Coaxial-Injector Surrogate Modeling Based on Reynolds-Averaged Navier–Stokes Simulations Using Deep Learning

Moritz Krügener, Jose Felix Zapata Usandivaras, Michaël Bauerheim, Annafederica Urbano

Coaxial-Injector Surrogate Modeling based on Reynolds-Averaged Navier-Stokes Simulations using Deep Learning

M. Krügener *, J. F. Zapata Usandivaras †, M. Bauerheim ‡, A. Urbano§
ISAE-SUPAERO, University of Toulouse, France

Facing the need to increase the accuracy of rocket engines design tools, the present work introduces an innovative methodology for the design and optimisation of rocket engine combustion chambers using numerical simulations and deep learning. An experimental test case of a single coaxial injector is taken as a reference point and a design of experiments is generated by varying nine parameters (geometrical and operative conditions). Reynolds-Averaged Navier-Stokes simulations are carried out to generate the dataset. The data are used to train surrogate models of different fidelity, from low dimensional outputs (0D and 1D) towards the 2D temperature field. Attention is given on the selection of the proper machine learning technique. For low dimensional outputs, results show that deep neural networks outperform other standard machine learning tools, namely Radial Basis Function and Kriging. Regarding high-dimensional outputs, convolutional neural networks with gradient-based loss functions are found effective to capture the large and smooth temperature variations, as well as the thin and sharp temperature gradients at the flame front. Eventually the models are used in the framework of an optimisation problem. Results highlight the benefits of new design and optimisation tools based on deep learning, capable of real-time predictions of complex flow fields.

Nomenclature

ρ	=	density [kg/m^3]
p	=	pressure [Pa]
T	=	temperature [K]
u_i	=	velocity field component along the x_i axis [m/s]
λ	=	thermal conductivity [W/m/K]

*Master Thesis Student, Technical University of Munich/CSE, Boltzmannstr. 3, 85748 Garching, Germany/moritz.kruegener@gmail.com
†Ph.D. candidate, ISAE-SUPAERO/DCAS, 10 Avenue Edouard Belin, 31055 Toulouse, France/jose.zapata-usandivaras@isae-supaero.fr
‡Associate Professor, ISAE-SUPAERO/DAEP, 10 Avenue Edouard Belin, 31055 Toulouse, France/michael.bauerheim@isae-supaero.fr
§Associate Professor, ISAE-SUPAERO/DCAS, 10 Avenue Edouard Belin, 31055 Toulouse, France/annafederica.urbano@isae-supaero.fr

a	=	speed of sound [m/s]
c_p	=	specific heat at constant pressure [J/kg/K]
h	=	specific total enthalpy [J/kg]
h_j^0	=	formation enthalpy of species j [J/kg]
γ	=	specific heat ratio
Γ	=	function of γ
τ_{ij}	=	fluid shear stress tensor [Pa]
μ	=	dynamic viscosity [Pa.s]
k	=	turbulent kinetic energy [J/kg]
ϵ	=	kinetic energy dissipation rate [J/kg/s]
P_k	=	production of turbulent kinetic energy [W]
Pr	=	Prandtl number
M	=	Mach number
Le	=	Lewis number
Sc	=	Schmidt number
$\mathcal{Z}$	=	mixture fraction
Y	=	species mass fraction
$P(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''^2})$	=	β-PDF function
D	=	mass diffusion coefficient [m^2/s]
d	=	radius or height [m]
l	=	length [m]
A	=	cross section area [m^2]
$\dot{m}$	=	mass flow rate [kg/s]
O/F	=	oxydizer to fuel mass ratio
q_w	=	wall heat flux [W/m^2]
Q_w	=	integrated wall heat flux [W]
Δp	=	relative total pressure drop [%]
L_{fl}	=	flame length [m]
$\lvert V_E \rvert$	=	velocity magnitude at the nozzle exit [m/s]
I_{sp}	=	specific impulse [s]
F	=	thrust [N]
c_F	=	thurst coefficient

c^*	=	characteristic velocity [m/s]
x	=	vector of inputs, coordinate
y	=	vector of outputs
N	=	number of samples in a dataset
n_x, n_y	=	number of inputs and outputs
ε	=	mean relative error of a model
η	=	surrogate model accuracy
w_R, H	=	radial basis function parameters and coefficients
$\beta_0, \beta_1, \theta_K$	=	Kriging model parameters
δ_F	=	uncertainty on the target thrust F_t
f, g, G	=	generic functions
W	=	molar mass [kg/mol]
X_p	=	Pareto solutions
J	=	momentum flux ratio
Subscripts		
t	=	turbulent quantity, throat, tip
c	=	chamber
e	=	noozle exit
w	=	wall
o	=	oxydizer
f	=	fuel
in	=	injector
a	=	atmospheric
ref	=	reference
max	=	maximum
i, j, k	=	indexes
ap	=	approximate value
Upscript		
$\hat{.}$	=	prediction
t	=	target

I. Introduction

Liquid Rocket Engines (LRE) are complex systems that are composed of several sub-systems (cooling system, feed system, thrust chamber...) that are designed to be reliable and optimised in order to obtain the highest performance and the lowest mass. The process is iterative, and in the preliminary sizing phase, heavily relies on low order semi empirical correlations calibrated on existing engines. These low fidelity models induce high uncertainties in the subsystems designs that propagate and affect the global engine sizing. As a consequence, the engine development strongly relies on experimental studies, that go up to hot-tests of the engine, making use of very expensive experimental facilities. Increasing the numerical modelling reliability is a strategic matter that needs to be addressed in order to reduce the development cost of innovative rocket engines for future launchers, in particular LRE, and thus ensuring access to space. The combustion chamber and injector design are key elements of any rocket engine [1]. High fidelity numerical simulations, and in particular Large Eddy Simulations (LES), have demonstrated their ability to finely describe complex physical phenomena in LRE combustion chambers [2–5]. Nevertheless, LES is still a high computational cost methodology, with long computational times, which can limit its use in the context of concrete applications. An alternative, when only stationary operating conditions are of interest, is to use Reynolds-Averaged Navier-Stokes (RANS) simulations [6–10]. However, even restitution times of RANS are too high if used in an optimisation framework loop of a whole engine. What is the best way to exploit the capability of Computational Fluid Dynamics (CFD) in this context is an interesting question which is under investigation today[11, 12]. In order to answer this question, a strategy is proposed in the present paper using models of various fidelity. The main idea is to build surrogate models with real-time inference capabilities for the injector and the combustion chamber, using machine learning algorithms and data from numerical simulations. The obtained surrogate models can then be used in the framework of an optimisation problem [13, 14]. Once deployed, the surrogate model will allow a large number of different configurations to be spanned, providing the results of the CFD simulations with a very low inference time, of the order of the millisecond. In the context of design and optimisation of the whole rocket engine, the surrogate model of the chamber can permit to increase the accuracy of the final configuration performance. This eventually can reduce the number of experimental tests and consequently the whole development cost. The derivation and use of data driven surrogate models in the aerospace community is an active field of research. In particular, response surface models have been applied in a wide range of applications including nozzle shape optimization [15] and rocket injector design [16, 17]. In these works, surrogate models are either interpolators, usually high polynomials surface functions, or regressors, usually based on gaussian processes. A survey of these methods can be find for instance in the paper from Hwang and Martins[18]. Recently, the great developments of artificial intelligence algorithms have been recognised in the fluid mechanics community and their application to CFD is a growing field of investigation [19]. In particular, deep learning (DL) has shown a great potential for both modelling and analysing of complex systems, including 2D and 3D geometries [20, 21], flow control and optimisation [14]. The use of deep learning for regression on CFD data, in order to obtain data based

models, has been investigated in the framework of both steady external aerodynamics [14], unsteady fluid mechanics [22] , acoustics [23], and internal reactive flows problems [24, 25]. Of particular interest is the work by Thuerey et al [26], who investigated the accuracy of deep learning for the inference of the flow around an airfoil. The developed surrogate models, based on U-net, a particular class of Convolutional Neural Networks, were able to reproduce the pressure and velocities fields with errors smaller than 3%. Despite the increasing number of works in this domain, the use of these approaches applied to rocket engines remains limited [24, 25, 27, 28] and to the authors knowledge there are no work available focusing on their application for injector design.

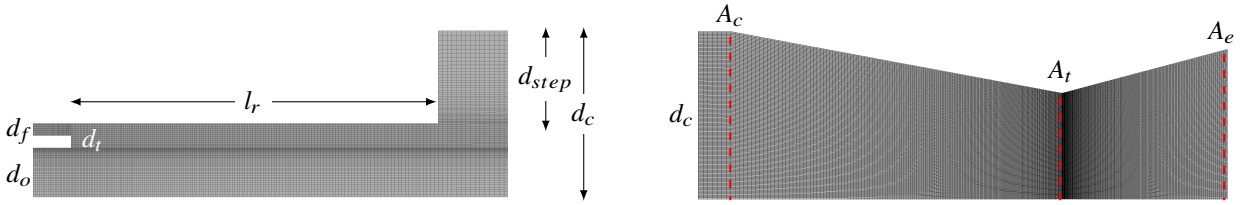
The aim of the present paper is to carry out a proof of concept of the proposed methodology based on machine learning by applying it on a simple configuration. The selected test case is a single coaxial injector chamber, using gaseous methane and oxygen as propellants. The database will be generated using 2D axisymmetric RANS simulations. Specific geometric parameters (for instance the chamber diameter and injector diameters) and specific operative conditions are defined as design parameters. The objective is to build surrogate models able to provide, with a very low inference time, three different types of quantities: performance parameters (0D), the wall heat flux distribution (1D) and a visualisation of the chamber temperature field (2D). In order to achieve this objective several research questions will be addressed. First, the paper will investigate how to deploy a strategy in order to build a design of experiments from the selected case. The second question addresses the type and architecture of the machine learning tool necessary, depending on the size of the target quantity of interest (0D, 1D and 2D). Eventually, the deployment of the developed models in the context of an optimization problem will be tested.

In the following, the experimental test case used as a reference point for the present investigation is presented in Section II. The numerical setup is introduced and the ability in reproducing the results in agreement with the experimental data on the target configuration is demonstrated. Then, the design of experiments is introduced (Section III) and the methodology is deployed to generate the data (Section IV). The data are then used to train different algorithms depending on the fidelity level of interest: global quantities (0D), wall quantities (1D) or field quantities (2D). Sensitivity analyses are carried out to establish the best set of hyperparameters and loss function. Finally, an example of application of the developed models in the framework of an optimisation problem is shown in Section VI.

II. Reference test case: coaxial single injector

The database generated and analysed in the present work is based on a reference experimental test case [29]. It corresponds to a single coaxial injector chamber, operated at 20 bar, with gaseous methane and gaseous oxygen. The gaseous oxygen enters the square chamber of side 12 mm through the inner tube of the injector that has a radius of $d_o = 2$ mm. The gaseous methane flows through the annulus around the oxygen, that has a height of $d_f = 0.5$ mm. RANS simulations are carried out with the commercial software *Ansys Fluent* with the objective of retrieving the stationary condition. As a first step, a numerical study has been carried out in order to find a numerical configuration

that reproduces the experimental data at the reference point. This includes a mesh independence study, turbulence model sensitivity analysis and a comparison between several chemical schemes and turbulent combustion interaction models. The test case is largely documented and has been used as a benchmarck for several numerical studies, including RANS simulations [4, 30]. In the following sections are given the details about the governing equations, the numerical setup and grid selected to carry out the present study.



(a) Fuel and oxidizer inlet detail with recessed post tip.

(b) Refined nozzle and outlet detail.

Fig. 1 Parametrized geometry and meshes close to the injector tip (a) and nozzle (b). Note that the recess length is zero in the reference experiment ($l_r = 0$).

