



A Transendothelial Leukocyte Transmigration Model Based on Computational Fluid Dynamics and BP Neural Network

Qingjia Chi¹, Zichang Yang¹ and Hua-Ping Liang^{2*}

¹Department of Engineering Structure and Mechanics, School of Science, Wuhan University of Technology, Wuhan, China, ²State Key Laboratory of Trauma, Burns and Combined Injury, Department of Wound Infection and Drug, Daping Hospital, Army Medical University, Chongqing, China

OPEN ACCESS

Edited by:

Gongfa Li,
Wuhan University of Science and
Technology, China

Reviewed by:

Jing Xie,
Sichuan University, China
Ahmet Fatih Tabak,
Kadir Has University, Turkey
Jiangan Xie,
Chongqing University of Posts and
Telecommunications, China

*Correspondence:

Hua-Ping Liang
13638356728@163.com

Specialty section:

This article was submitted to
Bionics and Biomimetics,
a section of the journal
Frontiers in Bioengineering and
Biotechnology

Received: 23 February 2022

Accepted: 04 April 2022

Published: 21 June 2022

Citation:

Chi Q, Yang Z and Liang H-P (2022) A
Transendothelial Leukocyte
Transmigration Model Based on
Computational Fluid Dynamics and BP
Neural Network.
Front. Bioeng. Biotechnol. 10:881797.
doi: 10.3389/fbioe.2022.881797

The mechanism of immune infiltration involving immune cells is closely related to various diseases. A key issue in immune infiltration is the transendothelial transmigration of leukocytes. Previous studies have primarily interpreted the leukocyte infiltration from a biomedical perspective. The physical mechanism of leukocyte infiltration remains to be explored. By integrating the immune cell transmigration computational fluid dynamics (CFD) data, the paper builds a time-dependent leukocyte transmigration prediction model based on the bio-inspired methods, namely back propagation neural networks (BPNN) model. The model can efficiently predict the immune cell transmigration in a special microvascular environment, and obtain good prediction accuracy. The model accurately predicted the cell movement and flow field changes during the transmigration. In the test data set, it has high prediction accuracy for cell deformation, motion velocity and flow lift forces during downstream motion, and maintains a good prediction accuracy for drag force. The two prediction models achieved the prediction of leukocyte transmigration in a specific microvascular environment and maintained a high prediction accuracy, indicating the feasibility and robustness of the BPNN model applied to the prediction of immune cell infiltration. Compared with traditional CFD simulations, BPNN models avoid complex and time-dependent physical modeling and computational processes.

Keywords: random forest, back propagation neural network, cox regression, prognostic prediction, immune infiltration

INTRODUCTION

The mechanism of immune infiltration involving immune cells is closely related to various diseases. A key issue in immune infiltration is the transendothelial transmigration of leukocytes (immune cells). Among them, numerous microvascular stenosis fragments smaller than the cell size and tiny pores on the vessel wall are the key factors hindering the infiltration of immune cells. Previous studies have primarily interpreted the infiltration of immune cells from a biomedical perspective, ignoring the physical role in immune regulation. This paper combines various machine learning algorithms and computational fluid dynamics (CFD) simulation methods to systematically analyze the immune cell infiltration mechanism associated with liver cancer from the perspectives of biomedicine and mechanics.

In recent years, physics-based modeling approaches have become widely used analytical tools in engineering and environmental systems (Willard et al., 2022). However, the deviation between the actual engineering process and the physical laws adopted in the physical model makes us need to approximate the real value, leading to errors (Yan et al., 2020). In addition, the physical model contains many parameters estimated from limited observational data, which further degrades the performance of the physical model (Günther et al., 2020). Machine learning (ML) methods based on neural network models automatically extract complex relationships from data, theoretically solving scientific problems in physical models (Bentley, 2014; Kelley et al., 2016; Biswas et al., 2018; Huang et al., 2021; Jiang et al., 2021). The introduction of ML and deep learning methods achieved a leap from “model-driven” to “data-driven” (Xue et al., 2015; Jiang et al., 2021; Liu et al., 2022). This saves us from knowing the specific equations between the data, and the algorithm will automatically derive the data to match the target value. In many respects, the prediction accuracy of this method far exceeds that of traditional modeling methods due to the coverage of the vast training dataset (Valsamis et al., 2017; Sun et al., 2022; Wu et al., 2022; Yun and). But even state-of-the-art neural network models can be non-generalizable in specific studies due to the large-scale demand for data (Karpatne et al., 2017; Zhao et al., 2022). Therefore, researchers began to explore combining physical models and advanced neural network models to achieve their complementary strengths (Ansari et al., 2020).

In this regard, the currently most widely used neural network model (BPNN) plays a critical role (Dai and MacBeth, 1997). The combination method of physical and neural network models to practical problems is mainly divided into four aspects: parameterization (S. and S. et al., 2019), solution of partial differential equations (Yong et al., 2020), derivation of governing equations (Teng et al., 2020) and inverse modeling (Henry and F. et al., 2015). In fluid mechanics, researchers have gradually realized the efficiency and accuracy of predictive models (Wang et al., 2021). The forward calculation of physical problems is to predict the next state of the system by using physical parameters such as temperature, deformation, mass, and spatial position of the system. In contrast, inverse modeling uses the output of the system to derive the physical parameters of the system. In this regard, many high-fidelity datasets describing fluid mechanics phenomena facilitate the application of neural network methods to fluid mechanics (Fei et al., 2020; Yong et al., 2020). In recent years, deep learning methods for biophysical problems have emerged (Almagrabi et al., 2021). This modeling paradigm of exploring the laws of physics using efficient deep learning tools is known as data-driven modeling. These studies include predicting average Navier-Stokes uncertainty regions in high Reynolds (Re) environments (Templeton, 2015) and prediction studies of cylindrical velocity fields with different Re using fused convolutional neural networks (Jin et al., 2018).

Abundant experimental and numerical simulation data can help researchers understand the flow around a circular object. We used machine learning in previous studies to examine the role of immune infiltration in various diseases (Chi et al., 2020; Yang

et al., 2021; Xu et al., 2022). This paper tries to adopt a research method combining ML method and physical model and take immune cell transmigration as the research object. We used CFD simulation data to build a neural network method for immune cell transmigration prediction. This method enables data value to be used repeatedly and achieves second-by-second forecasts that traditional methods cannot reach. Among them, the bio-inspired BPNN model is adopted and the prediction performance of the neural network and its superiority compared to the CFD method are discussed.

MATERIALS AND METHODS

The CFD Model

In the blood flow of the human body, the size of immune cells is generally between 7 and 15 μm (Luo et al., 2011). To simplify the calculations, we use a cell model with a diameter of 10 μm for benchmark simulations. The cell model consists of viscous incompressible cell fluid and linear elastic cell membrane. The cell membrane thickness is 0.4 μm , the elastic modulus is 100 Pa, and the Poisson's ratio is 0.25 (Aubry et al., 2015). Here, we choose the elastic modulus range of 50–500 Pa to reflect the physiologically possible mechanical response of the cell membrane. Cell fluid and plasma were designated as homogeneous fluids with the same parameters, with a kinematic viscosity of 0.0008 Pa s. The density of immune cells and plasma is 1,000 kg/m³. The phenomenon of cells passing through pores generally occurs in the capillary or venule vascular environment, where the blood flow rate is generally controlled at Re between 0.01 and 0.2 (Haber et al., 2013). Therefore, $Re = 0.1$ was adopted as the baseline plasma flow field. To investigate the retention effect of microvascular stenosis on cell transport, we selected six sets of parameters with channel widths of 10, 12, 14, 16, 18, and 20 μm to establish a microvessel model containing stenotic pores (Vollmar et al., 2019).

The immune cell transmigration model established in this paper includes a cell droplet and a microvascular model with narrow pores. The initial position of the immune cell is located on the upper side of the pore, and it will passively deform and pass through the pores smaller than its size under the drive of the fluid. Although a 3D model would be more accurate, we chose a 2D model to simplify the calculations. Studies have shown that the model is insensitive to depth in cells entering narrow pores (Leong et al., 2011). At the same time, an axisymmetric model containing half of the computational domain was adopted to analyze the transmigration of cells. The hemisphere model composed of the blue and cyan areas in the figure is the immune cell model, composed of a uniform cytosol wrapped by a linear elastic film. Both cellular fluid and extramembrane plasma are considered homogeneous incompressible Newtonian fluids. This paper regards the plasma flow in microvessels as Poiseuille flow in circular conduits. The velocity of the flow field near the wall is relatively small, and the velocity from the near wall to the microvessel axis rises parabolically and reaches a maximum value. The velocity of the flow field is symmetrical about the microvascular axis. Immune cells are located at the axis

of the microvessels and are subjected to a symmetrically distributed fluid shear force. In this paper, COMSOL's "laminar flow" is used to quantify the fluid flow of cell fluid and plasma. Immune cells are elastically deformed under the drive of blood flow, and "solid mechanics" is used to calculate the deformation of the cell membrane. We used "Fluid-Structure Interaction" to calculate the force-displacement transfer at the fluid-structure interface. The rust-colored area in the figure is the blood vessel wall. This paper ignores its deformation and regards it as a rigid wall.

The CFD model in this paper mainly uses COMSOL multiphysics simulation software for numerical calculation. The governing equation used in the model calculation is the Navier-Stokes (N-S) equation,

$$\rho_f \left[\frac{\partial \mathbf{U}_f}{\partial t} + (\mathbf{U}_f \cdot \nabla) \mathbf{U}_f \right] = \nabla \left[\mu \nabla \mathbf{U}_f + \mu (\nabla \mathbf{U}_f)^T \right] + \mathbf{F} - \nabla \mathbf{P} \quad (1)$$

The boundary condition is a no-slip boundary, namely the blood flow velocity near the wall is zero. The PARDISO solver is used in the COMSOL solver setting, which uses the stiffness decomposition matrix inversion to solve the N-S equation and uses parallel computing to improve the solution efficiency. The Euler method is used for meshing in COMSOL, and the motion of the fluid-solid boundary is described using an arbitrary Lagrangian-Euler (ALE) method. Distortion of the mesh near the fluid-solid boundary can be solved by remeshing. The COMSOL multiphysics simulation software has meshing capabilities that meet the computational accuracy requirements, and all meshing is completed within the software.

The ALE coordinate system, independent of the Euler and Lagrange coordinate systems, is generally not completely fixed in space, nor is it completely fixed on the material node. And the ALE grid could achieve the appropriate motion.

The Navier-Stokes equation described by ALE is

$$\rho (v_{t[x]} + (c \cdot \nabla_x) v) = \nabla_x \cdot \sigma + \rho b \quad (2)$$

The mesh velocity described by ALE can be easily obtained

$$\hat{v} = v - \frac{\partial x}{\partial X} w \quad (3)$$

where w is the velocity of the solid.

The model is calculated in a two-dimensional environment, and the leukocytes can move along the axial and radial directions of the tube with two degrees of freedom.

The numerical precision of the calculation adopts the default settings of the software. CFD tasks are completed on different computers due to too many cases to be solved.

