International Journal of Heat and Mass Transfer 194 (2022) 123072

Contents lists available at ScienceDirect

International Journal of Heat and Mass Transfer

journal homepage: www.elsevier.com/locate/hmt

An investigation of the thermal behavior of constructal theory-based pore-scale porous media by using a combination of computational fluid dynamics and machine learning

Mehrdad Mesgarpour[a,*], Kittipong Sakamatapan[a], Ahmet Selim Dalkılıç[b], Rasool Alizadeh[c], Ho Seon Ahn[d], Somchai Wongwises[a,e,*]

[a] Fluid Mechanics, Thermal Engineering and Multiphase Flow Research Lab. (FUTURE), Department of Mechanical Engineering, Faculty of Engineering, King Mongkut's University of Technology Thonburi (KMUTT), Bangmod, Bangkok 10140, Thailand
[b] Department of Mechanical Engineering, Yildiz Technical University, Yildiz, Besiktas, Istanbul, Turkey
[c] Department of Mechanical Engineering, Quchan Branch, Islamic Azad University, Quchan, Iran
[d] Department of Mechanical Engineering, Incheon National University, Incheon, Republic of Korea
[e] National Science and Technology Development Agency (NSTDA), Pathum Thani 12120, Thailand

ARTICLE INFO

Article history:
Received 6 January 2022
Revised 24 April 2022
Accepted 22 May 2022

Keywords:
Constructal theory
Block-based machine learning
Heat transfer
Optimization
Pore-scale porous media

ABSTRACT

The present study investigates the flow pattern and thermal behavior of constructal theory-based pore-scale porous media (CTPSPM) using computational fluid dynamics (CFD) and machine learning. Due to the increasing complexity of the geometry, as the number of pairs (Np) and clusters (Nc) increase, the CFD approach is unable to present the precise results. For the first time, we present a novel hybrid computational method to predict the flow pattern and thermal behavior of pore-scale porous media (PSPM) based on the direct training data set from high-fidelity numerical simulation. Tensors of component velocity, temperature for fluid and solid, and pressure drop will be used to update the training dataset in accordance with the geometry matrix. A tensor-based combination of CFD and block-based machine learning (BBML) is developed for this target due to the complex interface between flow and solid. An elliptical tube is filled with two distinct types of PSPM: CTPSPM and constant-porosity PSPM (CPPSPM) with different porosity ratios (β=0.6, 0.8, 1.2, 1.4). A laminar flow (Re=300, 500, and 800) of Al_2O_3 nanofluids through an elliptical tube in constant volume fractions (φ=0.06) is studied in both CFD and BBML methods. The effects of various combinations of CTPSPM pairs and clusters on pressure drop and heat transfer are evaluated using machine learning in over 1200 states of calculation to determine the optimal configuration. Additionally, we evaluated the effect of multiple cores on calculation time in three distinct optimizations (machine learning, surface response method, and genetic algorithm) approaches in CTPSPM for the first time. The results show that the multiblock neural network could reduce the computational cost by up to 70% compared with the regular CFD. In addition, the results show that the constructal theory significantly influences heat transfer in the low Re range. The results of this study could lead to a new understanding of fluid flow due to complex geometry, such as that of a catalyst and membrane.

Published by Elsevier Ltd.

1. Introduction

Optimization is important in all aspects of life, from energy consumption [1] to industry [2]. Optimization methods can also yield the optimum state, such as maximum heat transfer with a minimum drop in pressure [3]. Process optimization aims to maximize efficiency while minimizing undesirable effects, such as drag force [4] and pressure drop [5]. In recent decades, energy requirements have increased significantly, prompting the need for a new generation of standards that place a premium on energy efficiency [6]. Numerous algorithms for mathematical optimization have been used over the last several decades to maximize system efficiency [7]. Researchers have attempted to minimize the drop in pressure using various optimization techniques. They have tried to determine the optimal geometry by analyzing the interaction between shape and fluid flow (see, for example, Weber et al. [8] and Shalaby et al. [9]). The most popular optimization methods include

* Corresponding authors at: Fluid Mechanics, Thermal Engineering and Multiphase Flow Research Lab. (FUTURE), Department of Mechanical Engineering, Faculty of Engineering, King Mongkut's University of Technology Thonburi (KMUTT), Bangmod, Bangkok 10140, Thailand.
E-mail addresses: Mehrdad.mes@kmutt.ac.th (M. Mesgarpour), Somchai.won@kmutt.ac.th (S. Wongwises).

https://doi.org/10.1016/j.ijheatmasstransfer.2022.123072
0017-9310/Published by Elsevier Ltd.

Nomenclature

A	Surface area, [m^2]
$A_1, A_2, A_3\ A_4, A_5$	Constant
Bi	Biot number
C	Heat capacity [$J.K^{-1}$]
C_P	Specific heat, [$J.kg^{-1}.K^{-1}$]
D	Diameter, [m]
D_0	Thickness of the central strip, [m]
Da	Darcy number
D_h	Hydraulic diameter, [m]
E_f	Total fluid energy, [J]
f_j	Values of j-th neuron
g	Gravity, [$m.s^{-2}$]
g^*	Activation function
H	Height, [m]
$\tilde{H}_0$	Dimensionless height
h_{sf}	Convection heat transfer coefficient, surface to fluid, [$W.m^{-2}.K^{-1}$]
h_{nf}	Convection heat transfer coefficient, [$W.m^{-2}.K^{-1}$]
K	Permeability, [m^2]
$\tilde{K}_0$	Dimensionless permeability
k_{nf}	Thermal conductivity of nanofluid, [$W.m^{-1}.K^{-1}$]
k_f	Thermal conductivity of water, [$W.m^{-1}.K^{-1}$]
k_s	Thermal conductivity of solid, [$W.m^{-1}.K^{-1}$]
L	Length, [m]
Nu	Nusselt number
Nu^*	Nusselt number ratio
Nu_{CTPSPM}	Nusselt number of constructal theory pore-scale porous media
Nu_{CPPSPM}	Nusselt number of constant porosity pore-scale porous media
n^*	Sample size
O_i	Model output
P	Pressure, [Pa]
P_{CTPSPM}	Pressure drop of constructal theory pore-scale porous media, [Pa]
P_{CPPSPM}	Pressure drop of constant porosity pore-scale porous media, [Pa]
Pr	Prandtl number
q_s''	Heat flux for solid zone, [$W.m^{-2}$]
q''	Heat flux, [$W.m^{-2}$]
q_{nf}''	Heat flux of nanofluid, [$W.m^{-2}$]
Ri	Richardson number
S_f^h	Fluid enthalpy, [J]
T_{max}	Maximum temperature, [K]
T_w	Wall temperature, [K]
T_s	Temperature of solid, [K]
T_i	Target value for sample i
Re	Reynolds number
Ra	Rayleigh number
T_{nf}	Temperature of nanofluid, [K]
T_i	Initial temperature, [K]
V_{duct}	Total duct volume, [m^3]
u, v, w	x,y,z velocity, [$m.s^{-1}$]
w_i	Weight of i-th neuron of hidden layer
$w2_{0,z}$	Weights of the bias
$w2_{j,z}$	Weights of the j-th hidden neuron
x_i	Values of i-th neuron
x, y, z	x,y and z-direction
$y_z(ML)$	Output of z−th neuron
Z	z-direction

Greek symbols

ρ_f	Density of pure water, [$kg.m^{-3}$]
ρ	Density, [$kg.m^{-3}$]
ρ_{nf}	Density of nanofluid, [$kg.m^{-3}$]
η_p	Pump efficiency
$\beta^*{}_f$	Thermal expansion of fluid, [K^{-1}]
$\beta^*{}_s$	Thermal expansion of solid, [K^{-1}]
β^*	Thermal expansion, [K^{-1}]
β	Porosity ratio
ε	Porosity
υ	Kinematic viscosity, [$m^2.s^{-1}$]
μ	Dynamic viscosity, [$kg.m^{-1}.s^{-1}$]
μ_{nf}	Viscosity of nanofluid, [$kg.m^{-1}.s^{-1}$]
ε_{CTPSPM}	Porosity of constructal theory pore-scale porous media
ε_{CPPSPM}	Porosity of constant porosity pore-scale porous media
φ	Volume fraction
μ_i	Center of i-th neuron
α_{angle}, β_{angle}	Angle, [degree]
$\phi_i(X^{**})$	Gaussian function
σ_i^2	Radius of i-th neuron
θ	Dimensionless temperature
θ_{nf}	Dimensionless temperature for nanofluid

Abbreviation

BBML	Block-based machine learning
CT	Constructal theory
CTPSPM	Constructal-theory pore-scale porous media
CPPSPM	Constant porosity pore-scale porous media
CFD	Computational fluid dynamics
DDTA	Direct data transfer algorithm
GPU	Graphical process unit
PSPM	Pore-scale porous media
Np	Number of pairs
Nc	Number of clusters
ML	Machine learning
MSE	Mean square error

mathematical modeling [10] and thermodynamic approaches [11,12]. A numerical approach can help optimize by identifying the optimal combination of parameters, such as the volume fraction, porosity, and types of nanofluids. Thermodynamics-based methods of optimization, such as entropy generation minimization (EGM) [13] and constructal theory (CT) [14], are multidimensional techniques that can be applied to various cases [15].

Understanding flow behavior through the pore-scale porous media (PSPM) has a long history [16]. Mou and Chen [17] analyzed the fluid transport in PSPM using the lattice Boltzmann method (LBM) and direct numerical simulation (DNS). Results indicate that both methods are capable of modeling the flow through PSPM. Habib et al. [18] used the transient form of the Navier-Stokes equation to study the cylindrical shape of PSPM. They found that variations in the Reynolds number (Re) and porosity of PSPM could cause the system's dynamic response to become nonlinear. Bakhshian et al. [19] studied two-phase flow in a heterogeneous rock sample, revealing that the viscosity ratio is an important parameter for flow through the rock.

The new generation of manufacturing processes and materials, such as nanofluids [20] or porous media [21], has opened a new window to improve efficiency and heat transfer, particularly in thermal devices [22]. Porous media are among the most promising candidates for improving heat transfer due to their unique

surface-area-to-volume ratio [21]. Hatami and Ganji [23] focused on heat transfer in microchannels and analyzed heat transfer using a porous medium. Their results showed that when the channel aspect ratio (AS) increased, the Nusselt number (Nu) also increased, owing to more significant interactions between fluid flow and the porous media surface. Bourantas et al. [24] studied a meshless technique to examine heat transfer in porous media by using Darcy models. Their results indicated that at low Rayleigh numbers (Ra = 103), the average Nu appeared to be sensitive to the presence of a porous medium. Sheikholeslami [25] investigated heat transfer in porous media by using various boundary conditions, including the volume fraction of the nanofluid, and found that force convection-induced heat transfer within a lid-driven porous cavity could increase the rate of heat transfer.

The CT is one of the most interesting nature-based optimization methods [26]. Configurations that use CT introduce a groundbreaking physics-based approach to comprehending and predicting designs found in nature and engineering [27]. This approach can yield the optimum rate of heat transfer (see, for example, Wu et al. [28], Tang et al. [29], and Dehaj and Hajabdollahi [30]). Mehrgoo and Amidpour [31] examined the effects of the CT on the operational and geometric design parameters of heat recovery in a steam generator. Their findings show that increasing the total value to an optimal point significantly reduces entropy generation. After a specific point, the curve of entropy reduction decreases and exergy-induced destruction becomes flat. Wu et al. [32] investigated the effect of CT on the design of an evaporator. The results showed that the dimensionless pumping power decreased by 37.1% after CT-based optimization compared with the design point at which the optimization was performed.

Redesigning equipment and applying the CT to a new generation of thermal systems necessitates a significant amount of calculation. When the number of pairs (Np) and clusters (Nc) is increased, the required grid size and calculation cost increase rapidly. As a result of this issue, the majority of CT geometries investigated have the lowest possible Np and Nc [33]. The development of a new generation of computational methods and algorithms necessitates enormous processing power. These novel computational tools may aid in our comprehension of this complicated process. Neural networks are the best example of how these new calculation tools have become widespread, leading to the birth of a new generation of computational tools that can assist with complex calculations and analyzes [34]. A combination of advanced numerical simulations and neural networks can assist in the comprehension of complex events, such as virus outbreaks [35] and combustion [36]. The ability to predict complex flow patterns [37] or a wide range of geometries [38] using real-time data is the primary reason for using a combination of neural networks and numerical simulations.

