GPPS | JOURNAL

JOURNAL OF THE GLOBAL POWER AND PROPULSION SOCIETY
journal.gpps.global

Non-parametric surrogate model method based on machine learning with application on low-pressure steam turbine exhaust system

Original article

Article history:
Submission date: 17 February 2022
Final revision date: 7 June 2022
Acceptance date: 28 June 2022
Publication date: 11 August 2022

This is the updated version of a paper originally presented at the Global Power and Propulsion Virtual Technical Conference, GPPS Xi'an21, April 11–13, 2022.

***Correspondence:**
JC: jc980@cam.ac.uk

Peer review:
Single blind

Keywords:
optimization; neural network; surrogate model method; low-pressure steam turbine exhaust system

Citation:
Cao J., Li Q., Xu L., Yang R., and Dai Y. (2022). Non-parametric surrogate model method based on machine learning with application on low-pressure steam turbine exhaust system. Journal of the Global Power and Propulsion Society. 6: 165–180. https://doi.org/10.33737/jgpps/151661

Jiajun Cao[1,*], Qingbiao Li[2], Liping Xu[1], Rui Yang[3], Yuejin Dai[3]

[1]*Whittle Lab, University of Cambridge, 1 JJ Thomson Ave, Cambridge CB3 0DY, UK*
[2]*Computer Lab, University of Cambridge, 15 JJ Thomson Ave, Cambridge CB3 0FD, UK*
[3]*Shanghai Turbine Works Co., Ltd., No.130 Huaping Road, Shanghai 200240, China*

Abstract

Current surrogate model methods that are widely used in optimization and design processes rely on manual parameterization to describe the geometry of objects. The loss of geometric information in this process limits the prediction accuracy of surrogate model. To tackle this problem, the new method directly picks important geometric features from surface meshes of fluid domain using Graph Neural Networks (GNNs) and predicts contours of fluid variables based on extracted information with Convolutional Neural Networks (CNNs). The prediction error of CNNs propagates backwards to train GNNs to select sensitive features from surface meshes. This framework reduces uncertainties introduced by manual parameterization and the loss of geometric information because the input of this new method is from the meshes used in the numerical simulations. With CNN and larger amount of extracted geometric information, this method can also predict higher dimensions distributions of flow variables rather than only several performance metrics. The nature of non-parametric representation of geometry also allows users to access designs defined by other parameterization methods to create a larger database. Additionally, thanks to the generic nature of the new method, it can be used for any other design or optimization processes governed by partial differential equations involving complicated geometries. To demonstrate this new method, a non-parametric surrogate model is built for a low-pressure steam turbine exhaust system (LPES). The new surrogate model uses 10 surfaces meshes of the LPES as input and it is used to predicts the energy flux contours at the exit of the last stage of the turbine. Altogether 582 designs have been generated, which contains two types of geometries defined by different methods. Among them, 550 cases are used for training, and 32 cases for testing. The power output of the last two stages of the turbine predicted by the surrogate model has average 0.86% difference compared with those of numerical simulations over a wide range of power ratings. The structural similarity index measure (SSIM) is used to measure the differences between the simulated and predicted contours at the exit of the last rotor, where the average SSIM of 640 contours is 0.9594 (1.0 being identical).

Introduction

Surrogate model is widely used to accelerate the process of design and optimization of turbomachinery (Kipouros et al., 2007; Samad and Kim, 2008; Samad et al., 2008; Lee and Kim, 2009; Lee et al., 2014; Schnoes et al., 2018; Persico et al., 2019). It can predict the performance of new designs by utilizing previous results from numerical

simulations or experiments rather than performing similar simulations or experiments repetitively. In the recent decades, a lot of surrogate model methods have been developed and applied in the industrial applications, for example, (1) Polynomial Response Surface Method (Myers et al., 2016; Hecker et al., 2017; Shuiguang et al., 2020); (2) Kriging Model (Sakata et al., 2003; Jung et al., 2021); (3) Radial Basis Function and Extended Radial Basis Function (Gutmann, 2001); (4)Artificial Neural Network (Mengistu and Ghaly, 2008); (5)Support Vector Machine (Lal and Datta, 2018). Figure 1 shows the process of existing surrogate model method, and one key step in this process is manual parameterization, which is to choose some important geometric parameters according to the experience of researchers to describe the geometry. In this step, if too few parameters were used, some geometric information will be lost because it is insufficient to describe geometries with high-order surfaces (e.g., airfoils and blades). On the other hand, using too many parameters or choosing irrelevant parameters will cause problems of over-fitting (Claeskens and Hjort, 2008). It is recognized that manual parameterization is the bottleneck that prevents further improvement of the performance of surrogate model method in terms of both accuracy and flexibility.

In this paper, a novel surrogate model method is presented, which establishes a mapping relationship between the surface meshes of fluid domain and two-dimensional distributions of fluid variables (in the form of contour maps) with Graph Neural Networks (GNNs), Convolution Neural Networks (CNNs) and a Conditional Variational Autoencoder (CVAE). Figure 2 shows the differences among numerical simulation used as analysis tools, the existing surrogate model method and the proposed novel surrogate model method. The new method can process the bounding surfaces of the fluid domain from the surface meshes used in the numerical simulations and extract relevant geometric features according to their significance to the result. Compared with the existing surrogate models, new method contains less uncertainties introduced by manual parameterization. This new method also allows different types of designs from different sources to be used because the geometry input to the model is the surface mesh, not user-defined parameters. In addition, the new method has the ability of predicting two-dimensional distributions of variables (in forms of contour maps) by utilizing CNNs to process the images.

The ability of predicting two-dimensional distributions of variables is achieved by the application of CNNs. With the combination of convolutional layers, it can extract information from images, and recognize the convolutional result with multi-layer perceptron (Valueva et al., 2020). In this study, CNNs are used to predict the contours based on the latent distribution.

The ability of processing surface mesh is achieved by the application of GNNs. The nature of the graphic operation of GNNs make it capable to process the non-Euclidean domain by defining the connectivity of mesh points, while CNNs are only able to process regular Euclidean data like figures. Among existing GNNs variants (Scarselli et al., 2009), GNNs are categorized into three types: Recurrent GNNs, Spatial GNNs and Spectral GNNs. In this study, the surrogate model is built based on Spectral GNNs for its strengths in extracting features from meshes with large number of vertices (Wu et al., 2020). Spectral GNNs are built on signal processing theory. The key step, convolutional operation, is done by Chebyshev polynomial approximation (Hammond et al., 2011). In this study, GNNs are used to extract geometric information by optimizing parameters in the neural networks via back propagation of loss. This is to pick relevant information based on the feedback of prediction error, which avoids loss of geometric information and over-fitting problem. Also, the input of the new surrogate model can be both unstructured mesh and structural mesh thanks to the ability of GNNs to process non-Euclidean data.

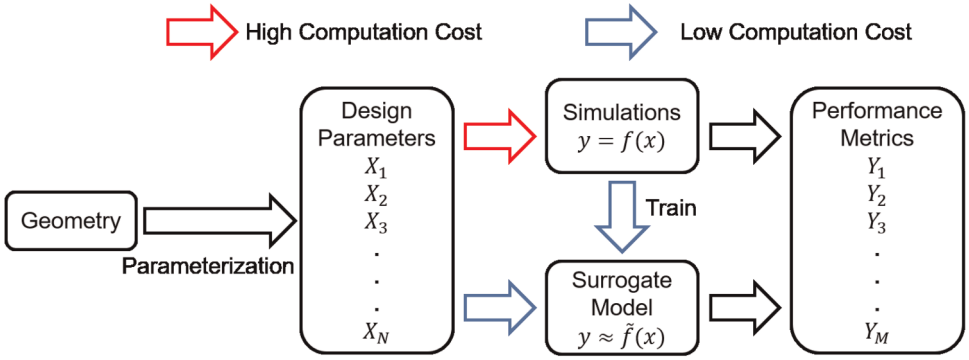


Figure 1. Process of existing surrogate model method.

