Available online at www.sciencedirect.com

ScienceDirect

Proceedings of the Combustion Institute 39 (2023) 5289–5298

Proceedings of the Combustion Institute

www.elsevier.com/locate/proci

Applying physics-informed enhanced super-resolution generative adversarial networks to turbulent premixed combustion and engine-like flame kernel direct numerical simulation data

Mathis Bode [a,b,*], Michael Gauding [b], Dominik Goeb [b], Tobias Falkenstein [b], Heinz Pitsch [b]

[a] *Jülich Supercomputing Centre, Forschungszentrum Jülich GmbH, Germany*
[b] *Institute for Combustion Technology, RWTH Aachen University, Germany*

Received 6 January 2022; accepted 11 July 2022
Available online 23 February 2023

Abstract

Models for finite-rate-chemistry in underresolved flows still pose one of the main challenges for predictive simulations of complex configurations. The problem gets even more challenging if turbulence is involved. This work advances the recently developed PIESRGAN modeling approach to turbulent premixed combustion. For that, the physical information processed by the network and considered in the loss function are adjusted, the training process is smoothed, and especially effects from density changes are considered. The resulting model provides good results for a priori and a posteriori tests on direct numerical simulation data of a fully turbulent premixed flame kernel. The limits of the modeling approach are discussed. Finally, the model is employed to compute further realizations of the premixed flame kernel, which are analyzed with a scale-sensitive framework regarding their cycle-to-cycle variations. The work shows that the data-driven PIESRGAN subfilter model can very accurately reproduce direct numerical simulation data on much coarser meshes, which is hardly possible with classical subfilter models, and enables studying statistical processes more efficiently due to the smaller computing cost.

Keywords: Generative adversarial network; Direct numerical simulation; Large-eddy simulation; Premixed combustion; Engine

1. Introduction

The availability of larger and larger, often extensively labeled datasets, either from large scale experiments, social media interactions, or simulations, has massively boosted the development of

* Corresponding author at: Jülich Supercomputing Centre, Forschungszentrum Jülich GmbH, Germany.
E-mail address: m.bode@itv.rwth-aachen.de (M. Bode).

https://doi.org/10.1016/j.proci.2022.07.254

algorithms, software, and hardware to efficiently use data. For example, modern graphics processing units (GPUs) feature more than 100 TeraFLOPS FP16 performance, and state-of-the-art algorithms and implementations can almost fully employ this potential. As a consequence, data-driven methods have evolved as an important tool in many areas, enabling new use cases not possible a few years ago [1–3]. They also emerged as useful tools, such as machine learning (ML) and deep learning (DL), in fluid dynamics and reactive flows applications, e.g., for replacing simpler algebraic closure models in Reynolds-averaged Navier–Stokes (RANS) simulations and large-eddy simulations (LESs) [4] and for efficiently storing complex data [5,6].

Technically, the ML and DL approaches rely on continuously updating network weights in a data-fed training process to minimize loss functions. A particular combination of two DL networks, generator and discriminator, coupled by an adversarial loss term, is called a generative adversarial network (GAN) [7]. GANs aim to estimate the unknown probability density of observed data without an explicitly provided data likelihood function. These generative models implicitly learn by requiring access to only data samples from the unknown distribution. Therefore, the GANs learning of unknown data probability distribution is unsupervised and does not need any labels necessary in supervised learning scenarios. Finally, the trained generator network is used for prediction.

Bode et al. [8–11] developed physics-informed enhanced super-resolution generative adversarial networks (PIESRGANs) by advancing ESRGANs (enhanced super-resolution GANs) [12] to handle 3-D physical data and add physical information in the training process. Furthermore, they performed an ablation study with turbulent data to find a suitable network architecture for physical problems. The main idea of their model algorithm is to use high-fidelity data ("H"), e.g., from a fully resolved direct numerical simulation (DNS), and a corresponding filter operation to get pairs of high-fidelity data and filtered data ("F"). These pairs are used to train PIESRGAN to recover the high-fidelity data as accurately as possible based on the filtered data only. The generated data are called reconstructed data ("R"). Bode [13] recently extended this framework to laminar reactive flows with finite-rate-chemistry modeling by suggesting to solve additional transport equations on the reconstructed data and introducing an integrated lookup table approach to significantly reduce the time to the solution for parametric variations and the computational cost. They demonstrated the usability of this framework for a laminar lean premixed combustion case and received good results for emission prediction. However, Bode [13] did not use PIESRGAN on turbulent finite-rate-chemistry scalar fields. To fully focus on chemistry, they did not even directly train the network with the velocity fields. More details including further relevant literature can be found in previous works [10,13–15]. The PIESRGAN is extended to turbulent finite-rate-chemistry data in this work for the first time based on the example of an engine-like premixed flame kernel.

Cycle-to-cycle variations (CCVs) in spark-ignition (SI) engines are still not fully understood, as many different in-cylinder processes are involved, interact complexly, and include a large range of spatial and temporal scales. However, CCVs are known to be affected by early flame kernel development [16,17]; faster early flame kernel growth correlates with faster combustion cycles. 3-D DNSs of early flame kernel development are possible and very well-suited to derive more insight into the complex phenomena involved [18–22]. One problem though are the costs of such simulations if they are run with finite-rate-chemistry and sufficiently high turbulence levels. As turbulence is a stochastic phenomenon, multiple realizations of flame kernels need to be run to accurately reproduce CCV behavior. Therefore, this is a very appealing application for complex data-driven model development such as PIESRGAN. Particularly, the reconstruction approach allows direct evaluation of the filtered density function (FDF) instead of relying on filtered-probability-functions, which are usually used in conventional models. Additionally, this approach enables the consideration of stochastic deviations of the FDF, which have not received much attention in the engine context yet. Accurate simulation data exist for training on one hand, but on the other, the cost per simulation needs reduction to create a statistically representative database for further analysis.