A novel solution is presented in this paper to study flow through complex geometries such as PSPMs, including laminar flows of AL_2O_3 nanofluids (Re=300, 500, 800) with constant volume fractions (φ= 0.06) through the constant-porosity PSPM (CPPSPM) and CT-based PSPM (CTPSPM). Additionally, 4 porosity ratios (β=0.6, 0.8, 1.2, and 1.2) are assumed for CTPSPM. The results of high-fidelity CFD are used to generate training tensors in this method. After thorough validation, separate tensors for velocity component, the temperature of fluid and solid (PSPM), and pressure are defined for training feed-forward neural networks (FNN) by taking geometry into account. In the training section, we trained the FNN model on all calculation domains for various Re and porosity ratios at specific Np (Np=4 and 7) and Nc (Nc=2). For other Nc and Nc values (1<Np<3, 4<Np<6, 8<Np<10, and Nc=1 and 3<Nc<10), machine learning will predict the flow and heat transfer patterns through the CTPSPM and PSPM. A tensor-based combination of block-based machine learning (BBML) and numerical simulation predicted the nanofluid flow's pattern and thermal behavior through the elliptical tube. Additionally, we investigate the effect of the number of GPU cores on calculation time and optimization methods for the first time. We demonstrate how this solution predicts the pattern of heat transfer and pressure drop for a wide range of the numbers of pairs (1<Np<3, 4<Np<6, 8<Np<10) and clusters (Nc=1 and 3<Nc<10) and study the calculation cost of Constructal theory in PSPM domain. This solution provides a new way of understanding flow through complex geometries such as PSPM or membranes.

2. Problem statement

A steady-state Al_2O_3 nanofluid flowing through an elliptical tube filled with CPPSPM and CTPSPM is used to investigate the effect of CT on the thermal behavior and flow pattern in the PSPM domain. As shown in **Fig. 1a**, the CTPSPM includes clusters and pairs. The porosity (ε) of CPPSPM is 80%, and pores are 2 mm in diameter on average. The porosity distribution and other properties are constant and uniform for CPPSPM and CTPSPM. The porosity and pore size of CTPSPM depend on β. A numerical simulation for Nc=4 and 7 is considered to provide training data set for component velocity, temperature for fluid, and solid and pressure. These two calculation points are the foundation for the wide variation in Np (1<Np<3, 4<Np<6, 8<Np<10) and Nc (Nc=1 and 3<Nc<10). **Fig. 1b** shows the CTPSPM and copper tube dimensions. The CTPSPM is composed of copper with a uniform porosity distribution.

Table 1 provides additional information about porosity, permeability, and thermal expansion. Constant volume fractions of Al_2O_3 nanofluid (φ= 0.06) are flows through the CPPSPM and CTPSPM at different Re ranges (Re = 300, 500, and 800). As shown in **Fig. 1c** and **1d**, steady-state and fully developed Al_2O_3 nanofluids are introduced into the tubes for the CTPSPM and CPPSPM, respectively. The CTPSPM and tubes shared the same physical characteristics, including thermal expansion and conductivity. The outer walls of the tube are maintained at a constant temperature (T = 350 K) to impose thermal boundary conditions. Additional details about the PSPM are presented in **Appendix A1.**

2.1. Block-based machine learning

To combine numerical simulation and machine learning, it is essential to understand several key concepts in neural networks. A training data set is the basis for machine-learning-based prediction. The training data set for this study is generated using the results of a numerical simulation. Each parameter is assigned as an individual data matrix, such as pressure, velocity, fluid, or solid. The numerical simulation results in **Fig. 2** are assumed to be a series of inputs for a neural network. Six categories of numerical results are used to re-generate the three main matrices (pressure, component velocity, and temperature for fluid and solid). These matrices define the values of parameters in each calculation node.

As a result of the case study's, these matrices are fed into two layers of deep neural networks (CTPSPM and CPPSPM) to extract hidden relationships among inputs and outputs. Study outputs include pressure drop, heat transfer coefficient, and numeric values such as temperature distribution for the CTPSPM and CPPSPM. Activation and loss functions are calculated based on each parameter of these two layers. In addition to the regression model and the number of hidden layers, each prediction layer has its unique characteristics.

Table 2 provides further information about these two prediction layers. The output of the prediction layer is used as an input for the integration layer. Therefore, output patterns require an additional layer in order to be precise. A single calculation layer or multiple calculations layers are used [34]. Every data block (prediction layer) may have a pattern between the input and output.

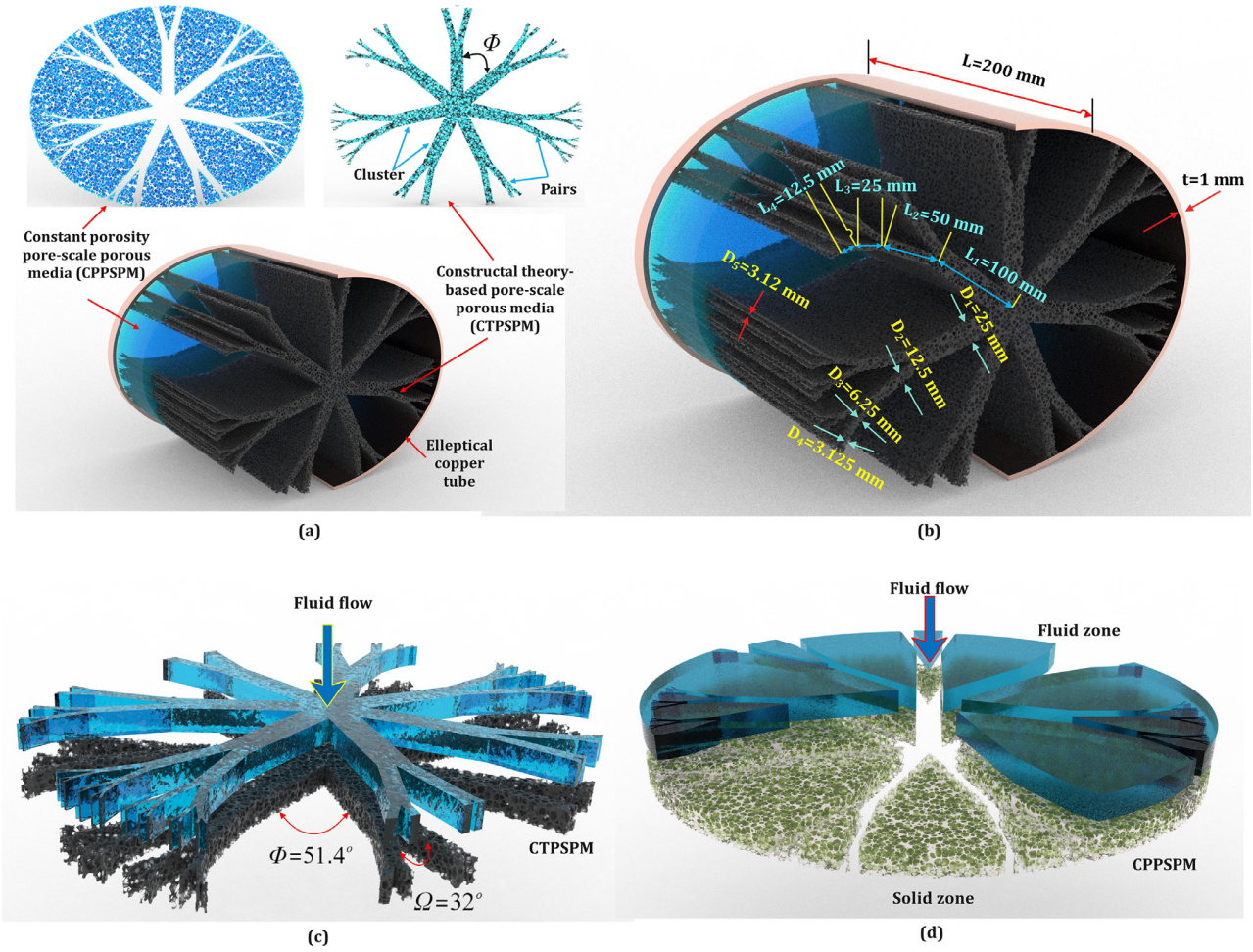


Fig. 1. Graphical illustration of the CTPSPM and CPPSPM a) Isometric view of CTPSPM, CPPSPM, and tube, b) Dimension of pairs and clusters, and fluid and solid zones in c) CTPSPM and d)CPPSPM.

Table 1
Physical properties of the CTPSPM and CPPSPM.

Zone	Porosity [%]	$K_{permability}$ $[m^2]$	Re	T_{tube}[K]	k_{porous} $[W.m^{-1}K^{-1}]$	β^* $[K^{-1}]$	Ref.
CPPSPM	80	5.23×10^{-10}	300 500 800	350	300	1.65×10^{-5}	[39]
CTPSPM	85 90 95 98	6.27×10^{-10}					

Table 2
Details of neural network for CTPSPM and CPPSPM zones.

Layer	Number of hidden layers	Regression method	Number of training data set	Number of validations	R^2	Mean average error (MAE)
CPPSPM	86	MLP	3897,411	536,397	0.971	0.045
CTPSPM	112	SVR	4289,278	364,871	0.985	0.087

As a result of the nature of this study, an integration layer is necessary to reconcile the results of a series of tensors. This study introduces a BBML technique on the layers of the data block and serial calculation. When applied to complex geometry, the blocking approach can form a distinct and independent prediction layer. Additionally, the optimal number of training data depends on the individual blocks for each category. In other words, a lack of information about specific parameters in the training data set can affect output. This method concentrates error distribution on a single block. As a result, this method can compensate for any gaps in the training data set. Additionally, BBML improves the results of loss functions for each parameter by reducing the number of possible values. Prediction of quality is contingent upon the precision of the loss function. As a result, BBML aids in refining the loss function for the parameter. The primary advantage of the BBML method is that it enables the definition of neural networks with multiple hidden layers. The flow through the PSPM is a combination of fluid-solid interaction and solid-state heat transfer. Due to the complex geometry of the PSPM, variations in the fluid parameters necessitate the addition of a hidden layer to establish relationships between the parameters.

Table 3 shows the details of the solution and describes the simulation procedure.

The PSPM geometry was generated in Python using PoreSpy (https://github.com/PMEAL/porespy). Numerical simulation was based on the SIMPLE algorithm. This study developed BBML in GPU-based Python code for three calculation blocks (https://github.com/babrodtk/conslaws_python_opencl). All codes in the