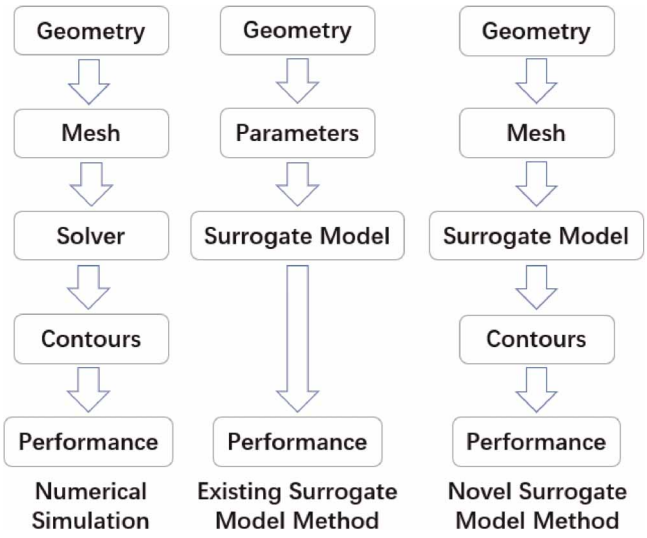


Figure 2. Comparison among numerical simulation, existing surrogate model method and novel surrogate method.

Methodology

In this section, this non-parametric surrogate model method is introduced "top-down" by its architectural designs, namely, framework, blocks and then layers. Finally, the loss function used to train this surrogate model is demonstrated.

Framework of non-parametric surrogate model

The framework of this new surrogate model method is shown in Figure 3. The mesh encoder block is a GNN encoder, which is to pick important geometric features from surface meshes. And the decoder block is a CNN decoder, which is to reconstruct the contours. Between the mesh encoder and contour decoder, there is a bottleneck that maps the extracted geometric information to latent flow variable distribution. The mapping relationship built through bottleneck needs much less training samples than directly mapping mesh vertices to contours because its input and output dimensions are much smaller. Also, it is easier to build a latent distribution for variational interface based on latent representation of geometric information in the bottleneck. The conditions in the model is blade passage index, which is to label different blade passages since the flow condition is asymmetric in the demonstration. In other applications, it can also be inlet boundary condition, material property and other important factors.

The structure of this surrogate model is basically a CVAE, which is an extension of AutoEncoder (AE). It is to compress graphical data to a latent vector and then reconstruct the graph with the latent vector. The neural network is trained to reconstruct the graphs with less error. Variational AutoEncoder (VAE) uses variational inference to estimate the latent vector rather than directly encoding from input graph (Blei et al., 2017). The latent vector z can be estimated by observation vector x using the following equation:

$$p(z|x) = \frac{p(x|z)p(z)}{p(x)} \tag{1}$$

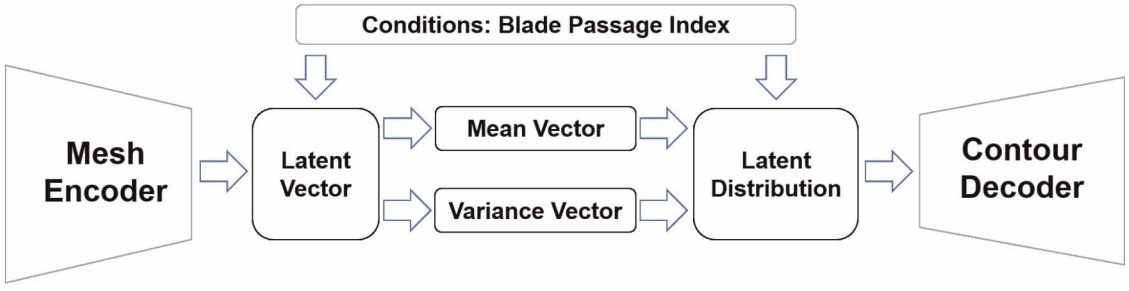


Figure 3. Framework of non-parametric surrogate model.

However, $p(z|x)$ is usually very difficult to compute directly (Kingma and Welling, 2019). Therefore, another distribution is used to approximated $p(z|x)$ in the training process. The Kullback-Leibler divergence is used to measure the difference between two probability distributions, which is to be minimized during the training process. Compared with VAE, CVAE adds conditions into the latent distribution so that different classes of input data are categorized into different groups.

Blocks in non-parametric surrogate model

The surrogate model presented in this paper contains two types of blocks: mesh encoder blocks and contour decoder blocks. The mesh encoder and contour decoder each consists of several blocks, the number of which depends on the size of mesh vertices and contour pixels. More mesh encoder blocks means a smaller latent vector, which contains less geometric information. But larger latent vector needs more training cases to prevent over-fitting.

Mesh encoder blocks

The structure of mesh encoder block is illustrated in Figure 4. One block has a Chebyshev convolution layer, a normalization layer (batch normalization), an activation layer (the rectified linear unit function), a down-sampling layer and a pooling layer (max pooling layer). The Chebyshev convolution layer is to scan the mesh vertices with Chebyshev polynomial filter and change the dimension of mesh vertices vectors. The normalization layer is to normalize the value of vectors in the same batch. And the activation layer is to zero the negative value in the vector. The down-sampling layer is to drop out irrelevant vertices based on the transformation matrix. The pooling layer is to keep the largest value and discard other values in the filter, which reduces the dimension of the vector. After several blocks, relevant geometric information is picked to form the latent vector. During a training process, the optimizer will optimize the parameters of each layer according to the loss function. In this way, the mesh encoder can keep important geometric information and discard insensitive information.

Contour decoder blocks

Contour decoder block is shown in Figure 5, which has four layers: a Chebyshev convolution layer, a normalization layer (batch normalization), an activation layer (the rectified linear unit function) and a upsampling layer. Convolution layer is to change the dimension of vector with convolutional filter. And then the vectors in the same batch are normalized by normalization layer. Activation layer is to zero the negative value in the vector. Upsample layer is to increase the number of elements in the vector. After several blocks, a latent distribution of large size is expanded to a contour of smaller size.

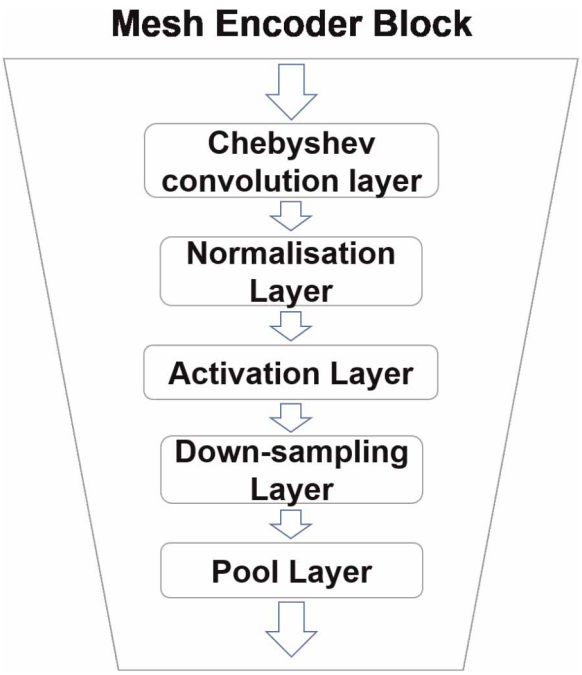


Figure 4. Mesh encoder block.