A. Governing equations

The compressible, reactive, turbulent flow is described with the following conservation equations for mass, momentum and energy:

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_j}}{\partial x_j} = 0, \tag{1}$$

$$\frac{\partial \overline{\rho}\widetilde{u_i}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_i}\widetilde{u_j}}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_j} + \frac{\partial \tau_{ij}}{\partial x_j}, \tag{2}$$

$$\frac{\partial \overline{\rho}\widetilde{h}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_i}\widetilde{h}}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\frac{\lambda + \lambda_t}{c_p}\frac{\partial \widetilde{h}}{\partial x_i}\right), \tag{3}$$

where ρ is the density, p is the pressure, u_i the velocity field component along the x_i axis, λ and λ_t are laminar and turbulent conductivities and c_p is the specific heat at constant pressure. h is the specific total enthalpy (in J/kg) which is given by:

$$\tilde{h} = \frac{\tilde{u_i}\tilde{u_i}}{2} + \sum_j \left[\int_{T_{ref}}^{\tilde{T}} c_{p,j}dT + h_j^0(T_{ref,j})\right] \tag{4}$$

where h_j^0 is the formation enthalphy of species j at the reference temperature $T_{ref,j}$. The Favre formalism has been used and for every flow variable f, $\overline{f}$ is an ensemble average and $\tilde{f}$ is a mass weighted average [31, 32]. τ_{ij} is the stress tensor which is given by the following expression:

$$\tau_{ij} = (\mu + \mu_t)\left[\frac{\partial \widetilde{u_j}}{\partial x_i} + \frac{\partial \widetilde{u_i}}{\partial x_j} - \frac{2}{3}\delta_{ij}\frac{\partial \widetilde{u_k}}{\partial x_k}\right], \tag{5}$$

where μ is the laminar dynamic viscosity of the mixture and μ_t is the turbulent viscosity. Thermophysical models are needed in order to close the system. The perfect gas equation is considered. NASA polynomials [33] are employed for the specific heat at constant pressure c_p with an ideal gas mixture law. Laminar conductivity and dynamic viscosity for the mixture are considered constant and equal respectively to $\lambda = 0.0454\text{W m}^{-1}\text{K}^{-1}$ and $\mu = 17.2\,\mu\text{Pa s}$. For what concerns turbulence closure modelling, the $k-\epsilon$ model [34] is used and therefore transport equations for the turbulent kinetic energy k and its rate of dissipation ϵ have to be solved in addition to the system of Eqs. (1)-(3):

$$\frac{\partial \overline{\rho} k}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_i} k}{\partial x_i} = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_i}\right] + P_k + \overline{\rho}\epsilon\left(2\frac{k}{\widetilde{a^2}} - 1\right), \tag{6}$$

$$\frac{\partial \overline{\rho} \epsilon}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_i} \epsilon}{\partial x_i} = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_\epsilon}\right)\frac{\partial \epsilon}{\partial x_i}\right] + C_{1\epsilon}\frac{\epsilon}{k}P_k - C_{2\epsilon}\overline{\rho}\frac{\epsilon^2}{k}, \tag{7}$$

where a is the sound speed. The production term of turbulent kinetic energy P_k is expressed as follow:

$$P_k = \mu_t\left[\frac{\partial \widetilde{u_j}}{\partial x_i} + \frac{\partial \widetilde{u_i}}{\partial x_j}\right]\frac{\partial \widetilde{u_i}}{\partial x_j}. \tag{8}$$

The model constants used are $C_{1\epsilon}$=1.44, $C_{2\epsilon}$=1.92, $\sigma_k = 0.9$ and $\sigma_\epsilon = 0.9$. The turbulent dynamic viscosity can be computed from k and ϵ:

$$\mu_t = \overline{\rho} C_\mu \frac{k^2}{\epsilon}, \tag{9}$$

where $C_\mu = 0.09$ is a constant. Eventually the turbulent conductivity k_t is given by:

$$\lambda_t = \frac{c_p}{\overline{\rho}}\frac{\mu_t}{Pr_t}, \tag{10}$$

where $Pr_t = 0.9$ is the turbulent Prandtl number. The non-adiabatic Steady Diffusion Flamelet (SDF) approach is used to model the multi-species reactive flow [35], making use of a β-PDF to account for turbulence-chemistry interaction. The Flamelet table is generated with the GRI-3.0 mechanism, considering 20 species (H_2, H, CH_4, O_2, H_2O, CO, CO_2, OH, O, C_2H_2, C_2H_4, C_2H_6, HO_2,CH_3, H_2O_2, HCO, CH_2CO, CH_2O, C_3H_8, C_2H_3). The GRI-3.0 mechanism, which was originally built for pressures around 1 atm, has been successfully applied to simulate higher pressure combustion in liquid rocket engines [30, 36]. Therefore, it is considered suitable for the present investigation. A mixture fraction $\mathcal{Z}$ is introduced:

$$\mathcal{Z} = \frac{Y_i - Y_{i,o}}{Y_{i,f} - Y_{i,o}} \tag{11}$$

where Y_i is the mass fraction for the i_{th} species, and the subscript f and o refer to the fuel and oxydizer inlets. Density $\widetilde{\rho}$, temperature $\widetilde{T}$ and species mass fractions $\widetilde{y_i}$ (generically called $\widetilde{\phi}$ in the following) are estimated from the local

average mixture fraction $\widetilde{\mathcal{Z}}$, its variance $\widetilde{\mathcal{Z}'^2}$ and the local average total enthalpy $\widetilde{h}$ (described by (Eq.3)):

$$\widetilde{\phi} = \int_0^1 \phi(\widetilde{\mathcal{Z}}, \widetilde{h}) P(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}'^2}) d\mathcal{Z}\,, \tag{12}$$

where $P(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}'^2})$ is a standard β-PDF function. Therefore, two additional transport equations for $\widetilde{\mathcal{Z}}$ and $\widetilde{\mathcal{Z}'^2}$ are considered:

$$\frac{\partial \overline{\rho}\widetilde{\mathcal{Z}}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_i}\widetilde{\mathcal{Z}}}{\partial x_i} = \frac{\partial}{\partial x_i}\left[\overline{\rho}(D + D_t)\frac{\partial \widetilde{\mathcal{Z}}}{\partial x_i}\right], \tag{13}$$

$$\frac{\partial \overline{\rho}\widetilde{\mathcal{Z}'^2}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_i}\widetilde{\mathcal{Z}'^2}}{\partial x_i} = \frac{\partial}{\partial x_i}\left[\overline{\rho}(D + D_t)\frac{\partial \widetilde{\mathcal{Z}'^2}}{\partial x_i}\right] + C_g \mu_t . \left(\frac{\partial \widetilde{\mathcal{Z}}}{\partial x_i}\right)^2 - C_d \overline{\rho}\frac{\epsilon}{k}\widetilde{\mathcal{Z}'^2}, \tag{14}$$

where $C_g = 2.86$ and $C_d = 2$. D and D_t are laminar and turbulent diffusion coefficients respectively. Considering a unitary number of Lewis, Le=1, the laminar diffusion is expressed as a function of λ and μ:

$$D = \frac{\lambda}{\overline{\rho} c_p}. \tag{15}$$

A turbulent Schmidt number $\mathrm{Sc}_t = 0.5$ is introduced in order to express D_t as a function of μ_t according to:

$$D_t = \frac{\mu_t}{\overline{\rho} Sc_t}. \tag{16}$$

B. Numerical setup

To solve the above governing equations with the *Ansys Fluent* code, a pressure based implicit scheme, accounting for compressibility effects, has been selected to carry out axisymmetric simulations looking for a steady solution. The experimental chamber is rectangular and the nozzle has a rectangular cross section. In order to carry out axisymmetric simulations, the geometry has thus been modified into a cylindrical chamber and a conical nozzle. The radius of the chamber is $d_c = 6.77$ mm , which corresponds to the same cross section area of the experimental square chamber (namely $d_c = \sqrt{A_c/\pi}$ with A_c cross section area of the chamber). A schematic of the computational domain is reported in Fig. 1, with a close-up view of the injector and nozzle regions. Inlet boundary conditions are specified in terms of mass flow rate $\dot{m}$ and temperatures for both propellants, while no boundary condition is applied at the nozzle outlet where the flow is expected to be supersonic. Specifically, oxygen and methane are at 278 K and 269 K during injection respectively, the mass flow rate is $\dot{m} = 0.062$ kg/s and the oxidizer-to-fuel ratio is $(O/F) = 2.62$.This translates to inlet velocities of 122 m/s and 132 m/s for oxygen and methane respectively. The walls of the injector assembly are assumed adiabatic with a no slip boundary condition. A no slip, isothermal boundary condition is applied on the chamber wall,

considering a thermal resistance for a solid wall of copper with a thickness of 1 mm. The outer temperature profile is a linearization of the experimental wall temperature provided in [29] in the same position, as reported in Fig.2.

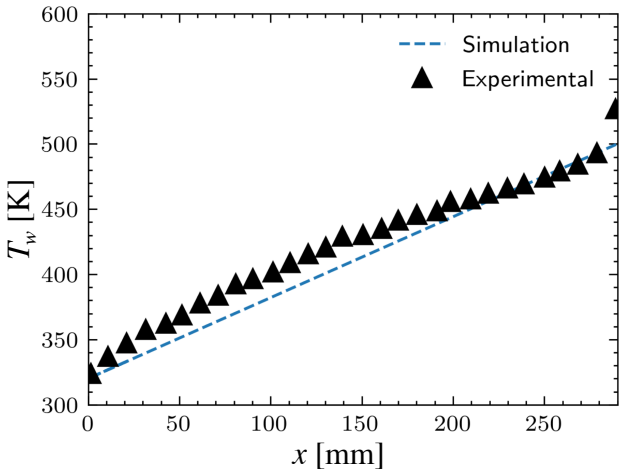


Fig. 2 Experimental wall temperature and linearization imposed as a boundary condition in all the simulations in the outer side of the 1mm thick chamber wall.

C. Grid convergence analysis

The mesh is not uniform and the resolution is increased in the region where stronger gradients are expected. Indeed, a refinement is applied around the area of fuel and oxidizer mixing as can be observed in the detailed view of Fig.1a. A grid sensitivity study was carried out in order to verify the requirements in term of resolution. Fig. 3 shows the wall heat flux profiles obtained with three different grids: coarse (181×10^3 cells), medium (720×10^3) and fine ($2\,770\times10^3$ cells). The ratio of the cells height from one grid to another is 2. Wall heat flux profiles are similar, with differences between coarse and fine meshes smaller than 8%. In Fig.3 are also reported the experimental curve [29] (▲) and the numerical results provided by [30] (- -). The present numerical results are in agreement with experimental data and present the same level of errors obtained with other simulations carried out with similar models in [30]. From a quantitative point of view, the error between the solution and an extrapolated exact solution can be evaluated from the results obtained with the three grids, using the Richardson method [37]. As an example, the error for the integrated wall heat flux Q_w, the average total pressure p_c and temperature T_c at the chamber outlet (nozzle inlet) versus the number of grid cells are reported in log scale in Fig.4. The error decreases by increasing the number of cells, showing a convergence of the results.

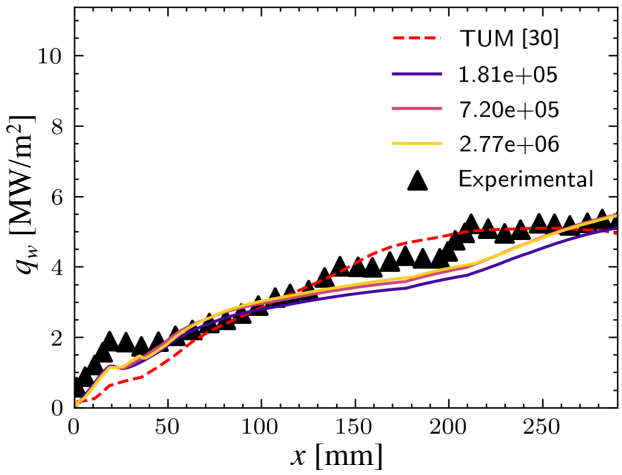


Fig. 3 Wall heat flux profile. Comparison between experimental data [29], numerical results from [30], and the present setup for different meshes.

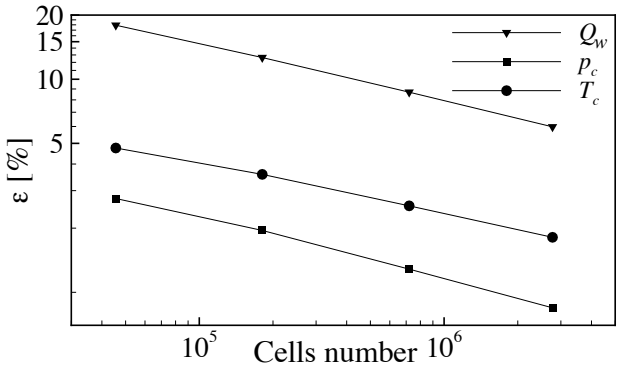


Fig. 4 Error versus number of cells for different grid refinements, for the integrated heat flux Q_w, the stagnation temperature T_c and pressure p_c.

In conclusion the convergence analysis is satisfactory and the obtained results are in agreement with experimental data. Considering that the purpose of the present work is to put in place a methodology by generating a large database, it has been decided to use the coarse mesh (181 000 cells) in order to reduce the computational time for the data generation. All the results presented in the following are therefore carried out with a grid similar to the coarse mesh, with variations according to the geometry parametrisation.

