validation set (the batch size 256 and the initial learning rate $1e - 3$) are chosen.

The trend of the maximum relative error of the Exhaust Gas Temperature (EGT) during the simulation pre-training process is shown in Figure 6. It can be seen that when the number of training epochs reaches about 200, the EGT tends to converge and the maximum relative error is less than 0.1% which can be acceptable considering the max error of corresponding thermodynamic model can reach 0.5%.

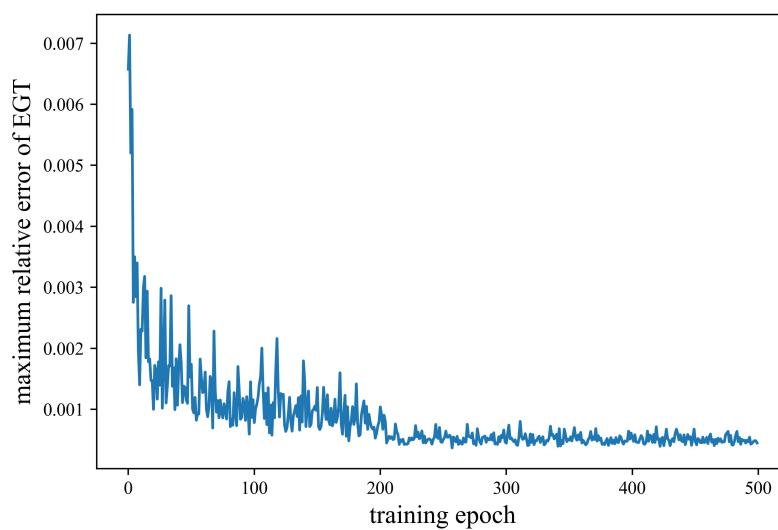


Figure 6. The Exhaust Gas Temperature trend.

The trend of the maximum relative error of each equilibrium loss in (9) with the training process is shown in Figures 7–9. It can be seen that each equilibrium loss gradually converges to an acceptable error range as the training proceeds.

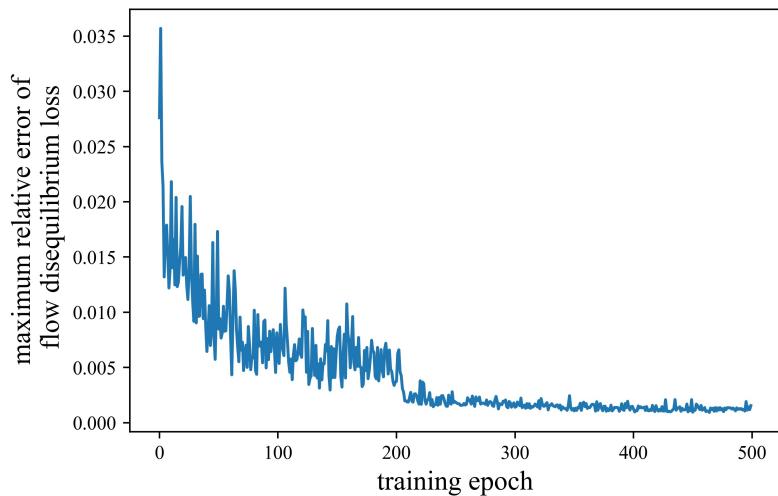


Figure 7. The flow equilibrium loss trend.