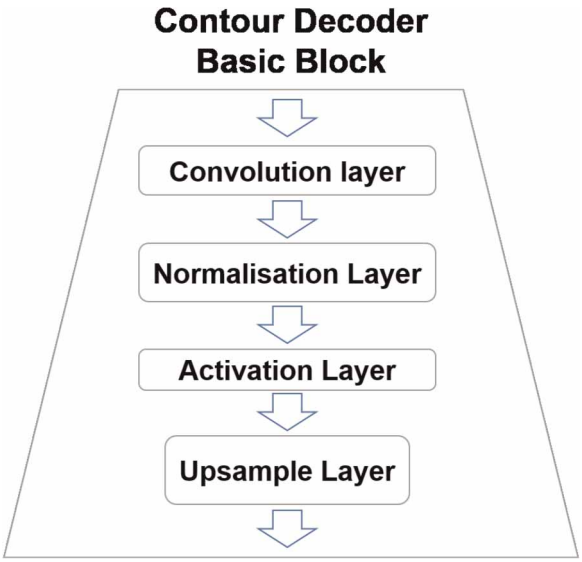


Figure 5. Contour decoder block.

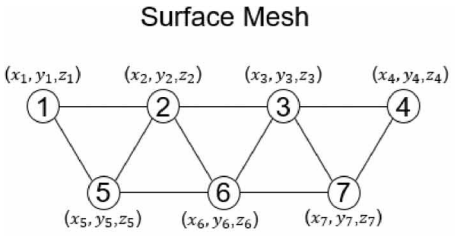
Two important layers in non-parametric surrogate model

Chebyshev convolution layer

In this paper, as shown in Figure 6, the surface mesh M of the design is defined by the coordinates of vertices V and edges E. V is the n vertices in the Euclidean space, which is an $n \times 3$ vector. The edges, E, are represented by the sparse adjacency matrix A with a size of $n \times n$, where $A_{ij} = 1$ denotes a connection between vertex i and vertex j. Otherwise,$A_{ij} = 0$. More information about representation of GNNs can be found in (Sanchez-Lengeling et al., 2021).

The most important layers used in the mesh encoder is the fast spectral convolution layer, demonstrated in (Defferrard et al., 2016). The mesh convolution operator $*$ is defined as a Hadamard product in Fourier space:

$$x * \mathrm{y} = \mathrm{U}((\mathrm{U}^{\mathrm{T}}\mathrm{x}) \odot (\mathrm{U}^{\mathrm{T}}\mathrm{y})) \tag{2}$$



Feature Matrix **F**

X1	X2	X3
x_1	y_1	z_1
x_2	y_2	z_2
x_3	y_3	z_3
x_4	y_4	z_4
x_5	y_5	z_5
x_6	y_6	z_6
x_7	y_7	z_7

Adjacency Matrix **A**

	1	2	3	4	5	6	7
1		1			1		
2	1		1		1	1	
3		1		1		1	1
4			1				1
5	1	1				1	
6		1	1		1		1
7			1	1		1	

Figure 6. Demonstration of GNNs matrix.

To reduce the computational cost, convolution is conducted by a kernel g_θ with Chebyshev polynomial of order K.

$$g_\theta(L) = \sum_{k=0}^{K-1} \theta_k T_k(\tilde{L}) \tag{3}$$

where $\tilde{L} = 2L/\lambda_{\max} - I_n$ is graph Laplacian matrix. It is defined as $L = D - A$, where diagonal matrix $D_{ii} = \sum_j A_{ij}$. The $\lambda_{\max}$ is the variable to normalize matrix L. And θ_k are the coefficients of the Chebyshev polynomials. T_k can be expressed as:

$$T_k(x) = 2xT_{k-1}(x) - T_{k-2}(x) \tag{4}$$

with the initial condition $T_0 = 1$ and $T_1 = x$. This represents a Chebyshev polynomial of order K.

With the mesh filter shown above, the fast spectral convolution layer can be expressed as the following equation with $n \times F_{\text{in}}$ input and $n \times F_{\text{out}}$ output

$$y_j = \sum_{i=1}^{F_{in}} g_{\theta_{ij}}(L) x_i \tag{5}$$

where y_j means the j^{th} feature of output matrix, x_i is the i^{th} column of input feature matrix.

Mesh sampling layer

Another important layer used in the mesh encoder is the mesh sampling layer, which includes the down-sampling layer and up-sampling layer in autoencoder (Ranjan et al., 2018). In the encoder, important information is chosen by down-sampling layer and irrelevant information will be discarded.

The convolution layers used in this study represents mesh in multi-scales so that mesh sampling layer can capture the local and global geometric information. The down-sampling operation is conducted by a transform matrix $Q_{\text{down}} \in \{0, 1\}^{n \times m}$, where m is the number of vertices in the original mesh and n is the number of vertices in the down-sampled mesh. $Q_{\text{down}}(p, q) = 1$ means the q-th vertex is kept during the down-sampling, while $Q_{\text{down}}(p, q) = 0$ means the vertex is discarded. The transformation matrix is optimized to minimize the surface error by quadric matrices (Garland and Heckbert, 1997).

Loss function

The loss function of the neural network consists of three types of losses: mean squared error (MSE) loss, Kullback-Leibler divergence (KLD) loss and structural similarity loss.

MSE measures the average of pixel-wise error between the contours predicted by model (Y_i) and contours predicted by numerical simulation ($\hat{Y}_i$). It is defined mathematically by:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2 \tag{6}$$

KLD (Kullback and Leibler, 1951) measures the difference between one probability distribution and the reference probability distribution. In variational autoencoder, KL loss is the sum of all the KLD between the components in latent distribution and the standard normal distribution. With minimizing the KL loss, the latent distribution is closer to the standard normal, which can improve the interpolation and extrapolation ability of the

surrogate model. KLD can be defined by:

$$\begin{aligned}\mathrm{KLD}(p,\,q) &= -\int p(x)\log q(x)dx + \int p(x)\log p(x)dx \\ &= \frac{1}{2}\log(2\pi\sigma_2^2) + \frac{\sigma_1^2 + (\mu_1-\mu_2)^2}{2\sigma_2^2} - \frac{1}{2}(1 + log2\pi\sigma_1^2) \\ &= \log\frac{\sigma_2}{\sigma_1} + \frac{\sigma_1^2 + (\mu_1-\mu_2)^2}{2\sigma_2^2} - \frac{1}{2}\end{aligned} \tag{7}$$

As it is to measure the KLD between the components in latent distribution and the standard normal ($\sigma_2 = 1, \mu_2 = 0$), it can be simplified as the following equation for convenience:

$$KL_{\mathrm{loss}} = \sum_{i=1}^{n}(\sigma_i^2 + \mu_i^2 - \log(\sigma_i) - 1) \tag{8}$$

where μ is the mean vector, σ is the variance vector.

Structural similarity loss measure, or SSIM (Wang et al., 2003), is a method to measure the similarity between two figures. Here, it is used to optimize the neural network to make predicted contours and simulated contours more structurally similar. It is defined by:

$$\mathrm{SSIM}(Y_i,\,\hat{Y}_i) = \frac{(2\mu_{Y_i}\mu_{\hat{Y}_i} + c_1)(2\sigma_{Y_i\hat{Y}_i} + c_2)}{(\mu_{Y_i}^2 + \mu_{\hat{Y}_i}^2 + c_1)(\sigma_{Y_i}^2 + \sigma_{\hat{Y}_i}^2 + c_2)} \tag{9}$$

where μ_{Y_i}, $\mu_{\hat{Y}_i}$ are the mean of Y_i and $\hat{Y}_i$, $\sigma_{Y_i}^2$, $\sigma_{\hat{Y}_i}^2$ are the variances of Y_i and $\hat{Y}_i$, $\sigma_{Y_i\hat{Y}_i}$ is the covariance of Y_i and $\hat{Y}_i$, c_1, c_2 are two variables to stabilize the division with weak denominator.

Finally, the loss function of the surrogate model is defined by the following equation:

$$\mathrm{loss} = k_1\mathrm{MSE} + k_2\mathrm{KLD} + k_3(1 - \mathrm{SSIM}) \tag{10}$$

where three coefficients k_1, k_2 and k_3 are user-defined hyperparameters.

Demonstration of non-parametric surrogate model and application

In this section, the process of building a non-parametric surrogate model for LPES is introduced in details.

