Label Free Uncertainty Quantification

Huirui Li*, Jianhua Yin†, Xiaoping Du‡

Indiana University - Purdue University Indianapolis, Indianapolis, IN, 46202, United States

Uncertainty quantification (UQ) is essential in scientific computation since it can provide the estimate of the uncertainty in the model prediction. Intensive computation is required for UQ as it calls the deterministic simulation repeatedly. This study discusses a physics-based label-free deep learning UQ method that does not need predictions at training points or labels. It satisfies the physical equations from which labels could be generated without solving the equations during the training process. Then inexpensive surrogate models are built with respect to model inputs. The surrogate models are used for UQ with a much lower computational cost. Two examples demonstrate that the label-free method can efficiently produce probability distributions of model outputs for given distributions of random input variables.

I. Nomenclature

d	=	number of training points
$f(\cdot)$	=	function whose surrogate model is built
$f^{NN}(\cdot)$	=	surrogate model built from neural network
$h(\cdot)$	=	physical equation
L	=	loss function
m	=	number of output variables
n	=	number of input variables
$\boldsymbol{w}$	=	vector of weights and biases in a neural network
$\boldsymbol{x}$	=	vector of input variables
$\boldsymbol{y}$	=	vector of input variables

II. Introduction

Scientific computation and simulation have increasingly used in engineering analysis and design. The primary role of computational models is to make predictions for both analysis and design. Given a set of conditions or a design, we can predict how a component or system behaves. The computational model, however, is never perfect and exact, and uncertainty is always present. For example, the model input may have random variables, and as a result, the model prediction, the model output, or the response, is also uncertain. It is therefore important to quantify the uncertainty associated with the response predicted, and this is the major task of uncertainty quantification (UQ) [1-3].

Since uncertainty always exists in model parameters and model predictions, UQ is performed to quantify the effect of uncertainty in model input and model error on the model output (prediction). UQ needs to call computational models repeatedly to obtain probability distributions of responses given distributions of model input variables. In general, UQ is computationally expensive, especially when the dimension of model input is high [4, 5]. To this end, regression is commonly used to build a surrogate model to replace the original computational model. Regression can be performed by many methods, such as response surface modeling, Gaussian Process (GP) method, support vector machines (SVM), and neural network [6-8]. Regression starts from generating a set of input samples, getting labels (outputs) by

* Graduate Assistant, Department of Mechanical and Energy Engineering.
† Graduate Assistant, Department of Mechanical and Energy Engineering.
‡ Professor, Department of Mechanical and Energy Engineering.

This is the author's manuscript of the article published in final edited form as:

Li, H., Yin, J., & Du, X. (2022). Label Free Uncertainty Quantification. AIAA SCITECH 2022 Forum. https://doi.org/10.2514/6.2022-1097

calling the computational model at the input samples, and then fitting a surrogate model with the input samples and their labels.

As indicated by many studies, including our own, machine learning is particularly useful for UQ problems encountered in engineering design. Machine learning techniques can significantly reduce the computational burden in the following aspects: Reduce the dimension of uncertain input variables [9] and create accurate but inexpensive surrogate models [10-13], both for higher computational efficiency. Specific machine learning techniques have increasingly used. For example, the GP method for quantifying model structure uncertainty [14, 15]; the SVM method for estimating rare event probabilities [16], and other methods for predicting component and system reliability [17]. Many studies show that add physical constraints can significantly improve prediction accuracy [18, 19]. For instance, the constraints from physics can be added to the loss function of artificial neural networks [18], used as a priori information for deep learning [20], and incorporated into GP [21].

Getting labels is usually costly since it runs the computational model many times. Physics-informed neural network (PINN) is proposed [22, 23] to obtain the solution of partial differential equations (PDEs) without solving the PDEs by classical numerical methods (e.g. the Finite Element Method). The PDEs are served as the so-called physical constraints during the training process of the neural network. This method has gained much attention because it makes the regression task feasible without solving the true label. Besides, the physical constraints prevent the regression from severe overfitting in conventional neural network, especially when labeled data are limited.

Motivated by the label-free deep learning, this study investigates a label-free UQ method so that the computational efficiency can be enhanced. The method works for problems where responses are solved out from a system of equations derived from physical principles, and solving the system of equations is expensive while evaluating them is inexpensive. We call these equations physical equations. The strategy is to use only the input samples to build a surrogate model that satisfies the physical equations without solving them. As a result, there is no need to generate labels. Neural network is used to build the surrogate model.

