



Machine learning tabulation of thermochemistry in turbulent combustion: An approach based on hybrid flamelet/random data and multiple multilayer perceptrons

Tianjie Ding, Thomas Readshaw, Stelios Rigopoulos*, W.P. Jones

Imperial College London Mechanical Engineering South Kensington Campus London SW72AZ, United Kingdom



ARTICLE INFO

Article history:

Received 30 October 2020

Revised 28 April 2021

Accepted 30 April 2021

Available online 24 May 2021

Keywords:

Turbulent flames

Machine learning

Artificial neural networks (ANNs)

Large Eddy Simulation (LES)

Probability density function (PDF) methods

Stochastic fields

ABSTRACT

A new machine learning methodology is proposed for speeding up thermochemistry computations in simulations of turbulent combustion. The approach is suited to a range of methods including Direct Numerical Simulation (DNS), Probability Density Function (PDF) methods, unsteady flamelet, Conditional Moment Closure (CMC), Multiple Mapping Closure (MMC), Linear Eddy Model (LEM), Thickened Flame Model, the Partially Stirred Reactor (PaSR) method (as in OpenFOAM) and the computation of laminar flames. In these methods, the chemical source term must be evaluated at every time step, and is often the most expensive element of a simulation. The proposed methodology has two main objectives: to offer enhanced capacity for generalisation and to improve the accuracy of the ANN prediction. To accomplish the first objective, we propose a hybrid flamelet/random data (HFRD) method for generating the training set. The random element endows the resulting ANNs with increased capacity for generalisation. Regarding the second objective, a multiple multilayer perceptron (MMP) approach is developed where different multilayer perceptrons (MLPs) are trained to predict states that result in smaller or larger composition changes, as these states feature different dynamics. It is shown that the multiple MLP method can greatly reduce the prediction error, especially for states yielding small composition changes. The approach is used to simulate flamelets of varying strain rates, one-dimensional premixed flames with differential diffusion and varying equivalence ratio, and finally the Large Eddy Simulation (LES) of CH₄/air piloted flames Sandia D, E and F, which feature different levels of local extinction. The simulation results show very good agreement with those obtained from direct integration, while the range of problems simulated indicates that the approach has great capacity for generalisation. Finally, a speed-up ratio of 12 is attained for the reaction step.

© 2021 The Combustion Institute. Published by Elsevier Inc. All rights reserved.

1. Introduction

In order to improve the efficiency and to reduce the pollutant emissions of combustion equipment, accurate modelling methods have to be employed in the design process. Some approaches for modelling turbulent combustion, such as the steady flamelet and flamelet-generated manifold (FGM) methods, are based on pre-computed simulations of laminar flames. On the other hand, several other methods require real-time computation of the chemical source term at every grid node and time step. These methods include Direct Numerical Simulation (DNS), Probability Density Function (PDF) methods, unsteady flamelet, Conditional Moment Closure (CMC), Multiple Mapping Closure (MMC), Linear Eddy Model

(LEM), Thickened Flame Model, the Partially Stirred Reactor (PaSR) method employed in OpenFOAM and the computation of laminar flames (more information about these methods can be found in Refs. [1–3]). Several of these methods benefit from greater generality and absence of modelling assumptions regarding the flame structure, but their application is hampered by the CPU time required for the integration of chemical kinetics, which requires the numerical integration of a large and usually stiff set of ordinary differential equations (ODEs). Even reduced mechanisms derived via methods such as Computational Singular Perturbation (CSP) [4], Rate-Controlled Constrained Equilibrium (RCCE) [5–7] or Directed-Relation Graph (DRG) [8] are computationally expensive if real-time computation of chemical kinetics is employed.

The bottleneck posed by the integration of chemistry can be alleviated by employing a form of tabulation, such that the reaction source term is obtained from a database of pre-computed chemistry integrations. The most straightforward tabulation approach

* Corresponding author.

E-mail address: s.rigopoulos@imperial.ac.uk (S. Rigopoulos).

is the Look-up Table (LUT) [9], where interpolation from a table of pre-computed values is performed. The memory requirements grow exponentially with the number of species, however, thus rendering this approach suitable only for very small mechanisms. A more advanced method includes the Intrinsic Low-Dimensional Manifolds (ILDM) [10] in combination with In-Situ Adaptive Tabulation (ISAT) [11], where the table is built during the simulation. While this approach can reduce the CPU time, it has considerable memory requirements and furthermore the speed-up relies on the new states being similar to previously encountered ones.

Another alternative is to use Artificial Neural Networks (ANNs), which are machine learning models that can be used for a wide range of tasks, including non-linear regression. For a given chemical mechanism, the species concentrations after a reaction time step are functions of the initial state, and ANNs can be trained to approximate these (unknown) functions. Compared with conventional tabulation methods, ANNs can provide accurate interpolation for highly nonlinear functions with low storage requirements and fast computation speed.

Christo et al. [12] were the first (to the authors' knowledge) to apply ANNs to combustion chemistry tabulation. In their work, an ANN was trained to represent a reduced three-step mechanism for $H_2/CO_2/O_2$ mixtures. Later, this ANN methodology was further applied to PDF simulations of turbulent flames [13,14]. The simulation results showed good agreement with the results obtained by direct integration (DI). These works showed the potential of ANNs in chemistry tabulation. However, the chemical mechanism employed was rather simple. The ANN tabulation method was further developed by Blasco et al. [15], who applied ANNs to model the temporal evolution of a 4-step combustion chemical system. Satisfactory results were obtained and large CPU time savings were reported. However, with an increasing number of species, the composition space expands exponentially, which makes it difficult to tabulate the whole space using a single ANN. Blasco et al. [16,17] proposed a partitioning method to solve this problem. In their work, the composition space was split into several subdomains based on mixture fracture or temperature, and then each subdomain was fitted by an individual ANN. This approach was applied to plug-flow reactor (PFR) and PaSR simulations, which showed reasonable results. Later, the Self Organising-Map (SOM) was introduced by them to cluster the composition space into several sub-domains [18].

In all of the works mentioned, the problem used for training data generation was the same as the one used to test the ANNs. As a result, every time the ANNs are applied to a new problem, the problem must be simulated beforehand with DI, thus negating any speed benefits. Ideally, ANNs should have good generalisation capacity so that they can be applied to a wide range of realistic problems. In order to achieve this goal, the training data must cover the appropriate composition space that is likely to be encountered in real simulations. Therefore, the next challenge was to generate a representative training dataset via simple methods. Sen et al. [19] trained ANNs with a dataset generated via a DNS of a laminar flame-vortex interaction, as well as via stand-alone LEM simulations in later works [20,21], and subsequently applied them to LES of syngas/air flames. The mechanism used in their works involved 11 species with 21 reaction steps. Chatzopoulos et al. [22] proposed a method of generating training data with a canonical problem comprising dynamic flamelet simulations, augmented with an artificial pilot to generate ignition conditions. The training data was clustered into 400 sub-domains using an SOM, and each subdomain was fitted by an individual MLP. The chemistry used was an RCCE-reduced mechanism that included 17 species. The ANNs were successfully applied to the DLR-A/B flames using the RANS-PDF method. The SOM-MLP methodology was further developed by Franke et al. [23] to include

extinction events and combined with LES-PDF. The LES-PDF-ANN methodology was applied to the simulation of the Sydney L turbulent flame, which features significant levels of local extinction and re-ignition, and very good agreement was obtained between the ANNs and DI. The SOM-MLP methodology was also applied by An et al. [24] to the simulation of the hydrogen/carbon monoxide/kerosene mixture in a rocket-based combustor, where a skeletal mechanism with 41 species was used. The training data was collected via a RANS simulation and then the ANNs were applied to a LES simulation, which showed good agreement with DI. In a recent work, Wan et al. [25] generated the training dataset via a canonical stochastic micromixing problem. Two ANNs were trained in their work, with one focussing on reactions with slow burning rate. The ANNs were applied to a DNS of a turbulent non-premixed syngas oxy-flame, and good agreement with DI was achieved.

Another line of works sought to use ANNs in the context of flamelet/progress variable models. In Kempf et al. [26], ANNs were used to represent steady laminar flamelet solutions and applied to the LES of Sandia flame D. Later, Ihme et al. [27] generated optimal ANNs for the steady flamelet model and applied them to the LES of a bluff-body swirl-stabilised flame. Hansinger et al. [28] developed a single deep ANN to represent all combustion quantities. The ANN was trained on flamelet/progress variable (FPV) tables and then employed in an LES of the Sydney/Sandia flame with an inhomogeneous inlet, which showed good agreement with the conventional tabulated FPV simulations. However, flamelet/progress variable models do not feature the very high chemistry integration CPU time demands raised by models where the chemistry is computed in real time, and therefore the need for tabulation is not as essential for that class of methods.

In addition to the tabulation of chemical kinetics, machine learning tools can also aid in other aspects of turbulent combustion modelling. Nikolaou et al. [29] used convolutional neural networks (CNN) to estimate the unfiltered progress variable field, so that filtered functions of the progress variable can be approximated on the LES mesh. Seltz et al. [30] also applied CNNs to predict unresolved source and transport terms. The CNN method can thus be used to replace subgrid scale modelling in LES. The application of CNNs was further extended by An et al. [31] to replace the whole CFD simulation of the hydrogen flame in a cavity. In their work, the CNN was used to predict the flow field and concentration fields at next time step, given the fields at the current time. A series of RANS simulation with different inlet velocities were carried out first to generate the training data. The CNN was tested on several cases and the results for the spatial distributions and temporal dynamics showed good agreement with the CFD simulations.

The present work proposes an ANN chemistry tabulation methodology that can be used in the context of methods for turbulent combustion modelling where the reaction source term is computed in real time, using all of the species in a detailed mechanism. The methodology is applied here to LES-PDF simulations, but it is equally applicable to other methods in this category including DNS, unsteady flamelet, CMC, MMC, LEM, Thickened Flame Model, the Partially Stirred Reactor (PaSR) method employed in OpenFOAM and the computation of laminar flames. The concept of a canonical problem for the generation of the training set is employed but, departing from our previous work [22,23] and following the approach of Readshaw et al. [32], no partitioning of the composition space is employed, thus aiming at generalisation rather than specialisation. The methodology comprises two main elements: a Hybrid Flamelet/Random Data (HFRD) method for generating the training dataset and a Multiple Multilayer Perceptrons (MMP) method for ANN chemistry tabulation. The aim of HFRD is to improve the generalisation ability of ANNs, while the MMP uses multiple ANNs to improve accuracy. The mechanism tabulated is the complete GRI 1.2 mechanism [33], as opposed to the

reduced mechanisms used in our previous work [22,23]. The generalisation ability and accuracy of the HFRD-MMP methodology is demonstrated by application to a number of problems with different composition states: test data from a previous turbulent flame simulation, one-dimensional flamelets with varying strain rates, one-dimensional premixed flames with differential diffusion, and finally the LES-PDF-ANN simulations of the Sandia flame series (D, E, and F).

The paper is structured as follows: first, the main ideas of chemistry tabulation via ANNs and the HFRD-MMP method are described in detail. Subsequently, the method is applied to one-dimensional flamelets and premixed flames. The method is then applied to the LES-PDF simulations of the Sandia flame series, before presenting the final conclusions.

2. Machine learning tabulation methodology

2.1. Basic concepts of chemistry tabulation via ANNs

Given a mechanism, the time evolution of the chemical species can be expressed by the following system of ODEs:

$$\frac{dy_i}{dt} = f(y_i, h, p) \quad (1)$$

where y_i is the concentration of each species, h is the specific enthalpy (including the enthalpies of formation) of the system and p is the pressure. The function f is determined by the chemical kinetics. If the pressure is fixed, then the change of each species concentration after a time step δt is:

$$\delta y_i = F(y_i, h) \quad (2)$$

Therefore, for a given chemical mechanism and a fixed pressure, the concentrations after a reaction step are functions of the initial concentrations and specific enthalpy (note that the latter does not change due to chemical reaction).

ANNs are a class of machine learning models that can be used for nonlinear function fitting. Therefore, ANNs can be used to fit the function $F(y_i, h)$ and replace the integration of the chemical kinetics. The ANN methodology for thermochemistry tabulation contains the following key steps:

- **Training data generation.** The determination of the ANN parameters, which is called training, is carried out using representative input-output pairs, and the ANNs can be expected to reliably represent the chemical kinetics within the composition space covered by the training data. Therefore, a method for generating the training dataset should be devised so as to cover the composition space that will be encountered in the combustion problems, while avoiding the use of data derived from the problems that will be simulated, as that would not guarantee generalisation to different problems. A major aspect of this work is the development of a method for generating representative training data, which will be presented in Sections 2.2.1 and 2.2.2.
- **ANN training.** In the training process, the parameters of the ANNs are adjusted in order to best fit the training data. Training is a nonlinear optimisation process, where training data is presented and the network parameters are adjusted in order to minimise a given loss function. The ANN training methods employed in this work are discussed in Section 2.3.
- **ANN application.** Once the training is complete, the ANNs can be applied to combustion problems. In this work, the ANNs are tested on a wide range of problems including laminar flamelets, freely-propagating premixed flames and turbulent flames (shown in Sections 3 and 4), in order to assess their capacity for generalisation.

2.2. The hybrid flamelet/random data generation method

The training data should cover the composition space that will be encountered in the combustion problems. The most straightforward way to generate the training dataset is sampling from the same or similar turbulent combustion simulations which the ANNs will be applied to. However, this means that every time the ANNs are applied to a new problem, the same or similar simulation must be performed using DI first. On the other hand, it would be very difficult to generate a dataset that can cover the composition space for all combustion problems, especially since the space has a large number of dimensions in the case of a detailed mechanism. In our previous work [22,23], we employed an ensemble of flamelets as a canonical problem for generating data that is representative of a family of combustion problems, namely non-premixed laminar and turbulent flames. In this work, we propose a hybrid flamelet/random data generation method. The random element endows the method with considerably greater generality, which allows for the simulation of diverse problems such as laminar premixed flames with differential diffusion and non-premixed turbulent flames using ANNs derived from the same generic dataset.

2.2.1. Flamelet data generation

The flamelet data generation method developed in our previous work [22,23] is employed as a basis for generating the dataset in the proposed HFRD approach. Laminar non-premixed flamelets can be described by the following equation in mixture fracture space [1]:

$$\rho \frac{\partial y_i}{\partial t} = \rho \frac{\chi(z)}{2} \frac{\partial^2 y_i}{\partial z^2} + \dot{\omega}_i \quad (3)$$

where z is the mixture fraction, y_i is the species concentration (kmol/kg), $\dot{\omega}_i$ is the reaction source term of the i th species, and χ is the scalar dissipation rate given by the following equation:

$$\chi(z) = \frac{S}{\pi} \exp[-2(\text{erfc}^{-1}(2z))^2] \quad (4)$$

where S is the strain rate. By varying the strain rate and the mixture fraction, a wide range of composition states can be collected, ranging from equilibrium to extinction. It must be emphasised that the ANNs trained in this way are not subject to the flamelet model assumptions because the data are shuffled prior to training and the flamelet structure is thus lost.

