Symposium on Thermoacoustics in Combustion: Industry meets Academia (SoTiC 2021)
Sept. 6 - Sept. 10, 2021
Munich, Germany
Paper No.: XXX

Real-time parameter inference of nonlinear bluff-body-stabilized flame models using Bayesian neural network ensembles

Maximilian L. Croci[1] [2], Ushnish Sengupta[1] and Matthew P. Juniper[1]

Abstract

This study uses a Bayesian machine learning method to infer the parameters of a physics-based model of a bluff-body-stabilized premixed flame in real-time. An ensemble of neural networks is trained on a library of simulated flame fronts with known parameters, generated using a level-set solver, LSGEN2D. This trained ensemble then observes experimental images of a qualitatively similar flame. The ensemble provides reliable estimates of the parameters and their uncertainties, from which the flame can be re-simulated beyond the observation window of the experiment. Using the re-simulated flame, the flame surface area, a proxy for the heat release rate, is calculated. The method is general: once trained, the ensemble can be used to infer the parameters from any bluff-body-stabilized premixed flame as long as the flame is qualitatively similar and the parameters lie within the ranges in the training library. Recognizing each set of 10 frames takes milliseconds, which is fast enough to work in real-time.

Keywords

Bayesian inference, deep learning, thermoacoustic instability, uncertainty quantification, data assimilation

Introduction

Thermoacoustics

The prediction and control of thermoacoustic instability is a persistent challenge in jet and rocket engine design[1]. The drive towards lower NO_x emissions in gas turbines has led to the use of lean premixed combustion, which is particularly susceptible to thermoacoustic instabilities[2]. When the heat release rate and the pressure are in phase during combustion, thermoacoustic instabilities can arise[3]. Heat release rate fluctuations are caused by flame surface area fluctuations, which in turn are caused by velocity perturbations and flame dynamics[4–7]. Any physics-based model must therefore contain the flame's response to velocity perturbations. Detailed CFD simulations of the flame can be used to calculate this response. However, these CFD simulations are expensive. In this paper we use data to tune the parameters of physics-based reduced-order models, in order to reduce the cost while retaining as much accuracy as possible.

In the simplest physics-based model, the heat release rate fluctuation is a linear multiple of the velocity perturbation at the base of the flame some time earlier. The time taken for perturbations to travel down the flame is modelled by a delay, τ. This is known as the $n-\tau$ model[8]. This model is too simple for our purposes as it cannot be used to simulate the flame dynamics. In this paper we model the flame as the zero contour (or level-set) of a continuous function that advects with the flow. This is known as the G-equation model[9] and allows the flame dynamics to be simulated cheaply. To render the model quantitatively accurate, the parameters of this model need to be assimilated from experimental data. The ensemble Kalman filter[10] (EnKF) has been used previously to assimilate data into the G-equation model[11,12]. The EnKF performs Bayesian inference to infer the parameters of the G-equation model by statistically combining model forecasts with experimental measurements. However, the computational requirements of the EnKF render online Bayesian inference unfeasible when measurements are taken at high frequency. Furthermore, the method can fail to infer parameters when these vary quickly in time, and suffers from numerical stability issues when the measurements are noisy. These are both the case for our experimental data, which come from high frequency OH planar laser induced fluorescence (OH PLIF) measurements of a version of the the Volvo combustor rig[13]. This study uses an alternative method for practical online assimilation of data into the G-equation model of a bluff-body-stabilized flame.

Bayesian deep learning

Bayesian deep learning refers to the use of deep learning algorithms, such as deep artificial neural networks (NNs), for Bayesian inference[14]. Bayesian NNs[15] replace the point estimates of each of the NN's weights and biases with Gaussian probability distributions, with means and variances learned during training. The distribution of every weight and bias in the NN can be used to infer the outputs from the inputs, for example inferring the parameters of a model from experimental measurements. Unfortunately, Bayesian

[1]Department of Engineering, University of Cambridge, Cambridge, UK.,
[2]Code available at: https://github.com/nailimixaM/sotic-2021-baynne

Corresponding author:
Matthew P. Juniper
Email: mpj1001@cam.ac.uk

NNs of practical size are too expensive to train[16]. More recently, ensembles of deep, wide NNs have been used to perform approximate Bayesian inference[17–19], with the approximation improving with increasing width of the NN's hidden layers. These Bayesian NN ensembles (BayNNEs) learn the mean and variance of the posterior distribution of the outputs given the inputs. When multiple outputs are to be inferred, heteroscedastic BayNNEs learn the means and variances of each output, without assuming a common variance for all outputs. This study uses heteroscedastic BayNNEs to infer the parameters of the G-equation model given experimental observations[20,21].

