Deep Monte Carlo Quantile Regression for Quantifying Aleatoric Uncertainty in Physics-informed Temperature Field Reconstruction

Abstract—For the temperature field reconstruction (TFR), a complex image-to-image regression problem, the convolutional neural network (CNN) is a powerful surrogate model due to the convolutional layer’s good image feature extraction ability. However, a lot of labeled data is needed to train CNN, and the common CNN cannot quantify the aleatoric uncertainty caused by data noise. In actual engineering, the noiseless and labeled training data is hardly obtained for the TFR. To solve these two problems, this paper proposes a deep Monte Carlo quantile regression (Deep MC-QR) method for reconstructing the temperature field and quantifying aleatoric uncertainty caused by data noise. On the one hand, the Deep MC-QR method uses physical knowledge to guide the training of CNN. Thereby, the Deep MC-QR method can reconstruct an accurate TFR surrogate model without any labeled training data. On the other hand, the Deep MC-QR method constructs a quantile level image for each input in each training epoch. Then, the trained CNN model can quantify aleatoric uncertainty by quantile level image sampling during the prediction stage. Finally, the effectiveness of the proposed Deep MC-QR method is validated by many experiments, and the influence of data noise on TFR is analyzed.

Index Terms—Physics-informed, convolutional neural network, aleatoric uncertainty, temperature field reconstruction, quantile regression

I. INTRODUCTION

Heat monitoring analysis plays an important role in many expensive engineering systems [1]–[3], such as airplanes, satellites, rockets, etc. If the system overheats, it can lead to system failure or even serious disaster. Generally, the heat monitoring analysis is performed using the temperatures measured by many sensors. However, a limited number of temperature monitoring sensors can not provide detailed temperature information at any location of the system, which will result in the heat monitoring analysis results being inaccurate. Therefore, it is essential to reconstruct the temperature field of the whole system based on the temperature measured by sensors.

In recent years, CNN has been applied in many engineering problems [4]–[6] due to its powerful image feature extraction ability. For the TFR, it is a complex image-to-image regression problem. Thus, this paper chooses CNN to reconstruct the temperature field using sensor monitoring temperatures as inputs. It is important to notice that engineers hardly obtain each input’s label, i.e., temperature field. However, the training of CNN usually requires a large amount of labeled training data. In order to solve the problem of insufficient labeled data, the physics-informed CNN is used to bearing fault detection [7], partial differential equation solution [8]–[10], topology optimization [11]. For the TFR, the steady-state temperature field meets the Laplace equation [5], [12] and some boundary conditions. Thus, this paper uses physical knowledge to guide the training of CNN.

The training of CNN is essentially extracting model parameters from training data. The trained model is then used to make predictions. Generally, the training data may include data noise which will result in aleatoric uncertainty [13] in the prediction results. Unlike epistemic uncertainty [14] can be eliminated by increasing the number of training data, aleatoric uncertainty is inherent to the training data with noise. In recent years, many methods [15]–[19] have been studied to quantify aleatoric uncertainty. Given the Gaussian distribution input hypothesis, one common approach of aleatoric uncertainty quantification is that one output of the neural network is used to estimate the conditional variance of the prediction by maximizing likelihood estimation [15], [16]. The obvious disadvantage of this
The main contributions of this paper include the following two points:

- By physical knowledge to guide the training of CNN, the proposed Deep MC-QR method can reconstruct an accurate TFR surrogate model without any labeled training data;
- By constructing a quantile level image for each monitoring point temperature image, the proposed Deep MC-QR method can quantify aleatoric uncertainty.

## II. RELATED WORK

### A. Quantile Regression

Suppose that there is a stochastic system \( Y = f(X) \) with the random input variable \( X \). The cumulative distribution function of the output \( Y \) taking the value \( y \) is \( F_{Y|X=x}(y|x) = P(Y \leq y) \). Thus, the \( \tau \) quantile \( Q_\tau(Y|X=x) \) of the output \( Y \) is

\[
Q_\tau(Y|X=x) = F_{Y|X=x}^{-1}(\tau),
\]

where \( \tau \) \((0 \leq \tau \leq 1) \) is the quantile level. The quantile regression constructs a surrogate model \( Y = \hat{f}_\tau(X) \) to approximate the stochastic system \( Y = f(X) \). Given \( K \) training data \( \{(x_k, y_k) | k = 1, 2, \ldots, K\} \), the surrogate model \( \hat{Y} = \hat{f}_\tau(X) \) is trained by minimizing the pinball loss function \( \mathcal{L}[y, \hat{f}_\tau(y)] \) \([18], [21]\), i.e.,

