

## Velocity reconstruction in puffing pool fires with physics-informed neural networks

Sitte, Michael Philip; Doan, Nguyen Anh Khoa

**DOI**

[10.1063/5.0097496](https://doi.org/10.1063/5.0097496)

**Publication date**

2022

**Document Version**

Final published version

**Published in**

Physics of Fluids

**Citation (APA)**

Sitte, M. P., & Doan, N. A. K. (2022). Velocity reconstruction in puffing pool fires with physics-informed neural networks. *Physics of Fluids*, 34(8), Article 087124. <https://doi.org/10.1063/5.0097496>

**Important note**

To cite this publication, please use the final published version (if applicable).  
Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights.  
We will remove access to the work immediately and investigate your claim.

# Velocity reconstruction in puffing pool fires with physics-informed neural networks

Cite as: Phys. Fluids **34**, 087124 (2022); <https://doi.org/10.1063/5.0097496>

Submitted: 29 April 2022 • Accepted: 24 July 2022 • Published Online: 22 August 2022

Published open access through an agreement with Technische Universiteit Delft Faculteit Luchtvaart- en Ruimtevaarttechniek

 Michael Philip Sitte and  Nguyen Anh Khoa Doan

## COLLECTIONS

Paper published as part of the special topic on [Artificial Intelligence in Fluid Mechanics](#)



View Online



Export Citation



CrossMark

## ARTICLES YOU MAY BE INTERESTED IN

[Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations](#)  
Phys. Fluids **34**, 075117 (2022); <https://doi.org/10.1063/5.0095270>

[On Oreology, the fracture and flow of “milk's favorite cookie®”](#)  
Phys. Fluids **34**, 043107 (2022); <https://doi.org/10.1063/5.0085362>

[Dense velocity reconstruction from particle image velocimetry/particle tracking velocimetry using a physics-informed neural network](#)  
Phys. Fluids **34**, 017116 (2022); <https://doi.org/10.1063/5.0078143>

Physics of Fluids

Special Topic: Hydrogen Flame and Detonation Physics

Submit Today!



# Velocity reconstruction in puffing pool fires with physics-informed neural networks

Cite as: Phys. Fluids **34**, 087124 (2022); doi: 10.1063/5.0097496

Submitted: 29 April 2022 · Accepted: 24 July 2022 ·

Published Online: 22 August 2022





View Online



Export Citation



CrossMark

Michael Philip Sitte<sup>1</sup>  and Nguyen Anh Khoa Doan<sup>2,a)</sup> 

## AFFILIATIONS

<sup>1</sup>Siemens Mobility Austria GmbH, Leberstrasse 34, 1110 Vienna, Austria

<sup>2</sup>Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, 2629 HS Delft, The Netherlands

Note: This paper is part of the special topic, Artificial Intelligence in Fluid Mechanics.

<sup>a)</sup> Author to whom correspondence should be addressed: [n.a.k.doan@tudelft.nl](mailto:n.a.k.doan@tudelft.nl)

## ABSTRACT

Pool fires are canonical representations of many accidental fires which can exhibit an unstable unsteady behavior, known as puffing, which involves a strong coupling between the temperature and velocity fields. Despite their practical relevance to fire research, their experimental study can be limited due to the complexity of measuring relevant quantities in parallel. In this work, we analyze the use of a recent physics-informed machine learning approach, called hidden fluid mechanics (HFM), to reconstruct unmeasured quantities in a puffing pool fire from measured quantities. The HFM framework relies on a physics-informed neural network (PINN) for this task. A PINN is a neural network that uses both the available data, here the measured quantities, and the physical equations governing the system, here the reacting Navier–Stokes equations, to infer the full fluid dynamic state. This framework is used to infer the velocity field in a puffing pool fire from measurements of density, pressure, and temperature. In this work, the dataset used for this test was generated from numerical simulations. It is shown that the PINN is able to reconstruct the velocity field accurately and to infer most features of the velocity field. In addition, it is shown that the reconstruction accuracy is robust with respect to noisy data, and a reduction in the number of measured quantities is explored and discussed. This study opens up the possibility of using PINNs for the reconstruction of unmeasured quantities from measured ones, providing the potential groundwork for their use in experiments for fire research.

© 2022 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). <https://doi.org/10.1063/5.0097496>

## I. INTRODUCTION

In fire research, pool fires are characterized by their stabilization on a horizontal surface, solid or liquid, where the fuel is supplied through evaporation or pyrolysis, which provides a feedback between heat release, heat transfer, and fuel supply. In contrast to (jet) flames, fires are further characterized by a low initial Reynolds number of the fuel stream from evaporation or pyrolysis, a buoyant turbulent diffusion flame, and a non-reacting buoyant plume.<sup>1</sup> The study of pool fires is relevant from both a theoretical and a practical point of view as they constitute a canonical case for fire research that includes most key physical phenomena essential to the dynamics of natural or accidental fires. Consequently, the study of pool fires is of considerable interest in fire safety research, where they have been commonly used, for instance, as a model fire to study the suppression of fires<sup>2</sup> or as ignition sources to study the spread of fires in rail vehicles.<sup>3,4</sup> The case of liquid sodium pool fires is relevant to the safety of sodium cooled nuclear reactors.<sup>5</sup> Finally, a similar behavior to the puffing of pool fires has also been found to play an important role in the spreading of wildfires.<sup>6,7</sup>

Pool fires have been studied in depth, starting from some early investigations,<sup>8–10</sup> and several extensive reviews are available on the topic.<sup>1,11–13</sup> Here, we focus on an unsteady pulsating motion with a well-defined frequency that pool fires can exhibit called “puffing,” which has been shown to be the result of a fluid dynamic rather than a combustion instability.<sup>14</sup> In fact, at the onset of puffing, the pool fire undergoes a bifurcation to a globally unstable puffing state driven by baroclinic and buoyant vorticity generation.<sup>15</sup> In a recent work, Moreno-Boza *et al.*<sup>15</sup> performed a global linear stability analysis of 2D axisymmetric pool fires to determine the critical condition for the onset of puffing, which was found to occur when exceeding a critical Rayleigh number. While the above-mentioned onset of puffing occurs in the laminar regime, recent progress on the modeling of turbulent pool fires include several works using large eddy simulation for predictive modeling.<sup>16–18</sup>

Despite their importance described above, the experimental study of pool fires has been somewhat limited by the difficulties that arise from the complexity of performing measurements on them, as such

techniques are often restricted in either the regions that can be investigated or the quantities that can be measured. This becomes even more challenging for transient fire dynamics, such as puffing fires, as they would require simultaneous diagnostics. In this context, the development of data analysis and postprocessing techniques that can supplement the missing information by, for example, reconstructing unmeasured quantities or augmenting the resolution of the available data is of great interest to fire research. To achieve this, recent techniques in data science and machine learning have shown a great potential.<sup>19–27</sup> 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<sup>22</sup> or by using feedforward neural networks to achieve a similar task.<sup>23,26</sup> 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.<sup>27</sup> Further attempts were made at using convolutional neural networks (CNNs) to reconstruct velocity fields from OH-planar laser induced fluorescence (PLIF) data.<sup>19,20</sup> 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.*,<sup>25</sup> 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. Such a network was coined a “physics-informed neural network” (PINN) and was originally designed to solve ordinary differential equation (ODE) or partial differential equation (PDE) problems<sup>41</sup> such as the Navier–Stokes equation<sup>28</sup> or its Reynolds-averaged version.<sup>29</sup> In the context of experimental measurements reconstruction, as a proof of concept, Raissi *et al.*<sup>25</sup> applied a PINN to the Kármán vortex street in the wake of a cylinder. This problem presented itself as the prediction of a periodic laminar flow, i.e., velocity and pressure fields, from a conserved scalar variable. The so-called physics-informed neural network was shown able to reconstruct the velocity field in this canonical non-reacting flow. Cai *et al.*<sup>30</sup> have applied HFM to a buoyancy-driven flow using experimental data as input. They inferred the velocity and pressure fields above an espresso cup from snapshots of the temperature field recorded with tomographic background-oriented Schlieren imaging. However, the result was only validated qualitatively since velocity measurements were not carried out simultaneously with the temperature measurements. A similar PINN-based framework was also applied to the problem of super-resolution, i.e., inferring higher resolution flow features from low resolution ones in space and time.<sup>31,32</sup> It was shown that the PINN could reconstruct the higher resolution flow characteristics in space and time for a series of canonical non-reacting flows both from simulation data<sup>31</sup> or from experimental data.<sup>32</sup> Therefore, the HFM approach has only been applied to flows of constant density or small density variations that can be described by a Boussinesq approximation. The applicability of this approach has not been fully demonstrated for reacting flows with large variations and steep gradients of density and temperature as well as significant variations of composition.

In this work, we propose to extend this HFM methodology of Raissi *et al.*<sup>25</sup> to the reconstruction of the velocity field in a puffing pool fire, thereby demonstrating the potential of this method in fire

research, which deals with complex reacting flows that include chemical reactions, large density variations, and buoyancy. This work opens up the use of HFM to practical fire-related applications where obtaining a full set of measurements is extremely challenging. In this first attempt to apply HFM to a reacting flow proposed here, the dataset of measured quantities will be obtained from numerical simulations as this provides all quantities of interest from which only a subset will be used for the development of the HFM framework and as it allows for a thorough cross-comparison between the reconstruction from the HFM framework and the values obtained from the simulations. Nevertheless, these simulations are set up as numerical representations of real flames studied experimentally and will be shown to reproduce their behavior accurately.<sup>15</sup> The basic reconstruction problem in scope is the reconstruction of the full velocity field from measurements of density, pressure, and temperature.

The remainder of this paper is organized as follows. Section II A presents the setup of the numerical simulations of the pool fire. Section II B provides details on the HFM framework. Numerical results of the pool fire simulations are provided in Sec. III A, and the velocity reconstruction with HFM is discussed in Sec. III B. Additional comments on the robustness of the HFM framework with respect to noisy data and a reduced number of measured quantities are also discussed. Section IV summarizes the results obtained and provides directions for future work.

