






RESEARCH ARTICLE | AUGUST 16 2024

A data-driven phase change model for injection flow modeling

Yanfei Li ; Chenxiang Zhao ; Song Cheng ; Hengjie Guo  



Physics of Fluids 36, 083324 (2024)

<https://doi.org/10.1063/5.0223244>



Articles You May Be Interested In

Anomaly detection of cyber attacks in cyber-physical systems based on deep learning and genetic algorithm

AIP Conf. Proc. (March 2025)

A new dynamic stall prediction framework based on symbiosis of experimental and simulation data

Physics of Fluids (December 2021)

An improved sliding mode observer algorithm for PMSM based on deformable fuzzy neural network

AIP Advances (February 2025)



Physics of Fluids

Special Topics Open for Submissions

[Learn More](#)

A data-driven phase change model for injection flow modeling

Cite as: Phys. Fluids **36**, 083324 (2024); doi: [10.1063/5.0223244](https://doi.org/10.1063/5.0223244)

Submitted: 13 June 2024 · Accepted: 28 July 2024 ·

Published Online: 16 August 2024



View Online



Export Citation



CrossMark

Yanfei Li,¹ Chenxiang Zhao,² Song Cheng,³ and Hengjie Guo^{2,a)}

AFFILIATIONS

¹School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China

²School of Power and Energy, Northwestern Polytechnical University, Xi'an 710129, China

³Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hong Kong, China

^{a)}Author to whom correspondence should be addressed: hengjie.guo@nwpu.edu.cn

ABSTRACT

A deep learning approach is developed to swiftly evaluate phase change in computational fluid dynamics (CFD) simulations of a multi-component, liquid–gas two-phase injection flow. This method significantly improves computational efficiency by using a deep feedforward neural network (DFNN) to replace the complex iterative solution of multi-species vapor–liquid equilibrium (VLE). The DFNN takes instantaneous pressure, temperature, and system composition as input and predicts the corresponding phase equilibrium state. A parametric study was conducted to optimize the neural network's hyperparameters, including the activation function, number of hidden layers, and neurons per hidden layer. The rate of phase change is then calculated as a linear relaxation toward phase equilibrium, guiding subsequent computational steps in the CFD solver. A case study was performed to test the proposed methodology, involving the injection of a superheated liquid ethanol–water mixture into a gaseous nitrogen environment. The simulation results and computational cost were examined. It is found that the DFNN model, while accurately representing the non-ideal non-equilibrium phase change of a multi-component injection flow, speeds up the VLE solution by four orders of magnitude, leading to a 30%–40% reduction in overall flow simulation time. This model shows promise for injection flow simulations, especially for systems with a large number of compositions, such as sustainable aviation fuels.

Published under an exclusive license by AIP Publishing. <https://doi.org/10.1063/5.0223244>

I. INTRODUCTION

With recent advancements in high-performance computing, there has been a significant enhancement in high-fidelity numerical simulations for cutting-edge propulsion and power systems. This has sparked a growing interest in computational fluid dynamics (CFD) research on fuel injection flows inside and near nozzles.^{1–3} A comprehensive understanding of the fluid dynamics within and immediately outside injectors is paramount for accurately representing subsequent phenomena such as atomization, fuel–air mixing, and combustion processes.

The homogeneous mixture model (HMM) is an important method for modeling fuel injection flow, known for its effectiveness in dealing with the small-scale flow channel inside the injectors.⁴ The homogeneous relaxation model (HRM)⁵ is frequently coupled with HMM to describe phase transitions. This model takes non-equilibrium effects into account by conceptualizing phase change as a linear approach to equilibrium. There is abundant evidence in the literature

that validates HRM's ability to accurately predict transitions from liquid to gas in internal flow scenarios. For instance, Schmidt *et al.*⁶ demonstrated its accuracy in predicting thermal non-equilibrium flows under various conditions in flash boiling channel studies. Guo and Torelli⁷ reported consistency between simulated cavitation behavior using HRM and x-ray 3-D tomographic data from a single-hole diesel injector. Beyond its application in internal flow applications, HRM has also been used in modeling two-phase external flow near nozzle exits.^{8–10}

The HRM has a defect in that it is only feasible for single-component liquids, while most fuels are multi-component. It is also unable to describe the mixing-driven vaporization outside the nozzle. To overcome these drawbacks, a non-equilibrium phase transition model, named the unified model, was introduced.^{11,12} This model incorporates VLE for mixtures of any composition and takes into account the timescales of different phase change mechanisms. Integrated as a user-defined function (UDF) into a CFD software named CONVERGE,¹³ this model underwent rigorous testing via

simulations of the Engine Combustion Network (ECN)'s Spray G injector using a well-studied 9-component fuel, PACE-20, under conditions similar to those in gasoline direct injection engines. The results showcased the model's ability to capture preferential vaporization and mixing-driven vaporization, representing a significant improvement over the HRM.

The unified model, while a valuable extension to HRM, still has certain limitations. It relies on Raoult's law to calculate VLE, so it cannot account for non-ideal mixing, which is important for components with different molecular structures. This issue could be addressed by including activity coefficients in the VLE solver to consider intermolecular interactions, but that would enhance algorithmic complexity and memory requirements. Moreover, the computational efficiency of the unified model decreases significantly as the number of components in the system increases. For example, a previous study reported a 360% increase in wall-clock time when the number of species increased from 2 to 10.¹²

Deep learning is a thriving subdomain within the vast landscape of artificial intelligence. It focuses on creating sophisticated algorithms that mimic the intricacies of the human brain. At the forefront of this domain is the deep neural network (DNN), which is known for its architecture enriched with numerous hidden layers.¹⁴ This architecture gives DNNs remarkable proficiency in learning and approximating complex nonlinear functions, as well as handling the interpolation of unseen data encountered beyond the training phase. The capabilities of DNN offer a potential solution to address the constraints of the unified model. Specifically, the non-ideal mixing effect can be easily considered by optimizing the network parameters, and the computational cost can be significantly reduced, especially when applied to thermodynamic systems with numerous components.

In this study, a framework has been developed to swiftly and accurately predict VLE using DNNs. This enables a significant improvement over the traditional iteration-based approach. The framework has been seamlessly incorporated into the CFD solver, CONVERGE version 3.0,¹³ using UDFs. To compare the performance of the data-driven model with the iteration-based unified model, simulations have been carried out involving the injection of a superheated liquid ethanol–water mixture into gaseous nitrogen using both the new model and the unified model.

The structure of this article is as follows: Sec. II introduces a comprehensive description of the data-driven phase change model. Section III outlines the computational setup established for the injection flow simulations. In Sec. IV, there is a careful evaluation of the data-driven model's performance, along with a comparative assessment vs the unified model. Finally, in Sec. V, the key takeaway points from this study are summarized.

II. MODEL DEVELOPMENT

The process of developing the deep learning-based approach involves three main steps. First, a database of VLE was created to account for non-ideal mixing effects in a multi-species system across a wide range of thermodynamic conditions. Next, a deep feedforward neural network (DFNN) was fine-tuned with optimal hyperparameters to enable accurate and efficient VLE predictions. Finally, the DFNN was incorporated into the flow solver to replace the iterative VLE calculator, and the rate of phase transition was calculated using the predicted phase equilibrium. This phase change modeling approach is referred to as the DFNN model from here on. It is noted

that in this work, the deep neural network is essentially a complicated curve fit to data generated by the solution of the vapor–liquid equilibrium (VLE) model. The details of these three steps are outlined below.

A. Vapor–liquid equilibrium calculation

In this section, we will outline the methodology for determining the VLE of a two-phase liquid–gas system, which can handle mixtures of any composition, whether they are ideal or non-ideal. In this framework, the components in the system are categorized into two groups. The first category consists of components that undergo phase transitions (henceforth called active components), typically including the components in the fuel. However, the second group consists of species that maintain their phase (henceforth called inert components), such as dry air. The state of equilibrium at any given moment is determined using the following set of basic equations:

First, conservation of mass for all active components is expressed as

$$X_{f,i} = X_{l,i}^0 + X_{v,i}^0 = X_{l,i}^1 + X_{v,i}^1, \quad (1)$$

where X symbolizes the mole fraction, with subscripts v , l , and f indicating the vapor part, liquid part, and the entirety of the fuel, superscripts 0 and 1 denote the current and equilibrium conditions, and i marks the i th species among the active ones.