III. Problem Statement

Computational models are given by

$$y_i = f_i(\boldsymbol{x}), i = 1,2, \dots, m \tag{1}$$

where $\boldsymbol{x}$ is a vector of input variables, and y_i is a response. In many applications. The relationship between the input and output is not explicit, and it is derived from the following system of physical equations:

$$\begin{cases} h_1(\boldsymbol{x}, y_1) = 0 \\ h_2(\boldsymbol{x}, y_2) = 0 \\ \cdots\cdots \\ h_m(\boldsymbol{x}, y_m) = 0 \end{cases} \tag{2}$$

where $\boldsymbol{y} = (y_1, y_2, \dots, y_m)^{\mathrm{T}} = \left(f_i(\boldsymbol{x})\right)_{i=1,m}^{\mathrm{T}}$ is the solution of the system of physical equations. Solving the system of physical equations is usually computationally expensive.

The task of UQ in this study is illustrated in Fig. 1 and is stated as follows:

- Given: Distributions of $\boldsymbol{x}$ and the system of physical equations $h_i(\boldsymbol{x}, y_i) = 0, i = 1,2 \dots m$
- Find: Distributions of $\boldsymbol{y}$

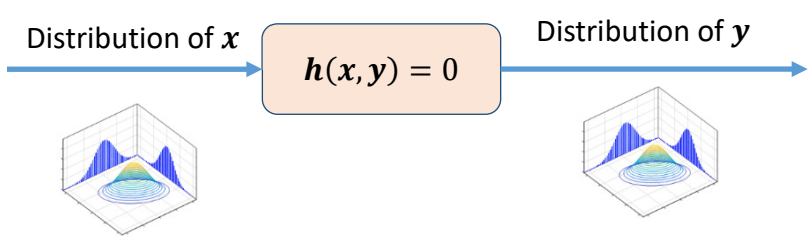

Fig. 1 Task of UQ.

After the joint distribution of $\boldsymbol{y}$ is available, it can be used for reliability analysis, robustness assessment, and many other analysis tasks, as demonstrated in Example 1. It can also be used for design under uncertainty, such as reliability-based design, robust design, and design decision making, as demonstrated in Example 2.

IV. Methodology

Traditional UQ needs to evaluate computational models $y_i = f(\boldsymbol{x})$, where $y_i, i = 1,2,\dots,m$, must be solved out from the system of physical equations in Eq. (2). Solving for y_i may be computationally expensive but evaluating the system of physical equations is much cheaper. The label-free UQ method does not solve the system of physical equations and only evaluates them because no labels are needed. The deep learning neural network method is used. The label-free UQ method consists of two steps.

Step 1 creates surrogate models. The surrogate models are given by

$$y_i = f_i^{NN}(\boldsymbol{x}), \qquad i = 1,2,\dots,m \tag{3}$$

The regression method we use is neural network, which builds surrogate models based on data. In regular neural network regression, a dataset with labels is given by $\{(\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)})\}_{i=1,d}$, where d is the size of the dataset. The model parameters of a neural network, which are to be learned, include weights and biases in all layers. They are denoted by $\boldsymbol{w}$.

The to-be-determined parameters $\boldsymbol{w}$ will be obtained during the model training process. In this work, we do not have labels, and our dataset is $\{(\boldsymbol{x}^{(i)})\}_{i=1,d}$. To this end, we will use the system of physical equations directly as proposed in other physics-based informed deep learning [22]. The strategy is to satisfy the physical equations at the training points $\{(\boldsymbol{x}^{(i)})\}_{i=1,d}$.

$$\begin{cases} h_1(\boldsymbol{x}^{(i)}, \boldsymbol{y}_1^{NN}) = 0 \\ h_2(\boldsymbol{x}^{(i)}, \boldsymbol{y}_2^{NN}) = 0 \\ \cdots\cdots \\ h_m(\boldsymbol{x}^{(i)}, \boldsymbol{y}_m^{NN}) = 0 \end{cases} \tag{4}$$

where

$$\boldsymbol{y}_j^{NN} = f_j^{NN}(\boldsymbol{x}^{(i)}; \boldsymbol{w}), \qquad i = 1,2,\dots,d; j = 1,2,\dots,m \tag{5}$$

