Ensemble Learning Based Surrogate Modeling for Gas Turbine Blisk Temperature Predictions

Thambirajah Ravichandran[1], Glenn Heppler[2], and Avisekh Banerjee[3]

[1,2]*Department of Systems Design Engineering, University of Waterloo, Waterloo, Ontario, N2L 3G1, Canada*
travicha@uwaterloo.ca
heppler@uwaterloo.ca
[3]*Life Prediction Technologies Inc., 23-1010 Polytek Street, Ottawa, ON, K1J 9J1, Canada*
banerjeea@lifepredictiontech.com

ABSTRACT

Temperature prediction in complex systems like gas turbines provides insights to temperature dependent damage accumulation but usually involves a huge computational cost. For simulation-based prognostics, the computational cost is a major hindrance to a real time implementation. In this work an ensemble learning based multistage surrogate modeling approach is investigated as a possible solution for reducing the computational cost. First the nodal temperature of a turbine blisk is predicted using computational fluid dynamic (CFD) simulations for a limited number of engine operating points. Next the proposed ensemble learning based surrogate modeling approach is implemented to train surrogate models for every node defining the blisk. To achieve computational efficiency, the proposed surrogate modeling framework implements in sequence, clustering techniques for data analysis, multistage polynomial regression modeling, and ensemble learning based model combination. Finally the prediction errors are quantified using the leave-one-out cross-validation method. The result suggests that the computational time could be significantly reduced using the proposed ensemble learning based multistage surrogate modeling technique. The threshold value used to tune the polynomial regression model complexity is also shown to influence the time for surrogate model training.

1. INTRODUCTION

Gas turbine engines are complex systems that operate under extremely high temperature and mechanical loads. In the turbine section, the gas exiting from the combustor hits the hot gas path components in the turbine section and results in temperature dependent damage accumulation that may result in component failures. The heat is transferred from the fluid to the solid through convection, within the solid through conduction as well as from solid back to the fluid through radiation. The heat transfer analysis for a turbine part becomes complicated and usually computational fluid dynamics techniques like finite volume methods are deployed for solving them. However the computational cost is very high, not only in terms of the hardware costs and time but also in terms of the software. The computation of flow parameters over a dense numerical grid using rotary frames has to be repeated for every distinct engine operating condition. The reduction of the CFD grid computation burden would allow a simulation-based prognostics system to expand its application range for real-time solutions. Hence development of advanced and cheap-to-evaluate surrogate models that emulate the expensive CFD solvers is required.

Over the past three decades surrogate models have been used extensively in the design, analysis and optimization of problems involving computationally expensive simulations. Various aspects of the development of surrogate models for computer experiments including data sampling schemes, metamodeling techniques, and model validation methods, have been investigated and reported in the literature (Wang & Shan, 2007; Viana et al., 2014). A variety of regression modeling techniques used as surrogates for computationally expensive simulations have been studied and reported in the literature (Wang & Shan, 2007). More widely used techniques for this purpose include polynomial response surface models, neural networks, Kriging models, radial basis functions (RBF), multivariate adaptive regression splines (MARS), and support vector regression (SVR). These different techniques resulted in many comparative studies to determine their merits by applying them to various problems (Jin et al., 2001; Chen et al., 2006; Giunta & Watson, 1998). There is no conclusion about which model is definitely superior to the others; instead the literature confirms that the surrogate model performance

depends on both the nature of the problem and the sampling method used. Because no single best surrogate model performs well for all problems, and the cost of developing multiple surrogate models is often small compared to the cost of high fidelity simulations, there has been increasing interest in using ensembles of surrogates (Goel et al, 2007; Viana & Haftka, 2008; Acar & Rais-Rohani, 2009; Viana et al., 2009) for applications requiring surrogate models to replace expensive simulations.

A large amount of recent work reported in the gas turbine literature has used surrogate models for different types of applications, such as design optimization, sensitivity analysis, model calibration, and uncertainty quantification (Keskin et al., 2008; Song et al., 2011; Schmitz et al., 2011; Lin et al., 2011; Cui and Wang, 2011; McFarland et al., 2012). In particular, a few recent studies have investigated the use of surrogate models for reducing the computational cost for gas turbine blade temperature predictions involving CFD simulations (McFarland et al., 2012; van Enkhuizen et al., 2017).

One of the main challenges often faced when using surrogate models for CFD applications is that one has to deal with a large number of nodal outputs (McFarland et al., 2012). Thus, the surrogate model developed for CFD applications must be capable of handling a highly multivariate output by representing the model response for all nodal locations. McFarland et al. (2012) presents an approach in which individual node-based surrogate models are developed and demonstrated for 106 nodal locations. The above approach is significantly extended in this paper to handle more than 100 thousand nodal locations on the blisk by addressing the following research issues: a) node-based surrogate modeling with limited data, b) multistage processing adopted for improving computational efficiency, and c) ensemble learning to augment accuracy/robustness of node-based surrogate models constructed using limited data.

In this work, the feasibility of using an ensemble learning based surrogate modeling method for predicting the temperature distribution over gas turbine components is studied. In the first step, CFD simulation of a blisk is performed for a limited number of operating conditions. In the next step the proposed ensemble learning based surrogate modeling approach is implemented to train surrogate models for every CFD node. To achieve computational efficiency, the proposed surrogate modeling framework implements in sequence, clustering techniques for data analysis, multistage polynomial regression modeling (PRM), and ensemble learning based model combination. Finally the prediction errors are quantified using the leave-one-out cross-validation method.

The prediction of the temperature of gas turbine components is critical to prediction of stress-strain states and damage accumulation under different operating conditions. The long-term goal of this work is to develop a cheap-to-evaluate and near real-time processing of the gas turbine usage data and to estimate the remaining useful life and predictive maintenance of critical components.

2. APPLICATION FOR REAL-TIME PROGNOSTICS

Simulation-based prognostics rely on the numerical modeling and simulation of different aspects of the operation and behaviour of the engine components. This will enable the accurate damage accumulation and prognosis as a function of the actual usage. This is particularly useful in gas turbines where sensors are not available to detect performance or health parameters. However only limited engine operating data is available that can be used to define an operating envelope. As the engine operates at different operating points, the engine performance and the hot gas temperature keep on varying. Engine modeling using thermodynamic principles can be utilized to generate boundary conditions for CFD based heat transfer analysis to obtain the temperature distribution over the components. The component temperatures can then be converted into stress-strain states using the mechanical loading which then allows for the determination of the damage accumulation levels under different modes.