## II. METHODS

### A. Computational fluid dynamics

Simulations of the pool fire were carried out using the computational fluid dynamics (CFD) toolbox OpenFOAM-7 with the solver *fireFoam*.<sup>33</sup> The governing equations are the continuity equation, Navier–Stokes equations, species mass fraction equations (for each chemical specie  $k$ ), and enthalpy equation,<sup>34</sup>

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0, \quad (1)$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} + \rho g_i, \quad (2)$$

$$\frac{\partial \rho Y_k}{\partial t} + \frac{\partial \rho u_i Y_k}{\partial x_i} = \frac{\partial}{\partial x_i} \left[ \rho D \frac{\partial Y_k}{\partial x_i} \right] + \rho \dot{\omega}_k, \quad (3)$$

$$\frac{\partial \rho h}{\partial t} + \frac{\partial \rho u_i h}{\partial x_i} = \frac{\partial}{\partial x_i} \left[ \rho \alpha \frac{\partial h}{\partial x_i} \right], \quad (4)$$

where  $\tau_{ij}$  is the molecular stress tensor,  $\dot{\omega}_k$  is the chemical reaction rate,  $D$  is the molecular diffusivity, and  $\alpha = \lambda/(\rho C_p)$  is the thermal diffusivity.

A 2D axisymmetric pool fire with n-heptane fuel ( $Y_F = Y_{C_7H_{16}}$ ) and at ambient conditions ( $p_0 = 100\,000$  Pa,  $T_0 = 300$  K,  $Y_{O_2} = 0.233$ ,  $Y_{N_2} = 0.767$ ) was investigated, equivalent to the n-heptane flames with an isothermal brass baseplate studied experimentally and numerically by Moreno-Boza *et al.*<sup>15</sup> A schematic of the numerical domain is shown in Fig. 1. 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  $600 \times 130$  grid points. In radial direction, the pool radius  $a$  was resolved by 91 grid points. In the region of interest, where the flame was located, the same resolution was used in axial direction. So, depending on the pool diameter, the grid resolution was in the



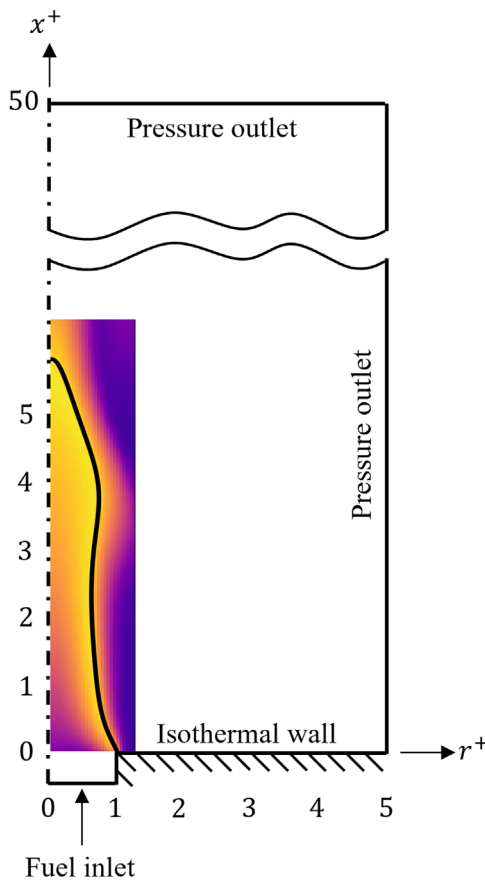
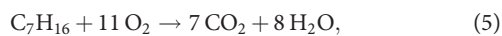


FIG. 1. Schematic of the numerical domain. Coordinates normalized by pool radius  $a$ .

range from approximately 0.088–0.11 mm, which is sufficient to resolve all scales of the flow and the reaction zone. This resolution was the result of a grid sensitivity study, where the resolution was refined until key characteristics of the flame, i.e., stable flame length and puffing frequency in the transient case, did not change any more and grid convergence was achieved. 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 with adaptive step size for the chemical source term. This was found to be sufficiently accurate to replicate the characteristics of the flames observed in the experiments and effective to avoid numerical instabilities.

In the present modeling approach, radiation was not explicitly included as a sink term in the enthalpy equation. Instead, radiative heat loss was accounted for by reducing the heat of combustion of the chemical reaction. Turbulence modeling did not need to be considered, since the flame is in the laminar regime at the onset of puffing. Combustion chemistry was modeled as the irreversible single-step reaction,



whose rate constant is given by the Arrhenius law,

$$K = BT^\beta \exp(-T_A/T) \quad (6)$$

with the model constants  $\beta = 0$ ,  $T_A = 12000$  K and  $B = 5.5 \times 10^7 \text{ m}^3/(\text{mol s})$ . The choice of  $T_A$  and  $B$  was made to best fit the laminar flame speed curve for heptane–air mixture with the single-step mechanism, following the reasoning of Fernandez-Tarrazo *et al.*,<sup>35</sup> but without correcting heat release rate (HRR) and  $T_A$  with equivalence ratio  $\phi$ . This modeling of the chemical reaction was chosen for the sake of simplicity, and it is sufficient for the present case of a laminar diffusion flame characterized by high Damköhler number and controlled by mixing. Indeed, Moreno-Boza *et al.*<sup>15</sup> reproduced the puffing behavior correctly using infinitely fast chemistry. Reproduction of the laminar flame speed  $S_L(\phi)$  and the HRR( $\phi$ ) over a broad range of equivalence ratios, as suggested by Fernandez-Tarrazo *et al.*,<sup>35</sup> is less important for a diffusion flame where the reaction occurs at stoichiometric conditions.

Density is given by the ideal gas law. Viscosity is computed from Sutherland's law,  $\mu = A_s \sqrt{T}/(1 + T_s/T)$ , independent of species composition with  $A_s = 1.672 \times 10^{-6} \text{ kg}/(\text{m s K}^{1/2})$  and  $T_s = 170.67$  K. 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,  $C_p$ , and enthalpy,  $h$ , 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.<sup>15</sup>

The liquid surface is modeled as a boundary condition following Moreno-Boza *et al.*<sup>15</sup> 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  $u_n$ ,

$$\rho u_n = \frac{1}{l_V} \lambda \frac{\partial T}{\partial n}, \quad (7)$$

where  $l_V = L_V + C_L(T_B - T_0)$  with the latent heat  $L_V = 360000 \text{ J}/\text{kg}$  and the liquid heat capacity  $C_L = 2240 \text{ J}/(\text{kg K})$ . The jump condition at the phase interface for fuel mass fraction (species index  $k = F$ ) is

$$\rho u_n = \rho u_n Y_F - \rho D \frac{\partial Y_F}{\partial n}, \quad (8)$$

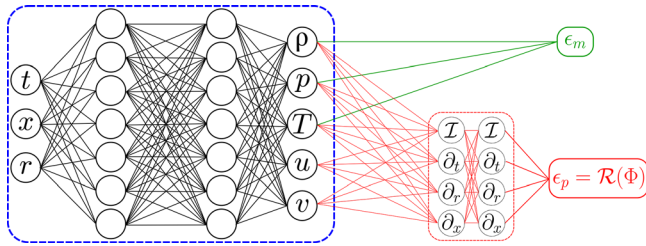
and for all other species ( $k \neq F$ ),

$$0 = \rho u_n Y_k - \rho D \frac{\partial Y_k}{\partial n}. \quad (9)$$

In the experiments, the fuel surface was maintained  $0.75 \pm 0.25$  mm below the surface of the brass baseplate to avoid fuel spillage and the flame behavior was found to be very sensitive to that value.<sup>15</sup> Therefore, this vertical offset between the isothermal baseplate, and the liquid surface is included in the present simulations.

## B. Reconstruction method

To reconstruct the unmeasured quantity, the approach called hidden fluid mechanics proposed by Raissi *et al.*<sup>25</sup> will be used. This approach relies on a physics-informed neural network,<sup>36</sup> illustrated in Fig. 2, 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, listed in Sec. II A. To achieve the inference of



**FIG. 2.** Schematic of the HFM framework. The dashed blue box indicates the actual neural network. The dashed red box indicates the differential operators ( $\mathcal{I}$  indicates the identity operation,  $\partial$  is the partial derivative), obtained via automatic differentiation, required to compute the physical residual,  $\epsilon_p$ . The green box indicates the calculation of the measurement reconstruction error,  $\epsilon_m$ .

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, described in Sec. II A. A PINN is a conventional feedforward neural network (blue box in Fig. 2) which is trained with a specific loss function that accounts for the governing physical equations, which we will call the “physical residual error”  $\epsilon_p$  (red part in Fig. 2) and the measurement reconstruction error ( $\epsilon_m$ , green part). Feedforward neural networks map the input to the output, as shown in Fig. 2 where an input, two hidden, and an output layer 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  $Z_l = W_l^T X_{l-1} + b_l$ , where  $X_{l-1}$  is the output of the layer  $l-1$  and  $b_l$  is the bias in layer  $l$ . Following this, nonlinearities are introduced through the element-wise activation function  $g$ , so that  $X_l = g(Z_l)$ . For what follows, the activation function used for all hidden layers will be the swish function<sup>37</sup> with a linear activation in the final output layer. The choice of the swish activation function was made following the propositions on HFM architecture in previous work.<sup>25</sup> The swish function was found to provide sufficient accuracy for the present cases as well, which will be discussed in Sec. III B. Other activation functions, such as the sigmoid and Rectified Linear Unit (ReLU) activation functions, were also tested but in those cases the accuracy of the resulting PINN was smaller, in line with previous findings.<sup>37</sup> The interested reader is referred to this latter reference for details on how the swish activation function outperforms previous activation function.

In the HFM architecture, the PINN takes as input a space–time location  $(x, t)$  (here, in axisymmetric 2D,  $\mathbf{x} = [x, r]$ ) and outputs the flow state vector at that location, i.e., its output is  $\Phi = [\rho, p, T, \mathbf{u}, \mathbf{Y}]$ , where  $\vec{\cdot}$  indicates a prediction from the PINN,  $\mathbf{u} = [u, v]$  is the velocity vector, with axial component  $u$  and radial component  $v$ , and  $\mathbf{Y}$  is the chemical composition vector. For simplicity, in this first attempt at using the HFM for reacting flows, the species mass fraction will not be considered in the reconstruction problem and, therefore, the considered flow state is just  $\tilde{\Phi} = [\rho, p, T, u, v]$ . 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  $\phi_m$  for the target data and  $\tilde{\phi}_m$  for the PINN prediction, with

their associated space–time locations noted as  $(\mathbf{x}_m, t_m)$ . In what follows, it will be considered that the measurable quantities are  $\phi_m = [\rho, p, T]$  unless mentioned otherwise. A visual representation of the quantities that are considered measured and the ones to be reconstructed is provided in Fig. 3.