Methods

bluff-body-stabilized premixed flame experiment

Experiments are performed on a version of the Volvo combustor[13,22,23] shown schematically in Figure 1. Premixed air and propane flow into the combustor and are burnt by a flame stabilized on a triangular bluff body with side length $D = 3.8$ cm. Images of the flame are recorded at 10kHz using OH planar laser induced fluorescence (OH PLIF) through a window $3D$ tall and $3.4D$ wide. As the air-fuel mixture flows through the combustor, vortices are shed periodically from the bluff body. These vortices cause wrinkling and cusping of the flame front.

The images are processed to find discretisations of the position $y = f(x)$ of the flame front by thresholding and interpolating the magnitude of the OH gradient vector at each point. The vectors of positions $\mathbf{y}$ are smoothed using splines with 10 knots. To create an observation vector $\mathbf{z}$ representing a sequence of 10 flame front positions, 10 position vectors are appended together.

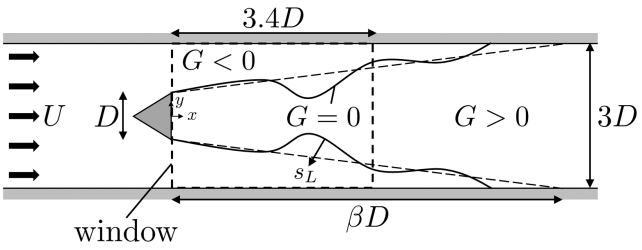

Figure 1. Diagram of the Volvo combustor rig and G-equation model of the flame. A flame burns premixed air and propane stabilized on a triangular bluff body. As the air-fuel mixture flows through the combustor, vortex shedding causes wrinkling and cusping of the flame front. In the G-equation model, the flame front is represented by the $G = 0$ contour (or level-set) of a continuous scalar field $G(x, y, t)$. Unburnt and burnt gases are regions where $G < 0$ and $G > 0$ respectively. The flame front travels normal to itself into the unburnt gases with speed s_L. The flame front advects under the prescribed velocity field, which comprises continuity-obeying velocity perturbations $u'(x, y, t)$ and $v'(x, y, t)$ superimposed onto a steady base flow U.

Flame front model

The flame front is assumed to be a thin boundary between unburnt and burnt gases (see Figure 1). The flame travels normal to itself into the unburnt gases with laminar flame speed s_L which depends on the gas composition. The velocity in the burnt gases does not affect the flame kinematics. The unburnt and burnt gases are assumed to travel with velocity $\mathbf{u}(x, y, t)$. Under these assumptions, the flame front is modelled by the $G(x, y, t) = 0$ contour (or level-set) of a continuous scalar field G whose motion is governed by the G-equation:

$$\frac{\partial G}{\partial t} + \mathbf{u} \cdot \nabla G = s_L |\nabla G|. \tag{1}$$

The flow velocity field $\mathbf{u}$ is assumed to comprise a constant and uniform base flow U and super-imposed continuity-obeying velocity perturbations $u'(x, y, t)$ and $v'(x, y, t)$:

$$\frac{\mathbf{u}(x, y, t)}{U} = (1 + u'(x, y, t))\,\mathbf{i} + v'(x, y, t)\mathbf{j}, \tag{2}$$

$$u'(x, y, t) = \epsilon \left(\frac{x}{D}\right)^{\eta} \sin\left(\mathrm{St}\left(K\left(\frac{x}{D}\right) - t\right)\right), \tag{3}$$