Introduction of low-pressure steam turbine exhaust system

LPES is designed to maximize the recovery of the kinetic energy leaving the low-pressure steam turbine and convert it to static pressure rise for the condenser. It usually consists of three parts: an axial-to-radial diffuser, an asymmetric collector, and an extension.

It is quite challenging to build surrogate model for LPES with the existing surrogate model methods because parameterization of geometry is problematic. Figure 7 shows a parameterization method to define the geometry of the system from (Ding, 2019), namely, the diffuser length ratio (L_1/L_0), diffuser area ratio (A_1/A_0), flow guide height ratio (H_1/L_0), diffuser turning angle ($\Delta\theta$), tip kink angle (Θ_{tip}), hub kink angle (Θ_{hub}), hood height ratio (H_1/L_0) and hood width ratio (W_1/L_0). With these geometric parameters, there are still many geometric features missing, for example, the curvature distribution of the diffuser, the height change of the collector, the width change of the extension and many more details. If using more parameters to describe the geometry, it needs more training cases. And the irrelevant parameters falsely selected by users will cause the over-fitting problem and reduce the accuracy of prediction.

In this surrogate model, there are two sets of LPES geometries defined by different parameterization methods. One is defined by 95 parameters, the other one by 66 parameters. The first one firstly defines the cross-section of diffuser and collector, and then revolves it with an ellipse equation to generate circumferential distribution. It also has asymmetric features in extension. The second one directly defines the cross-section along the axial direction with ellipse equations. Two genetic-based optimiztion systems with these two sets of parameters are used to

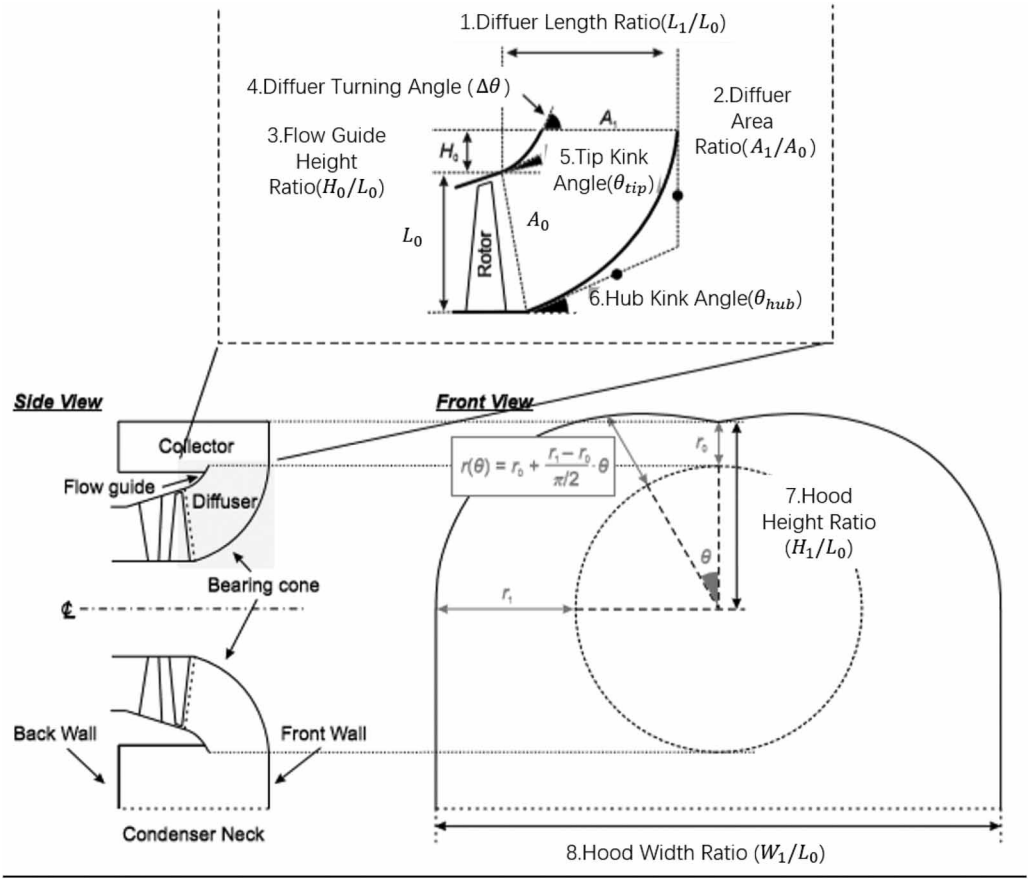


Figure 7. A typical down-flow type low-pressure exhaust system for large steam turbine. Figure adapted from (Ding, 2019).

accumulating dataset for surrogate model. With two types of geometries, the ability of processing geometries from different sources can be tested.

Mesh generation

The mesh generation process is shown in Figure 8. It starts from the coordinates of control points given by genetic algorithm. The control points are used to generate Non-Uniform Rational B-Spline (NURBS) surfaces, and then evaluate the NURBS surfaces to generate the surface mesh. These surface meshes, shown in Figure 9 are the input of surrogate model. Volume meshes are generated with the elliptic mesh generation method similar to (Spekreijse, 1995), which is by solving Laplace equation:

$$\begin{aligned} \frac{\partial^2 x}{\partial i^2} + \frac{\partial^2 x}{\partial j^2} + \frac{\partial^2 x}{\partial k^2} &= 0 \\ \frac{\partial^2 y}{\partial i^2} + \frac{\partial^2 y}{\partial j^2} + \frac{\partial^2 y}{\partial k^2} &= 0 \\ \frac{\partial^2 z}{\partial i^2} + \frac{\partial^2 z}{\partial j^2} + \frac{\partial^2 z}{\partial k^2} &= 0 \end{aligned} \tag{11}$$

The boundary condition is defined by the coordinates of surface mesh. Since the Laplace equation represents a potential field, equipotential lines do not intersect and are orthogonal at vertices. The volume mesh can be generated by solving x, y, and z coordinates potential field respectively. And the refinement of mesh can be done by refining the surface mesh, and then the volume mesh is also refined because surface mesh is the boundary condition of Laplace equation. The surface mesh can be refined by re-evaluating the NURBS surface with

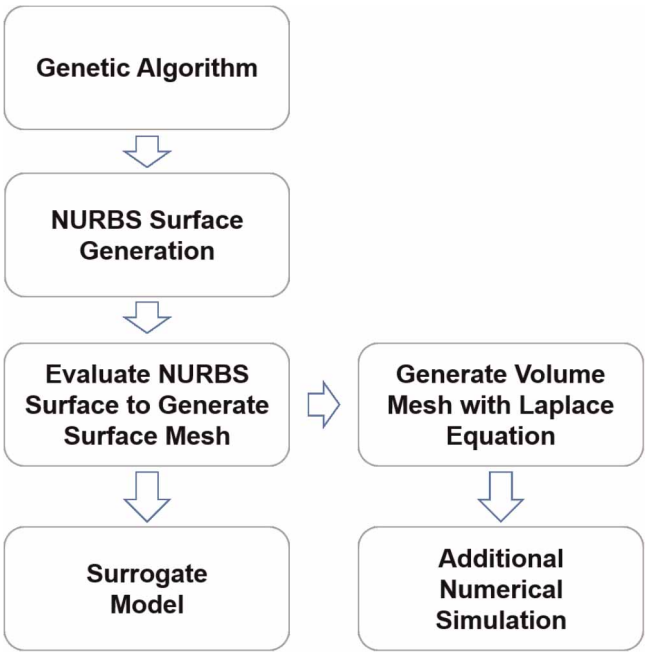


Figure 8. Framework of mesh generation process.

different distribution functions. This mesh generation method is also able to generate meshes for the fluid domain with the same topology, regardless changes in geometry.

Numerical simulation setup

Simulation domain

The fluid domain of numerical simulations includes: the last two stages of low-pressure steam turbine, an axis-asymmetric axial-to-radial diffuser, a collector and an extension. Figure 10 shows the geometry of two turbine stages, which represents the last two stages of a typical large steam turbine. It can generate a representative inlet boundary condition for the exhaust system. Figure 11 shows the geometry of the axial-to-radial diffuser, collector and extension in the downstream. The stage-hood interface treatment method is multiple mixing plane method with four blade passages. Each of them generates inlet boundary condition for a 90-degree diffuser section (Ding, 2019).