In this work, an iso-octane/air premixed flame kernel configuration, representing the underlying physical processes in real engines described above, is chosen as a first target case for the development of PIERSGAN for reactive turbulent finite-rate-chemistry flows.

2. Case setup of a turbulent premixed flame kernel

Falkenstein et al. [18–20] computed a database of different iso-octane/air flame kernels, including fully turbulent kernels and planar cases, as well as cases with unity Lewis number and constant Lewis numbers. They used a uniform mesh with 960 grid points per direction to discretize a periodic box, and a reaction mechanism with 26 species. The box was initialized with homogeneous isotropic turbulence (HIT), which decays over time and mimics the conditions in SI engines. The unburnt temperature is $T_u = 600$ K, the initial pressure $p^0 = 6$ bar, and the air-fuel equivalence ratio $\phi = 1.0$. Consequently, the cases were designed to lie in the thin reaction zone combustion regime. All were computed in the low-Mach limit using the Curtiss–

Hirschfelder approximation [23] for diffusive scalar transport and including the Soret effect.

To simplify the analysis, Falkenstein et al. [18] introduced a simplified reaction progress variable ζ with the thermal diffusion coefficient D_{th} as the diffusion coefficient reading

$$\partial_t(\rho\zeta) + \nabla \cdot (\rho \boldsymbol{u}\zeta) = \nabla \cdot (\rho D_{\mathrm{th}} \nabla(\zeta)) + \dot{\omega}_\zeta \tag{1}$$

with bold notation for vectors, ∇ as del operator, and ∂_t as time derivative. ρ is the density, $\boldsymbol{u}$ the velocity vector, and $\dot{\omega}_\zeta$ the chemical source term of the simplified reaction progress variable, which is defined as

$$\dot{\omega}_\zeta = \dot{\omega}_{\mathrm{H_2}} + \dot{\omega}_{\mathrm{H_2O}} + \dot{\omega}_{\mathrm{CO}} + \dot{\omega}_{\mathrm{CO_2}}. \tag{2}$$

To better understand the setup and its temporal development, 2-D slices through the center of the box are presented in Fig. 1 for ζ, $\dot{\omega}_\zeta$, and a velocity component u_1 for the case with unity Lewis numbers at three different time steps. As anticipated, over time, the kernel grows as shown by the larger area with higher ζ-values and wrinkles more. Most of the reactions occur in the thin reaction zone. The turbulent structures indicated by the velocity component grow with time, implying a decrease of turbulent intensity in the box. Note that most of the analysis of this case in this work will be performed on later time steps than are presented here. These typically offer better statistical convergence due to a larger surface area.

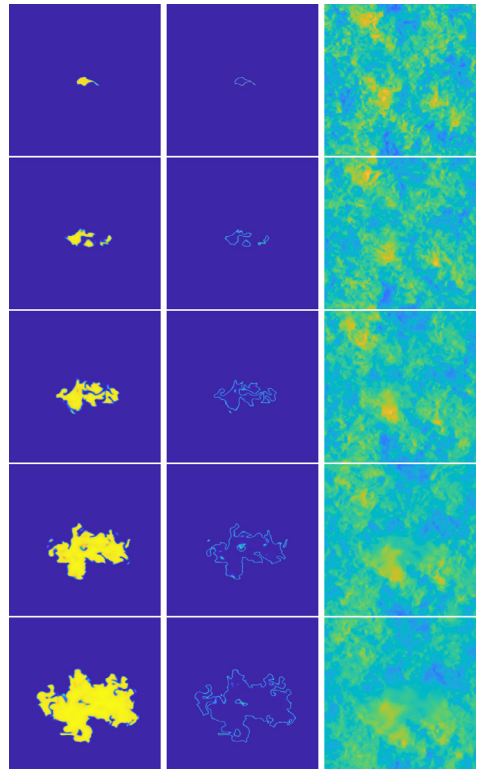

Fig. 1. Visualization of 2-D slices of ζ, $\dot{\omega}_\zeta$, and u_1 (left to right) for five different increasing time steps (top to bottom: 0.60 $\times 10^{-4}$ s, 1.35×10^{-4} s, 2.10×10^{-4} s, 2.85×10^{-4} s, 3.60×10^{-4} s) for the fully turbulent flame kernel with unity Lewis number. Note that the flame kernel does not break into parts at the latest time shown. A coherent flame kernel topology was maintained at all times.

3. PIESRGAN for turbulent premixed combustion

The PIESRGAN approach for finite rate chemistry by Bode [13], except for the velocity and turbulence treatment, was mainly followed in this work. The turbulent case considered here is much more challenging, as the chaotic nature of turbulence makes the training more difficult and the data much bigger, requiring more advanced computing approaches. Comparisons are possible only as ensemble statistics, which complicates the analysis. It includes the reconstruction of fields, a consecutive solution step of the resolved transport equations on the finer mesh, the evaluation of the terms unclosed in the filtered equations on the coarser mesh, and the advancement of the filtered equations. This algorithm is described in detail in Bode [13]. Moreover, following their suggestion, the species are split into primary and secondary in this work. Transport equations are solved only for the primary species, while the secondary species are advanced by a chemtable-like lookup. One important reason is the reduced cost for the PIESRGAN with split species, which is thus called reduced PIESRGAN and denoted as PIESRGAN$_{\mathrm{S}}$. A PIESRGAN without secondary species is called a full PIESRGAN. Details about the different aspects of PIESRGAN and ways to deal with the turbulence are offered in the next subsections.