$$\begin{aligned} v'(x, y, t) =& \epsilon\eta \left(\frac{x}{D}\right)^{\eta-1} \left(\frac{y}{D}\right) \sin\left(\mathrm{St}\left(K\left(\frac{x}{D}\right) - t\right)\right) \\ &- \mathrm{St}K\epsilon \left(\frac{x}{D}\right)^{\eta} \left(\frac{y}{D}\right) \cos\left(\mathrm{St}\left(K\left(\frac{x}{D}\right) - t\right)\right), \end{aligned} \tag{4}$$

where U is a characteristic speed, ϵ is a non-dimensional amplitude, St is the Strouhal number of the flame with characteristic length D, excitation frequency f, and nominal aspect ratio β: $\mathrm{St} = 2\pi f \beta D/U$, and K is the ratio of the characteristic speed U to the perturbation phase speed. The parameter η is introduced to the flame perturbation model to allow for the horizontal velocity perturbations to increase in size with distance from the flame holder, which is the qualitative behaviour observed in the experiment. This has proven to be a versatile flame front model in several previous studies, despite having only a few parameters[24]. To make the G-equation model quantitatively accurate, the parameters K, ϵ, η, St and β must be tuned to fit an observed flame shape.

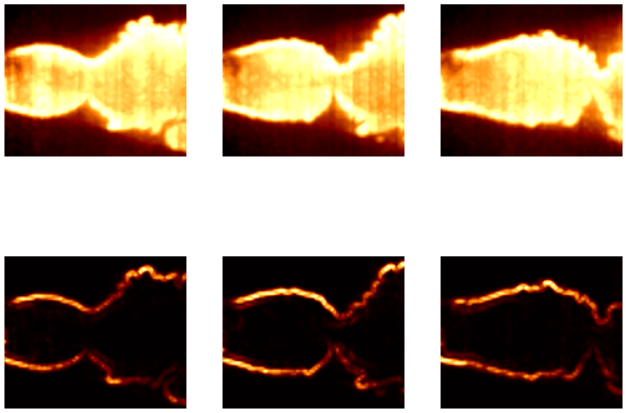

Figure 2. *(Top)* Plots of OH planar laser induced fluorescence (OH PLIF) intensity images of the flame taken through the observation window of the rig. *(Bottom)* Plots of the magnitude of the gradient vector of OH intensity, which is taken to be the flame front.

Simulated flame front library

LSGEN2D[25] is a level-set solver that iterates the G field of the G-equation model for known parameters K, ϵ, η, St and

β. In this study, the G field is iterated until convergence to a set of 200 different periodic solutions. This is repeated for 2400 combinations of parameters sampled from the ranges shown in Table 1. The forced cycle states are processed to find a $y = f(x)$ discretisation of the $G = 0$ contour, for all x in the range of the experiment observation window. This is done by interpolating the G field values for every vertical coordinate, and recording the positions y in vectors $\mathbf{y}$. To create a single observation vector $\mathbf{z}$ representing a sequence of 10 flame front positions, 10 position vectors are appended together. This is repeated for every state in the forced cycle. The result is a library of 4.8×10^5 observation - parameter pairs.

Table 1. Parameters of the G-equation model that are varied in this study and the range over which they are varied in order to generate the synthetic flame front library. v_p is the perturbation phase speed.

Parameter	Range	Description
K	0.5 - 2	Ratio U/v_p
ϵ	0 - 0.5	Perturbation amplitude
η	0 - 3	Perturbation exponent
St	5 - 30	Strouhal number
β	4 - 8	Flame aspect ratio

