

# A-posteriori evaluation of a deep convolutional neural network approach to subgrid-scale flame surface estimation

By C. J. Lapeyre<sup>†</sup>, A. Misdariis<sup>†</sup>, N. Cazard<sup>†</sup> AND T. Poinso<sup>‡</sup>

Deep learning (DL) and the field of artificial intelligence (AI) have been hot topics in the software industry in 2018, notably in the field of convolutional neural networks (CNNs). In fluid dynamics, recent studies are starting to show promising results, including for large eddy simulation (LES) applications. In this work, a CNN previously trained to replace a model for the unresolved flame surface in turbulent premixed combustion is implemented inside a parallel LES solver. *A-posteriori* comparisons are made with a direct numerical simulation (DNS) of a fully resolved flame, and show good agreement. A state-of-the-art dynamic method is included for comparison, and the CNN outperforms it on the target configuration.

---

## 1. Motivation and objectives

Deep learning is a fast-paced and hot topic in the software industry, at the center of many so-called AI applications. However, a logical model of biological neurons was first introduced by McCulloch & Pitts (1943), rendering the development of artificial neural networks (ANNs) possible. The mechanism by which neural networks learn was explained much later by Fukushima (1975), and this led to some high-performing computer vision applications for simple datasets in the 1990s by LeCun *et al.* (1998). However, the training phase of these algorithms, called learning, limited their applications in more complex cases. Advancements in hardware performance, database sizes and network training techniques (Hinton *et al.* 2006) in the past decade have enabled DL to significantly outperform previous approaches from natural language processing (notable for beating top human players at the game Jeopardy!; Ferrucci *et al.* (2010)) to computer vision (winner of the 2012 ImageNet challenge; Krizhevsky *et al.* (2012)), to game playing (world's strongest algorithm for playing Go, Chess and Shogi; Silver *et al.* (2017)).

Implementation of such algorithms has gone rapidly from reserved to a small group of domain experts to largely available to the public, thanks to modern frameworks with high levels of abstraction (Abadi *et al.* 2015). Applications to the field of computational fluid dynamics (CFD), however, are still scarce. Some shallow networks have been applied successfully to turbulence modeling (Ling *et al.* 2016; Vollant *et al.* 2017), but the full power of DL that stems specifically from deep feedforward networks (Goodfellow *et al.* 2016), and notably CNNs, is still rare in CFD, as evidenced by a recent review (Duraiamy *et al.* 2018). In recent work by Beck *et al.* (2018), however, deep residual networks were trained to recover state-of-the-art turbulent viscosity models on homogeneous isotropic turbulence.

<sup>†</sup> CERFACS, Toulouse, France

<sup>‡</sup> Institut de Mécanique des Fluides de Toulouse, France

In the combustion community, the determination of the subgrid-scale (SGS) contribution to the filtered reaction rate in LES of reacting flows is an example of closure a problem that has been daunting for a long time. Indeed, SGS interactions between the flame and turbulent scales largely determine the flame behavior, and modeling them is an important factor in obtaining overall flame dynamics. Many turbulent modeling approaches are based on a reconstruction of the SGS wrinkling of the flame surface and the so-called flamelet assumption (Poinsot & Veynante 2011). Under this assumption, the mean turbulent reaction rate can be expressed in terms of flame surface area (Marble & Broadwell 1977; Candel *et al.* 1990). Indeed, the idea that turbulence convects, deforms, and spreads surfaces (Pope 1988) can be applied to a premixed flame front in a turbulent flow. The evaluation of the amount of flame surface area due to unresolved flame wrinkling has been at the core of all models based on flame surface areas in the last 50 years (Poinsot & Veynante 2011), both for Reynolds-averaged Navier-Stokes (Bray & Moss 1977; Peters 1986; Duclos *et al.* 1993; Bruneaux *et al.* 1997) and for LES (Boger *et al.* 1998; Knikker *et al.* 2004). Recent developments in dynamic procedures (Wang *et al.* 2011) have shown that the extraction of some topological information could increase the accuracy of models. CNNs may be viewed as a natural extension of this approach: multi-layer convolutions can be trained to automatically aggregate multi-scale information to predict the desired output. They can also be seen as an extension of earlier applications of multi-layer perceptrons for numerical combustion applications (Christo *et al.* 1996; Ihme *et al.* 2009). Initial *a priori* tests have shown that CNNs are capable of achieving very high accuracy in predicting unresolved flame wrinkling (Lapeyre *et al.* 2018).