Theoretical Description

The so-called Poiseuille flow is the flow of viscous fluid in a circular pipe. When the Reynolds number is less than 2000, the liquid flow in a straight circular pipe with equal cross-section is laminar flow. Poiseuille law

$$Q = \pi r^4 \Delta p / (8 \eta L) \quad (4)$$

It describes the steady flow of an incompressible viscous fluid in a horizontal circular pipe, and the Reynolds number is not

large. When the flow form is laminar flow, the flow rate Q and the pressure difference Δp at both ends of the pipe, the pipe radius r_0 , the pipe length l and the fluid, the relationship of viscosity coefficient η . Poiseuille's law is an essential law of fluid dynamics. Poiseuille flow is one of the few cases where there is an analytical solution to the N-S equation, the x -axis is chosen to be on the vessel axis, and y is the radial coordinate measured outward from the tube axis. Both the circumferential and radial velocity components are zero, and the velocity component parallel to the tube axis is denoted as u , which depends only on y . At the same time, the pressure is constant in each cross-section. In this way, in the N-S equation expressed in cylindrical coordinates, only one axial equation is left, which simplifies to:

$$\mu \left(\frac{d^2 u}{dy^2} + \frac{1}{y} \frac{du}{dy} \right) = \frac{dp}{dx} \quad (5)$$

According to the no-slip boundary condition, $y = R, u = 0$. Available velocity distribution:

$$u(y) = -\frac{1}{4\mu} \frac{dp}{dx} (R^2 - y^2) \quad (6)$$

It can be seen that the velocity from the proximal wall to the microvascular axis increases parabolically.

At $y = 0$, i.e., on the tube axis (the center of the vessel)

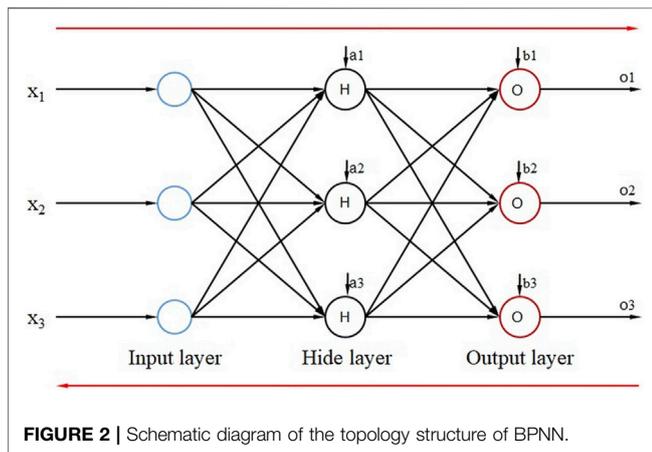
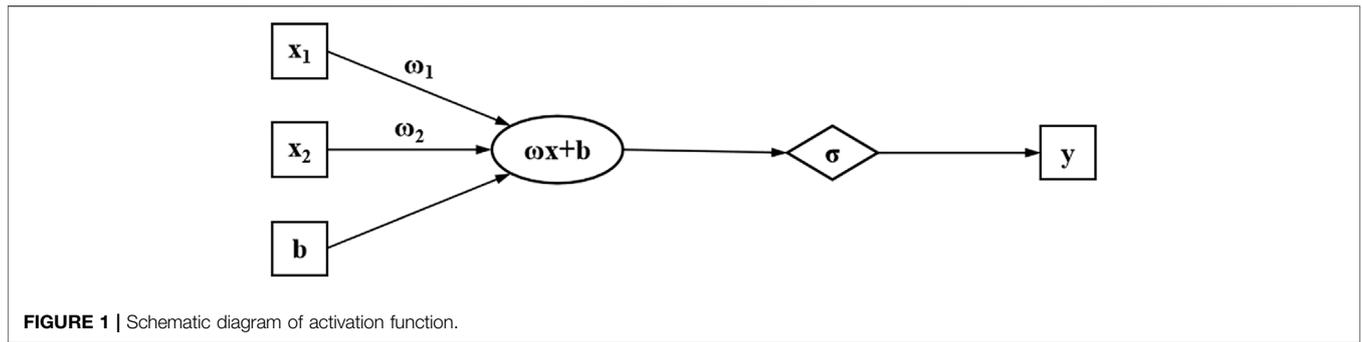
$$u_m = \frac{R^2}{4\mu} \left(-\frac{dp}{dx} \right) \quad (7)$$

Basic Theory of Neural Network

The basic unit of the neural network algorithm is the single-layer perceptron model. The structure of the single-layer perceptron model is simple, and its components only have an input layer and a single output layer. The figure below represents a perceptron model with two input neurons and one output neuron node. x_1 and x_2 represent the stimulus to the neuron, that is, the input information. w_1 and w_2 represent the connection weights between input and output nodes. b represents the excitation threshold of the neuron, that is, the bias. y is the output of the perceptron. The information transfer formula of the single-layer perceptron is expressed as follows:

$$y = \omega_1 x_1 + \omega_2 x_2 + b \quad (8)$$

The perceptron works by finding a linearly separable hyperplane separating the loaded data. What's more, stronger data plane classification capabilities can be obtained by integrating multiple perceptron models. In the original perceptron model, the essence of the inter-layer transfer of the neural network is to perform a linear transformation of the data. This determines that a simple perceptron model can only solve linearly separable problems, but cannot deal with non-linear problems. Based on this, applying a non-linear activation function to the neuron node endows the neural network with the ability to solve linear inseparable problems. The non-linear activation function completes the linear transformation to fulfill output characteristics. As shown in **Figure 1**, the figure represents



the non-linear modification of the linear transformation. The application of non-linear activation functions introduces non-linear factors into the neural network, allowing it to classify data planes using smooth surfaces. This enables the neural network to have superior data classification and regression capabilities.

BP Neural Network

Assume that the number of nodes in the input, hidden, and output layers are l , m and n . The weights and biases from the input layer to the hidden layer and from the hidden layer to the output layer are in order ω_{ij} , ω_{jk} and a_j , b_k , learning rate η . The excitation function is $g(x)$. The detailed steps of BPNN modeling are as follows:

BPNN is a multi-layer feedforward neural network based on an error back propagation algorithm. The principle is to use the common gradient descent algorithm and gradient search technology to minimize the error mean square error between the actual output value and the expected output value of the network in successive update iterations. The model topology of BPNN consists of an input layer, a hidden layer, and an output layer. **Figure 2** shows a simple three-layer neural network. The input signal of the neural network acts on the output node through the hidden layer node and generates the output signal. By adjusting the connection weights of input nodes and hidden layer nodes, the connection weights and thresholds between the hidden layer and output nodes during the training, the deviation between the actual output and the

expected output value decreases along the gradient direction. After repeated training and parameter adjustment, the network parameters (connection weights and thresholds) that match the minimum error are determined. The network parameters determined by the trained neural network process the input information of the input samples with similar characteristics and output the prediction information. Then the prediction result with the smallest error can be obtained.

Assume that the number of nodes in the input, hidden, and output layers are l , m and n respectively. The weights and biases from the input layer to the hidden layer and from the hidden layer to the output layer are w_{ij} , w_{jk} and a_j , b_k . The learning rate is η and the excitation function is $g(x)$. The detailed steps of BPNN modeling are as follows:

- (1) Network initialization. First, the weights and thresholds between the input and the hidden layers are initialized. Among them, the common activation function is the sigmoid function. Its function expression is,

$$f(x) = \frac{1}{1 + e^{-x}} \tag{9}$$

- (2) Data forward calculation. For the three-layer BPNN shown in **Figure 2**, the output sum of the hidden layer and the output layer H_j and O_k are respectively

$$H_j = g\left(\sum_{i=1}^l \omega_{ij}x_i + a_j\right) \tag{10}$$

$$O_k = g\left(\sum_{j=1}^m \omega_{jk}H_j + b_k\right) \tag{11}$$

- (3) Error calculation

Assuming that Y_k is the expected output of the network, the error calculation formula is

$$E = \frac{1}{2} \sum_{k=1}^n e_k^2 \tag{12}$$

$$e_k = Y_k - O_k \tag{13}$$

- (4) Update of weights and biases. The weights and biases are updated through the back-propagation of the error to make

the error function reach the minimum value, where the update formulas of the weights and biases are respectively

$$\begin{cases} \omega_{ij} = \omega_{ij} + \eta H_j (1 - H_j) x_j \sum_{k=1}^n \omega_{jk} e_k \\ \omega_{jk} = \omega_{jk} + \eta H_j e_k \end{cases} \quad (14)$$

$$\begin{cases} a_j = a_j + \eta H_j (1 - H_j) x_j \sum_{k=1}^n \omega_{jk} e_k \\ b_k = b_k + \eta e_k \end{cases} \quad (15)$$

Furthermore, we utilize mean squared error (MSE) to measure the accuracy of BPNN predictions. The formula for calculating the MSE is,

$$MSE = \frac{SSE}{I} = \frac{1}{I} \sum_{i=1}^n \omega_i (y_i - \hat{y}_i) \quad (16)$$

The Training Process of BP Neural Network

The network training of BPNN includes the following processes: (1) Integrate the training data set. The training of BPNN requires large-scale data support, and the training samples need to contain both input data and label data. The accuracy of the data directly affects the prediction accuracy of the neural network. Data normalization is often necessary for the specific training to obtain similar weighting tendencies for different features. Normalization can reduce the computational complexity and speed up the convergence, and also facilitate the subsequent training optimization because the output of the activation function is 0-centered.

- (2) Network structure design. The network structure aims to clarify the number of network layers, hidden layer nodes, the activation function between layers, the training and the loss function. As the network structure is determined, all data dimensions are established.
- (3) Initialize the weights. The parameter update in the training of BPNN depends on the gradient descent algorithm (mainly including batch gradient descent (BGD) method, stochastic gradient descent (SGD) method, mini-batch gradient descent (MBGD) method, etc.). Before starting training with the gradient descent algorithm, the weights and biases in the network need to be initialized. To eliminate the symmetrical weights, many initialization methods are currently used to randomly initialize the weights of each layer. Too small initialization weight made the non-linear activation function lose its non-linear modification ability. On the contrary, too large an initialization weight causes the problem of gradient disappearance.
- (4) Network training. BPNN is based on the back-propagation of the error between the network output and the label to update the weights and repeat the iterative calculation to make the loss function reach the expected desired value.

RESULTS

CFD Data

We used the CFD data of immune cell transmigration as the training data of the BPNN model. Considering the limited effect

of immune cell stiffness on cell motion and deformation and flow field in the transmigration simulation, it is not used as an input parameter of the neural network. This paper aims to construct a time-dependent prediction model for leukocyte transmigration to establish a model of immune cell movement and flow field. The data we use are all from the data obtained by our CFD simulation, including displacement parameters, cell deformation parameters, flow velocity, fluid lift, and drag parameters. We have fully investigated the hydrodynamic parameters of the environment in which immune cells migrate to ensure the accuracy of the data obtained by CFD. We performed blood flow Re of 0.1, 0.11, 0.12, 0.13, 0.14, 0.15, 0.16, 0.17, 0.18, 0.19, and 0.2 and AR of 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95, a total of 121 CFD simulations. Among them, the combination of Re and AR is (0.1, 0.5), (0.1, 0.55), (0.11, 0.5), (0.12, 0.5), (0.13, 0.5) and (0.14, 0.5). The impaction phenomenon is not included in the training. We will exclude 115 sets of CFD simulation data from these six sets of examples as the basic data set for training.