Then, the vapor–liquid equilibrium relationship for every active species is defined by the following equation:

$$y_i P = \gamma_i x_i P_{sat,i}, \quad (2)$$

where P stand for local pressure, y_i and x_i represent the mole fraction of species i in the vapor part and liquid part, respectively, γ_i is the activity coefficient of species i , and $P_{sat,i}$ is the saturation pressure of species i . In this formulation, the liquid phase is considered a non-ideal mixture, while the gas phase is considered an ideal gas. This assumption is valid at low pressures relative to the critical point, which aligns with the conditions discussed in this study. The activity coefficients of each species were determined using the UNIFAC mixing rule.¹⁵

By introducing the fraction of liquid phase in the system, L_f , Eqs. (1) and (2) can be combined as

$$\begin{aligned} X_{f,i} &= X_{l,i}^1 + X_{v,i}^1 = x_i^1 L_f^1 + y_i^1 (1 - L_f^1) \\ &= x_i^1 \left(L_f^1 + \frac{\gamma_i P_{sat,i}^1}{P^0} (1 - L_f^1) \right). \end{aligned} \quad (3)$$

Finally, energy conservation in the system, including both active and inert components, is expressed as

$$\begin{aligned} \sum_{i=1}^m (Y_{l,i}^0 h_{l,i}^0 + Y_{v,i}^0 h_{v,i}^0) + \sum_{j=1}^n Y_{g,j} h_{g,j}^0 \\ = \sum_{i=1}^m (Y_{l,i}^1 h_{l,i}^1 + Y_{v,i}^1 h_{v,i}^1) + \sum_{j=1}^n Y_{g,j} h_{g,j}^1, \end{aligned} \quad (4)$$

where Y stands for mass fraction, h stands for specific enthalpy, m and n represent the counts of active and inert components, the subscript g signifies inert components members, and j marks the j th species within the inert category.

To solve Eqs. (3) and (4), a tailored algorithm was devised, as shown in Fig. 1. This algorithm starts from computing the mixture's bubble point T_{bub} and dew point T_{dew} , guided by the subsequent formulations,

$$\sum_{i=1}^m X_{f,i} P_{sat,i}^{T_{bub}} / P_0 = 1, \quad (5)$$

$$\sum_{i=1}^m X_{f,i} P_0 / P_{sat,i}^{T_{dew}} = 1. \quad (6)$$

Subsequently, the maximum and minimum temperatures for achieving the full transformation of all the active components into liquid and vapor phases, denoted as T_{max} and T_{min} , are determined,

$$\begin{aligned} \sum_{i=1}^m (Y_{l,i}^0 h_{l,i}^0 + Y_{v,i}^0 h_{v,i}^0) + \sum_{j=1}^n Y_{g,j} h_{g,j}^0 \\ = \sum_{i=1}^m Y_{f,i} h_{f,i}^{T_{min}} + \sum_{j=1}^n Y_{g,j} h_{g,j}^{T_{min}}, \end{aligned} \quad (7)$$

$$\begin{aligned} \sum_{i=1}^m (Y_{l,i}^0 h_{l,i}^0 + Y_{v,i}^0 h_{v,i}^0) + \sum_{j=1}^n Y_{g,j} h_{g,j}^0 \\ = \sum_{i=1}^m Y_{f,i} h_{f,i}^{T_{max}} + \sum_{j=1}^n Y_{g,j} h_{g,j}^{T_{max}}. \end{aligned} \quad (8)$$

Afterward, two determinations are made according to the calculated temperatures. Should T_{bub} exceed T_{max} , it implies all active species equilibrate as the liquid phase. Conversely, if T_{dew} falls beneath T_{min} , equilibrium dictates a full transition of active species to the vapor phase. In scenarios where neither condition prevails, signifying coexistence of liquid and vapor phases at equilibrium, a vapor–liquid equilibrium scenario arises. Resolution of this VLE state is facilitated through the iterative algorithm illustrated in Fig. 1.

The algorithm mentioned above has been implemented into our in-house code. An iteration convergence threshold of 1.0×10^{-5} was used to ensure that iterative convergence was achieved. To validate the

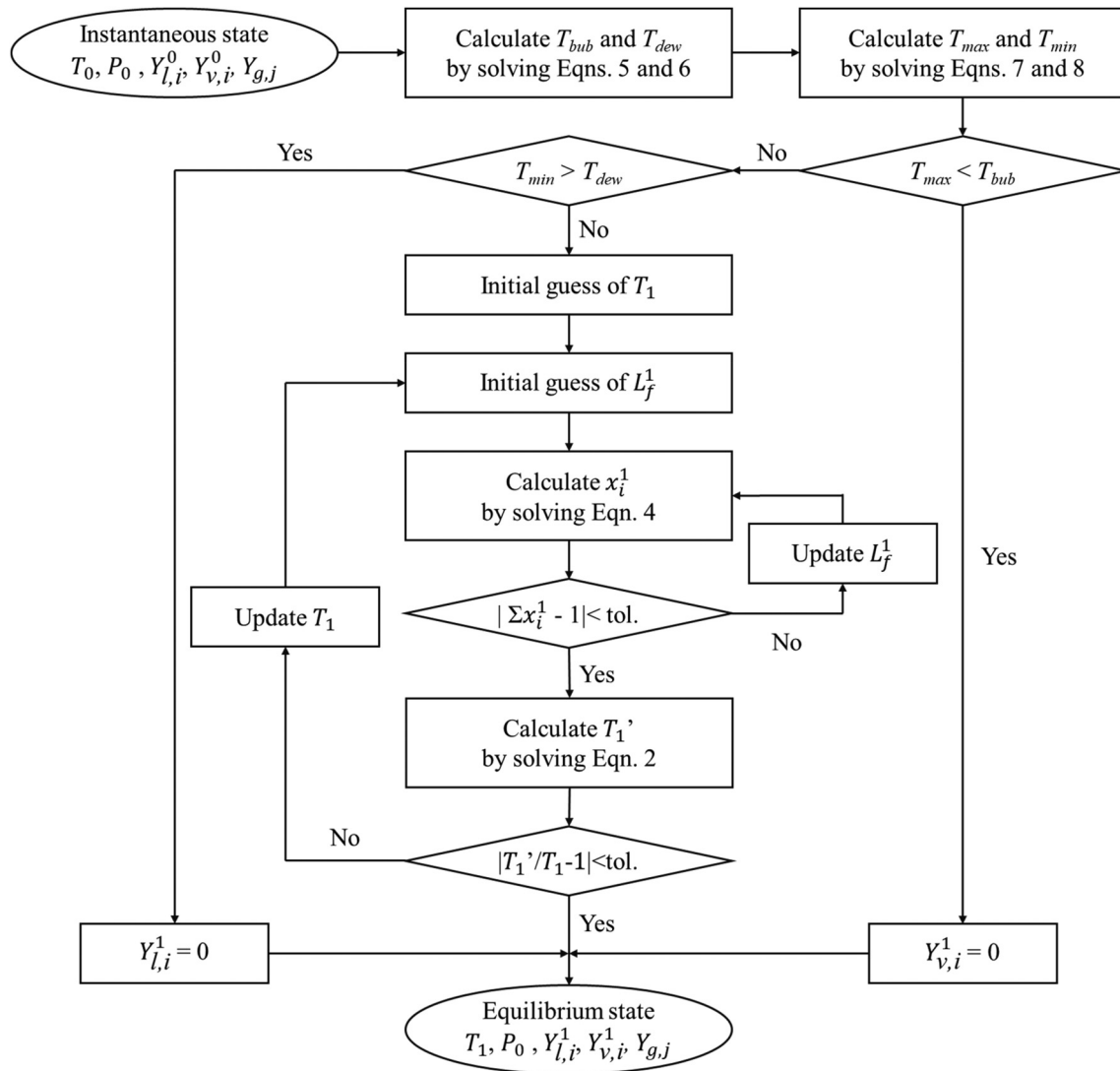


FIG. 1. Algorithm of vapor–liquid equilibrium calculation in a multi-component system.

solution, Fig. 2 compares the bubble and dew points obtained from experiments and our in-house solver. We studied binary mixtures of ethanol and water (on the left) and ethanol and n-heptane (on the right) as they are typical mixtures that show significant deviations from ideality. The experimental data for the ethanol–water mixture was reported in Ref. 16, and for the ethanol–heptane mixture in Refs. 17–19. The predicted results by the VLE solver matched well with the experimental data for both the bubble and dew points, representing the non-ideal mixing behavior and the azeotrope concentration accurately.