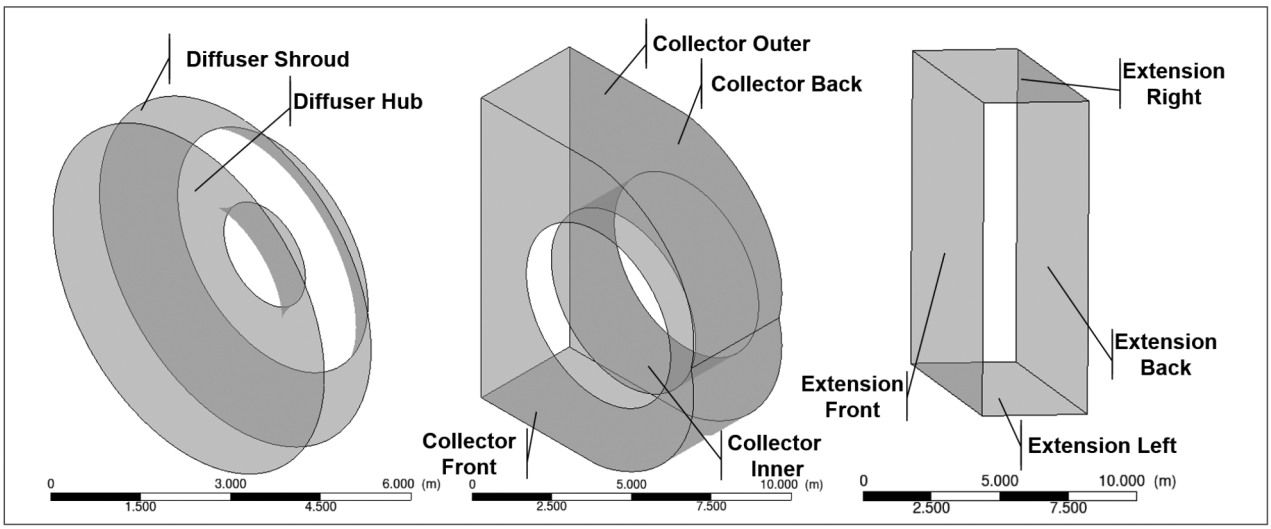


Figure 9. Ten surface meshes for surrogate model.

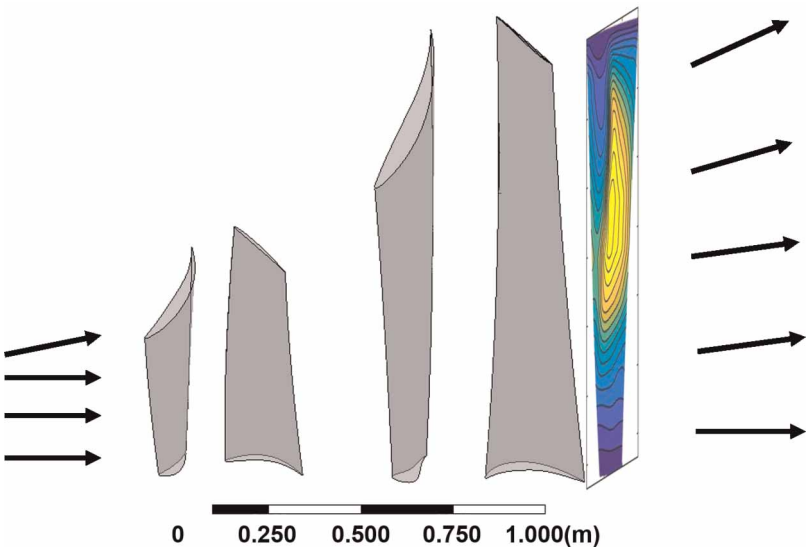


Figure 10. The last two low-pressure stage and contours to be predicted.

Simulation setup

The solver used for the numerical simulations is Ansys CFX, which is a widely-used commercial CFD solver for the research community and industry. The simulation is a Reynolds-averaged Navier–Stokes (RANS) simulation, which uses $k-\varepsilon$ turbulence model (Burton et al., 2013).

The inlet boundary condition is applied at the inlet of the simulation domain, which has total pressure and total enthalpy. The flow direction at the inlet is normal to boundary. And the outlet of the simulation domain is at the end of extension, which is applied with static pressure boundary conditions. The flow direction at the outlet is also normal to boundary. To perform the part-load simulations, the total pressure is reduced at the inlet to reduce the mass flow rate and work output. The static pressure at the outlet is 6.2 kPa according to the steam property at the condenser. For the property of the working fluid, steam, IAWPS data has been used, which is embedded in CFX and also widely used in the research community and industry.

Another setup worth mentioning is the interface treatment method between stages and the inlet of the diffuser. Because the downstream of the diffuser are asymmetric, it is necessary to model the circumferential non-uniformity. The multiple mixing plane method (Burton et al., 2013) is used in this study. Figure 12 shows only four-blade passages are simulated to generate inlet flow conditions for the low-pressure exhaust hood, which means the outlet boundary condition of one blade passage are copied to cover a 90-degree section of the diffuser.

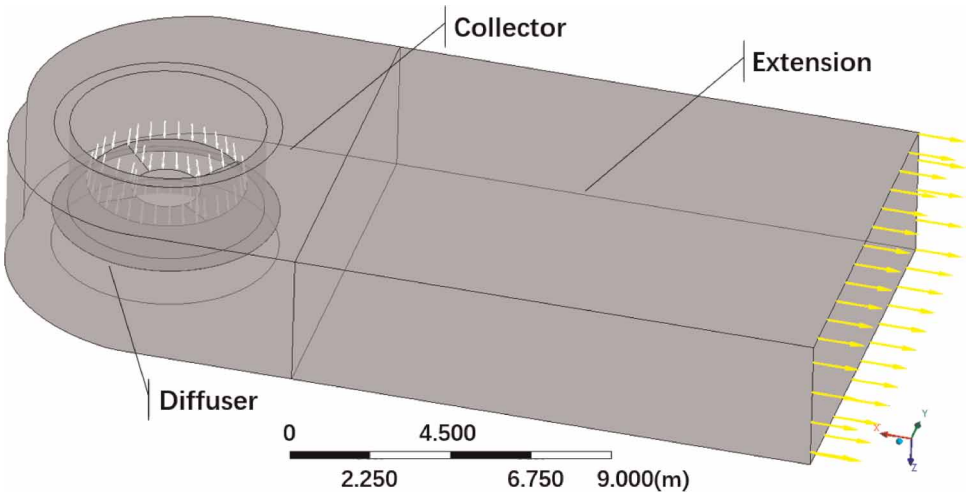


Figure 11. The geometry of exhaust hood. The white arrows represent inlet boundary and the yellow arrows represent outlet boundary.

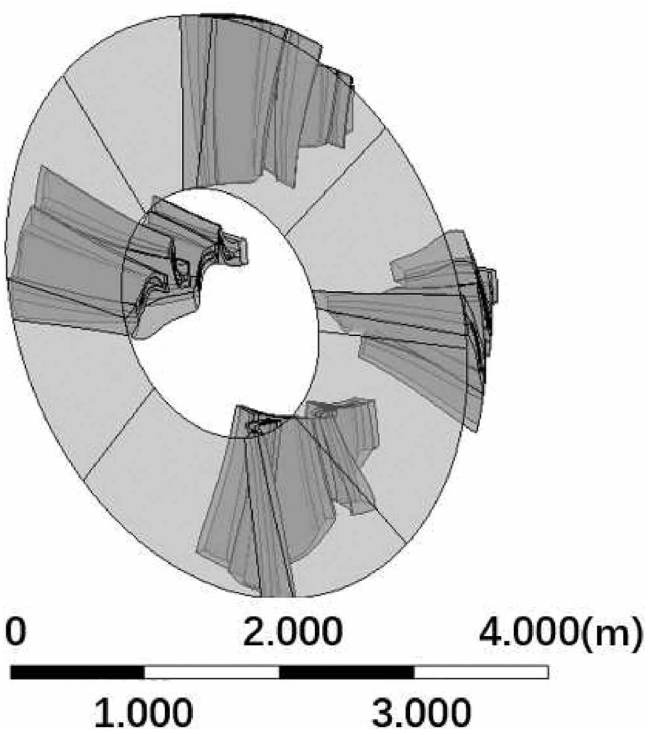


Figure 12. Demonstration of multiple mixing plane method. Four blade passages are modelled. Each of them generates inlet boundary condition for a 90-degree section of exhaust hood.

