

Article



Gas Turbine Engine Identification Based on a Bank of Self-Tuning Wiener Models Using Fast Kernel Extreme Learning Machine

Feng Lu * ^(D), Yu Ye and Jinquan Huang *

Jiangsu Province Key Laboratory of Aerospace Power Systems, College of Energy & Power Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, Jiangsu, China; nuaayyu@126.com

* Correspondence: lfaann@nuaa.edu.cn (F.L.); jhuang@nuaa.edu.cn (J.H.);

Tel.: +86-025-8489-5337 (F.L.); +86-025-8489-2203 (J.H.)

Received: 4 August 2017; Accepted: 1 September 2017; Published: 8 September 2017

Abstract: In order to simultaneously obtain global optimal model structure and coefficients, this paper proposes a novel Wiener model to identify the dynamic and static behavior of a gas turbine engine. An improved kernel extreme learning machine is presented to build up a bank of self-tuning block-oriented Wiener models; the time constant values of linear dynamic element in Wiener model are designed to tune engine operating conditions. Reduced-dimension matrix inversion incorporated with the fast leave one out cross validation strategy is utilized to decrease computational time for the selection of engine model feature parameters. An optimization algorithm is no longer needed compared to the former method. The contribution of this study is that a more convenient and appropriate methodology is developed to describe aircraft engine thermodynamic behavior during its static and dynamic operations. The methodology is evaluated in terms of computational efforts, dynamic and static estimation accuracy through a case study involving data that are generated by general aircraft engine simulation. The results confirm our viewpoints in this paper.

Keywords: gas turbine engine; system identification; block-oriented model; kernel extreme learning machine; reduced-matrix inversion

1. Introduction

Gas turbines are complicated thermodynamic machines and widely used as the power supply for commercial and military planes. The modeling and simulation of aircraft engines is a general methodology for optimization of design, performance and maintenance [1,2]. The engine operating parameters such as speed, pressure, temperature along the engine flow path are separately measured by the full authority digital electric control (FADEC) and engine monitoring unit (EMU) for control and health management [3,4]. An accurate on-board model is capable of providing real-time realistic estimations of aircraft engine operating parameters. The unmeasurable state parameters representing the health condition of engine modules can be calculated from the residuals between the real measurements of these operating parameters and their predicted values from an on-board engine model [5–7]. In addition, the engine model provides a redundancy channel of operating parameters [8]. For sensor fault tolerant control, this channel value can replace a faulty sensor signal used for feedback control [9,10]. An engine model with real-time performance and high confidence of both dynamic and static behavior becomes a key factor in aircraft control methodology and is of great interest.

There are two groups of engine models: physics-based and data-based models. The former models are established using information about the engine physics, such as characteristic maps, thermodynamic relationships, mass, momentum and energy balance [11,12]. Numerous studies have been conducted on the development of these aero-thermodynamic engine models, such as

Gas turbine Simulation Program (GSP) [13], GasTurb [14], Commercial Modular Aero-Propulsion System Simulation (CMPASS) [15], Numerical Propulsion System Simulation (NPSS) [16] and GGTS (general aircraft simulation) [17] on software platforms of C++ 6.0, JAVA 5, MATHEMATICA 8.0 and Simulink/MATLAB 6.0. The physics-based models sufficiently describe the engine static and dynamic behavior, but there are drawbacks; performance is easily affected by the engine component maps and empirical parameters from extensive testing, and the computational efforts are relatively serious [18]. Linearization is introduced to achieve a real-time engine model because of computational complexity and the time consumption of the aero-thermodynamic model.

The data-based models directly disclose the relationships between engine inputs and outputs derived from the sensed operating data by mathematical tools. The data-based model parameters or the functions between the inputs and outputs fit in-service operating data, but do not rely on any physical concepts of the aircraft engine. Machine learning algorithms [19,20] and polynomial regressions [21] are well known mathematical approaches to depict the engine static and dynamic operating process in the data-based modeling field. Polynomial regression has a simpler structure and less modeling parameters to determine to describe an aircraft's behavior [5,21]. The machine learning methods such as neural network (NN) and support vector machine (SVM) can be used for complex multiple-output system modeling [22,23], while the regression method is limited to single-output system modeling.

The block-oriented nonlinear model is one of the data-based modeling methods and has had more attention due to its simplicity and clear structure [24,25]. These models have both linear and nonlinear parts; the nonlinear static part is usually assumed continuous and differentiable. The block-oriented model parameters are identified by the polynomial regression method [24]. A number of attempts have been made to simplify model structure parameters of linear/nonlinear elements and employ appropriate parameterization methods to improve estimation accuracy and reduce computational costs [26]. Recently, an enhanced Wiener model (WM) method integrated the NN and optimization algorithms performed on the gray-box assumption [25], and adapted some linear dynamic coefficients to operating conditions to better describe the engine model.

This article proposes a novel block-oriented nonlinear model based on a bank of self-tuning WMs (SWMs) using fast kernel extreme learning machine (FKELM), and performance comparisons to predict engine operating parameters are carried out. The proposed engine model consists of four adaptive linear dynamic parts related to each nonlinear static element by a proper aero-thermodynamic correlation, and it represents the engine static and dynamic behavior in the form of a single input and multiple outputs (SIMO) system. Each linear dynamic part is reduced to a first-order inertia section, and the time constant values are tuned to the engine operating state. The goal of the innovation in the FKELM is to obtain appropriate model feature parameters, including kernel parameters of KELM and block-oriented regression factors with less running time. The following research is undertaken by the authors at Nanjing University of Aeronautics and Astronautics, China in collaboration with the lab at the University of Toronto, Canada. The simulations prove that this methodology is superior to the existing methods in terms of representing static and dynamic behavior of an engine.

The paper is organized as follows: Section 2 presents the self-tuning Wiener model with block structure and the improved KELM algorithm for SWM. The bank of SWMs is designed, and it is employed to model aircraft engine identification in Section 3. In Section 4, simulation and analysis indicate the improved performance of the proposed methodology with regards to estimation accuracy and computational time. Section 5 draws conclusions and discusses future research directions.

2. Self-Tuning Wiener Model

2.1. Block-Oriented Self-Tuning Wiener Model

Block-oriented models were first proposed by Narenda, and have been extensively applied to complex nonlinear systems [27]. Linear and nonlinear parts interconnect to form an open-loop

single branch block-structure model; it makes the description of the dynamic and static behavior easy [24]. There are three kinds of these block-oriented models, including Hammerstein, Wiener and Wiener-Hammerstein models [28].