3.1. Loss function

The loss function is the target function that is minimized by training the GAN. Bode [13] decoupled velocity and scalar fields to train their network for laminar partially premixed combustion. This approach is not possible here, as the fluctuations in the velocity field are expected to significantly impact all phenomena in the process. Thus, the loss function by Bode [13] needs to be complemented by a continuity loss term. However, this term cannot be similar to the corresponding term defined by Bode et al. [10], relying on a divergence-free condition for incompressible flows, which is obviously violated here. Consequently, the loss function for PIESRGAN for turbulent premixed combustion is chosen as

$$\mathcal{L} = \beta_1 L_{\mathrm{adversarial}} + \beta_2 L_{\mathrm{pixel}} + \beta_3 L_{\mathrm{gradient}} + \beta_4 L_{\mathrm{continuity}} + \beta_5 L_{\mathrm{species}}, \tag{3}$$

where β_1, β_2, β_3, β_4, and β_5 are scalar coefficients weighting the different loss term contributions; in

this work, these coefficients were always equally scaled such that the sum of all non-zero weighting coefficients equalled one. Note that all loss terms are non-dimensional, as all operators and input fields used are non-dimensionalized. The velocity field was zero mean-centered and rescaled with its root-mean-square deviation (RMSD) value. All loss terms are evaluated over all the cells of the domain. $L_{\text{adversarial}}$ is the discriminator/generator relativistic adversarial loss [1], which mainly communicates the discriminator feedback to the generator. The pixel loss L_{pixel} and the gradient loss L_{gradient} are defined as the mean-squared error (MSE) of the quantity itself and of the gradient of the quantity, respectively [8]. If the MSE operator is applied on tensors, including vectors such as the velocity, it is applied to all components separately. Afterward, the resulting tensor is mapped into a scalar using the L_1-norm. $L_{\text{continuity}}$ is the physics-informed continuity loss, enforcing physically plausible solutions of the reconstructed flow field in which a compressible continuity equation should be fulfilled, $\partial_t(\rho) + \nabla \cdot (\rho \boldsymbol{u}) = 0$. This again contrasts the original loss function by Bode et al. [10] for HIT in which any dilatation and expansion effects were neglected. This introduces an additional complexity to the training process as two consecutive time steps were used to evaluate the temporal derivative. To increase the numerical accuracy, a centered derivative for time was employed, technically introducing an offset between the current filtered data and DNS data during the training process, as training has to wait for the availability of the next DNS data in time. L_{species} is the loss term ensuring that the sum of all mass fractions in the domain equals one.

The physics-informed loss terms, $L_{\text{continuity}}$ and L_{species}, were found essential for using PIESRGAN as the subfilter model. In the trained network, these terms must be very close to zero to prevent the a posteriori simulation from blowing up.

3.2. Architecture

The training process and architecture of PIESRGAN is shown in Fig. 2. Both network parts, the generator and the discriminator, heavily utilize 3-D CNN layers (Conv3D) [24], which are activated by leaky rectified linear unit (LeakyReLU) layers. The central component of the generator is the residual in residual dense block (RRDB). It contains residual dense blocks (RDBs) with skip-connections, which are extended residual blocks (RBs) with dense connections inside. A residual scaling factor β_{RSF} is used to prevent instabilities in the forward and backward propagation. The generator has about 80 layers in total. On the other hand, the discriminator has only about 28 layers. However, it features layers for batch normalization (BN) as well as dropout with dropout factor β_{dropout}. Its final layer is a dense layer (Dense). The training is with pairs of data. The high-fidelity data ("H") are filtered in a prestep to receive filtered data ("F"), which serve as input to the network. The high-fidelity data are considered to evaluate the loss function terms.

3.3. Species splitting

For the turbulent premixed flame kernel application case in this work, a full PIESRGAN was trained first, which was then reduced by introducing primary and secondary species. Primary species are reconstructed on a fine mesh, and an unfiltered transport equation is solved to update the reconstructed field. The source term is evaluated on the fine mesh. On the other hand, secondary species are only reconstructed, including the source term. The accuracy is increased by shifting the primary and secondary species by a half-time step. The reduction happened using AutoML techniques [25]. Simply put, species are systematically moved from primary to secondary, the effect is evaluated, and an optimal reduced PIESRGAN found.

In contrast to the findings by Bode [13], C_8H_{18}, OH, H_2, CO, CO_2, and H_2O were found to be the most suitable selection of primary species here. This is especially interesting for H_2, which was a secondary species for Bode [13].