The present work extends those results: Section 2 recalls the theoretical context of the approach, as detailed by Lapeyre *et al.* (2018). Section 3 describes the model training. Its performance on a test set, as well as its capacity to generalize its knowledge to a new setup with an unknown flame, is assessed. Finally, Section 4 showcases the simulations performed for *a-posteriori* validation, with the CNN running live during the LES.

## 2. Theoretical aspects of the model

### 2.1. Flame surface density models

In (LES), flow features are split between resolved and non-resolved scales using a spatial filter. Defining a quantity  $Q$  in a fully resolved flow field, the low-pass spatial filter  $F_\Delta$  with width  $\Delta$  is used to isolate the resolved scales, yielding

$$\overline{Q(\mathbf{x}, t)} = \int_{\mathcal{V}} F_\Delta(\mathbf{x} - \mathbf{x}') Q(\mathbf{x}', t) d\mathbf{x}' \quad (2.1)$$

where  $\overline{Q(\mathbf{x}, t)}$  is the result of the filtering operation applied to  $Q(\mathbf{x}, t)$ . In this study, only perfectly premixed combustion is considered, and a progress variable  $c$  is defined as

$$c = \frac{T - T_u}{T_b - T_u}, \quad (2.2)$$

with  $T$  the temperature and subscripts  $u$  and  $b$  referring to unburnt and burnt gases, respectively. A balance equation can be written for  $c$  (Poinsot & Veynante 2011) by defining a density-weighted (or Favre) filtering  $\tilde{Q} = \overline{\rho Q} / \bar{\rho}$  for every quantity  $Q$ , where  $\rho$  is the fluid density. Filtering the progress variable equation written in a propagative form (G-equation; Kerstein *et al.* (1988)) assuming local flame elements gives (Knikker

*et al.* 2004)

$$\frac{\partial \bar{\rho} \bar{c}}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \bar{c}) + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \bar{c} - \bar{\rho} \tilde{\mathbf{u}} \bar{c}) = \rho_u S_L^0 \bar{\Sigma}, \quad (2.3)$$

where the term on the right hand side incorporates filtered diffusion and reaction terms into a single  $c$ -isosurface displacement speed assimilated to laminar flame speed  $S_L^0$ , and where  $\rho_u$  is the fresh gases density.  $\bar{\Sigma} = |\nabla \bar{c}|$  is the generalized flame surface density (Boger *et al.* 1998), and cannot be obtained in general from resolved flame surfaces. Indeed, when filtering  $c$  to  $\bar{c}$ , surface wrinkling decreases, resulting in less total  $c$ -isosurface. One popular method to model  $\bar{\Sigma}$  is to introduce the wrinkling factor  $\Xi$  that compares the total and resolved generalized flame surfaces. The term on the right-hand side of Eq. (2.3) is then rewritten as

$$\rho_u S_L^0 \bar{\Sigma} = \rho_u S_L^0 \Xi |\nabla \bar{c}|, \quad \text{where} \quad \Xi = \frac{\bar{\Sigma}}{|\nabla \bar{c}|}. \quad (2.4)$$

Fractal approaches such as introduced by Gouldin *et al.* (1989) suggest a relationship between  $\bar{\Sigma}$  and  $|\nabla \bar{c}|$  of the form

$$\bar{\Sigma} = \left( \frac{\Delta}{\eta_c} \right)^{D_f - 2} |\nabla \bar{c}|, \quad (2.5)$$

where  $D_f$  is the fractal dimension of the flame surface, and  $\eta_c$  is the inner cutoff scale below which the flame is no longer wrinkled. The  $\eta_c$  length scales with the laminar flame thickness  $\delta_L^0$  (Poinsot *et al.* 1991; Gulder & Smallwood 1995). More recent work, based on flame-vortex interactions and multi-fractal analysis (Charlette *et al.* 2002b), suggests a different form (modified to recover Eq. (2.5) at saturation by Wang *et al.* (2011))

$$\bar{\Sigma} = \left( 1 + \min \left[ \frac{\Delta}{\delta_L^0} - 1, \Gamma_\Delta \left( \frac{\Delta}{\delta_L^0}, \frac{u'_\Delta}{S_L^0}, Re_\Delta \right) \frac{u'_\Delta}{S_L^0} \right] \right)^\beta |\nabla \bar{c}|, \quad (2.6)$$

where  $\beta$  is a generalized parameter inspired from the fractal dimension. The  $\Gamma_\Delta$  function is meant to incorporate the strain induced by the unresolved scales between  $\Delta$  and  $\eta_c$ . Extensions of this model have also been proposed to compute the parameter  $\beta$  dynamically (Charlette *et al.* 2002a; Wang *et al.* 2011). From a machine learning standpoint, these all correspond to predicting the same output  $\bar{\Sigma}$ , but using several local input variables:  $(\bar{c}, \Delta/\delta_L^0, u'_\Delta/S_L^0)$ . More variables could be included to further generalize the approach, e.g. information about the chemical state, since the machine learning framework does not require a strict physical formulation.