III. Design of experiments and data generation

A. Inputs and outputs

The reference numerical setup detailed in Section II will be used to generate a large database to infer models for injector design optimization. To do so, the first task is to define the inputs and outputs. The outputs are the quantities needed to define the best injector design. Therefore, these outputs are related to both performances and the sizing

of other rocket engine components that depend on the injector design. Moreover, a visualisation of the temperature field inside the chamber can be useful to understand how the design choice impacts the flame and flow topologies. All the outputs are directly deduced from the results of each simulation. 0D outputs, referred as global quantities in the following, are computed with volume averages. 1D quantities are the wall quantities. 2D quantities are field quantities. More information about these outputs are given in the next sections.

Regarding the inputs, seven geometrical parameters are necessary to define the geometry of the single coaxial injector chamber. These parameters, shown in Figure 1 are: the chamber radius d_c, the contraction ratio between the chamber (c) and the throat (t) areas A_c/A_t, the expansion ratio between exit (e) and throat areas A_e/A_t, the oxygen inlet radius d_o, the tip width d_t, the fuel inlet height d_f and the recess length l_r. The only geometrical parameter which is kept constant for all the simulations is the chamber length l_c, which has been increased with respect to the experimental value and is equal to 580 mm. This has been done in order to ensure a combustion length higher than the flame length for all the simulations. Two more inputs are used in order to define the operating point: the total inlet mass flow rate $\dot{m}$ and the oxidizer to fuel mass ratio O/F. The propellants injection temperatures, the wall temperature and the outlet atmospheric pressure (p_a = 1 atm) are kept constants for all the simulations. To summarise, the design of experiments (DOE) will be build varying 9 parameters: 7 geometrical parameters and 2 parameters defining the operating point.

B. High-dimensional sampling

When sampling the 9D parameter space, it is required that design points are well spaced and give a good representation of possible parameters combinations. However, a purely random sampling along each axis does not guarantee a maximized spacing between points. Therefore, Latin Hypercube Sampling (LHS) [38] is used to sample the parameter space. This method takes into account all previously sampled points when determining the next one. Specifically, the Enhanced Stochastic Evolutionary [39] algorithm which is an efficient implementation adhering to the LHS optimally criteria, is used with the Surrogate Modeling Toolbox (SMT) [40].

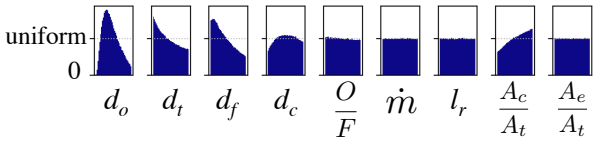


Fig. 5 Histograms (n_{Bin} = 40) **for all the parameters of the Design of Experiments (DOE) along their respective ranges: 1D projections of the 9D parameter space sampling.**

Parameter	Symbol	Range	Reference	Sampling Mode
Oxidizer inlet radius	d_o	0.5 – 4 mm	2 mm	linear in $\sqrt{}$-space
Injector tip width	d_t	0.2 – 1 mm	0.5 mm	linear in $\sqrt{}$-space
Fuel inlet height	d_f	0.2 – 1 mm	0.5 mm	linear in $\sqrt{}$-space
Chamber radius	d_c	4 – 10 mm	6.77 mm	linear in $\sqrt[4]{}$-space
Mixture ratio	O/F	2 – 4	2.62	linear
Total mass flow rate	$\dot{m}$	0.005 – 0.13 kg/s	0.062 kg/s	linear
Recess length	l_r	0 – 30 mm	0 mm	linear
Contraction ratio	A_c/A_t	2 – 5	2.5	linear
Expansion ratio	A_t/A_e	2 – 3	2	linear

Table 1 Parameter range, reference values and sampling mode for all 9 parameters.

Some combinations of input parameters could lead to non-physical configurations (for instance a supersonic flow in the injector). To avoid non physical behaviours, some limits are imposed in terms of minimum of A_c/A_t, d_o, d_c and d_f (see Eqs. (25), (30), (31) and (32) in Appendix VII.A). Note that lower inlet sizes and chamber diameters are more likely to result in violation of these parameter limits. If these parameter ranges are sampled in a linear space, this will result in few design points with a combination of small inlets and chamber sizes. To increase the likelihood of these edge cases being sampled, d_o, d_t and d_f are sampled in $\sqrt{}$-space and d_c is sampled in $\sqrt[4]{}$-space. This implies that samples are taken linearly from functions f, g image spaces such that $f(x) = \sqrt{x}$ and $g(x) = \sqrt[4]{x}$, with x representing the sampled variable. This can be observed in Fig. 5 that shows the histograms of the sampling points for the different parameters. The parameters sampling ranges and the sampling mode are summarised in Tab. 1.

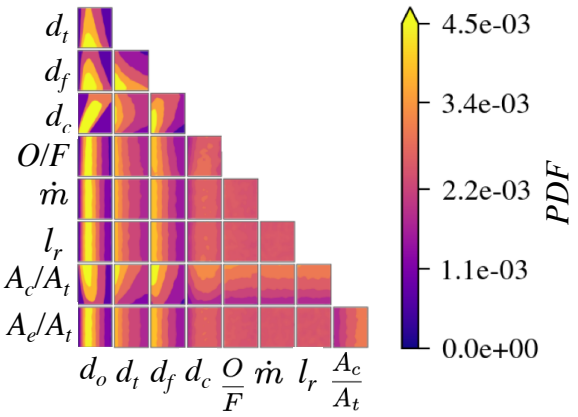


Fig. 6 Probability Density Function isosurfaces for all parameter combinations. x **and** y**-axis limits as in Table 1.**

To further investigate the distribution of the design points, Fig. 6 shows a 2D projection of the probability density function (PDF) for all parameters combinations. The influence of the parameters limits is clearly visible. For example, the third row from the top with d_c on the y-axis illustrates the dependence of Eq.(32) on d_o, d_t and d_f. The distribution of d_c is however also influenced by Eq. (25), which results in a relationship between the contraction ratio and the chamber diameter seen in row A_c/A_t column d_c. The contraction ratio is also influenced by d_o, d_t and d_f via the chamber pressure approximation (Eq. 28). Other parameters like O/F, $\dot{m}$, l_r and A_t/A_e do not feature prominently in the limit equations and therefore do not lead to significant changes in the uniformity of the distribution. This can be observed by looking at their respective rows which feature little to no change in the distribution along the y-axis. In general, if the distribution evolves only in x or y direction, the two parameters have fewer dependence on each other via parameter limits. The more the gradient features a combined x-y direction, the more the two parameters interact in the DOE. Note that this only highlights the correlations between input variables when sampling the database, but is not related to any later correlation in the learnt surrogate model.

IV. Data generation, overview and preparation for training

Following the previous rules, a DOE of 3640 points is considered and for each point a RANS simulation is carried out until convergence using 16 cores on Xeon-6126 processors at 2.6GHz. On average, a run takes 52 minutes (13.87 CPU hours) for a total of 50 000 CPU hours for the whole dataset generation. Over the total 3640 simulations, outliers outside of the 3σ range are removed which result in $N = 3437$ simulations that remain for training. Prior the training phase, a post-processing is necessary to extract outputs of interest from the generated data: these consist in global, wall and field quantities. These are described and analysed in the following sections.

A. Global quantities

Twelve global quantities of interest for the injector design will be computed: the average total pressure at the chamber outlet p_c, the ratio between the total pressure loss and the inlet total pressure in the fuel injector Δp_f and in the oxidizer injector Δp_o, the maximum average temperature along the chamber axis $\overline{T}_{max}$, the flame length L_{fl} estimated as the position of $\overline{T}_{max}$, the maximum wall heat flux $q_{w,max}$, the average wall heat flux $\overline{q_w}$, the velocity magnitude at the nozzle outlet V_E, the thrust F, the thrust coefficient c_F, the characteristic velocity c^* and the specific impulse I_{sp}. An overview of the ranges and distributions of the outputs is given by the histograms of Fig.7. The ranges of these outputs were not known in advance and are reasonable with respect to the parameters ranges. For instance, the means for the chamber pressure p_c and the peak wall heat flux $q_{w,max}$ are $\overline{p}_c = 31.82$bar and $\overline{q}_{w,max} = 7.6\text{MW/m}^2$ respectively, which are well in the same order of magnitude of the reference case ($p_c \approx 20$bar and $q_{w,max} \approx 5.5\text{MW/m}^2$).

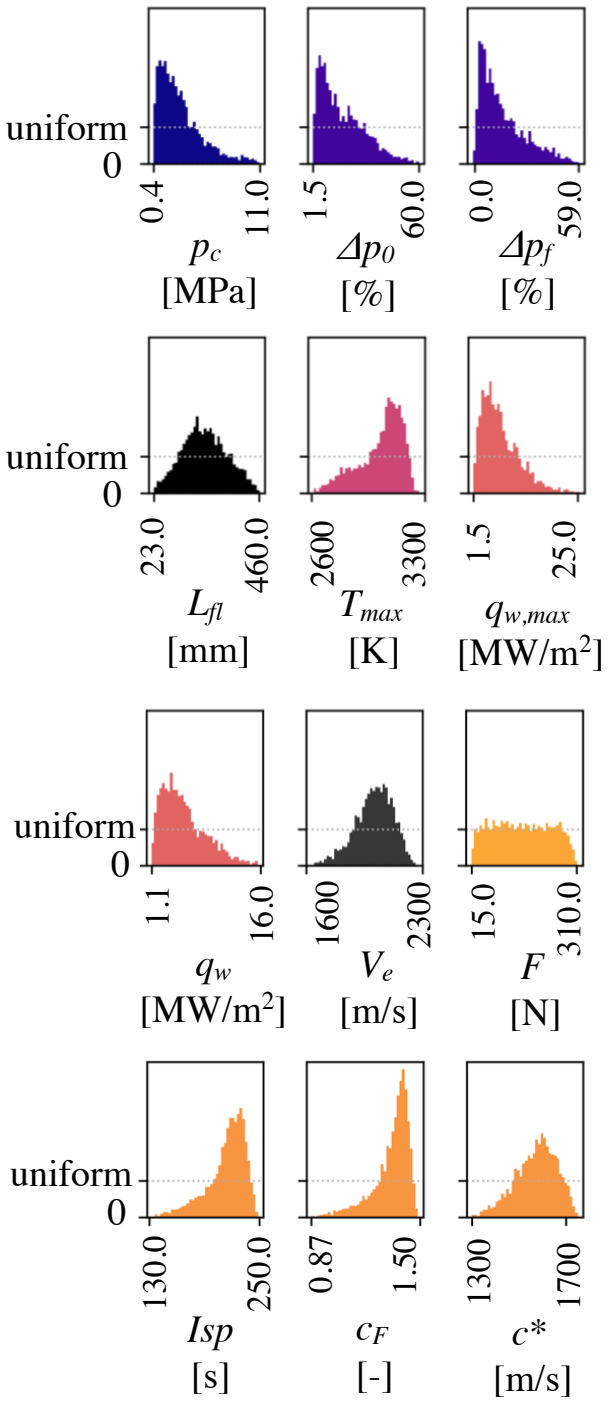


Fig. 7 Histogram of the global quantities for N=3437 simulations. Outliers larger than 3σ have been removed.

B. Wall quantity

The only wall quantity considered is the chamber top wall heat flux distribution q_w . The wall profile is extracted from the simulation and sampled at 1024 equidistant points along the wall, making an interpolation from the computational

grid. For the 3437 available samples, this amounts to a training dataset size of 30MB, which is 2 orders of magnitude larger than the size for global quantities. The heat flux exhibits a peak in the first 5-30 cm of the chamber and then remains almost constant. The distribution of the heat fluxes over the whole DOE is shown in Fig. 8. The minimum and maximum lines are combinations of different simulations results: these are the maximum and minimum heat flux at specific axial positions. The wall heat fluxes in rocket engines combustion chambers constitute a critical, but yet difficult to estimate, quantities during preliminary designs. Knowledge of these and their gradients is critical to the design of the chamber cooling system to ensure safe engine operation.

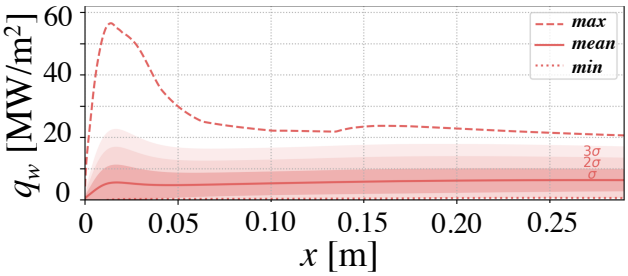


Fig. 8 Wall heat fluxes distributions over the Design of Experiments.