To satisfy the physical equations in Eq. (4), we define a loss function

$$L(\boldsymbol{w}) = \sum_{i=1}^{n}\sum_{j=1}^{m} h_j^2\left(\boldsymbol{x}^{(i)}; f_j^{NN}(\boldsymbol{x}^{(i)}; \boldsymbol{w})\right) \tag{6}$$

If $L(\boldsymbol{w}) = 0$, all the physical equations will be satisfied. Minimizing the loss function while training the neural network, we obtain the estimated model parameters $\boldsymbol{w}$; namely

$$\boldsymbol{w}^* = \arg\min L(\boldsymbol{w}) \tag{7}$$

Then surrogate models $y_i = f_j^{NN}(\boldsymbol{x}), i = 1,2,\dots,m,$ are available.

Step 2 performs UQ based on $y_i = \hat{g}_i(\boldsymbol{x})$. Many existing UQ methods can be used. Since the surrogate models are inexpensive, we may use Monte Carlo simulation (MCS) to find the distribution of a response.

The architecture of the neural network and procedure is illustrated in Fig. 2.

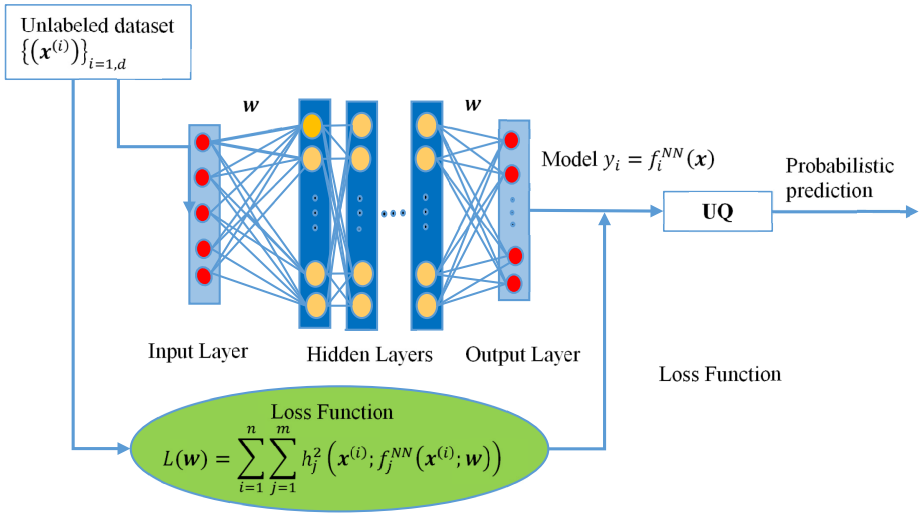

Fig. 2 Architecture of a neural network.

V.Examples

Two examples are used to demonstrate the label-free UQ method. The first example is an analysis problem where the probability distributions of responses are produced. The second example is a design problem where the design requirement is satisfied at a given probability level.

V.A. Example 1: UQ of a relative motion analysis

As shown in Fig. 3, a person rows a boat to cross a river from points A to B with a constant velocity v_M. If the river flows with a velocity v_R, the task is to determine v_M and the angle θ that the boat should direct. It is the angle between the relative velocity of the boat $v_{M/R}$ with respect to the river and the horizonal direction.