To further test the accuracy of the solver in handling flash calculations, we examined additional cases using the VLE solver and ASPEN Plus.²⁰ We looked at tertiary mixtures consisting of ethanol, water, and nitrogen, which represent the injection of an ethanol–water mixture into nitrogen. In this system, ethanol and water were considered active species, while nitrogen was considered inert. We ran calculations under various thermodynamic conditions, including pressure, temperature, and species fractions. The details of the calculations are summarized in Table I. The results showed that the liquid fraction predicted by the VLE solver closely matched the results from ASPEN Plus. Across the tested conditions, the difference in predicted liquid fraction between the VLE solver and ASPEN Plus was typically below 1%, with the maximum difference being less than 3%. Overall, both the results in Fig. 2 and the data in Table I demonstrate the reliability of the VLE solver in handling phase equilibrium calculations for non-ideal multi-species mixtures.

The validated VLE solver was then used to create a database of phase equilibrium for the ethanol–water–nitrogen tertiary system. Each data point in the database represents a combination of pressure, temperature, and mixture compositions. The pressure range covered was 10–500 Pa, and the temperature range was 250–420 K, encompassing the thermodynamic conditions in the entire computational domain of the injection flow simulations, which will be discussed in Sec. III. A total of 100,000 data points were created through random sampling in the parameter space.

B. Deep feedforward neural networks

The study uses a type of deep neuron network called DFNN to estimate the VLE for faster phase change calculations in CFD simulations. DFNN consists of multiple layers of neurons, with each layer fully connected to the previous one. The model is represented by an acyclic architecture, initiating with an input layer populated by D_{in} neurons, progressing through l hidden layers, and culminating in an output layer with D_{out} neurons, as illustrated in Fig. 3. In DFNN, each neuron in a layer receives the output from the previous layer, calculates a weighted sum, and performs an activation function operation to produce the output for the current layer. The vector of neurons in the i th layer, $\mathbf{h}^{(i)}$, is given as follows:

$$\mathbf{h}^{(i)} = \sigma_i(\mathbf{W}^{(i)}\mathbf{h}^{(i-1)} + \mathbf{b}^{(i)}). \quad (9)$$

In the above-mentioned expression, $\mathbf{W}^{(i)}$ is a weight matrix, $\mathbf{b}^{(i)}$ is a bias vector, and σ_i is the activation function, which serve a crucial role by infusing nonlinearity into the neural network. Some widely used activation functions include rectified linear unit (ReLU), hyperbolic tangent (tanh), and sigmoid.¹⁴

Upon the input dataset being introduced to the neural network, it traverses sequentially through each hidden layer. The ensemble of model parameters can be represented as

$$\Theta = \{\mathbf{W}^{(i)}, \mathbf{b}^{(i)}; i = 1, 2, \dots, l + 1\}. \quad (10)$$

Throughout the training phase, these parameters are adjusted to minimize a predefined objective function J as shown below. Specifically, the mean squared error (MSE) is used as the loss metric as a quantitative measure of the difference between the network prediction and the actual target output.

$$J(\Theta) = \frac{1}{N_{train}} \sum_{j=1}^{N_{train}} (f^*(\mathbf{x}_j) - f(\mathbf{x}_j; \Theta))^2, \quad (11)$$

where N_{train} is the number of training samples, f^* is the target function that the neural network approximates, and f is the neural network's

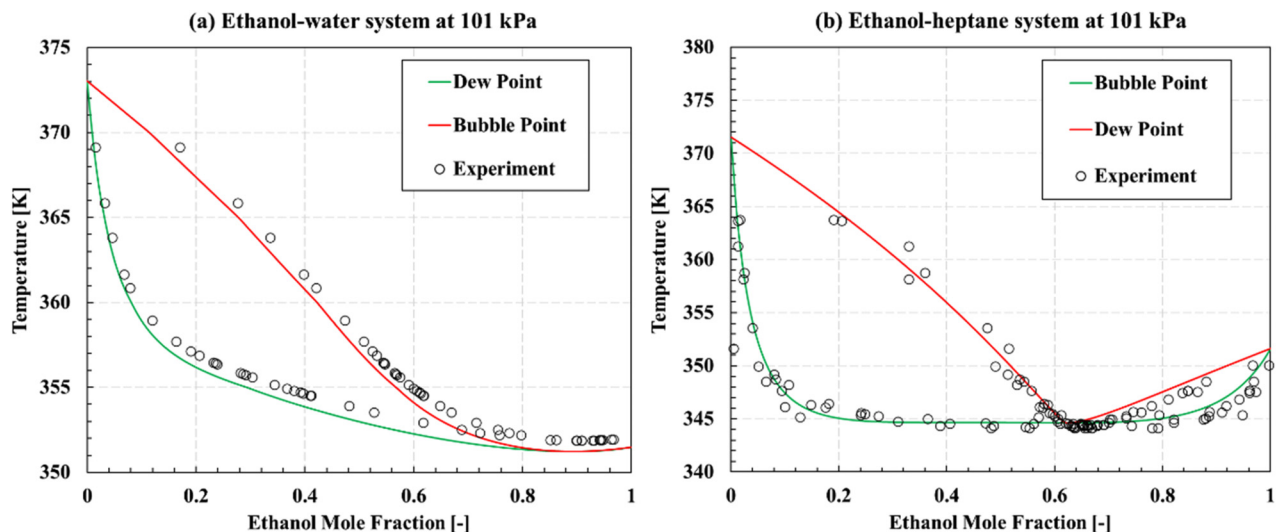


FIG. 2. Bubble and dew points calculated by the VLE solver compared with experimental measurement.

TABLE I. Comparison of flash calculation results by VLE solver and ASPEN Plus.

Case No.	P (kPa)	T (K)	Ethanol mole frac.	Water mole frac.	Nitrogen mole frac.	Predicted liquid fraction		
						ASPEN Plus	VLE solver	Bias (%)
1	136.45	339.81	0.595	0.106	0.299	0.439	0.445	0.6
2	118.66	330.61	0.297	0.224	0.479	0.285	0.288	0.3
3	51.90	308.10	0.801	0.006	0.193	0.736	0.737	0.1
4	249.42	348.29	0.237	0.350	0.413	0.394	0.397	0.3
5	265.36	362.70	0.489	0.319	0.192	0.551	0.557	0.6
6	190.04	377.04	0.038	0.852	0.110	0.639	0.639	0.0
7	58.37	331.64	0.335	0.452	0.213	0.401	0.402	0.1
8	446.55	382.92	0.699	0.060	0.242	0.148	0.177	2.9
9	250.05	375.61	0.979	0.015	0.006	0.689	0.701	1.2
10	50.90	300.23	0.173	0.323	0.503	0.410	0.410	0.0

prediction. Evaluating the performance of a trained neural network entails calculating the MSE on a validation dataset that comprises unseen examples, intentionally excluded from the training set to maintain independence in assessment.

In order to connect the DFNN model to a CFD solver, the selection of variables in the input and output layers depends on the solver's formulation. For this study, the input layer comprises six variables $\{P_0, T_0, Y_1^0, Y_2^0, Y_{i,1}^0, Y_{i,2}^0\}$, representing the instantaneous state of an ethanol–water–nitrogen tertiary mixture system. Meanwhile, the output layer contains a group of two variables $\{Y_{i,1}^1, Y_{i,2}^1\}$, representing the corresponding equilibrium state. The Adam optimization algorithm,²¹ which is a gradient-based method, is utilized to obtain the network parameters, with a fixed learning rate of $\lambda_{lr,ini} = 0.001$. The VLE database, as discussed in Sec. II A, was split into training and validation sets in an 80:20 ratio. Each set's data were standardized using the mean and standard deviation. This practice stabilizes the learning dynamics and mitigates biases that might arise due to variations in variable measurement scales or units. The training was carried out using PyTorch²² and scikit-learn.²³

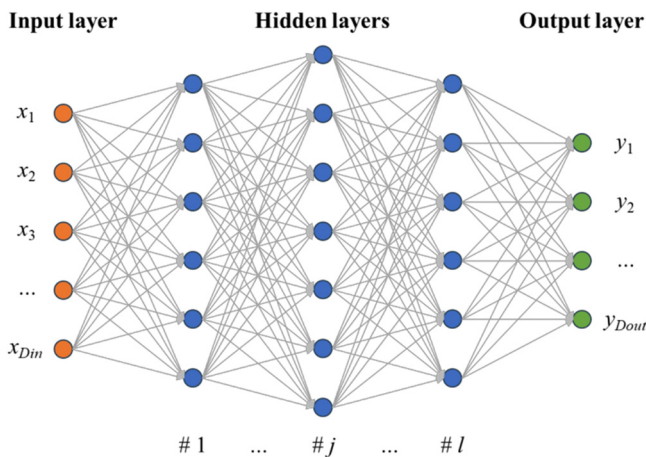


FIG. 3. Schematic representation of a DFNN with I hidden layer.