\[
\mathcal{L}[y_k, \hat{f}_\tau(x_k)] = \frac{1}{K} \sum_{k=1}^{K} \ell_\tau(y_k, \hat{f}_\tau(x_k)),
\]

\[
\ell_\tau(y_k, \hat{f}_\tau(x_k)) = \begin{cases} 
\tau(y_k - \hat{y}_k), & y_k \geq \hat{y}_k \\
(1-\tau)(\hat{y}_k - y_k), & y_k < \hat{y}_k 
\end{cases}
\]

where \( \hat{y}_k = \hat{f}_\tau(x_k) \) is the prediction corresponding to the input \( x_k \).

### B. Temperature Field Reconstruction

As shown in Fig.1, there is an green rectangular area \( \Omega \) with length \( W \) and width \( H \), where the grey areas \( \Omega_{HS} \) of circles, capsules and rectangles are heat sources. The yellow rectangle with width \( \delta \) is the position of the heat sink. The diagonal shaded area around \( \Omega \) is adiabatic. Besides, the red points are the positions \( \Omega_{MP} \) of the temperature monitoring sensors. Supposed that there are \( N_{MP} \) temperature monitoring sensors in the rectangular area \( \Omega \).

For this green rectangular area \( \Omega \), its steady-state temperature field \( T \) meets the Laplace equation \([5], [12]\), i.e.,

\[
\frac{\partial^2 T}{\partial u^2} + \frac{\partial^2 T}{\partial v^2} + \varphi(u,v) = 0, \quad (u,v) \in \Omega,
\]

and the boundary condition

\[
T(u,v) = T_0, \quad (u,v) \in \Omega_{BC},
\]

where \( (u,v) \) denotes the position coordinate, \( T_0 \) is a constant temperature, \( \varphi \) is the heat source intensity distribution depending on the power of heat source, and \( \Omega_{BC} \) denotes the boundary (red line in Fig.1). Generally, the heat source intensity distribution \( \varphi \) is unknown in the TFR problem. Thus, the TFR problem only considers the heat conduction in the area \( \Omega_{NC} \) without heat sources, i.e.,

\[
\frac{\partial^2 T}{\partial u^2} + \frac{\partial^2 T}{\partial v^2} = 0, \quad (u,v) \in \Omega_{NC}.
\]

For the TFR problem, its objective is to reconstruct the temperature field of the rectangular area \( \Omega \) from the temperatures of some monitoring points, as shown in Fig.2.

### III. DEEP MONTE CARLO QUANTILE REGRESSION

#### A. Physics-informed Deep MC-QR model

In this paper, the rectangular area \( \Omega \) is discretized into a \( h \times w \) two-dimensional array. According to the positions \( \Omega_{MP} \) of the temperature monitoring sensors, the corresponding element values in the \( h \times w \) two-dimensional array are equal to the the monitoring point temperatures. The rest of element values are set to be zero. Thereby, a monitoring point (MP) temperature image \( T_{MP} \) is constructed as shown in Fig.3 (left). Besides, this paper also builds a quantile level image \( \tau \) as shown in Fig.3 (right). Different from the MP temperature image \( T_{MP} \), the values of element in the positions \( \Omega_{MP} \) are equal to \( \tau \).
for the quantile level image \( \tau \), where \( \tau \) is randomly sampled from the uniform distribution \( U(0,1) \).

This paper builds a physics-informed Deep MC-QR model which can use the MP temperature image \( T_{MP} \) and the quantile level image \( \tau \) to reconstruct the temperature field of the rectangular area \( \Omega \), as shown in Fig. 4. The physics-informed Deep MC-QR model with two input channels includes two U-net [6] models, i.e., U-net-1 and U-net-2. Firstly, U-net-1 extracts the feature map of input \( (T_{MP}, \tau) \). Then, the feature map is flipped diagonally to ensure the reconstruction accuracy of temperature field in the upside and right-side boundaries [12]. Finally, U-net-2 uses the flipped feature map to predict the temperature field \( T \) of the rectangular area \( \Omega \).

Supposed that the physics-informed Deep MC-QR model is denoted as MQNN \( (T_{MP}, \tau; \theta) \), where \( \theta \) denotes the parameters of U-net-1 and U-net-2. Based on the training dataset \( \{ (T_{MP}^i, \tau_i) | i = 1,2, \cdots, N \} \), the parameters \( \theta \) are learned by minimizing the following proposed physics-informed loss function \( L(\theta) \).