The 115 groups of Re and AR parameters in the basic data set are used as input data. The transmigration is the target vector to train the time-dependent leukocyte transmigration prediction model based on the BPNN model. In addition, we used 14,221 samples consisting of Re , AR , and cell downflow displacement parameters as a training dataset to train cell motion and flow field prediction models during transmigration. Among them, the target parameters are the cell deformation, the downstream velocity, the fluid lift, and drag forces. In order to test the accuracy of the model, 10 groups of Re and AR were used as (0.2, 0.53), (0.19, 0.57), (0.18, 0.63), (0.17, 0.67), (0.16, 0.73), (0.15, 0.77). New examples of (0.14, 0.83), (0.13, 0.87), (0.12, 0.93) and (0.11, 0.97) are used as test datasets to validate the time-dependent transmigration prediction model. The above parameter combination simulates the migration environment in which most immune cells are located and guarantees the generality of this study. At the same time, four groups of leukocyte transmigration CFD examples with Re and AR of (0.16, 0.73), (0.15, 0.77), (0.14, 0.83) and (0.13, 0.87) were used to test the accuracy of the prediction model of the transmigration. The hydrodynamic simulation still uses the leukocyte transmigration CFD model shown in **Figure 3**.

Time-Dependent Prediction of Leukocyte Transmigration Based on BP Neural Network Model

Many studies have shown that a 4-layer neural network with only two hidden layers can predict most regular data matrices accurately. The input parameters of the time-dependent leukocyte transmigration prediction model are two vectors, Re and AR , so the number of neurons in the input layer is 2. To avoid the long training time caused by many nodes, we draw on the empirical formula for node selection in the hidden layer,

$$\sum_{i=0}^M C_i^n > h \quad (17)$$

Where M is the number of nodes in the input layer, namely 2. I is the number of hidden layer nodes to be used. h is the number of

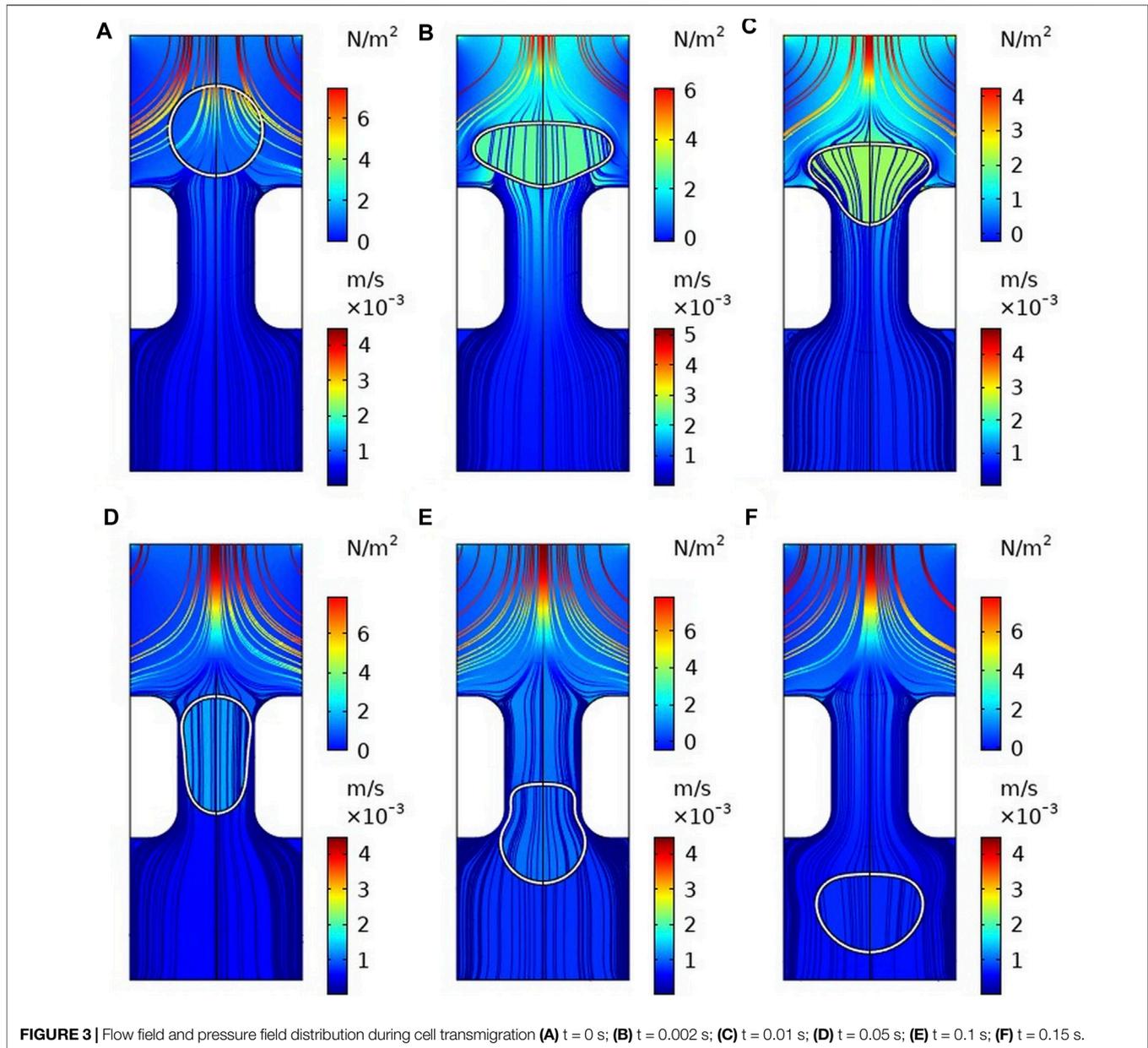


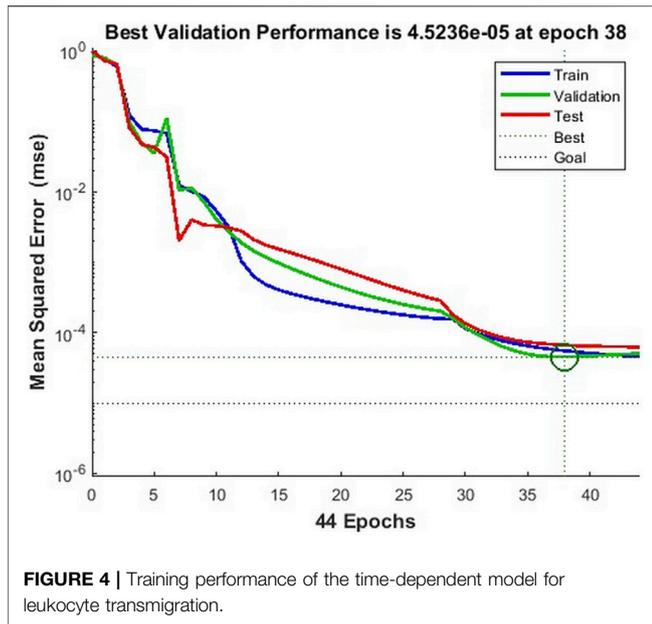
FIGURE 3 | Flow field and pressure field distribution during cell transmigration (A) $t = 0$ s; (B) $t = 0.002$ s; (C) $t = 0.01$ s; (D) $t = 0.05$ s; (E) $t = 0.1$ s; (F) $t = 0.15$ s.

training samples of the BP model. After calculation, the number of nodes in the hidden layer should not be less than 15. Therefore, we use two-layer hidden layers with 5 and 10 nodes, respectively, to construct the time-dependent cell transmigration prediction model. The activation function from the input layer to the hidden layer and the hidden layer adopts the tansig function, and the activation function from the hidden layer to the output layer adopts the purelin function. In addition, the trainlm function is used as the training function of the BP neural network. Meanwhile, the maximum training times, training target and learning rate are 1,000, 1×10^{-5} and 0.03, respectively. **Figure 4** shows the training performance of the time-dependent leukocyte transmigration prediction model. The model achieves the

best verification performance when the number of training times is 38, and its value is 4.524×10^{-5} .

Based on the default partitioning method, the BPNN model randomly assigns 115 samples to the training set, validation set and test set according to the ratio of 60%, 20%, and 20%. **Figure 5** shows that the regression accuracies on the training set, validation set, test set, and total dataset reached 0.99959, 0.99907, 0.99874, and 0.99949, respectively. This further proves the prediction robustness of the BPNN model.

To test the prediction accuracy of the leukocyte transmigration time prediction model, we predicted the transmigration time of 10 groups of cases in the test data set different from the training samples and compared them with the CFD results. We use the coefficient of determination as a criterion to analyze the deviation



between the BPNN and CFD simulation and further measure the robustness of the mechanical prediction system. The formula for calculating the coefficient of determination is:

$$R^2 = \frac{\sum (\hat{y}_j - \bar{y})^2}{\sum (y_j - \bar{y})^2} \quad (18)$$

Where \hat{y}_j is the predicted value of BPNN and \bar{y} is the average value of the real value of the CFD simulation, y_j is the true value of the CFD simulation. The interval range of the coefficient of determination is [0, 1], and the closer to 1, the better the prediction effect.

Figure 6 shows the comparison between the predicted time-dependent of BPNN and the time-dependent CFD for 10 groups of leukocyte transmigration examples from the Re interval of [0.1, 0.2] and the AR interval of [0.5, 1]. The results show that the predicted value in the middle region of the AR interval is in good agreement with the true value and slightly worse at the two ends. The R^2 was 0.99846, indicating that the leukocyte transmigration prediction model had excellent prediction accuracy. Based on this, we constructed a BPNN-based time-dependent leukocyte transmigration prediction model a three-dimensional map of the predicted leukocyte transmigration time in the Re interval [0.1, 0.2] and AR interval [0.6, 1] (**Figure 7**). As shown, the leukocyte transmigration time increased significantly with the decrease of Re . Compared with Re , the reduction of AR has a more prominent effect on prolonging the time-dependent leukocyte transmigration. Under the dual influence of Re and AR , the increase in leukocyte transmigration time was more pronounced and manifested as a sharp upward curl on the map.

Prediction of Leukocyte Transmigration Based on BP Neural Network Model

The prediction model can predict the time-dependent leukocyte transmigration in specific Re and AR intervals. However, the

morphological changes of cells during transmigration and the changes in the surrounding flow field remain unclear. The robustness of the BPNN model aided to construct a leukocyte transmigration prediction model. As mentioned above, 14,221 samples including Re , AR , cell down-flow displacement, cell deformation, down-flow velocity, received fluid lift and drag parameters were used as the training dataset to train the BPNN model. The data matrix containing Re , AR and cell forward displacement is used as input data to establish the corresponding relationship with the target data. The target data vectors are R_{max} , V_y , F_l and F_d , respectively. Using **Eq. 14** to calculate the number of neurons in the hidden layer, we use 50 nodes to fill the hidden layer. We construct a two-layer hidden layer with 20 and 30 nodes, respectively. The selection of activation and training functions is the same as the leukocyte transmigration time-dependent prediction model. At the same time, the determination coefficient of **Eq. 11** is still used to measure the prediction accuracy of the BPNN prediction model.

Cell Motion Prediction

First, we train the neural network with cell deformation parameters as target vectors. Unlike the setting threshold of the time-dependent prediction model of leukocyte transmigration, the better prediction accuracy can be obtained by setting the training target as 10^{-3} . As shown in **Figure 8A**, the prediction model achieves the best training performance when the number of training times is 34, and its value is 0.00094417. At the same time, a fitting accuracy of 0.99992 was obtained in training on the overall training set with a sample size of 14,221 (**Figure 8B**).