In order to optimize the topology of the DFNN, a parametric study was conducted using different activation functions, numbers of hidden layers, and numbers of neurons per layer. Figure 4 shows the learning process with the different configurations. The MSE of the training set is represented by solid lines, while the MSE of the validation set is represented by dashed lines. It was found that the difference between the two values is negligible in all cases.

Figure 4(a) compares the learning curve with three activation functions: sigmoid, tanh, and ReLU. The results indicate that all three activation functions exhibit a declining MSE with increased iterations. The performance of the ReLU function and tanh function are comparable after 5000 iterations, with about five times lower MSE compared with the sigmoid function. Due to its simple formulation, which makes it easy to integrate into a CFD solver, ReLU was chosen as the activation function for the remainder of the study. Figure 4(b) shows the learning curve with different numbers of hidden layers, L , with the activation function of ReLU and the number of neurons per layer fixed at 16. The results show that an increase in L introduces more nonlinearity to the network and results in some fluctuations in the learning process, with only a slight decrease in MSE. For the sake of a simpler network structure, the number of hidden layers was fixed at 2 for the remainder of the study. Figure 4(c) compares the learning curve with different numbers of neurons per layer, n , with the activation function of ReLU and the number of hidden layers fixed at 2. The results indicate that an increase in n leads to considerable improvement in terms of MSE. However, with n set at 64, the learning curve suffers from drastic fluctuations. As a result, the number of neurons per layer was fixed at 16 for the remainder of this work.

The final topology of the DFNN used in this study is determined based on the parametric study, with the activation function of ReLU, two hidden layers, and 16 neurons per layer.

In order to assess the accuracy of the DFNN's predictions, Fig. 5 displays regression plots that compare the output variables with the predicted values by DFNN on the y-axis and the expected values on the x-axis. Each data point in the validation dataset is represented by a scatter on the plot, and the R^2 -scores for each variable on the validation dataset are also included. Overall, the predictions made by DFNN closely match the expected results, with R^2 -scores exceeding 0.999 for each output variable.

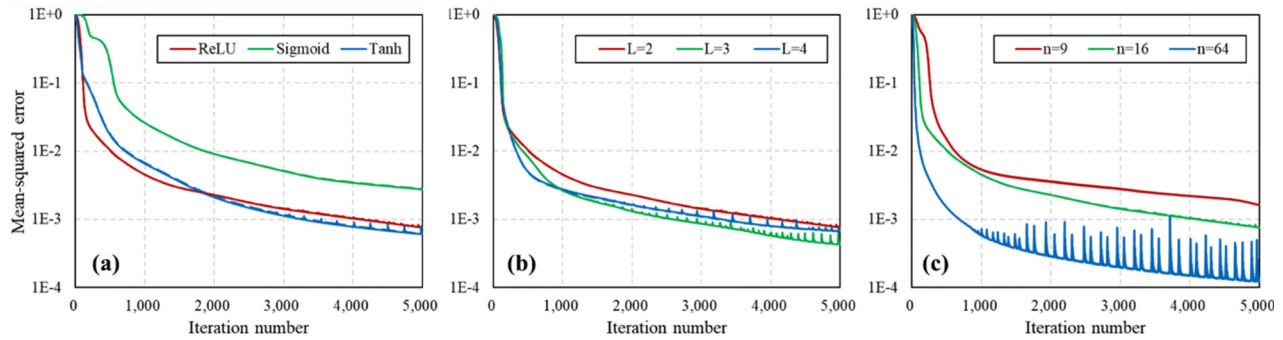


FIG. 4. Parameter optimization of the model. (a) activation function, (b) number of hidden layers, and (c) number of neurons per layer.

To assess the computational efficiency of the DFNN, Fig. 6 provides a comparison of the time taken to generate output between the VLE solver and the DFNN model. The indicated times correspond to generating up to 500 random data points. The calculations were performed on a single processor, with the only difference being the method of calculation. It is important to note that the y-axes are different for the two methods. It is shown that the computational cost is approximately four orders of magnitude lower with the use of DFNN. In summary, Figs. 5 and 6 demonstrate the significant advantage of using DFNN in reducing computational costs while maintaining high calculation accuracy when predicting VLE.

C. Integration into CFD solver

Finally, the DFNN model is incorporated into a CFD solver, specifically CONVERGE version 3.0, via a UDF. For every discrete time increment and at every grid cell, the flow characteristics is computed and then channeled as input data to the DFNN, which updates the corresponding phase equilibrium state. The rate of phase change is subsequently calculated in the same manner as in the unified model. Specifically, phase change is treated as a linear relaxation toward equilibrium. Given the current and the equilibrium states, the rate of phase change for each active component is determined employing the subsequent formulation,

$$dY_{l,i} = (Y_{l,i}^1 - Y_{l,i}^0)(1 - e^{-dt/\theta_i}). \quad (12)$$

The characteristic relaxation time attributed to each active component, symbolized as θ_i , is formulated as

$$\theta_i = F\theta_0\alpha^{-0.54} \left(\frac{\sum_{i=1}^m \gamma_i x_i P_{sat,i} - P_0}{\sum_{i=1}^m x_i P_{crit,i} - \sum_{i=1}^m \gamma_i x_i P_{sat,i}} \right)^{-1.76}, \quad (13)$$

where $P_{crit,i}$ is the critical pressure of species i . The value of θ_0 was kept the same as the one used in the unified model, which is 3.84×10^{-7} s. A time factor F was utilized to represent the timescales of different phase change mechanisms. The entire process is summarized in Fig. 7.

III. SIMULATION SETUP

Numerical simulations were conducted to assess the effectiveness of the DFNN model compared to the unified model. The simulation setup details are presented in this section.

A. Injector description

A single-hole injector that was adapted from a standard five-hole gasoline direct injector was studied. The injector's internal geometry

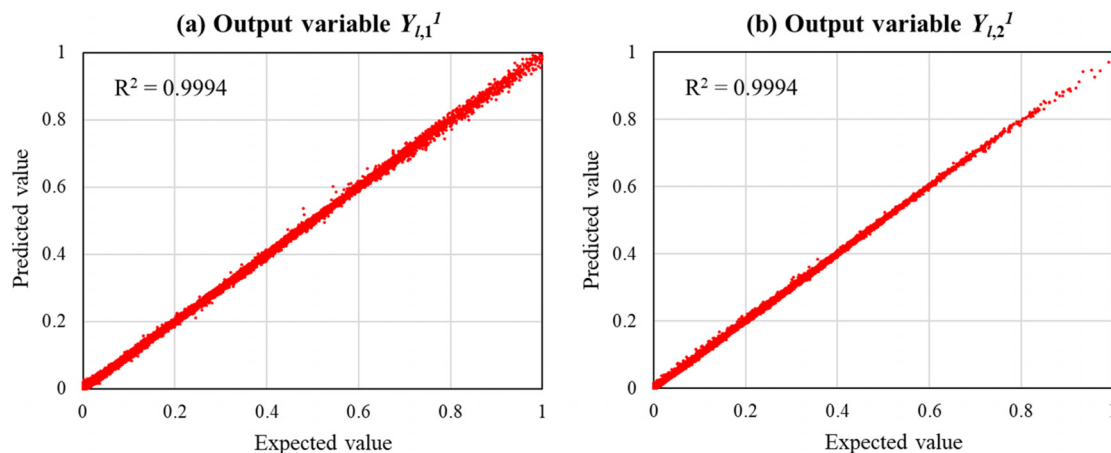


FIG. 5. Regression plots of output variables. The R^2 -score on validation data is indicated for each variable.