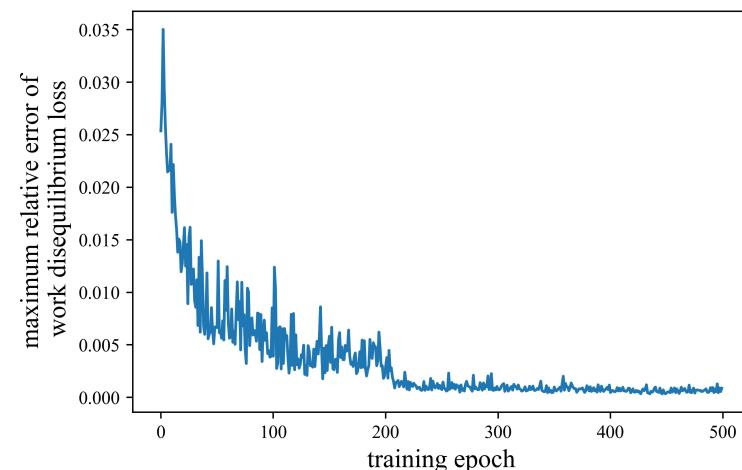


Figure 8. The work equilibrium loss trend.

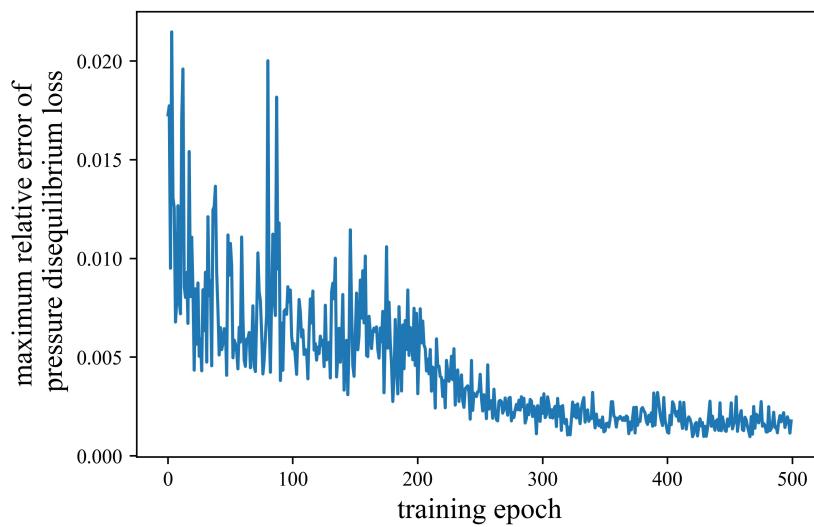


Figure 9. The pressure equilibrium loss trend.

Then, the hybrid model is mixed trained in the augmentation dataset and the degradation training set provided by the benchmark datasets as proposed in Section 3.3. The training epochs are also set to 500. The learning rate is set to $1e - 4$ with 10% decay every 200 epochs. The batch size is also 256.

At last, the hybrid model is post-trained in the degradation training set. The training epochs are set to 200. The learning rate is set to $5e - 4$. The batch size is 256.

The trend of the maximum relative error of each equilibrium loss in (9) after the post-training process is shown in Figures 10–12. It can be seen that although the error values of each equilibrium loss are enlarged compared with the pre-training results of the simulation data, each equilibrium loss is still in an acceptable range considering the tolerance of the corresponding equilibrium equations for the thermodynamic model can be set to 1%.

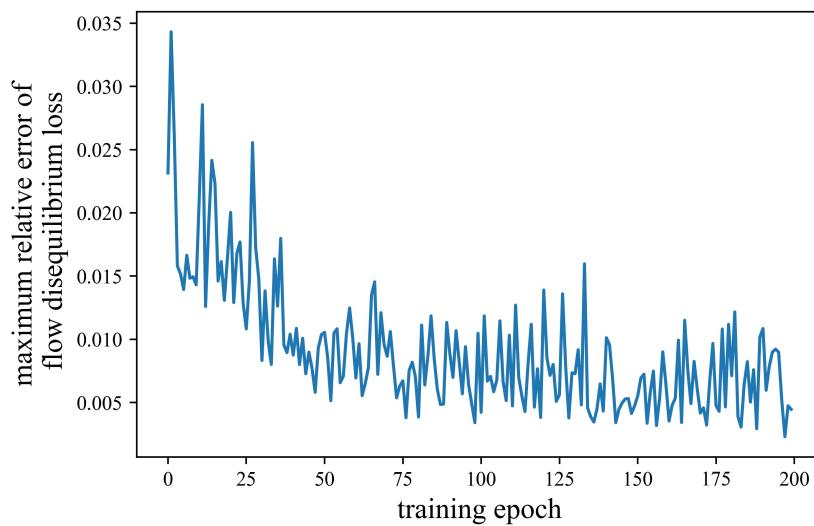


Figure 10. The flow equilibrium loss trend in the post-training process.

