Physical quantities reconstruction in reacting flows with deep learning

Tathawadekar, Nilam; Silva, Camilo; Sitte, Michael Philip; Doan, Nguyen Anh Khoa

Physical quantities reconstruction in reacting flows with deep learning

Nilam Tathawadekar[1]
Technical University of Munich
Bolztmannstr. 3, 85748 Garching, Germany

Camilo Silva[2]
Department of Mechanical Engineering, Technical University of Munich
Bolztmannstr. 15, 85747 Garching, Germany

Michael Philip Sitte[3]
Siemens Mobility Austria GmbH
Leberstr. 34, 1110 Vienna, Austria

Nguyen Anh Khoa Doan[4]
Faculty of Aerospace Engineering, Delft University of Technology
Kluyverweg 1, 2629HS Delft, Netherlands

ABSTRACT

Performing measurements in reacting flows is a challenging task due to the complexity of measuring all quantities of interest simultaneously or limitations in the optical access. To compensate for this, recent advances in deep learning have shown a strong potential in augmenting the information content in datasets composed of partial measurements by reconstructing the quantities that could not be measured. The present work analyses the use of such deep learning tools in two different cases. First, Convolutional Neural Networks (CNNs) are used to reconstruct the heat release rate (HRR) from velocity measurements in a methane/air premixed flame under harmonic excitation. The CNNs are trained from complete datasets at some specific frequencies and amplitudes of excitation and their ablility to reconstruct the HRR for different operating conditions with good accuracy is demonstrated. Secondly, an alternate approach based on Physics-Informed Neural Networks that do not require the training data to have all the quantities is explored. It is applied to a puffing pool fire where the velocity field is reconstructed from observations of pressure, temperature and density with good accuracy.

[1]nilam.tathawadekar@tum.de

[2]camilo.silva@tum.de

[3]philip.sitte@siemens.com

[4]n.a.k.doan@tudelft.nl

1. INTRODUCTION

Measurements in flow has become an essential feature of fluid mechanical research, allowing to improve our understanding of the flow physics or by providing data that can be used to validate mathematical models and numerical simulations [1, 2]. Among these experimental fluid mechanics techniques, a variety of methods have emerged depending on the quantity to be measured. Particle Image Velocimetry (PIV) or Particle Tracking Velocimetry (PTV) have become standard techniques to measure the velocity in flows [3, 4], starting from enabling the measurements along a single line, to nowadays complex three-dimensional stereoscopic measurements that allow to get the three components of velocity in a box region. Beyond velocity measurements, density measurements have been enabled with schlieren techniques [1], while temperature is typically obtained via a hot-wire measurements or Rayleigh-Raman techniques. In the case of reacting flows, added to the previous methods, Laser-Induced Fluorescence (LIF) techniques have been developed to assess the species distributions in flows enabling a finer understanding of reactions occurring therein.

The overview above does not attempt to provide an exhaustive list of flow measurements techniques but highlight the complexity and variety of techniques required depending on the quantity of interest. This complexity often results in the impossibility, in many experiments, of simultaneously measuring all quantities of interest, limiting our ability to analyse the (reacting) flows in detail. To tackle this issue, recent developments in machine learning techniques have shown a great potential [5–13]. For example, methods based on Proper Orthogonal Decomposition (POD) have been used to reconstruct the velocity field from sparse measurements by finding a data-driven based mapping between the measurable quantities (of, for example, sparse sensors) and the velocity field [8] or by using feedforward neural networks to achieve a similar task [9, 12]. For reacting flows, an extension of POD, called Gappy POD, was developed with some success to infer the velocity in regions where it is not measured [13]. Further attempts were made at using Convolutional Neural Networks (CNNs) to reconstruct velocity fields from OH-Planar Laser Induced Fluoresence (PLIF) data [5, 6]. Despite their success, the approaches in these works required a database with both the measured quantities and those to be reconstructed, to train the machine learning framework. To circumvent this, a recent framework was proposed by Raissi et al. [11], called Hidden Fluid Mechanics (HFM), where a feedforward neural network is trained with both the governing equations of the system and the measured quantities to perform such a reconstruction task without requiring data of the quantities to reconstruct. The obtained so-called Physics-Informed Neural Network (PINN) was shown able to reconstruct the velocity field in some canonical non-reacting flows. However, this approach has only been applied to non-reacting flows.

In this paper, we present the reconstruction capabilities of two different deep learning techniques applied to reacting flows. In the first one, presented in Section 2, the reconstruction of the heat release rate will be performed in a Bunsen flame under acoustic excitation by using a deep learning framework, called the U-net that relies on a series of CNNs. In the second test case, presented in Section 3, the reconstruction of the velocity field in a puffing pool fire will be performed using a physics-informed neural network extended to reacting flows. For both test cases, the data is generated from numerical simulations which have been validated against experimental measurements. Only a subset of all the quantities (the "measured" quantities) will be given as input to the deep learning framework. Using numerical data furthermore allows to make a thorough comparison between the quantities reconstructed using the deep learning framework and the actual quantity. A summary of the main results is provided in the final section.