C. Field quantity

The field quantity considered in the present work is the temperature field in the chamber (the nozzle is not considered). The temperature distribution in the chamber serves as a window into the topology of the flame. Locations with large temperature gradients are linked to heat-release areas where chemical reactions are stronger. The axial location of the maximum of the cross section averaged temperature serves as a proxy for the flame length L_{fl}. Smoother gradient regions are linked to flow recirculation or turbulent mixing. These structures can be studied in conjunction with the aforementioned wall heat-flux providing a degree of interpretability during the design process. As an example, in the left part of Fig. 9 are shown three different RANS temperature fields corresponding to three different points of the DOE. The flame topology is strongly affected by the parameters choice. The overall range of temperature is similar and varies between the inflow temperature around 250 K and a maximum temperature between 2600 K and 3300 K.

The field data provided by CFD on the computational non-uniform mesh must be processed into a uniform Cartesian grid with a pixel-like format prior to the training. In fact this is a requirement in order to use convolutional neural networks (CNN). The fields are therefore interpolated on a uniform fixed resolution grid of size 128×128. The fixed number of interpolation points results in different spatial resolutions in the transverse direction of the injector depending on the chamber diameter. In Fig. 9 are reported three examples of temperature fields from three different DOE samples. In the left part, the fields on the grid from the CFD simulation are shown. Note that the transverse y axis has been magnified to allow a better visualisation, being the chamber length always much greater than the diameter. In the right part of the figure are shown the corresponding transformed views of the same fields once interpolated on the 128×128

grid. These are the snapshots that will be used to train the CNN.

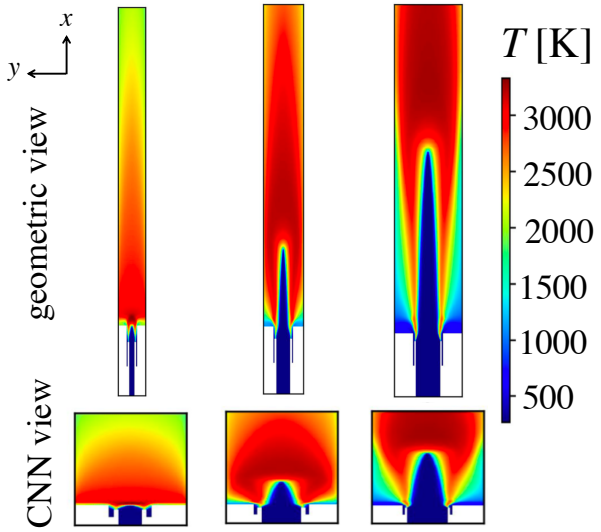


Fig. 9 Temperature field examples. Up: geometric view (y-axis magnified by a factor of 10). Bottom: transformed view used during training.

In addition, a Boolean mask of the geometry is provided at the same resolution of 128×128. Specifically, the Boolean mask is a 128×128 table that assigns *true* values only in cells containing the fluid field, and *false* values in cells outside the fluid domain. The error will be computed only in the masked subset of the chamber and therefore only the cells containing the temperature information will be used for training. Additionally, all 9 scalar DOE parameters are supplied by nine constant 128×128 matrices: for each matrix, all entries are equal to the respective DOE parameter. Both the 9 DOE parameters and the temperature field are scaled to 0 mean and 1 standard deviation.

V. Results

Depending on the dimension of the output expected from the surrogate models (0D global quantities, 1D wall quantities or 2D field quantities) different algorithms will be tested for the training. Data are split in samples for training (2/3) and samples for validation (1/3). Furthermore, 77 samples are held back and never used during training. They will be used as test data in order to prevent overfitting when optimizing the hyper-parameters to the validation data.

A. Global quantities

Global quantities values are the training targets with the lowest complexity in terms of overall dataset size and target dimensions. This translates to low computational cost when training models and the necessity to use simpler models to avoid the possibility of over-fitting. Training data always include all DOE parameters, but only one output target at the time: the models are trained on the 12 post-processing quantities described in Section V. In the following, we will

indicate with $\hat{y} \in \mathbb{R}^{n_y}$ the prediction of a model, which is generically given by:

$$\hat{y} = f(x, x^t, y^t) \tag{17}$$

where $x \in \mathbb{R}^{n_x}$ is the vector of inputs, $x^t \in \mathbb{R}^{n_x \times N}$ the vector of training inputs and $y^t \in \mathbb{R}^{n_y \times N}$ the vector of training outputs, where n_x, n_y and N are respectively the dimensions of the input, output and the number of samples. Therefore, for every global quantity it is $n_x = 12$ and $n_y = 1$. Three different algorithms are tested and compared for the surrogate model f. The first two are classical machine learning methods: Radial Basis Functions (RBF) and Kriging (KRG). For RBF, the following interpolating function f is used, with Gaussian basis functions:

$$f = \sum_{i}^{N} \exp\left(-\frac{\|x - x_i^t\|_2^2}{H^2}\right) w_R\,, \tag{18}$$

where the coefficients vector $w_R \in \mathbb{R}^N$ is evaluated during the training, and the scaling parameter is set to $H = 0.05$. For KRG, Gaussian covariance functions are used together with a linear model for the deterministic linear regression term, thus leading to the following expression for f:

$$f = \beta_0 + \beta_1 x + G(x)\,, \tag{19}$$

where $\beta_0 \in \mathbb{R}$ and $\beta_1 \in \mathbb{R}^{n_x}$. $G(x)$ is the realisation of a stochastic process with zero mean value and a spatial covariance given by a Gaussian function:

$$cov[G(x_i, x_j)] = \sigma^2 \Pi_{k=1}^{n_x} \exp\left[-\theta_k (x_i^k - x_j^k)^2)\right] \tag{20}$$

where σ^2 is the variance of the process $G(x)$ and the n_x hyper parameters θ_k are determined solving an optimization problem in order to obtain a maximum likelihood function. They are all initialised to 0.01. Both RBF and KRG are provided by the Surrogate Modeling Toolbox (SMT) open source package, and the reader can refer to [40] for the details about the algorithms.

In addition, a fully connected neural network (FCNN) is also trained using the *PyTorch* library [41]. A standard approach for regression is used with a ReLU activation function on all layers but the last, which has no activation function [42]. The Adam optimizer is used with the Mean Squared Error Loss (MSE), defined by:

$$MSE(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \tag{21}$$

where y_i is the target output and $\hat{y}_i$ is the network prediction. The quality of a network is evaluated considering its accuracy as a figure of merit, which is related to the mean relative error over the samples of the validation dataset.

Specifically, the relative error for a target variable y_i, for sample i, is defined with the following expression:

$$\varepsilon_i = \frac{|y_i - \hat{y}_i|}{y_i}, \tag{22}$$

and the accuracy η is defined accordingly by:

$$\eta = \left(1 - \frac{1}{N}\sum_{i=1}^{N} \varepsilon_i\right) \times 100. \tag{23}$$

In order to build the suitable FCNN, two tuning studies are carried out. The first one aims at choosing the network architecture. A set of 23 different architectures with depths ranging from 1-4 hidden layers with layer widths between 16-1024 were tested with chamber pressure as target output. Given that preliminary tests had shown a tendency for overfitting even in small networks, all 23 architectures were trained with a weight decay of 1×10^{-4} and a dropout of 0.01. Batch size was set at 64 with total training time of 500 epochs. 10-fold cross validation was used to allow for comparability between runs. Training these setups with 10-fold cross validation took roughly 3.5 hours on a 4-core machine. Given these results, the network architecture with the best combination of accuracy and lowest parameter count was chosen. This network has 2 hidden layers with 64 neurons each, so the overall architecture is 9 - 64 - 64 - 1 with 4865 trainable weights and biases.

Parameter	Learning Rate	LR Decay	Batch Size	Weight Decay	Dropout
Range	0.001 - 0.05	0.5 - 0.99	8 - 128	0 - 0.002	0 - 0.05
Discretization	Uniform	0.01	8	0.0001	0.001
Best setup	0.01214	0.97	32	0.0002	0.0

Table 2 Hyper-parameters, with range and discretization, used during tuning for the FCNN (9-64-64-1) on the global quantities.

The second analysis aims at finding the best hyper-parameters setup. The five hyper-parameters under consideration are the learning rate (LR), LR decay, batch size, weight decay and dropout. A Bayes search optimization approach is applied using the *wandb* sweep API. Three-fold cross validation is used leading to a 2/3-1/3 split between training and validation data per training run. All parameters ranges for tuning are listed in Table 2. Training would run for a maximum of 200 epochs or if the learning rate hits 10^{-10}. In total, 193 different setups were tested and the optimizer set to minimize the relative error on the validation dataset. Eventually, the best setup is summarized in Table 2.

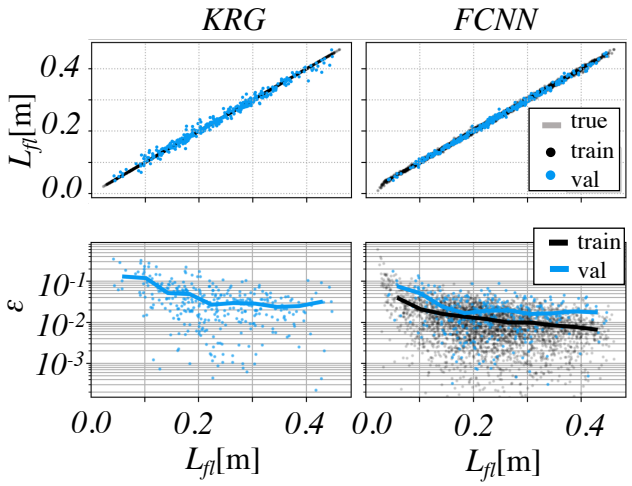
Model \ Quantity	p_c	Δp_o	Δp_f	L_{fl}	$\overline{T}_{max}$	$q_{w,max}$	$\overline{q_w}$	$\lvert V_E \rvert$	F	I_{sp}	c_F	c^*
Neural Network	98.86	**97.51**	**97.79**	**97.53**	**99.83**	**97.50**	98.75	**99.76**	99.54	**99.68**	**99.92**	**99.82**
Kriging (KRG)	**99.42**	95.64	92.51	95.55	99.68	96.64	**99.08**	99.71	**99.69**	99.63	99.90	99.75
Radial Basis Function (RBF)	98.15	94.27	83.76	91.65	99.45	94.45	98.22	99.53	99.46	99.11	99.79	99.65

Table 3 Accuracy η on validation dataset (Eq.(23)) for global quantities models, obtained using 20-fold cross validation. Highest accuracies are highlighted in bold.

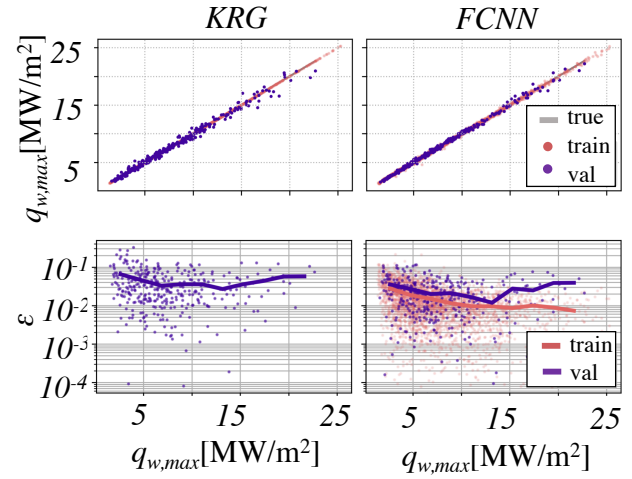
For the three algorithms, training times are negligible compared to computational cost to generate the data. Specifically, training on a 4-core CPU took 1s for RBF, 52 s for KRG and 85 s for FCNN (considering 200 epochs and a 32 batch size).

The results obtained with the selected FCNN are compared against KRG and RBF models. It is worth mentioning that the default hyper-parameters values were chosen for RBF and KRG. To achieve a high confidence in the accuracy values obtained during training, 20-fold cross validation is used and the accuracy η on the validation dataset is used as a figure of merit and reported in Table 3. The best model for each parameter is highlighted in bold. The RBF setup never produces the most accurate model, and the corresponding accuracy is usually within 1 - 2 % lower than the KRG accuracy. Generally, FCNN and KRG accuracies are in the same order of magnitude.