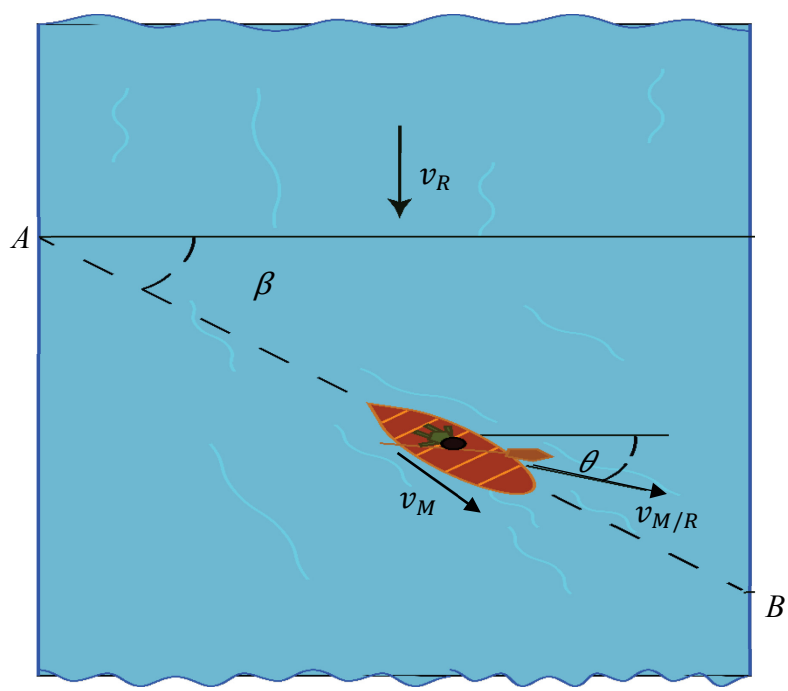

Fig. 3 Relative motion example.

Using the relative motion analysis, we have

$$\boldsymbol{v_M} = \boldsymbol{v_R} + \boldsymbol{v_{M/R}} \tag{8}$$

The angle between line AB and the horizontal direction is denoted by β. The two component equations from Eq. (8) are given by

$$\begin{cases} v_M cos\beta = v_{M/R} cos\theta \\ v_M sin\beta = v_R + v_{M/R} sin\theta \end{cases} \tag{9}$$

The two physical equations are therefore

$$\begin{cases} h_1 = v_m cos\beta - v_{M/R} cos\theta = 0 \\ h_2 = v_m sin\beta - \left(v_R + v_{M/R} sin\theta\right) = 0 \end{cases} \tag{10}$$

The random input variables are $\boldsymbol{x} = (x_1, x_2, x_3)^{\mathrm{T}} = (v_{M/R}, v_R, \beta)^{\mathrm{T}}$, and their distributions are given in Table 1.

Table 1 Distributions associated with input random variables

Random Variable	Distribution	Mean	Standard Deviation
$v_{M/R}$ (m/s)	Normal	5.0	0.5
v_R (m/s)	Normal	2.0	0.2
β (deg)	Normal	45.0	4.5

The first step is to generate models $v_M = f_1^{NN}\left(v_{M/R}, v_R, \beta\right)$, and $\theta = f_2^{NN}\left(v_{M/R}, v_R, \beta\right)$. $d = 200$ training points are used. The neural network includes six layers with four hidden layers, one input layer, and one customer regression output layer. Each hidden layer has 20 neutrons. The activation function is a hyperbolic tangent function. The learning rate is 0.01, and the optimizer is Adam.

A scatter plot of the predictions and true labels of the two responses at the training points is given in Fig. 4. The true labels are obtained by solved the two physical equations in Eq. (10). Note that in real applications, the true labels are not available. The scatter plot is for only a demonstration purpose. The scatter plot indicates good accuracy of the surrogate models.

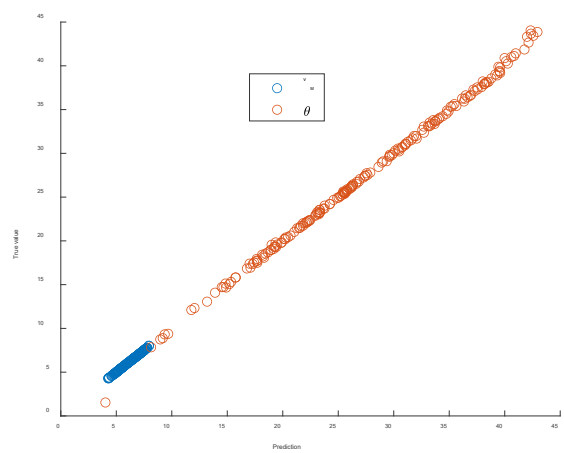

Fig. 4 Scatter plot of predictions and true labels.

After the two models are built, MCS is used for UQ. The sample size is 10^6. The PDFs of the two responses are provided in Figs. 5 and 6. The distribution parameters, means and standard deviations, of the two responses, are given in Tables 2 and 3. For a verification purpose, we also provide the accurate results from true responses and MCS with a sample size 10^6. The results show the good accuracy of the label-free UQ.