The number of blocks in Wiener-Hammerstein is relatively large, and data to determine the model parameters of the superimposed model are hard to produce [29]. A Hammerstein model is often used as an approximate expression for a system where nonlinearity only resulted from the change of direct-current gain with input amplitude; this model cannot depict aircraft engine operating process due to the engine dynamics distinctly varied with the input amplitude. Compared to the Hammerstein model, the Wiener model can show variations of dynamic behavior for different input amplitudes. The Wiener model has a simpler structure than the Wiener-Hammerstein model, and it requires fewer model parameters to be computed. Hence, the Wiener model is used to depict engine dynamic behavior varying with operating conditions and is recommended for engine identification.

The identification of the nonlinear static part in a Wiener model can be determined from the steady-state relationship of the aircraft engine. The aero-thermodynamic operating data includes engine physical spool speed, capacity flow, total temperature and total pressure. The dynamic relationships of these operating data are generally simplified in a first-order delay [6,30]. Therefore, the linear dynamic block of the Wiener model is expressed in an inertial element of a transfer function:

$$G_i(s) = \frac{1}{\tau_i s + 1} \tag{1}$$

where the coefficient in this linear dynamic part is the time constant τ of the first-order inertial element, and *i* represents the *i*-th linear dynamic part related to the *i*-th engine model output. For the nonlinear static element of Wiener models, it is the nonlinear expression y = f(v). The nonlinear static part related to the nonlinear relationship of the engine operating input- output is represented by a lookup table.

The linear dynamic part is employed to depict system dynamic behavior in a Wiener model; the time constant τ of the transfer function Equation (1) is unchanged. However, previous experimental results on gas turbines show that the time constant of a first-order lag for the operating data varies with different engine operations. This model does not have enough tunable parameters to represent engine dynamic characteristics, and it is difficult to meet the requirements of engine dynamic accuracy. Therefore, a self-tuning Wiener model approach is introduced to modify the transfer function coefficients of the linear dynamic block to system operating conditions to improve flexibility and estimation accuracy of the Wiener model.

The engine spool speeds, low-pressure spool speed and high-pressure spool speed, are the important parameters that represent the engine operating state and are added as regression variables in the NARX (nonlinear auto-regressive exogenous input) model to describe varying time constants of the nonlinear feature:

$$\tau(t) = F(Y(t-1), \dots, Y(t-n_b), \tau(t-1), \dots, \tau(t-n_a), NH(t-1), NL(t-1))$$
(2)

where *Y* is the engine operating data to be estimated, and the regression factors n_a and n_b are the count of output and input terms. *NL* and *NH* are high pressure spool speed and low pressure spool speed, which are defined as the relative regression variable. The time constant can be adaptive to the engine operations to improve the dynamic description of the engine nonlinearity in Equation (2). In this aspect, a learning machine (LM) is introduced to perform this NARX in self-tuning WM (SWM). Figure 1 shows the training schematic of the self-tuning Wiener model; UD indicates unit delay and W_f is fuel flow.



Figure 1. The training schematic of self-tuning Wiener model related to one examined engine output.

The sample interval of the nonlinear static part is 0.02 s with the form of the lookup table. The regression factors n_a and n_b determining the model structure have an important impact on the performance of engine identification except for the type of nonlinear estimator [31]. The Enhanced Wiener Model (EWM) is developed to identify system for a two-shaft industrial gas turbine in [25], and the regression factors of the LM estimator embedded in the WM are calculated by trial and error. However, a systematic method to obtain the structure parameters of these Wiener models, especially in aircraft engine applications, has not been presented due to the problem-specific nature.

In this paper, the GA-KELM is first extended from the EWM framework, where the genetic algorithm (GA) as a well-known optimization technique is combined with the KELM. The regressor factors of GA-KELM are gained by trial, and it has the similar training schematic as the EWM. It is noted that both the invasive weed optimization (IWO) and GA are the population-based optimization algorithms, and it is hard to reach a steady global optimal result in every searching calculation due to their stochastic nature. The optimization process needs to be performed several times to generate the appropriate result. In addition, the optimization algorithm used to tune the LM model parameters like weighting vector and biases consumes large time. With regards to computational efforts, static and dynamic estimation accuracy, a novel KELM especially for the SWM is proposed without use of population-based optimization techniques.

2.2. Improved KELM for Self-Tuning Wiener Model

Neural networks play an important role in machine learning field and have been widely applied in engineering. In the last decade, the ELM algorithm is developed by Huang from the NN [32]. The input weight and biases related to ELM hidden layer nodes are randomly generated, and thus less model parameters to be calculated leads to much faster learning speed. The kernel function mapping from input space to feature space by the kernel transformation is then introduced into the ELM to form the KELM, and it brings better steady calculation [33].

Given *N* samples $\{(x_i, y_i) | i = 1, 2, ..., N\}$, wherein $x_i = [x_{i1}, x_{i2}, ..., x_{in}]^T \in \mathbb{R}^n$ is the input and y_i is the output. The learning problem is to search an optimal function $F : x_i \to y_i$, and it is presented by $F(x_i) \approx y_i$. Once the number of hidden layer node is *L*, the function *F* is expressed as:

$$F(\mathbf{x}_i) = \sum_{j=1}^{L} \beta_j g(\mathbf{a}_j \cdot \mathbf{x}_i + b_j) \qquad i = 1, 2, \dots, N$$
(3)

where $a_j = [a_{1j}, a_{2j}, ..., a_{nj}]^T$ is the input weighting vector between the *j*-th hidden and input nodes, b_j is the bias of the *j*-th hidden node, β_j is the output weighting vector between the *j*-th hidden and output nodes. The expression $a_j x_i$ indicates the vector inner product and g() is activation function of hidden layer node.

A regularization factor *C* referred to SVM algorithm [33] is to produce a new risk functional $R = ||\beta|| + C||H\beta - y||$, and thus the learning problem is:

$$\begin{split} \min_{\boldsymbol{\beta}} \frac{1}{2} \|\boldsymbol{\beta}\|_{2}^{2} + \frac{C}{2} \sum_{i=1}^{N} \varepsilon_{i}^{2} \\ s.t. \quad \boldsymbol{h}^{T}(\boldsymbol{x}_{i})\boldsymbol{\beta} = y_{i} - \varepsilon_{i}, \quad i = 1, 2, \dots, N \end{split}$$

$$(4)$$

The factor *C* is the weight assignment of empirical risk and structural risk. The slack variable ε_i implies the differences between the predicted and actual value. The optimal output weight of Equation (4) is

$$\boldsymbol{\beta} = \boldsymbol{H}^{T} \left(\frac{1}{C} \boldsymbol{I}_{N} + \boldsymbol{H} \boldsymbol{H}^{T} \right)^{-1} \boldsymbol{y}$$

$$\boldsymbol{H}(\boldsymbol{a}_{i}, \dots, \boldsymbol{a}_{L}, \boldsymbol{b}_{i}, \dots \boldsymbol{b}_{L}) = \begin{bmatrix} g(\boldsymbol{a}_{1}^{T} \boldsymbol{x}_{1} + \boldsymbol{b}_{1}) & \dots & g(\boldsymbol{a}_{L}^{T} \boldsymbol{x}_{1} + \boldsymbol{b}_{L}) \\ \vdots & \dots & \vdots \\ g(\boldsymbol{a}_{1}^{T} \boldsymbol{x}_{N} + \boldsymbol{b}_{1}) & \dots & g(\boldsymbol{a}_{L}^{T} \boldsymbol{x}_{N} + \boldsymbol{b}_{L}) \end{bmatrix}_{N \times L}$$
(5)