As an example, Fig. 10a shows the predictions for training and validation data of both the KRG and FCNN models side-by-side for L_{fl} and the corresponding relative errors ε. A slightly higher mean accuracy of 97.53% is obtained with the FCNN versus a 95.55% with KRG on the validation dataset (see Table 3). This difference mainly presents itself in the low end of combustion length, with comparably low relative errors for higher values. Training data error for KRG is not shown in the plot, as the KRG error in the training data is 0. Figure 10b shows the predictions for training and validation data of both the KRG and FCNN models side-by-side for $q_{w,max}$. The FCNN again narrowly beats KRG with 97.50% mean accuracy versus 96.64%. The relative error of the FCNN model is consistently lower over the entire range of the target dimension.



(a) Prediction (top) and relative error ε (bottom) versus true value for the flame length L_{fl}.



(b) Prediction (top) and relative error ε (bottom) versus true value for the maximum wall heat flux $q_{w,max}$.

Fig. 10 Top figures: model predictions for L_{fl} and $q_{w,max}$ for training and validation data using KRG and FCNN. Bottom figures: relative error and moving mean.

Finally, KRG and FCNN are compared in terms of inference times. Fig. 11 shows the inference time for a single predicted value, if it is part of a batch inference. On CPU, FCNN is around 10 times faster than KRG. A plateau is reached for batch sizes around 16 for KRG and 256 for FCNN. The fastest inference can be achieved for FCNN using a GTX1060 GPU. Moreover, the same batch inference time is obtained no matter the batch size until 65536.

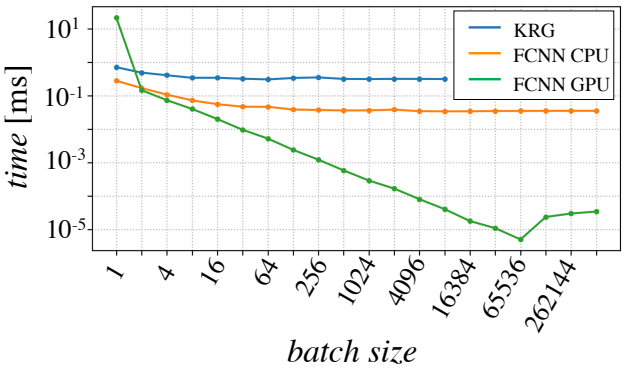


Fig. 11 Inference times for a batch divided by batch size with trained KRG and FCNN models. Mean of 10 runs.

B. 1D quantities: wall heat flux

The FCNN approach was again used to predict the heat flux all along the chamber top wall. The main difference being that the last layer of the network now has 1024 nodes instead of just a single one. Due to the high dimensionality of the output data and the overall data size, KRG and RBF models were no longer pursued. Given the increased network size and gained experience from global model training, cross validation was no longer deployed. The dataset is split 90/10 between training and validation samples with again 77 samples held back as an additional test dataset. A tuning is carried out following the approach used in section V.A. In total, five different architectures are tested. The ranges of parameters are increased compared to the one used for the global quantities, as reported in Table 4. Additionally, the maximum number of training epochs is increased to 1000.

Parameter	Layers	Learning Rate	LR Decay	Batch Size	Decay	Dropout
Range	9 - 256 - 256 - 1024 9 - 64 - 64 - 128 - 1024 9 - 64 - 128 - 256 - 1024 9 - 256 - 64 - 256 - 1024 9 - 128 - 128 - 128 - 1024	0 - 0.1	0.5 - 0.999	16 - 256	0 - 0.02	0 - 0.1
Discretization	Categorical	Uniform	0.001	16	0.0001	0.001

Table 4 Tested architectures and hyper-parameters ranges for training on wall heat flux.

Layers	Weights	LR	LR Decay	Batch	Epochs	Decay	Dropout	train. η	val. η	Test η
9 - 256 - 64 - 256 - 1024	300k	0.015	0.994	176	947	0	0.032	96.60%	96.19%	96.85%
9 - 64 - 128 - 256 - 1024	300k	0.0071	0.986	192	853	0	0.035	96.75%	96.04%	96.80%
9 - 256 - 256 - 1024	330k	0.013	0.867	32	99	0	0.014	96.81%	95.95%	96.93%
9 - 256 - 256 - 1024	330k	0.0051	0.98	208	245	0	0.099	96.66%	95.83%	96.83%

Table 5 Best training setups during hyper-parameters tuning for wall heat flux. Mean accuracy evaluated for the training, validation and test datasests.

The best four networks characteristics are summarised in Table 5. Most of the differences in peak accuracy for an architecture is due to the higher numbers of trainable weights leading to higher accuracy, albeit necessitating the use of higher dropout to prevent over-fitting. The mean accuracy is computed considering an average of Eq.(23) over the wall length. The mean accuracy for training, validation and test data shows consistent values of around 96%. However, this mean accuracy for the entire length of the combustion chamber is not necessarily meaningful if the error occurs in critical locations like peak heating and on sharp gradients along the chamber.

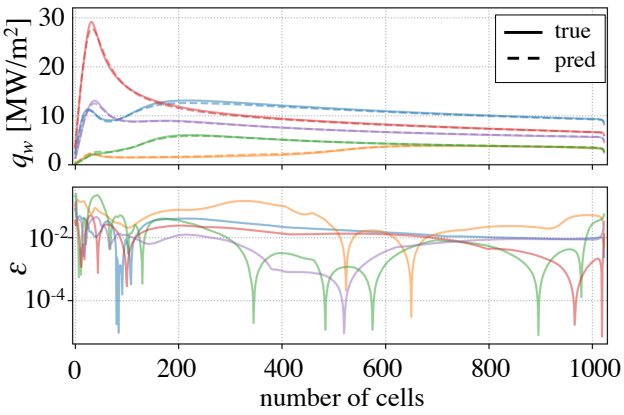


Fig. 12 Wall heat flux predictions (top) and corresponding relative errors (bottom) for five samples of the test dataset.

Figure 12 (top) shows predictions for five randomly chosen samples from the test dataset. These samples have never been part of the training process of the network and are a good representation of the diversity of the dataset. The predictions were obtained using the network with the highest validation accuracy in Table 5. The mean relative accuracy of 96.85% on the entire test dataset is comparable with the accuracy on the train and validation datasets, leading to the assumption that the network is able to generalize and has not overfit to training data or even to validation data during hyper-parameters tuning. The samples predictions show good accuracy on the overall shape of the wall heat flux curve along the chamber wall, as well as in capturing the sharp gradients and peaks in the first part of the chamber. The

highest relative error (Fig 12 bottom) occurs at the point of lowest wall heat flux at the chamber inlet. This behaviour is expected, since it corresponds to the zone with the highest gradients and lowest absolute values. The valleys in the relative error plot in Fig.12 indicate crossings were prediction and true curve intersect. Each sample exhibits at least three such crossings, indicating the prediction oscillates around the true curve instead of e.g. exhibiting a constant offset.

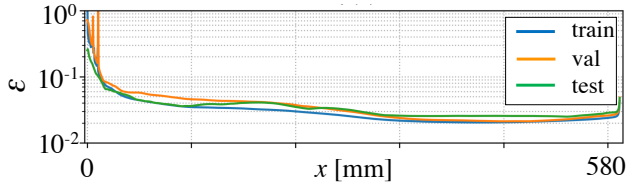


Fig. 13 Mean Relative Error on the wall heat flux prediction along the chamber axis, for the three dataset (training, validation and test).

Quantity	Training		Validation		Test	
	Mean Rel. Error [%]	Std. [%]	Mean Rel. Error [%]	Std. [%]	Mean Rel. Error [%]	Std. [%]
$q_{w,max}$	2.35	2.53	2.75	2.99	2.91	3.05
$\overline{q_w}$	1.05	1.35	1.60	2.94	1.44	1.11

Table 6 Mean Relative Error and Standard Deviation for peak ($q_{w,max}$) and average heat flux ($\overline{q_w}$) across different datasets (training, validation and test).

Fig. 13 illustrates the mean error distribution across all datasets and along the chamber. There is a strong correlation of the error distributions between all datasets. In agreement with the samples shown in Fig. 12, the larger mean relative errors occur early at the chamber inlet. The error quickly decreases thereafter and reaches lowest values in the last quarter of the chamber. Note that the deep learning models could be trained specifically to reduce this error if needed, for example by including the error on gradients into the loss function. Such a strategy is proposed in Section V.C, where it is proved effective to capture the sharp gradients of the temperature fields induced by the thin flame front.

Table 6 shows the mean relative error for the peak wall heat flux $q_{w,max}$ and the average wall heat flux $\overline{q_w}$ as calculated from the wall heat flux curve. They translate to a mean accuracy $\overline{\eta}$ on validation data of 97% and 98% respectively. These physical quantities are a good descriptors of the quality of the surrogate model, as they quantify the accuracy on typical quantities of interest. Comparing them to the accuracies obtained by the global modelling approach (reported in Table 3), both values are between 0.5% and 1% worse than the dedicated global model.

The 1D wall heat flux model under scrutiny was trained with a MSE loss function without any pondering of such quantities as $q_{w,max}$ and $\overline{q_w}$. In spite of this, the derived quantities from the model 1D predictions show to be in good agreement with those of the evaluated datasets. Recovering these quantities without specifically accounting for them in the loss function suggests the model is capturing the essential structures of the wall heat flux.

C. 2D quantities: temperature field

The entire temperature field in the injector and combustion chamber is learnt to provide in real-time a view of the complete field. A 2D field will be easier to interpret in terms of flame anchoring and shape, compared with 0D and 1D quantities. Therefore the output is a $128 \times 128 = 16384$ cells field. This 2D regression task was tackled with two variants of Convolutional Neural Networks (CNN). The first one is a simple encoder-decoder network using only convolutional layers. The second is a U-Net [43] architecture including batch normalization, up-sampling layers and skip connections. The use of CNN on 2D spacial data has been studied extensively in the context of image processing and has recently been extended to be used on simulation tasks [14, 22, 23, 26]. The setup of both architectures are displayed in Table 7. The channel and kernel sizing was partly based on networks in Thuerey et al. [26] and downscaled to fewer channels. The number of trainable weights is in the same order of magnitude as for the wall quantity learning task using FCNN networks. Learning rate and decay were also selected based on previously used values for this type of network [26]. Both architectures were trained using the same dataset of 3200 training, 170 validation and 77 test fields. The uncompressed dataset size is 2.3GB with 10 128x128 inputs and 1 target image per simulation.

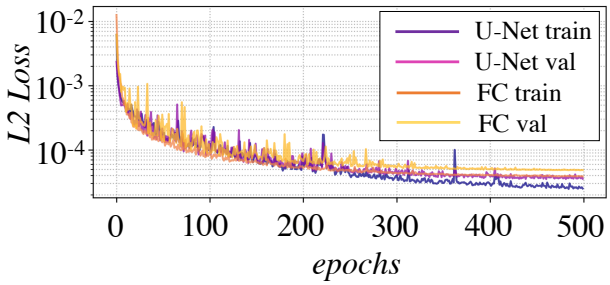


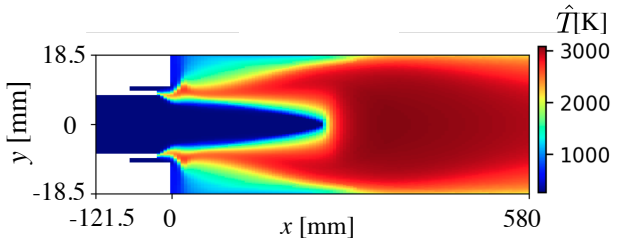
Fig. 14 Loss progression during training for base U-Net and Fully Convolutional (FC) networks , for training and validation datasets.

First, a classical MSE loss, defined by Eq.(21), is used. Note that this MSE is not computed over the whole 128×128 cells, but only on a masked subset containing the temperature information. The mask of the geometry provided as input is used for this purpose. The loss evolution over the number of epochs for both networks is shown in Figure 14. They both reach a plateau with regards to their validation loss after roughly 400 epochs of training. The U-Net training loss continues to decrease further, indicating an overfitting for larger epochs. This overfitting of the U-Net might come from its higher number of weights. No extensive hyper-parameters tuning was done for this task.

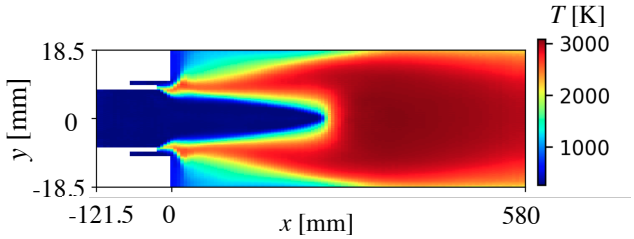
Type	Hidden Layers	Trainable Weights	Epochs	Batch Size	LR	LR-Decay
Fully Convolutional	9	400k	500	64	0.001	0.99
U-Net	13	490k				

Table 7 Convolutional Neural Network architectures used to predict temperature fields.