2. HEAT RELEASE RATE RECONSTRUCTION WITH CNNS

2.1. Test case: Bunsen flame under excitation

The Bunsen flame test case is shown in Fig. 1. It is the laminar multi-slit burner investigated by Kornilov et al. [14], where a premixed methane-air mixture with an equivalence ratio of 0.8 is used and is subjected to single tone velocity perturbation with a loudspeaker. The numerical setup and results
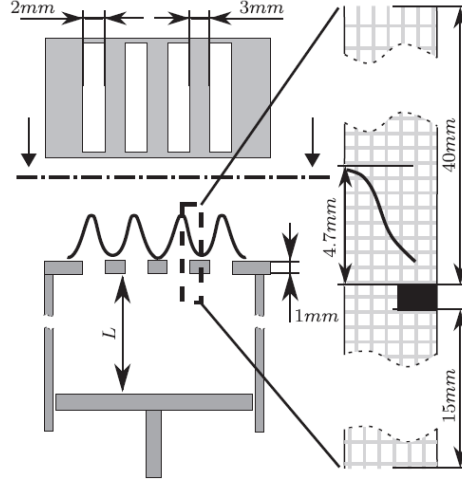

Figure 1: Left: Experimental configuration. Right: CFD domain. Figure adapted from [15].

similar to the one in [16, 17] are used: a 2D CFD domain with symmetric boundary condition in the transverse direction and inflow/outflow boundary condition in the streamwise direction is considered. The two-step chemical scheme as detailed in [18] and OpenFOAM [19] are used for the simulation. Additionally, no combustion model is necessary here as all species transport equations are fully resolved. The flame is laminar and the grid is composed of 122,300 cells with a cell size of 0.025mm in the flame region and area of contractions and cell stretched in the axial direction. This ensures that all flow and reaction lengthscales are fully resolved [15]. At the inflow, a mean inlet velocity of 0.4 m/s and inlet temperature of 293 K are imposed. The plate on which the flame is stabilized is modeled as a no-slip wall with a fixed temperature of 373K, as measured in the experiment [14]. The CFD simulation is run with an adaptive time-stepping scheme with an average timestep $\Delta t = 10^{-6}$. Given that this CFD set-up has already been validated in previous studies with respect to experimental data, these validation steps are not repeated here. The interested reader is referred to [16, 17] for additional details on the numerical set-up and its validation.

This initial set-up is then subjected to acoustic excitation at the inlet, where a normalized harmonic streamwise velocity fluctuation is imposed. The frequencies considered for excitation are $f =$ 100, 150 and 200Hz with a normalized amplitude of $A =$ 10, 50, 100, 125 and 150% of the mean inlet velocity. Different combinations of cases will be used to train the deep learning framework and it will then be tested on a different condition to assess its reconstruction performance.

A typical time-sequence of the varying HRR (normalized by its maximum value) over one period is shown in Fig. 2 for the case with $f = 100$ Hz and $A = 100\%$. It can be seen that the initial flame gets extremely elongated under the large amplitude of excitation before regaining its original shape.

2.2. Reconstruction method with CNNs

In this section, the reconstruction objective is the inference of the heat release rate (HRR) field from the velocity field. To achieve this, a deep learning architecture based on the U-net will be used [20]. This architecture is presented in Fig. 3. It is composed of a series of CNNs which perform multi-level filtering of the input fields and recombines these into one output field. For this reconstruction
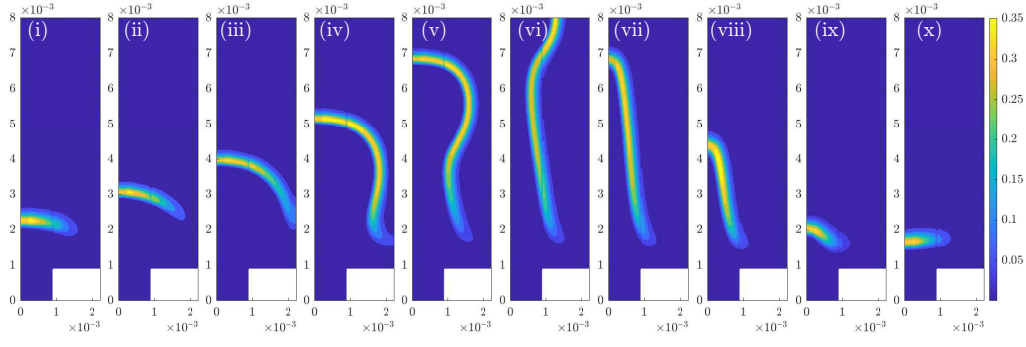
Figure 2: Evolution of HRR over one period of excitation for the case with $f = 100\,\text{Hz}$ and $A = 100\%$. Snapshots are spaced by a time $\Delta t = T/10$.
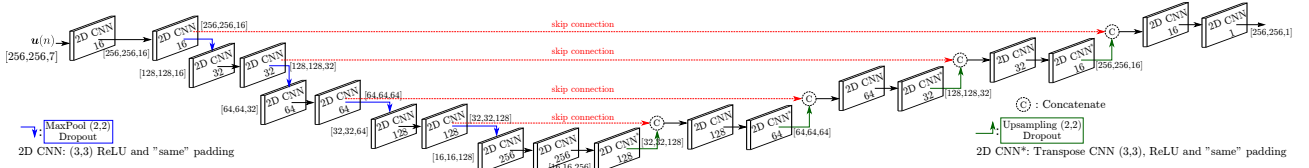

Figure 3: Schematic of the U-net architecture

problem, it will be considered that the U-net takes as input the full (2-component) velocity fields at 3 different successive time instants in the past (i.e. $\mathbf{u}(x, t)$, $\mathbf{u}(x, t - \Delta t)$, $\mathbf{u}(x, t - 2\Delta t)$) and the mean HRR field (i.e. $\langle \dot{q}(x) \rangle_T$, where $\langle \cdot \rangle_T$ represents the time-averaging operation over the period T of the velocity fluctuation) and outputs the HRR fluctuations $d\dot{q} = \dot{q} - \langle \dot{q} \rangle_T$. This combination of information allows the U-net to estimate velocity gradients which are necessary to estimate the HRR fluctuation.