In simulation-based prognostics, the heat transfer analysis is performed by a high fidelity and expensive CFD solver. Computation of the flow parameters is required over a dense numerical grid using rotary frames for every distinct engine operating condition. As such there can be significant wait times for the simulation-based prognostics.

The objective for real-time prognostics is to replace this high fidelity and expensive solver with advanced and cheap-to-evaluate interpolation models that emulate the CFD solver. With such an interpolation method for CFD results, the models can be trained with a minimal number of CFD simulation data generated up front by using expensive solvers. The simulation results are very rich in information so advanced learning techniques have to be also investigated with a view to minimizing the required number of simulations. The input includes the parameters for the boundary conditions of the CFD analysis estimated from the on-design engine modeling. Once trained, the models can now be used to directly predict the component temperatures for any engine operating point for which no simulation result exists.

A framework as shown in Figure 1 is proposed for applying the machine learning based interpolation utilizing limited pre-run CFD simulation results. Based on the engine operating conditions, the boundary conditions (e.g. gas temperature, pressure, mass flow) used as input to the CFD analysis keep on varying. This necessitates the rerunning of the CFD simulation for every operating point. If a limited set of CFD simulation results is available, then through machine learning techniques robust interpolation models can be defined based on every node in the CFD grid. This

model can then be used for predicting the nodal CFD results like temperature for new inputs. The prediction model also runs in real-time, leading to implementation of real-time prognostics within an envelope of existing operating points and their CFD simulation results.

3. MACHINE LEARNING APPROACH FOR SURROGATE MODELING

This section briefly outlines various machine learning techniques that can be utilized in developing the proposed multistage surrogate-modeling framework for generating node-based surrogate models.

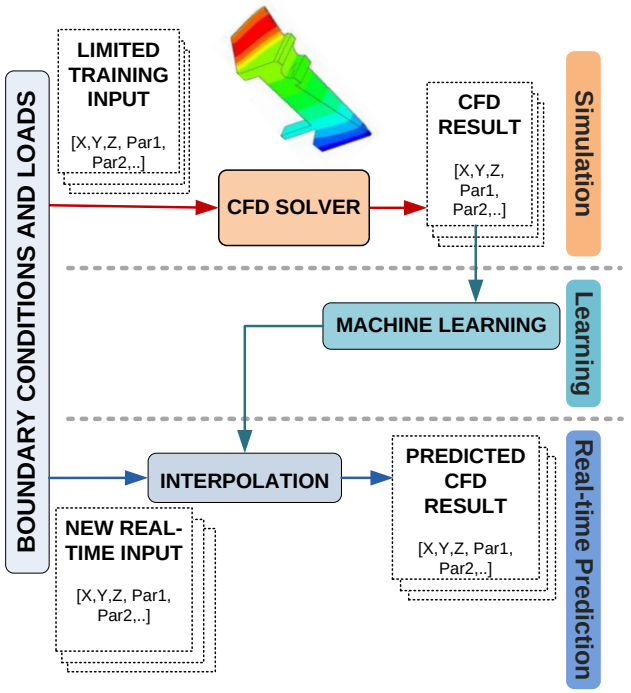

Figure 1. Application of machine learning and surrogate modeling to real-time prognostics of blisk temperature.

3.1. Regression Approaches for Surrogate Modeling

Polynomial regression models (PRM) also known as response surface models (RSM) have been widely applied for surrogate modeling using low-order polynomials. The constant parameters of the PRM are determined using a linear least squares algorithm. For problems with a large number of inputs and limited training data, polynomial regression models are limited to using linear or second-order polynomial models. In the literature, a class of kernel based surrogate models was studied for surrogate modeling of problems involving expensive computer simulations with considerable success (Wang & Shan, 2007). The members of this class of surrogate models include, Gaussian process regression (GPR) models, radial basis function (RBF) models, and support vector regression (SVR) models. Among these techniques, GPR models are the most widely used for metamodeling of computer simulations (Viana et al., 2014). For a given input vector x, the predicted output y of the GPR model is given as a combination of a known polynomial function f(x) representing the global trend of the output and a gap function g(x) representing deviations as a realization of a stochastic process with zero mean, constant variance, and nonzero spatial covariance function. GPR is a flexible technique because different variations can be created by choosing different pairs of f(x) and the correlation functions. For metamodeling, RBF models achieve approximation by using a linear combination of radially symmetric functions with weight coefficients. Gaussian basis functions are mostly used in RBF models where the basis functions are expressed in terms of the Euclidean distance between the basis function center and new input vector (Jin et al., 2001). In SVR, the goal is to find a function that has at most ε deviation from the training data (Clarke et al., 2005). In other words, the errors are considered zero as long as they are less than ε. Besides ε, the fitting of the SVR model has a regularization parameter which determines the compromise between the model complexity and the degree to which deviations larger than ε are tolerated in the model formulation. An open issue in SVR is the choice of parameter values for both the kernel and loss functions.

As noted before all the above regression models (PRM, GPR, RBF and SVR) were studied previously for surrogate modeling of various problems. However, there was no definite consensus reached about any single model type being superior to the others for all problems. This prompted the investigation of ensemble of surrogate models involving either homogeneous or heterogeneous model types as described in the following section. It should be noted here that in this work only homogeneous models involving PRMs are considered for developing ensembles of surrogate models to predict CFD temperatures. Future work will study the effectiveness of considering all the above regression models in an ensemble framework for surrogate modeling considering heterogeneous model types.

3.2. Ensemble Learning

Ensemble learning is an aggregation of multiple models using some combination methods to form a final prediction model. Unlike ordinary learning approaches, which try to construct a single model from training data, ensemble learning methods try to construct multiple models to solve the same problem. Ensemble learning generally provides solutions with improved accuracy and/or robustness in most applications due to the availability of accurate and diverse multiple models for combining them into a single solution. Well known ensemble learning algorithms include stacking (Wolpert, 1992; Breiman, 1996a), bagging (Breiman, 1996b), and boosting (Freund & Schapire, 1996) algorithms.