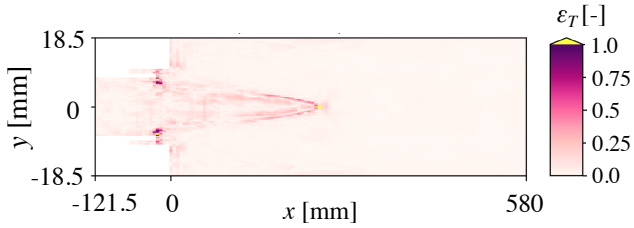
An overview of the relative error distribution for a test dataset prediction using either the fully convolutional net or the U-net is shown in Figure 15 (c) and (d), and the corresponding target output is shown in Figure 15 (a). These figures illustrate the comparison only on one test dataset sample, taken as an example, but the behaviour is similar for all the



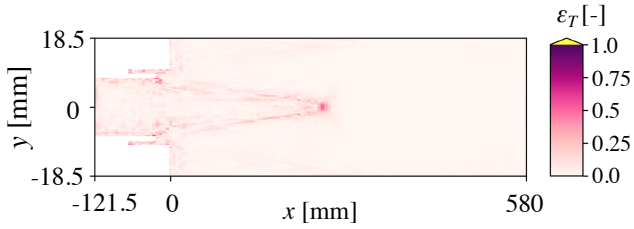
(a) Target temperature field.



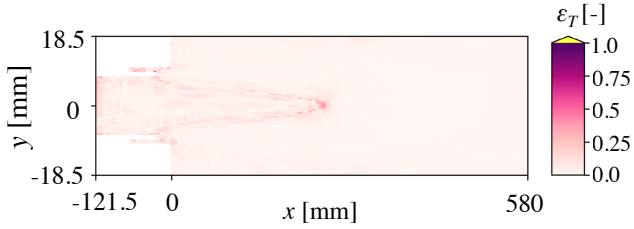
(b) Predicted temperature field using U-Net, 10/1 GDL .



(c) Relative error ε_T using the Fully Convolutional Net.



(d) Relative error ε_T using the U-Net.



(e) Relative error ε_T using the U-Net with a mixed 10/1 GDL.

Fig. 15 Target and predicted temperature fields via best performing network, and relative errors fields for different networks. Sample of test dataset

samples. It can be seen that the amplitude of the error is less for U-Net. However, the location of the error remains similar. Both networks exhibit the higher relative error in very small areas of strong temperature gradients, associated with the thin flame front. The location of a strong gradient in temperature has a correlation with the combustion length, which is a quantity of interest for optimization. Therefore further action was taken to decrease the error in this region. A mix between MSE Loss and a Gradient Difference Loss (GDL) was investigated. The GDL loss is defined by:

$$GDL(y,\hat{y}) = \frac{1}{N}\sum_{k=1}^{N}\left[\sum_{i,j} || \left[y_k(i,j) - y_k(i-1,j)\right] - \left[\hat{y}_k(i,j) - \hat{y}_k(i-1,j)\right] ||^2 + \right. \\ \left. ||(y_k(i,j) - y_k(i,j-1) - (\hat{y}_k(i,j) - \hat{y}_k(i,j-1)||^2\right] . \tag{24}$$

The GDL corresponds to an L2 loss on the gradient image, thus encouraging the network to capture correctly strong gradient regions. This strategy has been applied successfully in multiple fluid problems such as [23]. Four different mixing ratios between the two losses were tested for both network architectures: only MSE, equal weighting (1/1) and increased weights for the GDL/L2 mix (10/1 and 100/1) . Figure 16 gives an overview of the relative errors. The left graph shows a decline in overall relative field error when using mixing ratios of 1/1 and 10/1 for both architectures, but a significant increase for 100/1. It suggests that an optimal mixing between GDL and L2 can be achieved. The optimal weighting 10/1 corresponds to a contribution of 79 % and 21 % of GDL and L2 in the total train loss, respectively. Interestingly, the GDL allows for a better capture of the temperature gradients, i.e. a better accuracy on the flame shape. For all cases, training, validation and test errors are close indicating only minimal overfitting. On average, the U-Net exhibits a small advantage for all runs compared to the Fully Convolutional architecture. An example of field prediction and corresponding relative error achieved with the best performing U-Net architecture using 10/1 mixed GDL/L2 Loss during training are shown in Fig.15(b) and (e) respectively. The strongest temperature gradient is well resolved. The relative error is concentrated in a very limited area in the fuel injector and as noise in the oxidizer injector lines.

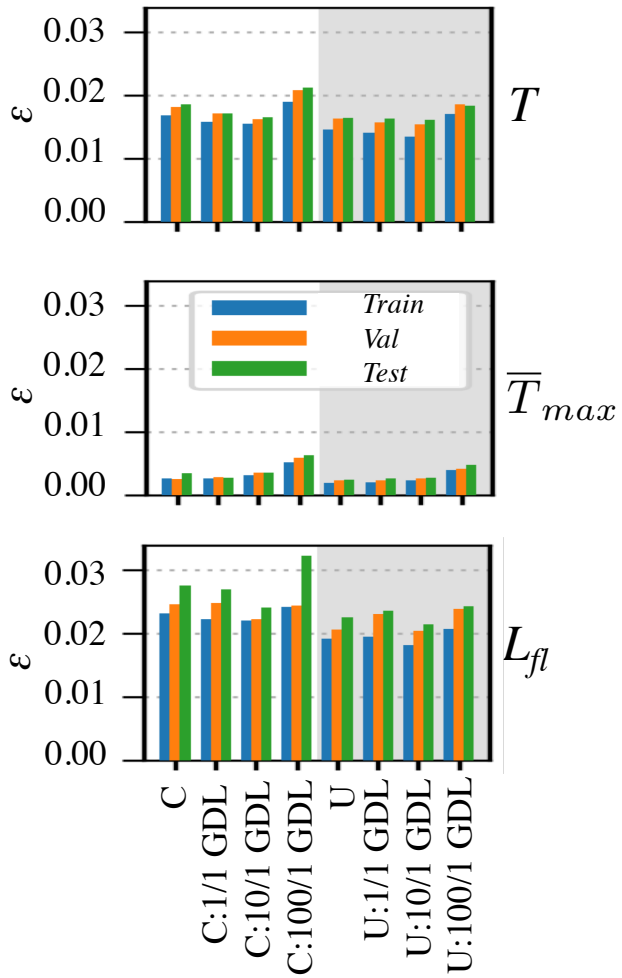


Fig. 16 Mean relative error for trained CNN and U-Net networks with varying levels of GDL/MSE mixing in the loss. Top: T field. Middle: $\overline{T}_{max}$. Bottom: L_{fl}.

The field prediction can also be used to extract lower dimensional quantities known from the simulation data. Figure 17 shows the cross-section averaged temperature along the chamber for 4 different configurations with the same chamber length. It can be seen that the predicted curves closely match the true temperature distributions and most of errors occur at peak gradient values.

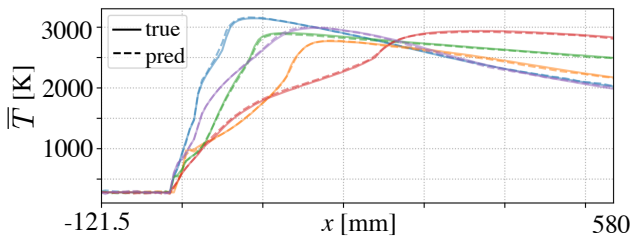


Fig. 17 Mean temperature prediction along the chamber length. Predictions via U-Net (10/1 GDL).

The average temperature curves in Fig. 17 were evaluated to extract the peak temperature $\overline{T}_{max}$ and its location,

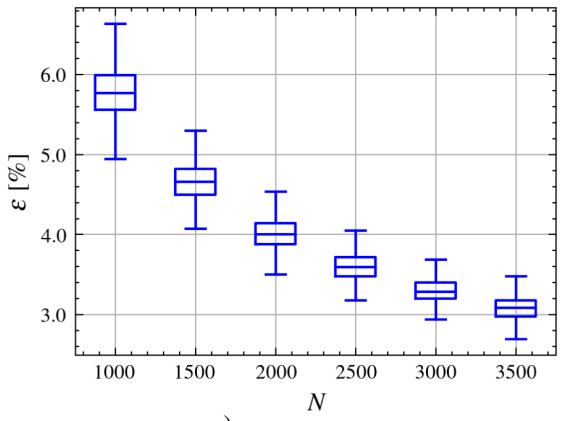
which is used as an indicator of the flame length L_{fl}. These values were also obtained by the FCNN architectures trained on global quantities (see Sec. V.A). Figure 16 shows lower error for the U-Net architecture on these computed quantities. A mixed loss approach showed no improvement for $\overline{T}_{max}$. The mixed loss does however marginally reduce error for L_{fl}. Indeed, a reduced gradient error might also yield to a reduced offset of gradient leading to better location accuracy down the chamber. Overall, the L_{fl} error translated into a cell resolution is between 1 and 2 cells. A further decrease might therefore need higher resolution of the field.

To summarise, the fact that using a GDL allows a better capture of the temperature gradients, is confirmed by the better accuracy obtained by the weighting 10/1 on both the whole temperature field as well as the location of the $\overline{T}_{max}$ (that is L_{fl}). However, mixing GDL and L2 with 10/1 penalize the prediction of the maximal temperature in the combustion chamber. Moreover, this conclusion holds for both convolutional networks. It suggests that in general for multi-objective tasks, no proper loss can be defined, and a careful choice should be made depending on which objective is the priority. For instance here, a different mixing weight will be selected if $\overline{T}_{max}$ or L_{fl} is chosen as a priority.

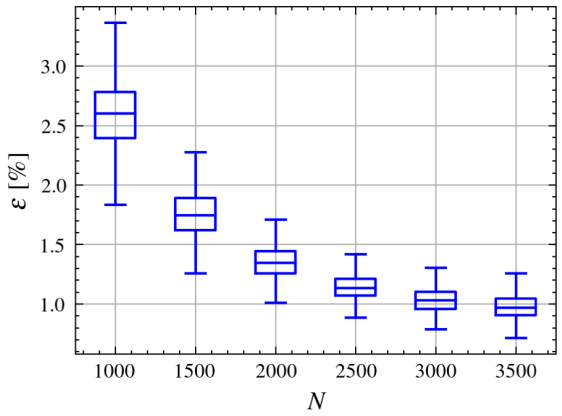
Quantity \ Architecture	Fully Convolutional	U-Net	FCNN
Temperature Field: T	98.35%	**98.39%**	–
Peak y-Avg Temperature: $\overline{T}_{max}$	99.64%	99.73%	**99.83%**
Location of $\overline{T}_{max}$: L_{fl}	97.59%	**97.85%**	97.53%

Table 8 Test data accuracy for CNN architectures using 10/1 GDL mixed loss, compared against FCNN networks predicting global quantities.

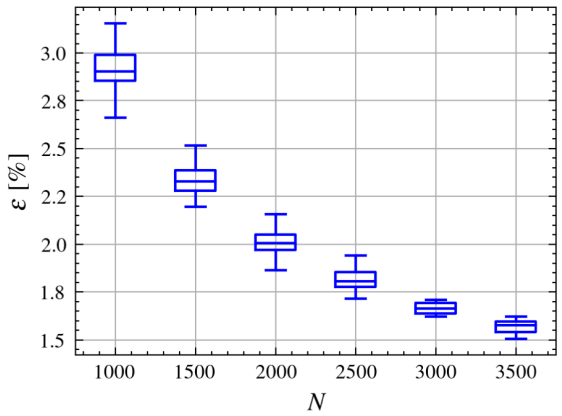
Finally, a field accuracy is estimated by computing the mean of Eq.(23) over all the mask points for every sample and averaging over all the samples of the validation dataset. The accuracy is reported in Table 8 for Fully Convolutional and U-Net networks using a 10/1 GLD/L2 loss mix. The achieved accuracy with both networks exceeds 98%. Performances for $\overline{T}_{max}$ and L_{fl} are also reported. The accuracy on computed global quantities from the predicted field is comparable with the accuracy of the dedicated FCNN model. The U-Net architecture does have a slight edge over the Fully Convolutional network in all cases, the difference being most notable for L_{fl}. While being more difficult and costly to train with respect to the FCNN on the global quantities, convolutional networks on the field data have the benefit to output the complete temperature field, much easier to interpret in terms of physics. Therefore, they constitute a first step towards reliable AI-based tools.



a) average pressure



b) wall heat flux



c) temperature field

Fig. 18 Error on the test dataset for different DOE sizes. Whiskers: complete range. Box: 1^{st} till the 3^{rd} quartile. Median line: mean error.