Compared to the ELM, the KELM adds a regularization term I_N/C . In addition, the kernel transformation $k(x_i, x_j)$ replaces an explicit expression in the input space of the ELM.

$$K = HH^{T} : K_{i,j} = h(\mathbf{x}_{i}) \cdot h(\mathbf{x}_{j}) = k(\mathbf{x}_{i}, \mathbf{x}_{j})$$

$$k(\mathbf{x}_{i}, \mathbf{x}_{j}) = \exp(-\gamma ||\mathbf{x}_{i} - \mathbf{x}_{j}||^{2})$$
(6)

where the kernel parameter γ indicates the kernel distribution width. The output function in Equation (3) can be calculated

$$F(\mathbf{x}) = \mathbf{h}^{T}(\mathbf{x})\mathbf{\beta} = \mathbf{h}^{T}(\mathbf{x})\mathbf{H}^{T}(\frac{1}{C}\mathbf{I}_{N} + \mathbf{H}\mathbf{H}^{T})^{-1}\mathbf{y}$$

$$= \begin{bmatrix} \mathbf{k}(\mathbf{x}, \mathbf{x}_{1}) \\ \vdots \\ \mathbf{k}(\mathbf{x}, \mathbf{x}_{N}) \end{bmatrix}^{T} (\frac{1}{C}\mathbf{I}_{N} + \mathbf{K})^{-1}\mathbf{y} = \mathbf{k}(\mathbf{x})\mathbf{\alpha}$$

$$\mathbf{\alpha} = (\frac{1}{C}\mathbf{I}_{N} + \mathbf{K})^{-1}\mathbf{y}$$
(7)

where α is the KELM output weight. The detailed KELM algorithm can be referred to [34].

The model feature parameters such as regressor factors and KELM kernel parameter γ should be given before the SWM implementation in engine model, which the keys to affecting SWM performance. Some Cross validation methods are employed to select these parameters, which essentially include hold-out, K-cross validation (KC) and leave one out cross validation (LOOC) methods. Among the methods above, the LOOC brings a better generalization performance, but computational time dramatically increases in the case of a large dataset [35].

For this purpose, an FKELM method is put forward where Fast LOOC is to improve the computational efforts in the framework of KELM theory. This methodology relies on block matrix inversion and the LOOC strategy. Provided the effects of *p*-th training sample is neglected, the estimate \tilde{y}_p and weighting vector $\boldsymbol{\alpha}_p$ of output can be written as:

$$\widetilde{y}_p = F_p(\mathbf{x}) = F(\mathbf{x}) = \sum_{j=1}^L \beta_j g(\mathbf{a}_j \cdot \mathbf{x} + b_j) = \mathbf{k}(\mathbf{x}) \mathbf{a}_p \quad p = 1, \cdots, N$$

$$\mathbf{a}_p = \mathbf{A}_p^{-1} \mathbf{Y}$$
(8)

where A_p is the one-dimension simplified matrix compared to original matrix A, and it is that the p-th column elements of A are zero but the p-th row element in this column invariant. That is to say, the contribution of the p-th sample to KELM output weighting vector can be almost neglected. The expression of these two matrices are given:

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1p} & \cdots & a_{1N} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{p1} & \cdots & a_{pp} & \cdots & a_{pN} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{N1} & \cdots & a_{Np} & \cdots & a_{NN} \end{bmatrix} A_{p} = \begin{bmatrix} a_{11} & \cdots & 0 & \cdots & a_{1N} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{p1} & \cdots & a_{pp} & \cdots & a_{pN} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{N1} & \cdots & 0 & \cdots & a_{NN} \end{bmatrix}$$
(9)

Define two vectors $\boldsymbol{u} = [a_{1p}, \dots, a_{(p-1)p}, 0, a_{(p+1)p}, \dots, a_{Np}]^T$ and $\boldsymbol{v} = [0, \dots, 0, -1, 0, \dots, 0]^T$. The *p*-th element of \boldsymbol{v} is -1, and the *p*-th element of \boldsymbol{u} is 0. We can obtain the following expression:

$$A_p = A + uv^T \tag{10}$$

The Sherman-Morrison-Woodbury equation is introduced:

$$\left(A + uv^{T}\right)^{-1} = A^{-1} - \frac{A^{-1}uv^{T}A^{-1}}{1 + v^{T}A^{-1}u}$$
(11)

The Equation (8) can be rewritten as follows using the conclusions of Equations (10) and (11):

$$\boldsymbol{\alpha}_p = (\boldsymbol{A}^{-1} - \frac{\boldsymbol{A}^{-1}\boldsymbol{u}\boldsymbol{v}^T\boldsymbol{A}^{-1}}{1 + \boldsymbol{v}^T\boldsymbol{A}^{-1}\boldsymbol{u}})\boldsymbol{Y} = \boldsymbol{\alpha} - \frac{\boldsymbol{A}^{-1}\boldsymbol{u}\boldsymbol{v}^T\boldsymbol{A}^{-1}\boldsymbol{Y}}{1 + \boldsymbol{v}^T\boldsymbol{A}^{-1}\boldsymbol{u}}$$
(12)

Furthermore, we can obtain the following two equations:

$$1 + \boldsymbol{v}^T \boldsymbol{A}^{-1} \boldsymbol{u} = \bar{\boldsymbol{a}}_{pp} \boldsymbol{a}_{pp} \tag{13}$$

$$\boldsymbol{A}^{-1}\boldsymbol{u}\boldsymbol{v}^{T}\boldsymbol{A}^{-1}\boldsymbol{Y} = \boldsymbol{L}\boldsymbol{a}_{pp}\boldsymbol{\alpha}(p) \tag{14}$$

where the vector $L = [\overline{a}_{1p}, \overline{a}_{2p}, \dots, (\overline{a}_{pp} \cdot a_{pp} - 1)/a_{pp}, \dots, \overline{a}_{Np}]^T$, and $\overline{a}_{ip}(i = 1, 2, \dots, N)$ is the corresponding subscript element of A^{-1} . $\alpha(p)$ is the *p*-th element of α , and the simplified output weight is calculated:

$$\boldsymbol{\alpha}_p = \boldsymbol{\alpha} - \frac{\boldsymbol{L}\boldsymbol{\alpha}(p)}{\overline{a}_{pp}} \tag{15}$$