Generally, ensemble learning is implemented in three phases (see Figure 2): 1) generation of base models, 2) selection of base models, and 3) aggregation of the selected base models using some combination methods. In the first phase, a pool of base models is generated, and the pool may consist of homogeneous base models (same model types) or heterogeneous base models (mixture of different model types). In the second phase, a subset of base models is selected. Finally, a model is formed by aggregating the selected models using a combination method. To get a final model with improved generalization, it is essential that the base models should be as accurate as possible, and as diverse as possible.

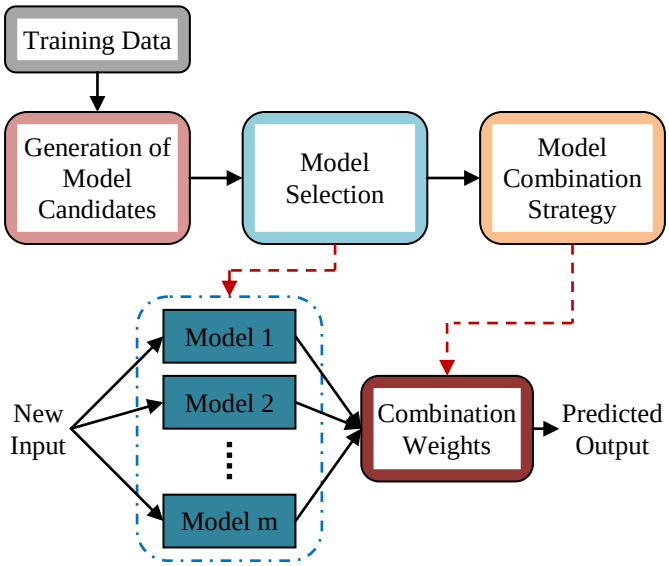

Figure 2. Ensemble learning process and architecture.

It should be noted here that generally the computational cost of constructing an ensemble of models is not much larger than creating a single model. This is because typically we need to generate multiple versions of the model for model selection when we want to construct a single model, and this is comparable to generating base models in ensemble learning, while the computational cost for combining base models is often small.

4. SURROGATE MODELING FRAMEWORK

This section outlines the methodology for the CFD surrogate model development by providing details for, the input data analysis, the multistage polynomial regression modeling which incrementally tunes the model complexity for each node, and the ensemble learning to utilize multiple models. As part of this methodology, a cluster-based model structure selection strategy and a node-based local model parameter estimation approach are introduced for an efficient CFD surrogate model development process.

4.1. Multistage Surrogate Modeling Framework

Inputs to the proposed framework include CFD data describing nodal spatial and temperature distributions, and input data describing boundary conditions and loads characterizing various operating parameters. Then the proposed methodology sequentially implements, i) a data clustering analysis along with a region-based input selection method (stage 1), ii) a multistage PRM based surrogate modeling which includes model structure selection, model parameter estimation and cross-validation procedures (stage 2), and finally, iii) an ensemble learning for model combination (stage 3) as shown in Figure 3.

More details of the functional descriptions for each block in all three stages of the surrogate modeling framework as shown in Figure 3 are given in the following subsections.

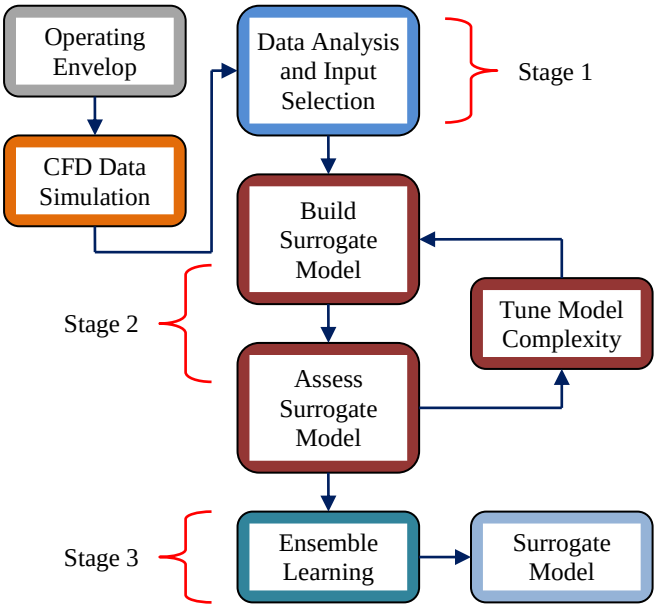

Figure 3. PRM based multistage surrogate modeling framework.

4.2. CFD Data Analysis and Input Selection

To solve the CFD interpolation problem at the node level, given the CFD data generated for various operating points, first one needs to identify suitable inputs with high predictive power by using problem specific knowledge and dividing the CFD data into regions.

Within a chosen region with suitable inputs identified (applying problem specific knowledge) the *CFD Data Clustering* block (see Figure 4) groups the CFD nodes into clusters by employing a K-means clustering method in phase 1 using temperature values and then a hierarchical clustering method in phase 2 to further cluster the data using nodal spatial locations. In phase 1, the number of clusters K is determined by the available temperature range (*max_T* –

min_T) and a preselected value (*delta_T*) which splits the temperature range into K clusters. In phase 2, from the above K clusters the hierarchical clustering automatically further selects the number of sub clusters based on the mean value of inter nodal distances. After forming the clusters, the *CFD Data Clustering* block outputs the cluster representative nodes that are used as inputs to the model structure search process as shown in Figure 4.

4.3. PRM based Surrogate Modeling

Having identified the suitable inputs within each region and performed a data clustering to group neighbouring nodes into clusters, the proposed PRM based surrogate modeling implements a cluster-based model structure selection by constraining all the nodes within a cluster to adopt the same model structure. Then, having selected the suitable inputs and the most appropriate model structure at each node, one can proceed to estimate the model parameters using the least squares method.

The two major steps involved in a model structure search and selection are: i) generation of candidate model structures, and ii) evaluation of model structures using pre-defined selection criteria that are mostly determined by the estimated model parameters. Therefore, it is clear that the cluster-based model structure selection and the node-based local model parameter estimation are closely coupled in a loop as shown in Figure 4 which shows the process flow in a block diagram format for the cluster-based model structure search and selection procedure.