To enable the PINN to reconstruct the other *unmeasured* quantities (here the velocity field  $\mathbf{u}$ ), 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  $(\mathbf{x}_c, t_c)$ , spread over the time–space domain covered by the simulation. It should be noted that given the specific choice made here in terms of the quantities taken into account (species not considered and  $T$  being measured), it becomes possible to only consider a subset of the complete reacting Navier–Stokes equations, namely, the continuity and momentum equations. Indeed, if  $T$  is measured, these equations would form a closed set of equations (for the density, pressure, and the velocity fields) and, therefore, the species transport equations and the enthalpy equation are not considered in what follows. Thus, the loss function to train the PINN is

$$\mathcal{L} = \underbrace{\frac{1}{N_m} \sum_{n=1}^{N_m} |\phi_m(\mathbf{x}_m^n, t_m^n) - \tilde{\phi}_m(\mathbf{x}_m^n, t_m^n)|^2}_{\epsilon_m} + \underbrace{\frac{1}{N_c} \sum_{i=1}^{N_c} |\mathcal{R}(\tilde{\Phi}(\mathbf{x}_c^i, t_c^i))|^2}_{\epsilon_p}. \quad (10)$$

In the equation above, the first term, the measurements reconstruction error  $\epsilon_m$  (green part in Fig. 2), corresponds to a standard mean-squared-error (MSE) between the target data (the measured fields) and the prediction of the PINN for the  $N_m$  space–time locations  $\{(\mathbf{x}_m^n, t_m^n)\}$  where those measurements data are available. It should again be emphasized that only a subset of all the flow states are considered measurable and therefore  $\phi_m$  does not account for the full flow state. The second term ( $\epsilon_p$ , red part in Fig. 2) 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  $\{(\mathbf{x}_c^i, t_c^i)\}$ . This second term enables the network to identify suitable predictions for the unmeasured quantities that satisfy the governing equations. The role of the collocation points is, therefore, quite important as these are the locations where the PINN will try to minimize the residual of the governing equations. This physical residual is computed using automatic differentiation<sup>38</sup> as in past work on PINN applications.<sup>25,28</sup> Automatic differentiation enables the efficient computation of the (partial) derivatives of the output (the flow states) with respect to the input (space and time). This allows to estimate all the required partial derivatives needed in the (reacting) Navier–Stokes equations i.e., Eqs. (1)–(4), therefore, enabling an estimation of the residual of those equations at the collocation locations. It should be noted here again that, given the specific choice of measured quantities and quantities to be reconstructed, it is only necessary to consider the continuity and momentum equations in the physical residual error calculations and, therefore, only Eqs. (1) and (2) are actually considered in the calculation of  $\epsilon_p$  in this work. Finally, for simplicity, the collocation points are taken to be the same as the space–time locations where target data are available, i.e.,  $N_c = N_m$  and  $\{(\mathbf{x}_c^i, t_c^i)\} = \{(\mathbf{x}_m^i, t_m^i)\}$ .

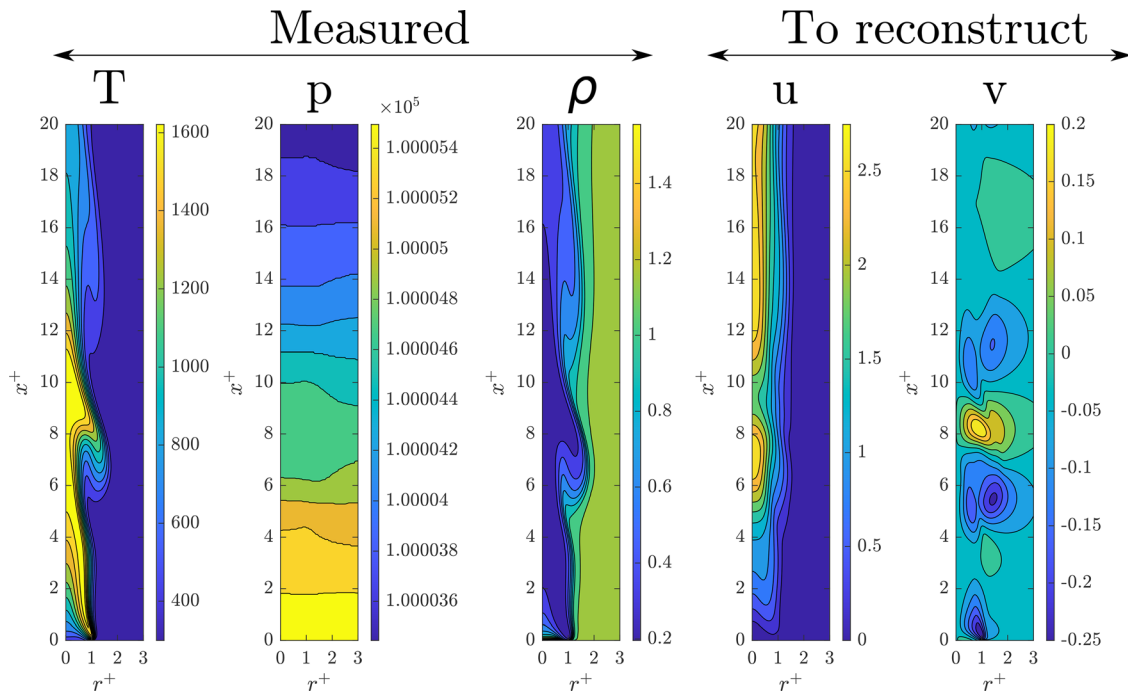


FIG. 3. Typical fields that are used for the reconstruction problem.  $T$ ,  $p$ ,  $\rho$  are considered as measurements while  $u$  and  $v$  are quantities to be reconstructed.

Regarding the boundary conditions, previous work on HFM<sup>25</sup> showed that an arbitrary domain could be considered for reconstruction. So, there was no need for an explicit knowledge of the boundary condition, and the measurement data (from the scalar) was sufficient for the reconstruction of the unmeasured quantities by the PINN. Similarly, in our work, we did not need to explicitly provide the boundary conditions to the PINN—the measured quantities were sufficient for the PINN to infer the unmeasured quantities. In general, other problems may exist where the boundary conditions would have to be explicitly specified (for example when trying to solve the direct PDE problem) to ensure the convergence of the PINN toward an appropriate solution.<sup>28,29</sup>

In the present work, following Raissi *et al.*,<sup>25</sup> a feedforward neural network of 20 hidden layers with 300 neurons each is used. The swish activation function is used for all neurons in all layers except in the last layer where the linear activation function is 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 Eq. (10), which allows for a rapid partial weight optimization as it corresponds to a traditional supervised training process with a mean-squared error (MSE) loss; (ii) all the network weights are optimized using the full loss function, as in Eq. (10) enabling the reconstruction of the unmeasured quantities. All the training processes are performed with the adaptive moment estimation (ADAM) optimizer<sup>39</sup> using a learning rate of 0.001 with a batch size of 10 000. During this training, the value of the loss function is monitored and the training is stopped when it has reached a plateau indicating a fully trained network. A typical evolution of the loss function is shown in Fig. 4 where the training process has converged after approximately 6000 epochs.

### III. RESULTS

#### A. Flame dynamics from CFD

The CFD simulations were validated against the experimental and numerical results for the n-heptane flame with isothermal walls from Moreno-Boza *et al.*,<sup>15</sup> notably the reproduction of the flame length, the critical point and the puffing frequency. Figure 5 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 (Bilger's definition of the mixture fraction<sup>40</sup> is used; the stoichiometric mixture fraction is  $\xi_{st} \approx 0.0622$ ). 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 the

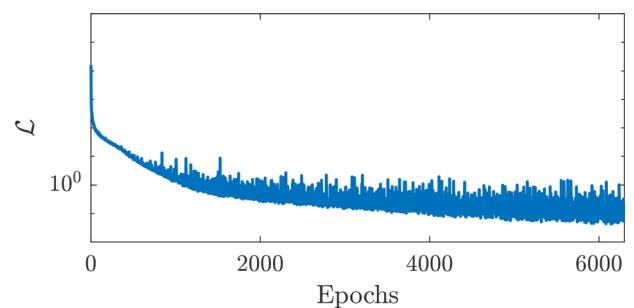
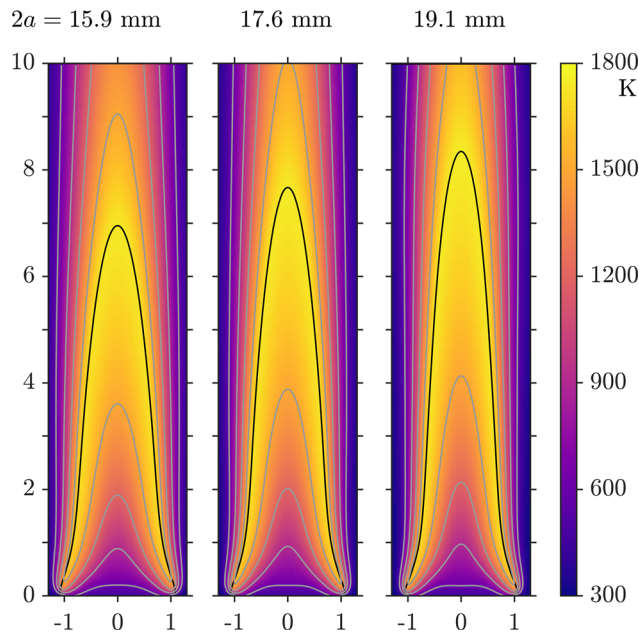


FIG. 4. Typical evolution of the loss function [Eq. (10)] during the training process.



**FIG. 5.** 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; gray lines: isocontours for  $T = 600, 900, 1200, 1500$  K.