Next, we use the cell deformation prediction model to predict the cell deformation during the transmigration for the four new examples mentioned above. Their Re and AR are set to (0.16, 0.73), (0.15, 0.77), (0.14, 0.83), and (0.13, 0.87), respectively. **Figure 9A–D** compares the predicted cell deformation values and the CFD values of the four groups of calculations, respectively. The figure shows the deformation prediction of the cell forward displacement by the prediction model is consistent with the CFD results. The coefficients of determination, which measure the degree of agreement between the BPNN predictions of the four groups of examples and the real solutions of CFD, are 0.9998, 0.9998, 0.9998, and 0.9999, respectively, showing an excellent prediction effect.

By changing the target vector of the prediction model to the velocity vector of the cell forward movement, we constructed a model for predicting the leukocyte transmigration speed. Due to the increased degree of non-linearity between the input data and the target vector, the threshold for the training target was set to 10^{-10} . The training performance of the model and the data fitting curve are shown in **Figure 10A,B**. When the number of training times reaches 64, the BPNN model obtains the best training performance with a value of 9.862×10^{-11} . At the same time, the R value of the training data fitting reached 0.99922, indicating a good mapping relationship between the input data and the output data in the forecast model. Subsequently, we tested the established cell forward velocity prediction model. Using the same 4 datasets, the good predictive ability of the forecasting model is revealed.

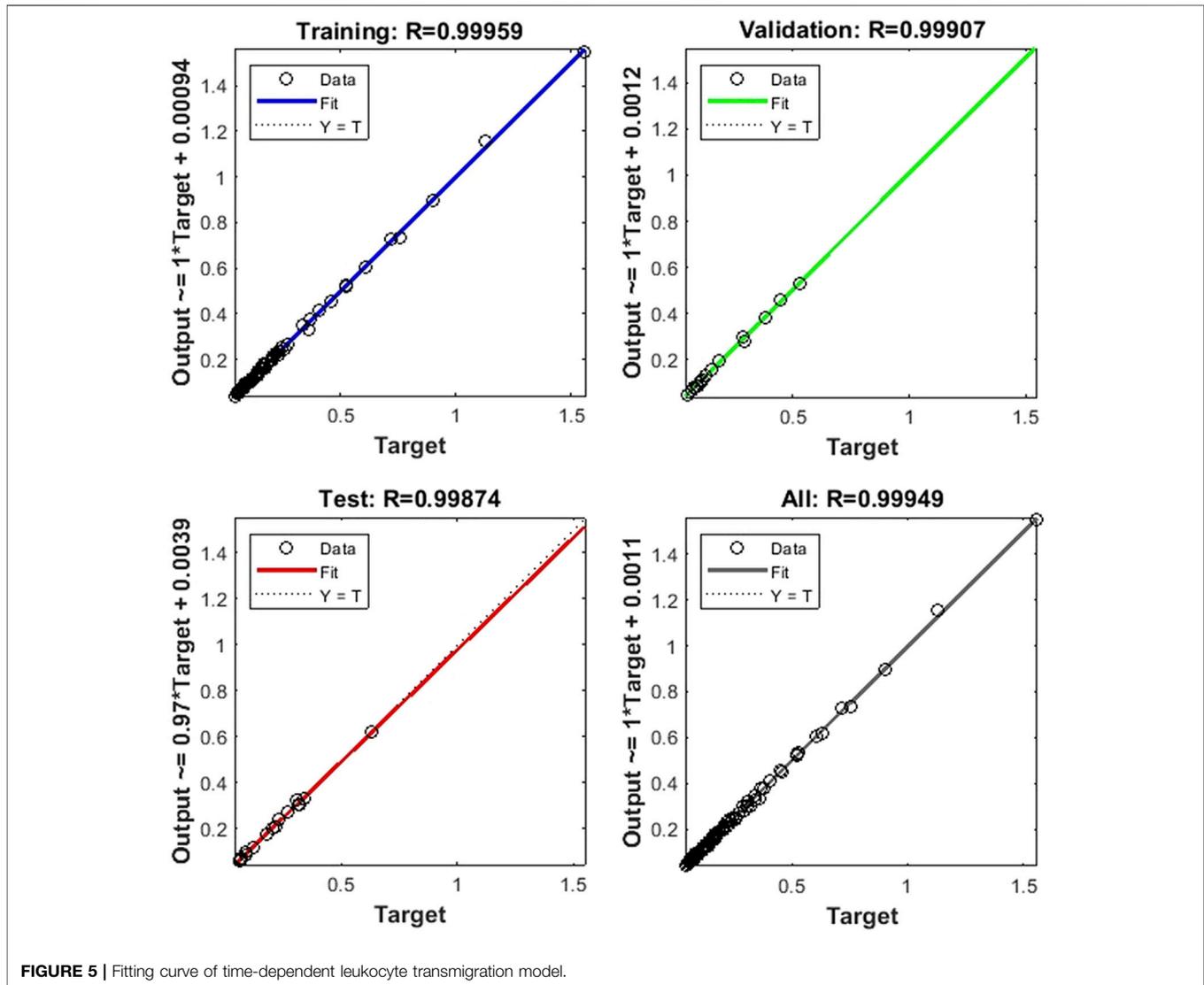


FIGURE 5 | Fitting curve of time-dependent leukocyte transmigration model.

Figure 11A–D all show that the predicted value of BPNN can well fit the CFD solution, and the coefficients of determination of the closeness between the predicted curve and the real curve are 0.9990, 0.9985, 0.9971 and 0.9979.

Prediction of Cell Lift and Drag Force

Based on the same network skeleton of the leukocyte transmigration prediction model, we replace the trained target vector with the cell fluid lift forces to calculate the accuracy of the prediction model in predicting the cell fluid lift forces. Since the non-linearity of the data further increases, the training objective and minimum gradient threshold are set to 1×10^{-13} and 1×10^{-10} , respectively, to obtain better prediction accuracy. Figure 12A shows that the prediction model achieved the best training performance when the training order reached 153, and its value was 9.970×10^{-14} . At the same time, the fitting curve of the model achieved an accuracy of 0.99867 (Figure 12B).

Similar to the prediction of cell deformation and downstream motion velocity, we also found that the prediction model of the transmigration is highly accurate in predicting the fluid lift force on the cell. Figure 13A–D of show the prediction curves of the prediction model for the fluid lift forces on cells in the four new cases, respectively. The results show that the predicted curve is consistent with the CFD curve, and the difference between the predicted and actual values at each data point is very small. The coefficients of determination of the predicted curve and the real CFD curve in the four groups of examples are 0.9915, 0.9943, 0.9946 and 0.9965, respectively, showing excellent prediction performance.

Finally, we replaced the target vector of the leukocyte transmigration prediction model with the drag force of the cell to observe the prediction ability of the model to the drag force. After adequate iterations were implemented, it was found that the training model with the training target and the minimum gradient threshold set to 1×10^{-14} and 1×10^{-12} , respectively,

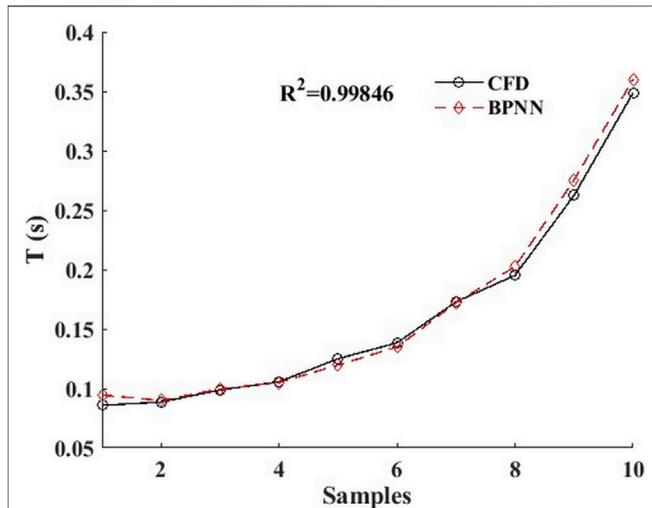


FIGURE 6 | BPNN prediction value and CFD real value of 10 groups of calculation examples based on time-dependent leukocyte transmigration prediction mode.

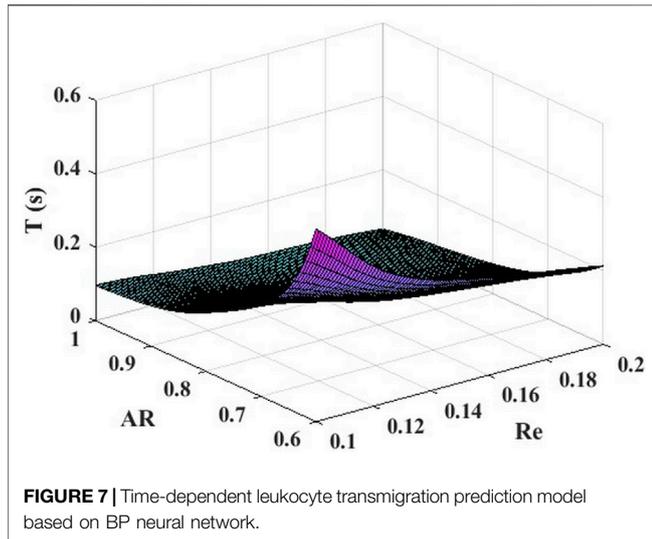


FIGURE 7 | Time-dependent leukocyte transmigration prediction model based on BP neural network.

could maintain a good prediction of the target. When the number of training times reaches 716, the model has the best training performance with a value of 9.997×10^{-15} (Figure 14A). At this point, the fitting parameter of the training data reached 0.99093 (Figure 14B). Based on the same test case, we analyzed the prediction accuracy of the prediction model for the drag force of cells during the transmigration. Figure 15A–D show the drag force prediction curves and CFD curves of different examples, respectively. The prediction accuracies of these examples are 0.9257, 0.9451, 0.9512 and 0.9569, respectively. The entire computation process, including CFD and BPNN, is shown in Figure 16. The prediction models have significant discrepancies in the prediction of the drag force in the initial cell movement stage, the pore entry stage and the pore exit stage. The model maintained a high prediction accuracy for drag force in other periods.

