

AN INTEGRATED PDF/NEURAL NETWORK APPROACH FOR SIMULATING TURBULENT REACTING SYSTEMS

F. C. CHRISTO, A. R. MASRI AND E. M. NEBOT

*Department of Mechanical and Mechatronic Engineering
The University of Sydney
NSW 2006, Australia*

S. B. POPE

*Sibley School of Mechanical and Aerospace Engineering
Cornell University, Ithaca, NY 14853 USA*

In this paper, we introduce an integrated PDF/neural network approach for the simulation of turbulent flames. The use of artificial neural network (ANN) to represent chemical reaction in turbulent flames offers a significant reduction in the computer memory and run time demands over the classical methods, namely, look-up tables and direct integration. The adequacy of the neural network model strongly depends on the selection of the training set, which should be representative of the most accessible composition space. This is essential for the network to give an accurate model of the chemistry.

A novel method known as *statistical mapping* is used to generate a training set for the neural network. This is a small-scale presimulation of the flame to produce a set of samples that are representative of the most accessible compositions during actual flame computations. In case of difficulties in achieving convergence of the network, "histogram redistribution" technique is used to smooth the PDF of the input samples. This technique is found to improve the convergence of network; however, its generality is yet to be determined. The integrated PDF/ANN approach is demonstrated here for a piloted flame with simple chemistry.

Introduction

The PDF/Monte Carlo approach for simulating turbulent reacting systems offers the advantage of representing the chemical source term in a closed form [1,2]. However, the numerical implementation of the chemistry using standard methods, namely, look-up tables or direct integration (DI), imposes severe limitations on the practical use of these approaches. These restrictions are dictated by the immense memory allocation required by the look-up tables approach and the prohibitive computation time that is required to perform direct integration. In both methods, the computational demands become unrealistically large when more than, say, four reactive species are necessary for representing the chemical kinetics. In recent years, a number of promising techniques have been developed to overcome these limitations. These are the intrinsic low dimensional manifolds (ILD) [3,4,5], repro-modeling [6,7], and artificial neural network (ANN) [8,9], which is used in this study.

The use of artificial neural networks is emerging as a promising, cost-effective alternative for representing chemical reactions in turbulent combustion simulations [8]. It aims at producing a simplified ver-

sion of the real system while retaining the same general behavior. The main advantages of the neural network approach are its capability to represent the chemistry accurately and the huge saving in memory storage and computational time, in comparison with the look-up table and direct integration approaches.

One of the basic and important factors involved in developing a neural network model is in generating an appropriate training set that adequately represents the chemistry over the entire accessible domain of compositions and that can be trained without major difficulties. This is often difficult to obtain. The problem emerges because of the lack of a priori knowledge about the most accessed domain of composition during the simulations. Current practice for generating the input samples of the training set is to determine the boundaries of the allowed composition space and to select samples evenly distributed within this domain. This procedure is described in detail elsewhere [10]. The changes in composition caused by chemical reaction over a certain time increment are computed and form the output samples of the training set. The difficulty with this approach is that the allowed space is often larger than the space that is frequently accessed during the simulation and the training set is not focused over this

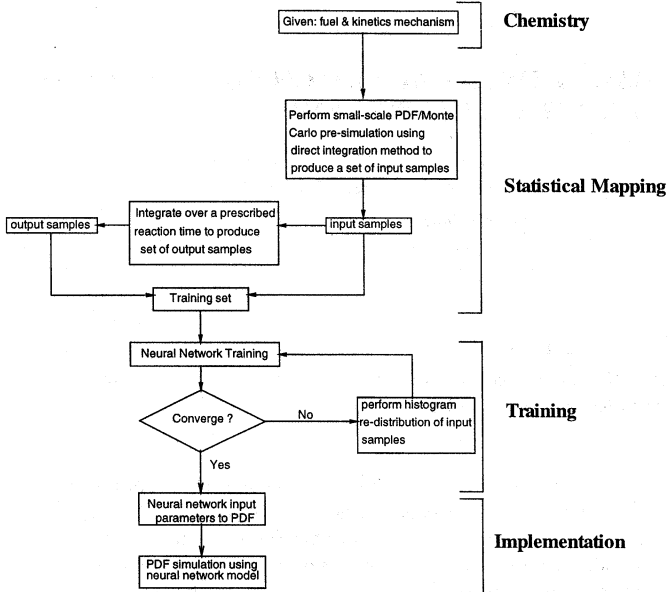


FIG. 1. Schematic illustration of an integrated PDF/neural network algorithm.

space. This leads to inaccuracies and difficulties in obtaining adequate network convergence.