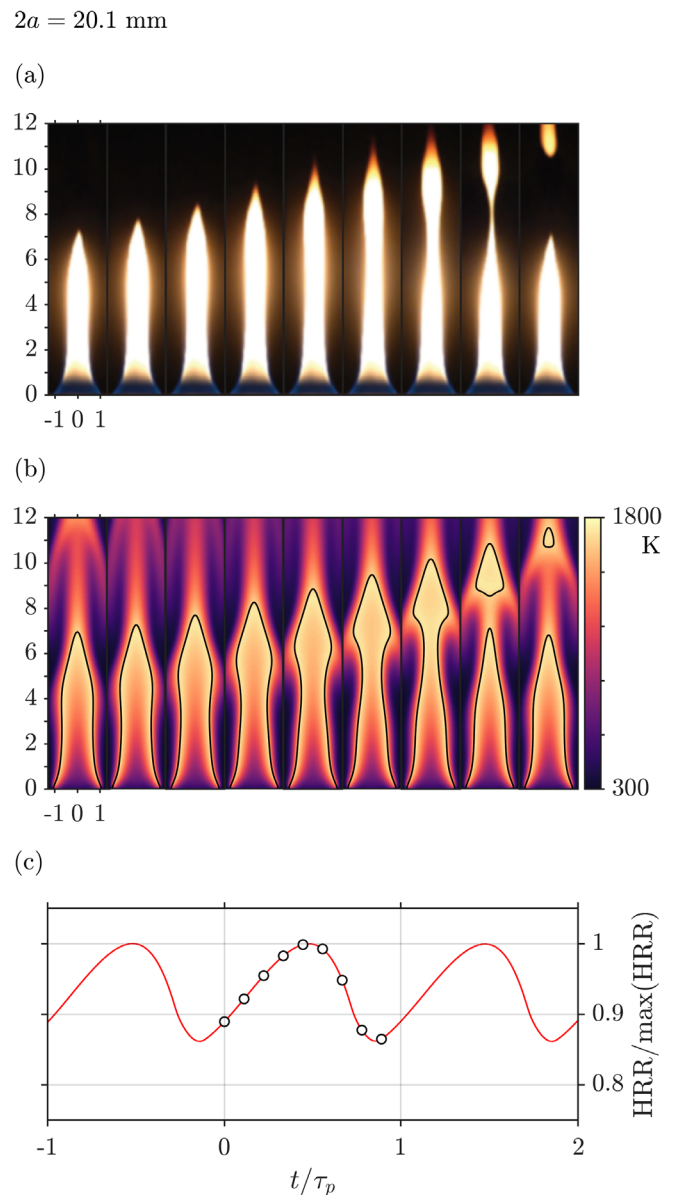
experimental findings for the isothermal baseplate. At this condition, the puffing establishes as a periodic process with a constant frequency of approximately 12.0 Hz, very close to the experimentally determined value of 12.8 Hz. Figure 6 shows one cycle of this puffing behavior; (a) a photo sequence of the puffing flame from the experiments by Moreno-Boza *et al.*<sup>15</sup> is compared to (b) a time sequence from the present CFD simulations; additionally, (c) the temporal variation of HRR during the puffing cycle are shown. In this sequence, we can see the dynamics of the puffing flame, where a “cusp” is formed due to buoyancy-driven vorticity generation. Even though photographic images and temperature fields are not fully equivalent, the numerical results closely resemble the time sequence of the flame recorded in the experiments. Therefore, the present simulations have been shown to accurately reproduce the behavior of the n-heptane pool fire in the vicinity of the critical point, studied by Moreno-Boza *et al.*<sup>15</sup>

Excellent agreement of flame length and puffing behavior suggests that the present highly resolved numerical simulations are a close representation of a real flame. Therefore, the comparison with experimental work is implicit in the present work. For the reconstruction problem discussed in Sec. III B, the puffing flame with  $2a = 20.1$  mm is considered. The dataset used to train the PINN consisted of 400 snapshots recorded at 1000 frames per second, corresponding to approximately five puffing periods with about 83 snapshots per period.

## B. Velocity reconstruction

### 1. Base case

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. The PINN was trained as discussed



**FIG. 6.** Puffing cycle of the flame with diameter  $2a = 20.1$  mm: (a) photo sequence from the experiments by Moreno-Boza *et al.*,<sup>15</sup> (b) time sequence from the present CFD simulations, and (c) the corresponding time evolution of HRR, normalized by period  $\tau_p$  and  $\max(\text{HRR})$ ; instances of snapshots are indicated as  $\circ$ ; snapshots are spaced by  $\tau_p/9$  to match the spacing of the photo sequence.

in Sec. II B using the dataset obtained from the simulations (and shown in Sec. III A). It should be emphasized 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 both components of the velocity fields using the residual of the reacting Navier–Stokes equations.

The reconstructed velocity field for a snapshot at time  $t = 0.2$  s (which is the mid-time of the database) is shown in Figs. 7 and 8 for



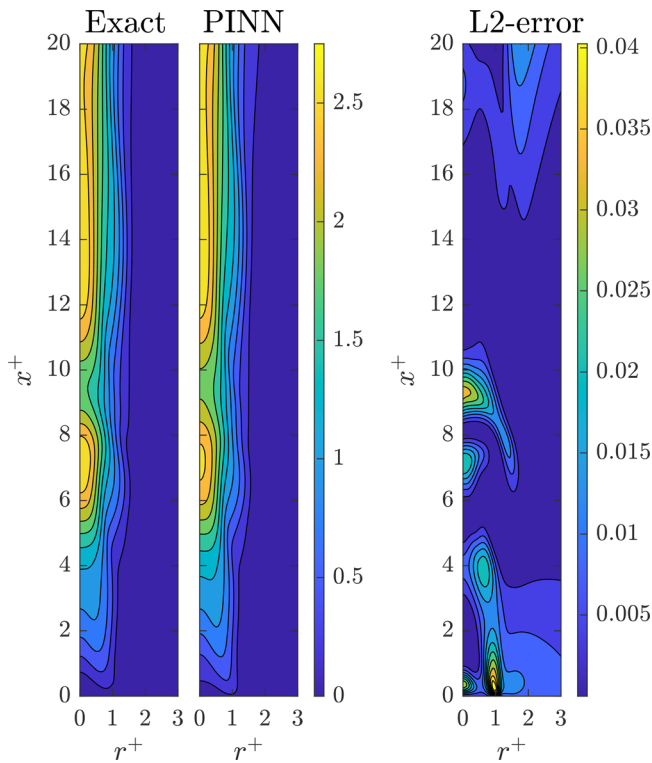


FIG. 7. Comparison between actual and reconstructed  $u$  velocity for the mid-time snapshot ( $t = 0.2$  s). Superscript + indicates coordinates normalized by  $a$ .

$u$  and  $v$ , respectively. 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. 9. It can be seen that the error is overall low with a higher value at the starting time ( $t = 0$  s).

We infer that this higher error for that initial time instant is related to the absence of data (in time) that prevents an appropriate estimation of the time-derivative in the residuals as was observed in previous work.<sup>25</sup> Here, it should be emphasized that the time between two snapshots was 1 ms (in contrast to the CFD time step of  $10^{-5}$  s) and, therefore, time-derivative quantities could not be inferred accurately from the measured fields, and even less so for the initial time instant. In contrast to previous work,<sup>25</sup> in our case the error at the final time instants remains lower. This may be due to the fact that the PINN has appropriately recognized the periodic nature of the quantities to be reconstructed while for the initial time instant, this apparent periodicity had not yet been “detected” by the PINN. However, despite the seemingly “large” error at the initial time (Fig. 9), the actually reconstructed velocity field at that

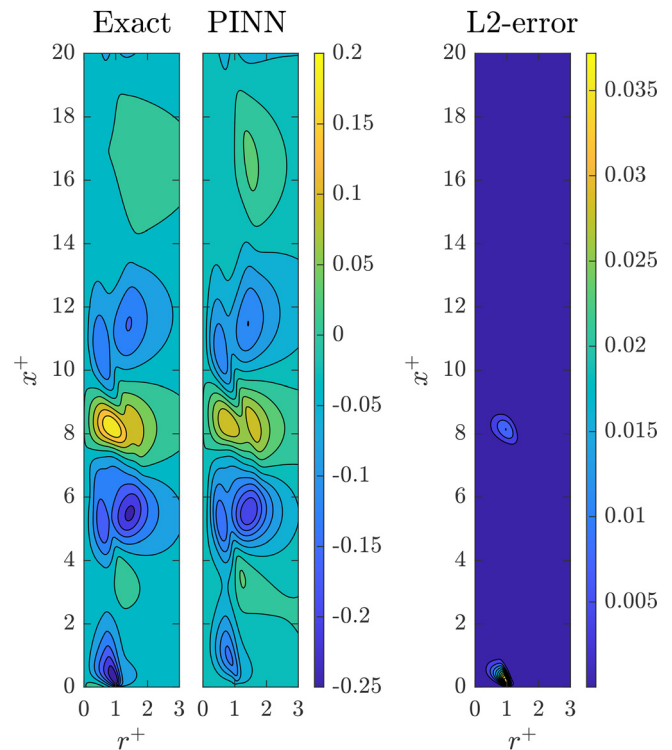


FIG. 8. Comparison between actual and reconstructed  $v$  velocity for the mid-time snapshot ( $t = 0.2$  s).

time instant, presented in Figs. 10 and 11 for the maximum error on  $u$  and  $v$ , respectively, shows that the PINN still manages to reconstruct the main features of the velocity fields. The PINN appears to underestimate the axial velocity of the buoyant plume in the downstream region, while the radial velocity is mispredicted near the flame anchoring point.

## 2. Effect of noise

An additional test of the proposed reconstruction method is performed by considering the effect of noise on the reconstruction

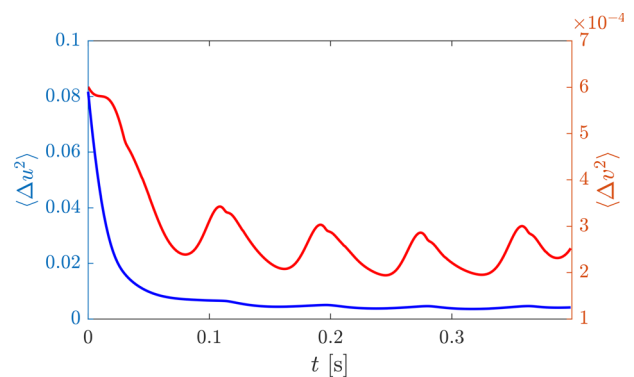


FIG. 9. Evolution of MSE of the reconstruction of  $u$  and  $v$ .