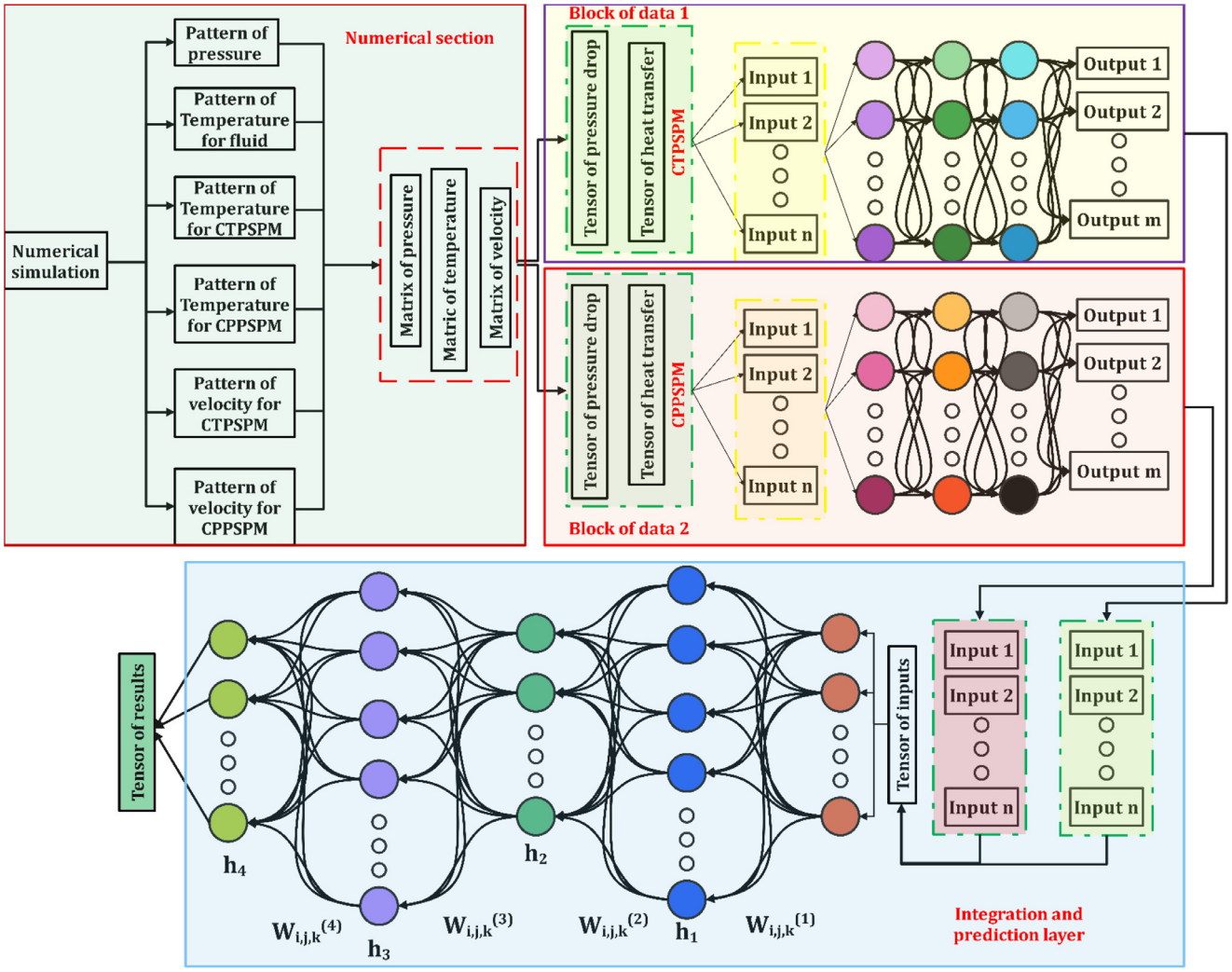


Fig. 2. Data flow diagram of numerical simulation, blocks of data, training data set, and neural network.

Table 3
Details of the software.

Topic	Target	Software or code
Geometry re-generation	Generating a new geometry based on new Np and Nc	Python 3.9.0
Grid generation	Grid generation on new geometry	ICEM/ Python 3.9.0
Simulation	Solveing Navier-Stokes equation	Python 3.9.0
Data transfer and analysis	Data transfer (real-time)	Pyansys 0.60/ Numpy
Neural	Training	Tensorflow/ Python 3.9.0
network	Prediction	Tensorflow/ Python 3.9.0

numerical section were developed based on second-order accuracy. Data transfer between numerical section and machine learning is based on tensor-based data transfer (https://github.com/pytorch). Multilayer perceptron (MLP) was used as the regression algorithm (https://github.com/filipecalasans/mlp). TensorFlow was used in this study's machine -learning section (https://github.com/tensorflow/tensorflow). All codes were developed using an Nvidia RTX 8000 graphics card and a 12-core CPU (Xeon E5–2630).

3. Governing equations

A steady-state form of continuity equation is used to satisfy the Navier-Stokes equation [18].

$$\frac{\partial(\rho u_x)}{\partial x}+\frac{\partial(\rho u_y)}{\partial y}+\frac{\partial(\rho u_z)}{\partial z}=0. \tag{1}$$

The momentum equation is required to describe the fluid flow through the PSPM. The steady-state form of momentum equation is defined as [18]:

$$\rho\left(u_x\frac{\partial(u_x)}{\partial x}+u_y\frac{\partial(u_x)}{\partial y}+u_z\frac{\partial(u_x)}{\partial z}\right)=-\frac{\partial p}{\partial x}$$
$$-\frac{\partial}{\partial x}\left(-2\mu\frac{\partial(u_x)}{\partial x}+\frac{2}{3}\mu\left(\frac{\partial(u_x)}{\partial x}+\frac{\partial(u_y)}{\partial y}+\frac{\partial(u_z)}{\partial z}\right)\right)$$
$$-\frac{\partial}{\partial y}\left(-\mu\left(\frac{\partial(u_x)}{\partial y}+\frac{\partial(u_y)}{\partial x}\right)\right)-\frac{\partial}{\partial z}\left(-\mu\left(\frac{\partial(u_x)}{\partial z}+\frac{\partial(u_z)}{\partial x}\right)\right), \tag{2}$$

$$\rho\left(u_x\frac{\partial(u_y)}{\partial x}+u_y\frac{\partial(u_y)}{\partial y}+u_z\frac{\partial(u_y)}{\partial z}\right)$$
$$=-\frac{\partial p}{\partial y}-\frac{\partial}{\partial x}\left(-\mu\left(\frac{\partial(u_x)}{\partial y}+\frac{\partial(u_y)}{\partial x}\right)\right)$$
$$-\frac{\partial}{\partial y}\left(-2\mu\frac{\partial(u_y)}{\partial y}+\frac{2}{3}\mu\left(\frac{\partial(u_x)}{\partial x}+\frac{\partial(u_y)}{\partial y}+\frac{\partial(u_z)}{\partial z}\right)\right)$$
$$-\frac{\partial}{\partial z}\left(-\mu\left(\frac{\partial(u_y)}{\partial z}+\frac{\partial(u_z)}{\partial y}\right)\right), \tag{3}$$

$$\rho\left(u_x\frac{\partial(u_z)}{\partial x}+u_y\frac{\partial(u_z)}{\partial y}+u_z\frac{\partial(u_z)}{\partial z}\right)$$
$$=-\frac{\partial p}{\partial z}-\frac{\partial}{\partial x}\left(-\mu\left(\frac{\partial(u_x)}{\partial z}+\frac{\partial(u_z)}{\partial y}\right)\right)$$
$$-\frac{\partial}{\partial y}\left(-2\mu\frac{\partial(u_z)}{\partial z}+\frac{2}{3}\mu\left(\frac{\partial(u_x)}{\partial x}+\frac{\partial(u_y)}{\partial y}+\frac{\partial(u_z)}{\partial z}\right)\right). \tag{4}$$

where ρ is fluid density, p is pressure, and μ is viscosity. The energy equation for fluid through the PSPM is defined as [18]:

$$\rho_w c_p\left(u_x\frac{\partial T}{\partial x}+u_y\frac{\partial T}{\partial y}+u_z\frac{\partial T}{\partial z}\right)$$
$$=\frac{\partial}{\partial x}\left(k_{nf}\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial y}\left(k_{nf}\frac{\partial T}{\partial y}\right)+\frac{\partial}{\partial z}\left(k_{nf}\frac{\partial T}{\partial z}\right). \tag{5}$$

According to Newton's law of cooling, heat transfer by forced convection within the PSPM is defined as [40]:

$$q_f^{//}=h_{nf}\left(T_{nf}-T_s\right). \tag{6}$$

$$h_{nf}=\frac{Q}{T_w-T_i}. \tag{7}$$

where h_{nf} is the convection heat transfer coefficient for nanofluid. The following equation describes heat conduction in a PSPM [41]:

$$\nabla.[k_s\nabla T_{solid}]=0. \tag{8}$$

where k_s is thermal conductivity of PSPM. The dimensionless numbers are expressed as:

$$Re=\frac{V_i D_h}{\upsilon_f}, Nu=\frac{q_f^{//}.D_h}{k_{nf}\left(T_s-T_{ref}\right)}, Nu^*=\frac{Nu_{CTPSPM}}{Nu_{CPPSPM}},$$
$$\Delta P^*=\frac{\Delta P_{CTPSPM}}{\Delta P_{CPPSPM}}, \beta=\frac{\varepsilon_{CTPSPM}}{\varepsilon_{CPPSPM}}. \tag{9}$$

The boundary condition in this case study is defined as follows:

- The solid-liquid interfaces boundaries are no-slip, ($u=0$).
- Dirichlet boundary condition for the momentum equation along the boundary.
- Thermal resistance between tube and PSPM is zero.

3.1. Nanofluid properties

The Al_2O_3/water nanofluid is considered the operation fluid. Its thermophysical properties are presented in **Table 4** [42].

3.2. Neural network

3.2.1. Activation function

An activation function (g*) is applied to the weighted sum of the input layer's outputs by the hidden layer's m* + 1 neurons, one of which serves as a bias [44].

$$f_j=g^*\left(w1_{0,j}+\sum_{i=1}^{n}w1_{i,j}\,x_i\right). \tag{10}$$

In this equation, x_i and f_j are the values of the i-th neuron and the j-th neuron, respectively. Neurons are activated (or not) by an activation function [45]. In other words, it uses simpler mathematical calculations to determine how important the neuron's input is during the prediction process. The model's output is generated by activating each neuron in this layer as follows:

$$y_z(M.L)=g^*\left(w2_{0,z}+\sum_{j=1}^{m}w2_{j,z}\,f_j\right). \tag{11}$$

$y_z(ML)$ is the output of the z-th neuron, and $w2_{0,z}$ and $w2_{j,z}$ are the weights of the bias and j-th hidden neuron's connection with the z-th output neuron, respectively.

3.2.2. Loss function

This function reduces the current output of the algorithm to the expected output by determining the loss. Basically, it is a way of determining how the algorithm models the data. A measurement of the mean squared difference between actual and predicted values is known as the mean square error (MSE). A penalty is applied to an excessive difference when the squared difference calculation is performed. The MSE criterion is used as the loss function [46].

$$MSE=\frac{1}{n^*}\sum_{i=1}^{n^*}(O_i-T_i)^2. \tag{12}$$

n^* is the sample size, and O_i and T_i are the model output and target value for sample i. The mean average error (MAE) is defined based on:

$$MAE=\frac{1}{n^{**}}\sum_{i=1}^{n^{**}}|X_i-X|. \tag{13}$$

where n^{**} is the number of errors, and $|X_i-X|$ is absolute errors.

3.2.3. Supervised neural network models

The following examples are provided (x_1,y_1), (x_2,y_2),..., (x_n,y_n), where $x_i\in\mathbb{R}^n$and $y_i\in\{0,1\}$. The MLP learns a function using a hidden layer underneath a hidden neuron $f(x)=W_{2g}(W_1^Tx+b_1)+b_2$, where $W_1\in\mathbb{R}^m$ and $W_2,b_1,b_2\in\mathbb{R}$. W_1, W_2 show the weights of the input layer and the hidden layer, respectively. Bias added to the hidden and output layers is represented by b_1. In this case, $g(.):R\to R$ is the hyperbolic tan activation function [47].

$$g^*(z)=\frac{e^z-e^{-z}}{e^z+e^{-z}}. \tag{14}$$

For binary classification, $f(x)$, a logistic process takes place $g^*(z)=\frac{1}{(1+e^{-z})}$ to obtain output values between 0 and 1. Samples exceeding the threshold of 0.5 are considered positive, while the rest are considered negative. There can be more than two classes, $f(x)$; as a vector, it would have the size (n-classes). In place of passing through a logistic function [48], it passes through the softmax function [49], which is written as:

$$softmax(z)_i=\frac{\exp(z_l)}{\sum_{l=1}^{k}\exp(z_l)}. \tag{15}$$

Table 4
Thermal conductivity and viscosity correlations of nanofluid.

Parameter	Correlation	Ref.
Dynamic viscosity	$\frac{\mu_{nf}}{\mu_w}=1+4.93\varphi+222.4\varphi^2$	[43]
Thermal conductivity	$\frac{k_{nf}}{k_f}=1+2.944\varphi+19.672\varphi^2$	

Table 5
MAE and MSE reports for the first 10 hidden layers.