The purpose of this paper is to present an integrated PDF/neural network approach to simulate turbulent flames by solving the transport equation for the joint PDF of velocity–composition–dissipation using the Monte Carlo technique. Once the fuel and the chemistry are specified, small-scale PDF simulations are first performed using direct integration to generate a training set for the neural network. This approach is called *statistical mapping*. This set is then used to train the network that is subsequently used in the PDF simulation of flames for the same fuel. If difficulty is encountered during the training of the network, a procedure for manipulating the training set that could ease convergence is presented. The validity of the approach is demonstrated here for the piloted flame albeit using very simple chemistry.

Integrated PDF/Neural Network Approach

The integration of the ANN and PDF approaches is aimed at making the simulation of turbulent flames using realistic chemistry as simple as possible and with minimal intervention by the operator. The procedure is described later and is shown schematically in Fig. 1. The approach consists of the following steps:

- *Chemistry*: For a given fuel, an appropriately reduced chemical kinetics mechanism is selected and the production rate of the reactive variables that are adopted in the calculations is determined.

- *Statistical Mapping*: The training set for the ANN is generated in two steps. First, small-scale PDF/Monte Carlo presimulations are performed using direct integration for the chemistry. This produces a set of input samples that are used to generate a set of composition increments over a given reaction time. More details about this process are given in the next section.
- *ANN Training*: Neural network training is then carried out using the training set until an appropriate convergence of the algorithm is obtained. If the network cannot be trained, then a histogram redistribution technique is applied to the input samples (this technique is described later) and the training of the network is repeated.
- *Implementation*: Once the network has converged to a satisfactory level, the neural network model is incorporated in the actual PDF simulation of turbulent flames.

Statistical Mapping Approach

The statistical mapping approach involves performing one or more small-scale PDF/Monte Carlo presimulations of turbulent flames using the same chemistry for which the training set is required. Compositions accessed during the simulation provide a good and representative coverage of the accessible space, and these are stored forming the input samples of the training set. In principle, those samples represent a set of potentially reactive compositions but do not ensure that condition. The output samples for the training set are then generated for each input by direct integration over specified

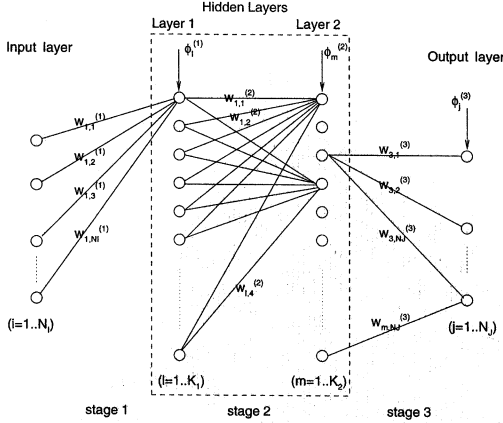


FIG. 2. Structure of multilayer perceptron (MLP) neural network architecture.

reaction time intervals. This approach is expected to result in smoother training of the neural network in terms of both convergence and accuracy. In addition, the size of the statistically mapped set is about 10% of the size of a set that would be used with the look-up table approach.

A new version of the PDF code is used to solve the transport equation for the joint PDF of velocity, composition, and dissipation. The IEM (interaction by exchange with the mean) model is used to represent molecular mixing. The new code is restricted to two-dimensional flows and uses the particle-mesh numerical method. A rectangular grid is used and the mean properties are determined for each grid point. The instantaneous properties are carried by stochastic particles within each cell. Detailed chemical kinetics has not yet been implemented into the code and a very simple thermochemical model that simulates one-step chemical reaction is used here. The reaction rate is given as a function of two variables: mixture fraction ξ , and b , where b is a reaction progress variable that varies between 0 and 1. The stoichiometric mixture fraction is $\xi_s = 0.5$ and the reactive limits extend from 0.3 to 0.7 in the mixture fraction range. It should be noted here that although this chemistry does not refer directly to any particular fuel, it could be related to a fuel mixture of H_2/N_2 where ξ_s is similar and the global chemistry may be represented by a one-step reaction.