3.4. Algorithm

PIESRGAN is used to reconstruct "fully-resolved" data from filtered data. Therefore, the outputs of the GAN are refined data fields used consecutively to evaluate source terms and advance the filtered equations. The LES subfilter algorithm for the primary species starts with the LES solution without secondary species Φ^n_{LES} at time step n and consists of repeating the following steps:

1. Use the PIESRGAN to reconstruct Φ^n_{R} from Φ^n_{LES}.
2. Use Φ^n_{R} to update the primary species fields of Φ to $\Phi^{n;\text{update}}_{\text{R}}$ by evaluating the source terms and solving the unfiltered scalar equations on the mesh of Φ^n_{R}.
3. Use $\Phi^{n;\text{update}}_{\text{R}}$ to estimate the unclosed terms Ψ^n_{LES} in the LES equations of Φ by evaluating the local terms with $\Phi^{n;\text{update}}_{\text{R}}$ and applying a filter operator.
4. Use Ψ^n_{LES} and Φ^n_{LES} to advance the LES equations of Φ to Φ^{n+1}_{LES}.

For the secondary species and velocities, the second step is skipped. Any source terms are integrated in the network.

3.5. Implementation and training details

The training process is one of the main challenges for GANs. Depending on how it is done, e.g., how the network is initialized and how the learning

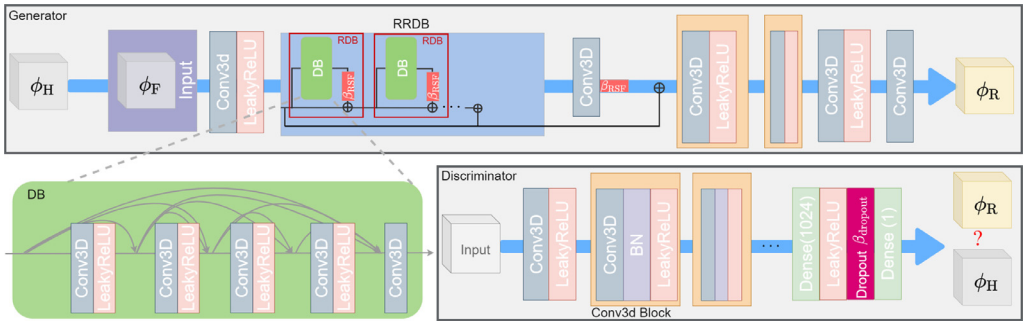

Fig. 2. Sketch of PIESRGAN. "H" denotes high-fidelity data, such as DNS data, "F" are corresponding filtered data, and "R" are the reconstructed data. The components are as follows: Conv3D - 3D Convolutional Layer, LeakyReLU - Activation Function, DB - Dense Block, RDB - Residual Dense Block, RRDB - Residual in Residual Dense Block, β_{RSF} - Residual Scaling Factor, BN - Batch Normalization, Dense - Fully Connected Layer, Dropout - Regularization Component, and β_{dropout} - Dropout Factor. Image from Bode et al. [10].

rate is varied, the training might lead to a converging result or a diverging system. In image applications, such as ESRGAN, a perceptual loss based on the VGG-feature space, which is pre-trained with, e.g., the ImageNet dataset, is often used to initialize the network coefficients and smoothen the training process. A similar approach was followed in this work. The fully trained network by Bode et al. [10] based on decaying HIT was used as a starting network to accelerate the training process.

The training was performed with multiple filter widths, using box filters for simplicity. The filter stencil width varied from 5 to 15 cells per direction. Furthermore, the training was performed on the fly to efficiently use compute nodes with GPUs, i.e., the Falkenstein et al. [18] configuration was rerun, and the obtained data was used for training without storing it permanently. This also minimized the probability of overfitting. Moreover, this always allowed comparison between training and test data. Two newly run DNSs were used for on-the-fly training and the accuracy of the prediction compared to the original DNS data by Falkenstein et al. [18].

The premixed flame kernel simulations were all run on a uniform mesh. Therefore, effects of non-uniform meshes were not further considered in this work. However, the presented method is not limited to uniform meshes. Bode [26] showed in the context of non-premixed flames that training with multiple filter widths can compensate mesh effects.

The numerical solver and network parameters were chosen by Bode et al. [10], equally weighting β_4 and β_5.

3.6. A priori testing

The accuracy of the PIESRGAN$_{\mathrm{S}}$ model for turbulent premixed combustion super-resolution is first evaluated based on an a priori test, and the results are shown in Fig. 3 for three species, a velocity component, and the simplified reaction

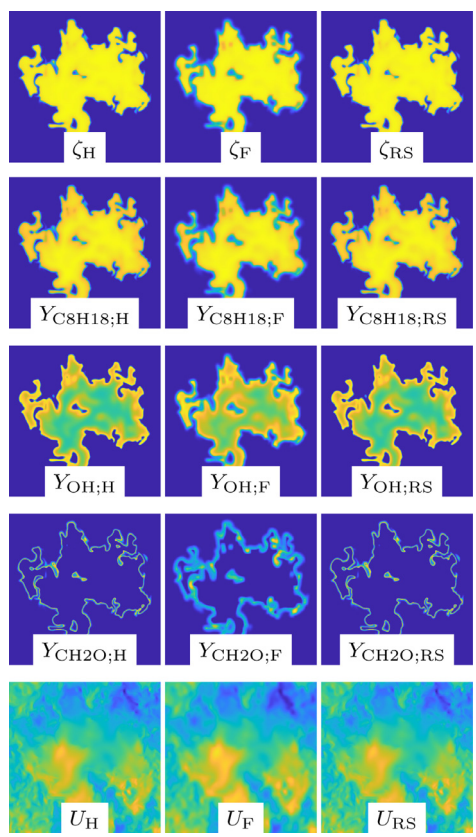

Fig. 3. Visualization of DNS, filtered, and reconstructed fields for the unity Lewis number case employing PIESRGAN$_{\mathrm{S}}$. Results for the simplified reaction progress variable, the C_8H_{18} mass fraction, the OH mass fraction, the CH_2O mass fraction, and a velocity component are shown. Note that the images show the same time step as the last row in Fig. 1 but are zoomed in. Furthermore, all images show 2-D slices of the full 3-D data.