## 2.2. Reformulation in the machine learning context

Flame surface density estimation can be seen as the issue of relating the input field  $\bar{c}$  to a matching output field  $\bar{\Sigma}$ . Supervised learning of this task can be implemented as follows:

- In a first phase, a dataset generated using a DNS is used, where both  $\bar{c}$  and  $\bar{\Sigma}$  are known exactly. Models are trained on this data in a supervised manner.
- In a second phase, the best-trained model is frozen. It is executed in an LES context, where  $\bar{c}$  is known but not  $\bar{\Sigma}$ .

Name	$u'/S_L^0$	Inlet velocity	Resolution	Turbulent combustion model
TRAIN1	1.23	Constant $\bar{u}$	DNS	Resolved
TRAIN2	2.47	Constant $\bar{u}$	DNS	Resolved
PULSE_DNS	1.23	Sinewave $\bar{u} + \hat{u} \sin \omega t$	DNS	Resolved
PULSE_CNN	1.23	Sinewave $\bar{u} + \hat{u} \sin \omega t$	LES	Neural network (Lapeyre <i>et al.</i> 2018)
PULSE_DYN	1.23	Sinewave $\bar{u} + \hat{u} \sin \omega t$	LES	Dynamic (Wang <i>et al.</i> 2011)

TABLE 1. Simulations performed in this study. Inlet velocity is  $\bar{u} + u' + \hat{u}$ , where  $u'$  corresponds to turbulent fluctuations and  $\hat{u}$  to harmonic forcing. For all cases,  $\bar{u}/S_L^0 = 24.7$ .

In order to obtain  $\bar{c}$  and  $\bar{\Sigma}$ , an explicit Gaussian filter is implemented. Its width is defined as the multiplying factor on the maximum gradient  $|\nabla c|$ , i.e.,

$$\Delta = \frac{\max|\nabla c|}{\max|\nabla \bar{c}|} \Delta x, \quad (2.7)$$

computed on a one-dimensional (1D) laminar flame at the same equivalence ratio, pressure, and inlet temperature. The resulting 1D filter function is therefore written in discrete form as

$$F_\Delta(x) = \begin{cases} e^{-\frac{1}{2}(\frac{x}{\sigma})^2} & \text{if } n \in \llbracket 1, N \rrbracket \\ 0 & \text{otherwise,} \end{cases} \quad (2.8)$$

and then normalized by its sum,  $\sum_{n \in \llbracket 0, N \rrbracket} F_\Delta(n)$ . Here,  $\sigma = 20$  and  $N = 15$  are optimized to obtain a filter width of  $\Delta = 4\Delta x \approx 1.2 \delta_l^0$ . Three-dimensional (3D) filtered fields are obtained by applying the 1D filtering successively on each direction  $x$ ,  $y$ , and  $z$ .

### 3. Deep learning approach to combustion modeling

In this study, a numerical setup of a methane-air burner is used to perform both resolved DNS and LES.

The target configuration is a premixed stoichiometric methane-air slot burner. The flame is stabilized by hot gases. A numerical domain consisting of a 3D box of  $512 \times 256 \times 256$  cells is used for DNS, with all cells  $\Delta x = 0.1$  mm in each direction. Figure 1(a) shows a view of the domain and typical flame. The reacting flow field is simulated using the fully compressible explicit code AVBP, which solves the filtered multi-species 3D Navier-Stokes equations with simplified thermochemistry on unstructured meshes (Schönfeld & Rudgyard 1999; Selle *et al.* 2004). A Taylor–Galerkin finite element scheme called TTGC (Colin & Rudgyard 2000) of third order in space and time is used. Inlet and outlet boundary conditions are treated using an Navier-Stokes characteristic boundary condition approach (Poinsot & Lele 1992) with transverse term corrections (Granet *et al.* 2010). Other boundaries are treated as periodic. The same domain is also used for LES, with a coarse mesh of  $128 \times 64 \times 64$  cells. In LES, SGS eddy viscosity is included using the model of Smagorinsky (1963). All simulations performed in this study are summarized in Table 1.