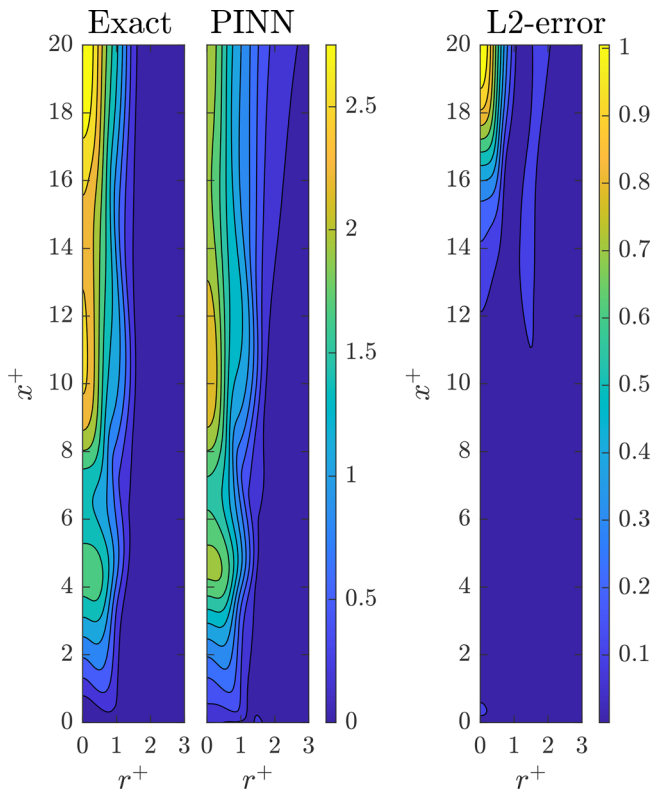


FIG. 10. Comparison between the exact and reconstructed  $u$  for the snapshot with maximal error ( $t = 0$  s).

accuracy. For this analysis, the originally noise-free dataset is corrupted artificially by adding noise on all measured quantities ( $\rho, p, T$ ) with a specified signal-to-noise ratio (SNR, based on each field average). Three levels of SNRs were considered: 20, 30, and 40 dB. Note that 20 dB refers

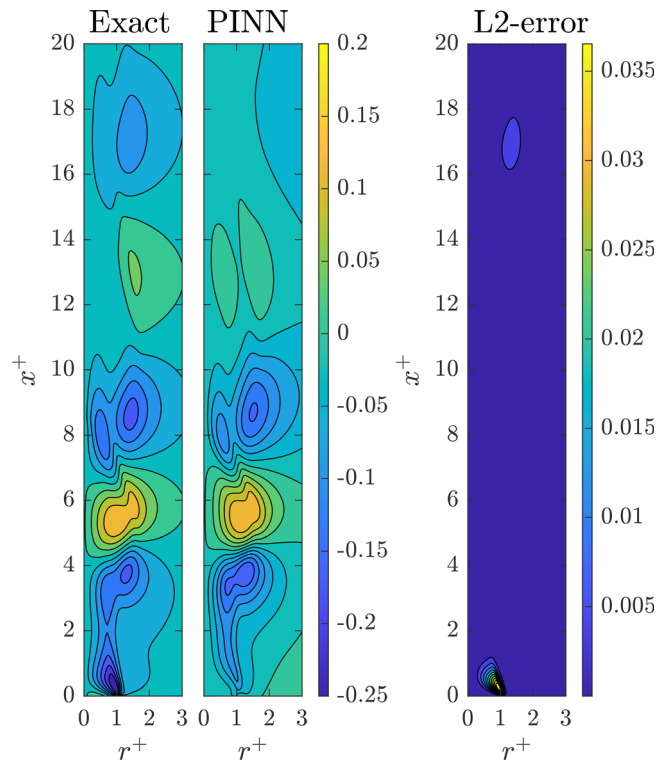


FIG. 11. Comparison between the exact and reconstructed  $v$  for the snapshot with maximal error ( $t = 0$  s).

to the highest noise level. A typical noisy dataset is shown in Fig. 12 for SNR = 20 dB. It can be seen that the data used for training by the PINN now exhibits noisy fluctuations. This makes the training of the PINN more complex as it has to identify how to smooth out the data to

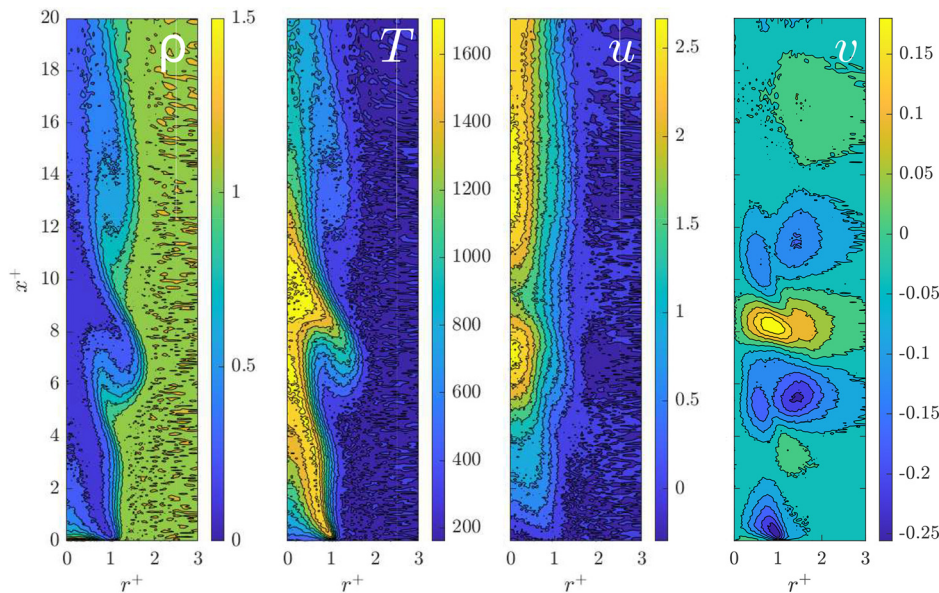


FIG. 12. Typical noisy data fields with SNR = 20 dB.



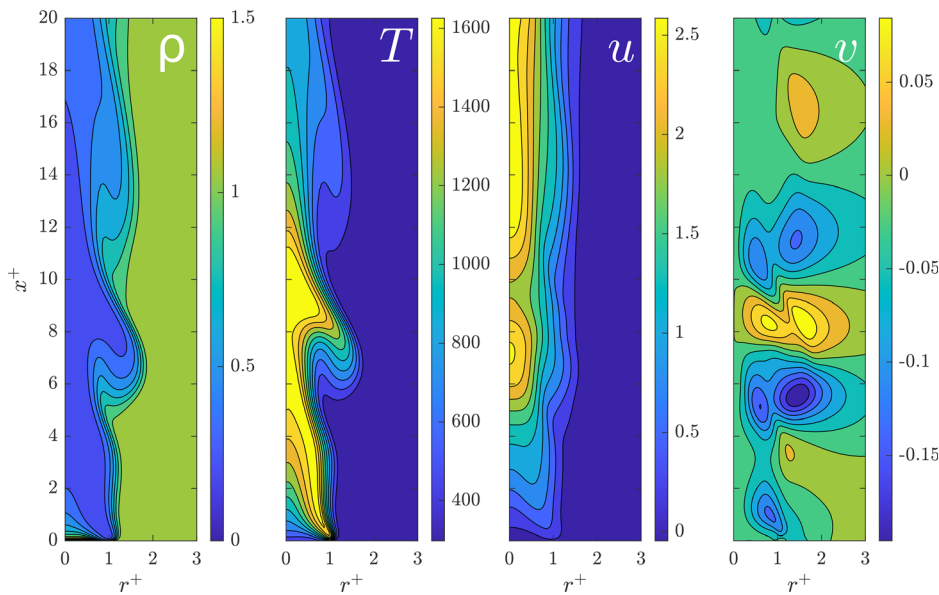


FIG. 13. Inferred density, temperature, and reconstructed velocity fields obtained from the PINN trained with noisy data with SNR = 20 dB.

minimize the physics-based residual in Eq. (10). For each noisy dataset, a PINN with the exact same hyperparameters as for the noise-free case is trained following the same procedure as earlier.

A typical reconstruction obtained from the trained PINN is shown in Fig. 13 for the same time instant as the one shown in Fig. 12. It can be seen that, for the measured quantities ( $\rho$  and  $T$ ), the inferred fields from the PINN do not exhibit the noisy behavior. This is achieved thanks to the physics-based loss in Eq. (10), which provides the means for the PINN to smooth out the inference of the measured quantities. Otherwise, fitting the data too closely, based uniquely on the mean-squared error, would lead to an excessively high physics-based loss due to the resulting spurious gradients. Instead, the trained PINN achieves a balance between fitting the noisy data and minimizing the residual of the governing equations. This illustrates that the PINN can also act as a physics-based “denoiser” of measurement data.

Regarding the reconstruction of the velocity field, it can be seen that the trained PINN reconstructed all features appropriately (compare velocity in Fig. 13 with Figs. 7 and 8), indicating that the

reconstruction from the PINN is robust with respect to noise. The resulting mean squared reconstruction error (computed with respect to the noise-free data) in function of time is shown in Fig. 14 for all considered noise levels. Overall, the level of reconstruction accuracy is

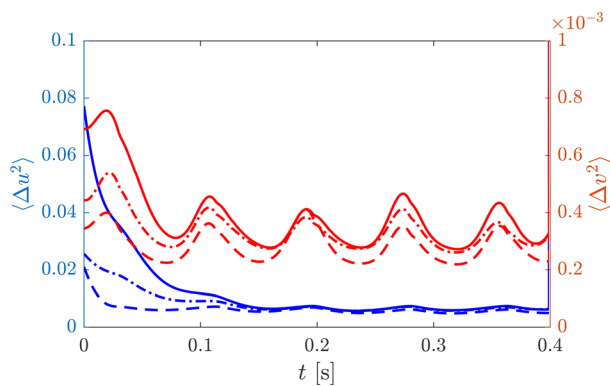


FIG. 14. Evolution of MSE of the reconstruction of  $u$  and  $v$  when using noisy datasets. (full line: SNR = 20 dB, dotted-dashed line: 30 dB, dashed line: 40 dB).

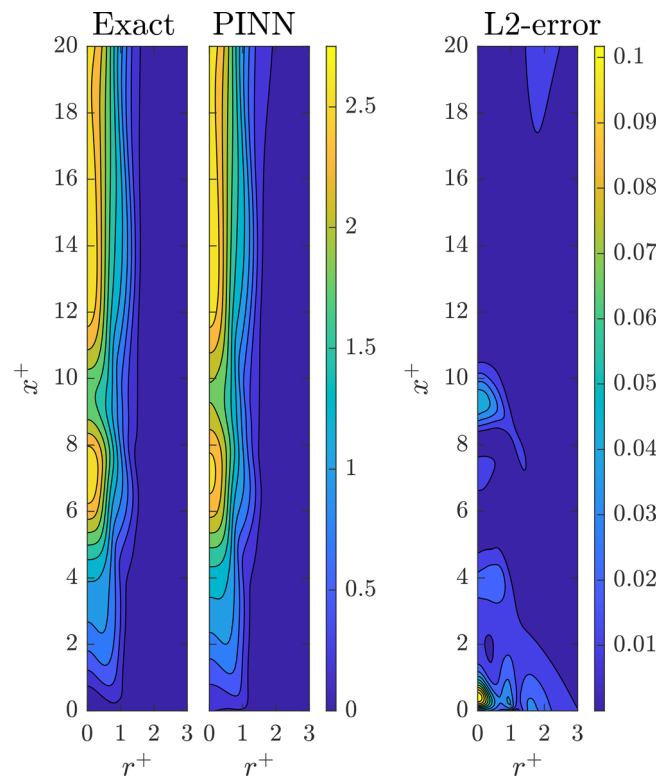


FIG. 15. Comparison between actual and reconstructed  $u$  velocity for the mid-time snapshot ( $t = 0.2$  s). Superscript + indicates coordinates normalized by  $a$ .

similar compared to noiseless data (shown in Fig. 9). In addition, as could be expected, the accuracy of the reconstruction decreases when larger noise levels are considered in the measured quantities. However, the overall accuracy remains within the range of what was obtained with the noiseless data indicating that the PINN is able to smooth out the measured data by using the equations and then use the relevant information to infer the appropriate reconstruction.