The sampling method used in the present work is very similar to that in Ref. [23], so only the differences will be mentioned here. The flamelet simulations in Ref. [23] used an RCCE-reduced mechanism, which only involves 16 species, while here the mechanism used is the detailed GRI 1.2 mechanism involving 30 reactive species and 175 reaction steps. In Ref. [23], all of the flamelets were initialised by setting a small part of the mixture fraction space to equilibrium in order to collect data representing ignition. In the present work, only half of the flamelet simulations are carried out in this way. The other half are initialised with the entire mixture fraction space set to equilibrium, which helps to collect more data close to the equilibrium state. In addition, the initial temperature of the fuel and air streams are randomly selected between 100 K and 600 K so that the collected data covers a wider range of specific enthalpy. Data are only collected within the approximate flammability limits, corresponding to a mixture fraction range of (0.02, 0.12). A total of 100 flamelet simulations were performed in order to generate the dataset in the present work, and data for about 960,000 states were collected.

2.2.2. Random data generation

Although the training dataset does not retain the flamelet structure as a result of the shuffling process, it is still limited to composition states that can be generated during a flamelet simulation

and thus biased towards these states. Furthermore, the flamelet is a model for non-premixed flames and the composition states collected thus are unlikely to be adequate for the prediction of premixed flames. In order to overcome these problems, it is proposed that the flamelet dataset should be used to generate a new random dataset and then discarded. In this way, the composition states sampled from the flamelet serve as a guidance for generating physically realistic states via the random data generation algorithm.

For each state sampled from the flamelet dataset, a new random state can be generated using the following procedure. The specific enthalpy of the new state is:

$$h' = h + \frac{c}{a} (h_{\max} - h_{\min}) \quad (5)$$

where h_{\min} and h_{\max} are the minimum and maximum specific enthalpies of the flamelet dataset respectively, h is the specific enthalpy of the sampled flamelet state, h' is the randomly generated specific enthalpy value, c is a uniform random number in the (-1,1) range, and a is a coefficient whose value is prescribed so as to satisfy certain constraints that will be explained below and has a value of 8 in the present work.

Over the entire flamelet dataset, the concentrations of the species vary by orders of magnitude. For example, the maximum value for CH_4 is roughly 10^{-2} kmol/kg while the minimum is less than 10^{-20} kmol/kg. If the concentrations are also randomly generated using the same method as for specific enthalpy, then most of the randomly generated values will be of the order of magnitude of their original maximum value. For example, if a given CH_4 concentration is $y = 10^{-5}$ kmol/kg, then the randomly generated value is $y + (c/a)(y_{\max} - y_{\min})$. This value is of the order of magnitude of y_{\max} , because both y and y_{\min} are negligible compared with y_{\max} . This will cause the random data to deviate greatly from the original flamelet data and lead to inadequate composition space coverage. In order to alleviate this issue, the random concentrations are generated as follows:

$$y'_j = y_j^{(1+\frac{c}{a})} \quad (6)$$

$$y''_j = \frac{y'_j}{\sum y'} \quad (7)$$

where y_j is the concentration of the j th species of the flamelet data, y'_j is the randomly generated species concentration, c is a random number between -1 to 1, while a is, like a in Eq. (5), a coefficient that ensures that the value satisfies the constraints that will be mentioned below and has a value of 5 in the present work. This approach ensures that y'_j is between $y_j^{0.8}$ and $y_j^{1.2}$, and thus of the order of magnitude as y_j . Finally, Eq. (7) is used to normalise the species concentrations so that the species mass fractions sum to unity.

The random data generated by the above method must satisfy several constraints. First of all, the species mass fractions must sum to unity, which is ensured by Eq. (7). Furthermore, there is no need to generate any data outside of the flammability limits. In this work, temperature and mixture fraction are used to determine the flammability range. The temperature limit is set to 500 K and the mixture fraction range is the same as the original flamelet data, $0.02 \leq z \leq 0.12$. Any random data generated with a temperature lower than the limit or a mixture fraction outside of the acceptable range is discarded. The ranges used for random data generation are wider than the ranges encountered in the turbulent flame simulations, in order to make sure that the training data can fully cover the composition space encountered.

Apart from the basic constraints mentioned above, the random data must have appropriate molar element ratios. In the combustion of a CH_4/air mixture, the molar element ratios of H/C and O/N

Table 1
Constraints for random data generation.

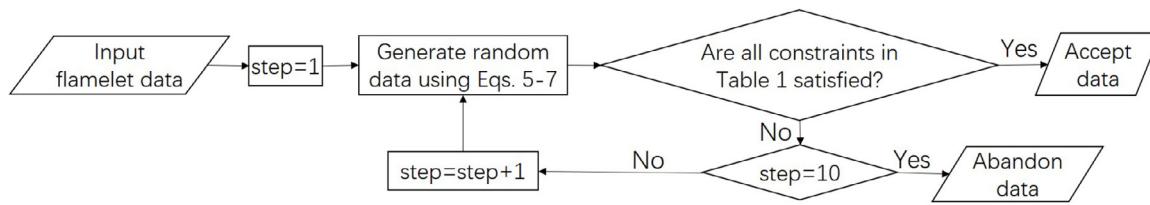
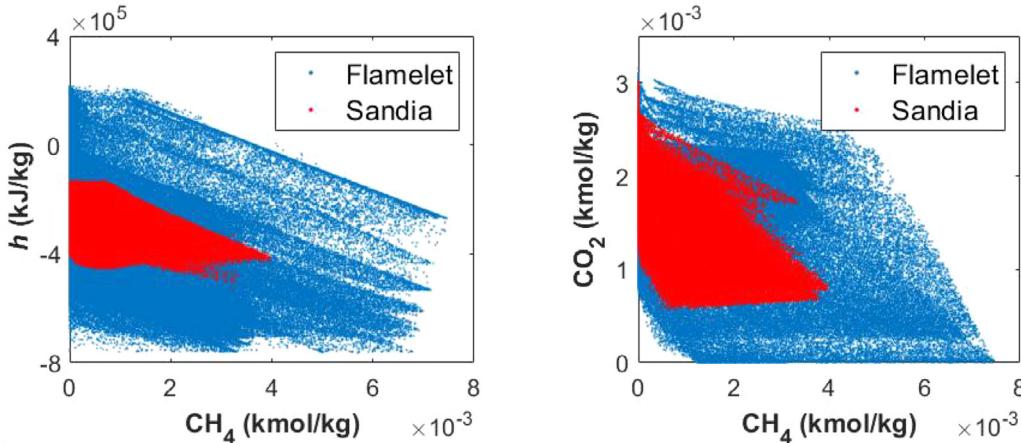
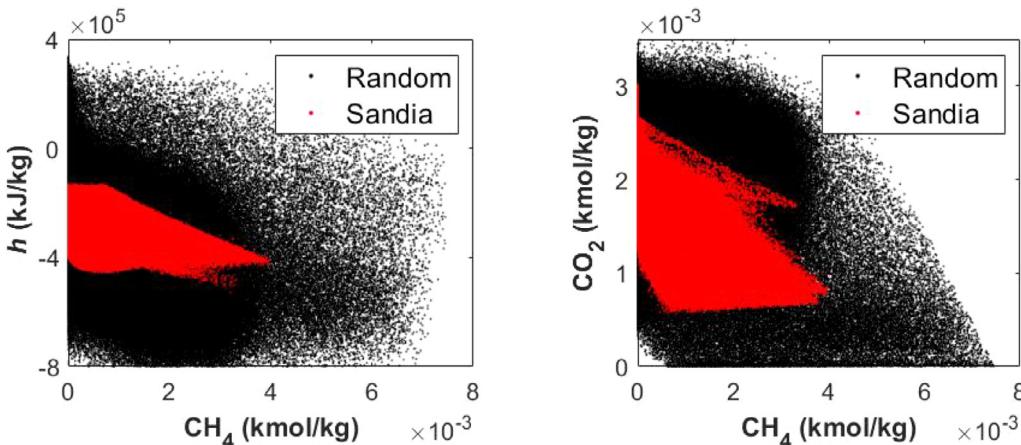
	H/C ratio	O/N ratio	Mixture fracture	Temperature (K)
Minimum	3.8	0.255	0.02	500
Maximum	4.2	0.275	0.12	-

are fixed if the equal Schmidt number assumption is applied. The H/C element ratio is exactly four because the fuel is CH_4 , while the O/N element ratio is the same as that of air. When using DI for reaction source term computations, the element ratios before and after reaction are the same, apart from numerical round-off errors. When applying ANNs, however, the conservation of elements cannot be guaranteed because of ANN prediction errors. If the ANN training dataset has fixed element ratios, the ANN predictions may have slightly different element ratios, and thus correspond to composition states not covered by the training dataset. If the ANNs are further applied on this data, the prediction error can accumulate over time and lead to unacceptable results. In order to overcome this problem, the element ratios in the training data should be set within a range rather than being fixed. In addition, if laminar or turbulent simulations are to be undertaken using differential diffusion, the element ratio at each grid point will also change with time. Therefore, ANNs trained using data with different element ratios are more general and can be applied to a wider range of combustion simulations. In the present work, the H/C ratio range is set to be (3.8,4.2) and the O/N ratio range is set to be (0.255,0.275). These element ratio constraints, as well as the aforementioned limits for temperature and mixture fraction, are shown in Table 1.

In order to generate random data satisfying the constraints in Table 1, the following approach is adopted. The random state is first generated using Eqs. (5) to (7), and subsequently the corresponding element ratios, mixture fraction and temperature are calculated. If all constraints are satisfied, then the new state is accepted; otherwise it is rejected and the process is repeated until the data is accepted or the number of rejections reaches 10. The coefficients a and b in Eqs. (5) and (6) respectively were determined in order to satisfy these constraints, as large coefficients make the random data unrealistic and increase the difficulty in satisfying all of the constraints while small coefficients make the random data very similar to the flamelet data. The flowchart of the complete random data generation method is shown in Fig. 1.

Once the random dataset has been obtained, all data goes through a single reaction step using DI, in order to calculate the target outputs for ANN training. In our work, about 900,000 random data are generated. The direct integration was performed with the VODE solver [34] employing a backward differentiation formula (BDF); the same solver was also employed for the flamelet and combustion simulations.

The composition space spanned by the flamelet and random data is shown in Figs. 2 and 3. Additionally, data collected from LES-PDF simulations of the Sandia flames, described in Section 4, are shown. Two representative scatter plots, specific enthalpy vs CH_4 concentration and CO_2 concentration vs CH_4 concentration are shown here. In Fig. 2, the flamelet data are compared with the Sandia flame data. Although the flamelet data covers the composition space encountered in the Sandia flames, the flamelet data show clear patterns due to the sampling procedure. In Fig. 3, where the random data are compared with the Sandia flame data, no patterns can be discerned in the random data. By eliminating the patterns in the flamelet data, the random data generation method has endowed the ANNs with greater capacity for generalisation, and this will be evidenced by the applications shown in Sections 3 and 4. It can also be observed that the composition space covered by the random data is only slightly larger than the one covered by the

**Fig. 1.** Random data generation process flowsheet.**Fig. 2.** Scatter plots comparing the composition space coverage of the flamelet data and the data collected from the LES-PDF simulations of the Sandia flames.**Fig. 3.** Scatter plots comparing the composition space coverage of the random data and the data collected from LES-PDF simulations of the Sandia flames.

flamelet data, which means that most of the random data are realistic and the values used for the coefficients in Eqs. (5) and (6) are appropriate.

2.3. Multiple MLPs

2.3.1. The MLP regression model

The multilayer perceptron (MLP) applied here is a widely used ANN model for function approximation. An MLP consists of multiple layers, with each layer containing several neurons. Figure 4 shows the structure of an MLP such as the one used in our work. The rightmost layer is called the “output layer” and yields the value that we wish to approximate. The intermediate layers are called “hidden layers”. Each layer contains several neurons, which are connected to all neurons in adjacent layers. The inputs to the neurons of the leftmost layer are the inputs values. For the neurons in the other layers, the inputs are the outputs of the neurons in the previous layer. The output of each neuron is calculated using

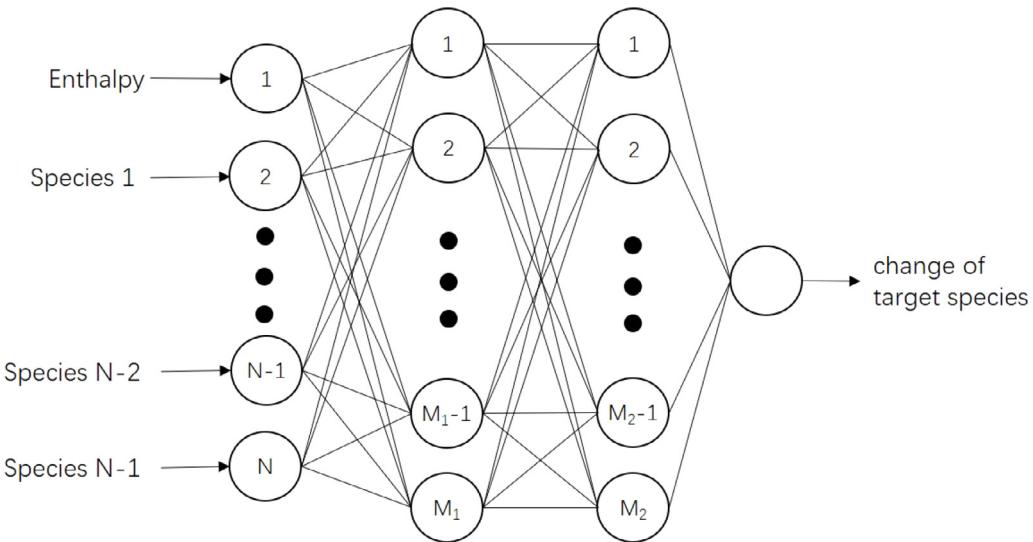
the following equation:

$$a = f(b + \sum p_i w_i) \quad (8)$$

where p_i are the inputs, w_i are weights applied to the inputs, b is the bias and f is the activation function. A neuron can apply a nonlinear mapping from a set of inputs to an output if the function f is a nonlinear function. Since the MLP is made up of several layers of neurons, it is a combination of many nonlinear functions. This nonlinear structure endows the MLP with the flexibility needed to approximate the highly nonlinear functions implicitly determined by the chemical kinetics.