4.3.1. Cluster-based Model Structure Selection

For each cluster representative node, the *Generation of Model Structures* block (see Figure 4) will produce a set of linear regression model structures given the set of *m* basic or natural inputs along with the model order *q* (allowed maximum degree for each regression term) and the number of regression variables *p*. The model structure of a linear regression model is defined by the number and the type of regression variables used in the model formulation. The regression variables used in the model are selected from a larger pool of regression variables that are formed by transforming a set of basic or natural input variables. The process of generating a larger set of regression variables from a smaller set of basic or natural input variables, and then choosing all possible combinations of *p* number of regression terms from the larger set of regression variables is performed by the *Generation of Model Structures* block in Figure 4. The procedure to implement these two steps is given below.

1. Consider a set of *m* basic or natural input variables denoted as $\{w_1, w_2, ..., w_m\}$ and from this set, let us generate a larger set of M regression variables denoted as $\{X_1, X_2, ..., X_M\}$ by using some input transformations (depending on the given model order *q*) such as linear, quadratic, cubic, cross-product terms of the original *m* basic and natural input variables.

$$\{w_1, w_2, ..., w_m\} \rightarrow \{X_1, X_2, ..., X_M\}$$

2. From the larger set of regression variables $\{X_1, X_2, ..., X_M\}$, choose all possible combinations of *p* number of regression terms and this will form a set of candidate linear regression model structures.

$$\{X_1, X_2, ..., X_M\} \rightarrow \{x_1, x_2, ..., x_p\}$$

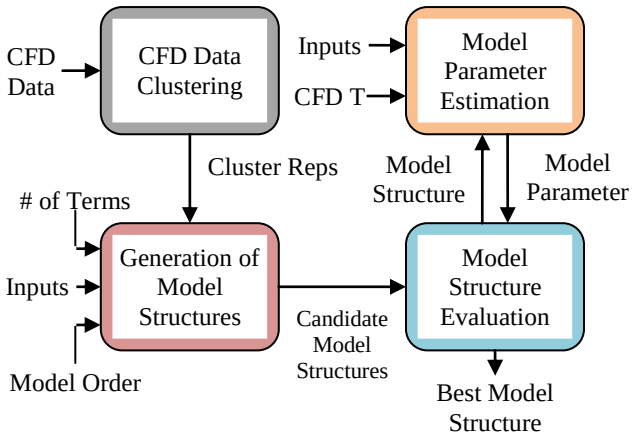

Figure 4. Process flow for cluster-based model structure search and selection.

4.3.2. Node-based Local Model Parameter Estimation

The node-based local model parameter estimation forms the core of the solution proposed for the CFD surrogate model development studied in this paper. At each node, given the model structure information along with the training datasets, the goal of the model parameter estimation procedure is to seek the values for the unknown parameters of the linear regression model by using the least squares method.

4.3.3. Evaluation of Candidate Model Structures

To select the best model structure from the set of candidate model structures generated by the *Generation of Model Structures* block as described above, one needs to specify a set of performance measures or evaluation criteria to assess the performance of each model structure generated. In the following, a few such performance measures are described.

Measure of Fit or Goodness of Fit

Measure of fit or goodness of fit for a linear regression model can be expressed as a function of the residual sum of squares R^2 also known as the **coefficient of determination**. Values closer to 1 for R^2 indicate a good measure of fit and high predictive power of the regression variables for the target variable, while values close to 0 indicate little predictive power. An equivalent representation of *R* is given

by the sample correlation coefficient (also known as the **multiple correlation coefficient**) between the observed and fitted target vector values which can be implemented using the Matlab function *corrcoef* as follows

$$R = corrcoef(\boldsymbol{y}, \hat{\boldsymbol{y}}) \quad (1)$$

R is a direct measure of how similar the observed ($\boldsymbol{y}$) and fitted target ($\hat{\boldsymbol{y}}$) vector values are.

Hat Matrix

For the linear regression model expressed in a matrix form as $\boldsymbol{y} = \boldsymbol{Xb} + \boldsymbol{\varepsilon}$, the hat matrix **H** is given as

$$\boldsymbol{H} = \boldsymbol{X}(\boldsymbol{X^T X})^{-1}\boldsymbol{X^T} \quad (2)$$

The diagonal elements h_{ii} of this hat matrix are important for evaluating the test performance of the model while predicting with new data. These diagonal elements also give a measure of extrapolation attempted while predicting with new data. In this study, the maximum value of the diagonal elements of the hat matrix is used as one of the important criteria when selecting the best model structure.

Variance Inflation Factor (VIF)

When some of the regression variables in a linear regression model are highly correlated with each other, this situation is called as collinearity. Regression variables that are highly correlated with each other can lead to instability in the regression parameters. A diagnostic procedure to determine this collinearity in general is the **variable inflation factor** (*VIF*) that quantifies the proportional increase in the variance of the estimated parameter for each regression variable compared to what it would have been if the regression variables had been uncorrelated. The VIF is defined as

$$VIF_i = \frac{1}{(1-R_i^2)} \quad (3)$$

where R_i^2 is the R^2 for the regression fit of the variable x_i on other regression variables. In this paper, the maximum value of VIF_i is used as one of the important criteria while selecting the best model structure. To avoid a collinearity problem with the selected model, the maximum value of VIF_i is kept within a low range. There are no formal guidelines to specify the cut-off for the maximum value of VIF_i. Collinearity is generally not a problem if the maximum value of VIF_i is below certain threshold given by the following equation

$$\max(VIF_i) = \max\left(10, \frac{1}{(1-R_m^2)}\right) \quad (4)$$

where R_m^2 is the usual R^2 for the chosen linear regression model (Chatterjee & Simonoff, 2013). The above cut-off implies that either the regression variables are more related to the target variable than they are to each other, or they are not related to each other very much. In either scenario, collinearity will not be a problem for regression modeling.

PRESS Criterion

If we delete the i^{th} data sample, fit the regression model to the remaining n-1 data samples, and calculate the predicted value of y_i corresponding to the deleted data sample, the corresponding **prediction error** is

$$e_{(i)} = y_i - \hat{y}_{(i)} \quad (5)$$

This prediction error calculation is repeated for each data sample i = 1, ..., n. These prediction errors are usually called **PRESS errors** because of their use in calculating the **PRESS** (Prediction Error Sum of Squares) **criterion** as shown below. It would initially seem that calculating the PRESS errors requires fitting n different regression models. However, it is possible to calculate PRESS errors from the results of a single least squares fit to all n data samples as shown here:

$$e_{(i)} = \frac{e_i}{1-h_{ii}}, \quad i = 1, \ldots, n \quad (6)$$

where it is easy to see that the PRESS error is just the ordinary error weighted according to the diagonal elements h_{ii} of the hat matrix.