- **Quantile MP temperature loss** \( L_\tau(\theta) \)

For the \( i \)th MP temperature image \( T_{MP}^i \), the predicted temperature field \( T \) of the satellite subsystem is \( T_i = \mathcal{M}(T_{MP}^i, \tau_i; \theta) \). For \( (u,v) \in \Omega_{MP} \), the error \( \ell_{\tau_i}(u,v;\theta) \) is

\[
\ell_{\tau_i}(u,v;\theta) = \tau_i \left[ T_{MP}^i(u,v) - T_i(u,v) \right]
\]

for the condition \( T_{MP}^i(u,v) \geq T_i(u,v) \), and

\[
\ell_{\tau_i}(u,v;\theta) = (1 - \tau_i) \left( T_i(u,v) - T_{MP}^i(u,v) \right)
\]

for the condition \( T_{MP}^i(u,v) < T_i(u,v) \), where \( \tau_i \sim U(0,1) \). Thus, the quantile MP temperature loss \( L_\tau(\theta) \) is

\[
L_\tau(\theta) = \frac{1}{N} \frac{1}{N_{MP}} \sum_{i=1}^{N} \sum_{(u,v) \in \Omega_{MP}} \ell_{\tau_i}(u,v;\theta).
\]

- **Laplace equation loss** \( L_{LE}(\theta) \)

Assumed that each discretized MP temperature image \( T_{MP}^i \) has \( N_{NC} \) points for \( (u,v) \in \Omega_{NC} \). Discretizing the Laplace equation (5), \( \partial^2 T_i(u_j,v_k)/\partial u_j^2 \) and \( \partial^2 T_i(u_j,v_k)/\partial v_k^2 \) are transformed respectively to be

\[
\frac{\partial^2 T_i(u_j,v_k)}{\partial u_j^2} = \frac{\Delta T_{j+1} \Delta u_j - \Delta T_{j-1} \Delta u_{j+1}}{\Delta u_{j-1} \Delta u_j},
\]

and

\[
\frac{\partial^2 T_i(u_j,v_k)}{\partial v_k^2} = \frac{\Delta T_{k+1} \Delta v_k - \Delta T_{k-1} \Delta v_{k+1}}{\Delta v_{k-1} \Delta v_k},
\]

where \( \Delta u_j = u_{j-1} - u_j \), \( \Delta v_k = v_{k} - v_{k-1} \). Thus, the Laplace equation loss \( L_{LE}(\theta) \) for \( (u,v) \in \Omega_{NC} \) is

\[
L_{LE}(\theta) = \frac{1}{N} \frac{1}{N_{NC}} \sum_{i=1}^{N} \sum_{(u,v) \in \Omega_{NC}} \left[ \frac{\partial^2 T_i(u,v)}{\partial u^2} + \frac{\partial^2 T_i(u,v)}{\partial v^2} \right]^2.
\]

- **Boundary condition loss** \( L_{BC}(\theta) \)

According to (4), the boundary condition loss \( L_{BC}(\theta) \) for \( N \) MP temperature images \( \{ T_{MP}^i | i = 1, 2, \cdots, N \} \) is

\[
L_{BC}(\theta) = \frac{1}{N} \frac{1}{N_{BC}} \sum_{i=1}^{N} \sum_{(u,v) \in \Omega_{BC}} \left[ T_i(u,v) - T_0 \right]^2.
\]

where \( N_{BC} \) means the number of points \( (u,v) \) in the area \( \Omega_{BC} \) of each discretized MP temperature image \( T_{MP}^i \).

- **TV regularization** \( L_{TV}(\theta) \)

TV regularization [12], [22], as shown in (13), can maintain the smoothness of the image,

\[
\mathcal{R}_{TV} = \int_{\Omega} \left[ \left| \frac{\partial f(u,v)}{\partial u} \right|^2 + \left| \frac{\partial f(u,v)}{\partial v} \right|^2 \right] \frac{\beta}{2} \ dudv,
\]

where \( f(u,v) \) is a continuous function. Due to the property that the steady-state temperature field will not mutate sharply, this paper adopts TV regularization to assist the training of model MQNN \( (T_{MP}, \tau; \theta) \). The TV regularization \( L_{TV}(\theta) \) is transformed to be (14) by the finite-difference approximation, i.e.,

\[
L_{TV}^u(\theta) = \frac{1}{h} \sum_{j=1}^{w-1} \left[ \Delta T_j^T \right]^2,
\]

\[
L_{TV}^v(\theta) = \frac{1}{w} \sum_{k=1}^{w-1} \left[ \Delta T_k^T \right]^2,
\]