Therefore, the inverse of simplified output weight vector can be reached directly by Equation (15), and no additional calculation about matrix inversion is needed. It implies that the number of inversion calculation can be reduced from N to 1, and the FKELM is obtained from the KELM combined with FLOOC (Fast LOOC) validation method. As we know, the matrix inversion calculation occupies the most calculation time, the reduction of matrix inversion count save lots of computational time in the training phase. It leads to greatly less computational efforts for selection of feature parameters by the FKELM. The detailed steps of this methodology are presented as follows:

- Step 1: Initialize model parameters, Let p = 1; Empirically produce a set of M candidates including the regressor factors n_a , n_b , and kernel parameter γ ; Select the KELM feature parameter combination $(n_a, n_b, \gamma)_k$ from candidate set, and k = 1.
- Step 2: Train the KELM using *N* samples to generate the matrices *A* and A^{-1} with full dimensions as the KELM parameter combination $(n_a, n_b, \gamma)_k$.
- Step 3: Calculate the simplified output weight vector α_p by Equation (15) and \tilde{y}_p by Equation (8).

Step 4: If p < N, then p = p + 1 and return to step 3; otherwise, compute the generalization performance index of KELM: align

$$RMSE_{k} = \sqrt{\frac{\sum\limits_{p=1}^{N} (y_{p} - \widetilde{y}_{p})^{2}}{N}}$$
(16)

Step 5: If k < M, then k = k + 1 and go back to step 2; otherwise, stop the procedure and select the optimal KELM feature parameter combination (*na*, *nb*, γ) and the output weight vector α who has the minimum value of $RMSE_k$.

The FKELM is used as the regression estimator to tune the model coefficients of linear dynamic part to engine operation for the SWM. The optimal model feature parameters and output weight in the SWM framework is obtained by the FKELM, and population-based optimization algorithm is no longer needed as the EWM.

3. Aircraft Engine Identification by Self-Tuning Wiener Models

The proposed methodology is tested on a virtual aircraft engine created from GGTS that is based on thermodynamic physical theory and with the architecture of engine component level model [11,36]. The GGTS is similar to CMAPSS, which is not available to Chinese now. GGTS provides various kinds of aircraft engine model, like turbojet, turbofan and turboshaft engine model and their simulation data, which has been well used in aviation industry corporation of China. The engine component characteristic maps and design operation data are loaded to GGTS to create a turbofan engine model (TEM) [14,37], which is coded using C language and packaged with a dynamic link library (DLL) for simulation in Matlab environment.

The TEM produces engine-sensed data comprising of snapshot measurements for dynamic behavior collected around design point. The available instrumentation at design operation is reported in Table 1, where the first four measurements define the engine operation and the rest flow path sensors could be used for FADEC or EMU. The examined engine operates at the condition of the International Standard Atmosphere (ISA).

A bank of self-tuning Wiener models is employed to identify the engine instead of conventionally a single stochastic model. There are one input (W_f) and four outputs (NL, NH, W_3 , and Exhaust Gas Temperature (*EGT*)) considered in the examined engine model. The variables like *H*, *Ma* and atmosphere condition are not included in the model inputs as in the International Standard Atmosphere condition at ground. Thus this engine model is a SIMO system, and consists of four SWMs related to four outputs. Each SWM is developed based on the blocked-oriented Wiener model, time constant of which is self-tuned to operating conditions by the FKELM. Figure 2 shows a bank of SWMs for low bypass ratio aircraft engine.

Label	Description	Value
Н	Altitude	0 m
Ma	Mach number	0
W_{f}	Fuel flow	2.48 kg/s
A_8	Throttle area	0.2597 m ²
NL	Physical low pressure spool speed	10,302 r/min
NH	Physical high pressure spool speed	13,340 r/min
W_3	Air flow	75.6594 kg/s
EGT	Exhaust gas temperature	1157.34 K

Table 1. Design point specifications of the examined engine.

Aircraft engine spool speeds *NL* and *NH* are the most important parameters representing its operating state [36]. Besides, the nature of speed sensors installed in the engine is high accuracy and quick response, and there is far less time lag than flow and temperature sensors. Hence, the last step of spool speeds act as the regression inputs not only for the speed SWMs themselves but also for the SWMs of W_3 and *EGT*. The fuel air ratio (*FAR*) is employed as the input for the SWM of *EGT*, which is different from the other SWMs using W_f . The FAR serves as control variable of the speed control loop in the FADEC, which can effectively reduce engine surge probability and lead to change the specific exhaust gas emissions. Hence, *FAR* is recognized as one of the inputs in the *EGT* SWM. The Quasi-amplitude-modulated pseudo random binary sequence (APRBS) data is loaded to learn each SWM [38], which is developed from a well-used excitation signal for nonlinear system identification [39].



Figure 2. The bank of self-tuning Wiener models for the examined engine.

The procedure schematic to engine identification by the proposed methodology is presented in Figure 3. The TEM is to generate the data for both of training and testing, which is developed from GGTS using the design point data and component maps. The training data are excited using Quasi-APRBS to the TEM, and then sorted and removes one sample out to obtain y_p for training each FKELM. The initial combinations of model feature parameters are fed into each FKELM corresponding to its SWM to obtain the optimal model structure, kernel parameter and output weighting vector. The linear dynamic element with tuned time constant and the nonlinear static part are together formed the SWM related to each model output. Then the identified engine model represented by the bank of SWMs is created based on the integration of four SWMs as given in Figure 2. In the testing process, the test data brought from the TEM is to evaluate performance of the established engine model.



Figure 3. The procedure schematic of the methodology for aircraft engine identification.