It has been suggested (Allen, 1974) that using the Prediction Error Sum of Squares (PRESS) criterion, defined as the sum of the squared PRESS errors, serves as a good measure of model quality. The PRESS criterion is defined as

$$PRESS = \sum_{i=1}^{n}[y_i - \hat{y}_{(i)}]^2 = \sum_{i=1}^{n}\left(\frac{e_i}{1-h_{ii}}\right)^2 \quad (7)$$

The PRESS criterion is generally regarded as a measure of how well a regression model will perform in predicting new data. A model with a small value of the PRESS criterion is desired.

The maximum VIF value and the maximum value of the hat matrix diagonal elements play the role of filters in reducing the number of model structures considered for selection thus improving the computational efficiency of the model structure selection process. Furthermore, as regression variables that are highly correlated with each other can lead to instability in the regression parameters, the maximum VIF value is used first to filter out model structures with high collinearity thus assuring stability in the subsequent computations. Also, as the computation of the hat matrix is slightly more involved than the computation of VIF values for a given model structure, it is advantageous to limit the model structure space using the maximum value of the hat matrix diagonal elements after reducing the model structure space with the maximum VIF value. After ensuring computational stability and efficiency, the PRESS criterion is used at the end to provide performance ranking on the reduced model structure space. By combining the maximum

VIF value, the maximum hat matrix diagonal value, and the PRESS criterion as defined above, the procedure to implement the evaluation of the model structures is:

1. From the set of candidate model structures generated by the Candidate Model Structures Generation block, select a subset of model structures that satisfy the requirement for a *VIF* value below a certain threshold value VIF_{max}.

2. From the subset of model structures selected in the above step, further select a reduced subset of model structures that satisfy the maximum value of the hat matrix diagonal elements h_{ii} below a certain threshold value h_{max}.

3. From the reduced subset of model structures selected in Step 2 above, choose the best model structure that has the lowest PRESS criterion value. Sometimes, it may be desired to choose a few top performing model structures instead of a single best model structure.

4.4. Cross-Validation Techniques

The simplest and most widely used method for estimating prediction error is cross-validation. This method directly estimates the generalization when the model is applied on a new data set. Typically, there are two main separate goals for which we need the cross-validation approach, namely, model selection and model assessment. If we are in a data-rich situation, the best approach to meet these goals is to randomly divide the dataset into three parts: a training set, a validation set, and a test set. Since data are often scarce, as is the case we are faced with in this study, the above division of data into three parts is not possible. To tackle this problem of data scarcity, K-fold cross-validation uses part of the available data to fit the model, and a different part to test it. We split the data into K equal-sized parts and fit the model using K-1 parts, and then calculate the prediction error of the fitted model using the remaining part. The case K = n is known as the leave-one-out cross-validation. This leave-one-out cross-validation is efficiently implemented using the PRESS criterion as discussed above.

4.5. Model Complexity Tuning Process

For the CFD surrogate modeling studied in this paper, polynomial regression modeling techniques are used and the *Tune Model Complexity* block in Figure 3 is responsible for incrementally changing the model complexity from linear terms to quadratic terms and then finally to cubic terms. That is, at each node, a model structure search is performed sequentially by searching over the model candidates. In the first stage, using RM1 (linear) regression modeling, the model structure search is performed using up to linear terms. In the next stage, for those nodes showing a maximum absolute error of more than a certain threshold (say TH), RM2 (quadratic) regression modeling is employed and the model structure search is performed using up to quadratic terms. In the final stage, for those nodes from the second stage with a maximum absolute error more than a certain threshold (say TH again), RM3 (cubic) regression modeling is employed and the model structure search is performed using up to cubic terms.

4.6. Ensemble Learning using Stacked Regression

The surrogate-modeling framework in Figure 3 uses the *stacked regression* method (Breiman, 1996a) for ensemble learning. Stacked regression is a general method for forming linear combinations of base level models (also known as level-1 models) to give improved prediction accuracy. Here, the combiner is called a level-2 model. The idea is to train the level-1 models using the original training data, and then use cross-validation data and least squares methods under non-negativity constraints to determine the coefficients for the level-2 model. The non-negativity constraint is needed to guarantee that the performance of the stacked ensemble will be better than selecting the single best model (Breiman, 1996a). For the ensemble learning process adopted here, as given in Figure 2, the generation of model candidates and the model selection are implemented using the process shown in Figure 4. For the model selection in Figure 2, the top performing M_t models, in terms of prediction errors, are identified first then this set is further reduced to m models by removing those models with a high correlation with the single best model in terms of cross-validation data. For the model combination strategy in Figure 2 where level-1 models are combined using a *weighted average* (WA) approach, a non-negative least squares method (Lawson & Hanson, 1974) is used with an added constraint that the sum of all the weights is equal to one. In other words, the weights or coefficients for the level-2 model are determined using the least squares method under non-negativity constraints with the outputs of level-1 models (computed using the cross-validation step) as input data to predict given output data. This process can be implemented using the Matlab function *lsqnonneg*. It is also noted here that the *simple average* (SA) strategy for combining m level-1 models can be considered as a special case of the above weighted average (WA) strategy where each weight takes the (equal) value of $1/m$.

5. MODEL TRAINING AND VALIDATION

In order to test the efficacy of the proposed multistage CFD surrogate modeling framework in general and the model structure selection and model parameter estimation procedures in particular, 13 CFD datasets of temperatures for a gas turbine blisk were considered at various operating-points. The following set of inputs is selected for building CFD interpolation models in the blade region from the boundary conditions and load inputs, and their values are shown in Figure 5.