Neurons	1	2	3	4	5	6	7	8	9	10
MAE	1.539	0.819	0.734	0.509	0.509	0.508	0.508	0.508	0.508	0.508
MSE	3.875	1.019	0.859	0.340	0.340	0.339	0.339	0.339	0.339	0.339

The input to softmax (z_l) is the L-th element, which corresponds to class l, and k is the number of classes. Thus, a vector containing probability values for each class is generated. The class determines the output with the highest probability. The output in regression remains as $f(x)$; as a result, the output activation function is nothing more than the identity function. The MLP employs various loss functions [50]. Cross-entropy [51] is the classification loss function written, as in the binary case,

$$Loss\left(\hat{y}, y, W\right) = \frac{1}{2}\left\|\hat{y} - y\right\|_2^2 + \frac{\alpha}{22}\|W\|_2^2. \tag{16}$$

By updating the weights, the MLP minimizes the loss function [52]. An incremental backward pass propagates each parameter of the weight function from the output layer to the previous layers to reduce the loss. In gradient descent, the gradient $\nabla LossW$ of the loss with respect to the weights is computed and deducted from W. The input parameters can be defined as a vector of real numbers:

$$X^{**} = [x_1\ x_2 \dots\ x_n]. \tag{17}$$

The model output is a scalar function of the input vector and is calculated as follows:

$$y^{**} = \sum_{i=1}^{m} w_i \phi_i(X^{**}). \tag{18}$$

where $\phi_i(X^{**})$ and w_i are the output and the weight of the i-th neuron of the hidden layer, respectively. A weight is assigned to each of the neurons in the hidden layer, which constituted the trainable parameter of the network. The Gaussian function [53] is often used to calculate ϕ_i by the following equation:

$$\phi_i(X^{**}) = \phi(\|X^{**} - \mu_i\|) = \exp\left(-\frac{\|X^{**} - \mu_i\|}{2\sigma_i^2}\right). \tag{19}$$

where μ_i and σ_i are the center and the radius of the i-th neuron, respectively. The MAE and MSE in this study for the first 10 hidden layers are presented in **Table 5**.

3.3. Tensor-based data transfer protocol (TBDTP)

A numerical simulation provides a large number of data values that change over time, such as temperature, velocity, and pressure. Combining these numbers with dimensions and dimensionless numbers can generate an extraordinary amount of data [54]. These data necessitate a variety of statistical methods and a significant amount of time for post-processing. Linear algebraically, a singular value decomposition (SVD) of a matrix corresponds to factorizing that matrix into three matrices [55]. Linear transformations convey significant geometrical and theoretical insights due to their algebraic properties. This technique can also be applied to data science. For matrix M, the SVD is defined as [56]:

$$M = U\Sigma V^T. \tag{20}$$

Matrix M consists of two singular vector spaces, U* and V*. For vector (S) in the L-dimensional tensor $S \in \mathbb{R}^{p_1 \times p_2 \times \dots \times p_L}$, the Kronecker matrix is defined as [57]:

$$Q \otimes W = [a_1 \otimes W \dots a_n \otimes W] = \begin{pmatrix} a_{11}W & \dots & a_{1n}W \\ \vdots & \ddots & \vdots \\ a_{m1}W & \dots & a_{mn}W \end{pmatrix}. \tag{21}$$

Tensor W admits a rank-R decomposition as [55]:

$$W_{(l)} = W_{(l)}(W_L \odot \dots \odot W_{l+1} \odot W_{l-1} \dots \odot W_1)^T. \tag{22}$$

The module is for computing and storing smaller sets of core data from high-dimensional tensorized data. Implementing the IHOSVD algorithm can further enhance the reduction by incrementally updating the orthogonal bases in the unfolded matrices [58].

4. Grid generation and validation

4.1. Grid study

Advanced meshing methods are used to generate various types of meshes (boundary layer grids (BLGs) [59], mosaic grids (MGs) [60], polyhedral grids (PGs) [61], and structural grids (SGs) [62]) based on their locations. **Table 6** shows detailed information on the various types of meshes. The results indicate that grid generation using a variety of mesh types may help reduce skewness. Such distinct features as the thickness of the boundary layer (BL) and interfaces with various types of grids on each side contribute to increasing the accuracy and speed of calculations.

Fig. 3a shows a grid analysis based on a velocity ratio for different values of β. Because the geometry of each sample case varied, a grid study is required for all new simulations. The results indicated that a specific value of grids is required for every β. Skewness analysis in the simulation domain provided insights into the quality of the generated grids. The average skewness value for the simulation domain is shown in **Fig. 3b** for ranges of grids for Nc=7 and 4. The results show that the BLG method can reduce the average skewness.

4.2. Validation

4.2.1. Numerical simulation validation

The simulation is validated using three distinct sources. The pressure drop in laminar flow is shown in **Fig. 4a**. The numerical simulation results for Nc=4 and 7 are validated by Liu et al. [63]. and Ergun equation [64]. In addition, for Nu, the results in **Fig. 4b** show that the maximum error is −11.86% less than that obtained by Haji-Sheikh and Vafai [65].

4.2.2. Neural network sensitivity and validation

The accuracy of the prediction is highly dependent on the quality of the training data set. The BBML method requires individual training data sets for each calculation block. As a result of the complex geometry and the large number of datasets involved in this study, the matrix method is used for preparing training datasets. There are specific numbers of hidden layers in each calculation block. Therefore, the maximum error in the predicted dimensionless velocity is 9.54%, and the maximum error in the temperature is 9.32%. **Table 7** summarizes the errors in predictions of the numerical method and the BBML algorithm. **Appendix A2** (Table A2) presents additional information about the validation of hidden layers in BBML.

Table 6
Details of grid generation.

Zone	Wall					CTPSPM				
β	0.6	0.8	1	1.2	1.4	0.6	0.8	1	1.2	1.4
Skewness	0.13	0.15	0.08	0.11	0.13	0.26	0.21	0.20	0.20	0.22
Grid type	BLG	BLG	BLG	BLG	BLG	PG+ BLG	PG	PG	PG	PG
Zone	**CPPSPM**					**Interface**				
β	**0.6**	**0.8**	**1**	**1.2**	**1.4**	**0.6**	**0.8**	**1**	**1.2**	**1.4**
Skewness	0.26	0.18	0.22	0.19	0.18	0.09	0.12	0.14	0.28	0.24
Grid type	PG + BLG	SG	SG	SG	SG	BLG	BLG	MG	MG+ BLG	MG

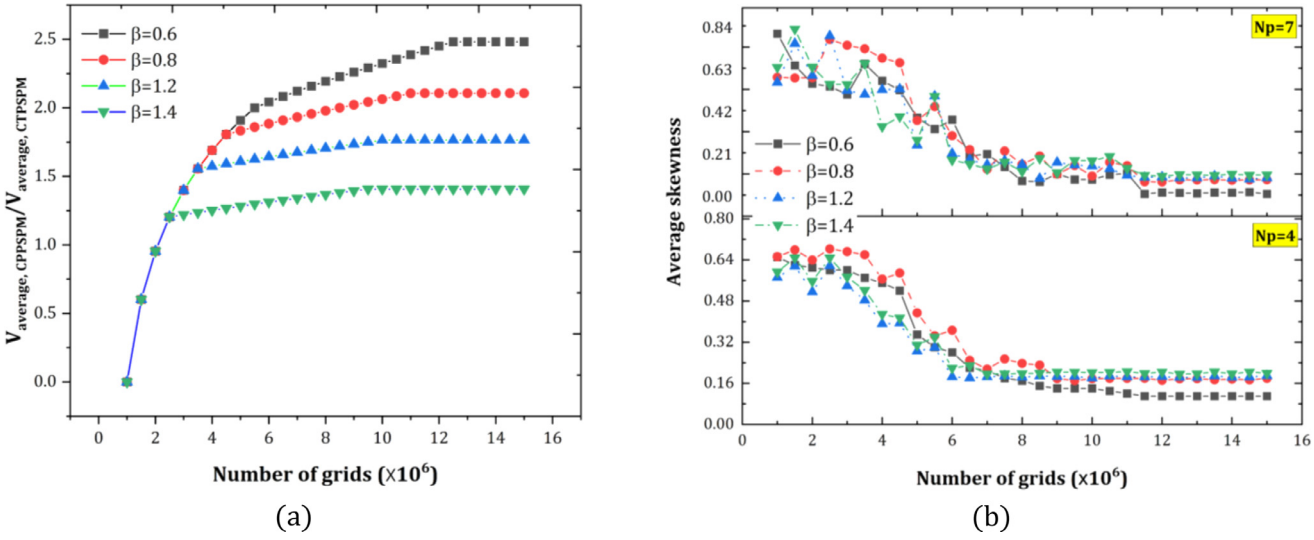


Fig. 3. Grid study for (a) Velocity ratio based on grid variation and (b) Average skewness in the calculation domain when Nc =4, and *Re*=300.

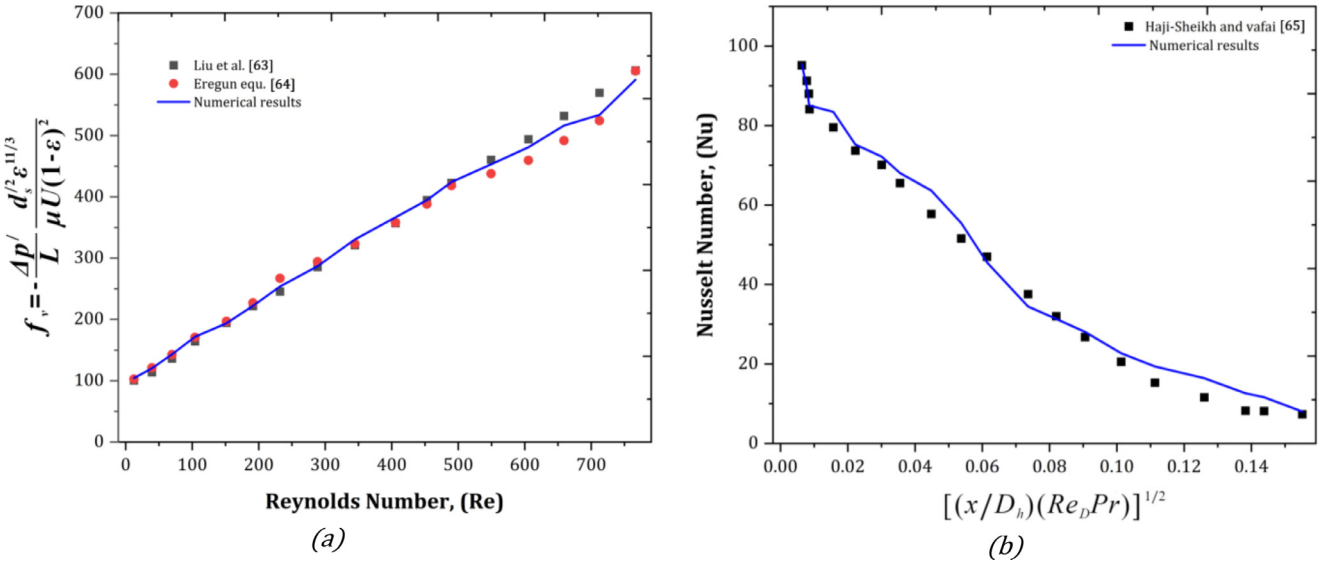


Fig. 4. Validation study for a) Comparison of the numerical results for the modified friction factor with Liu et al. [63] and Ergun equation [64] and b) *Nu* with dimensionless parameter (Haji-Sheikh [65]).

5. Results and discussion

5.1. The effect of CT on the geometrical properties and calculation cost analysis

Porous media can significantly affect the interface between a fluid and a solid. As illustrated in **Fig. 5a**, the variation of the surface area ratio for Nc=4 and Nc=7 is shown for all ranges of β. The results indicated that increasing β could improve the surface area ratio. For $\beta<1$, the CTPSPM reduced the surface area ratio, which may have an effect on heat transfer, pressure drop, and the number of grids due to porosity variation. According to the results, increasing Nc improves the surface area ratio, while increasing β has a greater effect on the surface area ratio.