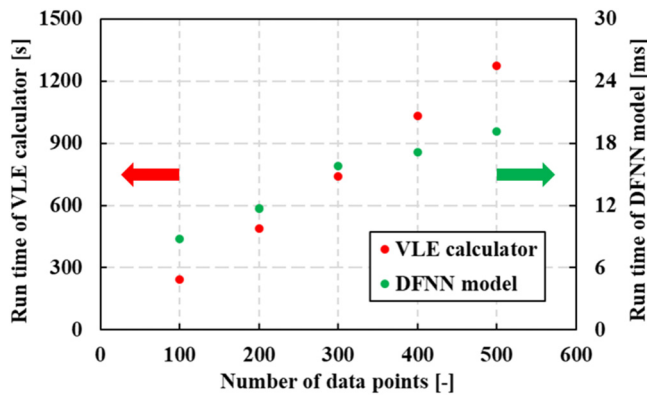


FIG. 6. The run time for VLE calculations using the VLE solver and DFNN model. Note the y-axes are different for the two methods.

details can be found in previous research.^{24,25} For the simulations in this study, a representative model of the injector's internal structure was used. In Fig. 8, the nozzle geometry and the computational domain are shown. The orifice comprises an inner hole and a counter-bore. The counterbore has a nominal diameter and length of 0.46 and 0.42 mm, respectively, while the inner hole has a nominal diameter and length of 0.18 and 0.16 mm, respectively. The orifice drill angle is 15.6° in relation to the injector axis. Downstream of the injector body, a hemispherical chamber with a radius of 5.0 mm was included to replicate the stationary environment in which the injector operates. A fixed needle lift of $50\ \mu\text{m}$ was utilized to represent full needle valve opening.

The computational domain had a base grid size of $160\ \mu\text{m}$. We used fixed embedded refinements with a minimum grid size of $10\ \mu\text{m}$ to resolve the orifices, following previous work.²⁶ Adaptive mesh refinement (AMR) was used to locally refine the mesh when the local velocity variations in adjacent computational cells exceeded $1.0\ \text{m/s}$, aiming for a minimum cell size of $10\ \mu\text{m}$ in all other locations. This mesh strategy resulted in a peak cell count of around 6×10^6 across all cases.

B. Boundary conditions

The injection fluid used was a mixture of ethanol and water, known for its non-ideal behavior with maximum positive deviation.

Specifically, the study focused on a 95% ethanol concentration by weight (E95), which is the azeotropic composition at 100 kPa. The injector and fuel temperature was set at 383 K, and the injection pressure at 10 MPa. Nitrogen was used as the ambient gas, with an ambient temperature of 383 K and ambient pressure at 100 kPa. As the liquid injection fluid exited the nozzle, it became superheated based on the specified thermodynamic conditions in this study. A summary of the operating conditions can be found in Table II.

C. Computational setup

The multiphase flow was modeled using the HMM,²⁷ premised on the principle that all phases within a computational cell share common temperature, pressure, velocity, and density. Unlike interface tracking methods, this approach is characterized by a diffused liquid-gas interface. Phase changes were modeled using either the DFNN model developed in this work, or the unified model specified in Ref. 12. Turbulence effects were accounted for through a transient Reynolds-averaged Navier-Stokes (RANS) framework, utilizing the RNG $k-\varepsilon$ turbulence model²⁸ for closure. The gas species were described using the Peng-Robinson EOS,²⁹ while the liquid phase was modeled as compressible using an exponential barotropic fluid assumption.

The computational simulations were executed utilizing a density-dependent transient solution algorithm over a span of $100\ \mu\text{s}$, deliberately selected to ascertain that the flow dynamics in proximity to the nozzle attained a near-steady-state equilibrium prior to the simulations' termination. To couple pressure and velocity, a modified pressure implicit splitting of operators (PISO) method³⁰ was used. A flexible time-stepping method was utilized, governed by a convective Courant-Friedrichs-Lewy (CFL) criterion set at 0.25 and a Mach number-based CFL criterion at 1.0. Spatial discretization was achieved using a second-order central difference scheme. Standard law-of-wall conditions were applied at all wall boundaries. Details of the CFD setup are summarized in Table III.

D. Model validation

The DFNN model's ability to predict VLE has been demonstrated in Sec. II. This subsection aims to confirm the accuracy of the computational setup for simulating the injection flow. To do so, the simulation results from the unified model and the DFNN model are compared with experimental jet morphology obtained through

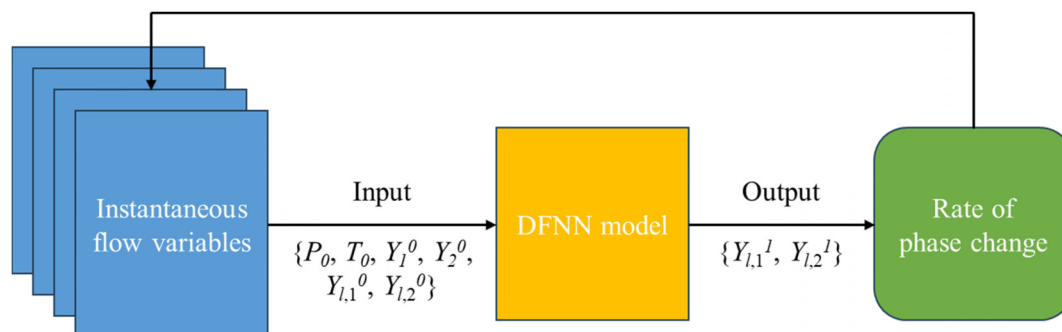


FIG. 7. Flowchart summarizing the integration of DFNN model in the CFD solver.

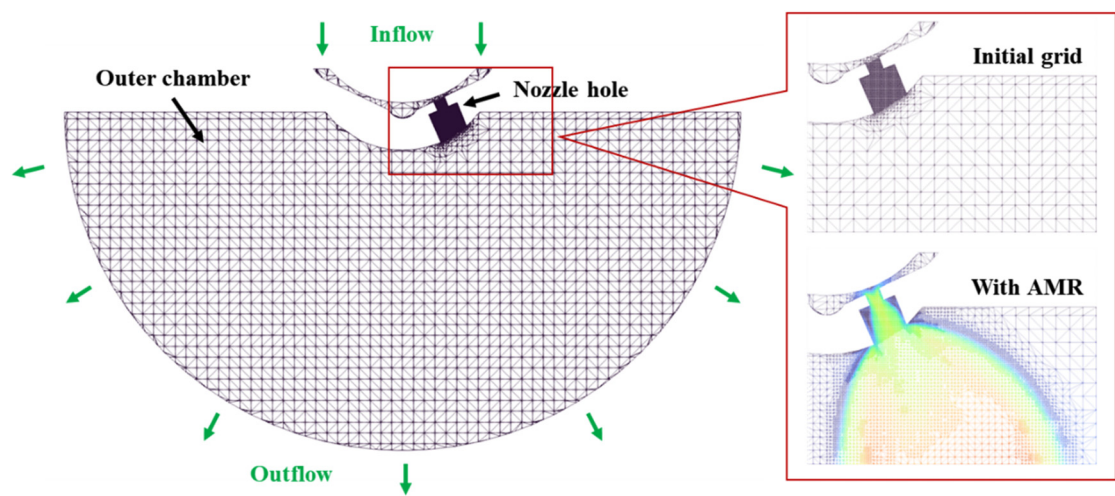


FIG. 8. Nozzle geometry and computational domain of the single-hole GDI injector.

high-speed imaging, as shown in Fig. 9. Details of the measurement technique can be found in Ref. 31. The experiments used pure ethanol and looked at three different superheat degrees ($Rp = P_{sat}/P_{amb}$), covering subcooled to flare flashing conditions. It was observed that as Rp increases, the injection fluid moves from subcooled to superheated, resulting in significant radial expansion of the jet, a characteristic feature of flash boiling jets.³² Both the unified model and the DFNN model accurately represented this phenomenon. Furthermore, under all three conditions, the two models provided similar predictions for the jet morphology, which aligned well with the experimental results. Overall, these findings demonstrate the validity of the simulation setup.