In this set-up, the U-net architecture requires the data of HRR fluctuations for training which may seem limiting. However, we will show that the U-net is also able to infer the HRR fluctuations for operating conditions not present in the training dataset. This situation would mimic an experimental campaign where measurements of HRR are only available for a few operating conditions while the velocity measurements would be available for all of them. It should be noted that the requirement of having the mean HRR as an input is not excessively constraining given that it can be straightforwardly obtained using luminescent photographs for example.

2.3. Results

Two different reconstruction problems are considered. In the first "amplitude interpolation" case, the U-net architecture is trained with the data at specific amplitudes of excitation (10, 50, 100, 150%) and frequencies of 100 and 200 Hz and the U-net is used to reconstruct the HRR fluctuation for the other cases at different amplitudes of excitation (125%), for the same frequencies. In the second "frequency interpolation" case, the U-net is trained with the case at 2 different amplitudes of excitation (50 and 100%) for frequencies 100 and 200Hz and it is tested for a case with frequency of 150 Hz and amplitude 50%.

A typical reconstructed velocity field for a representative timestep for the "amplitude" case is shown in Fig. 4a alongside the original data, where the U-net reconstruct the HRR field for a case whose amplitude was not present in the dataset ($A = 125\%$). It can be seen that the HRR profile is correctly reconstructed from the input (the velocity fields during several past time steps). Especially the formation of the HRR "cusps", which is due to the very large amplitude of the excitation, is recovered. The time evolution of the mean-squared error (MSE) between the predicted HRR field and the exact one is also shown in Fig. 4d where it can be seen that throughout the prediction, the MSE remains small. In addition, an important feature of the flame for thermoacoustic studies is the total

HRR fluctuation (the HRR integrated over the entire domain, $\dot{Q} = \int_V \dot{q}dV$). It is computed using the HRR reconstruction by the U-net ans is shown in Fig. 4d. It can be seen that it closely matches the actual total HRR fluctuation. This indicates that the U-net is able to reconstruct the HRR field from the velocity field, not only in terms of morphology and spatial features, but also in an integral sense. The results for the other test case ($f = 200$Hz and $A = 125\%$) exhibited a similar level of accuracy and are not shown here for brevity.

Figure 4: (a) Actual HRR field, (b) Reconstruction from U-net and (c) difference between the two. (d) Time-evolution of (left axis) the total HRR fluctuation (blue line: U-net, cross: exact) and (right axis) mean squared error for the amplitude interpolation problem (for case $f = 100$Hz and $A = 125\%$).