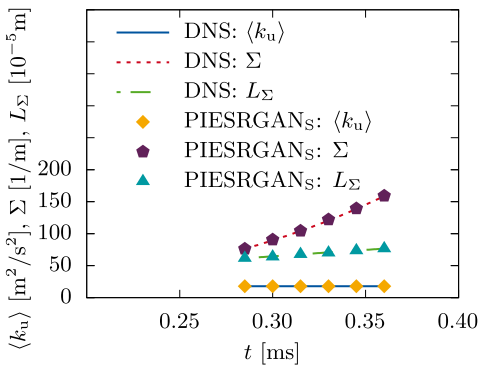

Fig. 4. Temporal evolution of the averaged turbulent kinetic energy in the unburnt mixture $\langle k_u \rangle$, the surface density Σ, and the characteristic length scale L_Σ for the DNS and PIESRGAN$_S$ for the unity Lewis number case.

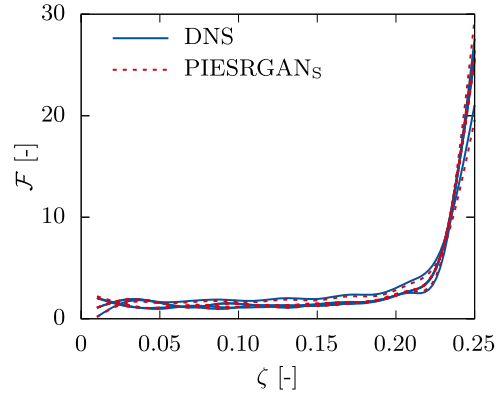

Fig. 5. FDFs $\mathcal{F}$ for DNS data and PIESRGAN$_S$ data in five different locations with the same filtered simplified reaction progress variable value.

progress variable. The reconstruction results of PIESRGAN$_S$ are denoted with "RS". The a priori test used the maximum filter stencil width of 15 cells per direction, while the ratio of cell size and Kolmogorov length on the fine mesh is about 0.85 at the considered time step. The agreement between the DNS data and the reconstructed data is very good. The filtered data are less sharp as they feature less small-scale structures, and it can be expected that running the simulation directly on such a coarse grid will result in unsatisfactory results. Note that while C_8H_{18} and OH are primary species, CH_2O is a secondary species and also shows good accuracy.

3.7. A posteriori testing

To assess the accuracy of the trained PIESRGAN$_S$ model, an a posteriori test has been conducted. For that, and to avoid different initial conditions, one of the already developed solutions of the DNS case with unity Lewis numbers, which was not used for training, was advanced in time by an LES employing the PIESRGAN$_S$ model. A constant filter width of five cells per direction was chosen, it was maintained that all gradients were numerically sufficiently resolved in the transportation step on the coarse mesh, and multiple domain-averaged quantities (denoted by $\langle \cdot \rangle$) were evaluated. Note that even for this relatively weak filtering, simulations without model were not running stably. In Fig. 4, the evolutions of the average turbulent kinetic energy in the unburnt mixture $\langle k_u \rangle$, the surface density Σ, and the characteristic length scale L_Σ (both defined in the application section) are shown. Satisfactory agreement between DNS and PIESRGAN$_S$-LES can be observed.

It can be concluded that the PIERSGAN$_S$ model can produce LES results that very closely mimic DNS results but on a much coarser grid. Particularly, only few DNSs, covering the parameter-bounding box, are required to train the initial network, and subsequent realizations can be generated using the much cheaper PIESRGAN$_S$-LES.

3.8. FDF analysis

In addition to the previously shown a posteriori test, another a posteriori test is performed. The target time step and model are the same, but a later start time is chosen, and the run time is only one fifth of the previous a posteriori test. The filter width was increased to eleven cells per direction to enable statistically better computation of FDFs (denoted $\mathcal{F}$) with respect to the simplified reaction progress variable. FDFs are typically replaced by presumed PDFs in classical LES models. For example, a presumed PDF constructed by the filtered simplified reaction progress variable can be used to model the subfilter distribution, e.g., for evaluating the chemical source term. A consequence is that cells featuring the same value for the filtered simplified reaction progress variable (and potentially additional parameters) always have the same subfilter distribution, i.e., cannot account for stochastic deviations. In Fig. 5, FDFs for DNS data and PIESRGAN$_S$ data in five different locations are compared. All selected locations feature the same filtered simplified reaction progress variable, which was evaluated as the progress variable value with maximum heat release. Deviations between the FDFs of different locations are obvious. This is interesting, and its impact on CCVs should be analyzed more systematically in the future. Another important result is that the agreement of the FDFs computed on the DNS data and the PIESRGAN$_S$ data is good. This result again emphasizes the accuracy of the PIESRGAN$_S$ model and reveals the advantages of the field reconstruction

compared to simplified models based on presumed PDF closures.