The pilot-stabilized burner [11], which consists of a central fuel tube $D_j = 7.2$ mm in diameter, surrounded by an 18-mm annulus for the premixed flame pilot, is used here. All computations are performed for the following conditions: bulk jet velocity, $\bar{u}_j = 41$ m/s, burned pilot velocity, $\bar{u}_{pb} = 24$ m/s, co-flow air velocity, $\bar{u}_a = 15$ m/s. The initial conditions for the velocity and turbulence profiles are identical to those specified earlier for similar jet

flames [12]. The values of mixture fraction at the exit plane of the jet, pilot, and air streams are 1, ξ_s , and 0, respectively. The solution grid covers the region from $x/D_j = 0$ to 25 and $r/R_j = 0$ to 10. The grid is not uniform and has 31×21 nodes in the X - Y directions. This is considered adequate for the purpose of this paper. Further details about the PDF code, the thermochemistry, and the solution procedure may be found elsewhere [14].

Artificial Neural Network

A multilayer perceptron (MLP) architecture is used for the neural network. The network's topology consists of two hidden layers with equal number of neurons in each layer, as shown in Fig. 2. A back-propagation supervised learning algorithm is adopted [15,16]. To reduce the possibility of the network being trapped in a suboptimal minimum and to increase the convergence rate of the algorithm, individual momentum terms and adaptive learning rates adjustment are used for the weights and bias matrices. A novel dynamic randomization procedure [8] is also included to improve the convergence of the algorithm. The network's architecture, optimization, and performance are described in detail by Christo et al. [8]. The convergence of the algorithm is measured in terms of an average error function, \bar{E}_p , defined as follows:

$$\bar{E}_p = \sum_{i=1}^N \sum_{j=1}^K [d^{(j)} - y^{(j)}]^2 / (N \cdot K) \quad (1)$$

where, $y^{(j)}$ and $d^{(j)}$ are, respectively, the actual network model output and the desired output of sample i (out of N training samples) for species j out of total K reactive species.

The implementation of the neural network model into the joint PDF code is carried out by modifying the reaction routine to access a neural network model routine for calculating the composition changes. For a given reactive composition (ξ , Γ_i , $i = 1 \dots N_I$), the composition changes ($\Delta \Gamma_j$, $j = 1 \dots N_J$) over a specific reaction time are obtained by applying the neural network model according to the following algorithm:

$$\Delta \Gamma_j = \sum_{i=1}^{N_I} \sum_{m=1}^{K_1} x_{2,m} w_{j,m}^{(3)} + \phi_j^{(3)} \quad (2)$$

$$x_{2,m} = \tanh \left[\sum_{l=1}^{K_1} x_{1,l} w_{m,l}^{(2)} + \phi_l^{(2)} \right] \quad (3)$$

$$x_{1,l} = \tanh \left[\sum_{i=1}^{N_I} \Gamma_i w_{l,i}^{(1)} + \phi_l^{(1)} \right] \quad (4)$$

where Γ_j is the specific molar abundance of species i ($\Gamma_j = Y_i/W_i$, Y_i , W_i are the mass fraction and molecular weight of species i , respectively); w and ϕ are, respectively, the weights matrices and bias vec-