Inference using a heteroscedastic Bayesian neural network ensemble

The posterior probability distribution $p(\mathbf{t}|\mathbf{z})$ of the G-equation parameters $\mathbf{t}$, given the observations $\mathbf{z}$ is assumed to be a multivariate Gaussian with mean vector $\boldsymbol{\mu}(\mathbf{z})$ and diagonal covariance matrix $\boldsymbol{\Sigma}(\mathbf{z}) = \text{diag}(\boldsymbol{\sigma}^2(\mathbf{z}))$. An ensemble of M neural networks are trained on the synthetic flame front library to predict the mean and variance vectors, $\boldsymbol{\mu}(\mathbf{z})$ and $\boldsymbol{\sigma}^2(\mathbf{z})$. Each neural network in the ensemble produces estimates $\boldsymbol{\mu}_j(\mathbf{z}_i)$ and $\boldsymbol{\sigma}_j^2(\mathbf{z}_i)$ for each observation vector $\mathbf{z}_i$. These estimates are combined as follows:

$$\boldsymbol{\mu}(\mathbf{z}_i) = \frac{1}{M}\sum_j \boldsymbol{\mu}_j(\mathbf{z}_i), \tag{5}$$

$$\boldsymbol{\sigma}^2(\mathbf{z}_i) = \frac{1}{M}\sum_j \boldsymbol{\sigma}_j^2(\mathbf{z}_i) + \frac{1}{M}\sum_j \boldsymbol{\mu}_j^2(\mathbf{z}_i) - \boldsymbol{\mu}^2(\mathbf{z}_i), \tag{6}$$

following Ref.[26]. Each neural network comprises 4 fully connected layers 600 hidden units wide, and two output layers (one for the mean vector, one for the variance vector) each 6 units wide. ReLU activations are used for the hidden layers, a sigmoid activation is used for the output layer for the mean and an exponential activation is used for the variance layer, to ensure positivity. The architecture of one such neural network is shown in Figure 3. Further hyperparameter details are listed in Table 3. The weights $\boldsymbol{\theta}_j$ of each neural network are initialised by sampling from Gaussian prior distributions with means $\mathbf{0}$ and covariance matrices $\boldsymbol{\Sigma}_{prior}$ according to He normalisation[27]. During training, the weights are anchored to their initial values $\boldsymbol{\theta}_{j,anc}$. The loss function used for training is:

$$\mathcal{L}_j = \left(\boldsymbol{\mu}_j(\mathbf{z}) - \mathbf{t}\right)^T \boldsymbol{\Sigma}_j(\mathbf{z})^{-1} \left(\boldsymbol{\mu}_j(\mathbf{z}) - \mathbf{t}\right) + \log\left(|\boldsymbol{\Sigma}_j(\mathbf{z})|\right) + \left(\boldsymbol{\theta}_j - \boldsymbol{\theta}_{anc,j}\right)^T \boldsymbol{\Sigma}_{prior}^{-1} \left(\boldsymbol{\theta}_j - \boldsymbol{\theta}_{anc,j}\right). \tag{7}$$

Training the ensemble in this way is known as Bayesian ensembling with maximum a-posteriori (MAP) sampling. An ensemble of size $M = 20$ is trained for 100 epochs on a Tesla P100 GPU on Google Colab*. This takes approximately 3 hours. Once converged, the ensemble is evaluated on the observations, which takes milliseconds.

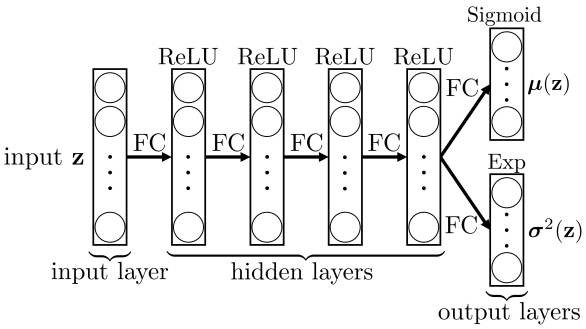

Figure 3. Architecture of each neural network in the ensemble of 20. The input and hidden layers have 600 units each, while each output layer has 6 units each. All layers are fully connected (FC). Rectified Linear Unit (ReLU) activation functions are used for the hidden layers and sigmoid and exponential (Exp) activation functions are used for the mean and variance output layers respectively.

Results

As each flame front position vector appears in 10 consecutive observation vectors, there are 10 sets of ensemble parameter predictions for each flame time step. For each time step, the re-simulated flame is a weighted average of the flames re-simulated with the 10 sets of parameters, with a greater weight given to flames re-simulated using parameters predicted from an observation vector with the flame front position vector appearing in the middle of the observation vector.