\[
L_{TV}(\theta) = \frac{1}{N} \sum_{i=1}^{N} [L_{TV}^u(\theta) + L_{TV}^v(\theta)],
\]

where \( \Delta T_j^T = T_i(\hat{u}_{j+1}, v_k) - T_i(u_j, v_k) \) and \( \Delta T_k^T = T_i(u_j, \hat{v}_{k+1}) - T_i(u_j, v_k) \). Thus, the proposed physics-informed loss function \( L(\theta) \) for learning the parameters \( \theta \) of model MQNN \( (T_{MP}, \tau; \theta) \) is

\[
L(\theta) = \alpha_1 L_\tau(\theta) + \alpha_2 L_{LE}(\theta) + \alpha_3 L_{BC}(\theta) + \alpha_4 L_{TV}(\theta),
\]
where $\alpha_1$, $\alpha_2$, $\alpha_3$, and $\alpha_4$ are hyperparameters.

B. Model Monte Carlo Training Algorithm

This section proposes the model Monte Carlo training algorithm to learning the parameters $\theta$. Different from the general model training methods, the model Monte Carlo training algorithm keeps changing the training data by randomly sampling quantile level images in each epoch. The detailed description is as follows: In the epoch $e_p$-th epoch, a quantile level image $\tau_i$ ($i = 1, 2, \cdots, N$) corresponding to the MP temperature image $T_{MP}^{i}$ is created by the method in section III-A. Then, the quantile level image $\tau_i$ and the MP temperature image $T_{MP}^{i}$ constitute the training dataset $\{(T_{MP}^{i}, \tau_i)_{ep} | i = 1, 2, \cdots, N\}$. Set the appropriate batch size, and the training dataset is divided into $N_{batch}$ batches. This paper chooses the Adam algorithm [23] to update the parameters $\theta$. In summary, the pseudo code of the model Monte Carlo training algorithm is shown in Algorithm 1.

Algorithm 1: Model Monte Carlo Training Algorithm.

Input:
- MP temperature images $\{T_{MP}^{i} | i = 1, 2, \cdots, N\}$.

Output:
- Trained model $\mathcal{MQNN}(T_{MP}, \tau; \theta)$.

1. Initialize the physics-informed Deep MC-QR model $\mathcal{MQNN}(T_{MP}, \tau; \theta)$;
2. for $ep = 1 : ep_{\text{max}}$ do
   3. for $i = 1 : N$ do
      4. Randomly take a sample $\tau_i$ from uniform distribution $U(0, 1)$;
      5. Create a $h \times w$ two-dimensional array $\tau_i$ with all zero elements;
      6. Perform $\tau_i(u, v) = \tau_i$ for $\forall (u, v) \in \Omega_{MP}$;
      7. Composite training data $(T_{MP}^{i}, \tau_i)_{ep}$;
   8. end
9. Dividing $N$ training data $\{(T_{MP}^{i}, \tau_i)\}$ to be $N_{batch}$ batches;
10. for $ba = 1 : N_{batch}$ do
      11. Estimate the reconstructed temperature field $\hat{T}_i = \mathcal{MQNN}(T_{MP}^{i}, \tau_i; \theta)$;
      12. Calculate the proposed loss function $L(\theta)$;
      13. Calculate the gradient of $L(\theta)$;
      14. Update the parameters $\theta$ by the Adam algorithm.
15. end
16. end

C. Model Prediction and Aleatoric Uncertainty Quantification

For a MP temperature image $T_{MP}^{pre}$, the reconstructed temperature field $\hat{T}_{pre}$ and the quantified aleatoric uncertainty $\sigma_{pre}$ are calculated as follows. In the predicted stage, $N_{pre}$ quantile level images are generated by the method in section III-A. Then, the composited prediction data $\{(T_{MP}^{pre}, \tau_A^{pre}) | A = 1, 2, \cdots, N_{pre}\}$ are obtained for the MP temperature image $T_{MP}^{pre}$. For $A = 1, 2, \cdots, N_{pre}$, the temperature field $T_{pre}^{A}$ is predicted by the trained model $\mathcal{MQNN}(T_{MP}, \tau; \theta)$. Thereby, the reconstructed temperature field’s approximation $\hat{T}_{pre}$ and the aleatoric uncertainty $\sigma_{pre}$ are estimated by calculating the mean and the standard deviation of $N_{pre}$ results $\hat{T}_{pre}^{A}$ ($A = 1, 2, \cdots, N_{pre}$), respectively, i.e.,

$$T_{pre} = \frac{1}{N_{pre}} \sum_{A=1}^{N_{pre}} \hat{T}_{pre}^{A},$$  \hspace{1cm} (16)

$$\sigma_{pre} = \left[ \frac{1}{N_{pre}} \sum_{A=1}^{N_{pre}} (\hat{T}_{pre}^{A} - \left(\hat{T}_{pre}\right)^2) \right]^\frac{1}{2}. \quad (17)$$

In summary, the pseudo code of the model prediction and aleatoric uncertainty quantification algorithm is shown in Algorithm 2.