The ANN used in this study was developed in previous work by Lapeyre *et al.* (2018). It was inspired from the convolutional U-Net segmentation architecture (Ronneberger *et al.* 2015), adapted for the regression task of predicting  $\bar{\Sigma}$  from  $\bar{c}$ . The CNN can be viewed as a function  $f$  that operates on an entire subset  $\Omega$  of the mesh at once. Instead of algebraically predicting  $\bar{\Sigma}$  from  $\bar{c}$  at one node, the CNN reads the vector  $\bar{c}$  of values over an entire subdomain,  $\Omega$ , and produces a prediction for the matching field of  $\bar{\Sigma}$  over

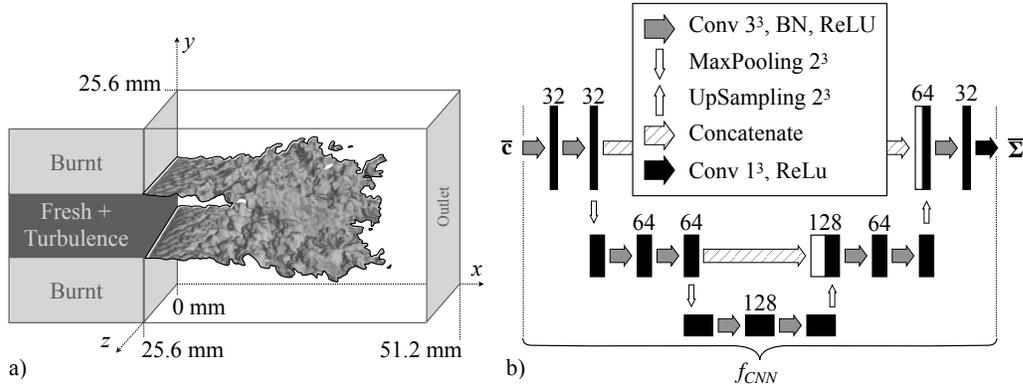


FIGURE 1. (a) Numerical domain and view of a typical flame from the TRAIN1 simulation. (b) Architecture of the U-net neural network used in this study. Numbers above convolutional layers show the amount of filters used in the layer.

$\Omega$ . The function  $f_{CNN}$  therefore performs

$$\begin{aligned} \bar{\Sigma} &= f_{CNN}(\bar{c}), \\ f_{CNN} : \mathbb{R}^{\Omega} &\mapsto \mathbb{R}^{\Omega}. \end{aligned} \quad (3.1)$$

The CNN consumes a matrix, meaning that  $\Omega$  must be a 3D regular grid. The neural network architecture is shown in Figure 1(b). Because of this architecture, a constraint is placed on  $\Omega$ : each dimension must be a multiple of four. Other than that, there are no constraints on the input to the network, and it can be used on arbitrarily large domains. It has been retrained on data from TRAIN1 and TRAIN2 simulations with the thickening factor of four of this study, according to the procedure from Lapeyre *et al.* (2018):

- $c$  and  $\Sigma$  are extracted from the DNS flow fields;
- $\bar{c}$  and  $\bar{\Sigma}$  are produced using the explicit filtering approach of Eq. (2.8) on these fields and
- the network is trained to learn the function  $f$  from Eq. (3.1).

#### 4. *A-posteriori* evaluation of the method

The CNN trained with TRAIN1 and TRAIN2 runs is used as an SGS model for  $\Sigma$  in an *a-posteriori* LES of the same configuration: PULSE\_CNN. To make the test more representative, the inlet mean velocity is pulsated with an amplitude  $\hat{u}/\bar{u} = 15\%$ . The forcing frequency is  $f = 1$  kHz. A simulation using the best present dynamic model (PULSE\_DYN) is also performed as a reference, with a flame thickening ratio of four matching the coarsening factor of the LES mesh versus the DNS.

Using the neural network output directly in the CFD code is challenging. Indeed, the Navier-Stokes solver is based on a parallel domain-decomposition approach, optimized for multi-central processing unit CPU architectures. The CNN, however, is not appropriate for domain decomposition: one CNN evaluation, called inference, must be performed on a single 3D regular matrix coinciding with the entire LES mesh. Since the model must be called at each time step and CNNs are CPU intensive, the inference time must be of the same order of magnitude as the Navier-Stokes solver computational time. Initial measurements by Cazard (September 2018) have shown that inference times on an Nvidia Tesla V100 were  $\sim 200$  times faster than on a 2.3 GHz Xeon Gold 6140 Skylake core. The

strategy retained for this study was therefore to use a hybrid graphical processing unit (GPU)/CPU approach: the AVBP solver ran in parallel on CPU cores, and the CNN inference exploited a GPU. The OpenPALM coupling library (Duchaine *et al.* 2015) handled interpolations on each grid and communications between CPU and GPU.