3.9. Full versus reduced PIESRGAN

The results so far obtained in this work but also the laminar results by Bode [13] provide the impression that the full and the reduced PIESRGAN approaches offer equally good results most of the time. This result is obviously not the case for all possible variations. For example, a reduced PIESRGAN trained with the unity Lewis numbers flame kernel case clearly overpredicts the evolution of the flame kernel case with constant, but non-unity Lewis numbers. Due to the learnt evolution of the secondary species on the reconstructed grid, the effect of non-unity Lewis numbers resulting in smaller source terms is not sufficiently considered. This effect can be seen even though a Lewis number effect is indirectly fed into the network through the primary species, for which the transport equations on the fine mesh include the Lewis number. The results for the full PIESRGAN model are much better, but still not perfect in this case. Overall, the PIERSGAN$_S$ approach to utilize direct network lookup for secondary species seems to lead to very good intra-case capabilities of the network and is computationally advantageous. However, the approach is more case-specific than a full model solving transport equations for all species.

4. Application

Overall, a priori and a posteriori results provide strong confidence that the developed PIESRGAN$_S$-LES model is able to accurately and quickly compute turbulent flame kernel realizations, even on smaller computing clusters as long as GPUs are available. Therefore, in this section, six new independent realizations of early flame kernel development with an 8-times lower resolution per direction than the original DNS were computed, and their CCVs are analyzed in this section. It is not possible to easily predict the speed-up due to different clusters used featuring CPUs and GPUs as well as partly strongly varying compute cost per time step. However, all six PIESRGAN$_S$-LESs together were slightly cheaper in terms of mixed FLOPS than one fully resolved DNS realization. Generally, the speed-up increases for cases with even more species and is further discussed by Bode [27]. The six flame kernel realizations featuring different initial locations in the domain and therefore different turbulence interactions, are used in this section to gain more insight into early flame kernel development from a physical and modeling point of view. Obviously, six additional kernel realizations are not sufficient for a full statistical analysis to understand the role of early flame kernel development in CCVs. However, due to brevity and the modeling focus of this paper, this number should be sufficient to demonstrate the usage of the developed model in a scale-sensitive analysis framework. Therefore, the discussion is also only limited to a single time step, ($t = 0.33$ ms) even though the simulations were run over time. The analysis is performed on the reconstructed fields, featuring the same resolution as the original DNS.

Assuming the laminar flamelet concept, the geometrical properties of a flame kernel are required to accurately predict its macroscopic features, such as flame speed and heat release. A key quantity is the flame surface density Σ, which determines the reaction rate [28]. Due to the interaction with the turbulent flow, the flame surface exhibits a wide range of different scales, extending from the integral scales, which contain information about initial and boundary conditions, down to the Kolmogorov length scale, which is the dissipative cut-off scale for which a quasi-universal statistical theory exists [29].

Given the progress variable ζ that varies in space $\boldsymbol{x}$, a threshold ζ_0 can be used to define an interface that separates the regions where $\zeta(\boldsymbol{x}) > \zeta_0$ from the regions where $\zeta(\boldsymbol{x}) < \zeta_0$. Consequently, for a given threshold ζ_0, a phase indicator function $\Gamma(\boldsymbol{x}, t)$ can be defined as $\Gamma(\boldsymbol{x}, t) = \mathcal{H}(\zeta(\boldsymbol{x}, t) - \zeta_0)$, with $\mathcal{H}$ being the Heaviside step function. To proceed, the structure function of the phase indicator function, i.e.,

$$\langle(\delta\Gamma)^2\rangle(\boldsymbol{r}, t) = \langle(\Gamma(\boldsymbol{x} + \boldsymbol{r}, t) - \Gamma(\boldsymbol{x}, t))^2\rangle \tag{4}$$

with $\boldsymbol{r}$ as spatial distance vector, is introduced and finally computed by taking an angular average to facilitate a scale sensitive analysis of the geometry of the surface of the flame kernel.

The connection between the indicator structure function and the morphology of the flame kernel surface can be demonstrated by taking the small-scale and large-scale limits. Kirste and Parod [30] proved that for interfaces of class $\mathcal{C}^2$, the small-scale limit of Eq. (4) is given by

$$\lim_{r\to 0}\langle(\delta\Gamma)^2\rangle = \frac{\Sigma r}{2} + \mathcal{O}(r^2). \tag{5}$$

Here, the surface density is given by

$$\Sigma = \langle|\nabla\Gamma|\rangle \tag{6}$$

and quantifies the area of the iso-scalar surface of the flame divided by the volume V. Note that the computation of the surface density by Eq. (4) is computationally very efficient and does not require the tessellation of the iso-surface or the definition of a level set function. In the large-scale limit, $\langle(\delta\Gamma)^2\rangle$ tends to

$$\lim_{r\to\infty}\langle(\delta\Gamma)^2\rangle = 2\langle\Gamma\rangle(1 - \langle\Gamma\rangle), \tag{7}$$

and is hence related to the volume $\langle\Gamma\rangle$ enclosed by the iso-surface. If a sufficient scale separation exists, Eq. (4) also provides information about the fractal dimension of the flame kernel surface [31].

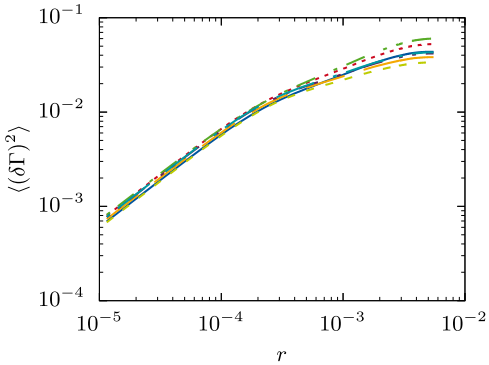

Fig. 6. Plot of the non-normalized phase indicator structure function for six realizations of the unity Lewis number case. Additionally, the non-normalized phase indicator structure function evaluated on the DNS data is shown (solid blue line).