A similar analysis was performed when considering the frequency interpolation problem, i.e. by requiring the U-net to reconstruct the HRR profile for a case whose frequency of excitation was not present in the training dataset. This is shown in Figs. 5a-c where the exact, reconstructed HRR profiles and their difference are shown. Similarly to the other case, one can observe that the reconstructed HRR matches well the actual one. The U-net is also able to reconstruct the HRR accurately in an integral sense (see the time-evolution of the MSE and total HRR in Fig. 5d). Similar accuracy was also found for the other test case in this problem set-up ($f = 150$Hz with $A = 100\%$) and the associated results are not shown for brevity.
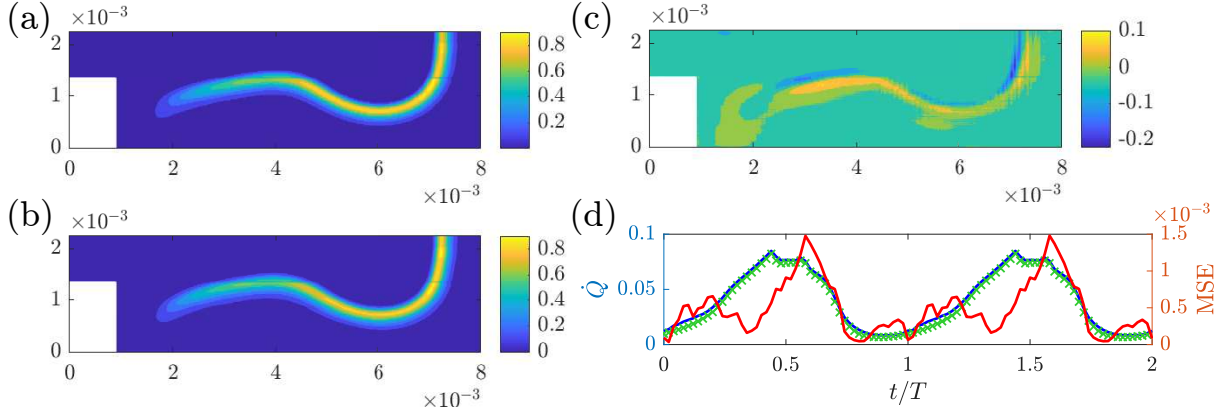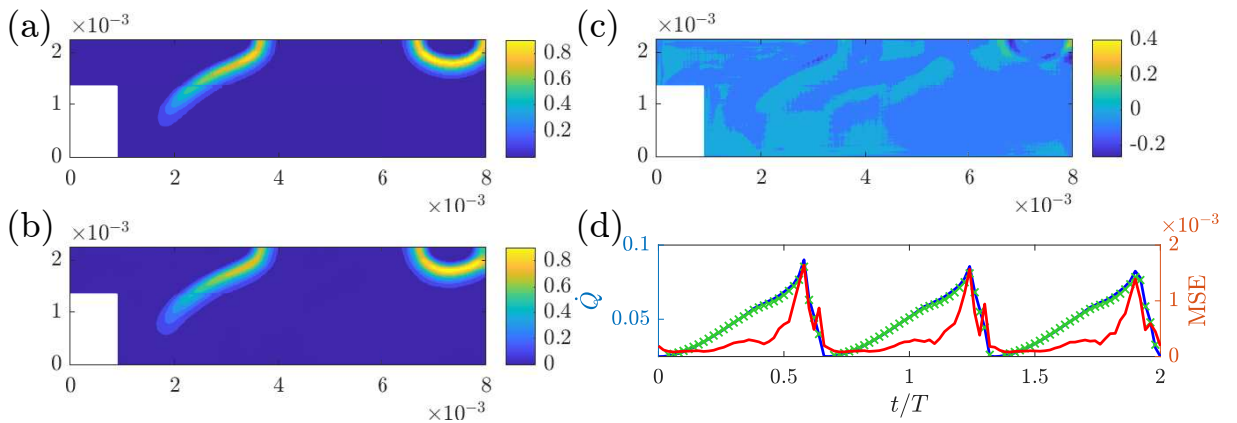

Figure 5: (a) Actual HRR field, (b) Reconstruction from U-net and (c) difference between the two. (d) Time-evolution of (left axis) the total HRR fluctuation (blue line: U-net, cross: exact) and (right axis) mean squared error for the frequency interpolation problem (for case $f = 150$Hz, and $A = 50\%$).

3. VELOCITY RECONSTRUCTION USING PINNS

3.1. Test case: Unstable pool fire

The second test case considered here is an unstable pool fire at the onset of puffing. Simulations were carried out using the computational fluid dynamics toolbox OpenFOAM-7 with the solver *fireFoam* [19]. The governing equations are the continuity equation, Navier-Stokes equations, species mass fraction equations and enthalpy equation.
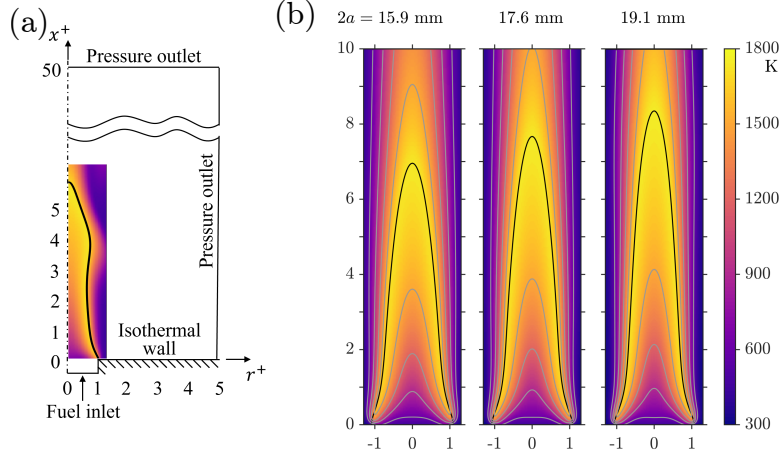

Figure 6: (a) Schematic of the numerical domain. Coordinates normalised by pool radius a. (b) Subcritical flames for different diameters, $2a = 15.9$, 17.6 and 19.1 mm. Black line: position of the reaction zone, indicated by the stoichiometric mixture fraction; grey lines: isocontours for $T = 600$, 900, 1200, 1500 K.