The outputs of each layer for the MLP shown in Fig. 4 can be calculated using the following equations:

$$a_j^1 = f^1(b_j^1 + \sum_{i=1}^N p_i w_{i,j}^1) \quad (9)$$

**Fig. 4.** MLP structure.

$$a_j^2 = f^2(b_j^2 + \sum_{i=1}^{M_1} a_i^1 w_{i,j}^2) \quad (10)$$

$$T = f^3(b_1^3 + \sum_{i=1}^{M_2} a_i^2 w_{i,1}^3) \quad (11)$$

The superscript indicates the layer and the subscript indicates the neuron that we refer to within that layer. The weight $w_{i,j}$ is associated with the connection pointing from the i th neuron of the previous layer to the j th neuron of the present layer. The number of inputs to the MLP is denoted by N and the number of neurons in the i th hidden layer is denoted by M_i . When using the MLP, the inputs will go forward through each layer using the above equations until the final output can be calculated. In this work, the activation function employed for hidden layers is the hyperbolic tangent and the activation function for the output layer is a linear function:

$$f^1(x) = f^2(x) = \tanh(x) \quad (12)$$

$$f^3(x) = x \quad (13)$$

The inputs to the MLP are the species concentrations and the specific enthalpy. The output of the MLP is the concentration change of a single species. In our previous work [22,23], we employed MLPs with multiple outputs where all of the variables were predicted at once, but the composition space was partitioned, with different MLPs operating in different subdomains. In a recent work [32], we have shown that separate MLPs for each species operating within the whole composition space afford better generalisation. The GRI 1.2 mechanism used in the present work involves 31 species including N_2 . Since the N_2 concentration is unchanged by reaction for this mechanism, it is not considered in our ANN tabulation, so the network has a total of 31 inputs (30 species + specific enthalpy). The network used here has 2 hidden layers, each containing 40 neurons. Since the input and output values may vary greatly, it is a standard practice to normalise the inputs and outputs before using them to train ANNs. In this work, the values are normalised to the (-1,1) range.

Typically, the training process aims to minimise the sum of all squared errors of the ANN predictions, as shown below:

$$E_D = \sum (t_i - a_i)^2 \quad (14)$$

where t_i is the correct output while a_i is the output predicted by the ANN. This optimisation process is achieved by adjusting the network parameters according to gradient descent-based algorithms, which use the backpropagation method to calculate the partial derivatives of the loss function E_D with respect to the network weights and biases. Some widely used gradient descent algorithms include steepest descent, conjugate gradient and Levenberg-Marquardt. More details on these algorithms can be found in Ref. [35]. The training algorithm employed here is Bayesian Regularisation, while the Levenberg-Marquardt backpropagation method is employed to minimise the loss function. In contrast to Eq. (14), the loss function used for Bayesian Regularisation is augmented with a new term constraining the magnitude of network weights and biases:

$$F = \beta E_D + \alpha E_w \quad (15)$$

where α and β are parameters that are adjusted during the training process [35,36] and E_w is the sum of squares of all the weights and biases:

$$E_w = \sum w^2 + \sum b^2 \quad (16)$$

Large magnitudes of network parameters will make the network more likely to overfit to the training data. By introducing this penalty term for network complexity, the magnitude of the network parameters is constrained, so that overfitting is avoided and the networks are able to generalise well. This method constraining the size of the network parameters is referred to as regularisation. Bayesian regularisation is a mathematical process that converts a nonlinear regression into a Bayesian statistical problem. Compared with networks trained using other standard backpropagation algorithms, Bayesian regularised networks are more robust and can avoid overfitting, so there is no need for any cross-validation in the training process. The Bayesian regularisation method aims to maximise the posterior density $P(x_i|d_j)$, where x_i is the vector containing all of the weights and biases in the network and d_j is the training data set. If a Gaussian prior distribution is assumed for the network parameters, maximising $P(x_i|d_j)$ is equivalent to minimising the regularised loss function (Eq. (16)). More details on Bayesian regularisation can be found in Refs. [35,36].

In this work, all ANN training was conducted on the Imperial College Research Computing Service using the MATLAB neural network toolbox. Each ANN is trained for 1500 epochs using 200,000 data. The total training time is about 45 h when training all ANNs

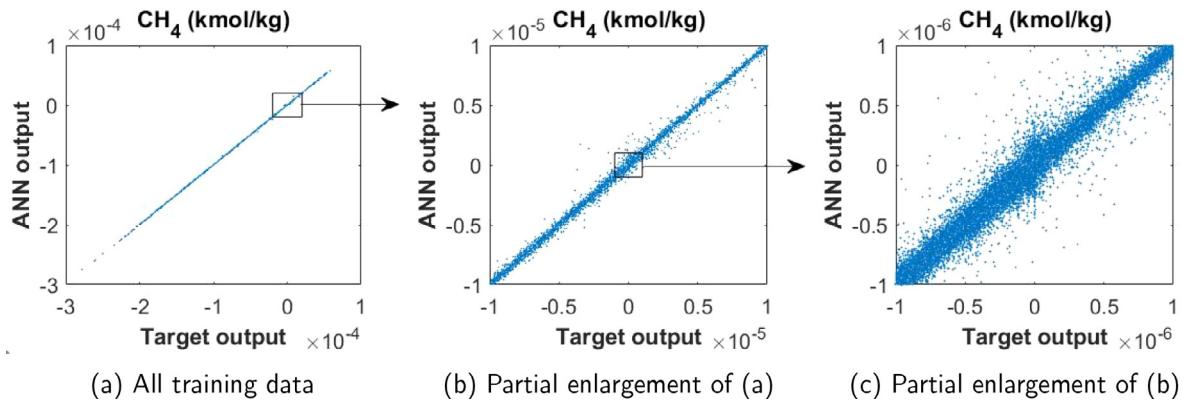


Fig. 5. MLP predictions for CH₄ for target outputs with different orders of magnitude.

in parallel (this time is reported on Xeon 6132 2.6 GHz cores). After training, the ANN weights and biases are used as inputs to our in-house ANN simulation program, which performs the prediction based on Eqs. (9)–(13) and is embedded within our CFD code.

2.3.2. MLP error analysis

As shown in Eq. (15), the training algorithm aims to minimise the sum of squares error. This means that all of the MLP outputs may have similar absolute errors, regardless of their target output values. For a target output with a large magnitude, the absolute error is small, so the MLP output is accurate. For a target output with a small value, the absolute error may be of the same order of magnitude, which leads to a large relative error. The training results for CH₄ are shown here to demonstrate this problem (Fig. 5a). Overall, the MLP predictions seem to be in good agreement with target get outputs. When zooming into the range of small output values ($-10^{-5}, 10^{-5}$), however, as shown in Fig. 5b, it can be seen that the relative errors can be considerable. Further zooming into the range ($-10^{-6}, 10^{-6}$) in Fig. 5c reveals even larger relative errors. It must be emphasised that the poor results shown in Fig. 5b and c do not mean that the MLP is not properly trained. The training data are chosen from the entire dataset, so the MLP performance should be evaluated based on the range of values of the entire dataset, and Fig. 5a shows the quality of the training. The problem lies in the fact that, for a fixed MLP absolute error, the relative error increases with decreasing output magnitude.

The large relative errors for small target outputs may have a significant influence when applying the MLPs to a real simulation. For example, if a composition state is near equilibrium state, the species concentration changes will be close to zero. If the ANN outputs have very large relative errors, the species concentration may be changed significantly, causing a composition state far away from equilibrium and eventually producing unreasonable simulation results. Therefore, it is necessary to reduce the ANN prediction errors for outputs with small magnitude.

One possible way of reducing the MLP prediction errors for outputs with small magnitudes is to continue training the MLPs using these data only. The drawback of this approach is that, since only data with small output magnitudes are presented to the MLPs in the new training process, the MLPs will gradually favour these data which leads to increased prediction error for data with large output magnitudes. A test of this approach showed that it indeed leads to an increase in the overall error, although the relative error for the data with small magnitudes decreased somewhat.

Another approach to improve MLP accuracy is to use a weighted loss function, given by the following equation:

$$E_D = \sum W_i^e (t_i - a_i)^2 \quad (17)$$

In this approach, each output t_i has its own error weight W_i^e , so that their contribution to the total loss varies with their magnitude. If W_i^e is equal to one for all outputs, then Eq. (17) is identical to Eq. (14). In order to improve the MLP accuracy for outputs with a small magnitude, the weights of the corresponding data can be set to large values. Here, a logarithmic function is used to determine the error weight for each data point:

$$W_i^e = \begin{cases} -\log(|t_i|) + k, & -\log(|t_i|) + k \leq 10 \\ 10, & -\log(|t_i|) + k > 10 \end{cases} \quad (18)$$

where k is a constant such that $\min(W_i^e) = 0$. The maximum weight is set to 10 in order to avoid very large weights being applied to data approaching zero. The training results for CH₄ using the weighted loss function are shown in Fig. 6. It can be seen that, although the accuracy for large magnitude outputs is acceptable, the prediction errors for outputs with small magnitude are still considerable and similar to those in Fig. 5b.

The distribution of the training data also affects the MLP training results. A histogram of the absolute change in CH₄ concentration after the default linear scaling to (-1,1) is shown in Fig. 7a. It can be seen that the majority of the outputs have very small magnitudes and these data fall within a very narrow range. In order to expand the range in which the data is distributed, a nonlinear transformation can be applied to the outputs. Such a transformation can be performed with the function shown below:

$$y' = \text{sign}(y)|y|^a \quad (19)$$

where y is the original data and y' is the transformed data. The value of a in Eq. (19) should be positive and smaller than 1. The histograms of the target outputs using different values of a are shown in Fig. 7b and c. It can be seen that, as the value of a decreases, the variance of the distribution decreases and the data with small output magnitudes take up a wider range. The transformed data have been used to train the MLPs and the training results for CH₄ are shown in Fig. 8. Compared with the linear output scaling (Fig. 5a), the MLP prediction error is larger, especially for data with small output magnitudes. The transformation introduces more non-linearity to the relation between inputs and outputs, thus hindering rather than facilitating the ANN training process.

The above analysis shows that it is difficult to obtain both low absolute error for data with large output magnitudes and low relative error for data with small output magnitudes, and the problem cannot be alleviated via error weighting or data scaling approaches. The origin of this issue lies in the physics of the combustion problem: a large concentration change indicates a highly reactive state (as would occur e.g. in an ignition zone), while a small concentration change indicates a state near equilibrium or at low

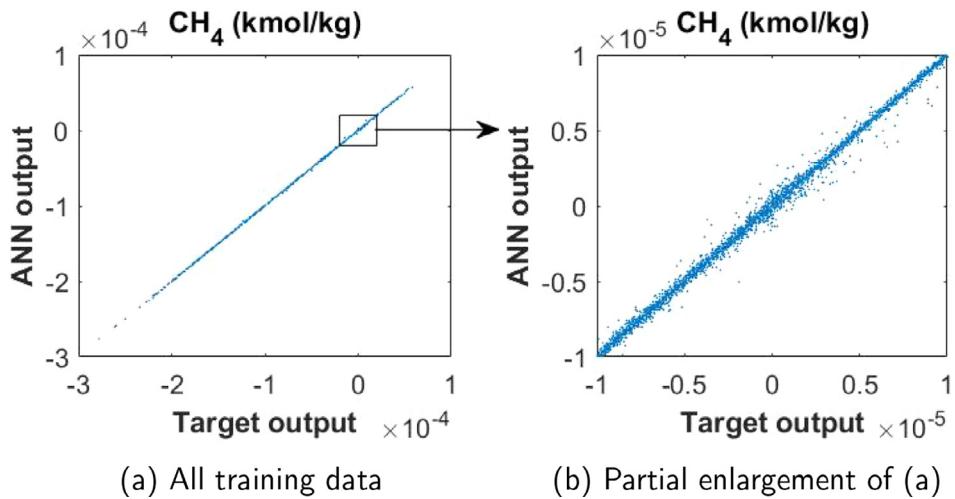


Fig. 6. MLP predictions for CH₄ for target outputs with different orders of magnitude using the weighted loss function.

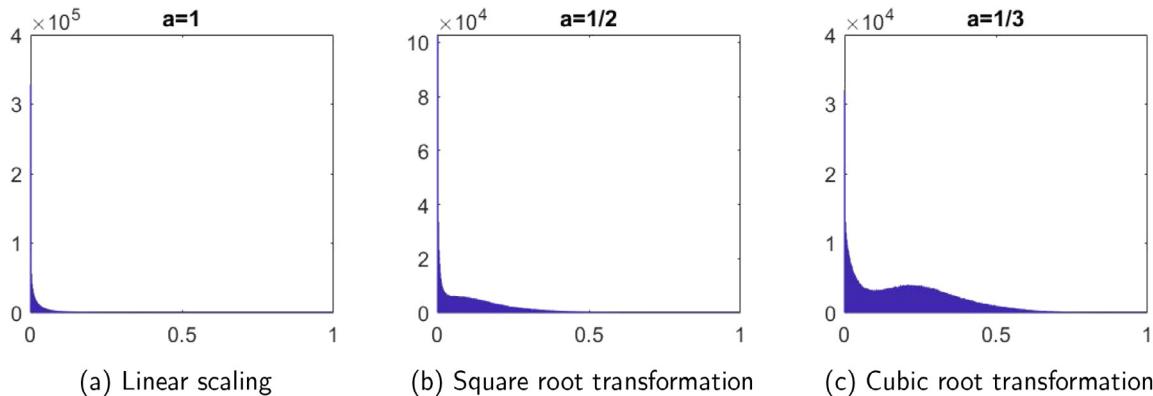


Fig. 7. Histograms of the absolute change of CH₄ concentration using different data transformation methods.

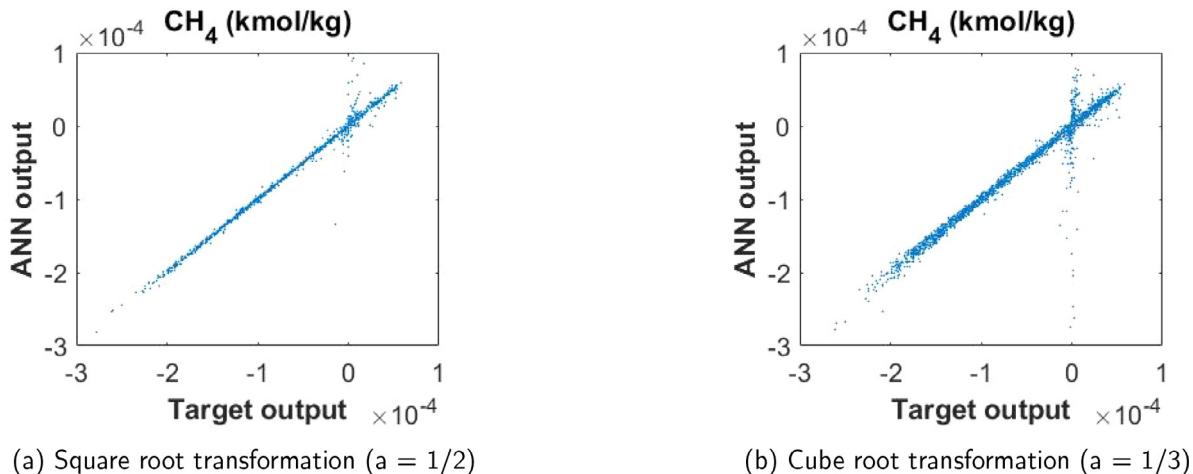


Fig. 8. MLP predictions for CH₄ for target outputs when using the root transformation.

temperature. The dynamical behaviour of the system is different at these states, which means it is challenging to represent them with only a single MLP.