Algorithm 2: Model prediction and aleatoric uncertainty quantification algorithm.

Input:
1. MP temperature image $T_{MP}^{pre}$,
2. Sampling number $N_{pre}$ of quantile levels;
3. Trained model $\mathcal{MQNN}(T_{MP}, \tau; \theta)$.

Output:
1. Reconstructed temperature field $\hat{T}_{pre}$;
2. Quantified aleatoric uncertainty $\sigma_{pre}$.

1. for $A = 1 : N_{pre}$ do
   2. Randomly take a sample $\tau_{pre}^{A}$ from uniform distribution $U(0, 1)$;
   3. Create a $h \times w$ two-dimensional array $\tau_{pre}^{A}$ with all zero elements;
   4. Perform $\tau_{pre}^{A}(u, v) = \tau_{pre}^{A}$ for $\forall (u, v) \in \Omega_{MP}$;
   5. Composite prediction data $(T_{MP}, \tau_{pre}^{A})$;
   6. Predict the reconstructed temperature field $\hat{T}_{pre}^{A} = \mathcal{MQNN}(T_{pre}^{A}, \tau_{pre}^{A}; \theta)$.
8. Calculate the mean $\hat{T}_{pre}$ and the standard deviation $\sigma_{pre}$ of $\{\hat{T}_{pre}^{A} | A = 1, 2, \cdots, N_{pre}\}$.

IV. EXPERIMENT

A. Experimental Setup

As shown in Fig. 5, this paper uses a system ($W = H = 0.2$ m) to validate the effectiveness of the proposed Deep MC-QR method. This system has 55 heat sources, and the heat sink with width $\delta = 0.02$ m is in the center of the left boundary of the system. The positions of monitoring points are shown in Fig. 6. In this experiment, the power of each heat source obeys the normal distribution with a mean of 20000 W and a standard deviation of 1000 W. Besides, the constant temperature $T_0$ is 298 K. The parameters $h$ and $w$ are set to be 200, i.e., $h = w = 200$. 15000 MP temperature images $\{T_{MP}^{1}, T_{MP}^{2}, \cdots, T_{MP}^{15000}\}$ are generated.
In this experiment, five kinds of noises are respectively added into temperature values of 15000 MP temperature images \( \{T_{MP}^{1}, T_{MP}^{2}, \cdots, T_{MP}^{15000}\} \) to generate five types of the dataset as follows:

- Gaussian noise \( \epsilon_1 \sim N(0, 0.3^2) \) for green monitoring points in Fig.6.
- Gaussian noise \( \epsilon_2 \sim N(0, 0.5^2) \) for green monitoring points in Fig.6.
- Uniform noise \( \epsilon_3 \sim U(-0.3, 0.3) \) for green monitoring points in Fig.6.
- Uniform noise \( \epsilon_4 \sim U(-1, 1) \) for green monitoring points in Fig.6.
- Gaussian noise \( \epsilon_5 \sim N(0, 0.3^2) \) for monitoring points to the right of the blue dotted line in Fig.6 (Excluding the five monitoring points on the right boundary).

Based on the above five kinds of the dataset, five trained models are denoted as MQNN\(_{\epsilon_1}\), MQNN\(_{\epsilon_2}\), MQNN\(_{\epsilon_3}\), MQNN\(_{\epsilon_4}\), and MQNN\(_{\epsilon_5}\), respectively. Besides, four evaluation criteria for model accuracy are as follows: a) the average of root mean square error (RMSE)

\[
\text{RMSE} = \frac{1}{6e5} \sum_{t=1}^{3000} \left[ \sum_{(x,y) \in \Omega} \left| T_t(x,y) - \hat{T}_t(x,y) \right|^2 \right]^{1/2}, \quad (18)
\]

b) the average of mean absolute error (MAE)

\[
\text{MAE} = \frac{1}{1.2e8} \sum_{t=1}^{3000} \sum_{(x,y) \in \Omega} \left| T_t(x,y) - \hat{T}_t(x,y) \right|, \quad (19)
\]

c) the average of mean relative error (MRE)