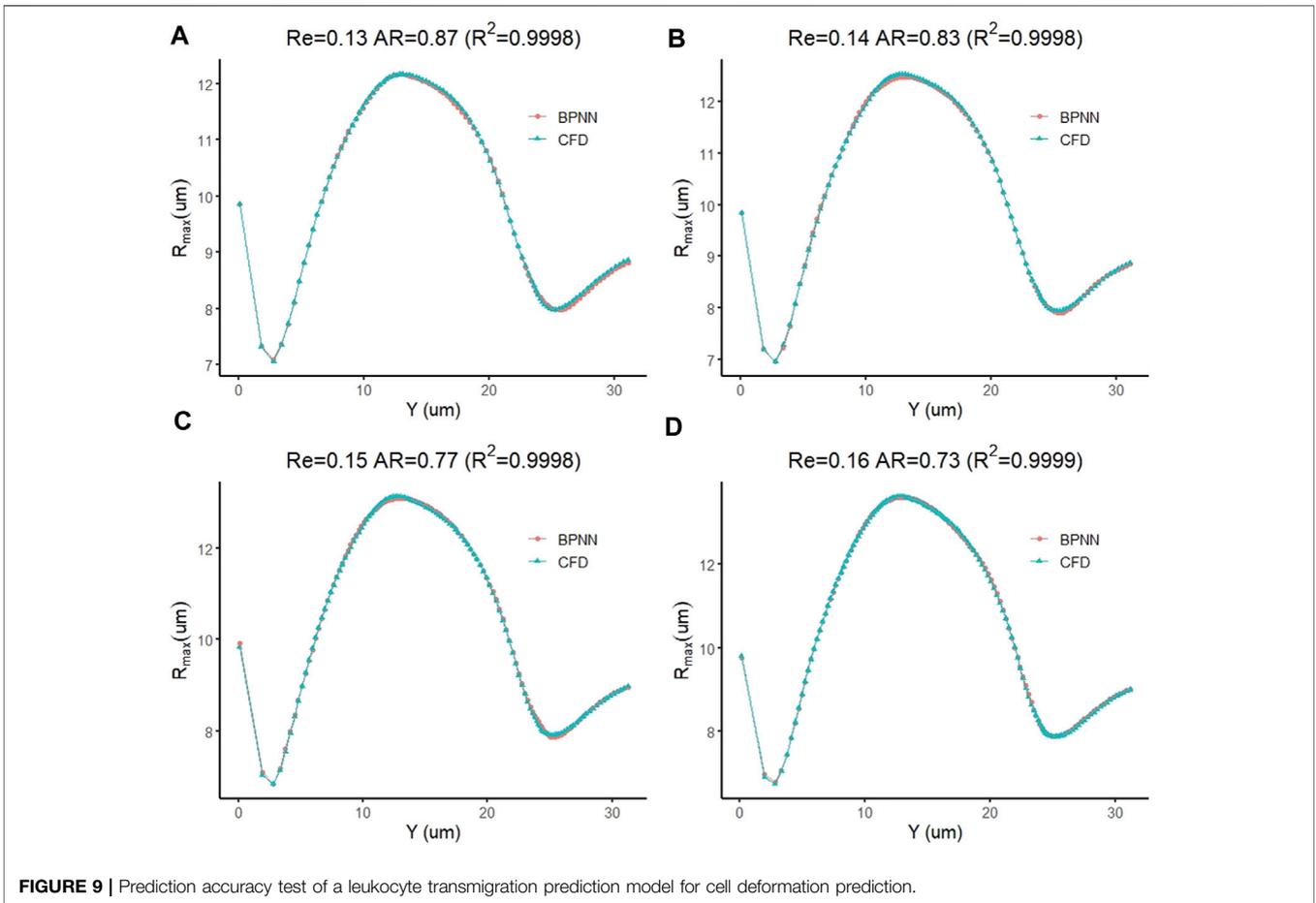
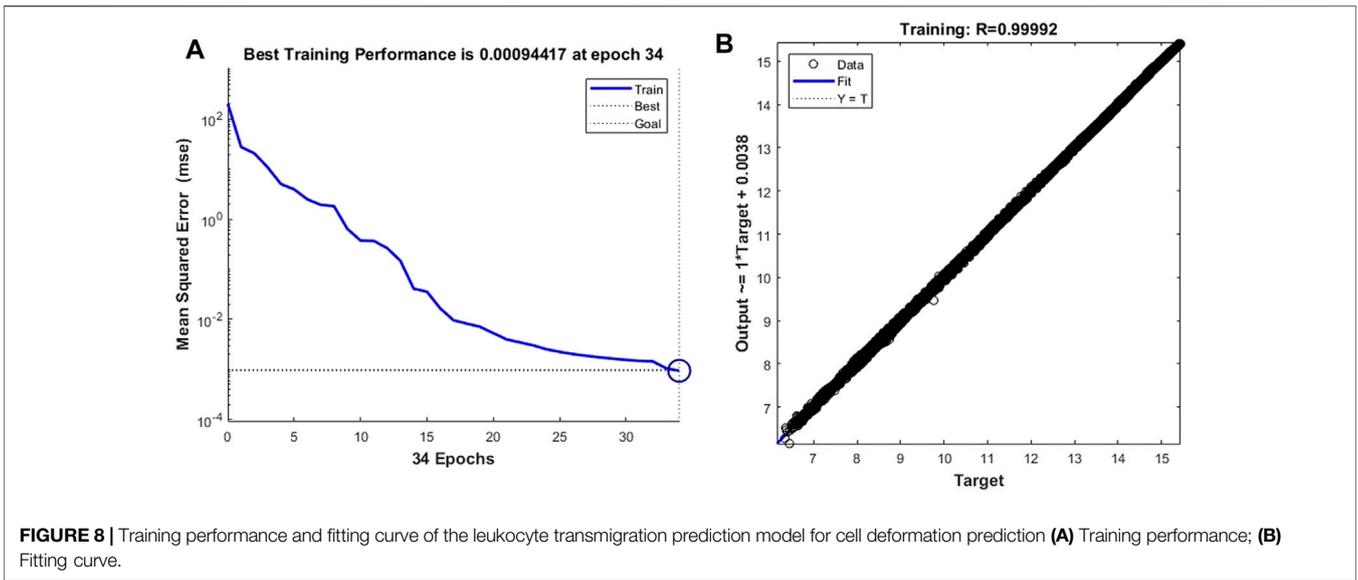
Computational Efficiency Comparison

Compared with traditional CFD simulation methods, we found that BPNN greatly shortens the simulation time of the model and maintains high accuracy under the condition of sufficient sample size. Once the network model is trained, BPNN can instantaneously predict the time-dependent transmigration of cells. The computer configuration for this article is (Intel(R) Core(TM) i5-4570 3.2 GHz RAM 8G). Based on the same computing resources, the training time of the BP model is less than 10 min in total. In contrast, the total time required for establishing and training the BP model is comparable to that of a CFD example. This fully reflects the punctuality and efficiency of BPNN forecasting.

DISCUSSION

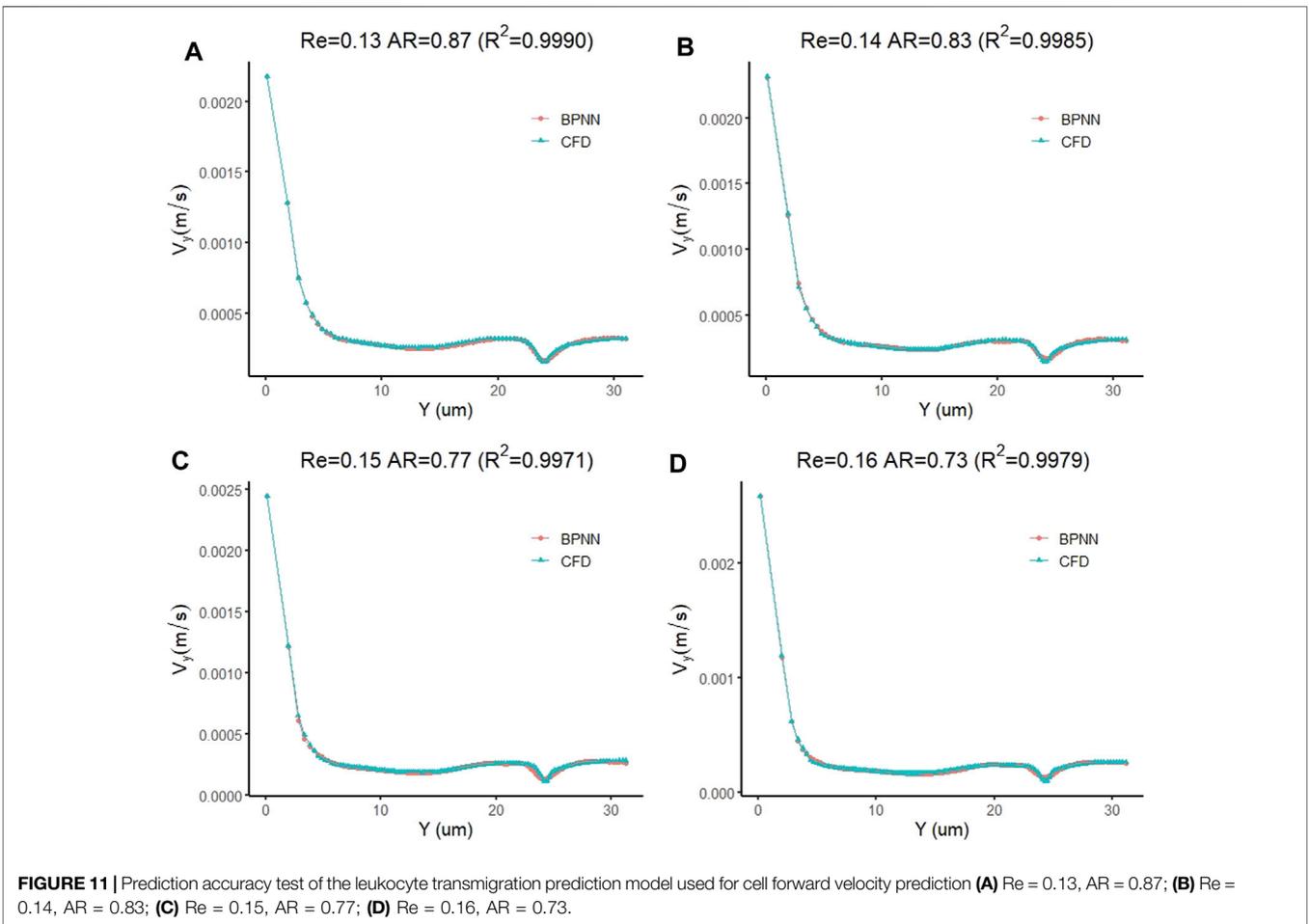
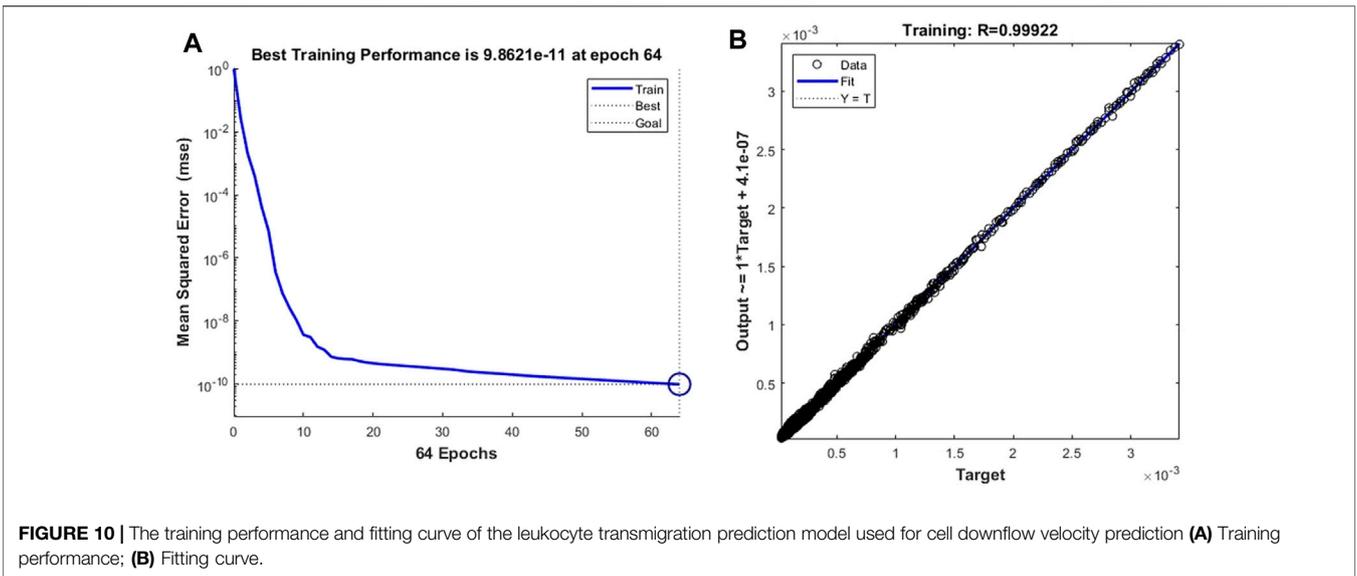
Immune cell infiltration and is of great concern in the biomedicine (Shen et al., 2020). Traditional CFD methods are often accompanied by complex and time-dependent modeling and calculations (Khaleghi et al., 2021). Based on this, many leukocyte transmigration CFD simulation data obtained in the previous paper, we tentatively established a time-dependent prediction model of the transmigration. The cellular transmigration time-dependent prediction model can predict the time it takes for cells to pass through the pores in the vascular environment in specific Re and AR intervals. On this basis, the leukocyte transmigration model can predict the cell movement and flow field changes during the transmigration with extremely high accuracy. Both prediction models have a double hidden layer structure, and the total number of hidden layer nodes is 15 and 50, respectively. The time-dependent leukocyte transmigration prediction model only has two input parameters, Re and AR . The input data of the leukocyte transmigration prediction model also includes the cell forward displacement parameter. Based on the constructed model, we selected 10 time-dependent samples of leukocyte transmigration and 4 examples of leukocyte transmigration in the training Re and AR intervals to test the prediction accuracy of the two prediction models and obtained superior predictions Effect. Compared to traditional CFD techniques, the predictive capability of the BPNN model significantly highlights its potential applications in fluid mechanics (Wong and Kim, 2018).