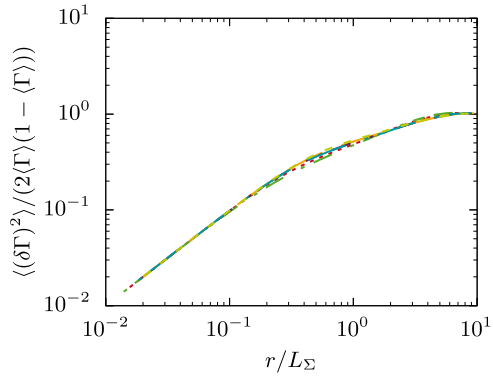

Fig. 7. Plot of the normalized phase indicator structure function for six realizations of the unity Lewis number case.

Furthermore, expressing the surface density by a characteristic length scale is customary

$$L_\Sigma = \frac{4\langle\Gamma\rangle(1-\langle\Gamma\rangle)}{\Sigma}, \tag{8}$$

which is related to the wrinkling scale $L_\Sigma^* = (4\Sigma_{\max})^{-1}$ of premixed flames, which falls between the Taylor microscale and the Kolmogorov scale as recently demonstrated by Kulkarni et al. [32].

The discussed phase indicator structure function is an easily computable method to evaluate important multi-scale geometrical aspects of a flame kernel and to understand sensitivities for CCVs. For example, the surface density values evaluated from the six new realizations at the considered time step are 138.7 m^{-1}, 141.7 m^{-1}, 127.7 m^{-1}, 134.0 m^{-1}, 134.9 m^{-1}, and 117.7 m^{-1}. The four largest values seem to be reasonably close together and the "standard" evolution, while the two smallest values deviate. Both deviating cases are slower in their evolution compared to the others, and thus, the smaller surface densities represent delayed flame kernels. This fact is also reflected in Fig. 6, which shows the non-normalized phase indicator structure function in log-log presentation. Additionally, Fig. 6 shows the phase indicator structure function evaluated with the DNS data. It nicely aligns itself with the PIESRGAN$_S$-LES computed realizations.

From a modeling perspective, knowing the characteristic scales of the phase indicator structure function leading to self-similarity is important. According to Eqs. (7) and (8), L_Σ and $2\langle\Gamma\rangle(1-\langle\Gamma\rangle)$ are suitable candidates for scaling the phase indicator structure function. The result is depicted in Fig. 7, which features excellent collapse at the small and large scales. At the intermediate scales, the influence of CCVs is clearly visible as variations of the flame kernel morphology.

5. Conclusions

This work shows the application of PIESRGAN-based modeling to turbulent finite-rate-chemistry flows for the first time. The model for laminar finite-rate-chemistry flows described in Bode [13] is extended with special focus on the correct consideration of the fluctuating velocity fields and the density change, causing an expansion and deviating the turbulence from HIT. Both the full and the reduced PIESRGAN models present good results in a priori and a posteriori tests on the DNS data of a fully turbulent premixed flame kernel case under engine conditions. The limits of the reduced model were discussed. The differences between FDFs evaluated from the DNS or the reconstructed field from PIERSGAN and typical PDF closures were highlighted.

The data-driven model allows to compute more realizations of the fully turbulent premixed flame kernel for significantly lower cost. Six new realizations were computed for the unity Lewis number case for the cost of roughly one additional DNS realization and analyzed with respect to their CCV by means of a scale-sensitive framework. The level of variation was evaluated, and it was, furthermore, shown that the structure functions collapse sufficiently with appropriate scaling. This is an important conclusion from a modeling point of view as it helps develop simpler models connecting the flame surface density and heat release.

Overall, this paper demonstrates that LES based on PIESRGAN subfilter modeling can very accurately predict complex turbulent finite-rate-chemistry flows. PIESRGAN probably outperforms all classical LES models.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relation-

ships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors acknowledge computing time grants for the projects JHPC55 and TurbulenceSL by the JARA-HPC Vergabegremium provided on the JARA-HPC Partition part of the supercomputer JURECA at Jülich Supercomputing Centre, Forschungszentrum Jülich, the Gauss Centre for Supercomputing e.V. (www.gauss-centre.eu) for funding this project by providing computing time on the GCS Supercomputer JUWELS at Jülich Supercomputing Centre (JSC), and funding from the European Union's Horizon 2020 research and innovation program under the Center of Excellence in Combustion (CoEC) project, grant agreement no. 952181.