Figures 4a - 4d show the experimental images, the inferred flame positions, and the predictions of the BayNNEs for 4 consecutive time steps. Each figure also shows the inferred flame surface area $\bar{a}(x)$ as a function of downstream distance, both in the observed region and in the un-observed region. As expected, the uncertainty of the flame surface area increases beyond the observed region. The inferred flames match the experimental images well in the observed region. The appropriately-tuned flame model matches the oscillations in the flame front position and their evolution with time†. This shows that the method is working well in the observed region. The model then extrapolates beyond the observed region, using a velocity field (for the unburnt

*Google Colaboratory ("Colab") is a Jupyter notebook environment for interactive development, https://colab.research.google.com/.

†Supplementary videos can be found at: https://github.com/nailimixaM/sotic-2021-baynne.

gas) that obeys continuity and that retains the spatial growth of oscillation amplitude observed in the observation window. Although there are no downstream observations, the predicted flame positions seem reasonable. The parameters and uncertainties at each time step, shown in Table 2, take 0.5 ms per ensemble member to calculate on a laptop.

Table 2. Mean $\pm$ 3 standard deviation parameter predictions by the ensemble for the sequence of images in Figs 4a to 4d.

Param.	Fig. 4a	Fig. 4b	Fig. 4c	Fig. 4d
K	0.7 ± 0.4	0.7 ± 0.2	1.0 ± 0.4	1.1 ± 0.4
ϵ	0.3 ± 0.2	0.3 ± 0.1	0.2 ± 0.1	0.2 ± 0.1
η	1.5 ± 0.5	1.6 ± 0.2	1.3 ± 0.5	1.1 ± 0.5
St	23.5 ± 6.8	23.9 ± 6.6	16.6 ± 4.2	15.6 ± 2.9
β	7.5 ± 0.7	7.4 ± 1.0	6.8 ± 0.8	7.2 ± 0.7

Conclusions

This study uses a Bayesian machine learning method to infer the parameters of a physics-based model of the flame front of a bluff-body-stabilized premixed flame. The method comprises an expensive neural network training step after which the parameters of any bluff-body-stabilized flame experiment can be inferred, so long as the parameters fall within the range of the training data set, in milliseconds. The surface area per unit depth, a proxy for the heat release rate, is calculated as a function of downstream distance by re-simulating the flame beyond the experiment observation window. Using this method, parameters can be inferred reliably and in real-time.

This study shows how BayNNEs can be trained to recognise the parameters of numerical simulations from images of those simulations, and then used to find the simulation that best fits experimental results. It is demonstrated here for a kinematic flame model, but this could in principle be performed for any CFD solution. Firstly, this provides a cheap way to store CFD data - e.g. the parameters of the most relevant CFD solution for a given experiment can be extracted cheaply, and the CFD solution then re-calculated. Secondly, it shows how sparse experimental results can be combined with complete numerical results to extrapolate, with defined confidence levels, beyond experimental observations.

Funding

This project has received funding from the UK Engineering and Physical Sciences Research Council (EPSRC) award EP/N509620/1 and from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement number 766264.

References

1. Juniper, M.P., Sujith, R.: "Sensitivity and Nonlinearity of Thermoacoustic Oscillations", Annual Review of Fluid Mechanics (2018).
2. Keller, J.J.: "Thermoacoustic Oscillations in Combustion Chambers of Gas Turbines", AIAA 33-12 (1995).
3. Strutt, J.W.: "The Theory of Sound (Vol II)", Macmillan and Co., London, UK (1878).