IV. RESULTS AND DISCUSSION

Figure 10 displays a cross section view comparing the predicted results of the unified model and the DFNN model in terms of vapor concentration, temperature, and pressure. It is important to note that all the results presented in this section correspond to the end of the simulations, when the flow has become relatively steady. The DFNN model generally predicted lower vapor generation within the jet, particularly along the jet axis. Additionally, the temperature predicted by the DFNN model is significantly higher. Moreover, a low-pressure core was observed near the orifice exit, which is linked to the formation of an under-expanded two-phase flow within the orifice and its

vicinity.³² In the case of the unified model, the expansion of the under-expanded flashing jet was so intense that it resulted in the formation of a stationary shock wave structure, which was not fully resolved in the case with the DFNN model.

In Fig. 11, the vapor concentration of ethanol and water is illustrated in the cross-sectional view, as predicted by the unified model and the DFNN model. The normalized vapor mole fraction, which is the species' vapor mole fraction in the computational cell divided by its nominal mole fraction in the fuel, was used to represent the progress of vaporization for each species. The unified model predicted a much faster vaporization of ethanol compared to water. However, the DFNN model predicted a similar percentage of species vaporized for the two species, which aligns well with the expected physics. This is because the composition of the ethanol–water mixture is near the azeotrope concentration, and the two species vaporize at the same pace. The difference in predictions between the unified model and the DFNN model can be attributed to the fact that the VLE calculation in the unified model is based on Raoult's law, while the VLE solution in the DFNN model takes the non-ideal mixing effect into account. Overall, the results demonstrate that the DFNN model performs well in handling non-ideal mixtures.

The results are further quantified in Fig. 12, which displays the normalized vapor mass fraction of ethanol and water along the jet axis

TABLE II. Simulation condition.

Item	Value
Injected fluid	E95
Ambient gas	Nitrogen
Injection temperature	383 K
Injection pressure	10 MPa
Ambient temperature	383 K
Ambient pressure	100 kPa

TABLE III. Summary of the computational setup.

Item	Value
Multiphase flow model	Homogeneous mixture model
Phase change model	DFNN model/Unified model
Turbulence model	Unsteady RANS RNG $k-\epsilon$ model
Liquid phase EOS	Barotropic fluid
Gas/vapor phase EOS	Peng–Robinson
Convection CFL number	0.25
Mach-based CFL number	1.0

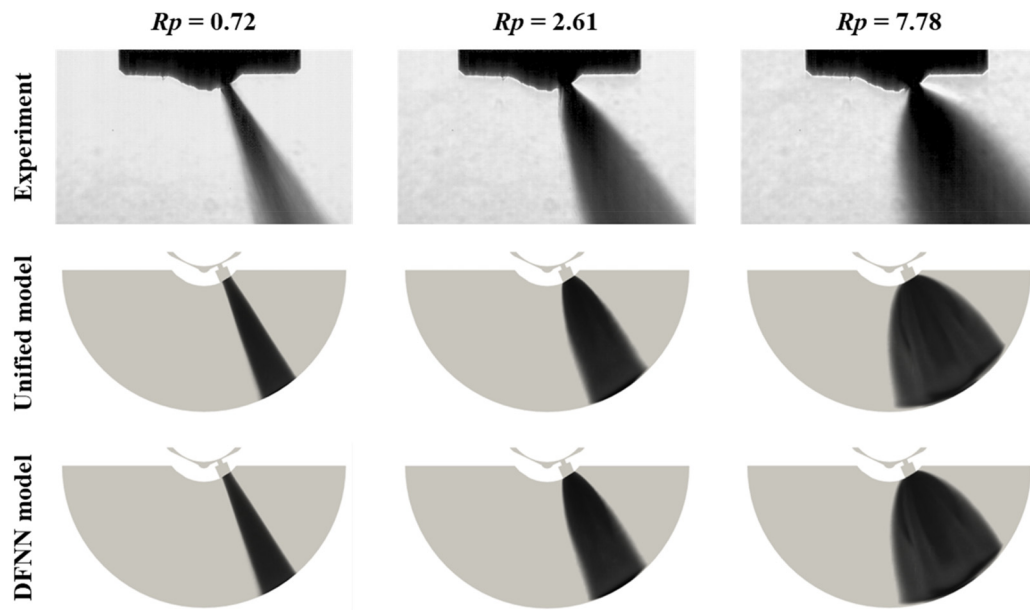


FIG. 9. Comparison of measured and predicted spray morphology under different conditions. $Rp = P_{sat}/P_{amb}$ represents the superheat degree. The injection fluid was pure ethanol and the ambient gas was nitrogen.

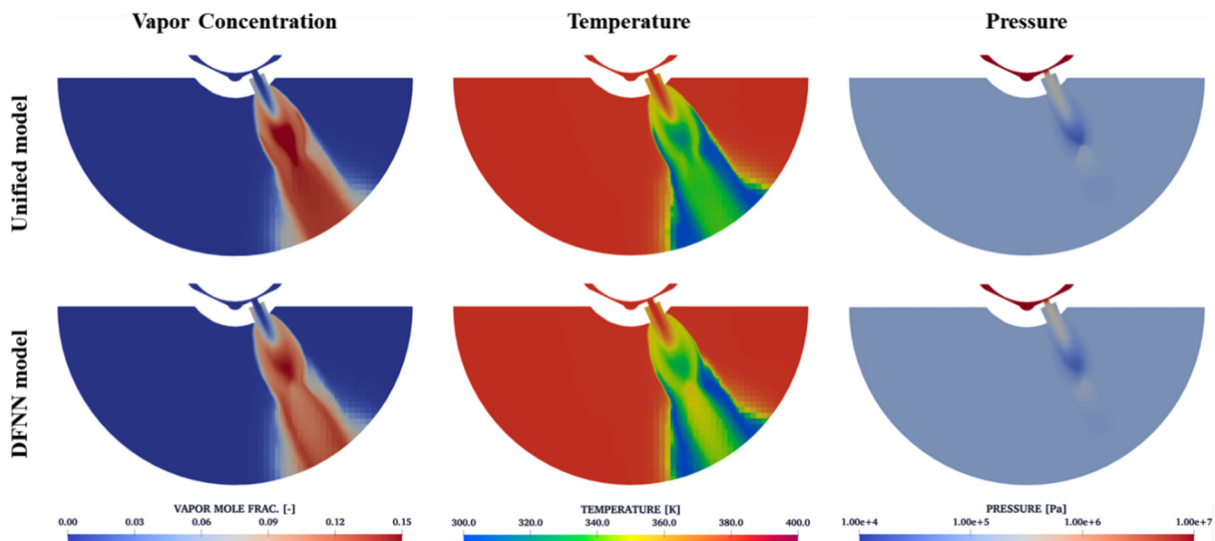


FIG. 10. Cross-sectional view of vapor concentration, temperature, and pressure distributions predicted by the unified model (top) and the DFNN model (bottom).

(left) and at 1 mm from the injector tip in the jet's radial direction (right). Using the DFNN model, the difference in normalized vapor mole fraction is less than 20% between the two species. However, the unified model shows a fourfold faster vaporization of ethanol compared to water. This results in the over-prediction of phase change in the flashing jet and a stronger flow expansion, as shown in Fig. 10.