### 3. Effect of reduced measurements

In this section, we analyze the reconstruction ability of the HFM when fewer measured quantities are considered. First, we consider the case when only the pressure and temperature measurements are available. The HFM framework is kept exactly the same as in Sec. III B 1, and the only change is made by considering that measurements of density are unavailable. Therefore, the PINN has to reconstruct that quantity as well. Initially, this task was attempted without any assumption about the density, but the convergence of the PINN during training was relatively poor, since the reconstructed fields were not accurate enough. This was inferred to be due to the increased complexity of the optimization problem solved by the PINN, where different combinations of reconstructed quantities ( $\rho$ ,  $u$ , and  $v$ ) could allow for a small loss during training. To compensate for this, and to drive the PINN toward an appropriate solution, during the pre-training phase (when the PINN is trained purely on data), a “physics-based” guess of the target density is introduced to drive the PINN toward predicting this guess (i.e., this physics-based guess replaces the density

data during the pre-training). The physics-based guess for density is defined as follows:

$$\rho_{\text{guess}} = \frac{p_{\text{data}}}{r_{\text{air}} T_{\text{data}}}, \quad (11)$$

where the subscript *data* indicate that the value from the measured data is used and  $r_{\text{air}} = 287.05 \text{ J}/(\text{kg K})$  is the ideal gas constant of air, which is taken as a constant for the whole domain. This is evidently an approximation given the variations in composition (fuel and reaction products) occurring throughout the domain, but this allows to have an initial guess to drive the training of the PINN with this reduced set of information.

Following this pre-training, the PINN is then further trained by using the physical equations as described in Eq. (10). The resulting reconstruction of the velocity fields when using such an approach is shown in Figs. 15 and 16 for the  $u$  and  $v$  velocity, respectively, at a representative time instance. It can be seen that the reconstruction accuracy is very similar to the case with three measurements shown previously. There is, however, a larger error locally close to the flame base at the fuel interface.

This latter error can be better understood from Fig. 17, where the reconstructed density field for the same time instant is shown. It can be seen that, overall, the density is very accurately reconstructed except near the fuel inlet. This error originates from the approximation made when proposing an initial guess for the density based on Eq. (11).

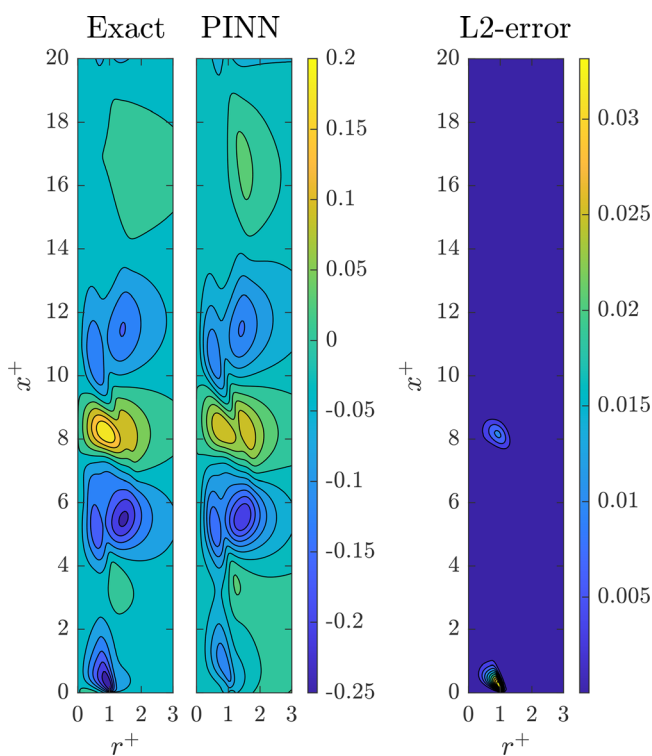


FIG. 16. Comparison between actual and reconstructed  $v$  velocity for the mid-time snapshot ( $t = 0.2 \text{ s}$ ). Superscript + indicates coordinates normalized by  $a$ .

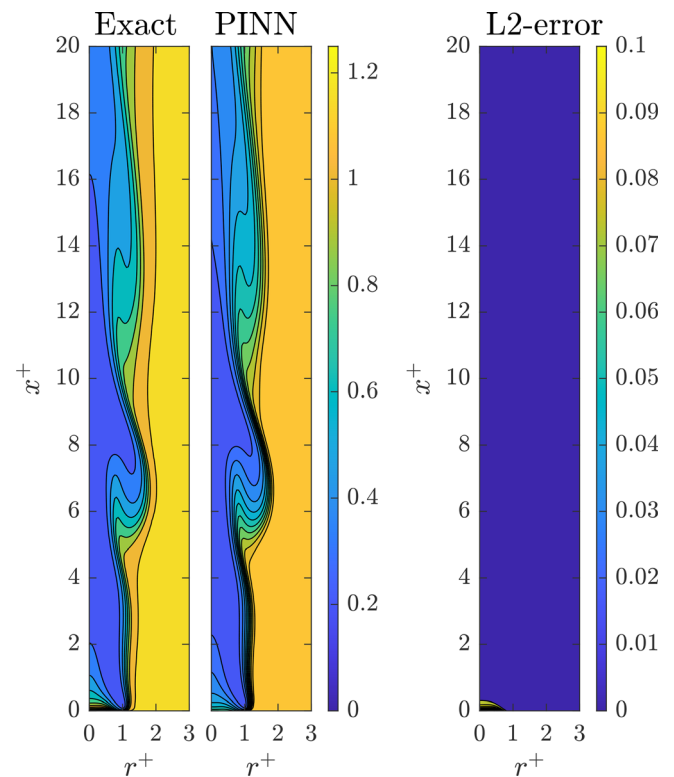


FIG. 17. Comparison between actual and reconstructed density  $\rho$  for the mid-time snapshot ( $t = 0.2 \text{ s}$ ). Superscript + indicates coordinates normalized by  $a$ .

Since the ideal gas constant of stoichiometric heptane–air reaction products,  $\approx 291 \text{ J}/(\text{kg K})$ , is very close to the ideal gas constant of air, this approximation is relatively accurate in all fuel-lean regions. In contrast, in the fuel-rich region near the fuel inlet, this assumption is most inaccurate, therefore leading to an inaccurate estimations of the density, which could not be fully compensated by the physics-based loss.

Nonetheless, despite this slightly inaccurate density reconstruction, the overall reconstruction accuracy of this two-measurement-based PINN remains of the same order of magnitude as for the three-measurement-based PINN in Sec. III B 1, as can be seen from the time-evolution of the average velocity reconstruction error in Fig. 18(a), when compared to Fig. 9. In both cases the error follows a similar evolution. In addition, Fig. 18(b) shows the time-evolution of the density reconstruction error, where it can be seen that the error remains more or less constant with fluctuations related to the puffing frequency.

Second, it is attempted to reconstruct the velocity, density and pressure fields solely from the measurements of temperature, to further analyze the reconstruction capability of the PINN. Here, temperature is chosen as it could be relatively straightforwardly measured in large-scale pool fires using infrared cameras, for example. Similarly as above, where a “guess” for the density was used for the pre-training of the PINN, a guess of the pressure field is also used to help during the pre-training of the PINN. The pressure field is assumed to follow a linear distribution with the streamwise coordinate, as follows:

$$p_{\text{guess}} = p_0 + \rho_{\text{air}} g_x x, \tag{12}$$

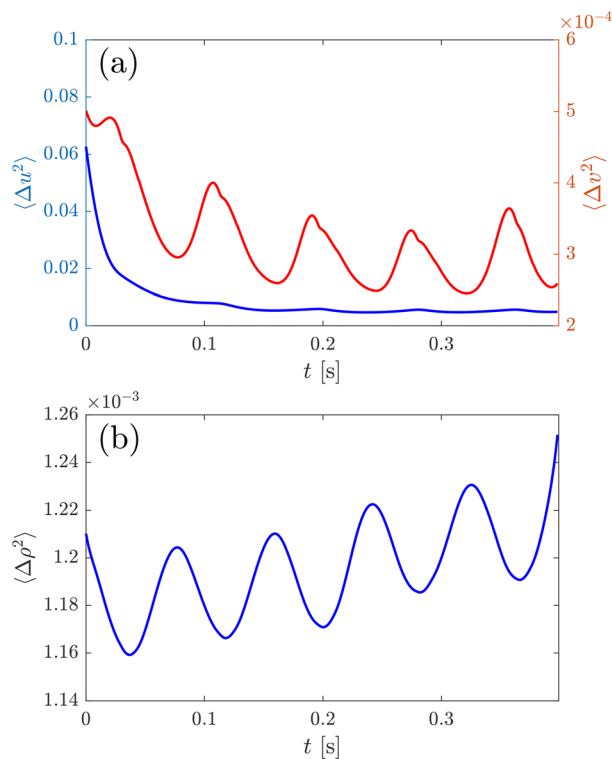


FIG. 18. Evolution of MSE of the reconstruction of (a)  $u$  and  $v$  (b)  $\rho$ .

where  $\rho_{\text{air}}$  is the density of air at standard conditions and  $g_x = -9.81 \text{ m/s}^2$  is gravitational acceleration in axial direction. Then,  $\rho_{\text{guess}}$  is estimated as in Eq. (11), but using  $p_{\text{guess}}$  instead of  $p_{\text{data}}$ . The values obtained in this manner for  $\rho$  and  $p$  are then used during the pre-training stage (i.e., they replace the data of density and pressure). The reconstructed velocity field (for the same time instant as in Sec. III B 1) is shown in Figs. 19 and 20. It can be directly observed that the reconstruction accuracy is lower than in the previous case specifically in the outer flame region (for  $r^+ > 1.5$ ) and that the reconstructed  $v$ -velocity field (Fig. 20) does not accurately reproduce all the features related to the puffing of the flame. This inaccuracy in the velocity reconstruction can be related to the inaccurate reconstructed pressure field (shown in Fig. 21), where it can be seen that the radial variations of pressure are not accurately captured by the PINN. However, such variations are crucial as they induce the radial velocity.