Table 2 Statistics of v_M

Method	Mean of v_M (m/s)	Standard deviation of v_M (m/s)
Label free method	6.1988	0.5505
True function	6.2011	0.5519

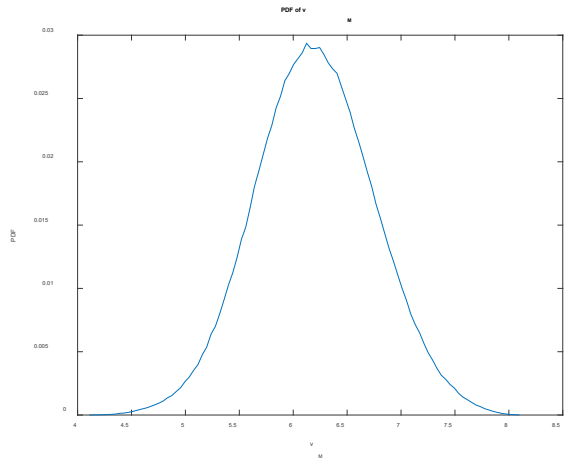

Fig. 5 PDF of v_m.

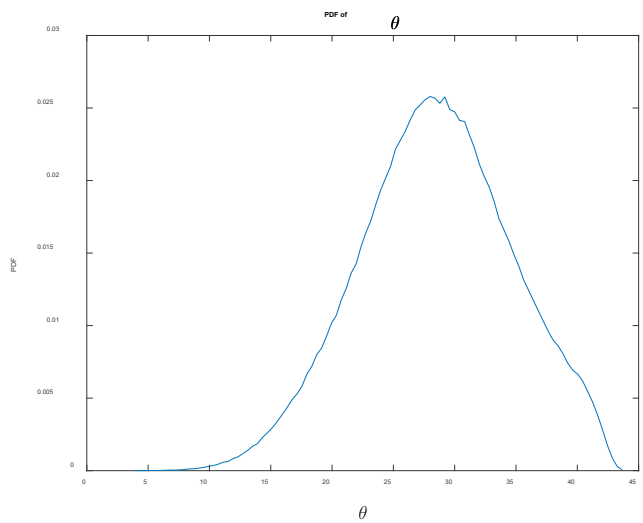

Fig. 6 PDF of $\boldsymbol{\theta}$.

Table 3 Statistics of $\boldsymbol{\theta}$

Method	Mean of θ (dge)	Standard deviation θ (deg)
Label free method	28.4511	6.2185
True function	28.4329	6.3401

V.B. Example 2: Cantilever shaft design under uncertainty

The design task is to design a cantilever shaft [24, 25] as shown in Fig. 7 so that it can withstand random forces F and P; and a random torque T, with the reliability greater than or equal to $[R] = 0.999$. The random variables are $\boldsymbol{x} = \left(S_y, P, F, T\right)^{\mathrm{T}}$, where S_y is the yield strength of the material. All the random variables are independent, and their distributions are given in Table 5. L is the length of the tube, and $L = 0.15$ m, The design variables are $\boldsymbol{d} = (d_0, t)^{\mathrm{T}}$, which should be chosen from the following list of preferred sizes for $d_0 \times t$ (mm): 12×2, 16×2, 16×3, 20×4, 24×4, 25×5, 30×4, 30×5, 42×4, 42×5, 50×4, 50×5. The distributions of random variables are given in Table 4.

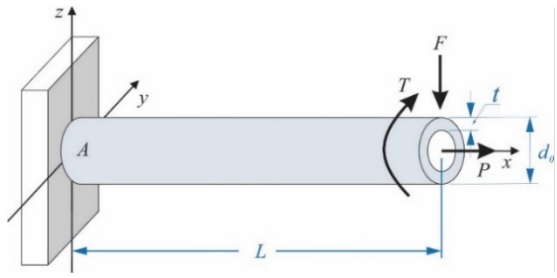

Fig. 7 A cantilever tube.