2.3.3. Multiple MLP approach

In this section, we propose an approach that aims to provide good predictions for both large and small concentration changes, using separate MLPs to capture the different dynamics of such cases. When using only a single MLP for the whole dataset, the

errors are of similar magnitude for all data. However, if the data with small output magnitudes are used to train a separate MLP, the output errors will be adjusted to these magnitudes and the prediction accuracy will be improved. A similar concept underlies the approach recently proposed by Wan et al. [25], which employed a single ANN for all the species but trained a separate ANN focussing on reactions with slow burning rate.

The training results for CH_4 are shown here to demonstrate the basic idea of our method. For the data in Fig. 5b, a new

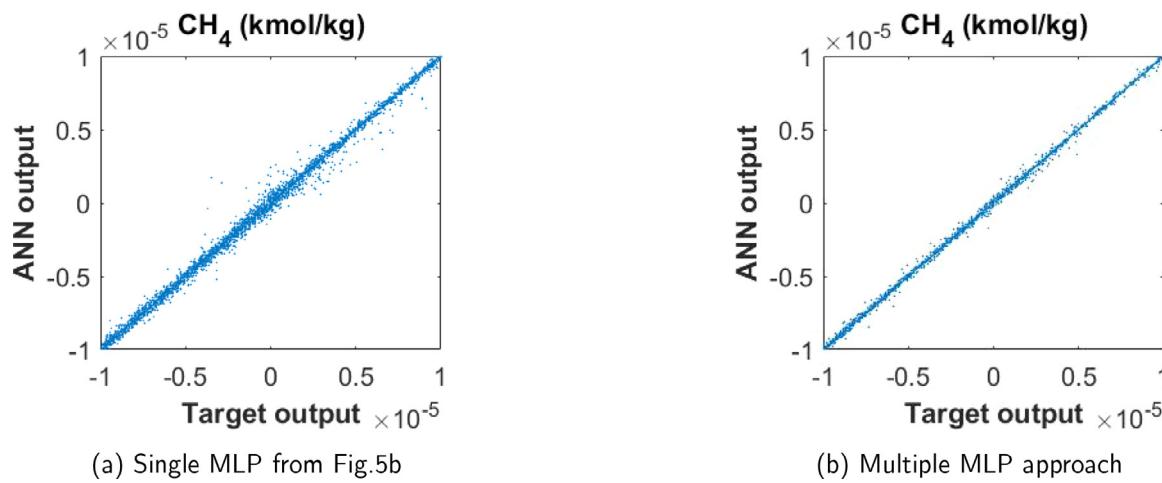


Fig. 9. Comparison between multiple MLPs and single MLP for target outputs within range ($-10^{-5}, 10^{-5}$).

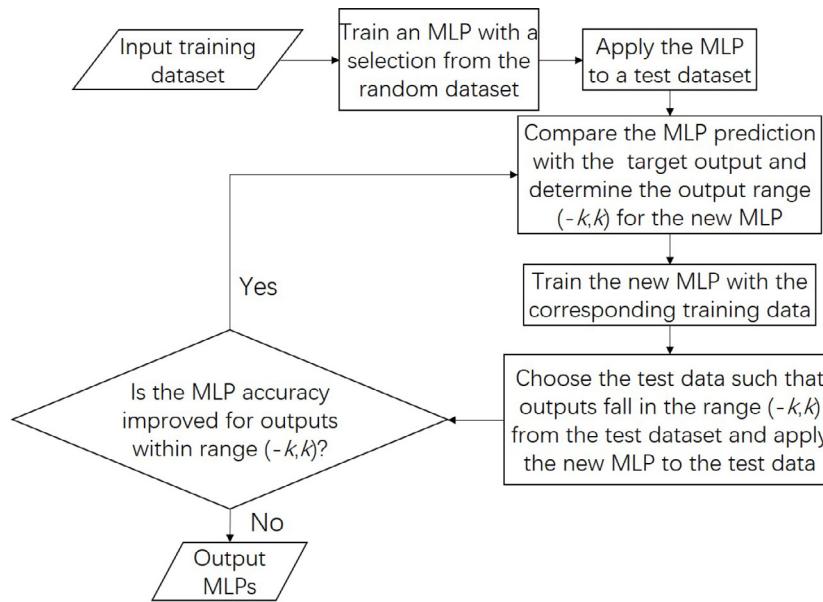


Fig. 10. Multiple MLP training process.

MLP is trained. Figure 9b shows the training results. As shown in Fig. 9, the predictions are substantially improved. A procedure is therefore developed for the training of multiple MLPs, shown in Fig. 10. An MLP is first trained with data from the whole training dataset and then applied to a test set. The MLP predictions are evaluated to determine the appropriate output range for this MLP and the output range with large relative errors, denoted as $(-k, k)$, is determined and used to train the next MLP. The new MLP is applied to the test data and the results of outputs within the range $(-k, k)$ are compared with the previous MLP. If the accuracy shows improvement, then the MLP is accepted. The above process is repeated until a new MLP fails to improve the accuracy any further.

In the present work, the test dataset is sampled from the LES-PDF simulation of Sydney L flame in our previous work [23]. Based on the MLP performance on outputs with low magnitudes, the output range for each new MLP is selected as 1/10, 1/5 or 1/4 of the previous range. For CH_4 , 5 MLPs are used and their RMSEs over the test data are listed in Table 2, with comparison to the RMSEs using only one MLP. The RMSEs of CH_4 , as well as several other species, are also plotted in Fig. 11. It can be seen that, when using only one

MLP, the RMSEs for different ranges of data are very similar, while when using multiple MLPs, the RMSEs decrease as the data range narrows. In the case of CH_4 , the RMSEs of the last 3 ANNs are very similar. This means that further MLPs will not significantly improve the accuracy.

When applying the multiple MLPs to a practical simulation, the target output is unknown, so it cannot be directly determined which MLP should be used. Therefore, all data will start with the first MLP; if the output falls in the range of the next MLP, the input data will go to the next MLP. This process is repeated until the data goes to the final MLP if needed. The MLP implementation process is shown as a flowchart in Fig. 12a. This process can also be realised in reverse, as shown in Fig. 12b. In this work, the reverse process is applied in the turbulent combustion simulations. This is because most outputs in these simulations have low magnitudes, so the reverse process can reduce the number of MLPs the data goes through and thus reduce the simulation time. Thirty species need to be predicted by the MLPs in our work, with each species using 3–5 MLPs. A total of 97 MLPs are trained and then used in the turbulent combustion simulation. The files storing all MLP parameters are about 11 MB.

Table 2
RMSE comparison.

Data range	(-1E-3,1E-3)	(-1E-4,1E-4)	(-1E-5,1E-5)	(-1E-6,1E-6)	(-2E-7,2E-7)
One MLP	1.89E-8	1.81E-8	1.76E-8	1.78E-8	1.76E-8
Multiple MLPs	1.89E-8	1.62E-8	4.93E-9	3.63E-9	2.72E-9

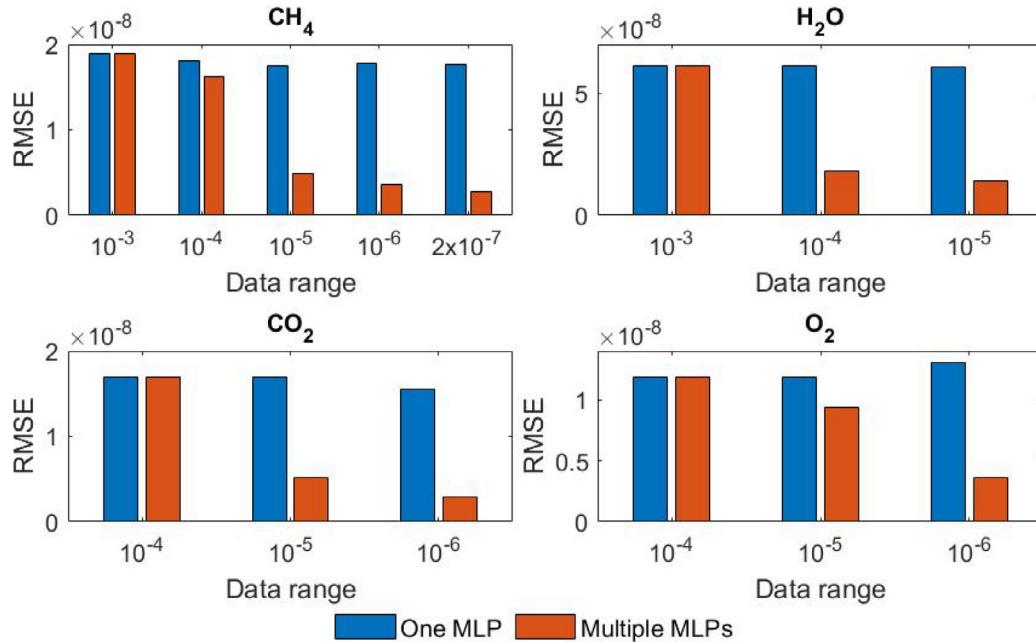


Fig. 11. RMSEs for the test set, when using a single MLP for each species compared to multiple MLPs for each species.

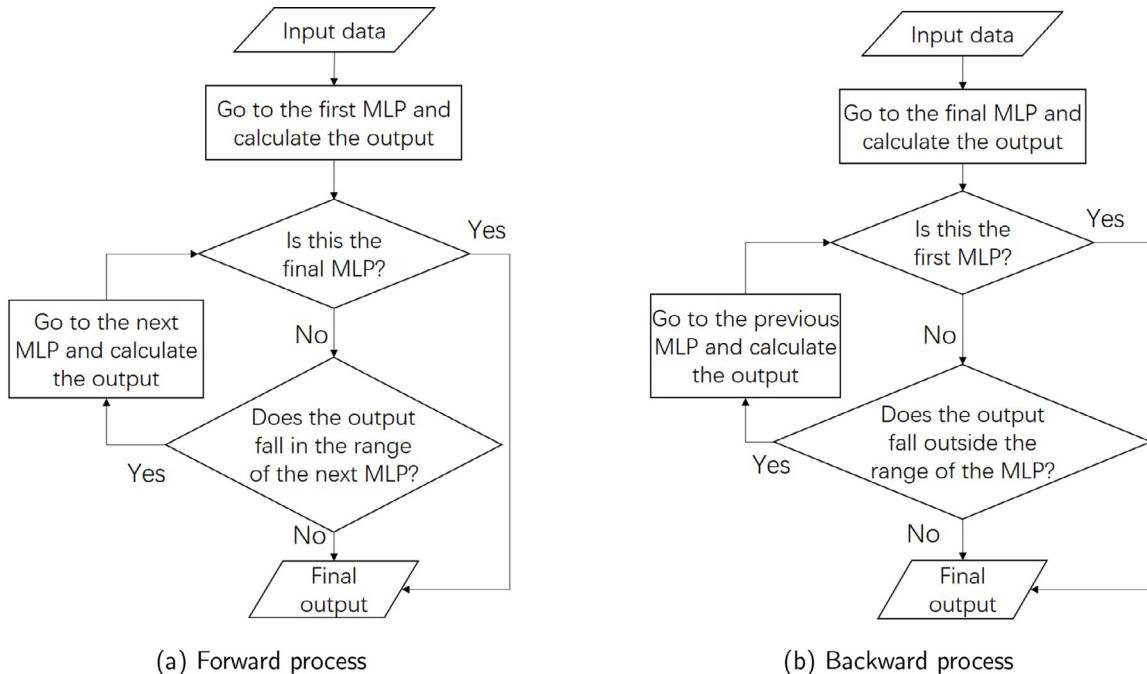


Fig. 12. Multiple MLP implementation process.

3. Application to one-dimensional laminar flamelet and premixed flames

The ANNs are first tested on one-dimensional laminar flamelet simulations, which is the problem used to generate the data from which the random data were derived (but the flamelet data were

not used in the training). The cases demonstrated here use 200 nodes in mixture fraction space, with a finer grid near stoichiometric mixture fraction. Three different strain rates (10 s^{-1} , 100 s^{-1} , and 500 s^{-1}) are chosen to validate the ANNs. The temperature of the inlet streams is set to 300 K. The flamelets are simulated with both direct integration and ANNs, and their results are compared

and shown in Fig. 13. In all three cases, the ANN results are in excellent agreement with the DI results, apart from a small error in CH_2O in the case of strain rate 10 s^{-1} . This strain rate is very low and the composition space in this case is close to equilibrium, hence the concentration changes due to reaction are very small. The application of the multiple MLP method can produce accurate results for such cases and improves on the approach of Readshaw et al. [32].

In order to test the generalisation ability of the ANNs, we now employ them to simulate a problem that is very different from the data generation problem, namely one-dimensional freely-propagating premixed flames. Differential diffusion is also included, which introduces variable element ratios, hence this is a good test of the ability of the HFRD approach. The grid consists of 200 nodes and is refined around the flame front, while the unburnt temperature is set to 300 K. Three cases with different equivalence ratios ($\phi = 0.8, 1.0$ and 1.2) are simulated. The results are shown in Fig. 14. Overall, the ANNs represent the chemical reaction in one-dimensional premixed flames with excellent accuracy, with the ANN and DI profiles virtually coinciding in almost all cases.

The successful application of the HFRD-MMP method to both flamelet and one-dimensional premixed flames, the latter including differential diffusion, even though none of this data was included in the training dataset (the flamelet was used to derive the random data but then discarded), indicates that the resulting ANNs have high generalisation ability. The next test of our methodology constitutes turbulent non-premixed flames, shown in the next section.

4. Application to Sandia flames

4.1. Review of previous work

The turbulent flames chosen as test cases for this work are the well documented Sandia piloted CH_4/air jet flames D, E and F. The experimental studies on these flames were conducted by Schneider et al. [37] and Barlow and Frank [38]. Numerical simulations of these flames have been performed in many works, especially for Flame D. Raman and Pitsch [39] applied the large-eddy simulation/filtered-density function (LES-FDF) method to flames D and E using a 16-species reduced chemistry mechanism. Mustata et al. [40] applied the LES-PDF method combined with stochastic fields to flame D and employed a four-step mechanism for the chemical kinetics. A recent LES study of flame D was performed by Jaravel et al. [41], where a detailed mechanism involving 22 species was used, including NO.