\[
\text{MRE} = \frac{1}{1.2e8} \sum_{t=1}^{3000} \sum_{(x,y) \in \Omega} \left| \frac{T_t(x,y) - \hat{T}_t(x,y)}{T_t(x,y)} \right|, \quad (20)
\]

d) the average of R square \( R^2 \)

\[
R^2 = \frac{1}{3e3} \sum_{t=1}^{3000} \left[ 1 - \frac{\sum_{(x,y) \in \Omega} \left| T_t(x,y) - \hat{T}_t(x,y) \right|^2}{\sum_{(x,y) \in \Omega} \left| T_t(x,y) - \bar{T}_t \right|^2} \right], \quad (21)
\]

For the first three evaluation criteria RMSE, MAE and MRE, the closer their values are to 0, the higher the accuracy of the trained model \( M(T_{MP}, \tau; \theta) \). \( R^2 \to 1 \) means that the trained model \( M(T_{MP}, \tau; \theta) \) is accurate.

### B. Model accuracy analysis

The prediction accuracy of five models MQNN\(_{\epsilon_1}\), MQNN\(_{\epsilon_2}\), MQNN\(_{\epsilon_3}\), MQNN\(_{\epsilon_4}\), and MQNN\(_{\epsilon_5}\) are shown in TABLE I. Apparently, three kinds of average errors, i.e., RMSE, MAE and MRE, are sufficiently small for five models. The \( R^2 \) of five models are greater than or equal to 0.99995. Therefore, the proposed Deep MC-QR method can reconstruct the temperature fields accurately.

| Noise | Model    | RMSE     | MAE      | MRE      | \( R^2 \) |
|-------|----------|----------|----------|----------|----------|
| \( \epsilon_1 \) | MQNN\(_{\epsilon_1}\) | 0.017370 | 0.014912 | 0.001013 | 0.99996  |
| \( \epsilon_2 \) | MQNN\(_{\epsilon_2}\) | 0.079592 | 0.050666 | 0.000114 | 0.99996  |
| \( \epsilon_3 \) | MQNN\(_{\epsilon_3}\) | 0.070904 | 0.044899 | 0.000122 | 0.99996  |
| \( \epsilon_4 \) | MQNN\(_{\epsilon_4}\) | 0.081173 | 0.052624 | 0.000143 | 0.99995  |
| \( \epsilon_5 \) | MQNN\(_{\epsilon_5}\) | 0.089211 | 0.060564 | 0.000114 | 0.99995  |

This experiment randomly chooses a MP temperature image in Fig.7 (adding five kinds of data noises) to show the prediction performance of five models. Fig.8 is the truth temperature field of the MP temperature image in Fig.7. Based on five models MQNN\(_{\epsilon_1}\), MQNN\(_{\epsilon_2}\), MQNN\(_{\epsilon_3}\), MQNN\(_{\epsilon_4}\), and MQNN\(_{\epsilon_5}\), five reconstructed temperature fields are presented in Fig.9, Fig.10, Fig.11, Fig.12, and Fig.13.
respectively. Compared with Fig. 8, five reconstructed temperature fields are basically consistent with the truth temperature field.

For the MP temperature image in Fig. 7, the quantified aleatoric uncertainties by four models $\mathcal{MQNN}_{\varepsilon_1}$, $\mathcal{MQNN}_{\varepsilon_2}$, $\mathcal{MQNN}_{\varepsilon_3}$ and $\mathcal{MQNN}_{\varepsilon_4}$ are shown in Fig. 14, Fig. 15, Fig. 16 and Fig. 17, respectively. According to section IV-A, these four models are trained by the datasets that include the Gaussian noises ($\varepsilon_1 \sim N(0, 0.3^2)$, $\varepsilon_2 \sim N(0, 0.50^2)$) and uniform noise ($\varepsilon_3 \sim U(-0.3, 0.3)$, $\varepsilon_4 \sim U(-1, 1)$) in the green points in Fig. 6. Referring to Fig. 14, Fig. 15, Fig. 16 and Fig. 17, the aleatoric uncertainties of the areas around the green points (Fig. 6) are much larger than the other areas. Besides, the maximum values of two aleatoric uncertainties in Fig. 14 and Fig. 15 are approximately equal to 0.3 and 0.5, respectively. For the models $\mathcal{MQNN}_{\varepsilon_3}$ and $\mathcal{MQNN}_{\varepsilon_4}$ trained by data with the uniform noises ($\varepsilon_3$, $\varepsilon_4$), the quantified aleatoric uncertainties

C. Aleatoric uncertainty quantification results analysis
are approximately equal to the standard deviations 0.173 and 0.577 of two uniform noises, respectively. In summary, the proposed Deep MC-QR method can accurately quantify the aleatoric uncertainty caused by data noise.