In order to evaluate how well the two models predicted mixing-driven vaporization, Fig. 13 illustrates a scatterplot of temperature vs

fuel mass fraction for all the computational cells in the outer chamber. The scatter points are color-coded based on the percentage of fuel vaporized. The plot demonstrates that as the local fuel mass fraction deviates from unity and ambient gas mixes in, the percentage of fuel vaporized increases gradually, leading to a decrease in temperature. When the percentage of fuel vapor reaches 100%, indicating complete vaporization, the temperature is at its lowest point. Subsequently, as the fuel mass fraction decreases further, the temperature increases

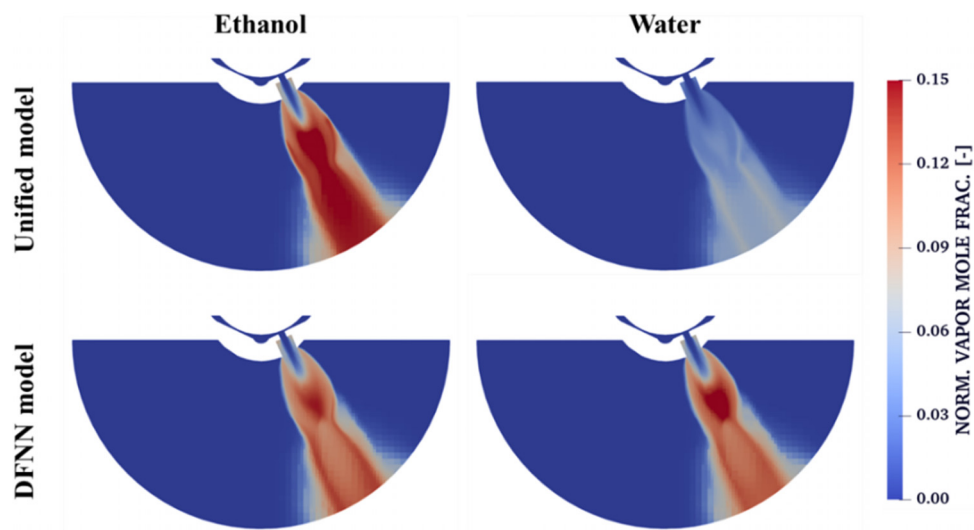


FIG. 11. Normalized vapor concentration of ethanol and water calculated using the unified model (top) and the DFNN model (bottom).

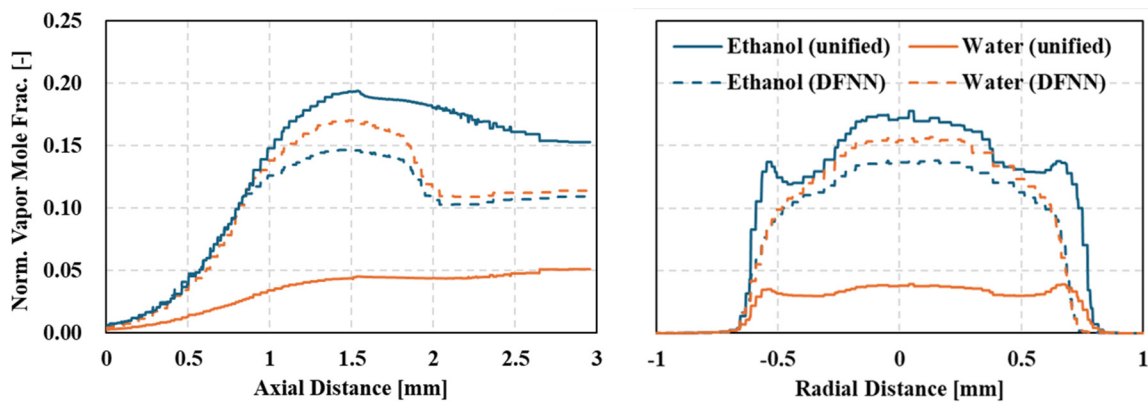


FIG. 12. Normalized vapor concentration of ethanol and water along the jet axis (left) and at 1 mm from the injector tip in the jet's radial direction (right), as calculated using the unified model and the DFNN model.

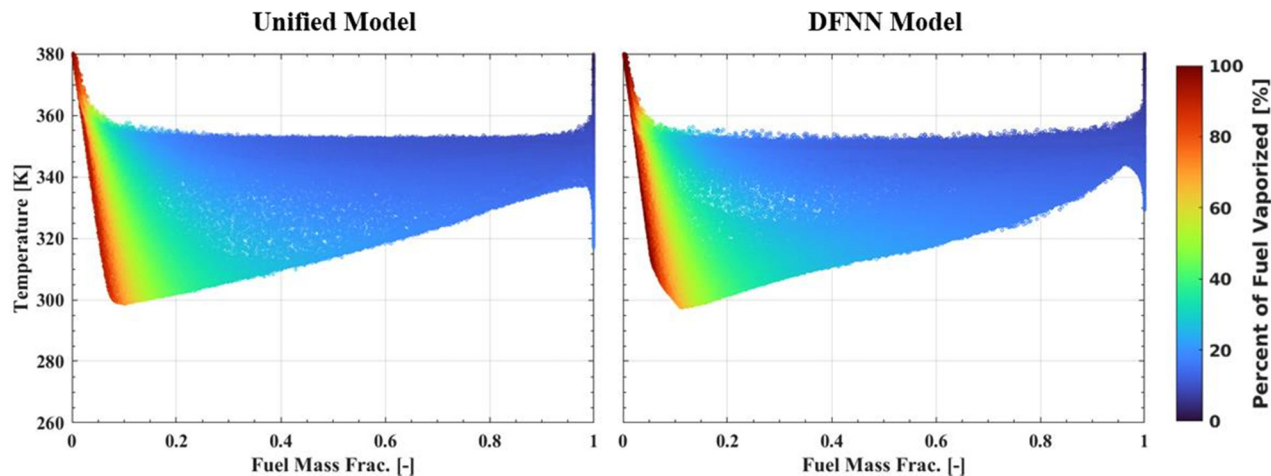


FIG. 13. Scatter plot of temperature vs fuel mass fraction for all computational cells in the outer chamber, colored by the percentage of the fuel vaporized.

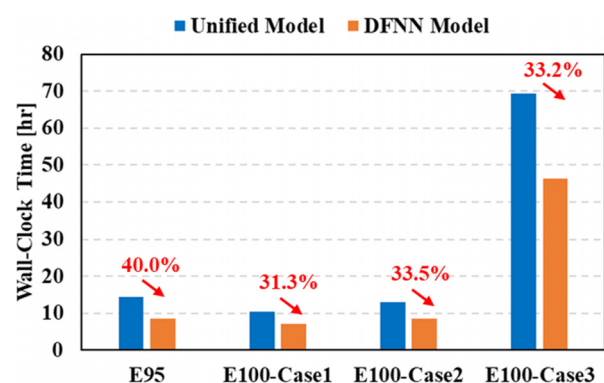


FIG. 14. Comparison of computational cost of the unified model and the DFNN model in different cases.

toward the ambient temperature. Furthermore, both models predicted a similar minimum temperature and the corresponding fuel mass fraction in the domain.

Figure 14 shows the computational costs of the two models used in the simulations discussed previously. This includes the case in Sec. IV and the three cases in Sec. III D where pure ethanol (E100) was studied. The DFNN model had significantly lower computational costs for the entire CFD simulations, benefitting from its advantage in solving the VLE, as demonstrated in Sec. II B. In cases with single-component injection fluid (E100), the wall-clock time was reduced by an average of 32.7% with the DFNN model. In the case with binary-component injection fluid (E95), the reduction in wall-clock time increased to 40.0%. This suggests that the DFNN model's advantage in reducing computational cost becomes even stronger with an increased number of compositions, indicating its potential in dealing with realistic fossil fuels such as gasoline and kerosene.

V. SUMMARY

In this study, a data-driven approach was proposed to model the liquid–vapor phase change in fuel injection flows. This approach was developed on top of a previously created iteration-based model and was integrated into a CFD solver. To evaluate the proposed methodology, CFD simulations of superheated liquid ethanol–water injection flow into ambient nitrogen were conducted using both the data-driven DFNN model and the iteration-based unified model. The key takeaway points are as follows:

1. A solver for vapor–liquid equilibrium (VLE) was developed to create a database for training a DFNN model. This model accounts for non-ideal mixing effects by considering the activity coefficients of each species. It was shown that the predictions of the VLE solver align well with the available experimental data for bubble and dew points, as well as flash calculation results. The VLE solver was used to create a database of phase equilibrium for an ethanol–water–nitrogen system.
2. A DFNN model was developed to approximate the calculation results of the VLE solver. This deep neural network takes instantaneous pressure, temperature, and composition of the system as input and calculates the corresponding phase equilibrium. A parametric analysis was conducted to optimize the hyperparameters of the neural network, including the activation function,

number of hidden layers, and neurons per hidden layer. With the optimal hyperparameters, a strong correlation between predicted values by the DFNN and expected values was achieved, with an R^2 -score above 0.999 for each output variable.