D. Sensitivity analysis on the DOE dimension

In the previous sections, surrogate models have been extracted for global, wall and field quantities, based on a DOE composed of 3600 points, among which 3437 where used for the training. Each point corresponds to a numerical

simulation. Decreasing the number of the DOE points would decrease the computational cost associated with the surrogate models generation. A sensitivity analysis of the results on the number of samples of the DOE has been carried out. This has been done by carrying out the networks training for global pressure, the wall heat flux, and the temperature field varying the number of samples N in the DOE between 1000 and 3437. In each case, no hyper parameters or network architecture variations have been carried out: the best networks selected in Sections V.A, V.B and V.C have been used.

The evolution of the mean relative error ε with the number N of samples in the DOE is shown in Fig .18.a), 18.b) and 18.c) for the average pressure, wall heat flux and temperature field respectively. The complete range of the relative error is also reported in order to highlight the sparsity of the results. For both 0D, 1D and 2D quantities, both the mean error and the sparsity of the error decrease when the number of samples N increases. However, the gain in increasing the DOE size is reduced for higher N. In particular, for N greater than 3000, the error for the global quantity reaches a plateau. This sensitivity analysis gives an idea of the requirements in term of DOE extension in order to obtain a specific accuracy. For instance, if an error between 1.3 and 2.3 % on the global quantities is acceptable, a dataset with $N = 1000$ samples is sufficient. On the other hand, if the surrogate model should provide high accuracy on the 1D and 2D quantities, the dataset used in the present analysis comprising $N = 3437$ samples, seems to be necessary. This analysis clearly points out the necessity in providing innovative strategies in order to reduce the dataset dimension if the methodology is to be applied on higher fidelity computations (namely Large Eddy Simulations). This will be the object of future works.

VI. Surrogate models application and exploitation

The surrogate models derived in section V can be used in the design and optimization of a coaxial injector and of the rocket configuration in which it is used. Specifically, these models can be useful in two ways. First, they can be used to carry out an optimization of the injector with respect to some specific figures of merit, and the optimization of a whole rocket system using the injector. In particular global and wall quantities are the more suitable outputs for this task. Second, they can contribute to gain insight into some specific flow and combustion features that eventually lead to the specific performances that bring to the optimal design. Indeed, having access to the field with a very low inference time allows the immediate observation of the flow topology induced by a selected design and thus to deduce specific conclusion on the observed behaviour.

To showcase the items mentioned above, a constrained multi-objective injector design problem has been addressed. It is of common interest for design bureaus to search for injector geometries which provide a best compromise between engine performance, typically represented by its specific-impulse I_{sp}, and engine compactness. Both variables have an opposing dynamics as larger engines will typically allow for better combustion as flow residence time in the combustion chamber is increased. The dimensioning of the combustion-chamber length is associated to the characteristic length

which directly depends on the flame length L_{fl}[44]. The rationale is smaller flame-lengths allow greater system compactness.

Fixed Values			**Variables Ranges**		
F_t $[N]$	δ_F $[N]$	$\overline{x} = \left[d_o, d_t, d_f, d_c, \frac{A_c}{A_t}, \frac{A_e}{A_t}\right]^T$	$\dot{m}$ [kg/s]	O/F	l_r [mm]
135.9	1.0	[2.0, 0.5, 0.5, 6.77, 2.5, 2.0]	0.001 - 0.13	2.0 - 3.99	0.0 - 30.0

Table 9 Multi-objective optimization problem variables values and ranges.

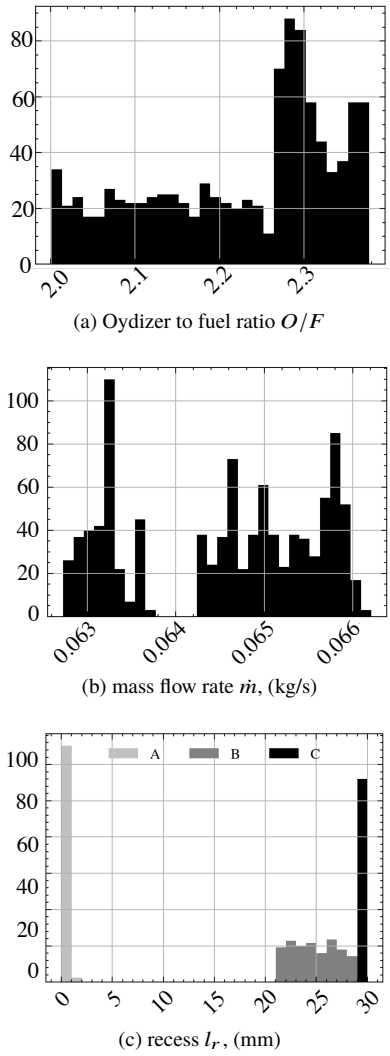
The problem addressed is to find the subset of mass-flow rate $\dot{m}$, mixture ratio (O/F) and recess length l_r within the domain of validity of the global quantities models detailed in Section V.A which maximizes specific-impulse and minimize flame-length. Additionally, the design is constrained to yield a target thrust F_t with a given uncertainty δ_F.

Among the 9 input parameters of the DOE (see Sec III.A) only 3 are considered variables: $\dot{m}$, O/F and l_r. Remaining 6 degrees of freedom, hereby called $\overline{x}$ have been given arbitrarily fixed values (Tab. 9) to further simplify the present problem.

The design problem under consideration is to find then $x \in \boldsymbol{X}_p$, where $\boldsymbol{X}_p$ corresponds to the set of non-dominated solutions, also called Pareto solutions, of the multi-objective optimization (MOO) problem with objective functions: $f_1(x) = -I_{sp}(x, \overline{x})$ and $f_2(x) = L_{fl}(x, \overline{x})$. L_{fl} corresponds to the axial location of the maximum average temperature in the chamber, hereby used as an indicator of flame-length. The problem is also subjected to the constraint $g(x) \leq 0$, such that $g(x) = |F_t - F(x, \overline{x})| - \delta_F$.

Several algorithms which treat the MOO problem described above are available in the literature. In the current work, the *NSGA-II* (Non Dominated Sorting Genetic Algorithm) is employed. NSGA-II, as with other algorithms of its type, is non-gradient based. Note that even though NSGA-II spares of the readily available gradients offered by a neural-networks, the fast inference times of the surrogates models used (F, I_{sp}, L_{fl}) allow to keep the optimization problem tractable. The algorithm is chosen due to its straight-forward implementation and accessibility. In the results to follow, the concerned values taken are detailed in Tab. 9.

The NSGA-II algorithm is run by means of the *PyMOO* python package [45], with a population size of 1000 agents, random sampling and crossover probability of 0.9. The algorithm reached a solution set after 45 generations with a tolerance on the objective function of 10^{-3}. Overall, 45000 evaluations where made of the objective functions $f_i(x)$. The algorithm is observed to explore the complete range of the individual coordinates of the search-space. This is a convenient property of the algorithm which makes it more robust in the face of local minima, even though it is not a sufficient condition in the path towards the global minima. Fig.19 displays the histogram for the non-dominated



(a) Oydizer to fuel ratio O/F

(b) mass flow rate $\dot{m}$, (kg/s)

(c) recess l_r, (mm)

Fig. 19 NSGA-II non-dominated solution set (Pareto solutions) frequency in the search-space.

solutions set. The points are distributed in the intervals $O/F \in [2.0,\ 2.37]$, $\dot{m} \in [6.27 \times 10^{-2}, 6.62 \times 10^{-2}]$ kg/s and $l_r \in [0.0, 29.99]$ mm. Note that the solutions interval in the $\dot{m}$ dimension corresponds to only 2.7% of the space defined in Table 9 which may result as a consequence of the strong relationship between the expected thrust and mass flow rate. In the l_r dimension the points are not distributed uniformly but rather three clusters are observed, as indicated in Fig. 19. Cluster A corresponding to solutions where $l_r \leq 2.0$ mm, Cluster B where 20 mm $\leq l_r \leq$ 29.9 mm and Cluster C with $l_r > 29.9$ mm. The non-dominated solutions labeled by cluster are plotted in the objective space in Fig. 20, also known as the Pareto front.

Points in Fig. 20 show a variation along the Pareto front of 216.18 mm in flame-length and greater than 13.64 s in I_{sp}. The front is seemingly continuous with a discontinuity between points of clusters A and B. Three points, *A, B* and *C* belonging to each of the clusters are highlighted in the graph. Their corresponding temperature field predictions through the field quantities surrogate model is shown in Fig. 21a. point *C* corresponds to a highly recessed case

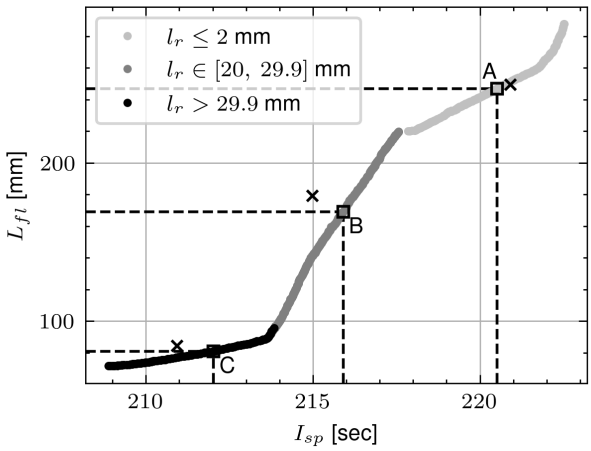
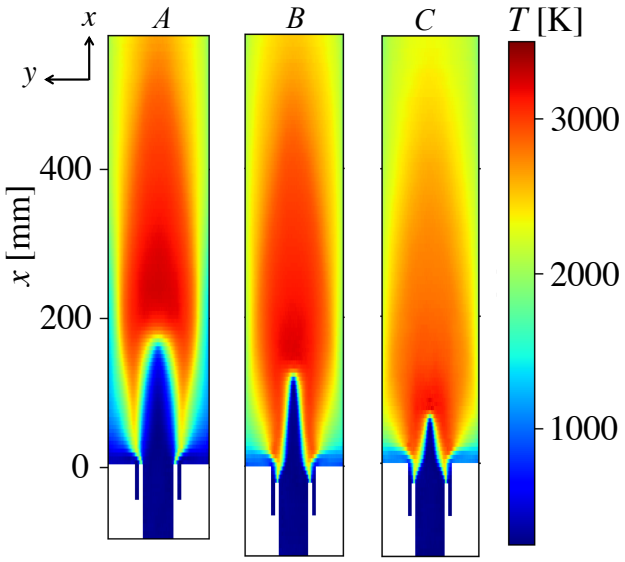


Fig. 20 Projection of the non-dominated solutions in the objective space (Pareto front), from the multi-objective problem. Crosses: RANS results.

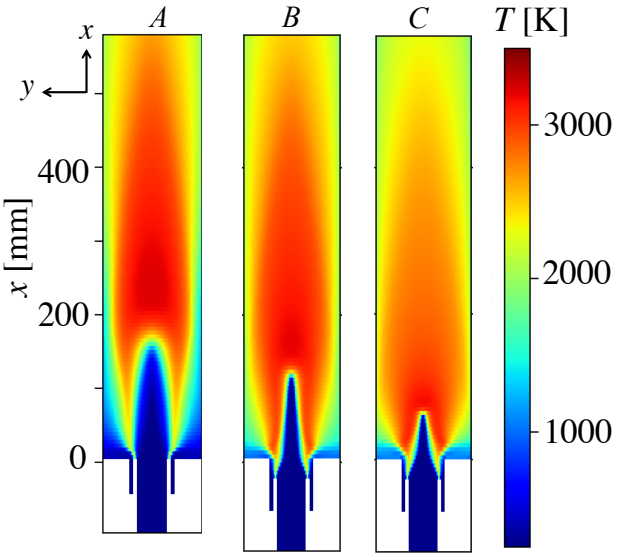
($l_r \approx 30.0$ mm) with $O/F = 2.18$ and $\dot{m} = 6.58 \times 10^{-2}$ kg/s. *Point B* in Fig. 21a corresponds to an intermediate value of recess ($l_r \approx 24.4$ mm) with $O/F \approx 2.29$ and $\dot{m} = 6.48 \times 10^{-2}$ kg/s. Finally, point A refers to a no-recess case ($l_r \approx 0.0$) with $O/F \approx 2.15$ and $\dot{m} = 6.32 \times 10^{-2}$ kg/s.