- Load - Shaft speed (rpm)
- BC1 - Inlet average total temperature (K)

- BC2 - Inlet maximum total temperature (K)
- BC3 - Inlet flow rate (kg/s)
- BC4 - Outlet pressure (kPa)
- BC5 - Bore wall fixed temperature (K)

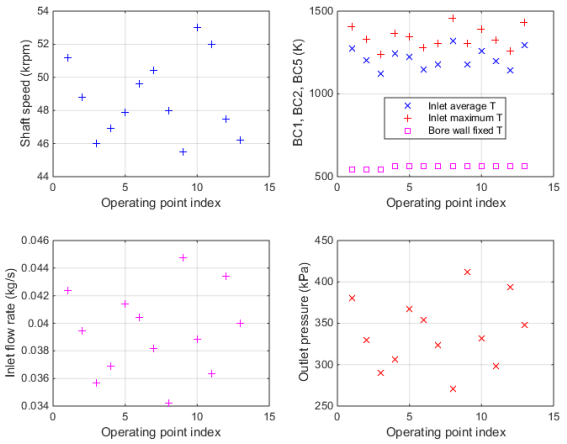

Figure 5. Boundary conditions and load values considered as inputs for surrogate modeling.

Figure 6 shows the temperature distribution over the blade section of the blisk as a function of different operating conditions and plotted against the Y position of the CFD node location. Based on a preselected value of *delta_T* = 5, the blade region nodes are grouped into K = 52 clusters of nodes during phase 1. From these 52 clusters, the hierarchical clustering further selects 2918 sub clusters during phase 2 based on the mean value of inter nodal spatial (Euclidean) distances.

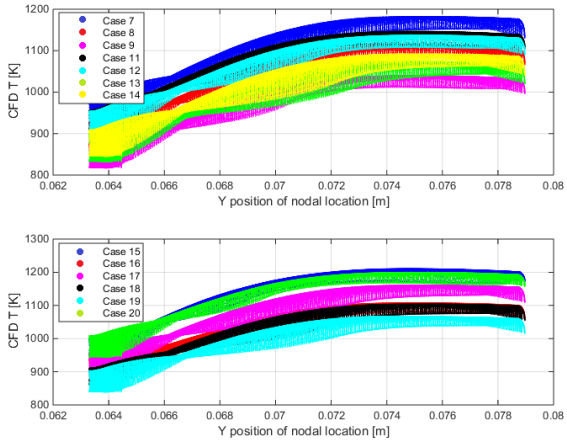

Figure 6. Blade region solid CFD temperature data simulated at 13 operating points (Cases 7-9: operating points 1-3 and Cases 11-20: operating points 4-13).

In the first stage, for the blade region, RM1 regression modeling is used along with five inputs (BC1 – BC4 and load) shown above in Figure 5. In the next stage, for those nodes showing a maximum absolute error of more than a threshold value of TH=4K, RM2 regression modeling is employed along with the same five inputs as above. In the final stage, for those nodes from the second stage with a maximum absolute error of more than a threshold value of TH=4K, RM3 regression modeling is employed along with the same five inputs as above. It should be noted here that multiple models can be chosen at each stage of the above multistage PRM training as they are readily available for selection without additional computational cost. In turn these multiple models can be used as an ensemble of PRMs at each node for temperature predictions. However, in this work multiple models are chosen only at stage 3 (RM3) at selected nodes with prediction errors higher than TH = 4 to form an ensemble of PRMs at each of those nodes.

Figure 7 shows the prediction error performance at each node for the blade region obtained using the multistage PRM (by applying RM1, RM2 and RM3 in stages) for 13 CFD data cases involving 99,577 nodes. This prediction error performance is obtained using the leave-one-out cross-validation test method - trained on 12 data cases and tested on the remaining single case and repeated for all 13 combinations – and resulting in 13 prediction errors at each node. The maximum temperature and temperature gradient is in the mid-airfoil region of the blade. Hence the higher errors in this region show the difficulties in training the surrogate models for this region across different operating points. Figure 8 shows the cross-validation errors for each operating point as box plots of absolute prediction errors in CFD temperatures for the blade region nodes. In Figure 8, as per the standard definition of the box plot, the minimum and maximum limits of the vertical error bars represents 0 and 100 percentile respectively, whereas the minimum and maximum values of the box represents 25 and 75 respectively. The red horizontal line inside the box represents the 50 percentile. The red crosses represent the outliers of the respective box plots.

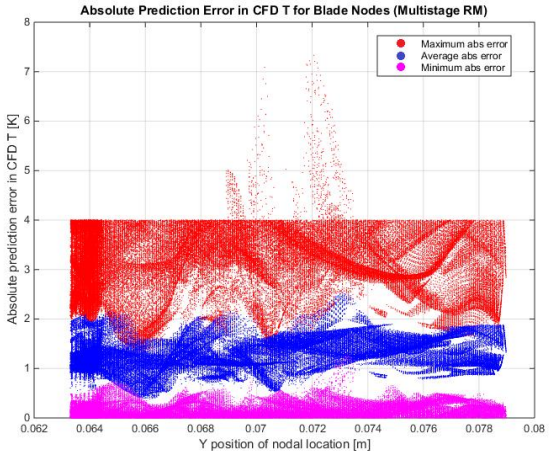

Figure 7. Absolute prediction errors in CFD temperature at each blade node.

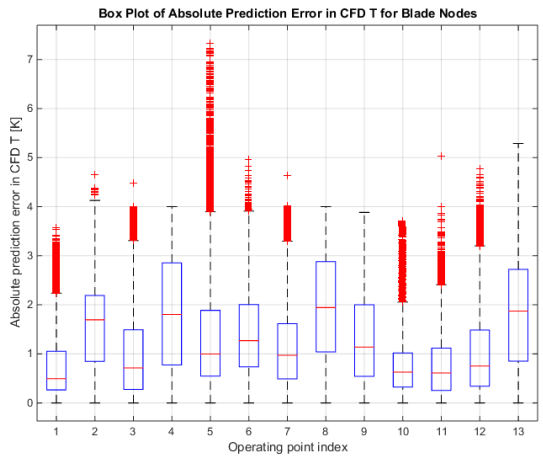

Figure 8. Box plots of absolute prediction errors in CFD temperatures for blade nodes at each operating point.

Figure 9 shows the performance comparison in terms maximum and mean absolute errors taken at each node for various threshold values by displaying the box plots of absolute errors taken over the entire blisk region. It should be noted here that in Figure 9, the box plots show the performance for 151,892 blisk nodes that include the disk region nodes also. Table 1 compares the CFD simulation time and surrogate model training and prediction time performances. The comparison of surrogate modeling training times that vary with prediction error threshold values is also shown in Table 1.