The Sandia flames D-F have an increasing level of local extinction. Flame D has very little local extinction while flame F is close to blow-off, which makes flame F more difficult to simulate than flames D and E. Jones and Prasad [42] performed LES-PDF simulations with stochastic fields to flames D-F, using a 15-step, 19-species mechanism. Although the degree of local extinction of flame F was underpredicted, the simulation still successfully reproduced the local extinction and re-ignition seen experimentally. Garmory et al. [43] applied the LES-CMC method to flames D and F with a 19-species mechanism. The local extinction of flame F was correctly captured at axial location $X/D = 7.5$, but underpredicted at $X/D = 15$. Ge et al. [44] applied the hybrid LES/Sparse-Lagrangian MMC model to flames D-F, using a reduced mechanism including 34 species and 219 reactions. The results for flames D and E were in good agreement with the experimental data, while the results of flame F were only qualitatively correct.

4.2. Numerical setup

The piloted burner of the Sandia flames has a main jet diameter of 7.2 mm and a pilot diameter of 18.2 mm. The jet composition is

Table 3
Flow parameters.

Flame	Re_{jet}	$U_{jet\text{-cold}}$ (m/s)	$U_{pilot\text{-burnt}}$ (m/s)	U_{coflow} (m/s)
D	22,400	49.6	11.4	0.9
E	33,600	74.4	17.1	0.9
F	44,800	99.2	22.8	0.9

25% CH_4 and 75% air by volume, resulting in a stoichiometric mixture fraction based on the jet composition of about 0.35. The annular pilot burnt composition has the same specific enthalpy and equilibrium composition as CH_4/air at $\phi = 0.77$, the mixture fraction of which is about 0.27. The inflow velocities and the corresponding Reynolds numbers of the different flames are shown in Table 3. In our simulation, the temperature of the jet and pilot is set to 298 K and 1880 K for all 3 flames. Radiative heat loss is not considered in our simulations. It must be noted, however, that the random data generation approach does create data with a variation in specific enthalpy, so the method could potentially be employed in non-adiabatic flames; such cases will be tested in future work.

In this work, the Sandia flames are simulated with the LES-PDF method and stochastic fields for numerical solution, and the implementation is identical to that described in the work of Jones et al. [42]. Eight stochastic fields are used, the dynamic Smagorinsky model of Piomelli and Liu [45] is applied, and equal diffusivities are assumed with a Schmidt number of $\sigma_{sgs} = 0.7$. The simulations are performed with our in-house CFD code BOFFIN [46]. The simulation domain has dimensions $36 \times 7.5 \times 7.5 \text{ cm}$, which is 50 jet diameters in the downstream direction and more than 10 jet diameters in the radial direction. A Cartesian finite volume grid of $300 \times 100 \times 100$ cells is employed, evenly distributed in the downstream direction while smoothly expanding in the radial direction. The time step in the simulations is set to be 10^{-6} s , the same as the time step used for the training data generation. In our simulations, a fixed time step is sufficient, but if a variable time step is required, it can be accomplished by consecutively calling the ANNs trained for the minimum time step. Alternatively, ANNs can be trained for larger time steps as well, and these would yield a greater speed benefit, as the CPU time required by ANNs does not depend on the time step. Two simulations were carried out for each flame, one using DI and one using the HFRD-MMP method for the reaction source term computations. For each flame, all statistical quantities were collected for a period of more than 5 flow-through times (based on U_{jet}), and the collection started once the flame had become fully developed.

4.3. Results and discussion

The mean and rms values for several species mass fractions and temperature are collected at 4 axial locations: $X/D = 7.5, 15, 30$ and 45 , where D is the jet diameter. The corresponding radial profiles are plotted for comparison in Figs. 15–20. The radial distance, r , in these figures is scaled by the jet diameter. Figs. 15, 17 and 19 show the results of several major species, as well as temperature (T) for flames D, E, and F respectively. Results for several minor species are also shown in Figs. 16, 18 and 20 for each flame. It can be seen that the ANN results are in excellent agreement with the DI results for most of the major species. At $X/D = 7.5, 15$ and 30 , the results are almost indistinguishable. At $X/D = 45$, some small differences near the centreline can be seen for H_2 , OH , CH_4 and CO , but overall the ANN and DI results are still in very good agreement. The results are very good for most of the minor species as well. At locations $X/D = 7.5, 15$ and 30 , the agreement is excellent with the exception of C_2H_6 and H_2O_2 . The former is slightly underpredicted by the ANNs in flame D, while the latter is underpredicted by the ANNs at $X/D = 5$ in flame F. At $X/D = 45$, some

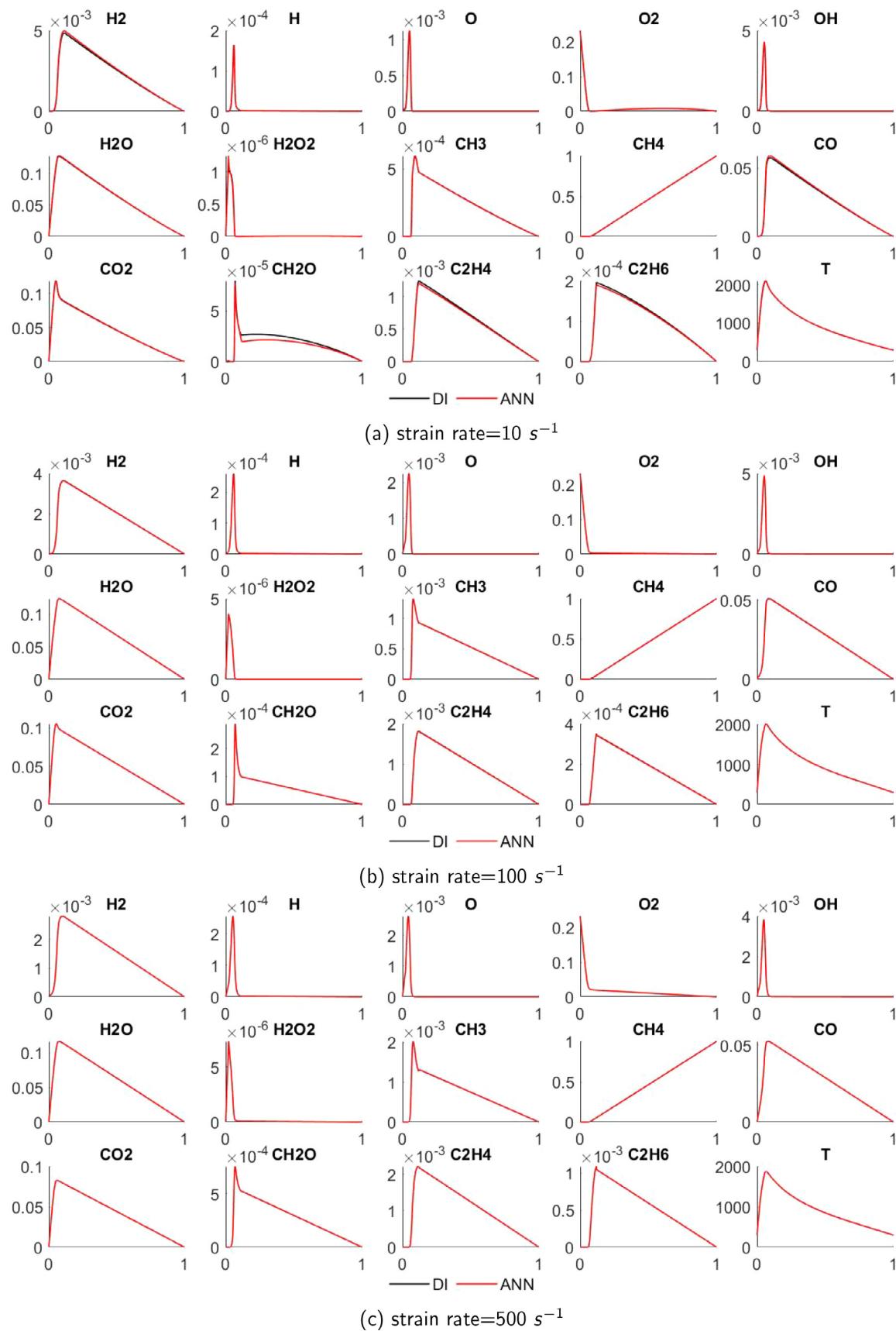


Fig. 13. Application of ANNs to one-dimensional laminar flamelets with different strain rates.

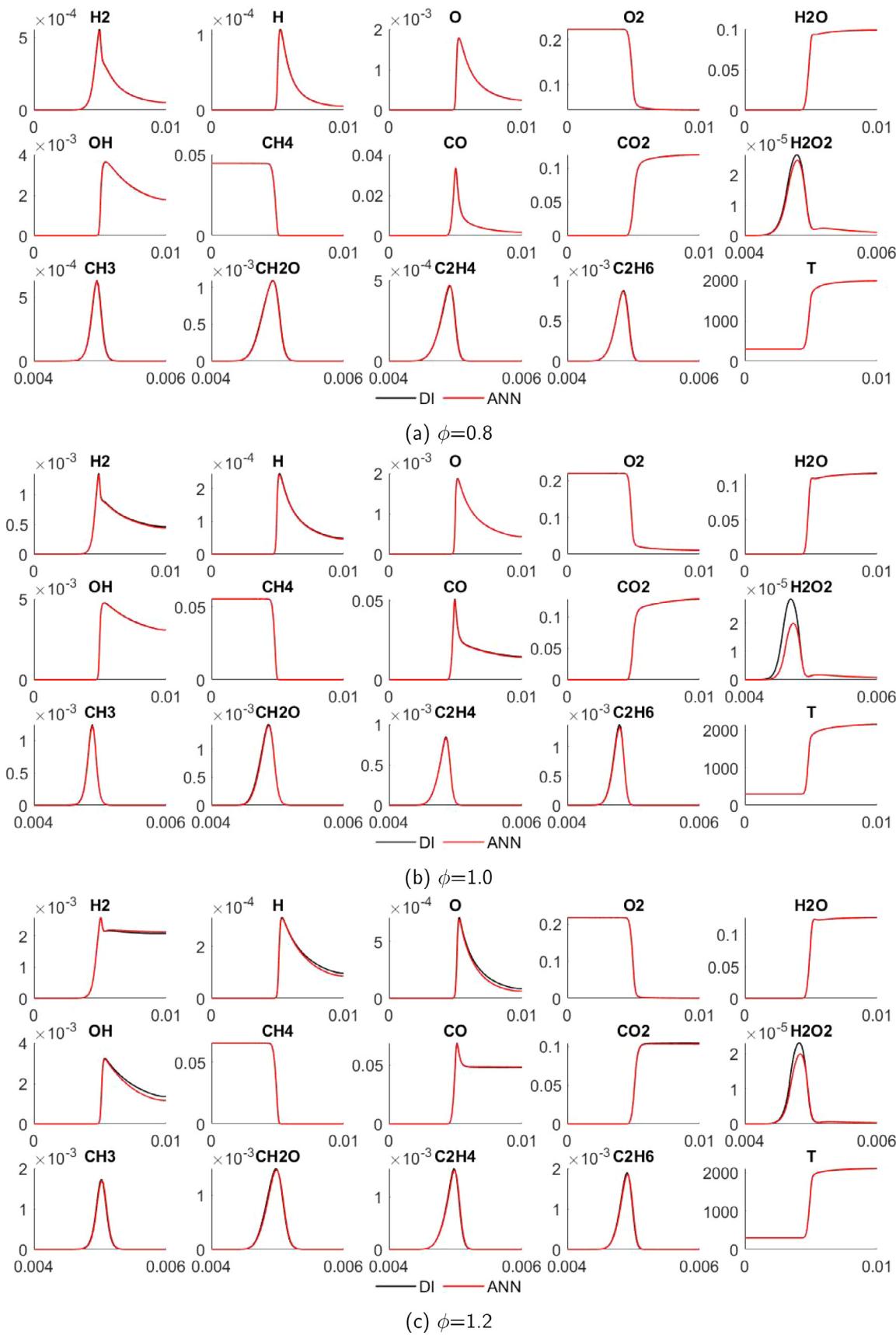


Fig. 14. Application of ANNs to the one-dimensional premixed flames with different equivalence ratios.

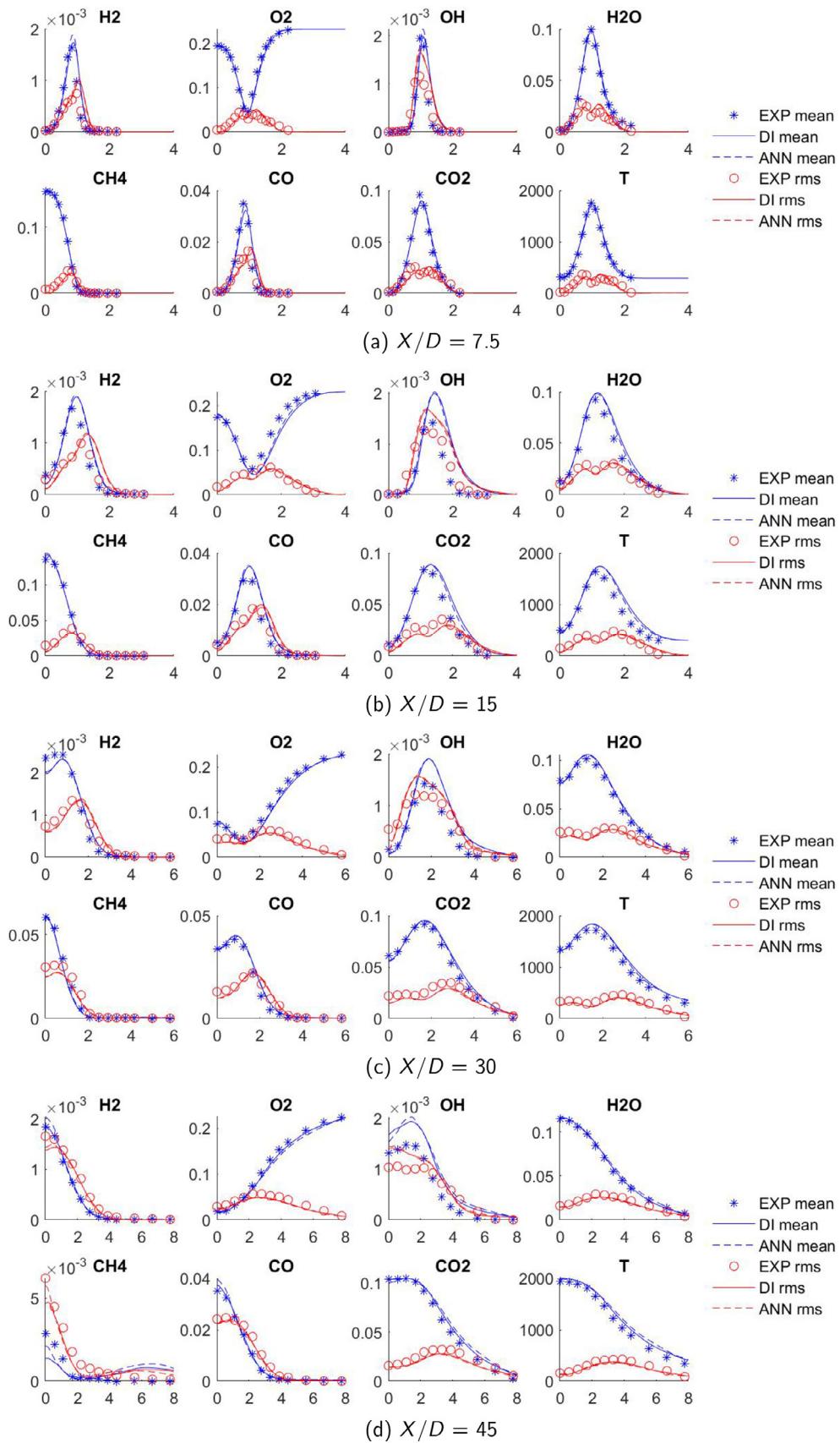


Fig. 15. Temperature and major species of flame D.