With the rise of the technology, many algorithms have emerged that incorporate physical models (Jiang et al., 2019; Liu et al., 2022; Liu et al., 2022). In this regard, the neural network family to which BP belongs plays an important role (Wong and Kim, 2018). Earlier studies have simply used neural network algorithms to improve the accuracy of data processing (Rediniotis and Vijayagopal, 1999). For example, Wang et al. applied the artificial neural network based on BP theory to the data processing of the five-hole probe in the fluid experiment, and put forward the conclusion that the accuracy and reliability of the prediction results are better than the linear three-dimensional interpolation method. Then, the predictive capabilities of neural networks, which can maintain high accuracy and rapidity, are gradually applied to fluid mechanics (Rediniotis and Vijayagopal,



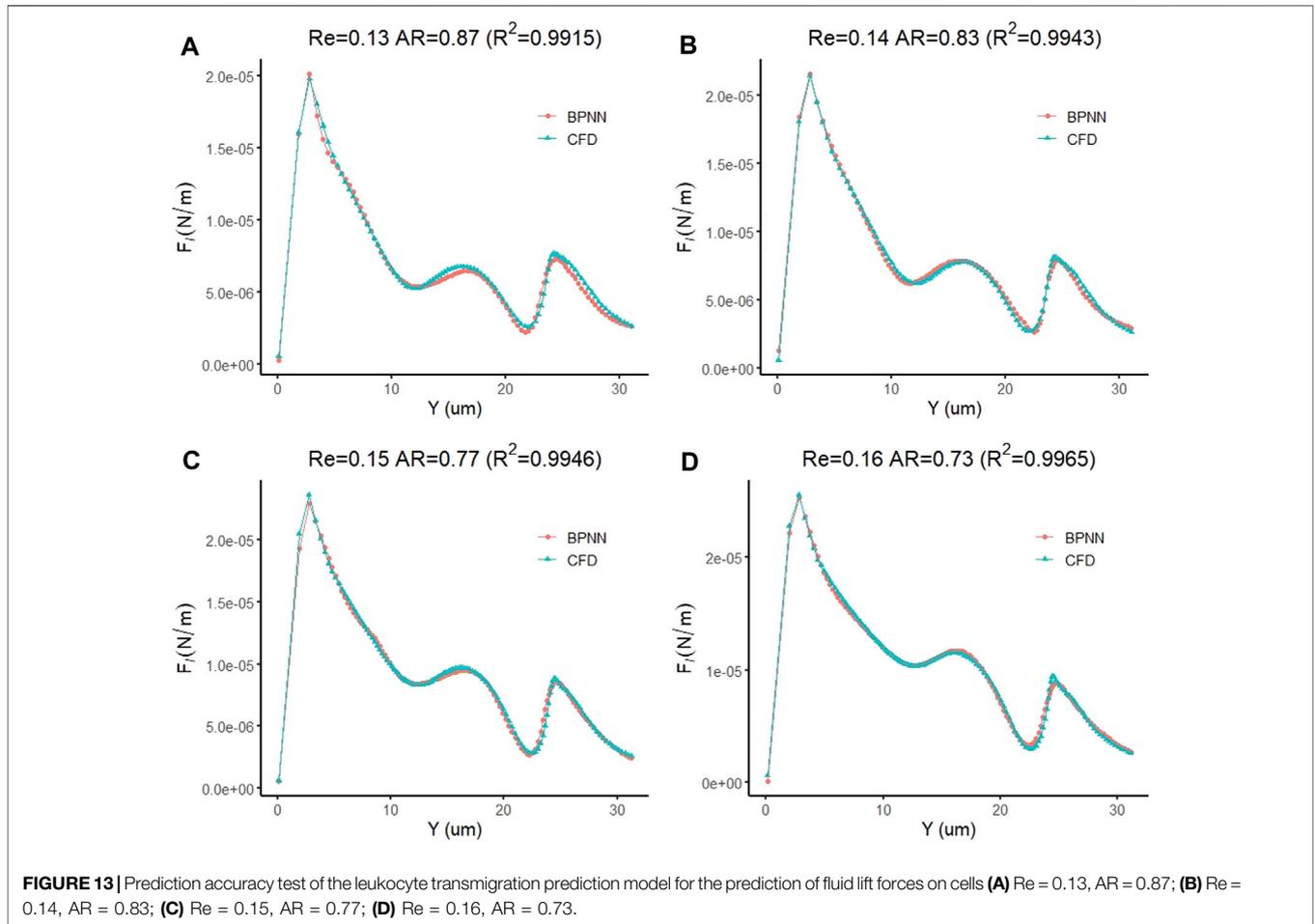
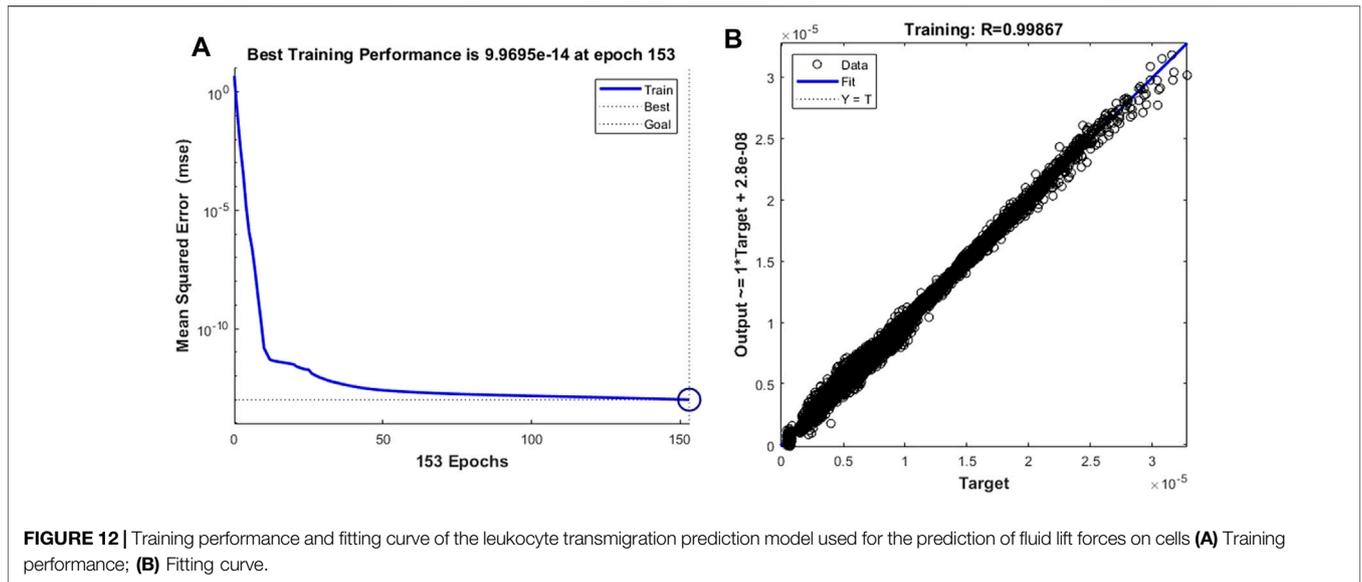
1999; Fan et al., 2003; Cheng et al., 2020). For example, the BP model can effectively predict the displacement and dominant frequency of the vortex-induced vibration of flexible cylinders

commonly found in engineering. But the neural network has a significant disadvantage, that is, the successful training of the network needs to rely on a considerable number of samples (Liu