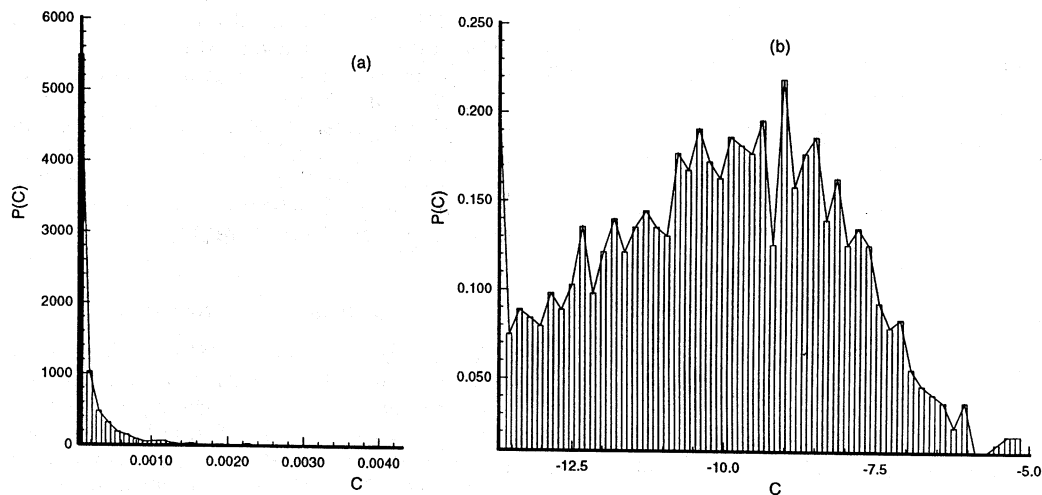


FIG. 3. Probability density function (PDF) of hydrogen radical specific molar abundance, in mixture fraction space for (a) statistically mapped samples and (b) transformed samples (natural logarithm).

tors obtained from the converged algorithm of the neural network; K_1 , K_2 are the number of neurons in the first and second hidden layers, respectively. N_I , N_f denote the number of input variables (number of reactive species plus one) and the number of reactive species, respectively. The superscripts 1, 2, and 3 in Eqs. (2) through (4) refer to the location before, within, and after the two hidden layers, respectively, as shown in Fig. 2. For the cases in which the chemistry is presented for multiple time intervals, each interval is handled by a separate neural model (i.e., different w and ϕ values). In the present study, computations are performed using the following values: $K_1 = K_2 = 8$, $K = N_I = 2$ (ξ and Γ_b), $N_f = 1$ ($\Delta\Gamma_b$), and $N = 1375$ samples.

Histogram Redistribution

The chemistry used here is very simple and the ANN has no difficulty in converging, regardless of the sample distribution in the training set. However, where more complex chemistry is used, the risk of the algorithm being trapped in suboptimal local minima increases. Detailed inspection of the training set used indicated that the convergence of the algorithm may depend on the nonuniformity in the distribution of the input samples in scalar space. A typical example is given in Fig. 3a for a highly skewed distribution of samples for composition C , which may cause a slowdown or total lack of network convergence. A technique known as *histogram redistribution* is used to transform the distorted distribution of the PDF into a more uniform one without altering the composition of the samples. The histogram redistribution technique is a special case of a general

algorithm known as histogram equalization, which is often used to the same effect in digital signal processing and image enhancement applications [17]. The histogram redistribution technique is based on using a nonlinear transformation to create a uniformly distributed PDF of the original samples. The transformation should be a continuous single-value function. Its specific form depends on the shape of the PDF and the dynamic range of the original samples. A natural logarithmic function is used here for the distribution shown in Fig. 3a and proved adequate in achieving good equalization of the histograms, as shown in Fig. 3b. The transformed set is then used as a training set for the neural network. It is found that applying histogram redistribution for the input set seems to improve the convergence of the neural network and produces error-function values that are smaller by a factor of 3 in comparison to those obtained using nontransformed sets.

The histogram redistribution technique is used whenever the neural network algorithm is not converging. However, the success of the transformation is not always guaranteed because this depends not only of the distribution of the inputs but on the distribution of the outputs as well.

PDF/Neural Network Simulation

The ANN models are coupled to the PDF code as plug-in modules. A range of reaction time increments may be used and should be selected to suit the timescale of the problem under consideration and to minimize interpolation error. Results are presented for the simple test case using ANN models for two time increments. The ANN results are com-

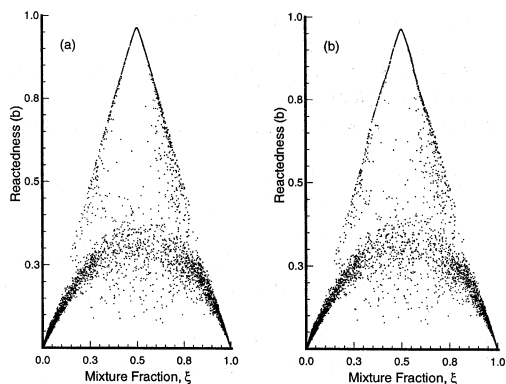


FIG. 4. Scatter plots of reactedness b , against mixture fraction ξ , across the physical space $x/D_j = 0-25$, obtained by using two different representations of the chemistry in the PDF transport equation: (a) direct integration, (b) neural network.