Multiple mixing plane method, though losing accuracy with only 4 blade passages, can reduce the computational cost considerably.

Processing of numerical simulation results

The objective of GA-based optimization is to increase power output of the last two low pressure steam turbine stages. It is calculated by the difference between the total energy fluxes pass through inlet and outlet of the last two stages, which is summed value of all the elements of contour map of energy flux. Assuming the system is adiabatic, the power of each element is obtained by the product of local total enthalpy and local mass flow rate:

$$\begin{aligned} \Delta \dot{W} &= \dot{m}h_{02} - \dot{m}h_{01} \\ &= \sum_{i=1}^{n} \dot{m}_{i_2} h_{i_2} - \sum_{i=1}^{n} \dot{m}_{i_1} h_{i_1} \end{aligned} \tag{12}$$

Since the inlet boundary condition is known in the simulation, only the total enthalpy contours and mass flow rate contours at the outlet of the last two stages are extracted from numerical simulations to generate power contours as shown in Equation 12, which will also be the output of the surrogate model. Because of the multiple mixing plane method, there are four blade passages for each simulation, and five workload conditions for each design. Admittedly, there is certainly some uncertainties in the numerical simulations, but it is not primary concern in this paper since the key of this study is to develop a new surrogate model method.

Neural network setup

The neural network is built under the framework of Pytorch. Figure 3 shows the main structure of network. Figure 13 shows the change of feature dimension through the network. The input is 10 bounding surface meshes of fluid domain, which have 195,200 vertices in total. All the samples need to be interpolated to the same number of mesh vertices to represent the geometry at the same details level. Therefore, the input data is the coordinates of vertices (195,200 × 3) and adjacency matrix (195,200 × 195,200). The mesh encoder has 6 mesh encoder blocks. A smaller filter size (16) in the front four blocks can capture local geometric features, and a larger filter size (32) in the rear two blocks can capture global geometric features better. After the mesh

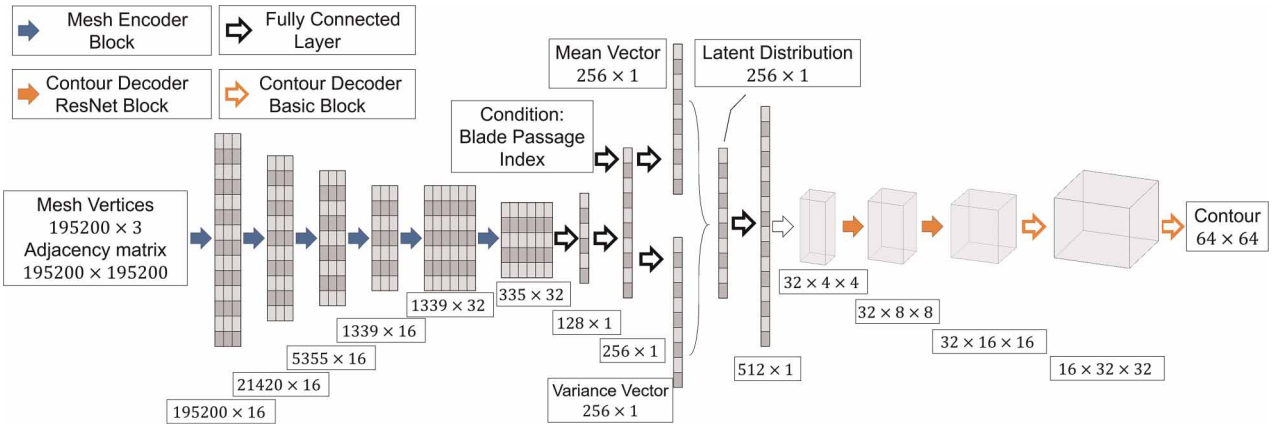


Figure 13. The change of feature dimension in the network.

encoder, the mesh is compressed into a 128 × 1 latent vector. Conditions (blade passage index) are added to the latent vector. Then, two fully connected layers are used to estimate the mean vector and variance vector of latent distribution. Conditions (blade passage index) are added into the latent distribution again. The latent distribution is reshaped to be the input of the contour decoder. In the contour decoder, the first two blocks are ResNet blocks, and the rear two blocks are basic blocks.

Test result

In this demonstration, 550 cases are used for training, and the test dataset has 32 cases. Each case contains total enthalpy flux distribution contours for 4 blade passages at 5 workload conditions (640 contours in total).

Figure 14 is to demonstrate the ability of predicting contours and capturing flow features. It shows some representative contours of the test cases, which presents some typical flow features of 5 workload conditions. Most flow features are predicted well in Figure 14, which provides more information than existing surrogate model methods. For example, the separations in the near hub regions are well captured in the contours of 50% workload condition. And the vortices are also well predicted in the contours of 70% workload condition. These information can tell users the reason of performance improvement during the optimization process rather than only several performance metrics.

Table 1 summarizes the results of all 640 test contours. The averaged summed value error of dataset is 0.86%, which is able to screen suitable designs and accelerate the optimization process. The last few optimization iterations may still rely on numerical simulation. Also, test dataset is consist of two sub-datasets that contain geometries defined by two parameterization methods, both of which have 16 cases and 320 contours. In Table 1, The similarity measure of sub-dataset1 and sub-dataset2 are 0.9580 and 0.9609 respectively, which has no significant difference. This proves the generalization capability of the proposed method in processing geometries defined by different parameterization methods.

Figure 15 plots the predicted performance against results from numerical simulations. Most of points are close to $y = x$, which means low prediction error. There is a group of points significantly beyond the $y = x$ line, and the error increases with the growth of the summed value. They are all from 2 designs, which are so geometrical different from designs in the training dataset that surrogate model is unable to predict the performance well. In this case, users can identify these designs by KLD and perform numerical simulations to evaluate the performance rather than using surrogate model.

Discussion

The new surrogate model method established a mapping relationship between the surface mesh of fluid domain and two-dimensional distribution of flow variables. Using GNNs to extract geometric information, the application of this new method can be extended beyond the area of aerodynamics optimization. Thanks to its ability of processing both structured and unstructured mesh, it is also applicable in various problems in different fields, which are solutions of partial differential equations, traditionally using finite element analysis and electromagnetic analysis methods.