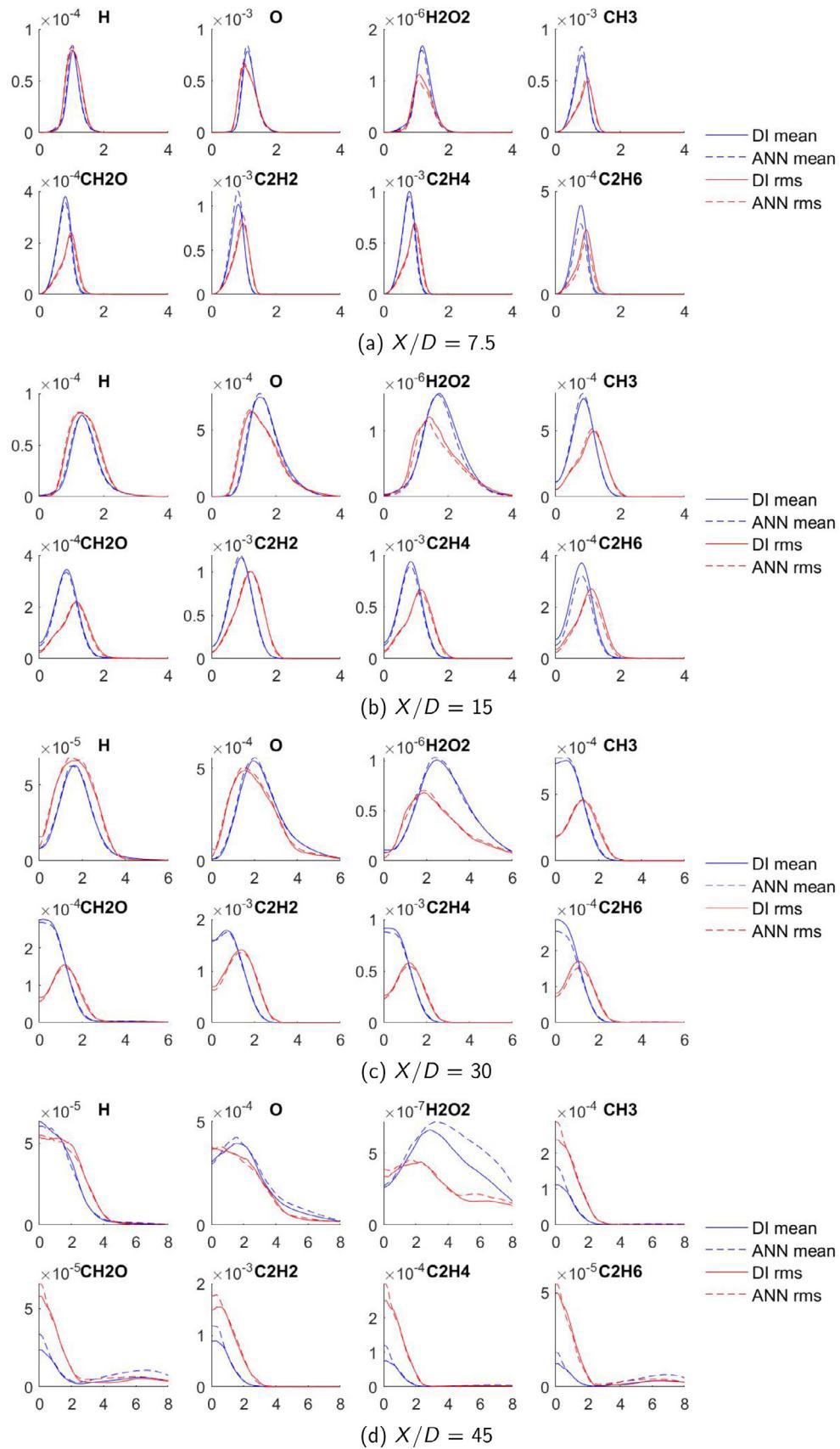


Fig. 16. Minor species of flame D.

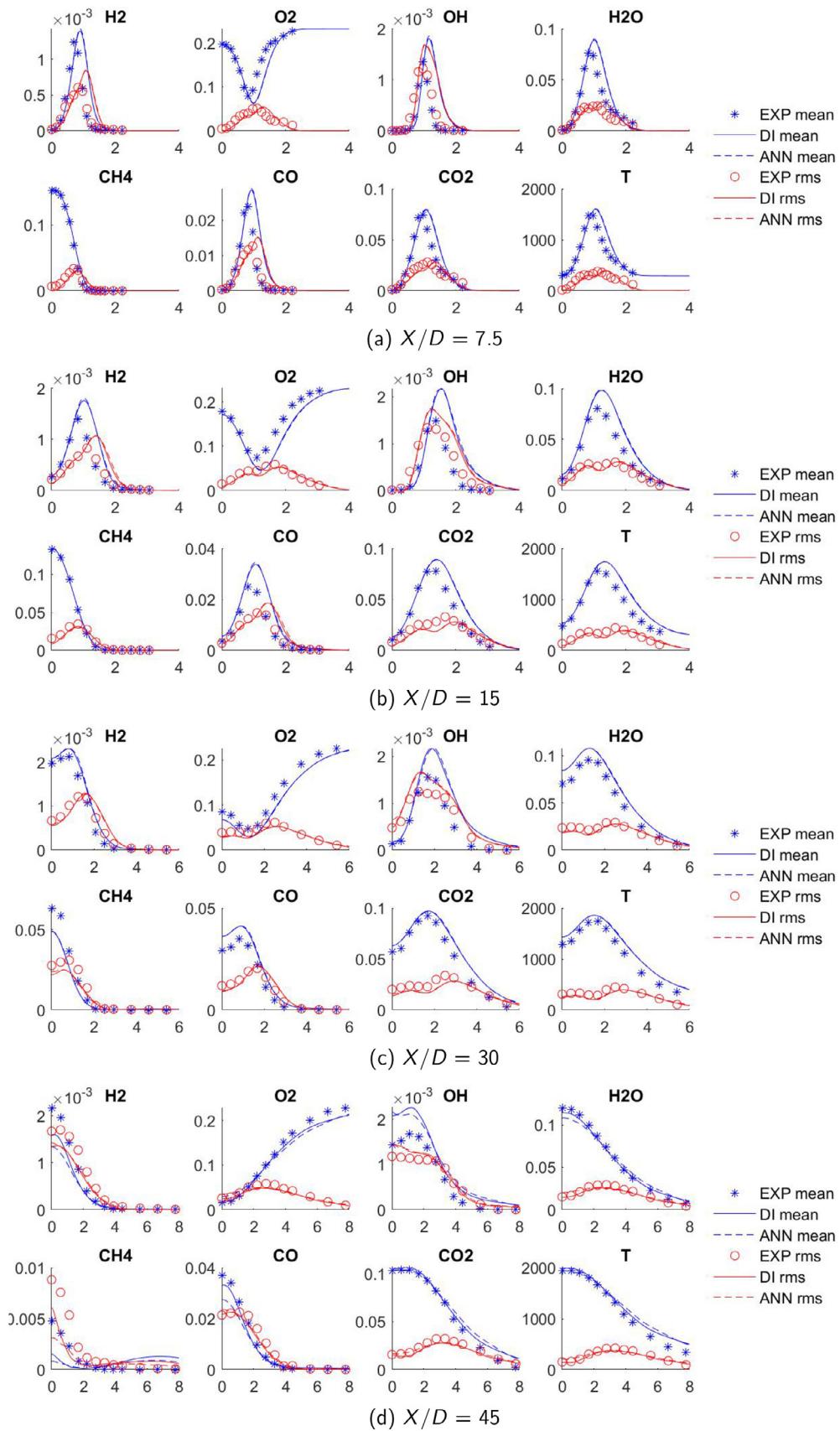
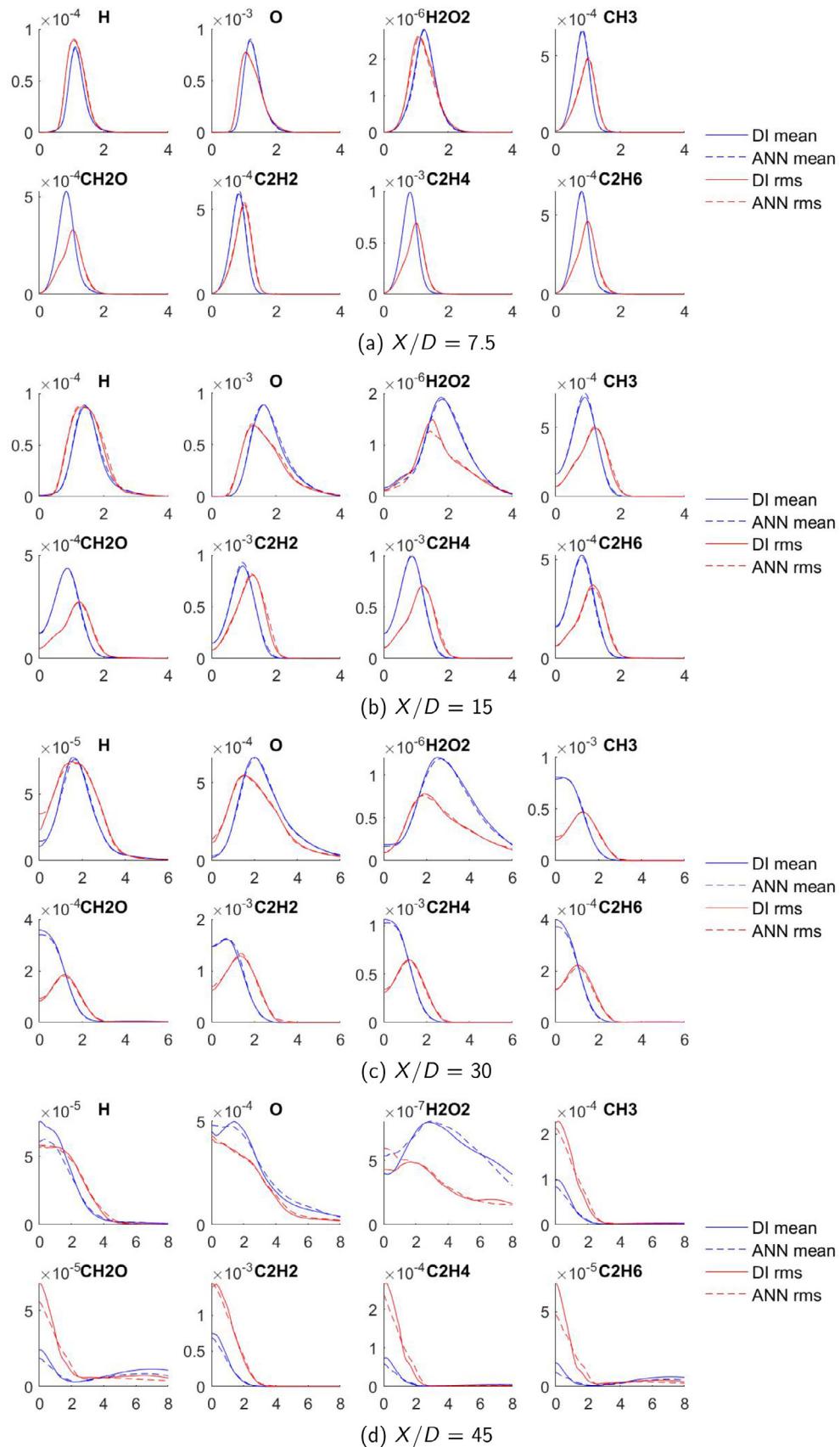


Fig. 17. Temperature and major species of flame E.

**Fig. 18.** Minor species of flame E.

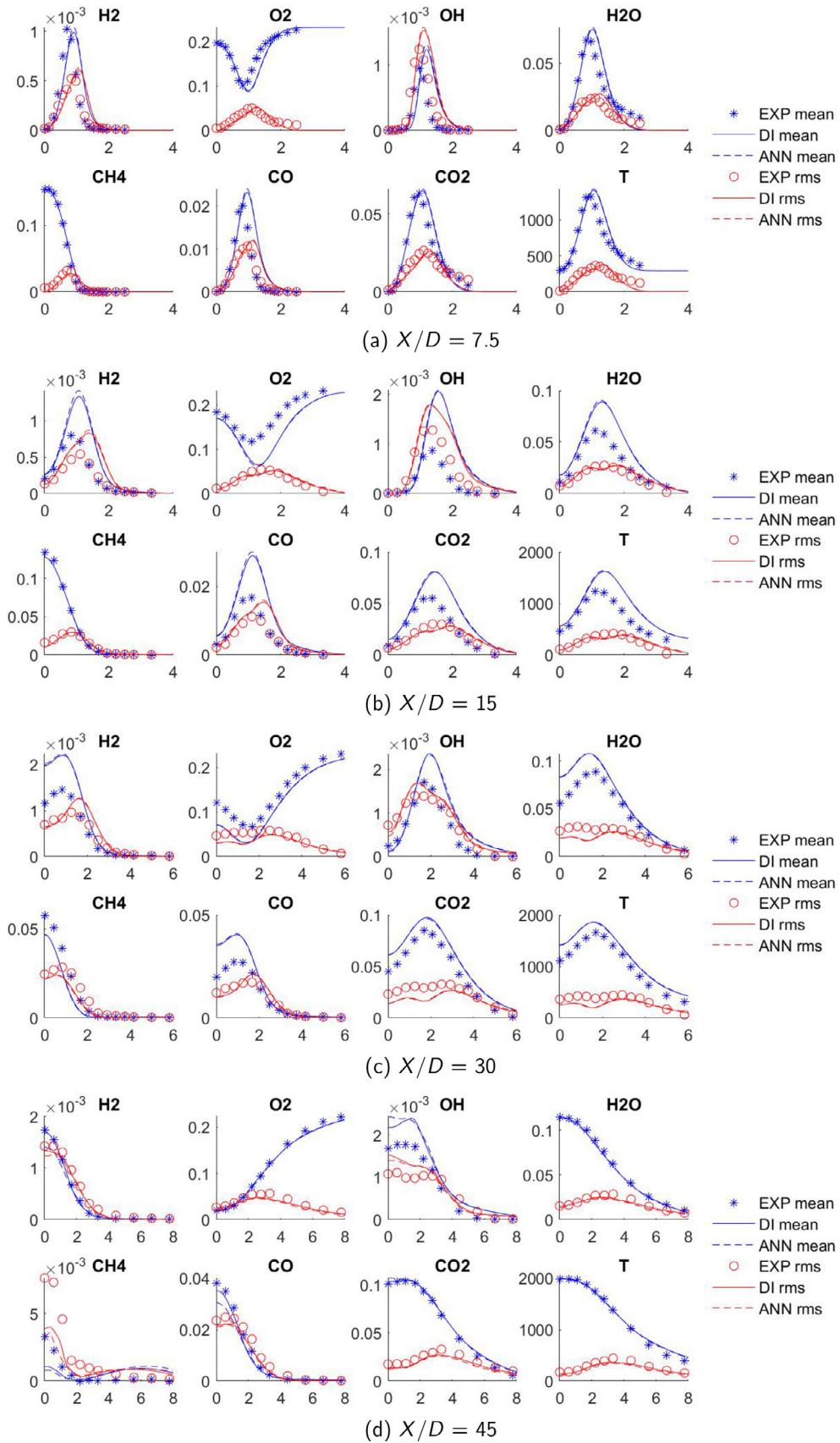


Fig. 19. Temperature and major species of flame F.