The PULSE\_CNN simulation requires two meshes, an unstructured one for the AVBP solver and a regular one for the CNN. Second-order interpolation weights between the two domains are computed once after domain partitioning, then used at each time step throughout the simulation. For this case, one hybrid node including 36 Intel Skylake (Xeon Gold 6140) cores and one NVIDIA Tesla M60 GPU was used. At each time step, the full progress variable  $c$  field was sent to the CNN. Inference on the GPU then yielded the full efficiency  $\Xi$  and flame sensor  $F$  fields, which were then interpolated back on the CFD solver mesh and used to finish the iteration. Artificial thickening by a factor of four is used to account for mesh coarseness. In order to limit the impact on the turbulence, a flame sensor was used to apply thickening only locally in areas where  $|\nabla c|$  was at 10% or more of its maximum value. The resulting thickening factor can be written as

$$F = 4 \times \frac{1}{2} \left[ 1 + \tanh \left( |\nabla c| - 0.1 |\nabla c|^{max} \right) \right]. \quad (4.1)$$

On the Tesla M60 GPU, each neural net inference requires 0.2 s for the whole domain. The CFD solver computation requires 0.6 s on the 36 Intel Skylake cores for each iteration. The cost of the inference is therefore lower than the cost of the CFD simulation, and the overhead introduced by this approach is shown to be manageable in the context of this first demonstrator. Further improvements on the GPU efficiency are, however, possible, as this approach is new and not yet as mature as the CFD solver.

All three PULSE simulations are run until a harmonic state is reached, then recorded every 0.2 ms for a total of 6 ms (i.e., six pulsation periods). Figure 2 shows a typical view of the heat release rate as a marker for the flame. The LES simulations (PULSE\_CNN and PULSE\_DYN) exhibit much less fine structures than PULSE\_DNS. While normalization is done on a per-simulation basis (hence scales are not directly comparable), it should be noted in this *a-posteriori* approach that the apparent flame thickness is not the same in both LESs. This could be due to the behavior of the CNN, and the resulting flame structure being different. Nevertheless, the flame length seems to be better recovered by the CNN-based LES. To explore this aspect, average fields of progress variable at several positions downstream of the slot are plotted versus distance from the axis in Figure 3. Both combustion models show reasonable agreement with the DNS. However, the PULSE\_DYN simulation significantly underestimates the progress variable on the axis in the intermediate stages. At the second observation position ( $x = 22.8$  mm), the progress variable is already above 50% for PULSE\_DNS, but still close to zero for PULSE\_DYN. PULSE\_CNN, however, recovers the DNS trend much better.

## 5. Conclusions

In this study, a neural network trained to predict flame surface density is used for an *a-posteriori* evaluation, and compared to a state-of-the-art method from the literature known as the dynamic formulation. The successful coupling between an LES solver and a CNN to act as a physical model opens many new possibilities for the field of CFD.

One of the major challenges in developing data-driven alternatives to physical models is to estimate how well they will generalize to new setups. Indeed, with physics-based models, even if some coefficient tuning is performed, the hypotheses made to derive it