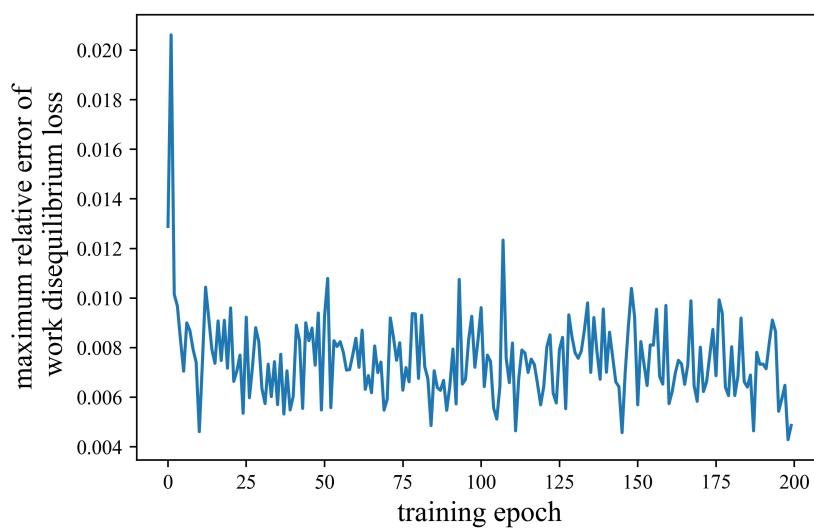


Figure 11. The work equilibrium loss trend in the post-training process.

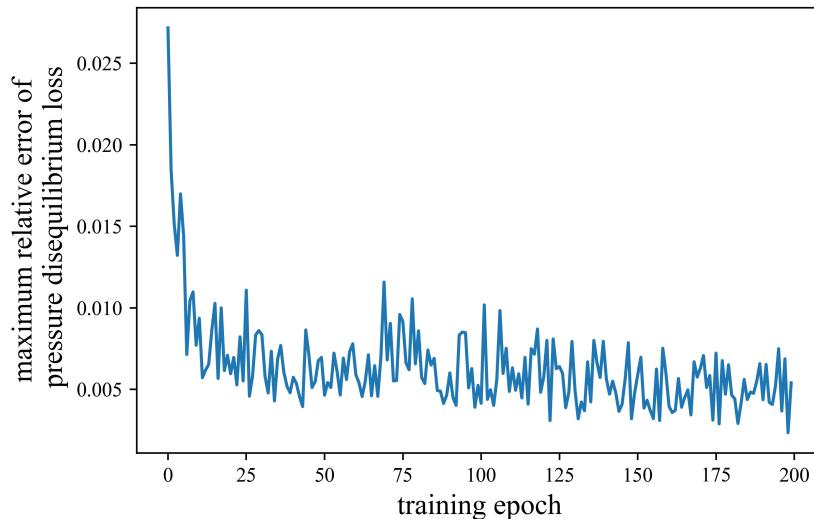


Figure 12. The pressure equilibrium loss trend in the post-training process.

The hybrid model is then tested on the testing sets in the benchmark datasets. A total of 200 repeated experiments are done. The RMSE values are computed as (17). The mean and standard deviation RMSE values and the R^2 scores mean of the repeated experiments are shown in Table 3. And R^2 scores is defined as follows:

$$R^2 = 1 - \frac{\sum(r_i - \hat{r}_i)^2}{\sum(r_i - \bar{r})^2}, \quad (20)$$

where r_i is the true RUL value of the i th sample, \hat{r}_i is the evaluated one and \bar{r} is the mean of all the true RUL values. The result shows that the proposed method's R^2 score is above 0.9 in FD001 and FD003 while above 0.65 in FD002 and FD004, validating the robustness of the proposed method.