Table 4 Distributions of random variables in Example 2

Random Variable	Distribution	Mean	Standard Deviation
S_y (MPa)	Normal	250	10
P (N)	Normal	8000	2000
F (N)	Lognormal	1500	50
T (N·m)	Normal	75	4

The design requirement is that the maximum stress should be less than the yield strength, and the design margin equation is derived from the strength theory, which provides a physical equation at the limit state.

$$h(\boldsymbol{x}) = S_y - \sqrt{\sigma_x^2 + 3\tau_{zx}^2} \tag{11}$$

where $\boldsymbol{x} = \left(S_y, P, F, T, t\right)^{\mathrm{T}}$.

$$\sigma_x = \frac{P}{\frac{\pi(d_0^2 - (d_0 - 2t)^2)}{4}} + \frac{FL\left(\frac{d_0}{2}\right)}{\frac{\pi(d_0^4 - (d_0 - 2t)^4)}{64}} \tag{12}$$

$$\tau_{zx} = \frac{T\left(\frac{d_0}{2}\right)}{\frac{\pi(d_0^4 - (d_0 - 2t)^4)}{32}} \tag{13}$$

The diameter d_0 is the response; namely, $y = d_0$. The input variables are $\boldsymbol{x} = (x_1, x_2, x_3, x_4, x_5)^{\mathrm{T}} = \left(S_y, P, F, T, t\right)^{\mathrm{T}}$. Equation 11 is rewritten as

$$h(\boldsymbol{x}) = x_1 - \sqrt{\sigma_x^2 + 3\tau_{zx}^2} \tag{14}$$

where

$$\sigma_x = \frac{x_2}{\frac{\pi(y^2 - (y - 2x_5)^2)}{4}} + \frac{x_3 L\left(\frac{y}{2}\right)}{\frac{\pi(y^4 - (y - 2x_5)^4)}{64}} \tag{15}$$

$$\tau_{zx} = \frac{x_4\left(\frac{y}{2}\right)}{\frac{\pi(y^4 - (y - 2x_5)^4)}{32}} \tag{16}$$

We generate $d = 200$ training points $\left\{\left(\boldsymbol{x}^{(i)}\right)\right\}_{i=1,200}$. We then use the label-free UQ method to build a surrogate model $y = f^{NN}(\boldsymbol{x})$, or $d_0 = f^{NN}(S_y, P, F, T, t)$. The neural network contains three hidden layer, one feature input

layer, and one custom regression output layer. Each hidden layer has 20 neutrons. The activation function is a hyperbolic tangent function. The optimizer is Adam.

To verify the accuracy, we generate true labels at training points $\{(\boldsymbol{x}^{(i)})\}_{i=1,200}$. The scatter plot of the model output is provided in Fig. 8. In the range of $d_0 \in [30, 60]$ mm, the model produces highly accurate predictions. The prediction of the reliability of a design is also accurate if the design falls into this range.

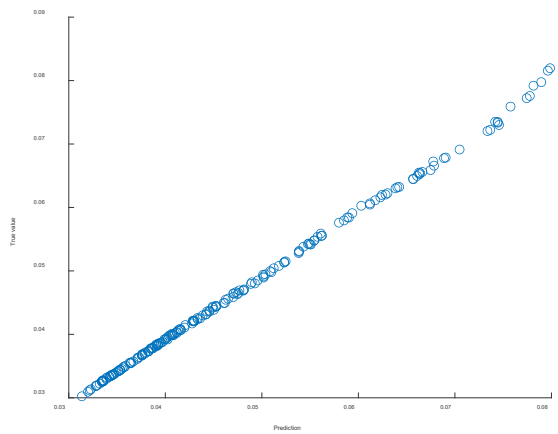

Fig. 8 Scatter plot of predictions and true labels.

Since the surrogate model is efficient, we use Monte Carlo simulation (MCS) to conduct several analyses, from where we can find the best design solution. We select a value of t from the list of the preferred design variables and perform MCS using the distributions given in Table 4. This yields a distribution of d_0, which is the minimum d_0, denoted by d_0^{min}, for the limit state. Since the limit state is determined by random variables, d_0^{min} is also a random variable.

How do we choose a deterministic design variable d_0 for the required reliability $[R]$? We use the percentile value to find d_0.

$$\Pr\left(d_0^{min} < d_0\right) = [R] \tag{17}$$

We now obtain a pair of design variables $(d_0, t)^{\mathrm{T}}$ for a given value of t. We then repeat this process until all values of t are used. The results are given in the first two columns of Table 5. Using the list of referred design variables, the candidate designs are determined, and they are 42×5, 50×4, and 50×5. Considering a smaller size of the shaft, we select the final design to be 42×5. To verify the reliability, we calculate the actual reliability at the calculated diameters with the true function in Eq. (17) by MCS. The sample size of MCS is 10^7. The actual reliability is higher than and close to the required one. Since the final design 42×5 is larger than the calculated design 38.36×5, the final reliability will also be larger than the required reliability. The PDF of the required d_0 obtained from the label-free UQ is given in Fig. 9. Note that in real applications, labels are not available, and MCS with real labels is not feasible, Using MCS herein is for only a demonstration purpose.