Figure 22 shows the evolution of the mean squared reconstruction error for the velocity field,  $\rho$  and  $p$ . As expected, it can be seen that compared to the case where more measurements are available, the error is larger (one order of magnitude larger for the  $v$ -velocity), indicating that relying solely on one measurement (here the temperature) to reconstruct all the other quantities of interest may be out of reach of the current PINN architecture. This result would suggest that we would generally need at least two measurements (of temperature and pressure) to be able to deduce an appropriate kinematic field in a reacting flow. It may be possible that another combination of

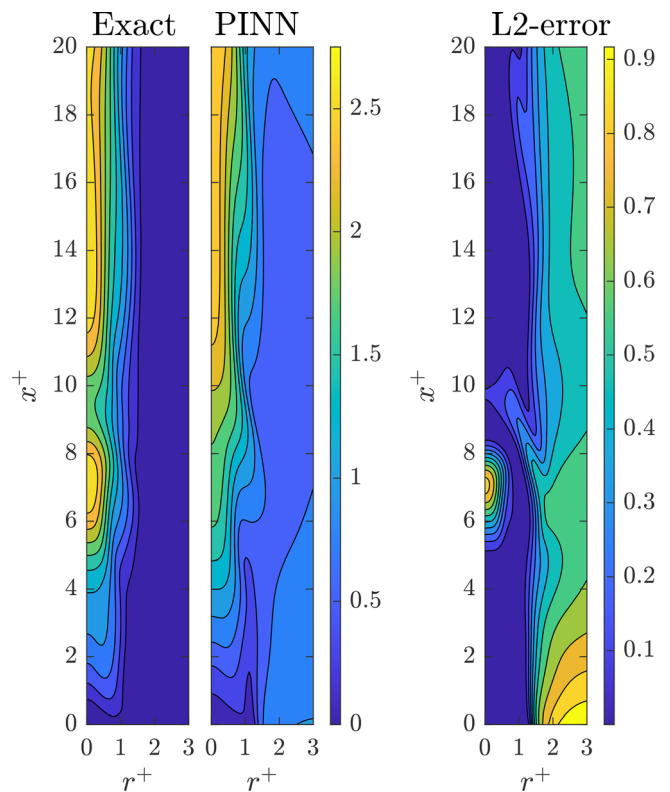


FIG. 19. Comparison between actual and reconstructed  $u$  velocity for the mid-time snapshot ( $t = 0.2 \text{ s}$ ). Superscript + indicates coordinates normalized by  $a$ .

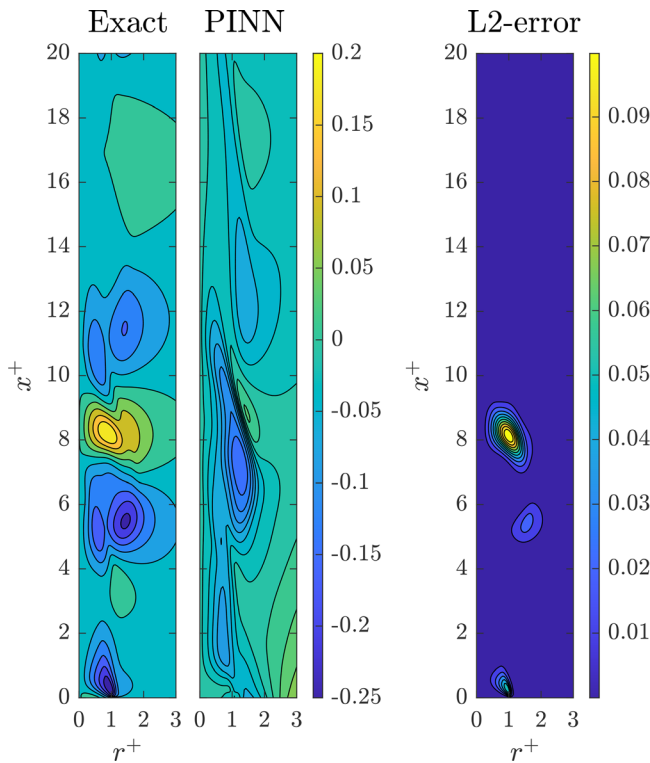


FIG. 20. Comparison between actual and reconstructed  $v$  velocity for the mid-time snapshot ( $t = 0.2$  s). Superscript + indicates coordinates normalized by  $a$ .

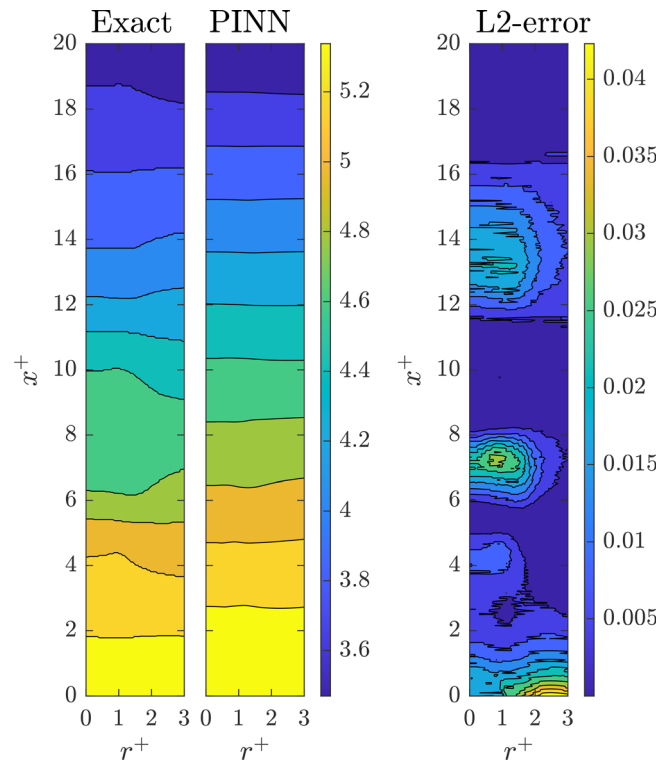


FIG. 21. Comparison between actual and reconstructed pressure  $p$  for the mid-time snapshot ( $t = 0.2$  s). Superscript + indicates coordinates normalized by  $a$ .

measured quantities (such as temperature and one of the chemical species) could achieve a higher accuracy but such an analysis is left for future investigations.

In addition, the observations above also highlight that, in the case when only one measurement becomes available, the optimization problem that the PINN has to solve (i.e., the minimization of the loss) becomes much more complex as there may exist various combinations of the other quantities to be reconstructed that could lead to a small loss values, but which are not the actual physical quantities. In other words, the PINN may reach a local minimum during its training which does not correspond to the actual flow solution. Such an observation shows the limit of the proposed PINN-based approach for the presented case of a puffing pool fire.

The presently noted deficiencies in the predictions of velocity from temperature can be compared to relative success in a previous work by Cai *et al.*,<sup>30</sup> albeit limited to a qualitative validation. The most apparent difference between the cases lies in the presence of both large temperature variations ( $\Delta T/T_0 > 1$ ) and significant variations of composition in the present pool fire, in contrast to small temperature variations, linearized density variations and negligible composition variations in the previous work. As a consequence, in the pool fire, larger errors are introduced when relating temperature with density and pressure. This problem is alleviated when two “measured” quantities are available. Evidently, two measured quantities allow to better describe two types of variations, temperature, and composition.

This may be related to the fact, that a diffusion flame structure cannot be described unambiguously by temperature alone. Nevertheless, variations of temperature and composition in a flame are not independent and could be related if the PINN was aware of the flame structure and able to detect the position of the flame through, for example, the measurements of a representative species.

#### IV. CONCLUSION

In this paper, we demonstrated for the first time the capability of the HFM framework to reconstruct unmeasured quantities from measured ones in the buoyancy-driven, reacting flow of a puffing pool fire. First, CFD simulations of pool fires at the onset of puffing were performed and validated against previous experiments to accurately reproduce the appropriate subcritical flame lengths, critical point, and puffing frequency. The present simulations can, therefore, be seen as a surrogate for experimental data, but the simultaneous availability of all variable fields allows to explore the capabilities and limitations of HFM. The dataset obtained from CFD results was then leveraged to train the PINN and to show its reconstruction capabilities: inferring the velocity field from measurements of density, pressure, and temperature—but without any observations of the velocity field itself—a high reconstruction accuracy was obtained. A further test was performed by considering noisy data. It was shown that the HFM framework could act as a physics-based “denoiser” as it could smooth out the added noise on the measured data. In addition, the PINN could still



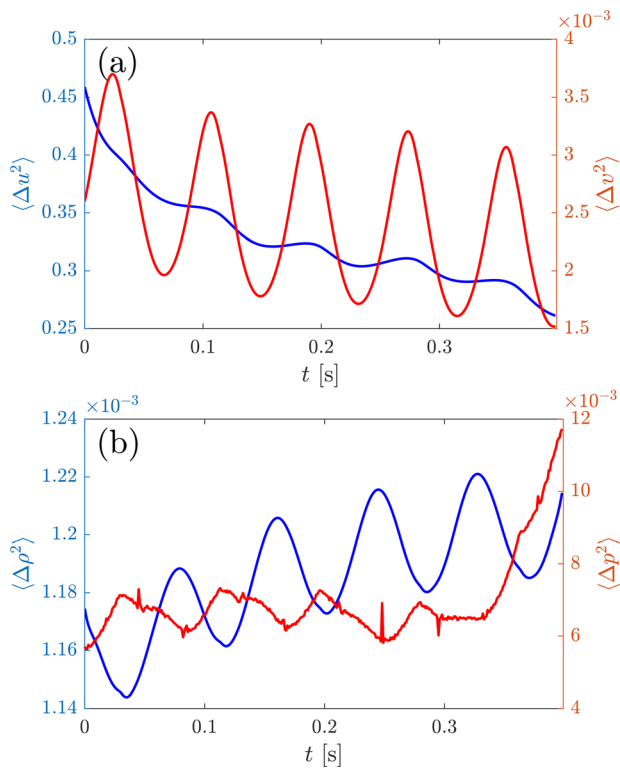


FIG. 22. Evolution of MSE of the reconstruction of (a)  $u$  and  $v$  (b)  $\rho$  and  $p$ .

accurately reconstruct the main features of the velocity field, which shows that the proposed method is robust with respect to noisy data. In a final step, it was attempted to reduce the number of measured fields required to predict the velocity field. Reasonable results were obtained when using density and pressure as inputs. However, it was not possible to make predictions from the temperature field alone. This emphasizes the complexity of the present case involving both large variations of temperature and chemical composition. This may suggest that a PINN capable of predicting the behavior of a flame needs to be able to identify the position and structure of the reaction zone.