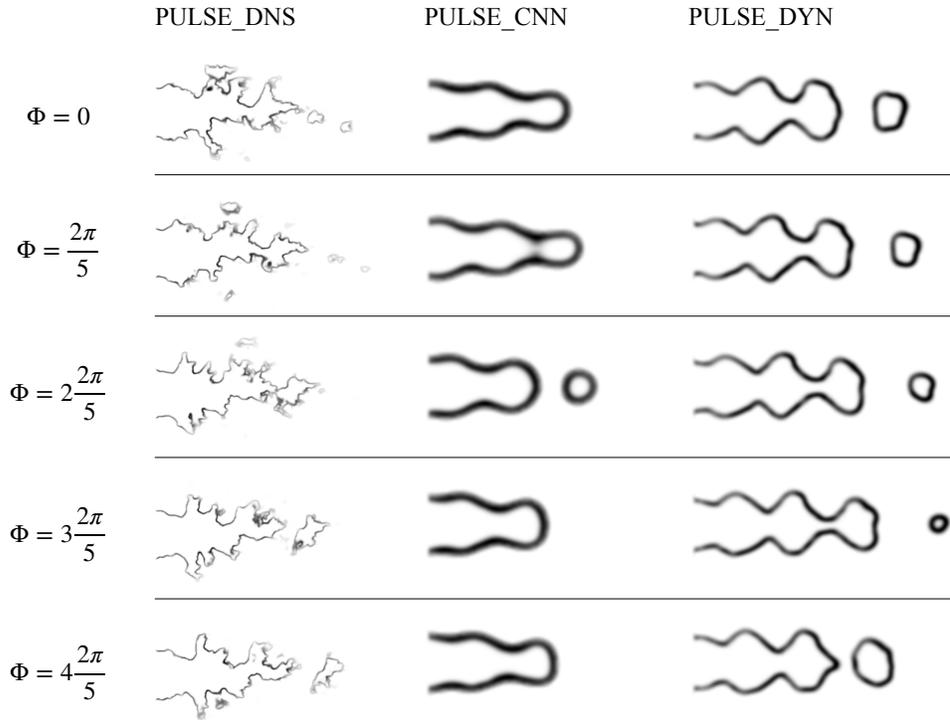


FIGURE 2. Views of instantaneous phase-synchronized flames (normalized heat release rate) for each PULSE simulation.

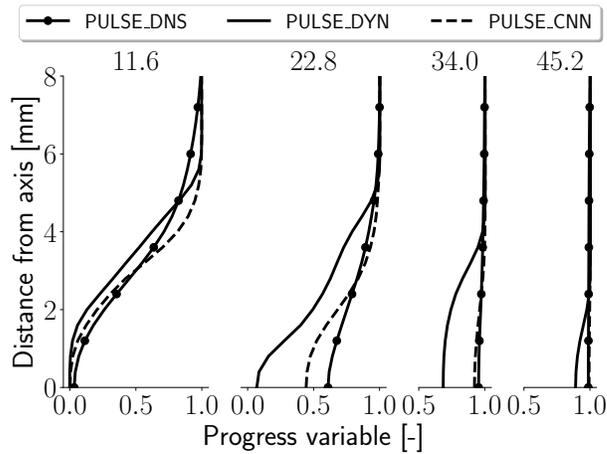


FIGURE 3. Mean profiles of progress variable for PULSE simulations at four downstream locations, as labeled in mm above each plot.

suggest the limits of its applicability. In the case of a model fully learned from the data, as used here, there are no physics-based hypotheses to rely on. Instead, the classical approach in machine learning is to identify a target configuration to produce a so-called validation dataset, which is never observed during training. The prediction error on this dataset is monitored during training. When this validation error increases, this demon-

strates that while the model is further learning the training set, it is doing so by losing generality.

During the Summer Program, exchanges with other participants led to identifying a similar configuration to the slot burner presented here, but with significant differences that could lead the model to not generalize well on it. Indeed, MacArt *et al.* (2018) performed DNS of a stoichiometric hydrogen-air slot burner with nitrogen dilution. The K1 setup with Karlovitz number 3.7 has  $u'/S_L^0 \approx 1.25$ , very close to the values used in the TRAIN1 DNS. Comparisons demonstrated that indeed in this case the network initially learned a function that described both datasets well. However, as the training process went beyond a certain point, the prediction error for the K1 case stopped improving, and eventually increased again. From a machine learning standpoint, this result is very encouraging. Indeed, it highlights the point when the neural network is no longer learning general features related to flame wrinkling, but instead specifics of the chosen training setup. Further work is now needed to improve upon these results in order to achieve a more general result for this type of approach.

#### Acknowledgments

The authors thank Prof. Matthias Ihme and Dr. Thomas Jaravel for fruitful discussions on this topic.