The PSPM approach to porous media necessitates the use of numerous grids that increase the calculation cost. When CT is applied

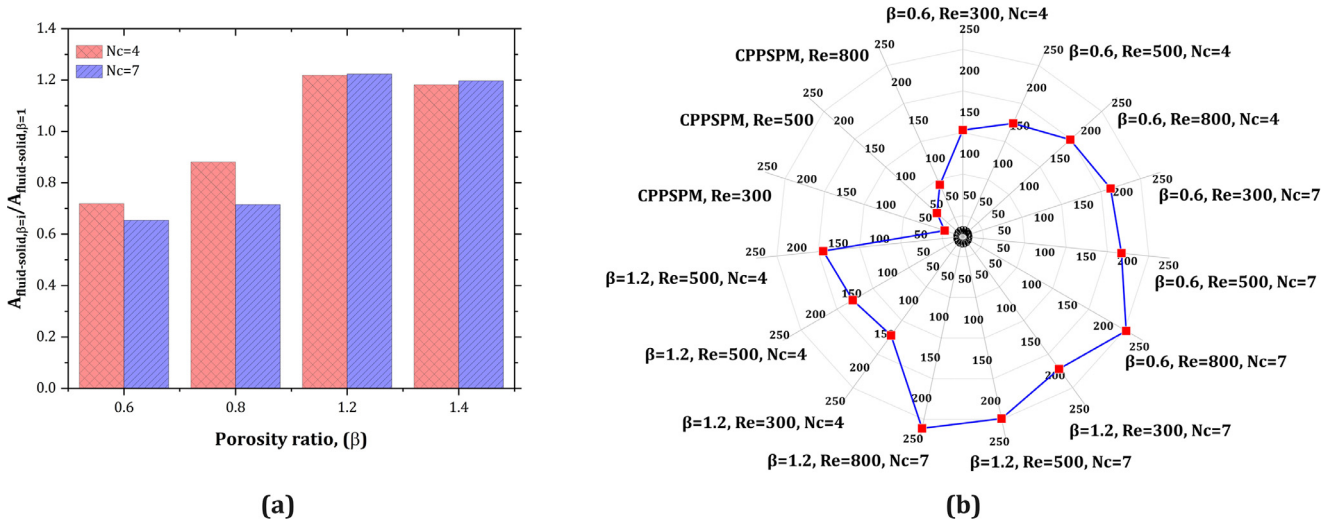


Fig. 5. Variations in a) Surface area ratio with porosity ratio, β and b) Calculation cost (minute) for wide ranges of *Re*, Nc, and β when Np=2.

Table 7
Variation of R^2 in predicted results when Np=2, and *Re*=300.

			Nc=3	
	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1.2$	$\beta = 1.4$
R^2	0.989	0.974	0.982	0.996
			Nc=4	
	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1.2$	$\beta = 1.4$
R^2	0.981	0.985	0.978	0.971
			Nc=8	
	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1.2$	$\beta = 1.4$
R^2	0.987	0.996	0.964	0.984
			Nc=9	
	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1.2$	$\beta = 1.4$
R^2	0.984	0.988	0.971	0.974
			Nc=10	
	$\beta = 0.6$	$\beta = 0.8$	$\beta = 1.2$	$\beta = 1.4$
R^2	0.991	0.987	0.986	0.983

to the PSPM, the calculation cost increases due to the variation in β. This factor (β) can affect the contact surface area of the CTPSPM with fluid and the number of grids by affecting the number of pores in the CTPSPM. As a result, the cost of calculation is directly related to β. In addition, the interface between CTPSPM and CPPSPM will change according to the variation in Nc and Np. The results indicated that when Nc=4 is replaced with Nc=7, the interface between CTPSPM and CPPSPM increases by 116% when Np=2. Due to the dense grids on both sides of the interface, a higher level of Nc has a significant effect on the calculation cost. **Fig. 5b** illustrates the average time required to calculate various case studies (in minutes).

According to **Fig. 5b**, increasing the Nc from 4 to 7 increases the calculation time by up to 77% when β =0.6, *Re*=800, and Np=2. The results indicate that the calculation cost becomes more complex as Nc increases. Based on the results, differences in Nc and Np can increase calculation cost by 120% compared to a constant porosity across all zones.

5.1.1. Comparison calculation cost between machine learning, surface response method (RSM), and genetic algorithm (GA)

As mentioned before, increasing Np or Nc can significantly increase the calculation cost of simulation. As a result of the numerous calculated data, an optimization approach based on the results can have an effect on the cost of optimization. An optimization algorithm finds the values of the parameterized mapping function's parameters in such a way as to minimize the error of the function when it maps inputs to outputs based on a machine learning algorithm (e.g., a weighted sum of inputs). A machine learning algorithm solves an optimization problem every time it is fitted to a training dataset [9,66]. Two distinct optimization approaches were compared to a machine learning model to assess the effect of optimization methods on calculation costs. The surface response method (RSM) and genetic algorithm (GA) are considered the most frequently used optimization methods for this case study [67].

To compare ML, RSM, and GA, the calculation time for 1<Np<10, Nc=7 and 1<Nc<10, Np=2 will be compared for three different values of *Re*. It's worth noting that the comparison codes are developed using the GPU in two different cores (12 and 48). Therefore, this section also presents the effect on the number of cores. According to **Table 8**, when compared to GA and RSM methods, the ML method can significantly reduce the calculation cost associated with optimization. Additionally, the results demonstrated that optimizing for Nc takes less time than optimizing for Np. This issue refers to the training data set (numerical results). The results also show that the number of cores can impact the calculation cost.

Table 8
Comparison between calculation time for ML, RSM and GA when 1<Np<10, Nc=7 and 1<Nc<10, Np=2.

Re	ML (minutes)				RSM (minutes)				GA (minutes)			
	1<Np<10 Nc=7		1<Nc<10 Np=2		1<Np<10 Nc=7		1<Nc<10 Np=2		1<Np<10 Nc=7		1<Nc<10 Np=2	
	12 Cores	48 cores	12 Cores	48 cores	12 Cores	48 cores	12 Cores	48 cores	12 Cores	48 cores	12 Cores	48 cores
300	218.24	84.22	192.54	78.31	598.74	458.22	411.74	322.87	477.67	241.88	347.85	255.78
500	320.54	124.55	274.41	81.25	896.21	702.31	636.31	301.67	782.04	367.97	496.65	306.44
800	364.05	134.46	314.96	97.21	964.14	721.23	715.88	481.5	855.27	397.24	704.48	524.68

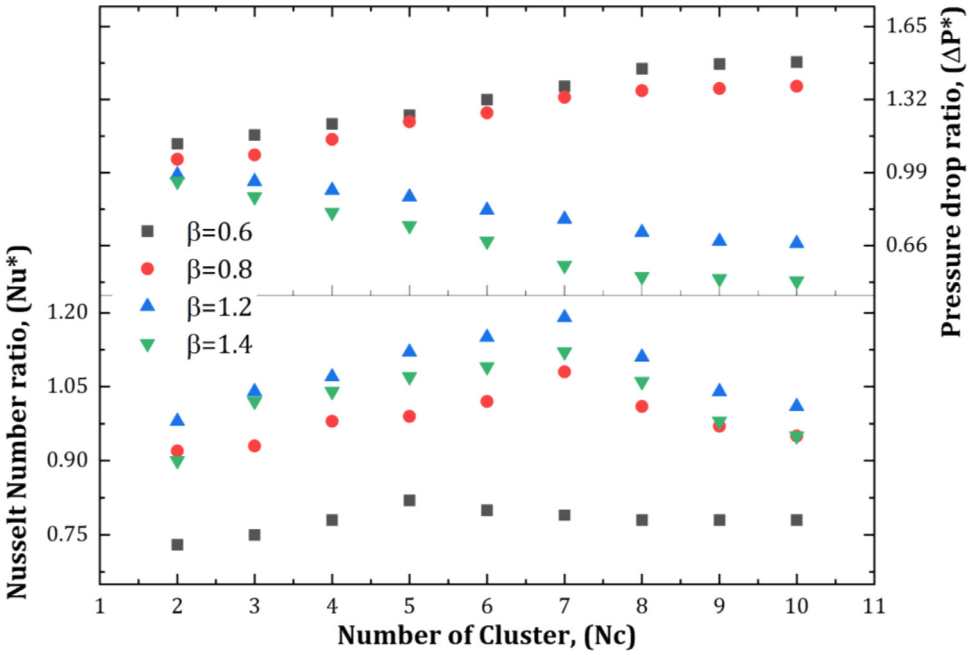


Fig. 6. Variation of $Nu^*$and ΔP^* with Nc (Re=800, Np =2).

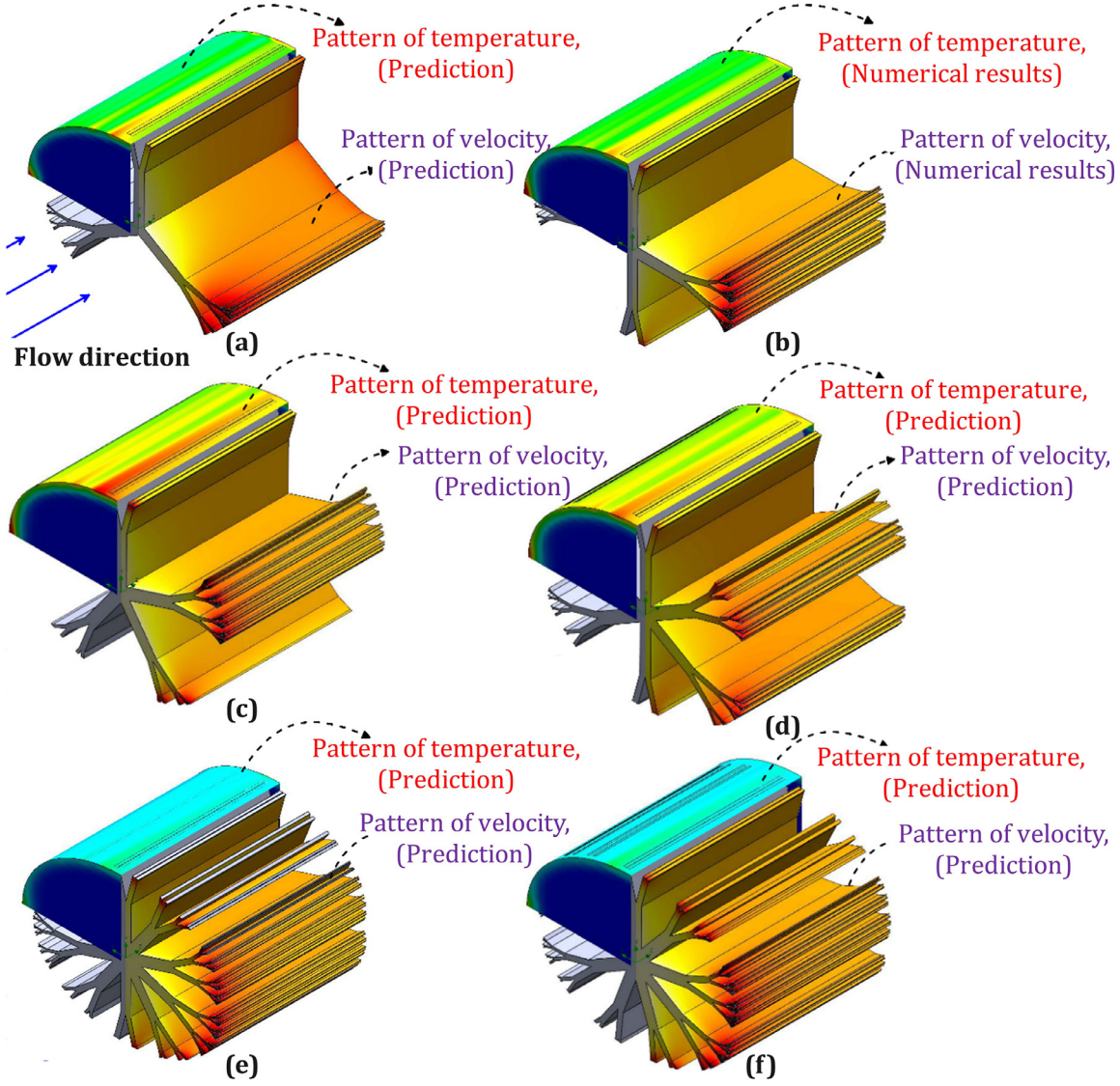


Fig. 7. Dimensionless temperature and velocity for variation of Nc (Np=2, Re=800, and $\beta = 1.2$) for (a) Nc=3, (b) Nc=4, (c) Nc=5, (d) Nc=6, (e) Nc=10, and (f) Nc=9.