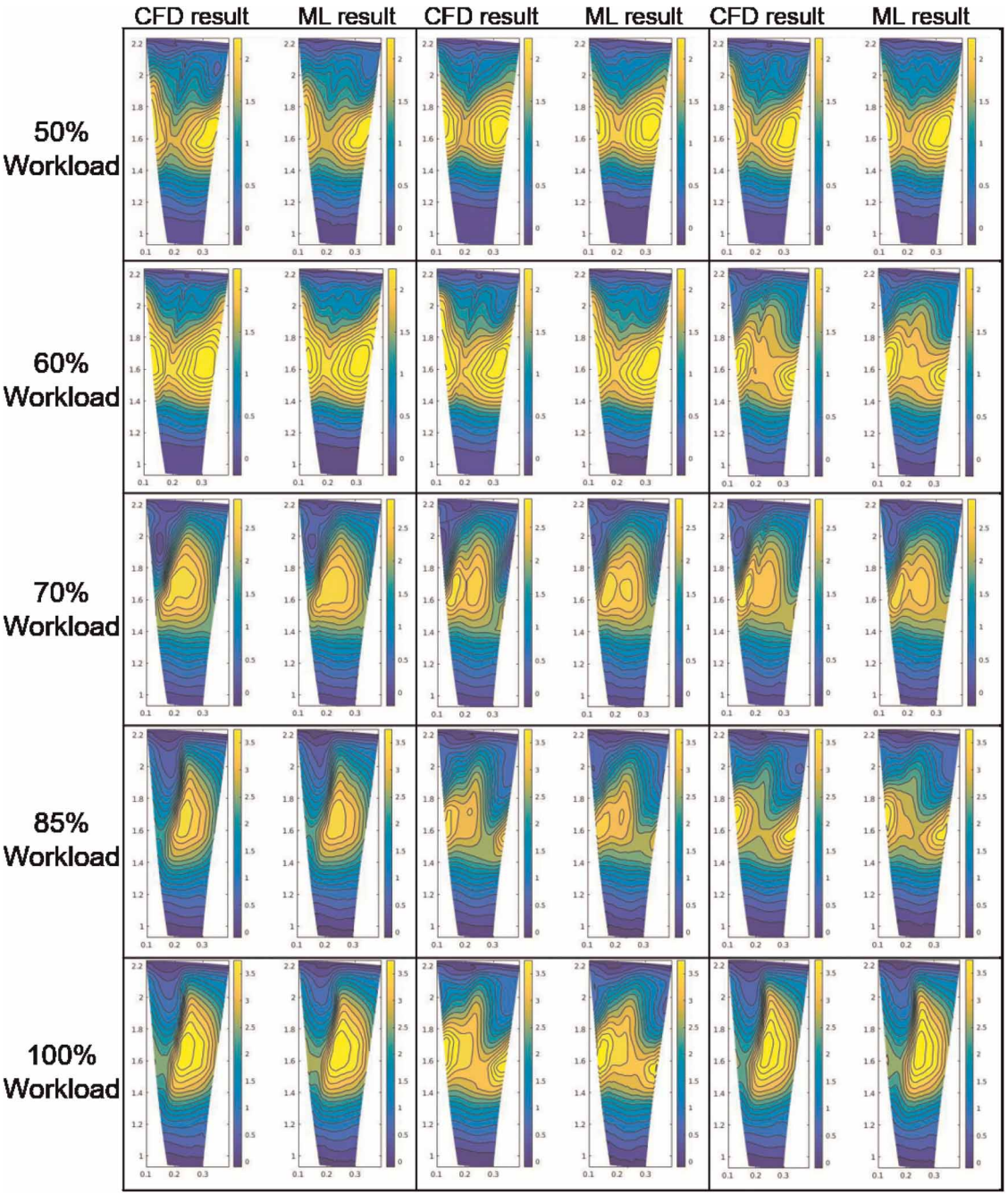


Figure 14. Contour results of numerical simulation and machine learning of some typical flow features at five workload conditions. (50%, 60%, 70%, 85% and 100%). Unit: 1×10^4 kJ/s.

This method can also be used as an inverse design method. It can be achieved by exchanging the input and output of the mapping relationship built in this paper. To be more specific, users can import the desired two-dimensional distributions of physical properties into the contour encoder, and the designs could be created by the mesh decoder.

In the application scenarios, one common problem of surrogate model method is users do not know whether they can trust the result of surrogate model or not. If the new design is quite different from designs in the training dataset, it is better to perform numerical simulations. To tackle this problem, the new method can identify the new design by calculating the KLD of their latent vectors. Higher KLD usually means the new design needs numerical simulation to evaluate its performance.

Table 1. Summary of result. Sub-dataset1 are geometries defined by 95 parameters and sub-dataset2 are defined by 66 parameters.

Work Condition	Performance Metrics	Dataset	Sub-dataset1	Sub-dataset2
50%	MSE Loss	0.0047	0.0038	0.0056
	Similarity Measure	0.9541	0.9520	0.9561
	Summed Value Error	0.0060	0.0068	0.0052
60%	MSE Loss	0.0068	0.0042	0.0094
	Similarity Measure	0.9482	0.9517	0.9447
	Summed Value Error	0.0096	0.0122	0.0070
70%	MSE Loss	0.0017	0.0016	0.0019
	Similarity Measure	0.9632	0.9602	0.9661
	Summed Value Error	0.0089	0.0095	0.0082
85%	MSE Loss	0.0016	0.0017	0.0016
	Similarity Measure	0.9649	0.9618	0.9679
	Summed Value Error	0.0090	0.0104	0.0076
100%	MSE Loss	0.0019	0.0019	0.0020
	Similarity Measure	0.9668	0.9641	0.9695
	Summed Value Error	0.0094	0.0108	0.0081
Mean	MSE Loss	0.0033	0.0026	0.0041
	Similarity Measure	0.9594	0.9580	0.9609
	Summed Value Error	0.0086	0.0099	0.0072

Conclusion

This study demonstrated a novel non-parametric surrogate model method with application on LPES design and optimization. The new method directly takes surface mesh as the input, which reduces the uncertainties introduced by manual parameterization and loss of geometric information. This feature gives the method greater advantages in building surrogate model for designs with complex geometries. In the test, the average summed value error of 640 contours is 0.86%.

This method also shows high flexibility and compatibility. Because the input of this method is surface mesh of simulation domain, it can take geometries of the same topology as its database. This means it can process geometries defined by different parameterization methods. This feature is very useful for further increasing the size of database of surrogate model using variable sources of data.

Compared with existing surrogate model methods, this new method can also predict two-dimensional distribution of variables (contour maps) based on surface mesh. Contours can help designers to identify physical mechanism, improve designs and for many other purposes. In the test, the average similarity score of 640 contours is 0.9594, which indicates that most of the pertinent flow features have been captured.

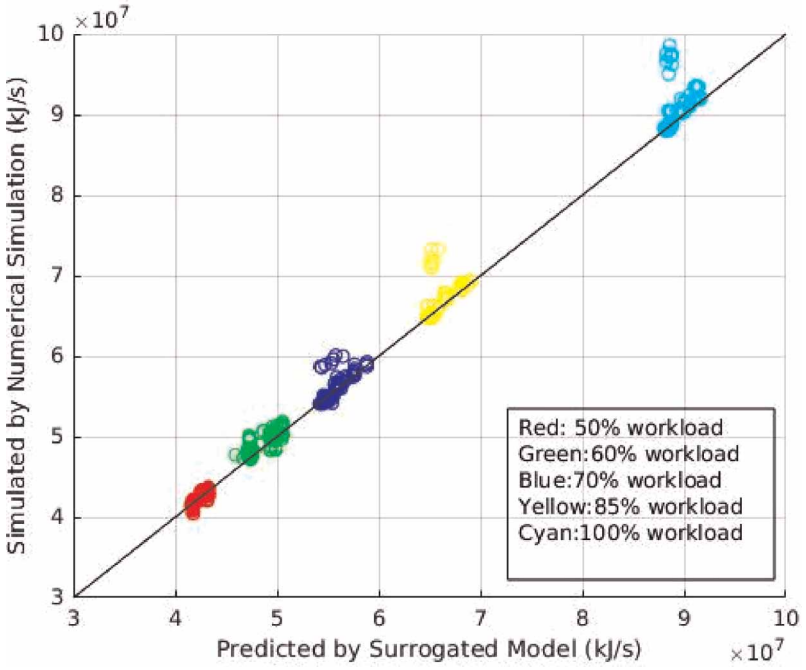


Figure 15. Comparison between results of numerical simulation and surrogate model. The closer to the line $Y = X$ means higher accuracy.

The essence of this method is the establishment of a mapping relationship between the surface mesh of the simulation domain and distributions of physical variables on certain two dimensional cut planes. With this method, the optimization process of an engineering system with complex geometry features such as LPES could be accelerated by utilizing the database created by various means to evaluate the performance of new designs.