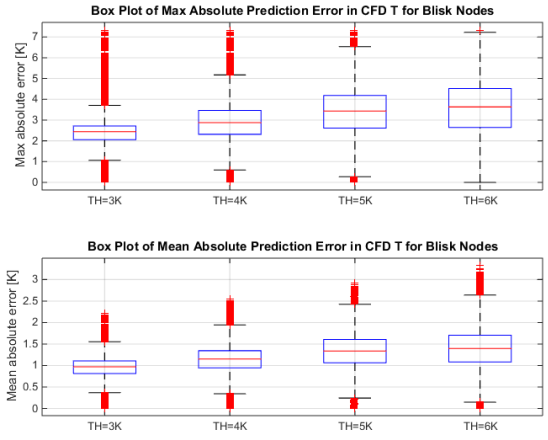

Figure 9. Box plots of maximum and average absolute prediction errors in CFD temperatures for blisk nodes using different threshold values.

Table 1. Training and Prediction Time.

CFD Mesh Details:	
Number of Nodes	151,892
Number of Elements	749,310
CFD Computation Time:	
2 cores, 64 GB RAM	241 min
4 cores, 64 GB RAM	131min
Training Time (4 cores, 16 GB RAM):	
Threshold: 3K	141 min
Threshold: 4K	60.1 min
Threshold: 5K	19.1 min
Threshold: 6K	7.6 min
Prediction Time (4 cores, 16GB RAM):	
All Nodes Combined	0.1 sec

For those nodes in Figure 7 with a maximum absolute error of more than the threshold value TH=4K were selected for the ensemble learning process. The ensemble learning was employed using the stacked regression method (weighted average) and the simple average strategy. The performance comparison between the single best model and the ensembles obtained using the weighted average (WA) and simple average (SA) are shown in Figure 10. This figure shows the box plots of maximum absolute error and RMSE in CFD temperatures for those selected blade nodes. As for the surrogate model training time with ensemble learning is increased only slightly by less than 2% compare to the times reported in Table 1.

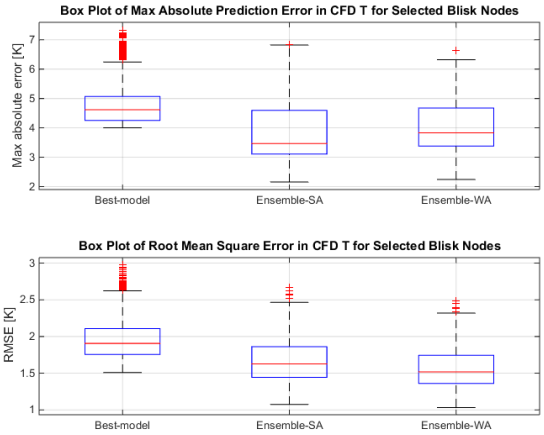

Figure 10. Box plots of maximum absolute error and RMSE in CFD temperatures for selected blade nodes using single best model and simple & weighted average ensembles.

6. RESULTS AND DISCUSSION

To demonstrate the efficacy of the proposed multistage surrogate modeling framework, a minimal number of operating points were selected using Latin Hypercube Sampling techniques (Santner et al., 2003) to maximize the space filling property of the operating points envelop. Corresponding to these minimal sets of 13 operating points, CFD datasets of temperatures were generated as shown in Figure 5. Utilizing problem knowledge, different sets of inputs (consisting of BCs and the load) were selected for developing surrogate models for the blade and disk regions. As demonstrated by the results here, selected inputs exhibit high predictive power for interpolating blisk temperatures.

This study demonstrates the effectiveness of the proposed multistage approach for developing CFD node-based surrogate models involving a very high number of CFD nodes (151,892 nodes). Figure 7 illustrates the very good performance for the PRM based surrogate modeling in terms of leave-one-out cross-validation errors for all the blade nodes (R^2 values range from 0.9983 to 0.9999). Only a small fraction (less than 1%) of these nodes exceeded the threshold value of 4K in terms maximum absolute error. As it can be seen from Figure 8, the only major contribution for these high maximum errors comes from operating point 5. One of the advantages of this node-based approach is that the goodness of fit of the surrogate models can be analyzed with respect to node location, making it possible to identify specific regions on the blisk where the goodness of fit of the surrogate models are not high enough, for example, the region between 0.069m and 0.073m in Figure 7.

Figure 9 and Table 1 illustrate the performance trade-off in terms of prediction accuracy and computational time for the entire blisk region when the threshold parameter TH is varied between 3K and 6K. When the threshold value is decreased, the number of nodes utilizing high order polynomials increases yielding reduced prediction errors and increased computation time as reflected in Figure 9 and Table 1. Less time is required to develop the surrogate models using the proposed multistage approach compared to full CFD simulations and real-time prediction is also achieved as shown in Table 1.

On selected blade nodes with high prediction errors, Figure 10 illustrates the performance enhancement of 20%-25% that can be achieved by using the ensemble learning approach as part of the proposed multistage surrogate modeling framework. The performance difference between the simple average (SA) strategy and the stacked regression method (weighted average) is not significant. In general, it is widely accepted that the SA strategy is suitable for combining models with similar performance, whereas the WA strategy with unequal weights may be more appropriate for combining models with diverse performances.

7. CONCLUSION

An ensemble learning based multistage approach has been investigated for developing surrogate models to predict gas turbine blisk temperatures. Using CFD simulations for a limited number of engine operating points, node-based surrogate models were developed using a multistage polynomial regression modeling approach. By adopting clustering techniques for data analysis, multistage polynomial regression modeling, cross-validation techniques, and ensemble learning based model combination, the proposed multistage surrogate modeling framework demonstrated its effectiveness in developing node-based surrogate models for more than 100 thousand nodes with high predictive performance and computational efficiency. Also, the computational time required for model training is influenced by the threshold limit used to tune the model complexity. Future work will consider testing the effectiveness of this methodology using different problem datasets.

ACKNOWLEDGEMENT

The authors would like to acknowledge Prameela Gunturu and Veda Erukulla at Life Prediction Technologies for their valuable guidance on CFD simulations. The authors would also like to acknowledge the financial support provided for this work by the Natural Sciences and Engineering Research Council (NSERC) of Canada and the Ontario Centres of Excellence (OCE).

REFERENCES

Acar, E., & Rais-Rohani, M. (2009). Ensemble of Metamodels with Optimized Weight Factors. *Struct and Multidisciplinary Opt*, 37(3), pp. 279-294.