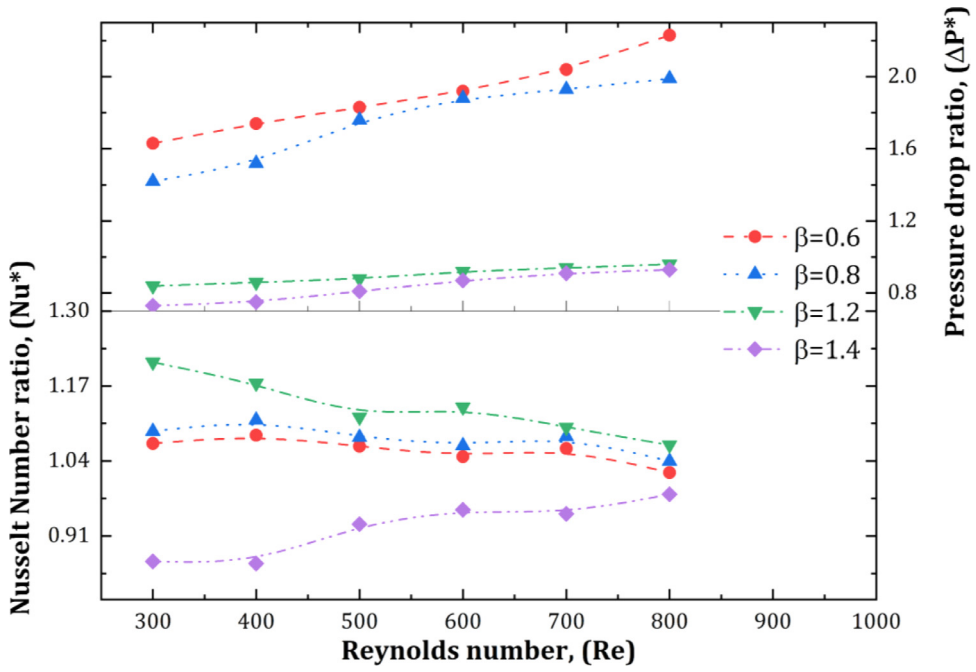


Fig. 8. Variation of $Nu^*$and $\Delta P^*$with Re, Nc =7 and Np=2 (Prediction).

5.2. Prediction of the thermal behavior of CTPSPM *(BBML results)*

Due to the high cost of calculation and the difficulty of generating a high level of CTPSPM, calculating the effect of different cluster configurations requires a novel calculation method. Fig. 6 illustrates the variation in Nu* and ΔP* over the Nc range when *Re*=800 and Np=2. The results indicate that Nu* is highest at β=1.2. After Nc=7, the pattern of Nu* decreases, implying that the CTPSPM is unable to improve Nu* in the high Nc range (Nc>7) due to the reduced volume of the CPPSPM–CTPSPM ratio. The results also indicated that ΔP* will increase when β=0.6 and 0.8. These results refer to the definition of β. Despite the fact that β<1, for β=1.2 and 1.4, the ΔP* will decrease with increasing Nc (up to 47.36%, Nc=1.4 when, *Re*=800, and Np =2). By increasing Nc, due to the tube's elliptical geometry, the contact surface area between the CTPSPM and tube wall can be increased by up to 36.28% when Nc=10 and Np=2, compared to Nc=2. As a result, the heat transfer rate from conduction to convection can be increased by 14.02% for the CTPSPM. Predicting the thermal behavior through the CTPSPM and CPPSPM requires re-generating the loss function associated with the thermal pattern. As a result, gradients in temperature for the fluid and solid zones require true temperature distributions (based on calculation nodes), particularly for the solid zone. As previously stated, the pore size can be altered by β. As a result, the upper bound of β has the smallest surface area available for conduction heat transfer.

The dimensionless velocity and temperature distributions for various Nc extracted by BBML in the calculation domain are shown in Fig. 7 when β=1.2, *Re*=800, and Np=2. The predicted results for CTPSPM and CPPSPM are visualized in Figs. 7**a** to 7**f**. According to the results, an increase in Nc can change both dimensionless velocity and temperature. At the end of the pairs, the temperature and velocity distributions clearly demonstrate the effect of PSPM modeling. The tube's average velocity increases by increasing Nc (22.18% for Nc=10 and Np=2, compared to Nc=3 and Np=2). Fig. 7**e** indicates that for the highest number of Nc, the temperature radiance in the solid zone of the CTPSPM could reach 5.63 K, whereas, for Nc=3, this parameter is 12.76 K, indicating that CTPSPM can aid in achieving a more uniform temperature pattern along the tube.

The results for the velocity distributions through the CTPSPM and CPPSPM indicate that increasing Nc can result in decreased average velocity (13.98% for Nc=10 versus Nc=3). To determine flow pattern and thermal behavior, the geometric properties of the CTPSPM must be considered. An open-cell geometry with a random distribution of pores and a constant porosity distribution may provide the best combination of CTPSPM and CPPSPM interfaces.

The most important factor for heat transfer and pressure drop in CTPSPM and CPPSPM is *Re*. The flow-through PSPM presented significant variation due to hydraulic diameters and pore sizes. The steady-state approach to flow through PSPM produces a uniform gradient in velocity through the PSPM. In Fig. 8, Nu* and ΔP* are shown for various β in the range of *Re*, based on predicted results when Nc=7 and Np=2.

As expected, the results indicated that increasing *Re* improves the total heat transfer of the tube (14.62% for *Re*=800 compared to *Re*=300). The pattern of Nu* with *Re* is a critical point in this figure. The results indicated that as *Re* increases, Nu* decreases, implying that the CTPSPM significantly affects heat transfer in the low range of *Re*. Fig. 8 also shows that the behavior of ΔP* for different types of β ascends. β <1 can be increased rapidly by increasing *Re*, while β>1 is barely affected by increasing *Re*.

Figs. 9**a** to 9**c** show the variation of Nu*and ΔP* along the elliptical tube for *Re*=300, 500, and 800 when Np=2, Nc=7 β=1.2, Nu* increases as tube length increases. The pattern of Nu* depends on *Re*. The total heat transfer of an elliptical tube could be improved by 14.55% for *Re*=300, 15.28% for *Re*=500, and 17.22% for *Re*=800 when Nc=7, Np=2, and β=1.2 compared with the similar case when β=1. Additionally, the result indicates that β=0.6 has the greatest pressure gradient along its length due to its porosity. Additionally, the pressure pattern demonstrates that CTPSPM can reduce a pressure drop while increasing heat transfer in the PSPM domain. This property has provided a new perspective on passive cooling methods without significant pressure drop.

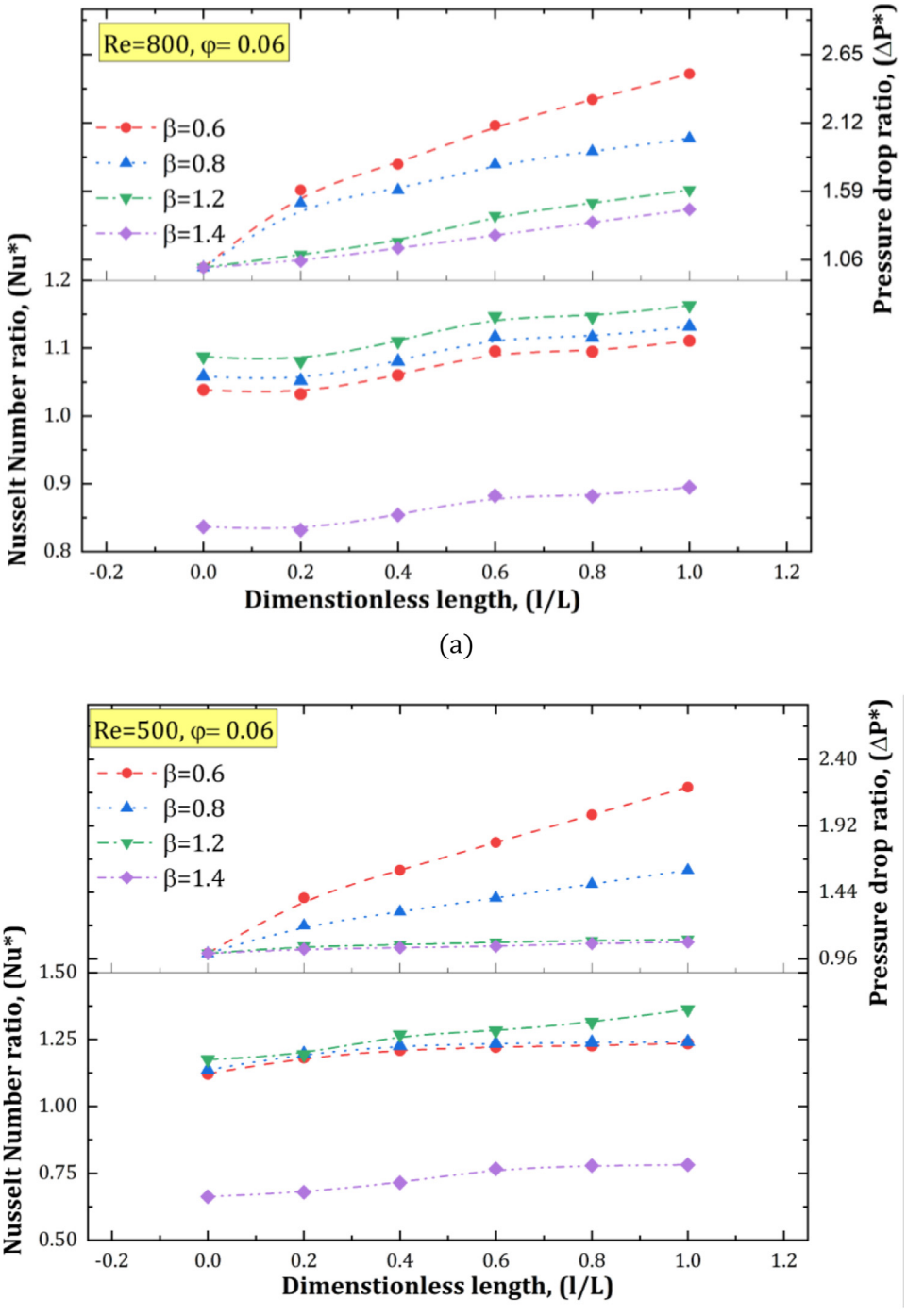
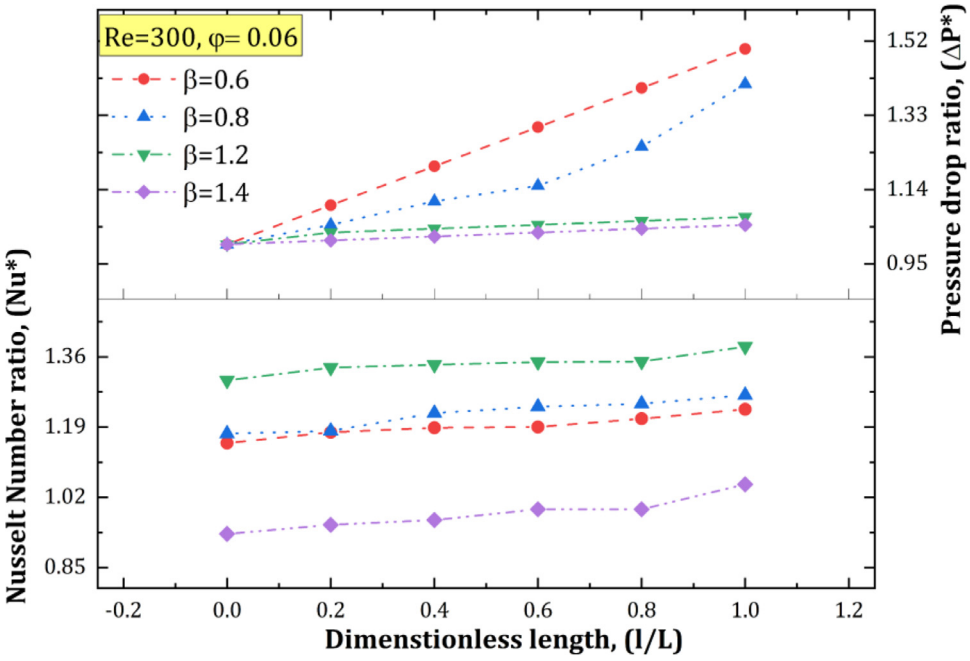


Fig. 9. Variation of Nu^* and ΔP^* along the tube for (a) *Re*=800, (b) *Re*=500, and (c) *Re*=300 when Np=2, Nc=7.