#### REFERENCES

- ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C., CORRADO, G. S., DAVIS, A., DEAN, J., DEVIN, M., GHEMAWAT, S., GOODFELLOW, I., HARP, A., IRVING, G., ISARD, M., JIA, Y., JOZEFOWICZ, R., KAISER, L., KUDLUR, M., LEVENBERG, J., MANÉ, D., MONGA, R., MOORE, S., MURRAY, D., OLAH, C., SCHUSTER, M., SHLENS, J., STEINER, B., SUTSKEVER, I., TALWAR, K., TUCKER, P., VANHOUCHE, V., VASUDEVAN, V., VIÉGAS, F., VINYALS, O., WARDEN, P., WATTENBERG, M., WICKE, M., YU, Y. & ZHENG, X. 2015 TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
- BECK, A. D., FLAD, D. G. & MUNZ, C.-D. 2018 Deep neural networks for data-based turbulence models. *ArXiv preprint arXiv 1806.04482*.
- BOGER, M., VEYNANTE, D., BOUGHANEM, H. & TROUVÉ, A. 1998 Direct numerical simulation analysis of flame surface density concept for large eddy simulation of turbulent premixed combustion. In *27th Symp. (Int.) on Combustion*, pp. 917–927. Boulder: The Combustion Institute, Pittsburgh.
- BRAY, K. N. C. & MOSS, J. B. 1977 A unified statistical model of the premixed turbulent flame. *Acta Astronaut.* **4**, 291 – 319.
- BRUNEAUX, G., POINSOT, T. & FERZIGER, J. H. 1997 Premixed flame-wall interaction in a turbulent channel flow: budget for the flame surface density evolution equation and modelling. *J. Fluid Mech.* **349**, 191–219.
- CANDEL, S., VEYNANTE, D., LACAS, F., MAISTRET, E., DARABIHA, N. & POINSOT, T. 1990 Coherent flame model: applications and recent extensions. In *Advances in combustion modeling. Series on advances in mathematics for applied sciences* (ed. B. Larrouturou), pp. 19–64. World Scientific, Singapore.
- CAZARD, N. September 2018 Exploration de la capacité du Deep Learning à assister des simulations de mécanique des fluides. *Tech. Rep.* Preprint 941. CERFACS.

- CHARLETTE, F., MENEVEAU, C. & VEYNANTE, D. 2002a A power-law flame wrinkling model for LES of premixed turbulent combustion part ii: dynamic formulation. *Combust. Flame* **131**, 181–197.
- CHARLETTE, F., VEYNANTE, D. & MENEVEAU, C. 2002b A power-law wrinkling model for LES of premixed turbulent combustion: Part I - non-dynamic formulation and initial tests. *Combust. Flame* **131**, 159–180.
- CHRISTO, F. C., MASRI, A. R., NEBOT, E. M. & POPE, S. B. 1996 An integrated PDF/neural network approach for simulating turbulent reacting systems. *Proc. Combust. Inst.* **26**, 43–48.
- COLIN, O. & RUDGYARD, M. 2000 Development of high-order Taylor-Galerkin schemes for unsteady calculations. *J. Comput. Phys.* **162**, 338–371.
- DUCHAINE, F., JAURÉ, S., POITOU, D., QUÉMERAIS, E., STAFFELBACH, G., MOREL, T. & GICQUEL, L. 2015 Analysis of high performance conjugate heat transfer with the openpalm coupler. *Comp. Sci. Dis.* **8**, 015003.
- DUCLOS, J. M., VEYNANTE, D. & POINSOT, T. 1993 A comparison of flamelet models for premixed turbulent combustion. *Combust. Flame* **95**, 101–117.
- DURASAMY, K., IACCARINO, G. & XIAO, H. 2018 Turbulence modeling in the age of data. *ArXiv preprint arXiv:1804.00183* .
- FERRUCCI, D., BROWN, E., CHU-CARROLL, J., FAN, J., GONDEK, D., KALYANPUR, A. A., LALLY, A., MURDOCK, J. W., NYBERG, E., PRAGER, J. *et al.* 2010 Building watson: An overview of the deepqa project. *AI magazine* **31** (3), 59–79.
- FUKUSHIMA, K. 1975 Cognitron: A self-organizing multilayered neural network. *Biological cybernetics* **20**, 121–136.
- GOODFELLOW, I., BENGIO, Y. & COURVILLE, A. 2016 *Deep Learning*. MIT Press, <http://www.deeplearningbook.org>.
- GOULDIN, F., BRAY, K. & CHEN, J. Y. 1989 Chemical closure model for fractal flamelets. *Combust. Flame* **77**, 241.
- GRANET, V., VERMOREL, O., LEONARD, T., GICQUEL, L., & POINSOT, T. 2010 Comparison of nonreflecting outlet boundary conditions for compressible solvers on unstructured grids. *AIAA Journal* **48**, 2348–2364.
- GULDER, O. L. & SMALLWOOD, G. J. 1995 Inner cutoff scale of flame surface wrinkling in turbulent premixed flames. *Combust. Flame* **103**, 107–114.
- HINTON, G. E., OSINDERO, S. & TEH, Y.-W. 2006 A fast learning algorithm for deep belief nets. *Neural computation* **18**, 1527–1554.
- IHME, M., SCHMITT, C. & PITSCH, H. 2009 Optimal artificial neural networks and tabulation methods for chemistry representation in {LES} of a bluff-body swirl-stabilized flame. *Proc. Combust. Inst.* **32**, 1527–1535.
- KERSTEIN, A. R., ASHURST, W. & WILLIAMS, F. A. 1988 Field equation for interface propagation in an unsteady homogeneous flow field. *Phys. Rev. A* **37**, 2728–2731.
- KNIKKER, R., VEYNANTE, D. & MENEVEAU, C. 2004 A dynamic flame surface density model for large eddy simulation of turbulent premixed combustion. *Phys. Fluids* **16**, L91–L94.
- KRIZHEVSKY, A., SUTSKEVER, I. & HINTON, G. E. 2012 Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pp. 1097–1105.
- LAPEYRE, C. J., MISDARIIS, A., CAZARD, N., VEYNANTE, D. & POINSOT, T. 2018