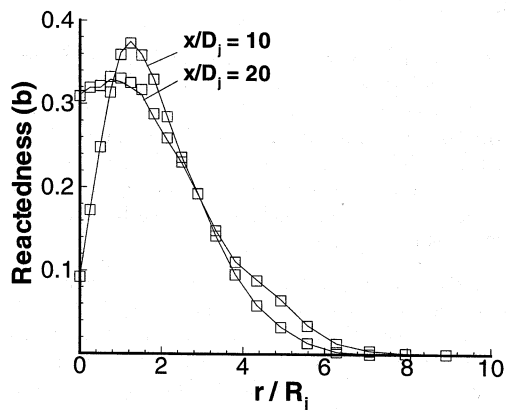


FIG. 5. Radial profiles of mean reactedness b , against mixture fraction ξ , at different axial locations, obtained by two different representations of the chemistry in the PDF transport equation. Symbols: — direct integration, \square neural network.

pared with those obtained using direct integration for converged solutions obtained after 10,000 iterations.

Figure 4 shows scatter plots of reactedness b , plotted versus mixture fraction ξ , for the entire solution domain ($x/D_j = 0-25$, $r/R_j = 0-10$). Results using the ANN and DI methods are presented. The agreement obtained is very good and is reinforced by the radial profiles of mean reactedness shown in Fig. 5, for $x/D_j = 10$ and 20. The results demonstrate clearly that ANN is providing an excellent representation of the chemistry. It should be noted, however, that any error associated with the ANN will accumulate over the large number of steps associated

with the PDF simulations. It is important, therefore, to keep this error to an absolute minimum, especially in the general case of multistep chemistry.

The computational cost of calculating the chemistry using the ANN approach is $\sim 20\%$ of the total CPU time of the PDF simulation, compared to $\sim 4\%$ if the DI method is used. This percentage changes depending on the number of time increments used in the ANN approach and on the distribution of these time intervals relative to characteristic time-scales of the flow. The large computational requirement of the ANN approach is due to the use of only single-step chemistry. It has been demonstrated earlier [8] that the full potential of the ANN approach, in terms of CPU and memory storage savings, is only realized when multistep chemistry is used. The simple cases presented here only demonstrate the procedure of an integrated PDF/neural network method and its feasibility.

Conclusions

An integrated approach coupling a neural network for the representation of chemistry to the PDF simulation of turbulent combustion is presented here. This approach is aimed at overcoming the difficulty associated with extreme computer memory and run time required for the classical look-up tables and direct integration methods.

Statistical mapping is shown to be a superior method for generating the training set for the network. This requires little or no intervention by the user and gives a better representation of the accessible composition space. When difficulties of neural network convergence arise because of the training set, a histogram redistribution technique is applied to smooth the PDFs of the compositions in the training set.

The feasibility of the approach is demonstrated here using single-step chemistry. The full potential of this approach, however, will be achieved with the implementation of multistep chemistry in the PDF simulations.

Acknowledgments

The Sydney University group acknowledges the financial support of the Australian Research Council. Professor Pope's work is supported in part by the U.S. Department of Energy under Grant DE-FG02-90ER 14128.

REFERENCES

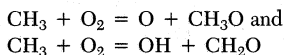
1. Nguyen, T. V. and Pope S. B., *Combust. Sci. Technol.* 42:13-45 (1984).
2. Pope, S. B., *Prog. Energy Combust. Sci.* 11:119-192 (1985).