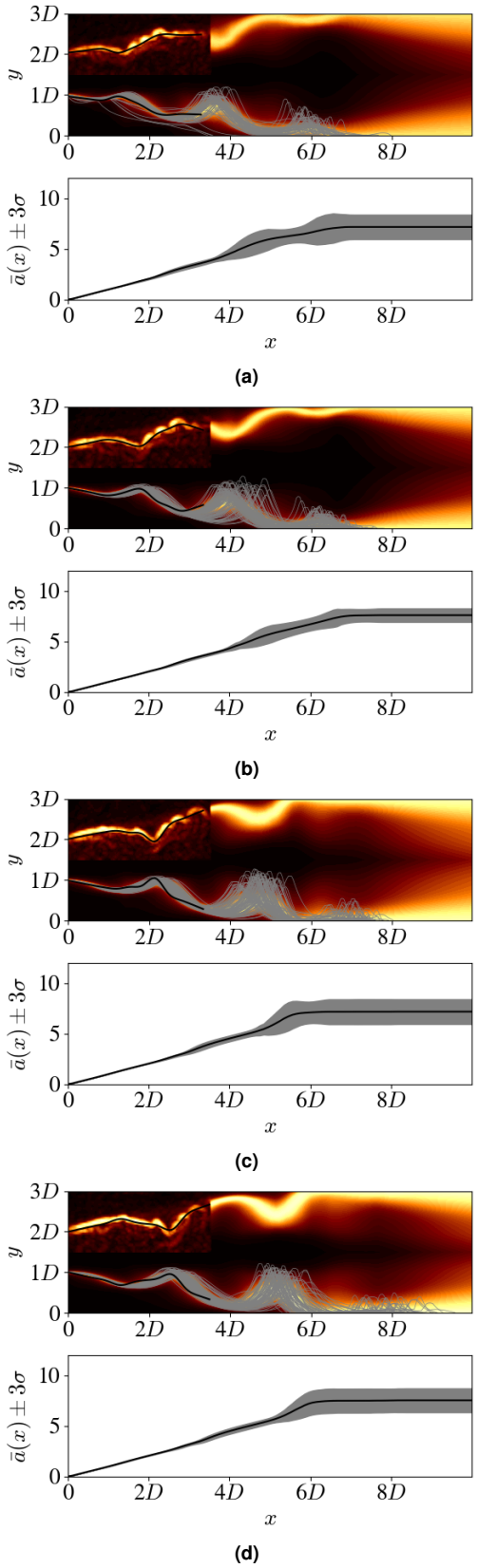

Figure 4. *(a-d, top)* The experimentally-determined flame location (see Fig. 2) is shown in the top half of the image from 0 to $3.4D$. The splined flame location $y(x)$ is shown as a black line. The position of each ensemble member prediction is shown as a grey line in the bottom-half of the image. The weighted mean of these is shown as a colormap in the top and bottom halves. *(a-d, bottom)* The surface area per unit depth $\bar{a}(x)$, as a function of downstream distance x, of the flame front, and its uncertainty.