Nomenclature

GNNs	Graph Neural Networks
CNNs	Convolutional Neural Networks
LPES	Low-pressure Steam Turbine Exhaust System
AE	Autoencoder
VAE	Variational Autoencoder
CVAE	Conditional Variational Autoencoder
KLD	Kullback-Leibler divergence
MSE	Mean Squared Error
SVE	Summed Value Error
SSIM	Structural Similarity Index Measure

Funding sources

Shanghai Turbine Works Co., Ltd.

Competing interests

Jiajun Cao declares that he has no conflict of interest. Qingbiao Li declares that he has no conflict of interest. Liping Xu declares that he has no conflict of interest. Rui Yang declares that he has no conflict of interest. Yuejin Dai declares that he has no conflict of interest.

References

Blei D. M., Kucukelbir A., and McAuliffe J. D. (2017). Variational inference: a review for statisticians. *Journal of the American Statistical Association*. 112 (518): 859–877. https://doi.org/10.1080/01621459.2017.1285773

Burton Z., Ingram G. L., and Hogg S. (2013). A literature review of low pressure steam turbine exhaust hood and diffuser studies. *Journal of Engineering for Gas Turbines and Power*. 135 (6). https://doi.org/10.1115/1.4023611

Claeskens G. and Hjort N. L. (2008). Lack-of-fit and goodness-of-fit tests, Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge: Cambridge University Press. p. 227–247.

Defferrard M., Bresson X., and Vandergheynst P. (2016). Convolutional neural networks on graphs with fast localized spectral filtering. https://arxiv.org/abs/1606.09375

Ding B. (2019). Aerodynamics of low pressure steam turbine exhaust systems. PhD thesis, University of Cambridge.

Garland M. and Heckbert P. S. (1997). Surface simplification using quadric error metrics. In Conference on Computer Graphics and Interactive Techniques, pp. 209–216.

Gutmann H.-M. (2001). A radial basis function method for global optimization. *Journal of Global Optimization*. 19 (3): 201–227. https://doi.org/10.1023/A:1011255519438

Hammond D. K., Vandergheynst P., and Gribonval R. (2011). Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis*. 30 (2): 129–150. https://doi.org/10.1016/j.acha.2010.04.005

Hecker S., Penkner A., Pfeiffer J., Glos S., and Musch C. (2017). Prediction of the windage heating effect in steam turbine labyrinth seals. *Journal of the Global Power and Propulsion Society*. 1: 84–94. https://doi.org/10.22261/JGPPS.ETJLRM

Jung Y., Kang K., Cho H., and Lee I. (2021). Confidence-based design optimization for a more conservative optimum under surrogate model uncertainty caused by gaussian process. *Journal of Mechanical Design*. 143 (9): 091701. https://doi.org/10.1115/1.4049883

Kingma D. P. and Welling M. (2019). An introduction to variational autoencoders. *arXiv preprint arXiv:1906.02691*.

Kipouros T., Molinari M., Dawes W. N., Parks G. T., Savill M., and Jenkins K. W. (2007). An investigation of the potential for enhancing the computational turbomachinery design cycle using surrogate models and high performance parallelisation. In Turbo Expo: Power for Land, Sea, and Air, Vol. Volume 6: Turbo Expo 2007, Parts A and B, pp. GT2007–28106.

Kullback S. and Leibler R. A. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*. 22 (1): 79–86. https://doi.org/10.1214/aoms/1177729694

Lal A. and Datta B. (2018). Development and implementation of support vector machine regression surrogate models for predicting groundwater pumping-induced saltwater intrusion into coastal aquifers. *Water Resources Management*. 32 (7): 2405–2419. https://doi.org/10.1007/s11269-018-1936-2

Lee K.-D. and Kim K.-Y. (2009). Optimization of a fan-shaped hole for film cooling using a surrogate model. In Turbo Expo: Power for Land, Sea, and Air, Vol. Volume 3: Heat Transfer, Parts A and B, pp. GT2009–59520.

Lee S., Lee S., Kim K.-H., Lee D.-H., Kang Y.-S., and Rhee D.-H. (2014). Optimization framework using surrogate model for aerodynamically improved 3d turbine blade design. In Turbo Expo: Power for Land, Sea, and Air, Vol. Volume 2B: Turbomachinery, American Society of Mechanical Engineers, pp. GT2014–26571.

Mengistu T. and Ghaly W. (2008). Aerodynamic optimization of turbomachinery blades using evolutionary methods and ann-based surrogate models. *Optimization and Engineering*. 9 (3): 239–255. https://doi.org/10.1007/s11081-007-9031-1

Myers R. H., Montgomery D. C., and Anderson-Cook C. M. (2016.) Response surface methodology: process and product optimization using designed experiments: John Wiley & Sons.

Persico G., Rodriguez-Fernandez P., and Romei A. (2019). High-fidelity shape optimization of non-conventional turbomachinery by surrogate evolutionary strategies. *Journal of Turbomachinery*. 141 (8): 081010. https://doi.org/10.1115/1.4043252

Ranjan A., Bolkart T., Sanyal S., and Black M. J. (2018). Generating 3D faces using convolutional mesh autoencoders. In European Conference on Computer Vision, pp. 725–741.

Sakata S., Ashida F., and Zako M. (2003). Structural optimization using kriging approximation. *Computer Methods in Applied Mechanics and Engineering*. 192 (7–8): 923–939. https://doi.org/10.1016/S0045-7825(02)00617-5

Samad A. and Kim K.-Y. (2008). Stacking and thickness optimization of a compressor blade using weighted average surrogate model. In Turbo Expo: Power for Land, Sea, and Air, Vol. Volume 6: Turbomachinery, Parts A, B, and C, pp. GT2008–50262.

Samad A., Kim K.-Y., Goel T., Haftka R. T., and Shyy W. (2008). Multiple surrogate modeling for axial compressor blade shape optimization. *Journal of Propulsion and Power*. 24 (2): 301–310. https://doi.org/10.2514/1.28999

Sanchez-Lengeling B., Reif E., Pearce A., and Wiltschko A. B. (2021). A gentle introduction to graph neural networks. *Distill*. https://distill.pub/2021/gnn-intro

Scarselli F., Gori M., Tsoi A. C., Hagenbuchner M., and Monfardini G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*. 20 (1): 61–80. https://doi.org/10.1109/TNN.2008.2005605

Schnoes M., Voß C., and Nicke E. (2018). Design optimization of a multi-stage axial compressor using throughflow and a database of optimal airfoils. *Journal of the Global Power and Propulsion Society*. 2: 516–528. https://doi.org/10.22261/JGPPS.W5N91I

Shuiguang T., Hang Z., Huiqin L., Yue Y., Jinfu L., and Feiyun C. (2020). Multi-objective optimization of multistage centrifugal pump based on surrogate model. *Journal of Fluids Engineering*. 142 (1). https://doi.org/10.1115/1.4043775

Spekreijse S. (1995). Elliptic grid generation based on laplace equations and algebraic transformations. *Journal of Computational Physics*. 118 (1): 38–61. https://www.sciencedirect.com/science/article/pii/S0021999185710789 https://doi.org/10.1006/jcph.1995.1078

Valueva M., Nagornov N., Lyakhov P., Valuev G., and Chervyakov N. (2020). Application of the residue number system to reduce hardware costs of the convolutional neural network implementation. *Mathematics and Computers in Simulation*. 177: 232–243. https://doi.org/10.1016/j.matcom.2020.04.031

Wang Z., Simoncelli E. P., and Bovik A. C. (2003). Multiscale structural similarity for image quality assessment. In Asilomar Conference on Signals, Systems & Computers, Vol. 2. IEEE, pp. 1398–1402.

Wu Z., Pan S., Chen F., Long G., Zhang C., and Philip S. Y. (2020). A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*. 32 (1): 4–24. https://doi.org/10.1109/TNNLS.2020.2978386