This work demonstrates the potential of deep learning techniques for reconstructing unmeasured quantities in reacting flows in a physically consistent manner. This opens up possibilities for augmenting data obtained from experimental measurements in fire and combustion diagnostics.

In future work, the ability of this framework to reconstruct physical quantities in 3D and turbulent conditions will be explored. In addition, a further analysis will be conducted on assessing which measurements are required to be able to infer a unique kinematic field. The reconstruction of quantities from actual experimental data will also be investigated.

## ACKNOWLEDGMENTS

Photos of the puffing flame from experiments were kindly provided by Dr. W. Coenen from Universidad Carlos III de Madrid.

## AUTHOR DECLARATIONS

### Conflict of Interest

The authors have no conflicts to disclose.

### Author Contributions

**Michael Philip Sitte:** Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Project administration (equal); Software (equal); Supervision (equal); Validation (equal); Visualization (equal); Writing – original draft (equal); Writing – review and editing (equal). **Nguyen Anh Khoa Doan:** Conceptualization (equal); Data curation (equal); Formal analysis (equal); Investigation (equal); Methodology (equal); Project administration (equal); Software (equal); Supervision (equal); Validation (equal); Visualization (equal); Writing – original draft (equal); Writing – review and editing (equal).

## DATA AVAILABILITY

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

## REFERENCES

- <sup>1</sup>P. Joulain, “The behavior of pool fires: State of the art and new insights,” *Symp. (Int.) Combust.* **27**, 2691–2706 (1998).
- <sup>2</sup>A. Jenft, A. Collin, P. Boulet, G. Pianet, A. Breton, and A. Muller, “Experimental and numerical study of pool fire suppression using water mist,” *Fire Saf. J.* **67**, 1–12 (2014).
- <sup>3</sup>A. Lönnermark, A. Claesson, J. Lindström, Y. Z. Li, M. Kumm, and H. Ingason, “Full-scale fire tests with a commuter train in a tunnel,” SP Report No. 0284-5172, 2012.
- <sup>4</sup>C. Shi, M. Zhong, C. Chen, W. Jiao, J. Li, Y. Zhang, L. Zhang, Y. Li, and L. He, “Metro train carriage combustion behaviors—Full-scale experiment study,” *Tunnelling Underground Space Technol.* **104**, 103544 (2020).
- <sup>5</sup>R. N. Newman, “The ignition and burning behaviour of sodium metal in air,” *Prog. Nucl. Energy* **12**, 119–147 (1983).
- <sup>6</sup>M. Sibulkin and A. G. Hansen, “Experimental study of flame spreading over a horizontal fuel surface,” *Combust. Sci. Technol.* **10**, 85–92 (1975).
- <sup>7</sup>M. A. Finney, J. D. Cohen, J. M. Forthofer, S. S. McAllister, M. J. Gollner, D. J. Gorham, K. Saito, N. K. Akafuah, B. A. Adam, and J. D. English, “Role of buoyant flame dynamics in wildfire spread,” *Proc. Natl. Acad. Sci. U.S.A.* **112**, 9833–9838 (2015).
- <sup>8</sup>V. I. Blinov and G. N. Khudiakov, “Certain Laws Governing Diffusive Burning of Liquids,” *SSSR Doklady*, 113, 1094-1098 (Izdatel'stvo Akademii Nauk, Moscow, 1957). English translation available at: <https://apps.dtic.mil/sti/citations/AD0296762>
- <sup>9</sup>H. C. Hottel, “Certain laws governing diffusive burning of liquids,” *Fire Res. Abstr. Rev.* **1**, 41–44 (1959).
- <sup>10</sup>P. Thomas, R. Baldwin, and A. Heselden, “Buoyant diffusion flames: Some measurements of air entrainment, heat transfer, and flame merging,” *Symp. (Int.) Combust.* **10**, 983–996 (1965).
- <sup>11</sup>H. W. Emmons, “Scientific progress on fire,” *Annu. Rev. Fluid Mech.* **12**, 223–236 (1980).
- <sup>12</sup>S. R. Bishop, D. D. Drysdale, and G. Heskestad, “Dynamics of the fire plume,” *Philos. Trans. R. Soc. A* **356**, 2815–2833 (1998).
- <sup>13</sup>S. R. Tieszen, “On the fluid mechanics of fires,” *Annu. Rev. Fluid Mech.* **33**, 67–92 (2001).
- <sup>14</sup>A. Hamins, T. Kashiwagi, and R. R. Buch, “Characteristics of pool fire burning,” in *Fire Resistance of Industrial Fluids* (ASTM International, 1996).
- <sup>15</sup>D. Moreno-Boza, W. Coenen, J. Carpio, A. L. Sánchez, and F. A. Williams, “On the critical conditions for pool-fire puffing,” *Combust. Flame* **192**, 426–438 (2018).

- <sup>16</sup>G. Maragkos, T. Beji, and B. Merci, “Advances in modelling in CFD simulations of turbulent gaseous pool fires,” *Combust. Flame* **181**, 22–38 (2017).
- <sup>17</sup>G. Maragkos, T. Beji, and B. Merci, “Towards predictive simulations of gaseous pool fires,” *Proc. Combust. Inst.* **37**, 3927–3934 (2019).
- <sup>18</sup>M. M. Ahmed and A. Trouvé, “Large eddy simulation of the unstable flame structure and gas-to-liquid thermal feedback in a medium-scale methanol pool fire,” *Combust. Flame* **225**, 237–254 (2021).
- <sup>19</sup>S. Barwey, M. Hassanaly, V. Raman, and A. Steinberg, “Using machine learning to construct velocity fields from OH-PLIF images,” *Combust. Sci. Technol.* **194**(1), 93–116 (2019).
- <sup>20</sup>S. Barwey, V. Raman, and A. Steinberg, “Extracting information overlap in simultaneous OH-PLIF and PIV fields with neural networks,” *Proc. Combust. Inst.* **38**, 6241–6249 (2021).
- <sup>21</sup>K. Fukami, K. Fukagata, and K. Taira, “Machine-learning-based spatio-temporal super resolution reconstruction of turbulent flows,” *J. Fluid Mech.* **909**, A9 (2021).
- <sup>22</sup>J. L. Callahan, K. Maeda, and S. L. Brunton, “Robust flow reconstruction from limited measurements via sparse representation,” *Phys. Rev. Fluids* **4**, 103907 (2019).
- <sup>23</sup>N. B. Erichson, L. Mathelin, Z. Yao, S. L. Brunton, M. W. Mahoney, and J. N. Kutz, “Shallow neural networks for fluid flow reconstruction with limited sensors,” *Proc. R. Soc. A: Math. Phys. Eng. Sci.* **476**, 20200097 (2020).
- <sup>24</sup>D. W. Carter, F. De Voogt, R. Soares, and B. Ganapathisubramani, “Data-driven sparse reconstruction of flow over a stalled aerofoil using experimental data,” *Data-Centric Eng.* **2**, e5 (2021).
- <sup>25</sup>M. Raissi, A. Yazdani, and G. E. Karniadakis, “Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations,” *Science* **367**, 1026–1030 (2020).
- <sup>26</sup>N. J. Nair and A. Goza, “Leveraging reduced-order models for state estimation using deep learning,” *J. Fluid Mech.* **897**, R1 (2020).
- <sup>27</sup>P. Saini, C. M. Arndt, and A. M. Steinberg, “Development and evaluation of gappy-POD as a data reconstruction technique for noisy PIV measurements in gas turbine combustors,” *Exp. Fluids* **57**, 122 (2016).
- <sup>28</sup>M. Raissi, P. Perdikaris, and G. 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).
- <sup>29</sup>H. Eivazi, M. Tahani, P. Schlatter, and R. Vinuesa, “Physics-informed neural networks for solving Reynolds-averaged Navier-Stokes equations,” in Proceedings of the 13th ERCOFTAC Symposium on Engineering Turbulence Modeling and Measurements (ETMM13), Rhodes, Greece, 15–17 September (2021).
- <sup>30</sup>S. Cai, Z. Wang, F. Fuest, Y. J. Jeon, C. Gray, and G. E. Karniadakis, “Flow over an espresso cup: Inferring 3-D velocity and pressure fields from tomographic background oriented Schlieren via physics-informed neural networks,” *J. Fluid Mech.* **915**, A102 (2021).
- <sup>31</sup>H. Eivazi and R. Vinuesa, “Physics-informed deep-learning applications to experimental fluid mechanics,” [arXiv:2203.15402](https://arxiv.org/abs/2203.15402) (2022).
- <sup>32</sup>H. Wang, S. Liu, and Y. Wang, “Dense velocity reconstruction from particle image velocimetry/particle tracking velocimetry using a physics-informed neural network,” *Phys. Fluids* **34**, 017116 (2022).
- <sup>33</sup>The OpenFOAM Foundation, see <https://openfoam.org/> for “OpenFOAM-7, 2019.”
- <sup>34</sup>T. Poinso and D. Veynante, *Theoretical and Numerical Combustion*, 2nd ed. (R.T. Edwards, Inc., Philadelphia, 2005).
- <sup>35</sup>E. Fernández-Tarrazo, A. L. Sánchez, A. Liñán, and F. A. Williams, “A simple one-step chemistry model for partially premixed hydrocarbon combustion,” *Combust. Flame* **147**, 32–38 (2006).
- <sup>36</sup>M. Raissi, Z. Wang, M. S. Triantafyllou, and G. Karniadakis, “Deep learning of vortex-induced vibrations,” *J. Fluid Mech.* **861**, 119–137 (2019).
- <sup>37</sup>P. Ramachandran, B. Zoph, and Q. V. Le, “Searching for activation functions,” [arXiv:1710.05941](https://arxiv.org/abs/1710.05941) (2017).
- <sup>38</sup>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 Proceedings of the NIPS 2017 Autodiff Workshop Long Beach, CA, USA, 9 December (2017).
- <sup>39</sup>D. P. Kingma and J. L. Ba, “Adam: A method for stochastic optimization,” in Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May (2015).
- <sup>40</sup>R. W. Bilger, “The structure of diffusion flames,” *Combust. Sci. Technol.* **13**, 155–170 (1976).
- <sup>41</sup>I. E. Lagaris, A. Likas, and D. I. Fotiadis, “Artificial neural networks for solving ordinary and partial differential equations,” *IEEE Trans. Neural Networks.* **9**, 987–1000 (1988).