Fig. 14. Quantified aleatoric uncertainty by $\mathcal{MQNN}_{\epsilon_1}$.

Fig. 15. Quantified aleatoric uncertainty by $\mathcal{MQNN}_{\epsilon_2}$.

Fig. 16. Quantified aleatoric uncertainty by $\mathcal{MQNN}_{\epsilon_3}$.

D. The effect of aleatoric uncertainty for TFR

For the model $\mathcal{MQNN}_{\epsilon_5}$, it is trained by the data with the Gaussian noise $\epsilon_5 \sim N(0, 0.3^2)$ for monitoring points to the right of the blue dotted line in Fig.6. The fusion image between the monitoring point positions (Fig.6) and the quantified aleatoric uncertainty by $\mathcal{MQNN}_{\epsilon_5}$ is shown in Fig.18.

Referring to Fig.18, the aleatoric uncertainty corresponding to the area to the right of the blue dotted line is much larger than the area to the left of the blue dotted line. Besides, there are four especial phenomena in Fig.18 as follows:

a) **The maximum aleatoric uncertainty (about 0.35) is larger than the actual data noise (0.3).** As shown in Fig.18, the maximum value of quantified aleatoric uncertainty is approximately equal to 0.35, while the value of actual Gaussian noise is equal to 0.3.

b) **There is also aleatoric uncertainty on the non-monitoring point area to the right of the blue dotted line.** Compared with the non-monitoring point area to the left of the blue dotted line, the aleatoric uncertainty of the non-monitoring point area to the right of the blue dotted line is relatively large as shown in Fig.18. However, the non-monitoring point area to the left of the blue dotted line does not contain data noise.

c) **The aleatoric uncertainty in the non-monitoring point area to the right of the blue dotted line has large differences.** As shown in Fig.18, the aleatoric uncertainty in the green dashed ellipse is larger than in the red dashed ellipse for the non-monitoring point area. In fact, both non-monitoring point areas do not include data noise.

d) **The uncertainty around five monitoring points (in the orange circle) on the right boundary is small.**

In fact, the spread of data noise results in the above four special phenomena. Firstly, The data noise propagation leads to the accumulation of aleatoric uncertainty. Thereby,
the maximum aleatoric uncertainty is larger than the actual Gaussian noise (phenomenon a)). Secondly, the aleatoric uncertainty is transferred to the surrounding area due to noise propagation, which brings about phenomenon b). Thirdly, the data noise may include positive or negative Gaussian noise. Thus, the accumulation of positive or negative Gaussian noise will lead to the phenomenon c). According to section IV-A, five monitoring points (in the orange circle) on the right boundary do not contain data noise. Therefore, the phenomenon d) appears in Fig. 18.

In summary, data noise causes the reconstructed temperature field to have aleatoric uncertainties at and around the monitoring point. Therefore, aleatoric uncertainty quantification is important to ensure the credibility of the reconstructed temperature field.

V. CONCLUSIONS

This paper proposes the Deep MC-QR method to reconstruct the temperature field and quantify the aleatoric uncertainty caused by data noise. By physical knowledge to guide the training of CNN, the proposed Deep MC-QR method can reconstruct an accurate TFR surrogate model without using any labeled training data. Besides, the proposed Deep MC-QR method can quantify aleatoric uncertainty by constructing a quantile level image for each MP temperature image. This paper adds five kinds of data noise in the MP temperature image. The results show that the proposed Deep MC-QR method can accurately reconstruct the temperature field and quantify the aleatoric uncertainty precisely. Especially, the results also show that data noise causes the reconstructed temperature field to have aleatoric uncertainties at and around the monitoring point. Thus, aleatoric uncertainty quantification is necessary for the TFR problem.

ACKNOWLEDGMENT

This work was supported by the Postgraduate Scientific Research Innovation Project of Hunan Province (No.CX20200006) and the National Natural Science Foundation of China (Nos.11725211).