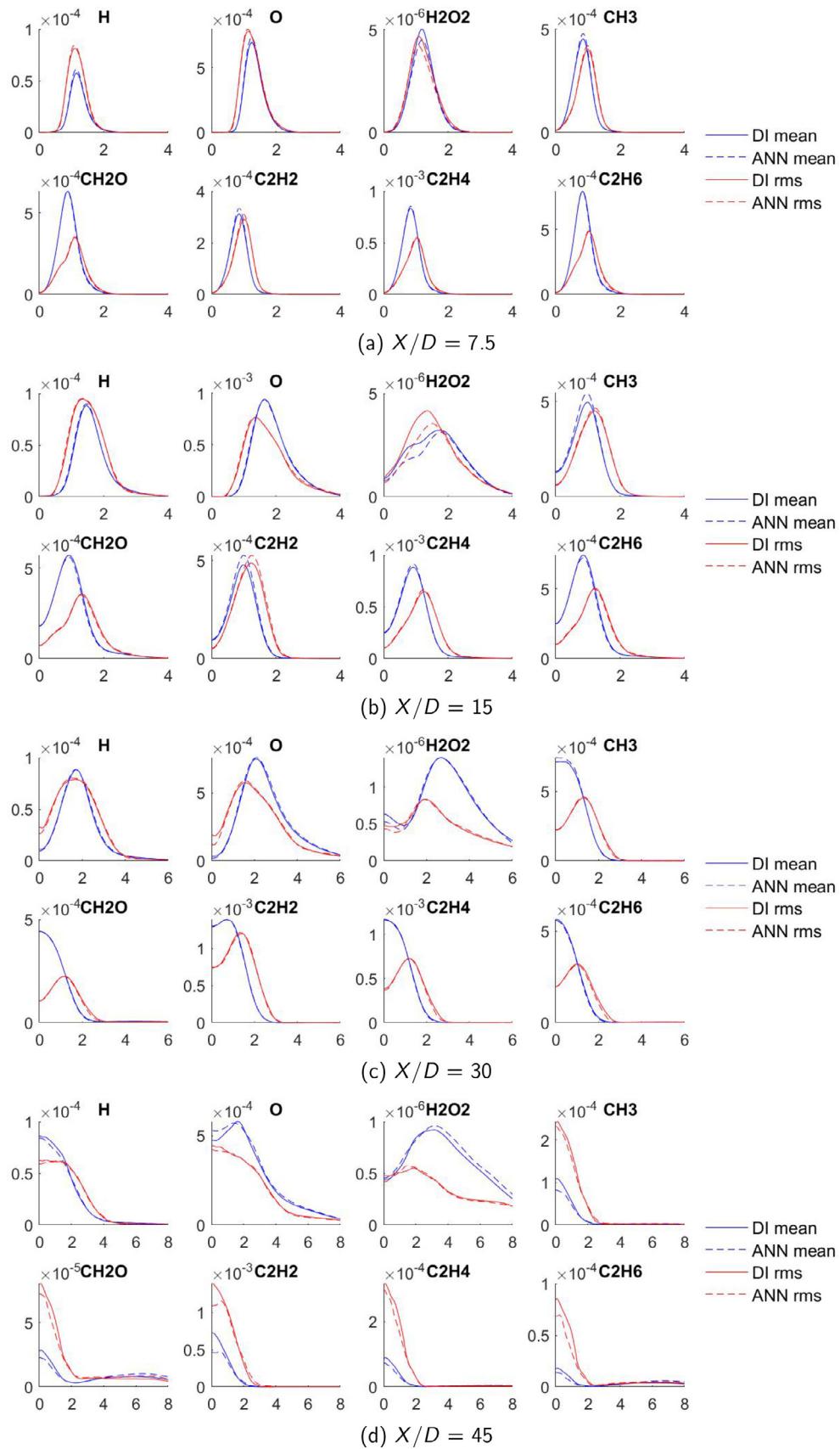


Fig. 20. Minor species of flame F.

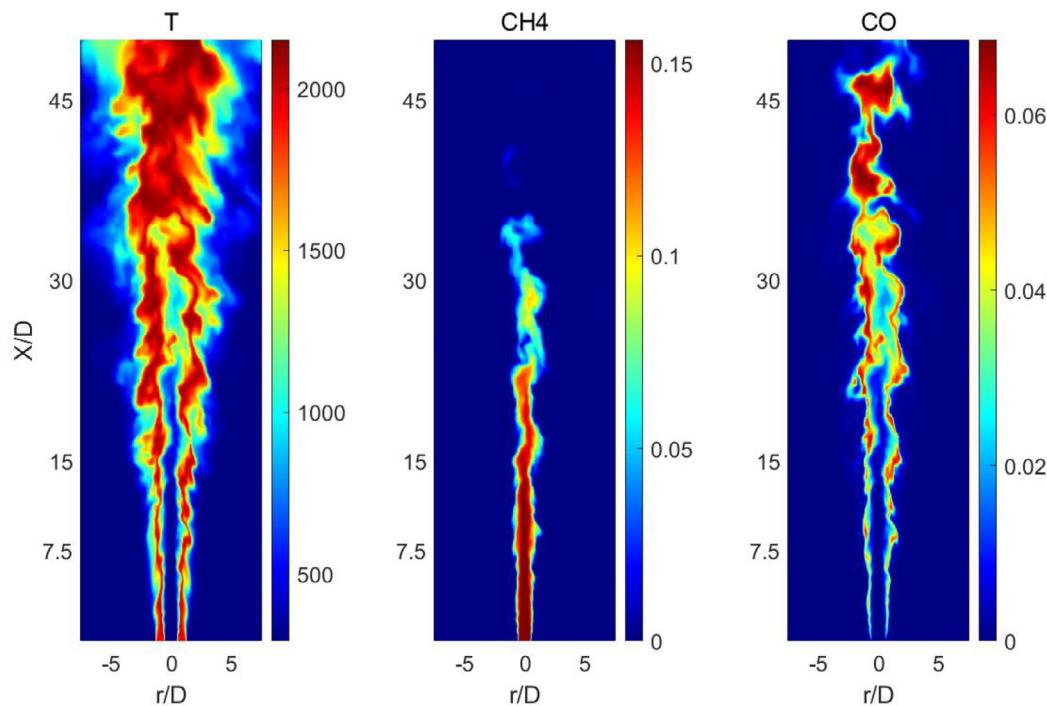


Fig. 21. Snapshots of HFRD-MMP simulation of flame D.

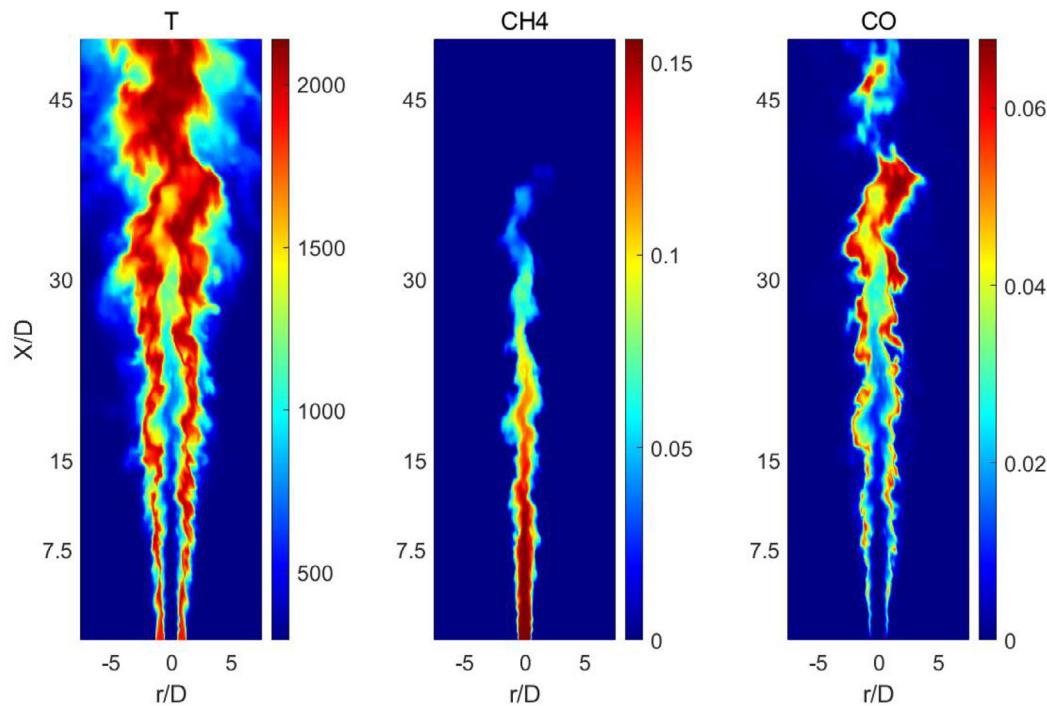


Fig. 22. Snapshots of HFRD-MMP simulation of flame E.

discrepancies can be seen near the peak of most minor species, especially for flame D. Overall, it can be seen that the ANNs produce overall very good results for all flames, apart from some discrepancies at $X/D = 45$, mainly for the minor species.

The instantaneous snapshots of some scalar fields from the HFRD-MMP simulations are shown in Figs. 21–23. It can be seen from the temperature field that flames D and E show almost no local extinction throughout the whole domain, while flame F shows very obvious extinction at low axial locations and almost no extinction at high axial locations. This indicates that the ANNs can

successfully reproduce the local extinction and re-ignition phenomena in flame F, as predicted by the DI.

4.4. Chemistry tabulation speed-up

A comparison of the averaged CPU time taken for the Sandia F simulation using the HFRD-MMP method and DI is shown in Table 4, taken after the flame is fully developed. For convenience, the time spent on the reaction step is normalised with the time required for DI. Compared with DI, the CPU time for the reaction

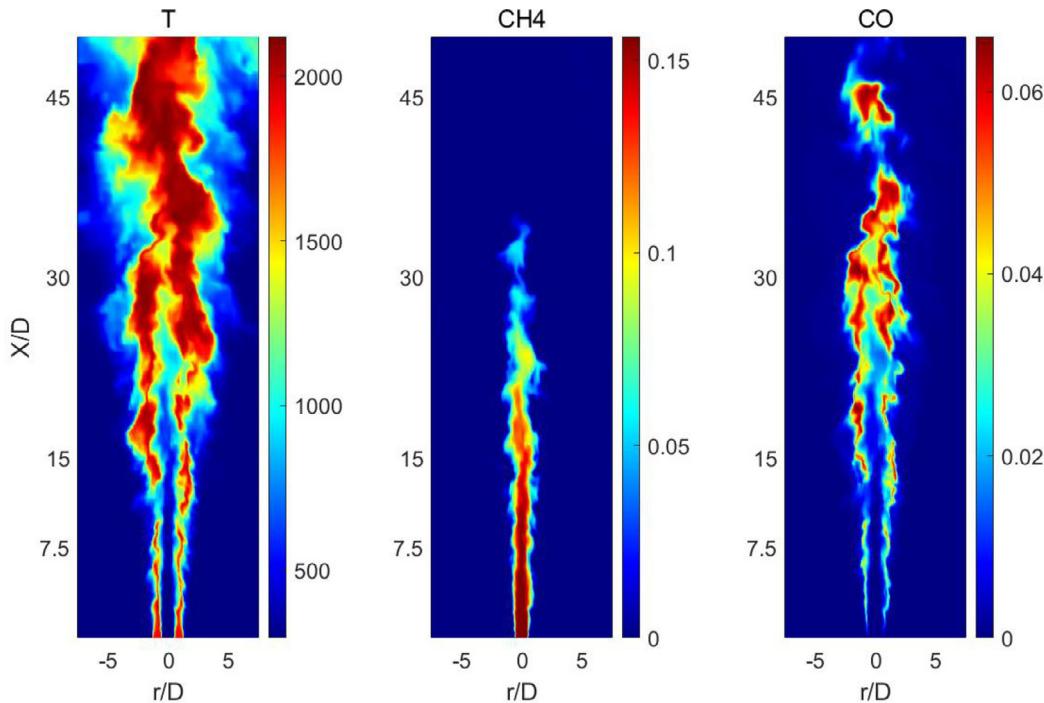


Fig. 23. Snapshots of HFRD-MMP simulation of flame F.

Table 4
Average CPU time for the HFRD-MMP method and DI.

Method	Reaction time(t_R)	Total time (t_T)	t_R/t_T
Direct integration	1	1.35	74.1%
ANN	0.087	0.44	19.9%

step is reduced by more than 90% if the HFRD-MMP method is employed. The percentage of reaction time to total time is reduced from 74.1% to 19.9%, which means that the computation of the reaction step is no longer the bottleneck of the entire simulation.

In our simulations, the ANN speed-up ratio (i.e. the CPU time for reaction required by DI over that required by ANNs) is about 12. The multiple ANNs that form part of our approach provide a further increase in accuracy at the expense of speed. It is possible to further optimise the method for speed, but the reduction in the percentage of total time occupied by the reaction is a more important target. This is because the total speed-up ratio (i.e. total CPU time using DI versus total CPU time using ANNs) is 3.1, while a speed-up ratio of 120 would yield a total speed-up of 3.8, a meagre further benefit that would require a tabulation approach 10 times more efficient.