A 2D axisymmetric pool fire with n-heptane fuel and at ambient conditions ($p_0 = 100\,000$ Pa, $T_0 = 300$ K, $Y_{O2} = 0.233$, $Y_{N2} = 0.767$) was investigated, equivalent to the n-heptane flames with an isothermal brass base plate studied experimentally and numerically by Moreno-Boza et al. [21]. A schematic of the numerical domain is shown in Fig. 6a. The domain size was $50a$ in axial and $5a$ in radial direction, where a is the fuel pool radius. A 2D structured grid was used with 600×130 grid points. In the region of interest the grid resolution was $O(0.1$ mm$)$ to ensure that all flow and reaction lengthscales are fully resolved. Numerical schemes were of second order in space and first order in time with a fixed time step of 10^{-5} s, so that CFL < 0.1, and using a fractional step scheme for the chemical source term. In the present simulation, radiation was neglected and turbulence modelling was not necessary since the flame was in the laminar regime at the onset of puffing. Combustion chemistry was modelled as the irreversible single-step reaction ($C_7H_{16} + 11\,O_2 \rightarrow 7\,CO_2 + 8\,H_2O$), whose rate constant is given by the Arrhenius law, $K = BT^\beta \exp(-T_A/T)$ with the model constants $\beta = 0$, $T_A = 12\,000$ K and $B = 5.5 \times 10^7$ m^3/(mol s), following the reasoning of Fernandez-Tarrazo et al. [22] but without correcting heat release rate (HRR) and T_A with equivalence ratio ϕ. This modelling of the chemical reaction was chosen for the sake of simplicity. It is sufficient for the present case of a laminar diffusion flame characterised by high Damköhler number and controlled by mixing [21]. Density is given by the ideal gas law. Viscosity is computed from Sutherland law, $\mu = A_s \sqrt{T}/(1 + T_s/T)$, independent of species composition with $A_s = 1.672 \times 10^{-6}$ and $T_s = 170.67$ in the appropriate SI-units. Molecular and thermal diffusivity are computed based on the assumptions of unity Lewis number, Le $= \alpha/D = 1$, and constant Prandtl number, Pr $= \mu/(\rho\alpha) = 0.7$. Specific heat and enthalpy are obtained from NASA polynomials but modifying the formation enthalpy of n-heptane to reduce the flame temperature to approximately 1750 K to account for the effect of radiative heat loss on the flame, following previous work [21]. The liquid surface is modelled as a boundary condition following Moreno-Boza et al. [21]. The liquid surface is assumed to be at the boiling temperature of n-heptane, $T_B = 371.5$ K. Walls are assumed isothermal at the ambient temperature

T_0. The fuel mass flow rate at the liquid surface is determined by the evaporation rate relating the conductive heat flux to the liquid with the surface normal velocity u_n as in [21].

The CFD simulations are validated against the experimental and numerical results for the n-heptane flame with isothermal walls from Moreno-Boza et al. [21], notably the reproduction of the flame length, the critical point and the puffing frequency. Figure 6b shows temperature contours for the subcritical flames of diameters $2a$ = 15.9, 17.6 and 19.1 mm with corresponding flame lengths of $7.0a$, $7.7a$ and $8.4a$. The flame length was determined based on the downstream tip of the stoichiometric mixture fraction isocontour. These values are in good agreement with the experiments, where the flame lengths ranged from $6.7a$ to $8.0a$. In the present simulations, the critical point – defined as the characteristic diameter where puffing first occurs – is found at $2a$ = 20.1 mm, in line with experimental findings for the isothermal base plate. At this condition, the puffing establishes as a periodic process with a constant frequency of approximately 12.0 Hz, very close to the experimentally determined value of 12.8 Hz. Figure 7 shows one cycle of this puffing behaviour and the temporal variation of HRR during the puffing cycle. In this sequence, we can see the dynamics of the puffing flame, where a "cusp" is formed due to buoyancy-driven vorticity generation. These numerical results closely resembles the time sequence of the flame recorded in the experiments. Therefore, the present simulations have been shown to accurately reproduce the behaviour of the n-heptane pool fire in the vicinity of the critical point, studied by Moreno-Boza et al. [21]. For the reconstruction problem discussed in the next section, the puffing flame with $2a$ = 20.1 mm is considered. The dataset used to train the PINN consisted of 450 snapshots recorded at 1 000 frames per second, corresponding to more than 5 periods with about 83 snapshots per period.
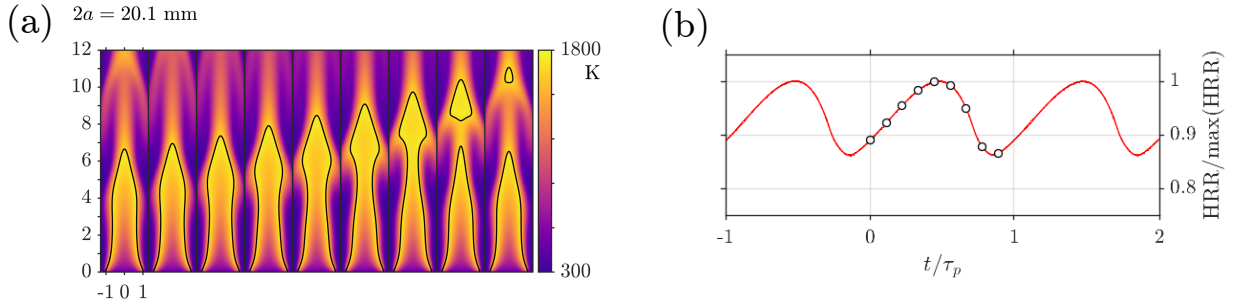

Figure 7: (a) Time sequence of the puffing flame with diameter $2a$ = 20.1 mm. (b) Corresponding time evolution of HRR, normalised by period τ_p and max(HRR); instances of snapshots are indicated as ∘.