4. Simulation and Analysis

4.1. FKELM Performance Evaluation

The FKELM is compared against LOOC-KELM and KC-KELM [36] in this section. The involved algorithms run in Matlab Software R2016b, and the computer hardware used for simulation is configured as follows: CPU i3-550 @ 3.20 GHz and RAM 4GB. The computational efforts of three cross validation methods are first assessed using the benchmark data set of "sinc". The expression of artificial data set "sinc" is:

$$f(x) = \begin{cases} \sin x/x, & x \neq 0 \\ 1, & x = 0 \end{cases}$$
(17)

There are *M* training samples generated in the range of (-10, 10), and the scales of these samples are separately 500, 1000, 1500, 2000, 2500, 3000. The regularization factor *C* is 0.5. Table 2 gives the computational efforts comparisons of the three methods. We can see that the computational time of LOOC is 0.75 s as data size 500 and 277.29 s as size 3000, and it adds evidently as dataset scale increases. Both of the FLOOC and KC consume less time than the LOOC especially in large data size cases, while the FLOOC is about half of that of KC in all cases. Hence, the FLOOC method is the best one for fast computational evaluation in the large scale dataset.

Table 2. The computational efforts of three cross validation method to benchmark dataset "sinc" (s).

Method	500	1000	1500	2000	2500	3000
LOOC	0.7491	7.7227	33.8026	74.1476	151.0138	277.2857
KC	0.1641	0.7006	3.1656	2.9968	5.1389	8.0324
FLOOC	0.0851	0.2268	1.2960	1.4037	2.5874	3.8879

The generalization performance of proposed FKELM is compared with those of LOOC-KELM and KC-KELM using two benchmark datasets "Boston housing" and "Abalone", which are taken from UCI Machine Learning Repository. The sample scales of the "Boston housing" and "Abalone" are

separately 455 and 1000, and the data are normalized to the range of [-1, 1]. Two kernel functions, namely Gaussian kernel in Equation (6) and asymptotic kernel in Equation (18) [34], are discussed:

$$k(x,y) = \frac{2}{\pi} \arcsin\frac{1 + \langle x, y \rangle}{\sqrt{(1/\gamma^2 + 1 + \langle x, x \rangle)(1/\gamma^2 + 1 + \langle y, y \rangle)}}$$
(18)

The kernel parameter γ are empirically selected 10^{-2} , 10^{-1} , 10^{0} , 10^{1} and 10^{2} . The *RMSE* used as generalized performance indices of these KELM algorithms defined by Equation (16), and the results are shown in Table 3.

Dataset	Ŷ	Gaussian Kernel			Asymptotic Kernel			
Dutubet	,	LOOC-KELM	KC-KELM	FKELM	LOOC-KELM	KC-KELM	FKELM	
	10^{-2}	0.41620	0.43200	0.45100	0.20877	0.19978	0.22111	
	10^{-1}	0.26079	0.37001	0.27346	0.10276	0.11424	0.11406	
Boston housing	10^{0}	0.09068	0.16412	0.09078	0.08604	0.10308	0.07717	
	10^{1}	0.07475	0.10137	0.07676	0.07375	0.10302	0.07170	
	10^{2}	0.10673	0.11301	0.10885	0.07493	0.10328	0.07259	
Abalone	10^{-2}	0.26021	0.26153	0.26066	0.17534	0.18383	0.18874	
	10^{-1}	0.12178	0.12364	0.11851	0.09174	0.09939	0.09623	
	10^{0}	0.09149	0.09912	0.09316	0.08744	0.09720	0.09188	
	10^{1}	0.08934	0.09720	0.09218	0.08961	0.09930	0.09215	
	10^{2}	0.09441	0.09888	0.09493	0.09367	0.09941	0.09308	

Table 3. The RMSE of various KELMs to benchmark dataset "Boston housing" and "Abalone".

In Table 3, the *RMSE* of LOOC-KELM is 0.07475 and the minimum in the case of the dataset "Boston housing" when the Gaussian kernel γ is 10. The minimal *RMSE* in the same dataset is also generated as the Asymptotic kernel $\gamma = 10$. The KC-KELM and FKELM produce the minimal *RMSEs* as $\gamma = 10$ as the Gaussian and Asymptotic kernel used. As was mentioned earlier, LOOC-KELM has the best generalization performance once the number of train sample being infinite, and the FKELM is an approximate to LOOC-KELM whose *p*-th row element effect to inversed matrix is ignored. The *RMSE* by FKELM is much closer to that by LOOC-KELM compared to KC-KELM. In the benchmark dataset "Abalone", the similar conclusion can be gained. Consequently, the proposed FKELM is the best choice among the examined KELM algorithms in terms to the computational efforts and generalized performance, and it will be applied to tune the linear dynamic part of SWM.

4.2. Engine Identification Application of the Methodology

Aircraft engine dynamic and static behavior is identified by the self-tuning Wiener model (FSWM), and the dynamic performance of the FKELM self-tuning Wiener model (FSWM) is compared to the WM, NN-WM, EWM and GA-SWM, while the static performance of FSWM is tested against the NN. The basic WM has the fixed time constant of linear dynamic part. The NN-WM has the similar architecture as the EWM, and the difference is that the EWM parameters is adjusted by the IWO algorithm but no optimization algorithm is employed to tune those of the NN-WM in training phase. The design parameters of the involved algorithms are given as [25]. The FKELM and basic KELM are separately used as regression estimators to yield the time constant of linear dynamic element in FSWM and GA-SWM, and the GA is no longer to tune the model parameters for the FSWM which is different from the GA-SWM.

The fuel flow is generated using Quasi-APRBS, and flown into the TEM to produce the I/O data for engine identification given in Figure 4. The input data are given in the first plot and the output data in the rest plots in Figure 4. W_f change context is from 0.55 to 1, and the dwell time is randomly selected between 10 and 24.



Figure 4. The input and output data from TEM for aircraft engine identification.

500 t/s

The Quasi-APRBS engine data totally 1000 s are divided into two subsets, which are the former 800 s for training samples and the remaining 200 s for testing samples. The performance evaluation function of the engine models for the purpose to search the optimal structure parameters is given:

$$J = \left(\frac{1}{w_1 + w_2}\right) \left(w_1 \frac{(|PC|)^{-1}}{N_1} + w_2 \frac{EP_{max}}{N_2} \right)$$

$$PC = \left(1 - \frac{||Y - \hat{Y}||}{||Y - \overline{Y}||}\right) \times 100$$

$$EP_{max} = \max_{1 \le i \le D} \left(\left| \frac{Y(i) - \hat{Y}(i)}{Y(i)} \right| \times 100 \right)$$

$$EP_{mean} = \sum_{i=1}^{D} \left(\left| \frac{Y(i) - \hat{Y}(i)}{Y(i)} \right| \times 100 \right) / D$$
(19)

where the *PC* is percentage of compliance, EP_{max} is maximum error percentage, EP_{mean} is mean error percentage, and *D* is *Y* dimension. The weighting coefficient w_i determines the contribution of different error indices (*PC* and EP_{max}) to the objective function, and w_1 and w_2 are 0.5 in this study. Two vectors *Y* and \hat{Y} are separately the desired and estimated of the engine outputs, and \overline{Y} is the mean of vector *Y*.