3. The DFNN model was integrated into a CFD solver in a robust manner. Simulations were then carried out to evaluate the performance of the DFNN model compared to the unified model. An ethanol–water mixture at azeotropic concentration injected into gaseous nitrogen was discussed. The results demonstrate that the DFNN model performed well in capturing the azeotropic effect, while the unified model did not. This is because the VLE calculation in the unified model is based on Raoult's law, while the VLE solution in the DFNN model takes the non-ideal mixing effect into account.
4. The computational efficiency of the DFNN model was evaluated. It was found that for VLE solutions, a runtime reduction of four orders of magnitude was achieved with the DFNN model compared to the iteration-based calculator. With respect to the CFD simulation cases in this study, a 30%–40% reduction in wall-clock time was achieved compared to the unified model. The DFNN model's advantage in reducing computational cost was found to be stronger with an increased number of compositions.

In general, this research demonstrates the advantage of the proposed approach in both calculation accuracy and computational efficiency when dealing with vapor–liquid equilibrium problems. This approach shows promise for injection flow simulations, especially for systems with a large number of compositions, such as sustainable aviation fuels.

ACKNOWLEDGMENTS

This research is sponsored by the National Natural Science Foundation of China under Grant No. 52306204 and 52076119, and the Young Elite Scientists Sponsorship Program by Shaanxi Association of Science and Technology under Grant No. 20230434.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Yanfei Li: Data curation (equal); Formal analysis (equal); Investigation (equal); Software (equal); Writing – original draft (equal). **Chenxiang Zhao:** Investigation (equal); Methodology (equal); Validation (equal); Writing – review & editing (equal). **Song Cheng:** Writing – review & editing (equal). **Hengjie Guo:** Conceptualization (equal); Funding acquisition (equal); Methodology (equal); Resources (equal); Supervision (equal); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

- ¹E. T. Baldwin *et al.*, “String flash-boiling in gasoline direct injection simulations with transient needle motion,” *Int. J. Multiphase Flow* **87**, 90–101 (2016).

- ²A. Pandal, J. M. Garcia-Oliver, and J. M. Pastor, "Eulerian CFD modeling of nozzle geometry effects on ECN Sprays A and D: Assessment and analysis," *Int. J. Engine Res.* **21**(1), 73–88 (2020).
- ³C. K. Mohapatra *et al.*, "Collaborative investigation of the internal flow and near-nozzle flow of an eight-hole gasoline injector (Engine Combustion Network Spray G)," *Int. J. Engine Res.* **24**, 2297 (2020).
- ⁴M. Manninen, V. Taivassalo, and S. Kallio, *On the Mixture Model for Multiphase Flow*, Technical Research Centre of Finland (VTT Publications, Finland, 1996).
- ⁵Z. Bilicki and J. Kestin, "Physical aspects of the relaxation model in two-phase flow," *Proc. R. Soc. London, Ser. A* **428**(1875), 379–397 (1990).
- ⁶D. P. Schmidt, S. Gopalakrishnan, and H. Jasak, "Multi-dimensional simulation of thermal non-equilibrium channel flow," *Int. J. Multiphase Flow* **36**(4), 284–292 (2010).
- ⁷H. Guo and R. Torelli, "Computational study of ECN Spray C via one-way coupling of internal nozzle flow and ensuing spray," *J. Aerosol Sci.* **174**, 106243 (2023).
- ⁸K. Saha, S. Som, M. Battistoni, Y. Li, E. Pomraning, and P. K. Senecal, "Numerical investigation of two-phase flow evolution of in- and near-nozzle regions of a gasoline direct injection engine during needle transients," *SAE Int. J. Engines* **9**(2), 1230–1240 (2016).
- ⁹S. K. Rachakonda *et al.*, "A computational approach to predict external spray characteristics for flashing and cavitating nozzles," *Int. J. Multiphase Flow* **106**, 21–33 (2018).
- ¹⁰R. Payri, J. Gimeno, P. Martí-Aldaravi, and M. Martínez, "Transient nozzle flow analysis and near field characterization of gasoline direct fuel injector using Large Eddy Simulation," *Int. J. Multiphase Flow* **148**, 103920 (2022).
- ¹¹H. Guo and R. Torelli, "On the effect of mixing-driven vaporization in a homogeneous relaxation modeling framework," *Phys. Fluids* **34**(9), 093304 (2022).
- ¹²H. Guo and R. Torelli, "A unified non-equilibrium phase change model for multicomponent mixtures," *Int. J. Heat Mass Transfer* **208**, 124063 (2023).
- ¹³K. J. Richards, P. K. Senecal, and E. Pomraning, *CONVERGE 3.0 Manual* (Convergent Science, Madison, WI, 2022).
- ¹⁴I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
- ¹⁵*Vapor-Liquid Equilibria Using Unifac: A Group-Contribution Method*, edited by A. Fredenslund, J. Gmehling, and P. Rasmussen (Elsevier, 1977).
- ¹⁶V. H. Álvarez, S. Mattedi, M. Iglesias, R. Gonzalez-Olmos, and J. M. Resa, "Phase equilibria of binary mixtures containing methyl acetate, water, methanol or ethanol at 101.3 kPa," *Phys. Chem. Liquids* **49**(1), 52–71 (2011).
- ¹⁷K. Katz and M. Newman, "Vapor-liquid equilibria for ethyl alcohol-n-heptane at low pressure," *Ind. Eng. Chem.* **48**(1), 137–141 (1956).
- ¹⁸J. D. Raal, R. K. Code, and D. A. Best, "Examination of ethanol-n-heptane, methanol-n-hexane systems using new vapor-liquid equilibrium still," *J. Chem. Eng. Data* **17**(2), 211–216 (1972).
- ¹⁹T. Tsuji, H. Matsuda, M. Kawakita, K. Kurihara, and K. Tochigi, "Vapor-liquid equilibrium (VLE) measurements of ethanol-heptane at isothermal (363.15, 393.15 and 423.15 K) and isobaric (101.33 kPa) conditions and correlation of liquid viscosity data," *Chem. Thermodyn. Therm. Anal.* **6**, 100041 (2022).
- ²⁰ASPEN Technology, *Aspen Plus®: Aspen Plus User Guide, Version 11.1* (Aspen Technology, Inc., Bedford, MA, 2001).
- ²¹D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980* (2024).
- ²²A. Paszke *et al.*, "Automatic differentiation in PyTorch," paper presented at the NIPS Autodiff Workshop, 2017.
- ²³F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.* **12**(85), 2825–2830 (2011).
- ²⁴H. Guo, Y. Li, X. Lu, Z. Zhou, H. Xu, and Z. Wang, "Radial expansion of flash boiling jet and its relationship with spray collapse in gasoline direct injection engine," *Appl. Therm. Eng.* **146**, 515–525 (2019).
- ²⁵H. Guo, Y. Li, B. Wang, H. Zhang, and H. Xu, "Numerical investigation on flashing jet behaviors of single-hole GDI injector," *Int. J. Heat Mass Transfer* **130**, 50–59 (2019).
- ²⁶M. Battistoni, S. Som, and D. E. Longman, "Comparison of mixture and multi-fluid models for in-nozzle cavitation prediction," *J. Eng. Gas Turbines Power* **136**, 061506 (2014).
- ²⁷J. C. Leung, "A generalized correlation for one-component homogeneous equilibrium flashing choked flow," *AIChE J.* **32**(10), 1743–1746 (1986).
- ²⁸V. Yakhot, S. A. Orszag, S. Thangam, T. B. Gatski, and C. G. Speziale, "Development of turbulence models for shear flows by a double expansion technique," *Phys. Fluids A: Fluid Dyn.* **4**(7), 1510–1520 (1992).
- ²⁹D. B. Robinson, D.-Y. Peng, and S. Y.-K. Chung, "The development of the Peng-Robinson equation and its application to phase equilibrium in a system containing methanol," *Fluid Phase Equilib.* **24**(1), 25–41 (1985).
- ³⁰R. I. Issa, "Solution of the implicitly discretised fluid flow equations by operator-splitting," *J. Comput. Phys.* **62**(1), 40–65 (1986).
- ³¹J. Zhang *et al.*, "Characterizing under-expansion behaviors induced by rapid phase change of flash-boiling jets," *Fuel* **329**, 125404 (2022).
- ³²H. Guo, L. Nocivelli, R. Torelli, and S. Som, "Towards understanding the development and characteristics of under-expanded flash boiling jets," *Int. J. Multiphase Flow* **129**, 103315 (2020).