Table 5 Candidate Designs

t (mm)	d_0 (mm) Calculated	Candidate design $d_0 \times t$ (mm)	Verified reliability at calculated d_0
2.0	72.39	None	0.9998
3.0	52.42	None	0.9995
4.0	43.37	50×4	0.9997
5.0	38.38	42×5, 50×5	1.0

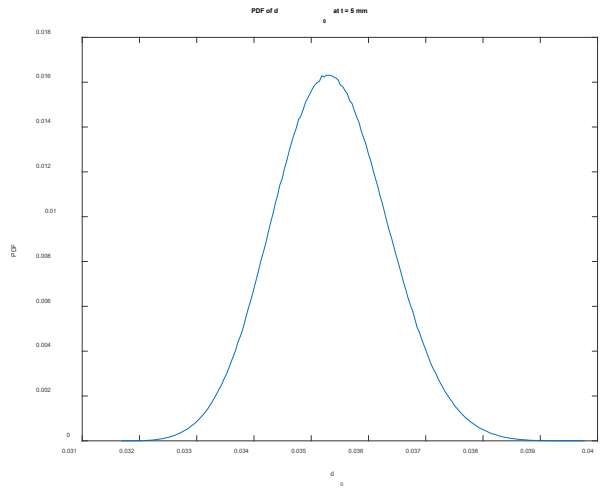

Fig. 9 PDF of required d_0.

In this example, there are two responses, but with only one physical equation. This is the case where the dimension of outputs is larger than the number of physical equations. This example indicates that this case can also be handled by the proposed method. This example also demonstrates that the label-free UQ method can be used for design. The model built from the label-free method may be used for other tasks, such as optimization under uncertainty and robust design.

VI. Conclusions

This study demonstrates the feasibility of label-free uncertainty quantification using neural network regression. A surrogate model is built without labels. The training points only contain a dataset of input variables at which a set of physical equations are satisfied during the learning process. The satisfaction is achieved by employing a loss function which is defined using a set of physical equations. The two examples demonstrate the potential applications of the method. The future work will be the extension to more general applications, where time- and space-dependent responses and models are involved. The other possible research direction is to integrate the regression and uncertainty quantification so that the quantified uncertainty is part of the regression output.

Acknowledgments

The support from the National Science Foundation under Grant No 1923799 is acknowledged.

References

[1] Wojtkiewicz, S., Eldred, M., Field, J., R, Urbina, A., and Red-Horse, J. "Uncertainty quantification in large computational engineering models," 19th AIAA Applied Aerodynamics Conference. 2001, p. 1455.

[2] Roy, C., and Oberkampf, W. "A complete framework for verification, validation, and uncertainty quantification in scientific computing," 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition. 2010, p. 124.

[3] Sankaran, S., and Marsden, A. L. "A stochastic collocation method for uncertainty quantification and propagation in cardiovascular simulations," 2011.

[4] Yin, J., and Du, X. "Active learning with generalized sliced inverse regression for high-dimensional reliability analysis," Structural Safety Vol. 94, 2022, p. 102151.
doi: https://doi.org/10.1016/j.strusafe.2021.102151

[5] Yin, J., and Du, X. "High-Dimensional Reliability Method Accounting for Important and Unimportant Input Variables," Journal of Mechanical Design, 2021, pp. 1-28.
doi: 10.1115/1.4051982

[6] Echard, B., Gayton, N., and Lemaire, M. "AK-MCS: an active learning reliability method combining Kriging and Monte Carlo simulation," Structural Safety Vol. 33, No. 2, 2011, pp. 145-154.

[7] Jiang, P., Missoum, S., and Chen, Z. "Optimal SVM parameter selection for non-separable and unbalanced datasets," Structural and Multidisciplinary Optimization Vol. 50, No. 4, 2014, pp. 523-535.

[8] Li, M., and Wang, Z. "Deep learning for high-dimensional reliability analysis," Mechanical Systems and Signal Processing Vol. 139, 2020, p. 106399.