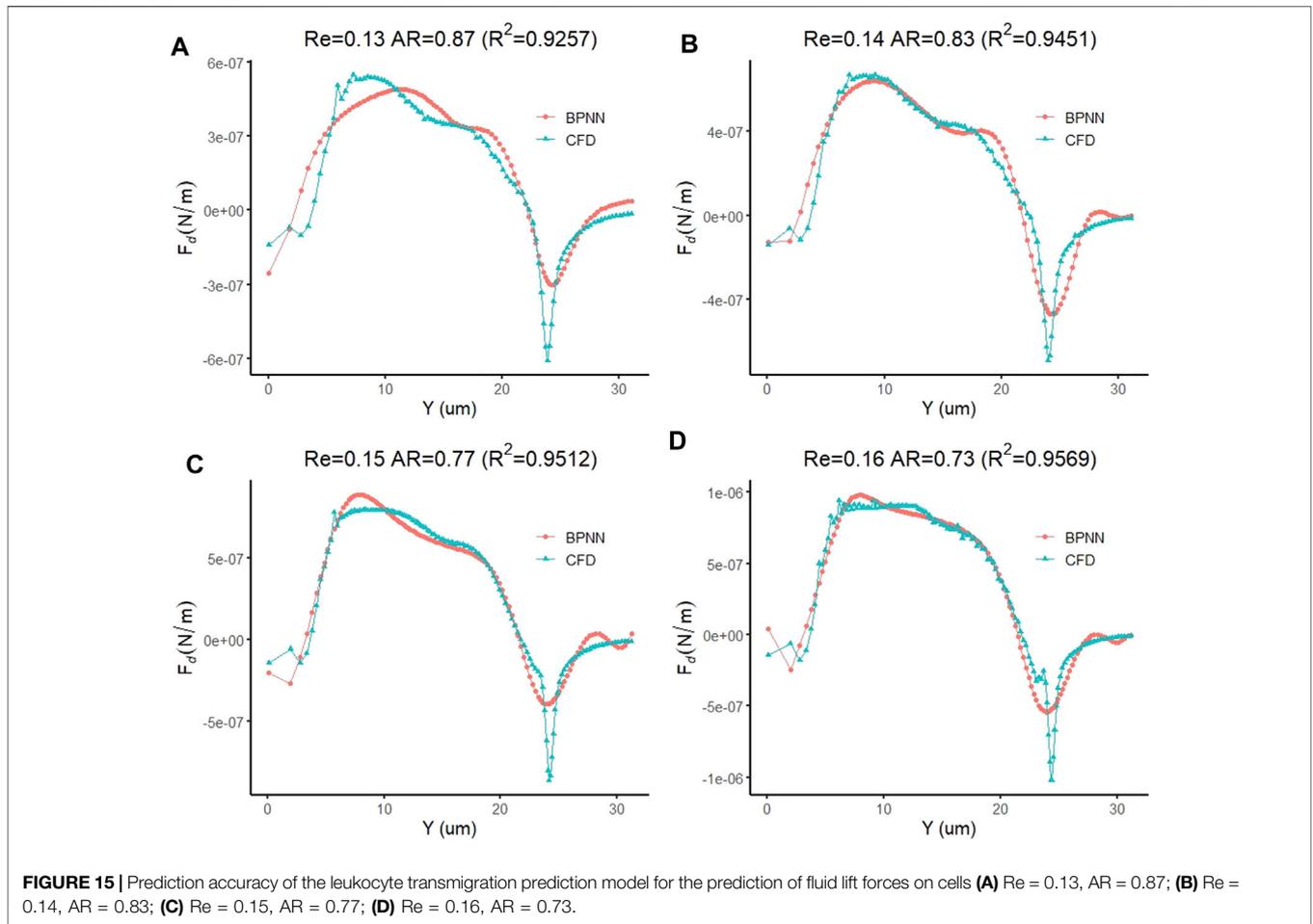
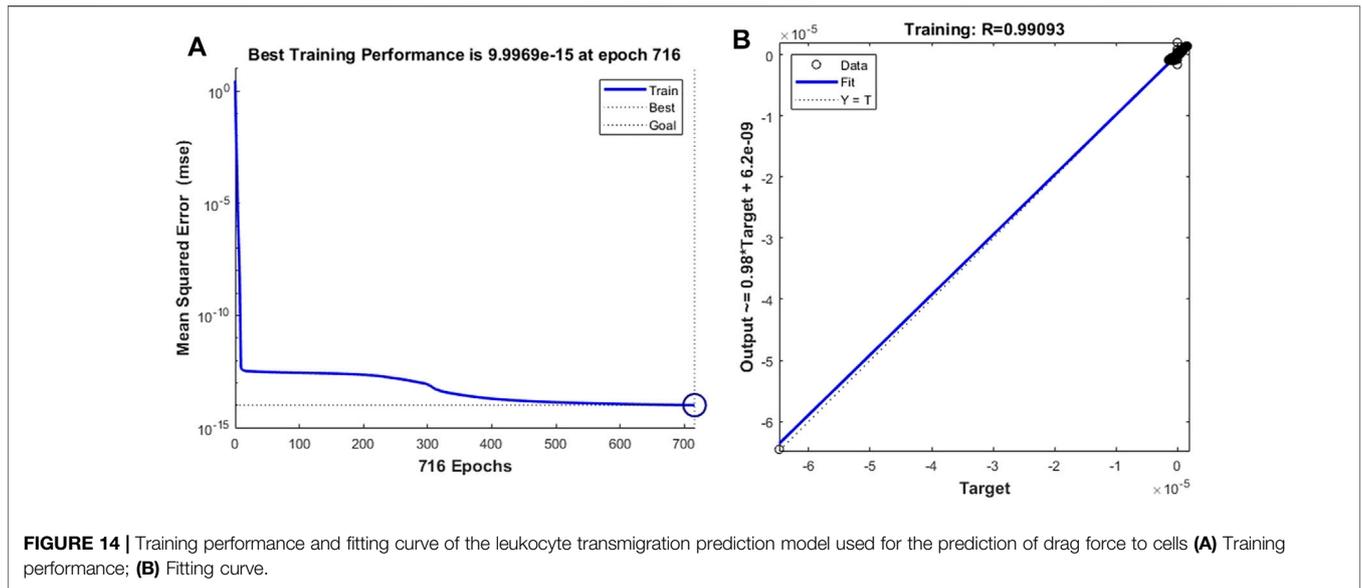
et al., 2015; Chen, 2017). In this regard, other algorithms can better address this problem (Jiang et al., 2019; Sun et al., 2022; Wang and Cao, 2029). The reason for abandoning the large-scale

physical models and choosing the neural network model is primarily because of its efficient and fast simulation and prediction capabilities (Li and Peng, 2007; Kang and Cho,



2014). Like most forecasting models, the two forecasting models established in this paper based on the BPNN model show the instantaneousness and high accuracy of forecasting.

A sufficiently trained neural network model also can guide engineering design, such as the rational planning of the airfoil design process by predicting the leading edge pressure



distribution of a hybrid airfoil (Yao and Sung et al., 2018; Sekar et al., 2019; 廖鹏与姚磊江等, 2019). What’s more, an active flow controller incorporating deep reinforcement learning neural

networks significantly reduces lift and drag fluctuations in the flow around a cylinder (Haryanto et al., 2014; Peng et al., 2020; Tang et al., 2020). Because of the wide application of BPNN

model in fluid mechanics, we integrated the leukocyte transmigration CFD dataset to establish a time-dependent leukocyte transmigration prediction model. We randomly predicted the CFD results of leukocyte transmigration models for several groups of different Re and AR using established two-class prediction models. Compared with the real CFD results, the prediction model showed robustness in both the time-dependent transmigration and the changes of cell motion and flow field during the transmigration.

This paper mainly focuses on the processing of leukocyte migration data obtained by CFD simulation using BPNN technology, and fully demonstrates the advantages of artificial neural networks for the prediction of immune cell perforation compared with traditional CFD simulation after having a certain amount of data in the early stage. This advantage is mainly reflected in forecasting speed and reliable accuracy. The effects of biochemical factors on leukocyte migration were significantly different from mechanical factors on leukocyte migration. The adhesion and migration of immune cells are crucial in the body immunity and host defense. Integrins on the surface of immune cells are the core molecules regulating immune cell adhesion and migration. The main scientific question to explore the influence of biochemical factors is how immune cells sense changes in the extracellular microenvironment, thereby controlling tissue-specific migration of immune cells by regulating integrin function. It is more of an on-off regulation. Migration often occurs because cells sense signals from the outside world, such as white blood cells sensing abnormal proteins released by bacteria. The cell then turns on a switch inside itself, initiating the migration process. The mechanical factor focuses on the fluid environment in which the immune cells in the blood vessels are located and the forces they experience during migration, such as the fluid environment characterized by the two parameters Re and AR in this paper.

We performed numerical simulations of the model based on the fine, finer, and ultrafine meshing methods, respectively, to verify the reliability of our CFD results. Relevant content has been added to the revised manuscript. The numbers of meshes included in the numerical model under the fine, finer and ultrafine meshing are 8522, 21,237 and 55,245 meshes, respectively. As shown in **Supplementary Figure S1A**, the time- R_{max} curves of the three meshing methods have the same trend. However, the resulting finer distribution of data points differs slightly from the other two meshing methods. The finer and ultra-fine results can fit well. Considering the increasing demand for computing resources brought about by the increase in grids, we use a finer grid division method for subsequent simulations. As shown in **Supplementary Figure S1B**, the finer

mesh division method locally refines the fluid-structure interaction boundary based on the physical model of cell perforation based on different physical field distributions. The mesh model based on this meshing method can better meet the calculation requirements of the physical model with large non-linear characteristics to ensure that the results are reliable enough.

Overall, our results demonstrate the feasibility and robustness of BPNN in prediction studies of leukocyte transmigration. The strong generalization ability of the neural network also provides a technical guarantee for a comprehensive and complete prediction model of cell infiltration in the future. Compared with the traditional CFD technology, the neural network model is easy to implement because it does not require an explicit fitting function. At the same time, the prediction time of this method also significantly shortens the simulation time based on traditional physical models.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

QC prepared the manuscript and implemented the data analysis. ZY was responsible for programming and data processing. H-PL proposed the idea and worked as a supervisor for all procedures. All authors contributed to the paper and approved the submitted version.

FUNDING

This work was supported by the Fund of Biosecurity Specialized Project of PLA (NO. 19SWAQ18).

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/fbioe.2022.881797/full#supplementary-material>

Supplementary Figure S1 | Mesh independence verification and ultra-fine meshing **(A)** Time- R_{max} curve under relatively fine, finer, and ultrafine meshing; **(B)** Based on the ultra-fine meshing method global and local meshing of the cell perforation physical model.

REFERENCES

- Ansari, A., Boosari, S. S. H., and Mohaghegh, S. D. (2020). Successful Implementation of Artificial Intelligence and Machine Learning in Multiphase Flow Smart Proxy Modeling: Two Case Studies of Gas-Liquid and Gas-Solid CFD Models. *J. Petroleum Environ. Biotechnol.* 101 (1), 1–8. doi:10.35248/2157-7463.20.11.401
- Bentley, D. L. (2014). Coupling mRNA Processing with Transcription in Time and Space. *Nat. Rev. Genet.* 15 (3), 163–175. doi:10.1038/nrg3662
- Biswas, S., Kuznetsov, G., Ogden, P. J., Conway, N. J., Adams, R. P., and Church, G. M. (2018). Toward Machine-Guided Design of Proteins. *BioRxiv*, 337154. doi:10.1101/337154
- Bonifacio, H. F., Maghirang, R. G., and Glasgow, L. A. (2014). Numerical Simulation of Transport of Particles Emitted from Ground-Level Area Source Using Aermot and CFD. *Eng. Appl. Comput. Fluid Mech.* 8, 488–502. doi:10.1080/19942060.2014.11083302