References

[1] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, C. Loy, ESRGAN: enhanced super-resolution generative adversarial networks, Lect. Notes Comput. Sci. 11133 (2019) 63–79.
[2] A. Bhati, S. Wan, D. Alfe, A. Clyde, et al., Pandemic drugs at pandemic speed: infrastructure for accelerating COVID-19 drug discovery with hybrid machine learning- and physics-based simulations on high performance computers, Interface Focus 11 (2021) 20210018.
[3] O. Vinyals, I. Babuschkin, W. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, et al., Grandmaster level in StarCraft II using multi-agent reinforcement learning, Nature 575 (2019) 350–354.
[4] K. Fukami, Y. Nabae, K. Kawai, K. Fukagata, Synthetic turbulent inflow generator using machine learning, Phys. Rev. Fluids 4 (6) (2019) 064603.
[5] M. Ihme, C. Schmitt, H. Pitsch, Optimal artificial neural networks and tabulation methods for chemistry representation in LES of a bluff-body swirl-stabilized flame, Proc. Combust. Inst. 32 (2009) 1527–1535.
[6] M. Bode, N. Collier, F. Bisetti, H. Pitsch, Adaptive chemistry lookup tables for combustion simulations using optimal B-spline interpolants, Combust. Theory Model. 23 (4) (2019) 674–699.
[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680.
[8] M. Bode, M. Gauding, K. Kleinheinz, H. Pitsch, Deep learning at scale for subgrid modeling in turbulent flows: regression and reconstruction, LNCS 11887 (2019) 541–560.
[9] M. Bode, et al., Development of physics-informed enhanced super-resolution generative adversarial networks for subfilter modeling, arXiv preprint (2021a).
[10] M. Bode, M. Gauding, Z. Lian, D. Denker, M. Davidovic, K. Kleinheinz, et al., Using physics-informed enhanced super-resolution generative adversarial networks for subfilter modeling in turbulent reactive flows, Proc. Combust. Inst. 38 (2021) 2617–2625.
[11] M. Bode, AI Super-Resolution: Application to Turbulence and Combustion, in: N. Swaminathan, A. Parente (Eds.), Machine Learning and Its Application to Reacting Flows, Lect. Notes Energy, 44, 2023.
[12] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, C. Change Loy, ESRGAN: enhanced super-resolution generative adversarial networks, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018.
[13] M. Bode, Applying physics-informed enhanced super-resolution generative adversarial networks to finite-rate-chemistry flows and predicting lean premixed gas turbine combustors, arXiv preprint arXiv: 2210.16219 (2022).
[14] M. Bode, Applying Physics-Informed Enhanced Super-Resolution Generative Adversarial Networks to Large-Eddy Simulations of ECN Spray C, SAE Int. J. Adv. Curr. Pract. Mobil. 4 (6) (2022) 2211–2219.
[15] M. Bode, Applying physics-informed enhanced super-resolution generative adversarial networks to direct numerical simulation data of spray and comparison to classical flamelet models, arXiv preprint (2022).
[16] W. Zeng, S. Keum, T.-W. Kuo, V. Sick, Role of large scale flow features on cycle-to-cycle variations of spark-ignited flame-initiation and its transition to turbulent combustion, Proc. Combust. Inst. 37 (2019) 4945–4953.
[17] P. Schiffmann, D. Reuss, V. Sick, Empirical investigation of spark-ignited flame-initiation cycle-to-cycle variability in a homogeneous charge reciprocating engine, Int. J. Engine Res. 19 (2018) 491–508.
[18] T. Falkenstein, S. Kang, L. Cai, M. Bode, H. Pitsch, DNS study of the global heat release rate during early flame kernel development under engine conditions, Combust. Flame 213 (2020) 455–466.
[19] T. Falkenstein, A. Rezchikova, R. Langer, M. Bode, S. Kang, H. Pitsch, The role of differential diffusion during early flame kernel development under engine conditions—Part I: analysis of the heat-release-rate response, Combust. Flame 221 (2020) 502–515.
[20] T. Falkenstein, H. Chu, M. Bode, S. Kang, H. Pitsch, The role of differential diffusion during early flame kernel development under engine conditions—Part II: effect of flame structure and geometry, Combust. Flame 221 (2020) 516–529.
[21] M. Klein, N. Chakraborty, K. Jenkins, R. Cant, Effects of initial radius on the propagation of premixed flame kernels in a turbulent environment, Phys. Fluids 18 (2006) 055102.
[22] H. Uranakara, S. Chaudhuri, K. Lakshmisha, On the extinction of igniting kernels in near-isotropic turbulence, Proc. Combust. Inst. 36 (2017) 1793–1800.
[23] J. Hirschfelder, C. Curtiss, R. Bird, M. Mayer, Molecular theory of gases and liquids, 1964.
[24] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, 2012, pp. 1097–1105.
[25] F. Hutter, L. Kotthoff, J. Vanschoren (Eds.), Automated Machine Learning - Methods, Systems, Challenges, Springer, 2019.

[26] M. Bode, Applying physics-informed enhanced super-resolution generative adversarial networks to turbulent non-premixed combustion on non-uniform meshes and demonstration of an accelerated simulation workflow. arXiv preprint 2210.16248 (2022).
[27] M. Bode, Accelerating complex simulations with AI super-resolution-based subfilter modeling, arXiv preprint (2022).
[28] K. Bray, P.A. Libby, J. Moss, Unified modeling approach for premixed turbulent combustion—Part I: general formulation, Combust. Flame 61 (1) (1985) 87–102.
[29] U. Frisch, A. Kolmogorov, Turbulence: The Legacy of AN Kolmogorov, Cambridge University Press, 1995.
[30] R. Kirste, G. Porod, Röntgenkleinwinkelstreuung an kolloiden systemen asymptotisches verhalten der streukurven, Kolloid-Zeitschrift Z. Polym. 184 (1) (1962) 1–7.
[31] J.H. Elsas, A.S. Szalay, C. Meneveau, Geometry and scaling laws of excursion and iso-sets of enstrophy and dissipation in isotropic turbulence, J. Turbul. 19 (4) (2018) 297–321.
[32] T. Kulkarni, R. Buttay, M.H. Kasbaoui, A. Attili, F. Bisetti, Reynolds number scaling of burning rates in spherical turbulent premixed flames, J. Fluid Mech. 906 (2021).