[9] Gadd, C., Xing, W., Nezhad, M. M., and Shah, A. "A surrogate modelling approach based on nonlinear dimension reduction for uncertainty quantification in groundwater flow models," Transport in porous media Vol. 126, No. 1, 2019, pp. 39-77.
[10] Du, X., Leifsson, L., Meeker, W., Gurrala, P., Song, J., and Roberts, R. "Efficient Model-Assisted Probability of Detection and Sensitivity Analysis for Ultrasonic Testing Simulations using Stochastic Metamodeling," Journal of Nondestructive Evaluation, Diagnostics and Prognostics of Engineering Systems, 2019, pp. 1-18.
[11] Zhang, Y., Kim, N. H., and Haftka, R. T. "General surrogate adaptive sampling using interquartile range for design space exploration," AIAA Scitech 2019 Forum. 2019, p. 2213.
[12] Deshmukh, A. P., and Allison, J. T. "Design of dynamic systems using surrogate models of derivative functions," Journal of Mechanical Design Vol. 139, No. 10, 2017, p. 101402.
[13] Qian, Z., Seepersad, C. C., Joseph, V. R., Allen, J. K., and Jeff Wu, C. F. "Building Surrogate Models Based on Detailed and Approximate Simulations," Journal of Mechanical Design Vol. 128, No. 4, 2005, pp. 668-677.
doi: 10.1115/1.2179459
[14] Jiang, Z., Li, W., Apley, D. W., and Chen, W. "A Spatial-Random-Process Based Multidisciplinary System Uncertainty Propagation Approach With Model Uncertainty," Journal of Mechanical Design Vol. 137, No. 10, 2015.
doi: 10.1115/1.4031096
[15] Xi, Z. "Model-Based Reliability Analysis With Both Model Uncertainty and Parameter Uncertainty," Journal of Mechanical Design Vol. 141, No. 5, 2019.
doi: 10.1115/1.4041946
[16] Hu, Z., and Du, X. "Integration of Statistics-and Physics-Based Methods—A Feasibility Study on Accurate System Reliability Prediction," Journal of Mechanical Design Vol. 140, No. 7, 2018, p. 074501.
[17] Stern, R. E., Song, J., and Work, D. B. "Accelerated Monte Carlo system reliability analysis through machine-learning-based surrogate models of network connectivity," Reliability Engineering & System Safety Vol. 164, 2017, pp. 1-9.
doi: https://doi.org/10.1016/j.ress.2017.01.021
[18] Zhao, W. L., Gentine, P., Reichstein, M., Zhang, Y., Zhou, S., Wen, Y., Lin, C., Li, X., and Qiu, G. Y. "Physics - constrained machine learning of evapotranspiration," Geophysical Research Letters Vol. 46, No. 24, 2019, pp. 14496-14507.
[19] Liu, D., and Wang, Y. "Multi-Fidelity Physics-Constrained Neural Network and Its Application in Materials Modeling," Journal of Mechanical Design, 2019, pp. 1-35.
doi: 10.1115/1.4044400
[20] Chen, Y., and Zhang, D. "Physics-constrained deep learning of geomechanical logs," IEEE Transactions on Geoscience and Remote Sensing, 2020.
[21] Wang, Z., Xing, W., Kirby, R., and Zhe, S. "Physics Regularized Gaussian Processes," arXiv preprint arXiv:2006.04976, 2020.
[22] Raissi, M., Perdikaris, P., and Karniadakis, G. E. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," Journal of Computational Physics Vol. 378, 2019, pp. 686-707.
doi: https://doi.org/10.1016/j.jcp.2018.10.045
[23] Karniadakis, G. E., Kevrekidis, I. G., Lu, L., Perdikaris, P., Wang, S., and Yang, L. "Physics-informed machine learning," Nature Reviews Physics Vol. 3, No. 6, 2021, pp. 422-440.
[24] Li, H., and Du, X. "Recovering Missing Component Dependence for System Reliability Prediction via Synergy Between Physics and Data," Journal of Mechanical Design Vol. 144, No. 4, 2021.
doi: 10.1115/1.4052624
[25] Yin, J., and Du, X. "A Practical Safety Factor Method for Reliability-Based Component Design," International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Vol. 84010, American Society of Mechanical Engineers, 2020, p. V11BT11A037.