3.2. Reconstruction method with PINNs

To reconstruct the unmeasured quantity, the approach called Hidden Fluid Mechanics proposed by Raissi et al. [11] will be used. This approach relies on a Physics-Informed Neural Network [23], illustrated in Fig. 8, and uses a neural network to infer the measured and unmeasured quantities in a given flow which is governed by the (reacting) Navier-Stokes equations. To achieve the inference of unmeasured quantities, the PINN relies on two sources of information: (i) the data which is related to the *measurable* quantities and (ii) the governing equations of the flow. A PINN is a conventional feedforward neural network (blue box in Fig. 8) which is trained with a specific loss function that accounts for the governing physical equations (ϵ_p, red part) and the data error (ϵ_m, green part). Feedforward neural networks map the input to the output, as shown in Fig. 8 where an input, two hidden and an output layers are represented. The network is termed feedforward as the output of a given layer is not fed back into the input or preceding layers. Each hidden layer consists of neurons which are fully connected, meaning that each neuron in a given layer $l - 1$ is connected to all neurons in the following layer l through a weight matrix W_l. Therefore, the intermediate output of the hidden

layer l can be written as $\mathbf{Z}_l = \mathbf{W}_l^T \mathbf{X}_{l-1} + \mathbf{b}_l$ where $\mathbf{X}_{l-1}$ is the output of the layer $l-1$ and $\mathbf{b}_l$ is the bias in layer l. Following this, nonlinearities are introduced through the element-wise activation function g, so that $\mathbf{X}_l = g(\mathbf{Z}_l)$. For what follows, the activation function used for all hidden layers will be the swish function with a linear activation in the final output layer. Other choices of activation functions are possible, such as the sigmoid or ReLU activation, but swish was shown to provide sufficient accuracy for the presented case as will be discussed in Sec. 3.3.
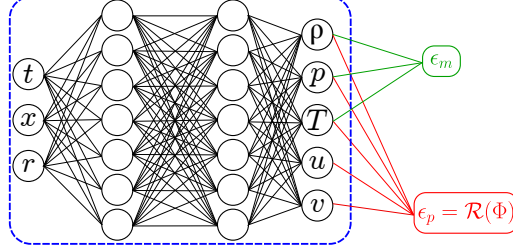

Figure 8: Schematic of the HFM framework.

In the HFM architecture, the PINN takes as input a space-time location $(\boldsymbol{x}, t)$ (here, in 2D axisymmetric coordinates, $\boldsymbol{x} = (x, r)$) and outputs the flow state vector at that location, i.e. its output is $\widetilde{\Phi} = [\rho, p, T, \boldsymbol{u}, \boldsymbol{Y}]$, where $\widetilde{\cdot}$ indicates a prediction from the PINN, $\boldsymbol{u} = (u, v)$ is the velocity vector, with axial component u and radial component v, and $\boldsymbol{Y}$ is the chemical composition vector. For simplicity, in this first attempt at using the HFM, the species mass fraction will not be considered in the reconstruction problem, and therefore, the considered flow state is just $\widetilde{\Phi} = [\rho, p, T, u, v]$. In the specific reconstruction problem considered here, it should be stressed that, for the training of the PINN, only a subset of these outputs have associated *target data* (i.e. the *measured* quantities). Such measured states will subsequently be noted ϕ_m for the target data and $\widetilde{\phi}_m$ for the PINN prediction, with their associated space-time locations noted as $(\boldsymbol{x}_m, t_m)$. In what follows, it will be considered that the measurable quantities are $\phi_m = [\rho, p, T]$. To enable the PINN to reconstruct the other *unmeasured* quantities (in this case, the velocity field $\boldsymbol{u}$), the loss function used to train the PINN includes the residual of the reacting Navier-Stokes equations, noted $\mathcal{R}$, which is estimated at collocation points (noted $(\boldsymbol{x}_c, t_c)$) spread over the time-space domain covered by the simulation. Therefore, the loss function to train the PINN is:

$$\mathcal{L} = \underbrace{\frac{1}{N_m} \sum_{n=1}^{N_m} |\phi_m(\boldsymbol{x}_m^n, t_m^n) - \widetilde{\phi}_m(\boldsymbol{x}_m^n, t_m^n)|^2}_{\epsilon_m} + \underbrace{\frac{1}{N_c} \sum_{i=1}^{N_c} |\mathcal{R}(\widetilde{\Phi}(\boldsymbol{x}_c^i, t_c^i)|^2}_{\epsilon_p} \qquad (1)$$

In the equation above, the first term, ϵ_m (green part in Fig. 8), corresponds to a standard mean-squared-error between the target data and the prediction of the PINN for the N_m space-time locations $\{(\boldsymbol{x}_m^n, t_m^n)\}$ where data is available. It should be again emphasised that only a subset of all the flow states are considered measurable and therefore ϕ_m does not account for the full flow state. The second term (ϵ_p, red part in Fig. 8) represents a physics-based loss which is the residual of the reacting Navier-Stokes equations computed using the prediction of the PINN at arbitrary space-time collocation points $\{(\boldsymbol{x}_c^i, t_c^i)\}$. This second term enables the network to identify suitable predictions for the unmeasured quantities that satisfy the governing equations. This residual is computed using automatic differentiation [24] as in past work on PINNs [11, 25]. For simplicity, in the present work, the collocation points are taken to be the same space-time locations as where target data is available, i.e. $N_c = N_m = 78000$ and $\{(\boldsymbol{x}_c^i, t_c^i)\} = \{(\boldsymbol{x}_m^i, t_m^i)\}$.