4. Smart, A.E., Jones, B., Jewel, N.T.: "Measurements of Unsteady Parameters in a Rig Designed to Study Reheat Combustion Instabilities", AIAA 26-88 (1976).
5. Pitz, R.W., Daily, J.W.: "Experimental study of combustion in a turbulent free shear layer formed at a rearward facing step", AIAA 81-106 (1981).
6. Smith, D.A., Zukoski, E.E.: "Combustion instability sustained by unsteady vortex combustion", AIAA 85-1248 (1985).
7. Poinsot, T., Trouve, A., Veynante, D., Candel, S., Esposito, E. : "Vortex-driven acoustically coupled combustion instabilities", Journal of Fluid Mechanics 177-220 (1987), pp 265-292.
8. Crocco, L.: "Research on Combustion Instability in Liquid Propellant Rockets", Symp. (Int.) Combust., 12(1) (1969), pp. 85–99.
9. Williams, F.A.: "Turbulent Combustion", The Mathematics of Combustion (1985), pp. 97-131.
10. Evensen, G.: "Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics", Journal of Geophysical Research 99 (1994) pp. 10143-10162.
11. Yu, H., Juniper, M.P., Magri, L.: "Combined State and Parameter Estimation in Level-Set Methods", J. Comp. Phys. 399 (2019).
12. Yu, H., Juniper, M.P., Magri, L.: "A data-driven kinematic model of a ducted premixed flame", Proceedings of the Combustion Institute, 399 (2020).
13. Caswell, A.W., Rankin, B.A., Huelskamp, B., Jiang, N., Lynch, A., Belovich, V., and Gord, J.R.: "Spatiotemporal Characterization of Flame-Vortex Interactions in Bluff-Body Stabilized Turbulent Premixed Flames Using Simultaneous High-Repetition-Rate OH-PLIF and PIV", 53rd AIAA Aerospace Sciences Meeting, Kissimmee, FL (2015).
14. Gal, Y.: "Uncertainty in Deep Learning", PhD Thesis (2016).
15. MacKay, D.J.C.: "Information theory, inference and learning algorithms", Cambridge university press (2003).
16. Gal, Y., Ghahramani, Z. : "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning", Proceedings of the 33 rd International Conference on Machine Learning, New York, NY, USA (2016).
17. Pearce, T., Zaki, M., Brintrup, A., Anastassacos, N., Neely, A.: "Uncertainty in neural networks: Bayesian ensembling", International Conference on Artificial Intelligence and Statistics (AISTATS) (2020).
18. Sengupta, U., Croci, M.L., Juniper, M.P.: "Real-time parameter inference in reduced-order flame models with heteroscedastic Bayesian neural network ensembles", Machine Learning and the Physical Sciences Workshop at the 34th Conference on Neural Information Processing Systems (NeurIPS) (2020).
19. Sengupta, U., Amos, M., Hosking, J.S., Rasmussen, C.E., Juniper, M.P, Young, P.J.: "Ensembling Geophysical Models with Bayesian Neural Networks", Advances in Neural Information Processing Systems (NeurIPS) (2020).
20. Croci M.L., Sengupta U. and Juniper M.P.: "Online parameter inference for the simulation of a Bunsen flame using heteroscedastic Bayesian neural network ensembles", Deep Learning for Simulation (SimDL) Workshop at the 9th International Conference on Learning Representations (ICLR) (2021).
21. Croci M.L., Sengupta U. and Juniper M.P.: "Data Assimilation Using Heteroscedastic Bayesian Neural Network Ensembles for Reduced-Order Flame Models". In: Paszynski M., Kranzlmüller D., Krzhizhanovskaya V.V., Dongarra J.J., Sloot P.M. (eds) Computational Science – ICCS 2021. ICCS 2021. Lecture Notes in Computer Science, vol 12746. Springer, Cham. (2021).
22. Kiel, B., Garwick, K., Gord, J.R., Miller, J., and Lynch, A.: "A Detailed Investigation of Bluff Body Stabilized Flames," 45th AIAA Aerospace Sciences Meeting and Exhibit, AIAA 2007-168, Reno, NV (2007).
23. Kiel, B., Garwick, K., Lynch, A., and Gord, J.R.: "Non-Reacting and Combusting Flow Investigation of Bluff Bodies in Cross Flow," 42nd Joint Propulsion Conference and Exhibit, AIAA 2006-5234, Sacramento, CA (2006).
24. Kashinath, K. and Li, L.K.B. and Juniper, M.P.: "Forced synchronization of periodic and aperiodic thermoacoustic oscillations: lock-in, bifurcations and open-loop control", Journal of Fluid Mechanics, 838 (2018) pp. 690-714.
25. Hemchandra, S.: "Dynamics of Turbulent Premixed Flames in Acoustic Fields", PhD Thesis (2009).
26. Lakshminarayanan, B., Pritzel, A., Blundell, C.: "Simple and scalable predictive uncertainty estimation using deep ensembles", Advances in neural information processing systems (2017) pp. 6402–6413.
27. He, K., Zhang, X., Ren, S., Sun, J.: "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". IEEE International Conference on Computer Vision (ICCV) (2015) pp. 1026-1034.

Supplemental material

Table 3. Hyperparameter settings used for neural network training.

Hyperparameter	Value
Training	
Train-test split	80:20
Batch size	256
Epochs	100
Optimiser	Adam
Learning rate	10^{-4}
Architecture	
Input units	600
Hidden layers	4
Units per hidden layer	600
Output layers	2
Units per output layer	6
Ensemble size	20