- Training convolutional neural networks to estimate turbulent sub-grid scale reaction rates. *ArXiv preprint arXiv 1810.03691*.
- LECUN, Y., BOTTOU, L., BENGIO, Y. & HAFFNER, P. 1998 Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **86**, 2278–2324.
- LING, J., KURZAWSKI, A. & TEMPLETON, J. 2016 Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *J. Fluid Mech.* **807**, 155–166.
- MACART, J. F., GRENGA, T. & MUELLER, M. E. 2018 Effects of combustion heat release on velocity and scalar statistics in turbulent premixed jet flames at low and high karlovitz numbers. *Combust. Flame* **191**, 468–485.
- MARBLE, F. E. & BROADWELL, J. E. 1977 The coherent flame model for turbulent chemical reactions. *Tech. Rep.* Tech. Rep. TRW-9-PU. Project Squid.
- MCCULLOCH, W. S. & PITTS, W. 1943 A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics* **5**, 115–133.
- PETERS, N. 1986 Laminar flamelet concepts in turbulent combustion. In *21st Symp. (Int.) on Combustion*, pp. 1231–1250. The Combustion Institute, Pittsburgh.
- POINSOT, T. & LELE, S. 1992 Boundary conditions for direct simulations of compressible viscous flows. *J. Comput. Phys.* **101**, 104–129.
- POINSOT, T. & VEYNANTE, D. 2011 *Theoretical and Numerical Combustion*. Third Edition ([www.cerfacs.fr/elearning](http://www.cerfacs.fr/elearning)).
- POINSOT, T., VEYNANTE, D. & CANDEL, S. 1991 Quenching processes and premixed turbulent combustion diagrams. *J. Fluid Mech.* **228**, 561–605.
- POPE, S. 1988 The evolution of surfaces in turbulence. *Int. J. Engng. Sci.* **26**, 445–469.
- RONNEBERGER, O., FISCHER, P. & BROX, T. 2015 U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer.
- SCHÖNFELD, T. & RUDGYARD, M. 1999 Steady and unsteady flows simulations using the hybrid flow solver avbp. *AIAA Journal* **37**, 1378–1385.
- SELLE, L., LARTIGUE, G., POINSOT, T., KOCH, R., SCHILDMACHER, K.-U., KREBS, W., PRADE, B., KAUFMANN, P. & VEYNANTE, D. 2004 Compressible large-eddy simulation of turbulent combustion in complex geometry on unstructured meshes. *Combust. Flame* **137**, 489–505.
- SILVER, D., SCHRITTWIESER, J., SIMONYAN, K., ANTONOGLIOU, I., HUANG, A., GUEZ, A., HUBERT, T., BAKER, L., LAI, M., BOLTON, A. *et al.* 2017 Mastering the game of go without human knowledge. *Nature* **550**, 354.
- SMAGORINSKY, J. 1963 General circulation experiments with the primitive equations: 1. the basic experiment. *Mon. Weather Rev.* **91**, 99–164.
- VOLLANT, A., BALARAC, G. & CORRE, C. 2017 Subgrid-scale scalar flux modelling based on optimal estimation theory and machine-learning procedures. *Journal of Turbulence* pp. 1–25.
- WANG, G., BOILEAU, M. & VEYNANTE, D. 2011 Implementation of a dynamic thickened flame model for large eddy simulations of turbulent premixed combustion. *Combust. Flame* **158**, 2199–2213.