In this work, similarly to an earlier work on HFM [11], a feedforward neural network of 20 hidden layers with 300 neurons each will be used. The training is performed in two stages: (i) the network is pre-trained using only the available data, i.e., the loss function only contains the first term in

Equation 1 which allows for a rapid partial weight optimisation as it corresponds to a traditional supervised training process with a mean-squared error (MSE) loss; (ii) all the network weights are optimised using the full loss function, as in Equation 1 enabling the reconstruction of the unmeasured quantities. All the training processes are performed with the ADAM optimiser [26] using a learning rate of 0.001 with a batch size of 10 000. The training is run until the loss function reaches a plateau indicating a fully trained network.

3.3. Results

In this section, we demonstrate the ability of the PINN to reconstruct the velocity field of the puffing flame from measurements of temperature, density and pressure. To do this, the PINN is trained as discussed in Sec. 3.2 using the dataset obtained from the simulations. It should be emphasised that, in the training dataset, only the temperature, density and pressure fields are provided to the network and that it never receives any velocity information. The network then infers the two components of the velocity fields using the residual of the reacting Navier-Stokes equations.

The reconstructed velocity field for a representative snapshot is shown in Figure 9a for u and Figure 9b for v. As can be seen in those figures, the two components of the velocity field are accurately reconstructed by the PINN and only minor differences can be observed at the base of the flame where strong gradients are present. This shows that the PINN manages to infer the dynamics of the buoyant plume from the residual of the Navier-Stokes equations *without* any observation of the velocity field. Additionally, the overall L^2-error remains small over the majority of the domain and most features of the velocity fields are accurately reconstructed. While this is shown for a specific time instant, the reconstruction accuracy was similar for most snapshots and the mean squared error (averaged over the computational domain) in function of time is shown in Fig. 9c. It can be seen that the error is overall low except at the initial time ($t = 0$ s). This higher error for that initial time instant is related to the relative lack of measurements data (in time) that prevents an appropriate estimation of the time-derivative in the residuals [11].
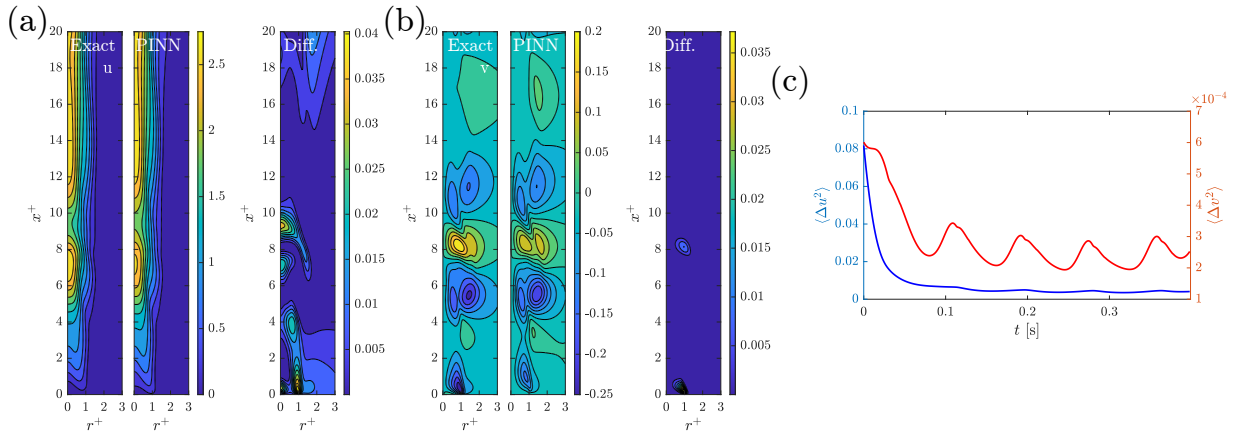

Figure 9: Comparison between actual and reconstructed (a) u and (b) v velocity for a representative snapshot. Superscript + indicates coordinates normalised by a. (c) Evolution of MSE of the reconstruction of u and v.

4. CONCLUSIONS

This work presented two different deep learning based methods for the reconstruction of unmeasured quantities in reacting flows. The first one is based on CNNs in a U-net architecture and was aimed at reconstructing the HRR fluctuation in an acoustically excited flame from past velocity information. The U-net was trained with the HRR dataset for some specific frequencies

and amplitudes of excitation and could accurately reconstruct the HRR for other frequencies and amplitudes not in that training dataset. The second method did not require any prior training data of the quantities to be reconstructed and is based on the physics-informed neural network architecture. It was shown able to reconstruct accurately the velocity field from the pressure, density and temperature field in an intrinsically unsteady puffing flame. These two techniques should be viewed as complementary of each other as in the first, no physical considerations are necessary while in the latter no information on the quantities to be reconstructed is required. Indeed, there can be cases where obtaining/implementing the relevant physical information (such as the complex chemical reactions) may be difficult, making the second PINN-based approach more difficult while other reconstruction tasks may not have any data on the quantities to be reconstructed making the first U-net approach unusable. Nonetheless, these results demonstrate the potential of deep learning techniques in supplementing information when only partial data is available.

In future work, these techniques are extended to three-dimensional turbulent flames and will be tested using experimental data.