Table 3. Results of the proposed method.

	FD001	FD002	FD003	FD004
RMSE-mean	11.34	19.89	11.87	22.07
RMSE-std	0.24	0.39	0.28	0.41
R^2 -mean	0.917	0.756	0.908	0.690

Figure 13 shows an example of the evaluated remaining useful life by the proposed model for the FD001 testing samples and the corresponding actual remaining useful life given by the benchmark. These samples are arranged by the actual remaining useful life values from small to large for better illustration. We can see that the differences are smaller in the region where the actual remaining useful life values are small. This is mainly because the changes in the air path are obvious near the aero-engine failure and the proposed method can capture these air path changes through thermodynamics-oriented modeling. However, the deviation becomes higher in the middle testing samples. This is mainly because the engine has passed the break-in period and the fault features during this stable period are not obvious, resulting in a relative higher evaluation error.

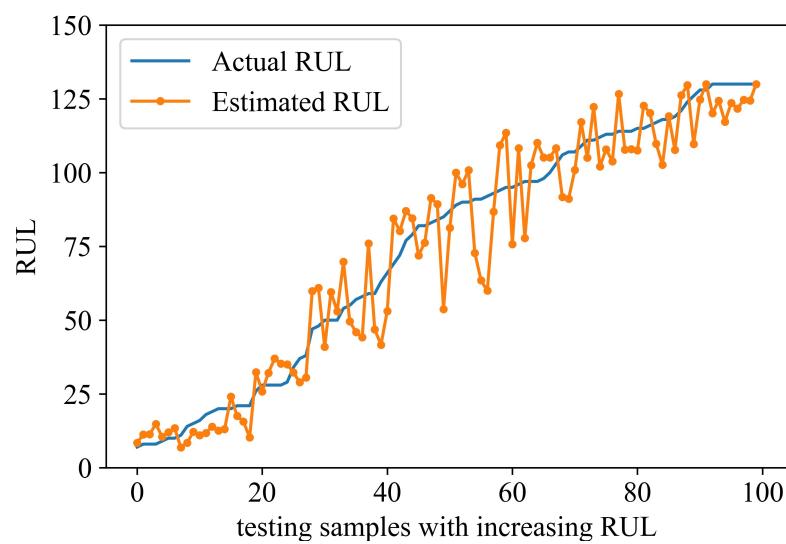


Figure 13. Sorted evaluations of the FD001 testing samples.

To verify the effectiveness of the fusion training proposed in Section 3.3, a comparison experiment is carried out using traditional pre-training process. The trends of the maximum relative error of the Exhaust Gas Temperature (EGT) with traditional pre-training process and fusion training after the post-training process are shown in Figure 14. It can be concluded that during the pre-training process, the training error converges slower than the fusion training process and the final error under the pre-training process is also larger than that under the fusion training process.