5.3. The effect of variation in the Np on heat transfer and pressure drop

In the previous section, the effects of Nc are examined, and when $\beta = 1.2$, *Re*= 800, and Nc = 7, the optimal configuration for heat transfer and pressure drop is generated. Calculation costs can be impacted significantly by increasing Np due to complex geometry. On the other hand, increasing Np also has an effect on the interface between CTPSPM and CPPSPM. Additionally, the practical approach to high-level pairs is ambiguous. The predicted results show that increasing Np would improve heat transfer and pressure drop (Fig. 10). As a result, varying Np could reduce the pressure drop by increasing the average velocity near the tube. Increasing Np will increase the density of CTPSPM geometry. The geometry of the CTPSPM becomes more complex as the Np increases, as illustrated in Fig. 10. The predicted values for Np variation are shown in Table 9. The results indicate that increasing Np improves the elliptical tube's total heat transfer by 14.38% when Np=6 instead of Np=2. Additionally, Nu^* can improve by 5.92% when Np=6 is used instead of Np=2.



(c)

Fig. 9. Continued

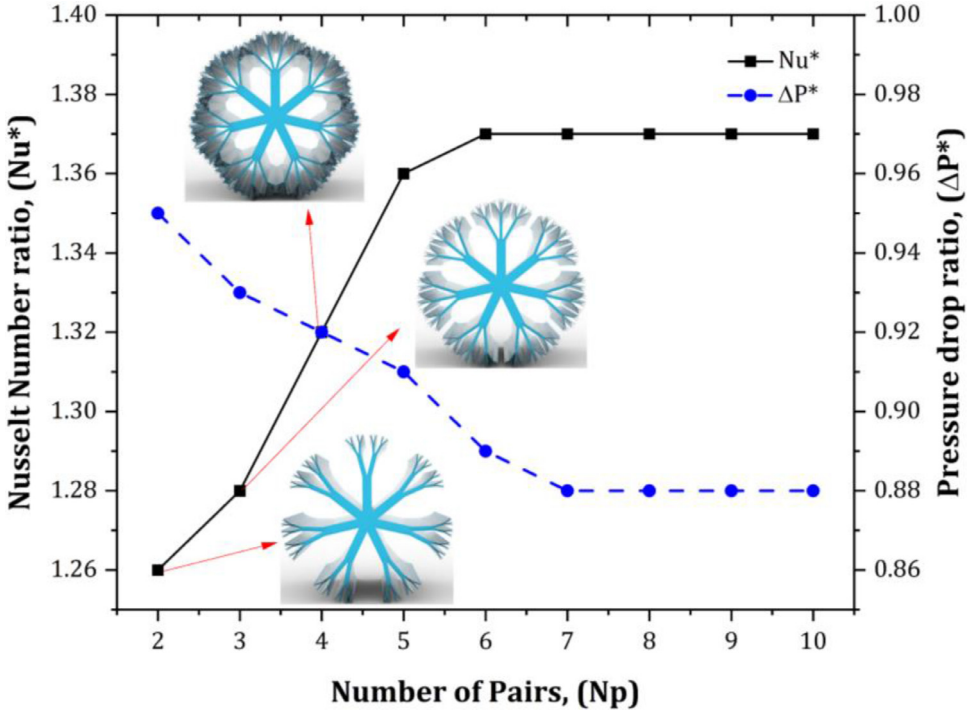


Fig. 10. The effect of variation of Np on Nu* and ΔP^* when Re=800, $\beta = 1.2$ and Nc=7.

6. Conclusion

As one of the nature-based optimization methods, construction theory is severely constrained by its computational methods when applied to complex geometry. For the first time, this paper presents a comprehensive study of heat transfer and pressure drop in a wide variety of pairs (1<Np<10) and clusters (1<Nc<10) in a constructal theory-based pore-scale porous media (CTPSPM). To determine the optimal combination of pairs and clusters, an individual combination of machine learning and numerical simulation was developed after evaluating three optimization methods (Genetic algorithm, Surface response method, and Machine learning) and examining their calculation costs. Specific tensors (component velocity, pressure, and temperature for fluid and solid) are trained in this method using numerical results from each location according to the geometry. A GPU-accelerated machine learning algorithm predicted the number of pairs (1<Np<3, 4<Np<6, 8<Np<10) and the number of clusters (Nc=1 and 3<Nc<10). Each

Table 9
Variation of parameters with changing Np when = 1.2, Nc=7 and *Re*=800.

Np	$\frac{Volume_{Np=i}}{Volume_{Np=2}}$	$\frac{Nu^*_{Np=i}}{Nu^*_{Np=2}}(\%)$	$\frac{\Delta P^*_{Np=i}}{\Delta P^*_{Np=2}}(\%)$
3	1.14	2.78	−2.15
4	1.37	4.02	−3.26
5	1.65	5.75	−4.39
6	1.79	5.92	−6.74
7	2.12	5.92	−7.95
8	2.35	5.92	−7.95
9	2.48	5.92	−7.95
10	2.65	5.92	−7.95

state contains four porosity ratios (β=0.6, 0.8, 1.2, and 1.4), and the total calculation domain contains 972 prediction states. The results indicate that:

1- When compared to genetic algorithms and surface response methods, machine learning can significantly reduce calculation costs.
2- When Nc=4 is replaced with Nc=7, the interface between the CTPSPM and CPPSPM increases by 116% when Np=2.
3- ΔP^* increases by up to 33% for β=0.6, *Re*=300, Np=2, and Nc=7 compared to β=0.6, *Re*=300, Np=2, and Nc=4.
4- Increasing *Re* improves the total heat transfer of the tube (14.62% for *Re*=800 compared to *Re*=300). The pattern of Nu^* with *Re* is a critical point in this figure. The results indicate that as *Re* increases, Nu^* decreases, implying that CTPSPM significantly affects heat transfer in the low range of *Re*.
5- Increasing Np improves the elliptical tube's total heat transfer by 14.38% when Np=6 instead of Np=2. Additionally, Nu^* can be improved by 5.92% when Np=6 instead of Np=2.

Declaration of Competing Interest

All authors have participated in (a) conception and design, or analysis and interpretation of the data; (b) drafting the article or revising it critically for important intellectual content; and (c) approval of the final version. o This manuscript has not been submitted to, nor is under review at, another journal or other publishing venue. The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript. The following authors have affiliations with organizations with a direct or indirect financial interest in the subject matter discussed in the manuscript:

CRediT authorship contribution statement

Mehrdad Mesgarpour: Conceptualization, Software, Validation, Formal analysis, Writing – review & editing, Writing – original draft. **Kittipong Sakamatapan:** Writing – review & editing. **Ahmet Selim Dalkılıç:** Writing – review & editing. **Rasool Alizadeh:** Writing – review & editing. **Ho Seon Ahn:** Writing – review & editing. **Somchai Wongwises:** Conceptualization, Supervision, Project administration.

Acknowledgment

The first author acknowledges the Postdoctoral Fellowship from KMUTT. The second author acknowledges the Royal Golden Jubilee Ph.D program. The third and fifth authors acknowledge the visiting professorship from KMUTT. The sixth author acknowledges the National Science and Technology Development Agency (NSTDA) under the "Research Chair Grant", and Thailand Science Research and Innovation (TSRI) under Fundamental Fund 2022.

Appendix A1. Geometrical properties of PSPM

As illustrated in **Fig. A1,** PSPM contains open-cell foam. According to the porosity ratio (β), PSPM can have a range of minimum and maximum pore diameters. **Table A1** shows the pores' average, maximum, and minimum diameters. Hydraulic diameter to find the Nu for PSPM is defined as:

$$D_h = \frac{6}{a_v}. \tag{A1}$$

where $\frac{6}{a_v}$ is the solid specific surface.

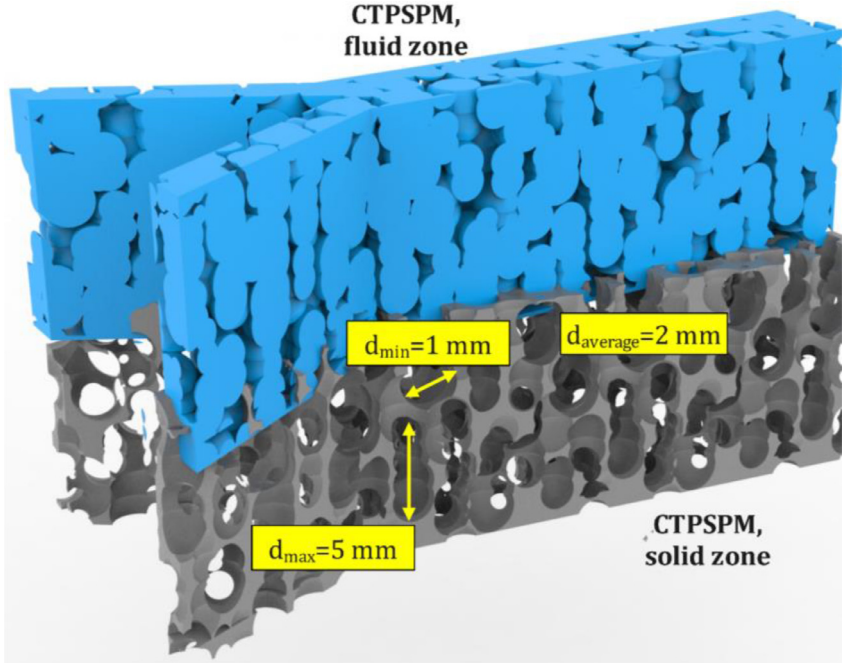


Fig. A1. Fluid and solid zones of CTPSPM.

Table A1
Geometrical details of CTPSPM.

β	Maximum pore diameter (mm)	Minimum pore diameter (mm)	Average pore diameter (mm)
0.6	3	1	1.59
0.8	5	1	2.0
1	6	1	2.5
1.2	7	1	3.24
1.4	8	1	4.86

Table A2
A study on the sensitivity analysis for hidden layers based on heat transfer target.

No.	Hidden layers	Input parameters	MAE	R^2
1.	(160,150,140,130,120,110,100,90,80,70,60,50,40,30,20,10)	T_{CTPSPM}, T_{tube}	0.036	0.940
2.	(150,140,130,120,110,100,90,80,70,60,50,40,30,20,10)	T_{CPPSPM},$V_{,component}$,T_{fluid}	0.057	0.980
3.	(140,130,120,110,100,90,80,70,60,50,40,30,20,10)	P	0.061	0.968
4.	**(130,120,110,100,90,80,70,60,50,40,30,20,10)**		**0.011**	**0.996**
5.	(120,110,100,90,80,70,60,50,40,30,20,10)		0.103	0.954
6.	(110,100,90,80,70,60,50,40,30,20,10)		0.028	0.994
7.	(100,90,80,70,60,50,40,30,20,10)		0.096	0.989
8.	(90,80,70,60,50,40,30,20,10)		0.105	0.913
9.	(80,70,60,50,40,30,20,10)		0.026	0.968
10.	(70,60,50,40,30,20,10)		0.040	0.946
11.	(60,50,40,30,20,10)		0.060	0.944
12.	(50,40,30,20,10)		0.018	0.918
13.	(40,30,20,10)		0.016	0.992
14.	(30,20,10)		0.104	0.942