The optimal estimator regressor factors n_a , n_b of the NNWM, EWM, GA-SWM are calculated with the minimal evaluation function as Equation (19) by several tries. The number of regressor factors n_a and n_b are separately changed in the intervals [1, 5] and [0, 5] by 1. The specification results of the optimal structure for different Wiener models are reported in Table 4, and the basic WM is not included because of no regression estimator used. It is noted that the FSWM structure parameters are obtained in the FKELM training phase, and at the same time the output weighting vector and kernel parameter are gained. In the FKELM training process, the evaluation function Equation (19) replaces Equation (16) as new generalization performance index. The kernel parameter varies in an interval as given in Section 4.1.

Table 4. Specifications of the optimal structure for Wiener models.

Model Trues		Number of R	egressors (n _a , n	ı _b)
widdel Type	NL	NH	W_3	EGT
NN-WM	(1, 1)	(1, 1)	(1, 2)	(0, 2)
EWM	(1, 1)	(1, 1)	(1, 1)	(0, 2)
GA-SWM	(1, 1)	(1, 2)	(1, 1)	(0, 2)
FSWM	(1, 1)	(1, 2)	(1, 2)	(1, 1)

The time constant of linear dynamic part in each SWM is continuously varied with aircraft engine operations, which are presented in Figure 5. The time constant variation indicates that the tunable dynamic characteristics in the linear part make the block-oriented Wiener model more flexible.



Figure 5. The time constant variation of linear dynamic block in the FSWM: (**a**) *NL*; (**b**) *NH*; (**c**) *W*₃; (**d**) *EGT*.

Table 5 gives the dynamic performances of five model identification methods against the training and testing data with regards to the performance indices (*PC*, EP_{max} , EP_{mean} , T_{train} and T_{test}).

Outputs		Train Mode				Test Mode			
	Methods	<i>PC</i> %	EP _{mean} %	EP _{max} %	T _{train} (s)	<i>PC</i> %	EP _{mean} %	EP _{max} %	T_{test} (s)
	WM	89.3425	1.2931	3.1574	6.5430	89.1452	1.3952	4.6521	0.0666
	NN-WM	91.6513	0.7611	3.2982	13.6896	91.0735	0.8211	3.3837	0.0980
NL	EWM	91.8945	0.6088	3.5432	25.4344	91.5883	0.6401	3.5615	0.1830
	GA-SWM	92.1245	0.5042	2.3628	19.5433	91.9654	0.5942	2.5627	0.0790
	FSWM	92.0860	0.4948	2.4983	12.5448	92.9837	0.5401	2.5875	0.0772
NH	WM	91.5461	0.1288	1.3294	6.8910	91.5142	0.2547	1.9017	0.0762
	NN-WM	92.8771	0.1093	1.0712	13.8384	92.0011	0.3223	1.4396	0.0991
	EWM	93.1091	0.0412	0.9093	26.0196	92.7981	0.2198	1.3015	0.1855
	GA-SWM	93.2503	0.0461	0.8872	20.3589	93.0218	0.2021	1.0637	0.0903
	FSWM	93.3025	0.0485	0.9016	13.0118	92.9824	0.1977	1.1174	0.0896
	WM	91.3657	1.8426	3.1154	7.0230	90.7521	2.3594	5.4214	0.0649
W3	NN-WM	94.1293	1.4875	3.5860	14.0188	93.3703	2.3100	4.8191	0.0973
	EWM	94.4033	1.2421	3.0077	25.8054	93.1342	1.9801	4.8994	0.1896
	GA-SWM	95.4259	1.2543	2.6931	19.7997	94.6555	1.8224	4.2287	0.0815
	FSWM	95.3912	1.2104	2.8706	12.3192	94.7123	1.8753	4.0411	0.0797
EGT	WM	89.5422	2.5423	6.3922	6.5970	88.2763	3.2452	7.4682	0.0674
	NN-WM	90.3145	1.5593	4.7812	14.1670	89.8872	1.8557	5.9163	0.0986
	EWM	90.6713	1.2431	4.0913	26.0361	90.2201	1.7411	5.0617	0.1879
	GA-SWM	91.1222	1.2933	3.7534	20.1074	90.1244	1.8875	4.9018	0.0833
	FSWM	91.0479	1.2706	3.7921	13.0536	90.1975	1.9452	4.9533	0.0819

Table 5. The dynamic performance indices of train and test mode by five Wiener models.

The Wiener models with tuned time constant have obviously better estimation accuracy than the basic WM from three former indices in Table 5. The block-oriented Wiener model using KELM including GA-SWM and FSWM, have superior estimation accuracy to NNWM and EWM. The GA-SWM and FSWM have almost the same dynamic estimation performance.

When computational efforts are concerned, WM's train time is the least, and the same conclusion is to the test time. There are no regression estimator parameters to be tried or trained in the basic WM. The FSWM is the best one among four Wiener models with tuned time constant. All FSWM parameters are generated in training stage of FKELM, and no optimization algorithm is involved. In the FSWM, the model feature parameters selection and FKELM output weight vector tuning are not stepwise performed. These indicate that the proposed FSWM strategy is an effective way to gain system parameters.

The NN performance indices are not presented, since the conclusion that NN's dynamic estimation performance inferior to EWM had been proved in [25]. The averages of engine model outputs in terms to three indices (\overline{PC} , EP_{max} , and EP_{mean}) are shown in Figure 6, where \overline{PC} is defined by $\overline{PC} = 1 - PC$. The Wiener models with tuned time constant have obviously less estimation error percentage than the basic WM both in the train mode and test mode as seen from Figure 6.



Figure 6. Comparisons of five Wiener models using the average performance indices: (**a**) Train mode; (**b**) Test mode.

The steady estimation error is another key index to evaluate performance of proposed modeling approach. Fifteen operating points above the idle condition are used, at which the steady errors are calculated. Figure 7 gives the steady estimation errors of the NN and block-structured FSWM methods. The FSWM brings few steady errors while NN produces maximum errors close to 3% for *NL*, *NH*, *W*₃, and *EGT* estimation around the idle from Figure 7. The NN's steady error approaches to that of FSWM at the engine operation near design point.

The basic WM and other tunable Wiener models developed on the basis of block structure, which have nonlinear static part to represent static behavior. Hence, the block-oriented Wiener models have almost the same steady performance, and only FSWM is given to compare the NN in Figure 7.



Figure 7. The steady error comparisons of identified engine model between the NN and FSWM: (**a**) NL; (**b**) NH; (**c**) W_3 ; (**d**) EGT.