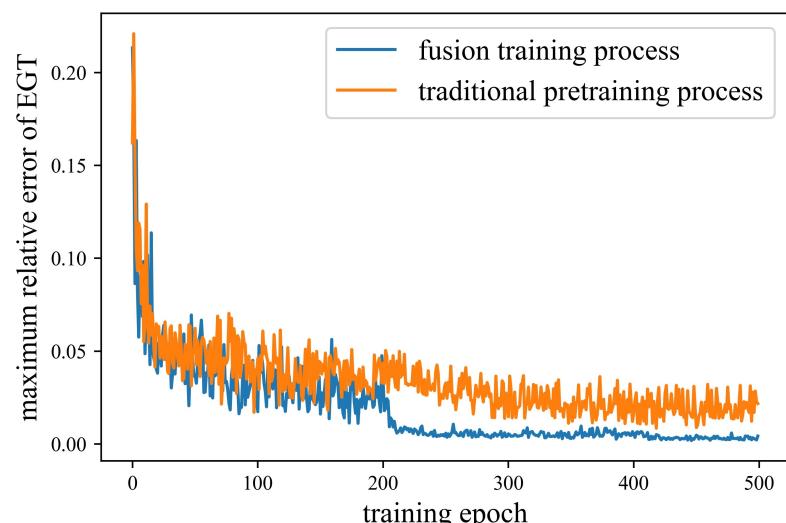


Figure 14. Comparison between pretraining and fusion training process.

4.4. Models Comparison

Table 4 shows comparisons among different degradation evaluation methods measured by RMSE metric on the four benchmark datasets.

In Table 4, networks ensemble [8] is ensembled by deep belief networks composed of three stacked Restricted Boltzmann Machines. Tensorcast [24] is a tensor-based method that forecasts time-evolving series by incorporating multiple data sources in coupled tensors. LSTM [25] used two Long short-term memory networks layers and an LSTM fusion layer to estimate the RUL. BiLSTM [26] used two BiLSTM layers and two fully connected layers to estimate the RUL. In the CNN LSTM hybrid [27], the CNN is used to extract the features of time-series data and the long short-term memory (LSTM) networks considers the timing of the data. The structure of 1D CNN [11] consists of 5 convolution layers and one fully connected layer.

Table 4. Performance comparisons of the proposed method and state of the art methods.

Methods	FD001	FD002	FD003	FD004
BiLSTM [26]	13.65	23.18	13.74	24.86
CNN LSTM hybrid [27]	12.58	19.34	12.18	20.03
1D CNN [11]	12.61	22.36	12.64	23.31
Bi-level LSTM [28]	11.80	23.14	12.37	23.38
MA-LSTM [29]	13.52	22.57	12.98	23.88
Proposed hybrid model	11.34	19.89	11.87	22.07

It can be seen that the model proposed in the paper achieves high accuracy on the benchmark, verifying its validity. From Table 4, we can see that the accuracy of the proposed method in FD002 and FD004 is not better than CNN LSTM hybrid [27]. We think there may be the following reasons. One is that there are more training samples (about 250) in FD002 and FD004, and in this situation, the effectiveness of the simulation training will be reduced. The other one is that the number of operating conditions in FD002 and FD004 (6 conditions) is more than that in FD001 and FD003 (only one condition), so the gap between the rebuilt thermodynamic model and the original model generating these datasets will be larger, resulting in the lower accuracy of our method in FD002 and FD004.

5. Conclusions

This paper proposed a novel hybrid degradation evaluation model for aero-engines. Thermodynamics-based network converting the iteration calculation into the forward flow in the network is constructed to improve the convergence. Disequilibrium loss functions considering air path components co-working and fusion training process taking advantage of the multi-source dataset are proposed to improve the accuracy. The test case is carried out on NASA's benchmark for aero-engine degradation. The comparison with other methods shows that the proposed method can significantly improve the remaining useful life estimation accuracy, which suggests its effectiveness in aero-engine degradation evaluation.

6. Future Work

Just as Simon Vollert and Andreas Theissler said in [15], it is one of the disadvantages that neural networks or deep learning based approaches cannot generally provide uncertainty quantification for their point predictions. However, uncertainty quantification that gives a distribution for RUL evaluation is critical to the use and maintenance of aero-engines. So, our future work mainly focuses on the uncertainty estimations of our method for the RUL evaluation. Here, a few trial ideas are introduced: First, drop-out module as a Bayesian approximation can be added to our network structures to model uncertainty in RUL evaluation as [30]. Second, Monte Carlo simulation can be carried out to give uncertainty analysis as [31].