Allen, D. M. (1974). The relationship between variable selection and data augmentation and a method for prediction. *Technometrics*, vol. 16, pp. 125-127.

Breiman, L. (1996a). Stacked regressions. *Machine Learning*, vol. 24(1), pp. 49-64.

Breiman, L. (1996b). Bagging predictors. *Machine Learning*, vol. 24(2), pp. 123-140.

Chatterjee, S., & Simonoff, J. S. (2013). *Handbook of Regression Analysis*. New Jersey: John Wiley & Sons.

Chen, V. C. P., Tsui, K. L., Barton, R. R., & Meckesheimer, M. (2006). A Review on Design, Modeling and Applications of Computer Experiments. *IIE Trans.*, vol. 38, pp. 273–291.

Clarke, S. M., Griebsch, J. H., & Simpson, T. W. (2005). Analysis of Support Vector Regression for Approximation of Complex Engineering Analyses. J. Mech. Des., 127 (6), pp. 1077–1087.

Cui, W., & Wang, J. (2011). Probabilistic Analysis of Gas Turbine Disk Multi-Crack Propagation. ASME 2011 Turbo Expo: Turbine Technical Conference and Exposition, pp. 715-721.

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. *Proceeding of 13th International Conference on Machine Learning*, Bari, Italy, July 3-6, pp. 148-156.

Giunta, A. A., & Watson, L. T. (1998). A Comparison of Approximation Modeling Techniques: Polynomial versus Interpolating Models. *Proc. of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization*, vol. 1, AIAA, St. Louis, MO, September 2–4, AIAA-98-4758.

Goel, T., Haftka, R. T., Shyy, W., & Queipo, N. V. (2007). Ensemble of Surrogates. *Structural and Multidisciplinary Optimization*, 33(3), pp. 199-216.

Jin, R., Chen, W., & Simpson, T. W. (2001). Comparative Studies of Metamodeling Techniques under Multiple Modeling Criteria. *Structural and Multidisciplinary Optimization*, vol. 23 (1), pp. 1–13.

Keskin, A., Swoboda, M., Flassig, P. M., Dutta, A. K., & Bestle, D. (2008). Accelerated Industrial Blade Design Based on Multi-Objective Optimization Using Surrogate Model Methodology. ASME Turbo Expo 2008: Power for Land, Sea, and Air, pp. 2339-2349.

Lawson, J., & Hanson, R. (1974). *Solving Least Squares Problems*. New Jersey: Prentice-Hall.

Lin, T., Mendoza, E., & Kestner, B. K. (2011). Model-Based Data Reconciliation and Bias Detection for Heavy-Duty Industrial Gas Turbines Performance Diagnosis. ASME 2011 Turbo Expo, pp. 271-280.

McFarland, J. M., Musgrove, G. O., Chang, S. Y., & Ransom, D. L. (2012), Calibration of Gas Turbine Blade Temperature Predictions using Surrogate Models. *ASME Turbo Expo 2012*, pp. 2263-2272.

Santner, T., Williams, B., & Notz, W. (2003). *The Design and Analysis of Computer Experiments*, Springer Verlag, New York.

Schmitz, A., Aulich, M., & Nicke, E. (2011). Novel Approach for Loss and Flow-Turning Prediction Using Optimized Surrogate Models in Two-Dimensional Compressor Design. *ASME 2011 Turbo Expo: Turbine Technical Conference and Exposition*, pp. 1103-1114.

Song, P., Sun, J., Wang, K., & He, Z. (2011). Development of an Optimization Design Method for Turbomachinery by Incorporating the Cooperative Coevolution Genetic Algorithm and Adaptive Approximate Model. *ASME 2011 Turbo Expo*, pp. 1139-1153.

van Enkhuizen, M., Dresbach, C., Reh, S., & Kuntzagk, S. (2017). Efficient Lifetime Prediction of High Pressure Turbine Blades in Real Life Conditions. *ASME Turbo Expo 2017: Turbomachinery Technical Conference and Exposition*, Charlotte, NC, USA, June 26-30, 2017.

Viana, F. A. C., & Haftka, R. T. (2008). Using Multiple Surrogates for Metamodeling. *7th ASMO-UK/ISSMO International Conference on Engineering Design Optimization*, International Society for Structural and Multidisciplinary Optimization, Bath, England.

Viana, F. A. C., Haftka, R. T., & Steffen, V. Jr. (2009). Multiple Surrogates: How Cross-Validation Errors Can Help Us to Obtain the Best Predictor. *Structural and Multidisciplinary Optimization*, vol. 39(4), pp. 439-457.

Viana, F. A. C., Simpson, T. W., Balabanov, V., & Toropov, V. (2014). Metamodeling in Multidisciplinary Design Optimization: How Far Have We Really Come? *AIAA Journal*, Vol. 52(4), pp. 670–690.

Wang, G. G., & Shan, S. (2007). Review of Metamodeling Techniques in Support of Engineering Design Optimization. *J. of Mech. Des.*, 129(4), pp. 370–380.

Wolpert, D. (1992), Stacked Generalization. *Neural Networks*, vol. 5, pp. 241-259.

BIOGRAPHIES

Dr. Thambirajah Ravichandran, P.Eng., is a Research Associate at the Department of Systems Design Engineering, University of Waterloo. Dr. Ravichandran obtained his PhD in ECE from the University of Waterloo in June 2005. His broad areas of interest and expertise are in applying machine learning and computational intelligence techniques for the design and development of control systems and pattern recognition systems.

Professor Glenn Heppler, P.Eng., is a graduate from the Institute for Aerospace Studies at the University of Toronto. Professor Heppler has been at the Department of Systems Design Engineering, University of Waterloo since 1986 and has served two terms as department Chair. He is a member of the American Society for Mechanical Engineering. Currently Dr. Heppler's research interests center on the dynamics and control of flexible structures, and gyroscopic systems as applied to MEMS devices.

Dr. Avisekh Banerjee, P.Eng., is the Director of Engineering at Life Prediction Technologies Inc. His broad areas of interest and expertise lies in performing physics based prognostics, ENSIP, data trending for failure prediction and analytics, design and development of PHM framework for varied applications. He liaises with the Clients and partners requiring engineering and consulting services, as well as manages technical teams within LPTi and oversees R&D collaborations.