Since the identified engine model is a SIMO system, it runs with an architecture of the bank of SWMs. In order to further evaluate the engine model identified by FSWM, the static and dynamic behavior joint tests are carried out. The time constant of all linear dynamic elements are simultaneously modified by regression estimator. Figure 8 indicates a more detailed FSWM performance compared to the EWM for the *NL*, *NH*, *W*₃ and *EGT* estimation, and the simulation time is 200 s on the joint test. The engine models identified by both of the EWM and FSWM have acceptance agreement with the target TEM. Compared to the EWM, the model outputs by the FSWM is more close to the TEM outputs in Figure 8. It indicates that the FSWM produces less modeling errors, and the FSWM has better dynamic estimation accuracy than the EWM. The FSWM produces similar steady estimation accuracy as the EWM, and it coincides with the previous ones in module tests.



Figure 8. Cont.



Figure 8. Variations of aircraft engine model outputs in joint test: (a) NL; (b) NH; (c) W₃; (d) EGT.

5. Conclusions

This paper develops a systematic approach which includes model structure selected, parameter tuned and model integrated steps that lead to an improved model identification. The novelty of this methodology lies in the development of FKELM algorithm and the SWMs bank architecture, and the combination of these techniques for aircraft engine modeling named the FSWM is proposed. One advantage of this methodology is that the dynamic estimation performance becomes more accurate as the tunable time constant of linear dynamic element compared with the fixed value in the basic WM. The steady estimation accuracy of FSWM is superior to the traditional machine learning method due to the nonlinear static element held in the block-oriented WM. Another advantage of this methodology is that the global optimal model feature parameters and weighting vector obtained directly by the training phase can evidently reduce computational time-consuming, and no population-based optimization algorithm is employed to tune the model parameters compared to the EWM. This indicates the bank of SWMs could be a more potential model-based tool for on-board real time control and fault diagnosis.

The methodology is tested and validated using the measurement data of a low-bypass ratio aircraft engine which are established from GGTS. Both of the SWM related to each engine output and the bank of SWMs as an entire engine outputs are numerically evaluated against the engine simulation data. The methodology developed in this paper is not limited to dual-spool engine, but also can be applied to other engine types. The model feature parameters including structure parameters of regressor and kernel parameters of KELM are the key elements to affect the FSWM performance to the aim of aircraft engine system identification. The proposed FKELM brings an effective way to rapidly select these model parameters altogether with the output weighting vector during the training process.

This research points out a new direction in block-oriented Wiener model identification by proposing an appropriate bank of SWMs using FKELM technique that is specifically beneficial for aircraft engine modeling applications. Although the FKELM produces less dynamic modeling errors than the EWM in dynamic tracking simulation, the maximum dynamic errors still more than 0.01 in Figure 8. There are several important topics for further study related to this research. First, the transfer functions for four outputs are all simplified to one-order inertial expressions, and it is better to utilize a two-order function to approximate a two-spool gas turbine engine. Further works can be performed to investigate the performance when the order of linear dynamic element is increased and each SWM has different transfer function order. Second, although this paper focuses on the SIMO system identification for gas turbine engine at ground, extensions to the cases that have more than the examined aero-thermodynamic parameters in this study or more wide operational condition in flight envelope are worthy of future exploration.

Acknowledgments: We are grateful for the financial support of the National Nature Science Foundation of China (No. 61304113), and China Outstanding Postdoctoral Science Foundation (No. 2015T80552). Gratitude is also extended to Prof. Viliam Makis for useful suggestions and to China Scholarship Council for supporting the first

author to carry out collaborative research in the Department of Mechanical and Industrial Engineering at the University of Toronto.

Author Contributions: Feng Lu and Jinquan Huang contributed in developing the ideas of this research, Yu Ye performed this research. All of the authors were involved in preparing this manuscript.

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

References

- 1. Mohammadi, E.; Montazeri-Gh, M. Performance enhancement of global optimization-based gas turbine fault diagnosis systems. *J. Propuls. Power* **2016**, *32*, 214–224. [CrossRef]
- 2. Zhong, S.S.; Luo, H.; Lin, L. An improved correlation-based anomaly detection approach for condition monitoring data of industrial equipment. In Proceedings of the IEEE International Conference on Prognostics and Health Management (ICPHM), Ottawa, ON, Canada, 20–22 June 2016.
- 3. Dimogianopoulos, D.; Hios, J.; Fassois, S. Aircraft engine health management via stochastic modelling of flight data interrelations. *Aerosp. Sci. Technol.* **2012**, *16*, 70–81. [CrossRef]
- 4. Qi, Y.W.; Bao, W.; Chang, J.T. State-based switching control strategy with application to aeroengine safety protection. *J. Aerosp. Eng.* **2015**, *28*, 04014076. [CrossRef]
- 5. Borguet, S.; Leonard, O.; Dewallef, P. Regression-based modeling of a fleet of gas turbine engines for performance trending. *J. Eng. Gas Turbines Power* **2016**, *138*, 021201. [CrossRef]
- 6. Xue, W.; Guo, Y.Q. Aircraft Engine Sensor Fault Diagnostics Based on Estimation of Engine's Health Degradation. *Chin. J. Aeronaut.* **2009**, *22*, 18–21.
- Simon, D. A comparison of filtering approaches for aircraft engine health estimation. *Aerosp. Sci. Technol.* 2008, 12, 276–284. [CrossRef]
- 8. Wen, H.; Zhu, Z.H.; Jin, D.; Hu, H. Model predictive control with output feedback for a deorbiting electrodynamic tether system. *J. Guid. Control Dyn.* **2016**, *39*, 2455–2460. [CrossRef]
- 9. Pourbabaee, B.; Meskin, N.; Khorasani, K. Robust sensor fault detection and isolation of gas turbine engines subjected to time-varying parameter uncertainties. *Mech. Syst. Signal Proc.* **2016**, *76*, 136–156. [CrossRef]
- 10. Chapman, J.W.; Lavelle, T.M.; May, R.D.; Litt, J.S.; Guo, T.H. Propulsion System Simulation Using the Toolbox for The Modeling and Analysis of Thermodynamic Systems (T-MATS). In Proceedings of the 50th AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Cleveland, OH, USA, 28–30 July 2014.
- 11. Sun, J.G.; Vasilyev, V.; Ilyasov, B. *Advanced Multivariable Control Systems of Aeroengines*; Beihang Press: Beijing, China, 2005.
- 12. Aretakis, N.; Roumeliotis, I.; Alexiou, A.; Romessis, C.; Mathioudakis, K. Turbofan Engine Health Assessment From Flight Data. *J. Eng. Gas Turbines Power* **2014**, *137*. [CrossRef]
- Visser, W.; Broomhead, M. GSP: A generic Object-Oriented Gas Turbine Simulation Environment. In Proceedings of the ASME Turbo Expo 2000: Power for Land, Sea, and Air, Munich, Germany, 8–11 May 2000.
- 14. Kurzke, J. GASTURB 12 User's Manual; GasTurb GmbH: Dachau, Germany, 2013.
- 15. Ruano, A.E.; Fleming, P.J.; Teixeira, C.; RodriGuez-Vazquez, C.K.; Fonseca, C.M. Nonlinear identification of aircraft gas-turbine dynamics. *Neurocomputing* **2003**, *55*, *551–579*. [CrossRef]
- 16. Lytle, J.; Follen, G.; Naiman, C.; Evans, A. *Numerical Propulsion System Simulation (NPSS)* 1999 Industry *Review*; NASA Glenn Research Center: Cleveland, OH, USA, 6–7 October 1999.
- 17. Zhou, W.X. Research on Object-Oriented Modeling and Simulation for Aeroengine and Control System. Ph.D. Thesis, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 2007.
- 18. Kyprianidis, K.G.; Sethi, V.; Ogaji, S.O.T.; Pilidis, P.; Singh, R.; Kalfas, A.I. Uncertainty in gas turbine thermo-fluid modelling and its impact on performance calculations and emissions predictions at aircraft system level. *Proc. Inst. Mech. Eng. Part G* **2012**, *226*, 163–181. [CrossRef]
- 19. He, K.; Xu, Q.S.; Jia, M.P. Modeling and Predicting Surface Roughness in Hard Turning Using a Bayesian Inference-Based HMM-SVM Model. *IEEE Trans. Autom. Sci. Eng.* **2015**, *12*, 1092–1103. [CrossRef]
- 20. Zaidan, M.A.; Mills, A.R.; Harrison, R.F.; Fleming, P.J. Gas turbine engine prognostics using Bayesian hierarchical models: A variational approach. *Mech. Syst. Signal Proc.* **2016**, *70*, 120–140. [CrossRef]
- 21. Basso, M.; Giarre, L.; Groppi, S.; Zappa, G. NARX models of an industrial power plant gas turbine. *IEEE Trans. Control Syst. Technol.* **2005**, *13*, 599–604. [CrossRef]