ACKNOWLEDGEMENTS

REFERENCES

[1] C. Tropea, J. F. Foss, and A. Yarin, editors. *Springer Handbook of Experimental Fluid Mechanics*. Springer, Berlin, 2007.

[2] S. Scharnowski and C. K. Kahler. Particle image velocimetry - Classical operating rules from today's perspective. *Opt. Lasers Eng.*, 135:106185, 2020.

[3] R. K. Adrian. Scattering particle characteristics and their effect on pulsed laser measurements of fluid flow: speckle velocimetry vs particle image velocimetry. *Appl. Opt.*, 23:1690–1691, 1984.

[4] A. A. Adamczyk and L. Rimai. 2-Dimensional particle tracking velocimetry (PTV): Technique and image processing algorithms. *Exp. Fluids*, 6:373–380, 1988.

[5] Shivam Barwey, Malik Hassanaly, Venkat Raman, and Adam Steinberg. Using machine learning to construct velocity fields from OH-PLIF images. *Combust. Sci. Technol.*, page available online, 2019.

[6] Shivam Barwey, Venkat Raman, and Adam Steinberg. Extracting information overlap in simultaneous OH-PLIF and PIV fields with neural networks. *Proc. Combust. Inst.*, 38(4):6241–6249, 2021.

[7] Kai Fukami, Koji Fukagata, and Kunihiko Taira. Machine-learning-based spatio-temporal super resolution reconstruction of turbulent flows. *J. Fluid Mech.*, 909:A9, 2021.

[8] Jared L. Callaham, Kazuki Maeda, and S. L. Brunton. Robust flow reconstruction from limited measurements via sparse representation. *Phys. Rev. Fluids*, 4(10):1–31, 2019.

[9] N. Benjamin Erichson, Lionel Mathelin, Zhewei Yao, S. L. Brunton, Michael W. Mahoney, and J. Nathan Kutz. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proc. R. Soc. A Math. Phys. Eng. Sci.*, 476(2238):20200097, 2020.

[10] Douglas W. Carter, Francis De Voogt, Renan Soares, and Bharathram Ganapathisubramani. Data-driven sparse reconstruction of flow over a stalled aerofoil using experimental data. *Data-Centric Eng.*, 2:e5, 2021.

[11] M. Raissi, Alireza Yazdani, and George Em Karniadakis. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science*, 367(6481):1026–1030, 2020.

[12] Nirmal J. Nair and Andres Goza. Leveraging reduced-order models for state estimation using deep learning. *J. Fluid Mech.*, 897:R1, 2020.

[13] Pankaj Saini, Christoph M. Arndt, and Adam M. Steinberg. Development and evaluation of gappy-POD as a data reconstruction technique for noisy PIV measurements in gas turbine combustors. *Exp. Fluids*, 57(7):1–15, 2016.

[14] VN Kornilov, R Rook, JHM ten Thije Boonkkamp, and LPH De Goey. Experimental and numerical investigation of the acoustic response of multi-slit bunsen burners. *Combustion and Flame*, 156(10):1957–1970, 2009.

[15] S Jaensch, M Merk, EA Gopalakrishnan, S Bomberg, T Emmert, RI Sujith, and W Polifke. Hybrid CFD/low-order modeling of nonlinear thermoacoustic oscillations. *Proceedings of the Combustion Institute*, 36(3):3827–3834, 2017.

[16] N. Tathawadekar, N. A. K. Doan, C. F. Silva, and N. Thuerey. Modeling of the nonlinear flame response of a Bunsen-type flame via multi-layer perceptron. *Proc. Combust. Inst.*, 38(4):6261–6269, 2021.

[17] Stefan Jaensch and Wolfgang Polifke. Uncertainty encountered when modelling self-excited thermoacoustic oscillations with artificial neural networks. *International Journal of Spray and Combustion Dynamics*, 9(4):367–379, 2017.

[18] Laurent Selle, Ghislain Lartigue, Thierry Poinsot, R Koch, K-U Schildmacher, W Krebs, B Prade, P Kaufmann, and Denis Veynante. Compressible large eddy simulation of turbulent combustion in complex geometry on unstructured meshes. *Combustion and Flame*, 137(4):489–505, 2004.

[19] The OpenFOAM Foundation. OpenFOAM-7. https://openfoam.org/, 2019.

[20] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(4):640–651, 2017.

[21] Daniel Moreno-Boza, Wilfried Coenen, Jaime Carpio, Antonio L. Sánchez, and Forman A. Williams. On the critical conditions for pool-fire puffing. *Combust. Flame*, 192:426–438, 2018.

[22] Eduardo Fernández-Tarrazo, Antonio L. Sánchez, Amable Liñán, and Forman A. Williams. A simple one-step chemistry model for partially premixed hydrocarbon combustion. *Combust. Flame*, 147(1):32–38, 2006.

[23] M. Raissi, Zhicheng Wang, Michael S. Triantafyllou, and G.E. Karniadakis. Deep learning of vortex-induced vibrations. *J. Fluid Mech.*, 861:119–137, 2019.

[24] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in PyTorch. In *NIPS 2017 Work. Autodiff*, pages 1–4, 2017.

[25] M. Raissi, P. Perdikaris, and G.E. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.*, 378:686–707, 2019.

[26] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, pages 1–15, 2015.