- Chen, D. (2017). Research on Traffic Flow Prediction in the Big Data Environment Based on the Improved RBF Neural Network. *IEEE Trans. Ind. Inf.* 13 (4), 2000–2008. doi:10.1109/TII.2017.2682855
- Cheng, Z., Ning, Z., and Dai, S. (2020). The Electroviscous Flow of Non-newtonian Fluids in Microtubes and Implications for Nonlinear Flow in Porous Media. *J. Hydrology* 590, 125224. doi:10.1016/j.jhydrol.2020.125224
- Dai, H., and MacBeth, C. (1997). Effects of Learning Parameters on Learning Procedure and Performance of a BPNN. *Neural Netw.* 10 (8), 1505–1521. doi:10.1016/S0893-6080(97)00014-2
- Fan, H.-Y., Lu, W.-z., Xi, G., and Wang, S.-j. (2003). An Improved Neural-Network-Based Calibration Method for Aerodynamic Pressure Probes. *J. Fluids Eng.* 125 (1), 113–120. doi:10.1115/1.1523063
- Günther, D., Hanzer, F., Warscher, M., Essery, R., and Strasser, U. (2020). Including Parameter Uncertainty in an Intercomparison of Physically-Based Snow Models. *Front. Earth Sci.* 8, 498. doi:10.3389/feart.2020.542599
- Guo, Y., Cao, X., Liu, B., and Gao, M. (2020). Solving Partial Differential Equations Using Deep Learning and Physical Constraints. *Appl. Sci.* 10 (17), 5917. doi:10.3390/app10175917
- Haryanto, I., Utomo, T. S., Sinaga, N., Rosalia, C. A., and Putra, A. P. (2014). Optimization of Maximum Lift to Drag Ratio on Airfoil Design Based on Artificial Neural Network Utilizing Genetic Algorithm. *Amm* 493, 123–128. doi:10.4028/www.scientific.net/AMM.493.123
- Huang, L., Fu, Q., He, M., Jiang, D., and Hao, Z. (2021). Detection Algorithm of Safety Helmet Wearing Based on Deep Learning. *Concurr. Comput. Pract. Exper* 33 (13), 1–14. doi:10.1002/cpe.6234
- Jiang, D., Li, G., Sun, Y., Hu, J., Yun, J., and Liu, Y. (2021). Manipulator Grabbing Position Detection with Information Fusion of Color Image and Depth Image Using Deep Learning. *J. Ambient. Intell. Hum. Comput.* 12 (12), 10809–10822. doi:10.1007/s12652-020-02843-w
- Jiang, D., Li, G., Sun, Y., Kong, J., Tao, B., and Chen, D. (2019). Grip Strength Forecast and Rehabilitative Guidance Based on Adaptive Neural Fuzzy Inference System Using sEMG. *Pers. Ubiquit Comput.*, 1–10. doi:10.1007/s00779-019-01268-3
- Jiang, D., Li, G., Tan, C., Huang, L., Sun, Y., and Kong, J. (2021). Semantic Segmentation for Multiscale Target Based on Object Recognition Using the Improved Faster-RCNN Model. *Future Gener. Comput. Syst.* 123, 94–104. doi:10.1016/j.FUTURE.2021.04.019
- Jiang, D., Zheng, Z., Li, G., Sun, Y., Kong, J., and Jiang, G. (2019). Gesture Recognition Based on Binocular Vision. *Clust. Comput.* 22, 13261–13271. doi:10.1007/s10586-018-1844-5
- Jin, X., Cheng, P., Chen, W., and Li, H. (2018). Prediction Model of Velocity Field Around Circular Cylinder over Various Reynolds Numbers by Fusion Convolutional Neural Networks Based on Pressure on the Cylinder. *Phys. Fluids* 30 (4), 047105. doi:10.1063/1.5024595
- Kang, S., and Cho, S. (2014). Approximating Support Vector Machine with Artificial Neural Network for Fast Prediction. *Expert Syst. Appl.* 41 (10), 4989–4995. doi:10.1016/j.eswa.2014.02.025
- Karpatne, A., Atluri, G., Faghmous, J. H., Steinbach, M. S., Banerjee, A., Ganguly, A. R., et al. (2017). Theory-guided Data Science: A New Paradigm for Scientific Discovery from Data. *IEEE Trans. Knowl. data Eng.* 29 (10), 2318–2331. doi:10.1109/TKDE.2017.2720168
- Kelley, D. R., Snoek, J., and Rinn, J. L. (2016). Basset: Learning the Regulatory Code of the Accessible Genome with Deep Convolutional Neural Networks. *Genome Res.* 26 (7), 990–999. doi:10.1101/gr.200535.115
- Khaleghi, M., Salimi, J., Farhangi, V., Moradi, M. J., and Karakouzian, M. (2021). Evaluating the Behaviour of Centrally Perforated Unreinforced Masonry Walls: Applications of Numerical Analysis, Machine Learning, and Stochastic Methods. *Ain Shams Eng. J.* 13 (3), 101631. doi:10.1016/j.asej.2021.10.026
- Kim, S., Cho, S., Uhm, K., Ji, S., Lee, S., and Ko, S. (2019). “Evaluating Parameterization Methods for Convolutional Neural Network (CNN)-Based Image Operators.” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 1862–1870. doi:10.1109/CVPRW.2019.00237
- Li, K., and Peng, J. (2007). Neural Input Selection - A Fast Model-Based Approach. *Neurocomputing* 70 (4-6), 762–769. doi:10.1016/j.neucom.2006.10.011
- Li, Y., Ge, D., Gu, J., Xu, F., Zhu, Q., and Lu, C. (2019). A Large Cohort Study Identifying a Novel Prognosis Prediction Model for Lung Adenocarcinoma through Machine Learning Strategies. *BMC cancer* 19 (1), 1–14. doi:10.1186/s12885-019-6101-7
- Liu, J., Hu, Y., He, Y., Chan, P. W., and Lai, L. K. (2015). *Deep Neural Network Modeling for Big Data Weather Forecasting. Information Granularity, Big Data, and Computational Intelligence*. Springer, 389–408. doi:10.1007/978-3-319-08254-7_19
- Liu, X., Jiang, D., Tao, B., Jiang, G., Sun, Y., Kong, J., et al. (2022). Genetic Algorithm-Based Trajectory Optimization for Digital Twin Robots. *Front. Bioeng. Biotechnol.* 9, 793782. doi:10.3389/fbioe.2021.793782
- Liu, Y., Jiang, D., Tao, B., Qi, J., Jiang, G., Yun, J., et al. (2022). Grasping Posture of Humanoid Manipulator Based on Target Shape Analysis and Force Closure. *Alexandria Eng. J.* 61 (5), 3959–3969. doi:10.1016/j.aej.2021.09.017
- Liu, Y., Jiang, D., Yun, J., Sun, Y., Li, C., Jiang, G., et al. (2022). Self-tuning Control of Manipulator Positioning Based on Fuzzy PID and PSO Algorithm. *Front. Bioeng. Biotechnol.*, 9. doi:10.3389/fbioe.2021.817723
- Omran Almagrabi, A., Daanial Khan, Y., and Afzal Khan, S. (2021). iPhosD-PseAAC: Identification of Phosphoaspartate Sites in Proteins Using Statistical Moments and PseAAC. *BIOCELL* 45 (5), 1287–1298. doi:10.32604/biocell.2021.013770
- Peng, J., Chen, S., Aubry, N., Chen, Z., and Wu, W. (2020). Time-variant Prediction of Flow over an Airfoil Using Deep Neural Network. *Phys. Fluids* 32 (12), 123602. doi:10.1063/5.0022222
- Rediniotis, O. K., and Vijayagopal, R. (1999). Miniature Multihole Pressure Probes and Their Neural-Network-Based Calibration. *AIAA J.* 37 (6), 666–674. doi:10.2514/2.79010.2514/3.14225
- Sekar, V., Zhang, M., Shu, C., and Khoo, B. C. (2019). Inverse Design of Airfoil Using a Deep Convolutional Neural Network. *AIAA J.* 57 (3), 1–11. doi:10.2514/1.J057894
- Shen, H., Ren, S., Wang, W., Zhang, C., Hao, H., Shen, Q., et al. (2020). Profiles of Immune Status and Related Pathways in Sepsis: Evidence Based on GEO and Bioinformatics. *BIOCELL* 44 (4), 583–589. doi:10.32604/biocell.2020.011345
- Sun, Y., Zhao, Z., Jiang, D., Tong, X., Tao, B., Jiang, G., et al. (2022). Low-illumination Image Enhancement Algorithm Based on Improved Multi-Scale Retinex and ABC Algorithm Optimization. *Front. Bioeng. Biotechnol.*, 10. doi:10.3389/fbioe.2022.865820
- Tang, H., Rabault, J., Kuhnle, A., Wang, Y., and Wang, T. (2020). Robust Active Flow Control over a Range of Reynolds Numbers Using an Artificial Neural Network Trained through Deep Reinforcement Learning. *Phys. Fluids* (5), 32. doi:10.1063/5.0006492
- Templeton, J. A. (2015). Evaluation of Machine Learning Algorithms for Prediction of Regions of High Reynolds Averaged Navier Stokes Uncertainty. *Phys. Fluids* 27 (8), 085103. doi:10.1063/1.4927765
- Teng, Q., Wang, J., Ding, Z., Zhang, L., and Wang, Z. (2020). Data Driven Governing Equations Approximations Using Attention Based Multistep Neural Networks. *AIP Adv.* 10 (6), 065227. doi:10.1063/5.0015600
- Valsamis, A., Tserpes, K., Zissis, D., Anagnostopoulos, D., and Varvarigou, T. A. (2017). Employing Traditional Machine Learning Algorithms for Big Data Streams Analysis: The Case of Object Trajectory Prediction. *J. Syst. Softw.* 127, 249–257. doi:10.1016/j.jss.2016.06.016
- Wang, F., Bao, W., Wang, Y., Wang, X., Ren, K., Wang, Z., et al. (2020). Ejection State Prediction for a Pneumatic Micro-droplet Generator by BP Neural Networks. *J. Adv. Mech. Des. Syst. Manuf.* 14 (1). doi:10.1299/jamdsm.2020jamdsm0001
- Wang, X., Yamamoto, T., Kadowaki, M., and Yang, Y. (2021). Identification of Key Pathways and Gene Expression in the Activation of Mast Cells via Calcium Flux Using Bioinformatics Analysis. *BIOCELL* 45 (2), 395–415. doi:10.32604/BIOCELL.2021.012280
- Wang, Y., and Cao, Y. (2019). Human Peripheral Blood Leukocyte Classification Method Based on Convolutional Neural Network and Data Augmentation. *Med. Phys.* 47 (1), 142–151. doi:10.1002/mp.13904
- Willard, J. D., Jia, X., Xu, S., Steinbach, M. S., and Kumar, V. (2022). *Integrating Scientific Knowledge with Machine Learning for Engineering and Environmental Systems*. New York: ACM Computing Surveys. doi:10.1145/3514228
- Wong, E. W., and Kim, D. K. (2018). A Simplified Method to Predict Fatigue Damage of TTR Subjected to Short-Term VIV Using Artificial Neural Network. *Adv. Eng. Softw.* 126, 100–109. doi:10.1016/j.advengsoft.2018.09.011
- Wu, X., Jiang, D., Yun, J., Liu, X., Sun, Y., Tao, B., et al. (2022). Attitude Stabilization Control of Autonomous Underwater Vehicle Based on Decoupling Algorithm and PSO-ADRC. *Front. Bioeng. Biotechnol.* 10, 843020. doi:10.3389/fbioe.2022.843020
- Xue, L. C., Dobbs, D., Bonvin, A. M., and Honavar, V. G. (2015). Computational Prediction of Protein Interfaces: A Review of Data Driven Methods. *FEBS Lett.* 589 (23), 3516–3526. doi:10.1016/j.febslet.2015.10.003

- Yan, B., Chen, B., Harp, D. R., and Pawar, R. (2020). *A Physics-Constrained Deep Learning Model for Simulating Multiphase Flow in Fully Three-Dimensional Heterogeneous Reservoirs*. New Orleans: AGU Fall Meeting Abstracts.
- Yong, Z., Yang, M., Pengyao, Y., Tianlin, W., and Shaojuan, S. (2020). Prediction of Fluid Force Exerted on Bluff Body by Neural Network Method. *J. Shanghai Jiaot. Univ. Sci.* 25 (6), 186–192. doi:10.1007/S12204-019-2140-0
- Yun, J., Sun, Y., Li, C., Jiang, D., Tao, B., Li, G., et al. (2022). Self-adjusting Force/bit Blending Control Based on Quantitative Factor-Scale Factor Fuzzy-PID Bit Control. *Alexandria Eng. J.* 61 (6), 4389–4397. doi:10.1016/j.aej.2021.09.067
- Zhang, Y., Sung, W., and Mavris, D. N. (2018). *AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*. doi:10.2514/6.2018-1903 Application of Convolutional Neural Network to Predict Airfoil Lift Coefficient
- Zhao, G., Jiang, D., Liu, X., Tong, X., Sun, Y., Tao, B., et al. (2022). A Tandem Robotic Arm Inverse Kinematic Solution Based on an Improved Particle Swarm Algorithm. *Front. Bioeng. Biotechnol.* doi:10.3389/fbioe.2022.832829

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Chi, Yang and Liang. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.