- 22. Wang, X.; Li, Q.; Xie, Z.H. New Feature Selection Method Based on SVM-RFE. *Adv. Mater. Res.* 2014, 926, 3100–3104. [CrossRef]
- 23. Fei, J.Z.; Zhao, N.B.; Shi, Y.; Feng, Y.M.; Wang, Z.W. Compressor performance prediction using a novel feed-forward neural network based on Gaussian kernel function. *Adv. Mech. Eng.* **2016**, *8*. [CrossRef]
- 24. Schoukens, J.; Pintelon, R.; Rolain, Y.; Schoukens, M.; Tiels, K.; Vanbeylen, L.; van Mulders, A.; Vandersteen, G. Structure discrimination in block-oriented models using linear approximations: A theoretic framework. *Automatica* **2015**, *53*, 225–234. [CrossRef]
- 25. Mohammadil, E.; Montazeri-Gh, M. A new approach to the gray-box identification of Wiener models with the application of gas turbine engine modeling. *J. Eng. Gas Turbines Power* **2015**, *137*, 071202. [CrossRef]
- 26. Luo, M.; Sun, F.; Liu, H. Joint block structure sparse representation for multi-input-multi-output (MIMO) T-S fuzzy system identification. *IEEE Trans. Fuzzy Syst.* **2014**, *22*, 1387–1400. [CrossRef]
- 27. Narendra, K.S.; Gallman, P.G. An iterative method for the identification of nonlinear systems using a Hammerstein model. *IEEE Trans. Autom. Control* **1996**, *11*, 546–550. [CrossRef]
- 28. Wills, A.; Ninness, B. Generalised Hammerstein-Wiener system estimation and a benchmark application. *Control Eng. Pract.* **2012**, *20*, 1097–1108. [CrossRef]
- 29. Paduart, J.; Lauwers, L.; Pintelon, R.; Schoukens, J. Identification of a Wiener- Hammerstein system using the polynomial nonlinear state space approach. *Control Eng. Pract.* **2012**, *20*, 1133–1139. [CrossRef]
- 30. Wakui, T.; Imaizumi, N.; Yokoyama, R. Model-based performance monitoring with dynamic compensation for heat utilization process in distributed energy system. *J. Syst. Des. Dyn.* **2012**, *6*, 597–615.
- 31. Na, J.; Herrmann, G. Online adaptive approximate optimal tracking control with simplified dual approximation structure for continuous-time unknown nonlinear systems. *IEEE/CAA J. Autom. Sin.* **2014**, *1*, 412–422.
- Huang, G.B.; Zhu, Q.Y.; Siew, C.K. Extreme learning machine: A new learning scheme of feed forward neural networks. In Proceedings of the International Joint Conference on Neural Networks, Budapest, Hungary, 25–29 July 2004.
- 33. Deng, W.Y.; Zheng, Q.H.; Zhang, K. Reduced Kernel Extreme Learning Machine. In Proceedings of the 8th International Conference on Computer Recognition Systems CORES, Milkow, Poland, 27–29 May 2013.
- 34. You, C.X.; Huang, J.Q.; Lu, F. Recursive reduced kernel based extreme learning machine for aero-engine fault pattern recognition. *Neurocomputing* **2016**, *214*, 1038–1045. [CrossRef]
- 35. Seber, G.A.F.; Lee, A.J. Linear Regression Analysis; John Wiley & Sons: Hoboken, NJ, USA, 2003.
- 36. Kulikov, G.G.; Thompson, H.A. *Dynamic Modeling of Gas Turbines: Identification, Simulation, Condition Monitoring and Optimal Control*; Springer: London, UK, 2004.
- 37. Lu, F.; Wang, Y.F.; Huang, J.Q.; Huang, Y.H. Gas Turbine Transient Performance Tracking Using Data Fusion Based on an Adaptive Particle Filter. *Energies* **2015**, *8*, 13911–13927. [CrossRef]
- Gringard, M.; Kroll, A. On the parametrization of APRBS and multisine test signals for the identification of nonlinear dynamic TS-models. In Proceedings of the IEEE Symposium Series on Computational Intelligence, Athens, Greece, 6–9 December 2016.
- 39. Deflorian, M.; Zaglauer, S. Design of experiments for nonlinear dynamic system identification. In Proceedings of the 18th IFAC World Congress, Milano, Italy, 28 August–2 September 2011.



© 2017 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 (http://creativecommons.org/licenses/by/4.0/).