Author Contributions: Conceptualization, L.R.; methodology, L.R. and Z.X.; software, L.R. and B.L.; validation, H.Q.; formal analysis, B.L.; writing—original draft preparation, L.R.; writing—review and editing, N.C.; visualization, B.L.; supervision, N.C.; funding acquisition, L.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Natural Science Foundation of Shandong Province (grant number ZR2021QE193).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

- Wei, Y.; Wu, D.; Terpenny, J. Robust incipient fault detection of complex systems using data fusion. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 9526–9534. [[CrossRef](#)]
- Urban, L.A.; Volponi, A.J. Mathematical methods of relative engine performance diagnostics. *SAE Trans.* **1992**, *101*, 2025–2050.
- Borguet, S.; Léonard, O.; Dewallef, P. Regression-based modeling of a fleet of gas turbine engines for performance trending. *J. Eng. Gas Turbines Power* **2016**, *138*, 138–152. [[CrossRef](#)]
- Sanchez de Leon, L.; Vega, J.M.; Montanes, J.L.; Rodrigo Ramirez, J. Gradient-like Minimization Methods for Signal Treatment and Aeroengines Diagnosis and Control ROM Methods and LSI Future Applicability. In Proceedings of the AIAA Propulsion and Energy 2020 Forum, Virtual Event, 24–28 August 2020; p. 3681.
- Ntantis, E.L.; Botsaris, P.N. Diagnostic Methods for an Aircraft Engine Performance. *J. Eng. Sci. Technol. Rev.* **2015**, *8*, 64–72. [[CrossRef](#)]
- Fong, S.; Narasimhan, S. An Unsupervised Bayesian OC-SVM Approach for Early Degradation Detection, Thresholding, and Fault Prediction in Machinery Monitoring. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–11. [[CrossRef](#)]
- Xiao, L.; Duan, F.; Tang, J.; Abbott, D. A Noise-Boosted Remaining Useful Life Prediction Method for Rotating Machines Under Different Conditions. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–12. [[CrossRef](#)]
- Zhang, C.; Lim, P.; Qin, A.K.; Tan, K.C. Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *28*, 2306–2318. [[CrossRef](#)] [[PubMed](#)]
- Li, Z.; Li, Y.; Yue, X.; Zio, E.; Wu, J. A Deep Branched Network for Failure Mode Diagnostics and Remaining Useful Life Prediction. *IEEE Trans. Instrum. Meas.* **2022**, *71*, 1–11. [[CrossRef](#)]
- Sateesh Babu, G.; Zhao, P.; Li, X.L. Deep convolutional neural network based regression approach for estimation of remaining useful life. In Proceedings of the International Conference on Database Systems for Advanced Applications, Dallas, TX, USA, 16–19 April 2016; pp. 214–228.
- Li, X.; Ding, Q.; Sun, J.Q. Remaining useful life estimation in prognostics using deep convolution neural networks. *Reliab. Eng. Syst. Saf.* **2018**, *172*, 1–11. [[CrossRef](#)]
- Wu, Y.; Yuan, M.; Dong, S.; Lin, L.; Liu, Y. Remaining useful life estimation of engineered systems using vanilla LSTM neural networks. *Neurocomputing* **2018**, *275*, 167–179. [[CrossRef](#)]
- Sayah, M.; Guebli, D.; Noureddine, Z.; Al Masry, Z. Deep LSTM enhancement for RUL prediction using Gaussian mixture models. *Autom. Control. Comput. Sci.* **2021**, *55*, 15–25. [[CrossRef](#)]
- Wang, Y.; Zhao, Y.; Addepalli, S. Remaining useful life prediction using deep learning approaches: A review. *Procedia Manuf.* **2020**, *49*, 81–88. [[CrossRef](#)]
- Vollert, S.; Theissler, A. Challenges of machine learning-based RUL prognosis: A review on NASA’s C-MAPSS data set. In Proceedings of the 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Västerås, Sweden, 7–10 September 2021; pp. 1–8.
- Fink, O.; Wang, Q.; Svensen, M.; Dersin, P.; Lee, W.J.; Ducoffe, M. Potential, challenges and future directions for deep learning in prognostics and health management applications. *Eng. Appl. Artif. Intell.* **2020**, *92*, 103678. [[CrossRef](#)]
- Sellers, J.F.; Daniele, C.J. *DYNGEN: A Program for Calculating Steady-State and Transient Performance of Turbojet and Turbofan Engines*; National Aeronautics and Space Administration: Washington, DC, USA, 1975; Volume 7901.
- Walsh, P.P.; Fletcher, P. *Gas Turbine Performance*; John Wiley & Sons: Hoboken, NJ, USA, 2004.
- Lucas, T.; Tallec, C.; Ollivier, Y.; Verbeek, J. Mixed batches and symmetric discriminators for GAN training. In Proceedings of the International Conference on Machine Learning, PMLR, Stockholm, Sweden, 10–15 July 2018; pp. 2844–2853.
- He, K.; Girshick, R.; Dollár, P. Rethinking imagenet pre-training. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 4918–4927.
- Saxena, A.; Goebel, K.; Simon, D.; Eklund, N. Damage propagation modeling for aircraft engine run-to-failure simulation. In Proceedings of the 2008 International Conference on Prognostics and Health Management, Denver, CO, USA, 6–9 October 2008; pp. 1–9.