3. Maas, U. and Pope, S. B., *Combust. Flame* 88:239–264 (1992).
4. Maas, U. and Pope, S. B., *Twenty-Fourth Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, 1992, pp. 103–112.
5. Maas, U. and Pope, S. B., *Twenty-Fifth Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, 1994, pp. 1349–1356.
6. Turányi, T., *Computers Chem.* 18(1):45–54 (1994).
7. Turányi, T., *Twenty-Fifth Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, 1995, pp. 948–955.
8. Christo, F. C., Masri, A. R., and Nebot, E. M., *Combust. Flame*, 106:406–427 (1996).
9. Christo, F. C., Masri, A. R., Nebot, E. M., and Turányi, T., *Proc. IEEE International Conference on Neural Networks, Uni. of Western Australia, Perth*, Vol. 1, 1995, pp. 911–916.
10. Taing, S., Masri, A. R., and Pope, S. B., *Combust. Flame* 95:133–150 (1993).
11. Stårmer, S. H. and Bilger, R. W., *Combust. Flame* 61:29–38 (1985).
12. Masri, A. R. and Pope, S. B., *Combust. Flame* 90:13–29 (1990).
13. Masri, A. R., Dibble, R. W., and Barlow, R. S., *Combust. Flame* 91:285–309 (1992).
14. Masri, A. R., Subramaniam, S., and Pope, S. B., *Twenty-Sixth Symposium (International) on Combustion*, The Combustion Institute, Pittsburgh, 1996, pp. 00–00.
15. Beale, R. and Jakkson, T., *Neural Computing: An Introduction*, Hilger Publishing, Bristol, 1990.
16. Wu, Jian-Kang, *Neural Networks and Simulation Methods*, Marcel Dekker, Inc., New York, 1994.
17. Bracewell, N. R., *Two-Dimensional Imaging*, Prentice Hall, Englewood Cliffs, NJ, 1995.

COMMENTS

Katharina Kohse Höinghaus, Universität Bielefeld, Germany. Your approach would be very valuable under conditions where complex chemistry is important, as, e.g., in pollutant formation. With your approach with two layers I just fail to see how you would deal with situations in which multi-channel reactions would be of great importance, as e.g. in thermal DENO_x type situations. How would the method perform this critical switching between the different pathways when you have more than one channel?

Author's Reply. The neural network “learns” the chemistry as dictated by the chemical kinetic mechanism and the training set which results from it. Multi-channel reactions will be accounted for if they are included in the mechanism. Typical multi-channel reactions are



The pathway followed will depend on temperature and on the Arrhenius parameters for each reaction. Such information will be implicit in the training set provided and hence will be accounted for by the network.

Norberto Fueyo, University of Zaragoza/LITEC, Spain. The most appropriate way of measuring how well your neural network is performing is by measuring the error it makes with respect to the traditional method of integrating the ODE equations for the chemical system. In neural networks, this is traditionally done by setting aside some of your samples in a ‘test data set’, which is not used for training the network but only for assessing how well it performs when presented with data that *are not* in the training set.

Have you followed this methodology? If so, which errors are you obtaining on the test (*not training*) set?

The chemistry system you are using is rather simple, with just one conserved scalar (that, being conserved, does not need to be predicted by the network) and a reactedness. Have you tried more complex systems, such as those including minor species and possibly radicals? If so, which RMS errors are you obtaining?

Author's Reply. Yes, this is exactly the methodology that was followed to test the generalisation capability of the neural network. The data are divided into a “training set” and a “test set”. Details of the training procedure and the errors involved are given in Ref. 8 of the paper. Errors are estimated using an “error function” definition, as given by Eq. 1 in the body of the paper. Typically, acceptable levels of the error-function for the training set are of the order of 1×10^{-6} , but these depend on the nonlinearity of the mechanism as well as the scalars used. The convergence of the training algorithm, however, is satisfied if and only if the values of the error functions (i.e. that of the training set and the testing set) are below a predetermined threshold level. Detailed analysis of the network’s error characteristics are given in Ref. 8. The most important factor in minimising the error is to provide a training set that is most representative of the real system.

We have tried complex systems involving up to ten reactions. An example with five reactions involving radical species like H and OH is described in Ref. 8. The resulting error-function varies with from scalar to scalar and ranges from 1×10^{-4} to 1×10^{-6} . Scalars with pdf’s that are highly skewed generally result in larger errors which can be reduced by histogram normalization as described in this paper. Increasing the number of reactions and scalars does not necessarily increase the difficulty of training or the error.