Appendix A2. Sensitivity analysis for the neural network

Table A2

References

[1] Y. Ge, Z. Liu, W. Liu, Multi-objective genetic optimization of the heat transfer for tube inserted with porous media, Int. J. Heat Mass Transf. 101 (2016) 981–987.
[2] F. Matsunaga, V. Zytkowski, P. Valle, F. Deschamps, Optimization of energy efficiency in smart manufacturing through the application of cyber–physical systems and industry 4.0 technologies, J. Energy Resour. Technol. 144 (10) (2022) 102104.
[3] H. Kazemi, C. Seepersad, H.A.. Kim, Topology optimization integrated deep learning for multiphysics problems. in AIAA SCITECH 2022 Page 0802, Forum (2022).
[4] A. Ramadan, K. Yousef, M. Said, M. Mohamed, Shape optimization and experimental validation of a drag vertical axis wind turbine, Energy 151 (2018) 839–853.
[5] B. Selma, M. Désilets, P. Proulx, Optimization of an industrial heat exchanger using an open-source CFD code, Appl. Therm. Eng. 69 (2013) 241–250.
[6] T. Wang, Y. Wen, B. Lin, Energy consumption and the influencing factors in China: a nonlinear perspective, J. Clean. Prod. 249 (2020) 119375.
[7] Y. Ding, W. Zhang, L. Yu, K. Lu, The accuracy and efficiency of GA and PSO optimization schemes on estimating reaction kinetic parameters of biomass pyrolysis, Energy 176 (2019) 582–588.
[8] J. Weber, W. Fullmer, A. Gel, J. Musser, Optimization of a cyclone using multiphase flow computational fluid dynamics, J. Fluids Eng. 142 (3) (2020) 211–213.
[9] A. Shalaby, A. Elkamel, P.L. Douglas, Q. Zhu, Q.P. Zheng, A machine learning approach for modeling and optimization of a CO_2 post-combustion capture unit, Energy 215 (2021) 119113.
[10] T. Robič, B. Filipič, Differential Evolution For Multiobjective optimization. in International conference On Evolutionary Multi-Criterion Optimization, Springer, 2005.
[11] M.W.A. Khan, M.I. Khan, T. Hayat, A. Alsaedi, Entropy generation minimization (EGM) of nanofluid flow by a thin moving needle with nonlinear thermal radiation, Phys. B 534 (2018) 113–119.
[12] K. Boonma, M. Mesgarpour, J.M. NajmAbad, R. Alizadeh, O. Mahian, A.S. Dalkılıç, H.S. Ahn, S. Wongwises, Prediction of battery thermal behaviour in the presence of a constructal theory-based heat pipe (CBHP): a multiphysics model and pattern-based machine learning approach, J. Energy Storage 48 (2022) 103963.
[13] A. Bejan, Fundamentals of exergy analysis, entropy generation minimization, and the generation of flow architecture, Int. J. Energy Res. 26 (7) (2002).
[14] Bejan, A. and Lorente S. , Design with constructal theory. 2008.
[15] L. Hermany, G. Lorenzini, R. Klein, F. Zinani, E. Dos Santos, L. Isoldi, L. Rocha, Constructal design applied to elliptic tubes in convective heat transfer cross-flow of viscoplastic fluids, Int. J. Heat Mass Transf. 116 (2018) 1054–1063.
[16] A. Golparvar, Y. Zhou, K. Wu, J. Ma, Z. Yu, A comprehensive review of pore scale modeling methodologies for multiphase flow in porous media, Adv. Geo Energy Res. 2 (4) (2018) 418–440.
[17] X. Mou, Z. Chen, Pore-scale simulation of heat and mass transfer in deformable porous media, Int. J. Heat Mass Transf. 158 (2020) 119878.
[18] R. Habib, N. Karimi, B. Yadollahi, M.H. Doranehgard, L.K. Li, A pore-scale assessment of the dynamic response of forced convection in porous media to inlet flow modulations, Int. J. Heat Mass Transf. 153 (2020) 119657.
[19] S. Bakhshian, S.A. Hosseini, N. Shokri, Pore-scale characteristics of multiphase flow in heterogeneous porous media using the lattice Boltzmann method, Sci. Rep. 9 (1) (2019) 1–13.
[20] O. Mahian, I. Pop, A.Z. Sahin, H. Oztop, S. Wongwises, rreversibility Analysis of a Vertical Annulus Using TiO2/ WaterNanofluid with MHD Flow Effects, International Journal of Heat and Mass Transfer 64 (2013) 671–679.
[21] J. Li, H. Xiao, Q. Li, J. Shi, Heat recirculation and heat losses in porous micro-combustors: effects of wall and porous media properties and combustor dimensions, Energy 220 (2021) 119772.
[22] K. Nilpueng, L.G. Asirvatham, A.S. Dalkılıç, O. Mahian, H.S. Ahn, S. Wongwises, Heat transfer and fluid flow characteristics in a plate heat exchanger filled with copper foam, Heat Mass Transf. 56 (12) (2020) 3261–3271.
[23] M. Hatami, D. Ganji, Thermal and flow analysis of microchannel heat sink (MCHS) cooled by Cu–water nanofluid using porous media approach and least square method, Energy Convers. Manag. 78 (2014) 347–358.
[24] G. Bourantas, E. Skouras, V. Loukopoulos, V. Burganos, Heat transfer and natural convection of nanofluids in porous media, Eur. J. Mech. B Fluids 43 (2014) 45–56.
[25] M. Sheikholeslami, CuO-water nanofluid flow due to magnetic field inside a porous media considering brownian motion, J. Mol. Liq. 249 (2018) 921–929.
[26] Bejan, A. and Dan N., Constructal trees of convective fins. 1999.
[27] A. Bejan, S. Lorente, The constructal law and the evolution of design in nature, Phys. Life Rev. 8 (3) (2011) 209–240.
[28] Z. Wu, H. Feng, L. Chen, Y. Ge, Performance optimization of a condenser in ocean thermal energy conversion (otec) system based on constructal theory and a multi-objective genetic algorithm, Entropy 22 (6) (2020) 641.
[29] W. Tang, H. Feng, L. Chen, Z. Xie, J. Shi, Constructal design for a boiler economizer, Energy 223 (2021) 120013.
[30] M.S. Dehaj, H. Hajabdollahi, Fin and tube heat exchanger: constructal thermo-economic optimization, Int. J. Heat Mass Transf. 173 (2021) 121257.
[31] M. Mehrgoo, M. Amidpour, Constructal design and optimization of a dual pressure heat recovery steam generator, Energy 124 (2017) 87–99.
[32] Z. Wu, H. Feng, L. Chen, Z. Xie, C. Cai, Pumping power minimization of an evaporator in ocean thermal energy conversion system based on constructal theory, Energy 181 (2019) 974–984.
[33] E.M. Nava-Arriaga, A. Hernandez-Guerrero, J.L. Luviano-Ortiz, A. Bejan, Heat sinks with minichannels and flow distributors based on constructal law, Int. Commun. Heat Mass Transf. 125 (2021) 105122.
[34] J. Liu, H. Wang, Machine learning assisted modeling of mixing timescale for LES/PDF of high-Karlovitz turbulent premixed combustion, Combust. Flame 238 (2022) 111895.
[35] M. Mesgarpour, J.M.N. Abad, R. Alizadeh, S. Wongwises, M.H. Doranehgard, S. Ghaderi, N. Karimi, Prediction of the spread of Corona-virus carrying

droplets in a bus – a computational based artificial intelligence approach, J. Hazard. Mater. 413 (2021) 125358.
[36] L. Helei, P. Tantikhajorngosol, C. Chan, P. Tontiwachwuthikul, Technology development and applications of artificial intelligence for post-combustion carbon dioxide capture: critical literature review and perspectives, Int. J. Greenh. Gas Control 108 (2021) 103307.
[37] Z. Liu, I.A. Karimi, Gas turbine performance prediction via machine learning, Energy 192 (2020) 116627.
[38] B. Hu, J. Yang, J. Li, S. Li, H. Bai, Intelligent control strategy for transient response of a variable geometry turbocharger system based on deep reinforcement learning, Processes 7 (9) (2019) 601.
[39] T. Nguyen, S. King, Y. Hassan, Experimental investigation of turbulent characteristics in pore-scale regions of porous media, Exp. Fluids 62 (4) (2021) 1–27.
[40] M. Kaviany, Principles of Heat Transfer in Porous Media, Springer-Verlag, 1991.
[41] M. Kaviany, Principles of Heat Transfer in Porous Media, Springer Science & Business Media, 2012.
[42] S. Senthilraja, K. Vijayakumar, R. Gangadevi, A comparative study on thermal conductivity of Al_2O_3/water, CuO/water and Al_2O_3–CuO/water nanofluids, Dig. J. Nanomater. Biostruct. 10 (4) (2015) 1449–1458.
[43] C. Nguyen, F. Desgranges, G. Roy, N. Galanis, T. Maré, e. Boucher, H.A. Mintsa, Temperature and particle-size dependent viscosity data for water-based nanofluids–hysteresis phenomenon, Int. J. Heat Fluid Flow 28 (6) (2007) 1492–1506.
[44] S.K. Vasudevan, S.R. Pulari, S. Vasudevan, Deep Learning: A Comprehensive Guide, Chapman and Hall/CRC, 2022.
[45] Elliott, D.L., A better activation function for artificial neural networks. 1993.
[46] T. Chai, R.R. Draxler, Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature, Geosci. Model Dev. 7 (3) (2014) 1247–1250.
[47] G.A. Carpenter, S. Grossberg, A self-organizing neural network for supervised learning, recognition, and prediction, IEEE Commun. Mag. 30 (9) (1992) 38–49.
[48] J. Berkson, Application of the logistic function to bio-assay, J. Am. Stat. Assoc. 39 (227) (1944) 357–365.
[49] M. Wang, S. Lu, D. Zhu, J. Lin, Z. Wang, A high-speed and low-complexity architecture for softmax function in deep learning, in: Proceedings of the 2018 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS), IEEE, 2018.
[50] Janocha, K. and Czarnecki W.M., On loss functions for deep neural networks in classification. arXiv preprint arXiv:1702.05659, 687-681 2017.
[51] P.T. De Boer, D.P. Kroese, S. Mannor, R.Y. Rubinstein, A tutorial on the cross-entropy method, Ann. Oper. Res. 134 (1) (2005) 19–67.
[52] M. Riedmiller, A. Lernen, in: Multi Layer Perceptron, Machine Learning Lab Special Lecture, University of Freiburg, 2014, pp. 7–24.
[53] C.E. Rasmussen, Gaussian Processes in Machine learning. in Summer school On Machine Learning, Springer, 2003.
[54] K. Makantasis, A.D. Doulamis, N.D. Doulamis, A. Nikitakis, Tensor-based classification models for hyperspectral data analysis, IEEE Trans. Geosci. Remote Sens. 56 (12) (2018) 6884–6898.
[55] C.F. Van Loan, Generalizing the singular value decomposition, SIAM J. Numer. Anal. 13 (1) (1976) 76–83.
[56] N. Zhou, X. Zhao, B. Han, P. Li, Z. Wang, J. Fan, A novel quick and robust capacity estimation method for Li-ion battery cell combining information energy and singular value decomposition, J. Energy Storage 50 (2022) 104263.
[57] Koroko, A., Anciaux-Sedastrian A., Gharbia I., Garès V., Haddou M., and Tran Q.H., Efficient approximations of the fisher matrix in neural networks using kronecker product singular value decomposition. arXiv preprint arXiv:2201.10285, 2022.
[58] L. Kuang, L.T. Yang, S. Rho, Z. Yan, K. Qiu, A tensor-based framework for software-defined cloud data center, ACM Trans. Multimed. Comput. Commun. Appl. (TOMM) 12 (2016) 1–23.
[59] H. Wurps, G. Steinfeld, S. Heinz, in: Grid-resolution Requirements For Large-Eddy Simulations of the Atmospheric Boundary Layer, Boundary-Layer Meteorology, 2020, pp. 1–23.
[60] K. Zore, D. Caridi, I. Lockley, Fast and Accurate Prediction of Vehicle Aerodynamics Using ANSYS Mosaic Mesh, SAE Technical Paper, 2020.
[61] R.V. Garimella, J. Kim, M. Berndt, Polyhedral mesh generation and optimization for non-manifold domains, in: Proceedings of the 22nd International Meshing Roundtable, Springer, 2014, pp. 313–330.
[62] X. Huang, Y. Xie, Convergent and mesh-independent solutions for the bi-directional evolutionary structural optimization method, Finite Elem. Anal. Des. 43 (14) (2007) 1039–1049.
[63] S. Liu, A. Afacan, J. Masliyah, Steady incompressible laminar flow in porous media, Chem. Eng. Sci. 49 (21) (1994) 3565–3586.
[64] A. Koekemoer, A. Luckos, Effect of material type and particle size distribution on pressure drop in packed beds of large particles: extending the Ergun equation, Fuel 158 (2015) 232–238.
[65] A. Haji-Sheikh, K. Vafai, Analysis of flow and heat transfer in porous media imbedded inside various-shaped ducts, Int. J. Heat Mass Transf. 47 (8–9) (2004) 1889–1905.
[66] D. Weichert, P. Link, A. Stoll, S. Rüping, S. Ihlenfeldt, S. Wrobel, A review of machine learning for the optimization of production processes, Int. J. Adv. Manuf. Technol. 104 (5) (2019) 1889–1902.
[67] D.S. Correia, C.V. Gonçalves, S.S. da Cunha Jr, V.A. Ferraresi, Comparison between genetic algorithms and response surface methodology in GMAW welding optimization, J. Mater. Process. Technol. 160 (1) (2005) 70–76.