Greater savings can be anticipated with larger mechanisms that are also often very stiff, in contrast to GRI 1.2 and methane chemistry in general, which is reasonably fast to integrate directly. Stiffness, which has a big effect on the time required for direct integration, does not affect the ANNs at all. With large mechanisms, however, the time spent in scalar transport is also increased due to the large number of scalars, so the percentage of CPU time occupied by chemistry integration is unlikely to get over 80–90%. Even for such cases, therefore, an ANN speed-up ratio in the range of 10–20 would still be sufficient. Only if the percentage of the total CPU time occupied by the chemistry is of the order of 99% would a speed-up ratio of the order of 100 be beneficial. The same situation occurs in the parallelisation of codes: with a code that is 80% parallelisable, the use of 10 cores to attain a speed-up ratio of 10 in the parallelisable part results in a total speed-up ratio of 3.57,

while the use of 100 cores increase the latter only up to 4.8 and would thus be considered wasteful.

5. Conclusions

In the present paper, a method was developed for the tabulation of comprehensive chemical kinetics, based on ANNs. There are two key features in the proposed HFRD-MMP method. The first feature is the training data generation method, where the training data are obtained via a hybrid flamelet/random data generation method. The hybrid flamelet/random data generation method endows the ANNs with increased capacity for generalisation. The second feature is the use of multiple MLPs for different output ranges, which correspond to composition states with different reactivity, in order to improve the accuracy of the prediction.

The generalisation ability of the ANNs was tested first by predicting a set of laminar flamelets with varying strain rate and a set of one-dimensional premixed flames with differential diffusion and varying equivalence ratio. Excellent results were obtained in both cases. It must be noted that flamelet data were used only as a basis for generating the random training dataset and then discarded, hence the success of predicting the flamelets and particularly the premixed flames with differential diffusion is a strong indication of generalisation ability. The Sandia D-F flame series were then chosen to evaluate the capacity of the proposed approach to generalise to the thermochemistry encountered in turbulent combustion simulations. The flames were simulated with the LES-PDF approach and the stochastic fields method for numerical solution. The simulation results were compared with those obtained via direct integration and overall very good agreement for both major species and minor species was achieved, apart from a few discrepancies for some minor species at the most downstream location.

The fundamental idea of the multiple MLP approach has one common element with the SOM-MLP method employed in previous works [18,23,24]: the training of multiple simple networks, each specialising on different subdomains of the composition space. The multiple MLP approach divides the domain based on the magnitude of the concentration change (which is an indica-

tion of reactivity), while the SOM-MLP approach divides the domain via clustering. Combining several simple networks to build a strong predictor is a promising method to overcome this problem, which is the focus of ensemble machine learning [47,48]. Future work could investigate ensemble strategies, such as boosting and mixture of experts. However, the development of the approach proposed here has been driven by the needs of the combustion kinetics tabulation problem, where different subdomains can feature different dynamics (e.g. ignition, extinction etc.), and the method is thus guided by a consideration of the underlying physics.

The HFRD-MMP method yielded a speed-up factor of about 12 for the reaction source term, while the chemistry occupied only 19.9% of the total CPU time in the LES-PDF-ANN simulation. The speed-up, the accuracy attained for both major and minor species and the capacity for generalisation to a wide range of problems (laminar non-premixed and premixed flames and turbulent non-premixed flames) using ANNs trained with a dataset that was derived with random data guided by a canonical problem, all indicate that the method has great potential for the tabulation of larger mechanisms and application to realistic combustion problems. Furthermore, the approach can be employed to speed up the computation of thermochemistry in any turbulent combustion modelling method where real-time calculation of the reaction source term is employed, including DNS, PDF methods, unsteady flamelet, CMC, MMC, LEM, Thickened Flame Model and PaSR (as in OpenFOAM), as well as in the computation of laminar flames.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

Tianjie Ding gratefully acknowledges the financial support provided by the [China Scholarship Council](#) (CSC). Thomas Readshaw gratefully acknowledges the support by the [Engineering and Physical Sciences Research Council](#) (EPSRC) and Rolls-Royce plc in the form of a Doctoral Training Partnership (DTP) and Cooperative Award in Science & Technology (CASE) award. The simulations in this work have been conducted on the Imperial College Research Computing Service (<http://doi.org/10.14469/hpc/2232>).

References

- [1] N. Peters, *Turbulent Combustion*, Cambridge University Press, 2000.
- [2] D. Veynante, L. Vervisch, Turbulent combustion modeling, *Prog. Energy Combust. Sci.* 28 (3) (2002) 193–266.
- [3] T. Poinsot, D. Veynante, *Theoretical and Numerical Combustion*, second ed., Edwards, 2005.
- [4] S.H. Lam, D.A. Goussis, The CSP method for simplifying kinetics, *Int. J. Chem. Kinet.* 26 (4) (1994) 461–486.
- [5] W.P. Jones, S. Rigopoulos, Rate-controlled constrained equilibrium: formulation and application to nonpremixed laminar flames, *Combust. Flame* 142 (3) (2005) 223–234.
- [6] P. Koniavitis, S. Rigopoulos, W.P. Jones, A methodology for derivation of RCCE-reduced mechanisms via CSP, *Combust. Flame* 183 (2017) 126–143.
- [7] P. Koniavitis, S. Rigopoulos, W.P. Jones, Reduction of a detailed chemical mechanism for a kerosene surrogate via RCCE-CSP, *Combust. Flame* 194 (2018) 85–106.
- [8] T. Lu, C.K. Law, A directed relation graph method for mechanism reduction, *Proc. Combust. Inst.* 30 (1) (2005) 1333–1341.
- [9] J.-Y. Chen, W. Kollmann, R.W. Dibble, Pdf modeling of turbulent nonpremixed methane jet flames, *Combust. Sci. Technol.* 64 (4–6) (1989) 315–346.
- [10] U. Maas, S.B. Pope, Implementation of simplified chemical kinetics based on intrinsic low-dimensional manifolds, *Symp. (Int.) Combust.* 24 (1) (1992) 103–112.
- [11] S.B. Pope, Computationally efficient implementation of combustion chemistry using in situ adaptive tabulation, *Combust. Theor. Model.* 1 (1) (1997) 41–63.
- [12] F.C. Christo, A.R. Masri, E.M. Nebot, T. Turanyi, Utilizing artificial neural network and repro-modelling in turbulent combustion, *IEEE International Conference on Neural Networks - Conference Proceedings*, 2 (1995), pp. 911–916.
- [13] F.C. Christo, A.R. Masri, E.M. Nebot, Artificial neural network implementation of chemistry with PDF simulation of H_2/CO_2 flames, *Combust. Flame* 106 (4) (1996) 406–427.
- [14] F.C. Christo, A.R. Masri, E.M. Nebot, S.B. Pope, An integrated PDF/neural network approach for simulating turbulent reacting systems, *Symp. (Int.) Combust.* 26 (1) (1996) 43–48.
- [15] J.A. Blasco, N. Fueyo, C. Dopazo, J. Ballester, Modelling the temporal evolution of a reduced combustion chemical system with an artificial neural network, *Combust. Flame* 113 (1–2) (1998) 38–52.
- [16] J.A. Blasco, N. Fueyo, J.C. Larroya, C. Dopazo, Y.J. Chen, A single-step time-integrator of a methane-air chemical system using artificial neural networks, *Comput. Chem. Eng.* 23 (9) (1999) 1127–1133.
- [17] J.-Y. Chen, J.A. Blasco, N. Fueyo, C. Dopazo, An economical strategy for storage of chemical kinetics: fitting in situ adaptive tabulation with artificial neural networks, *Proc. Combust. Inst.* 28 (1) (2000) 115–121.
- [18] J.A. Blasco, N. Fueyo, C. Dopazo, J.Y. Chen, A self-organizing-map approach to chemistry representation in combustion applications, *Combust. Theor. Model.* 4 (1) (2000) 61–76.
- [19] B.A. Sen, S. Menon, Turbulent premixed flame modeling using artificial neural networks based chemical kinetics, *Proc. Combust. Inst.* 32 I (1) (2009) 1605–1611.
- [20] B.A. Sen, S. Menon, Linear eddy mixing based tabulation and artificial neural networks for large eddy simulations of turbulent flames, *Combust. Flame* 157 (1) (2010) 62–74.
- [21] B.A. Sen, E.R. Hawkes, S. Menon, Large eddy simulation of extinction and reignition with artificial neural networks based chemical kinetics, *Combust. Flame* 157 (3) (2010) 566–578.
- [22] A.K. Chatzopoulos, S. Rigopoulos, A chemistry tabulation approach via rate-controlled constrained equilibrium (RCCE) and artificial neural networks (ANNs), with application to turbulent non-premixed $CH_4 / H_2 / N_2$ flames, *Proc. Combust. Inst.* 34 (1) (2013) 1465–1473.
- [23] L.L.C. Franke, A.K. Chatzopoulos, S. Rigopoulos, Tabulation of combustion chemistry via artificial neural networks (ANNs): methodology and application to LES-PDF simulation of sydney flame I, *Combust. Flame* 185 (2017) 245–260.
- [24] J. An, G. He, K. Luo, F. Qin, B. Liu, Artificial neural network based chemical mechanisms for computationally efficient modeling of hydrogen/carbon monoxide/kerosene combustion, *Int. J. Hydrogen Energy* 45 (53) (2020) 29594–29605.
- [25] K. Wan, C. Barnaud, L. Vervisch, P. Domingo, Chemistry reduction using machine learning trained from non-premixed micro-mixing modeling: application to DNS of a syngas turbulent oxy-flame with side-wall effects, *Combust. Flame* 220 (2020) 119–129.
- [26] A. Kempf, F. Flemming, J. Janicka, Investigation of lengthscales, scalar dissipation, and flame orientation in a piloted diffusion flame by LES, *Proc. Combust. Inst.* 30 (1) (2005) 557–565.
- [27] M. Ihme, C. Schmitt, H. Pitsch, Optimal artificial neural networks and tabulation methods for chemistry representation in LES of a bluff-body swirl-stabilized flame, *Proc. Combust. Inst.* 32 I (1) (2009) 1527–1535.
- [28] M. Hansinger, Y. Ge, M. Pitzner, Deep residual networks for flamelet/progress variable tabulation with application to a piloted flame with inhomogeneous inlet, *Combust. Sci. Technol.* (2020) 1–27.
- [29] Z.M. Nikolaou, C. Chrysostomou, L. Vervisch, S. Cant, Progress variable variance and filtered rate modelling using convolutional neural networks and flamelet methods, *Flow Turbul. Combust.* 103 (2) (2019) 485–501.
- [30] A. Seltz, P. Domingo, L. Vervisch, Z.M. Nikolaou, Direct mapping from LES resolved scales to filtered-flame generated manifolds using convolutional neural networks, *Combust. Flame* 210 (2019) 71–82.
- [31] J. An, H. Wang, B. Liu, K.H. Luo, F. Qin, G.Q. He, A deep learning framework for hydrogen-fueled turbulent combustion simulation, *Int. J. Hydrogen Energy* 45 (2020).
- [32] T. Readshaw, T. Ding, S. Rigopoulos, W.P. Jones, Modeling of turbulent flames with the Large Eddy Simulation-probability density function (LES-PDF) approach, stochastic fields, and artificial neural networks, *Phys. Fluids* 33 (3) (2021) 035154.
- [33] M. Frenklach, H. Wang, M. Goldenberg, G.P. Smith, D.M. Golden, C.T. Bowman, R.K. Hanson, W.C. Gardiner, V. Lissianski, GRI-Mech –an Optimized Detailed Chemical Reaction Mechanism for Methane Combustion, Technical Report, Gas Research Institute Topical Report No. GRI-95/0058, 1995.
- [34] P.B. Brown, G.D. Byrne, A.C. Hindmarsh, VODE: a variable-coefficient ODE solver, *SIAM J. Sci. Stat.Comput.* 10 (5) (1989) 1038–1051.
- [35] M.T. Hagan, H.B. Demuth, M.H. Beale, O. De Jesús, *Neural Network Design*, Ebook, second ed.,
- [36] F. Dan Foresee, M.T. Hagan, Gauss-Newton approximation to Bayesian learning, *IEEE International Conference on Neural Networks - Conference Proceedings*, 3 (1997), pp. 1930–1935.
- [37] C. Schneider, A. Dreizler, J. Janicka, E.P. Hassel, Flow field measurements of stable and locally extinguishing hydrocarbon-fuelled jet flames, *Combust. Flame* 135 (1–2) (2003) 185–190.
- [38] R.S. Barlow, J.H. Frank, Effects of turbulence on species mass fractions in methane/air jet flames, *Symp. (Int.) Combust.* 27 (1) (1998) 1087–1095.
- [39] V. Raman, H. Pitsch, A consistent LES/filtered-density function formulation for the simulation of turbulent flames with detailed chemistry, *Proc. Combust. Inst.* 31 II (2007) 1711–1719.
- [40] R. Mustata, L. Valiño, C. Jiménez, W.P. Jones, S. Bondi, A probability density function Eulerian Monte Carlo field method for large eddy simulations: application

- cation to a turbulent piloted methane/air diffusion flame (sandia d), *Combust. Flame* 145 (1–2) (2006) 88–104.
- [41] T. Jaravel, E. Riber, B. Cuenot, P. Pepiot, Prediction of flame structure and pollutant formation of Sandia flame D using Large Eddy Simulation with direct integration of chemical kinetics, *Combust. Flame* 188 (2018) 180–198.
- [42] W.P. Jones, V.N. Prasad, Large Eddy Simulation of the Sandia Flame Series (D-F) using the Eulerian stochastic field method, *Combust. Flame* 157 (9) (2010) 1621–1636.
- [43] A. Garmory, E. Mastorakos, Capturing localised extinction in Sandia Flame F with LES-CMC, *Proc. Combust. Inst.* 33 (1) (2011) 1673–1680.
- [44] Y. Ge, M.J. Cleary, A.Y. Klimenko, A comparative study of Sandia flame series (D-F) using sparse-Lagrangian MMC modelling, *Proc. Combust. Inst.* 34 (1) (2013) 1325–1332.
- [45] U. Piomelli, J. Liu, Large-eddy simulation of rotating channel flows using a localized dynamic model, *Phys. Fluids* 7 (4) (1995) 839–848.
- [46] W.P. Jones, F. di Mare, A.J. Marquis, LES-BOFFIN: user's guide, 2002.
- [47] C. Zhang, Y. Ma, *Ensemble Machine Learning: Methods and Applications*, Springer, 2012.
- [48] T.G. Dietterich, Ensemble methods in machine learning, International workshop on multiple classifier systems, Springer (2000), pp. 1–15.