REFERENCES

[1] W. Yao, X. Chen, W. Luo, M. van Tooren, and J. Guo, “Review of uncertainty-based multidisciplinary design optimization methods for aerospace vehicles,” Progress in Aerospace Sciences, vol. 47, no. 6, pp. 450–479, 2011.
[2] X. Zheng, W. Yao, Y. Xu, and X. Chen, “Improved compression inference algorithm for reliability analysis of complex multistate satellite system based on multilevel bayesian network,” Reliability Engineering & System Safety, vol. 189, pp. 123–142, 2019.
[3] X. Zheng, W. Yao, Y. Xu, and X. Chen, “Algorithms for bayesian network modeling and reliability inference of complex multistate systems: Part i – independent systems,” Reliability Engineering & System Safety, vol. 202, 2020.
[4] W. Zhang, X. Feng, F. Goerlantx, and Q. Liu, “Towards a convolutional neural network model for classifying regional ship collision risk levels for waterway risk analysis,” Reliability Engineering & System Safety, vol. 204, p. 107127, 2020.
[5] X. Zhao, Z. Gong, J. Zhang, W. Yao, and X. Chen, “A surrogate model with data augmentation and deep transfer learning for temperature field prediction of heat source layout,” Structural and Multidisciplinary Optimization, 2021.
[6] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in Proceedings of Medical Image Computing and Computer-Assisted Intervention. Springer International Publishing, 2015, Book Section, pp. 234–241.
[7] S. Chen, H. Lu, M. Sadoughi, C. Hu, V. Nemani, A. Thelen, K. Webster, M. Darr, J. Sidon, and S. Kenny, “A physics-informed deep learning approach for bearing fault detection,” Engineering Applications of Artificial Intelligence, vol. 103, 2021.
[8] Z. Fang, “A high-efficient hybrid physics-informed neural networks based on convolutional neural network,” IEEE Transactions on Neural Networks and Learning System, vol. 99, pp. 1–13, 2021.
[9] H. Gao, L. Sun, and J.-X. Wang, “Phygeonet: Physics-informed geometry-adaptive convolutional neural networks for solving parameterized steady-state pdes on irregular domain,” Journal of Computational Physics, vol. 428, p. 110079, 2021.
[10] H. Gao, L. N. Sun, and J. X. Wang, “Super-resolution and denoising of fluid flow using physics-informed convolutional neural networks without high-resolution labels,” PHYSICS OF FLUIDS, vol. 33, no. 7, 2021.
[11] Z. Zhang, Y. Li, W. Zhou, X. Chen, W. Yao, and Y. Zhao, “Tonr: An exploration for a novel way combining neural network with topology optimization,” Computer Methods in Applied Mechanics and Engineering, vol. 386, p. 114083, 2021.
[12] Z. Gong, W. Zhou, J. Zhang, W. Peng, and W. Yao, “Physics-informed deep reversible regression model for temperature field reconstruction of heat-source systems,” 2021. [Online]. Available: arXiv:2106.11929v2
[13] J. Postels, F. Ferroni, H. Coshun, N. Navab, and F. Tombari, “Sampling-free epistemic uncertainty estimation using approximated variance propagation,” in Proceedings of International Conference on Computer Vision, 2019, Conference Proceedings.
[14] Y. Gal, “Uncertainty in deep learning,” Thesis, 2016.
[15] A. Kendall and Y. Gal, “What uncertainties do we need in bayesian deep learning for computer vision?” in Proceedings of the 31st Conference on Neural Information Processing Systems, Conference Proceedings.
[16] B. Lakshminarayanan, A. Pritzel, and C. Blundell, “Simple and scalable predictive uncertainty estimation using deep ensembles,” in 31st Conference on Neural Information Processing Systems, Conference Proceedings.
[17] J. Z. Liu, J. Paisley, M.-A. Kioumourtzoglou, and B. A. Coull, “Accurate uncertainty estimation and decomposition in ensemble learning,” in Proceedings of the 33rd Conference on Neural Information Processing Systems, Conference Proceedings.
[18] N. Tagasovska and D. Lopez-Paz, “Single-model uncertainties for deep learning,” in Proceedings of the 33rd Conference on Neural Information Processing Systems, 2019, Conference Proceedings.
[19] A. Malinin, B. Mlodozeniec, and M. Gales, “Ensemble distribution distillation,” in Proceedings of International Conference on Learning Representations, 2020, Conference Proceedings.
[20] R. Koenker, “Quantile regression for longitudinal data,” in Quantile Regression Analysis, vol. 91, no. 1, pp. 74–89, 2004.
[21] I. Takeuchi, Q. V. Le, T. D. Sears, and A. J. Smola, “Nonparametric quantile estimation,” Journal of Machine Learning Research, vol. 7, pp. 1231–1264, 2006.
[22] L. I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” Physica D: Nonlinear Phenomena, vol. 60, no. 1, pp. 259–268, 1992.
[23] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in Proceedings of International Conference on Learning Representations, 2015.