22. Saxena, A.; Goebel, K. Turbofan Engine Degradation Simulation Data Set. Available online: <https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/> (accessed on 12 May 2021).
23. Heimes, F.O. Recurrent neural networks for remaining useful life estimation. In Proceedings of the 2008 International Conference on Prognostics and Health Management, Denver, CO, USA, 6–9 October 2008; pp. 1–6.
24. de Araujo, M.R.; Ribeiro, P.M.P.; Faloutsos, C. Tensorcast: Forecasting with context using coupled tensors (best paper award). In Proceedings of the 2017 IEEE International Conference on Data Mining (ICDM), New Orleans, LA, USA, 18–21 November 2017; pp. 71–80.
25. Zheng, S.; Ristovski, K.; Farahat, A.; Gupta, C. Long short-term memory network for remaining useful life estimation. In Proceedings of the 2017 IEEE International Conference on Prognostics and Health Management (ICPHM), Dallas, TX, USA, 19–21 June 2017; pp. 88–95.
26. Wang, J.; Wen, G.; Yang, S.; Liu, Y. Remaining useful life estimation in prognostics using deep bidirectional lstm neural network. In Proceedings of the 2018 Prognostics and System Health Management Conference (PHM-Chongqing), Chongqing, China, 26–28 October 2018; pp. 1037–1042.
27. Zhao, C.; Huang, X.; Li, Y.; Yousaf Iqbal, M. A Double-Channel Hybrid Deep Neural Network Based on CNN and BiLSTM for Remaining Useful Life Prediction. *Sensors* **2020**, *20*, 7109. [CrossRef] [PubMed]
28. Song, T.; Liu, C.; Wu, R.; Jin, Y.; Jiang, D. A hierarchical scheme for remaining useful life prediction with long short-term memory networks. *Neurocomputing* **2022**, *487*, 22–33. [CrossRef]
29. Nie, L.; Xu, S.; Zhang, L.; Yin, Y.; Dong, Z.; Zhou, X. Remaining Useful Life Prediction of Aeroengines Based on Multi-Head Attention Mechanism. *Machines* **2022**, *10*, 552. [CrossRef]
30. Gal, Y.; Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Proceedings of the International Conference on Machine Learning, PMLR, New York City, NY, USA, 19–24 June 2016; pp. 1050–1059.
31. Hou, T.; Nuyens, D.; Roels, S.; Janssen, H. Quasi-Monte Carlo based uncertainty analysis: Sampling efficiency and error estimation in engineering applications. *Reliab. Eng. Syst. Saf.* **2019**, *191*, 106549. [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.