The temperature field in point C reflects a sooner expansion due to the presence of a highly recessed oxidizer post. The faster expansion results in a shorter flame-length, to the expense of a higher temperature at the wall in the first half of the chamber (increasing wall heat-fluxes) and a shorter mixing length. The temperature of the gases decreases sharply in the second half of the chamber being greatly reduced at the chamber's end in contrast to point's A and B. This has a negative impact on the performance seen on the smaller I_{sp} linked to the points of cluster C. Note that the early expansion of the gases in the recess channel causes the necking of the oxidizer jet, which is also seen in the intermediate recess case (point B). The temperature field in point B is similar to that of point C, however a longer flame is observed congruent with a higher temperature of gases at the exit which explains the behavior of the Pareto front in the regime of cluster B. Note that point B, in comparison to point C, has an increased O/F. Scaling relations for reacting coaxial oxy-flames [46] provide insight on the longer flame as $L_{fl}/(2 * d_o) \propto J^{-1/2} \propto O/F$, where $J = (\rho u^2)_{\mathrm{CH_4}}/(\rho u^2)_{\mathrm{O_2}}$ is the momentum-flux-ratio and d_o the oxidizer post radius. Point B also shows a longer recirculation zone as a consequence of the longer flame as well as decreased wall temperatures. Finally, point's A temperature field shows a non-recessed injector with a fully concave average flame shape. This is due to the absence of the initial expansion in the channel linked to the recessed injector, as in points B and C. The peak temperature is located downstream due to a slower expansion. It is suspected that to counter-act the longer flame, the optimization algorithm has resolved a smaller O/F yielding a smaller oxidizer jet inlet velocity. The smaller temperatures observed at the wall as well as the longer flame with higher temperatures at the chamber's end are expected to be the cause of the greater I_{sp} in such configuration.

Finally, RANS simulations with the parameters corresponding to points A, B and C have been carried out. The



(a) Surrogate model predictions



(b) RANS fields

Fig. 21 Temperature fields prediction provided by the best performing model, U-Net GDL 10/1 , for selected points of Fig.20, and respective fields computed with RANS.

obtained temperature fields are reported in Fig.21b. The temperature fields are in agreement with the surrogate models predictions. From a quantitative point of view, for A, B and C the errors on L_{fl} are of 1%, 5.6% and 3.2 % respectively. The errors for the I_{sp} are 0.2%, 0.43 % and 0.52% respectively. These values are in the expected ranges discussed in Section V.A, since an average error of 3.47 % was reported for the L_{fl} and 0.32% for the I_{sp}. The impact of this error on the Pareto front estimation can be seen in Fig. 19 where the computed values of L_{fl} and I_{sp} for the three points from the RANS are also reported: points A, B and C from the RANS are in proximity of the Pareto front as expected from the estimated errors of the surrogate models.

The previous analysis showcased a particular application problem for multi-objective optimization with the aforementioned surrogate models, for a problem involving three design parameters. However, more intricate single objective optimization problems, involving a compromise objective function may be solved by harnessing the power of neural networks and the readily available gradients. Optimization problems, involving a larger number of design parameters could be addressed in future works. Combined with robust gradient based optimization algorithms such as Stochastic Gradient Descent and the fast inference time these models offer, multidisciplinary optimization loops may be solved in realistic timescales.

VII. Conclusions

In the present paper, a first attempt to jointly use data from computational fluid dynamics simulations and deep learning for the design of rocket engines injector and combustion chamber has been carried out. Reynolds-Averaged Navier-Stokes simulations of a single coaxial injector chamber, operated with gaseous methane and oxygen propellants, have been carried out in order to generate a large database which comprises more than 3600 different points. It has been shown that particular attention must be devoted to the design of experiments generation in order to ensure the physical correctness of the computed configurations. The core of the work has been devoted to the development of three classes of surrogate models: global quantities, wall quantities and field quantities surrogate models. All the models have the same nine inputs parameters that define the geometry and the operative conditions and differ in the dimension of the output. A comparison between fully connected neural networks (FCNN) and other machine learning approaches, namely Kriging and radial basis functions, has been done for the gloabl quantities models, showing a comparable accuracy between the approaches. After hyperparameter tuning, the best neural network accuracy for the global quantities varies between 97.5 and 99.92%. When increasing the dimension of the problem (1D and 2D) only deep learning methods are of interest. For the wall heat flux, the developed surrogate model based on a fully connected neural network is able to reproduce the data with an accuracy up to 97%. Two types of convolutional neural network have been tested for the temperature field models: a standard CNN and a U-Net. Particularly, an increase in the accuracy is achieved by considering a mixed loss which combines a Mean Squared Error loss and Gradient Difference loss, that allows to reach a mean accuracy of 98.49% with the U-Net. Interestingly, the field surrogate model allows to retrieve important global quantities, like the flame length, with the same accuracy of the best FCNN selected for the corresponding global model. Eventually, the obtained models have been used in the framework of an optimisation problem. In this example, the global quantities models allow to choose the best configuration with respect to the objective function and the specific constraints. Immediately, the field model allows to visualise the corresponding temperature field thus giving clear indications and explanations of the observed behaviour in terms of performance.

It is worth mentioning that from a quantitative point of view the presented results are of course affected by the validity and quality of the data. Reynolds-Averaged Navier-Stokes simulations do not allow to capture specific low scale

and non stationary phenomena that play an important role in the flow development inside the combustion chamber. Moreover, all the study has been carried out considering a constant wall temperature, which is a strong approximation. Nevertheless, the results show the potentiality of the methodology in permitting fast and comprehensive analysis of complex phenomena. This work represents a proof of concept of a methodology that needs to be further developed increasing the quality of the data. Specifically, it would be interesting to use Large Eddy Simulation to generate the data. Furthermore, it would be interesting to test the above developed surrogate models in the framework of a multi-disciplinary analysis and optimisation of a whole rocket engine.

Appendix

A. DOE generation and constraints

Having chosen the input parameters, a database need to be generated by properly varying then inputs in the 9D parameter space, that is by generating a proper Design of Experiments (DOE). One possibility would be to randomly sample points linearly within the parameter space. This approach has three main drawbacks. First, some combinations of input parameters could lead to non-physical configurations and therefore have to be withdrawn from the DOE. Second, a random sampling would not guarantee a maximised space between the points. Third, a linear sampling could not be suitable depending on the mathematical dependence expected between inputs and outputs. Therefore, two improvements were made to increase the viability of the design points and their use in training a neural network as explained in the following. As expected, preliminary tests showed that a purely random sampling of the parameter space can lead to nonphysical setups. Mostly, these issues would manifest themselves as supersonic flows in the oxidizer or fuel inlets or chamber pressures orders of magnitude higher than anticipated. To avoid those points, we need to impose the following three conditions that are usual for the design of liquid rocket engine combustion chambers and injectors [47]. First, 1), the contraction area should be high enough in order to ensure a choked nozzle. Second, 2), the injector Mach number should not exceed 0.7: $M_{in} < 0.7$. Third, 3), the chamber Mach number should not exceed 0.3: $M_c < 0.3$. Conditions 1) and 3) bring to the following restriction on the contraction area A_c/A_t:

$$\frac{A_c}{A_t} > \frac{1}{M_c}\sqrt{\left(\frac{1+\frac{\gamma-1}{2}M_c^2}{1+\frac{\gamma-1}{2}}\right)^{\frac{\gamma+1}{\gamma-1}}}, \tag{25}$$

with $M_c = 0.3$, and γ is the specific heat ratio of the gas flowing in the nozzle. To find suitable constraints for d_o, d_f and d_c in order to respect conditions 2) and 3) the pressure chamber p_c influence has to be investigated. In fact, as the mass flow rate and temperature are inlet boundary conditions, the flow velocity in the inlet mainly depends on chamber pressure p_c. The pressure chamber p_c is an output of the simulation. In fact the nozzle is choked, and therefore the mass

flow rate can be estimated with the well known expression derived from the theory of 1D isentropic flow in nozzles [47]:

$$\dot{m} = \Gamma \frac{p_c A_t}{\sqrt{RT_c}} \tag{26}$$

where $\Gamma = \sqrt{\gamma \left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}$, R is the gas constant and T_c is the total temperature in the chamber. This equation can be recast in order to express the dependence of p_c over three design parameters $\dot{m}$, A_c/A_t and d_c :

$$\frac{p_c}{p_{c,ref}} = K \left(\frac{\dot{m}}{\dot{m}_{ref}}\right) \left(\frac{(A_c/A_t)}{(A_c/A_t)_{ref}}\right) \left(\frac{d_{c,ref}}{d_c}\right)^2 \tag{27}$$

where the index ref refers to the reference setup and $K = (\sqrt{\Gamma/RT_c})/(\sqrt{\Gamma/RT_c})_{ref}$. From Eq.(27) we expect, in a first approximation, to find a linear relation between p_c and $\dot{m}$, and between p_c and A_c/A_t and a quadratic relation between p_c and d_c.

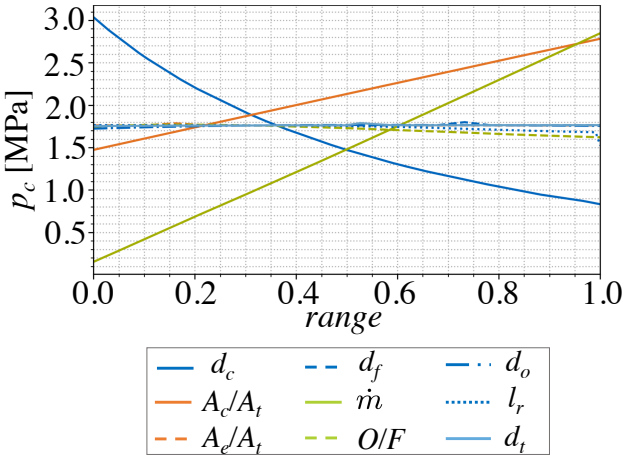


Fig. 22 Response of chamber pressure p_c to changes in parameters, sweeping through one parameter range at a time.

To verify this statement and to investigate the other dependencies, a study was carried out to investigate the influence each input parameter individually has on p_c when other values are set to their reference values. For each parameter, 20 simulations along their range where carried out and the obtained p_c with respect to the varied parameter is shown in Fig. 22. As expected, only the above mentioned 3 parameters change p_c to a significant degree (d_c, A_c/A_t, $\dot{m}$). Their interactions when changed simultaneously are not known at this point, but in order to estimate physical limitations an approximate pressure model was extracted from this study. For this, linear functions were fitted to $\dot{m}$ and A_c/A_t and a quadratic function to d_c with respect to their reference values. The approximate chamber pressure $p_{c,ap}$ was defined as follow :

$$p_{c,ap} = p_{c,ref} \overbrace{\frac{\dot{m}}{\dot{m}_{ref}}}^{\text{linear}} \overbrace{\frac{A_c/A_t}{A_c/A_{t_{ref}}}}^{\text{linear}} \overbrace{\left(2.1\left(\frac{d_c}{d_{c\,ref}}\right)^2 - 6.2\frac{d_c}{d_{c\,ref}} + 5.15\right)}^{\text{quadratic fit}} \tag{28}$$

$$\tag{29}$$

With this approximate pressure $p_{c,ap}$, we are able to express the following constraints on d_o, d_f and d_c in order to verify conditions 1) and 3):

$$d_o > \sqrt{\frac{\dot{m}_{o_2}}{0.7 p_{c,ap}\pi}\sqrt{\frac{c_v \mathcal{R} T_{o_2}}{c_p W_{O_2}}}}, \tag{30}$$

$$d_f > \sqrt{d_o^2 + 2d_o d_t + d_t^2 + \frac{\dot{m}_{CH_4}}{0.7 p_{c,ap}\pi}\sqrt{\frac{c_v \mathcal{R} T_{CH_4}}{c_p W_{CH_4}}}} - d_o - d_t, \tag{31}$$

$$d_c > \sqrt{\frac{a_c}{a_{in}}\left(d_o^2 + d_f\left(2d_o + 2d_t + d_f\right)\right)}. \tag{32}$$

where $\mathcal{R} = 8314$J/kg/K is the universal gas constant and W_i is the molar mass of species i.

Funding Sources

This work was supported by the Chair for Space Advanced Concepts Laboratory (SaCLaB) resulting from the partnership between Airbus Defence and Space, Ariane Group and ISAE-SUPAERO.

This work was supported by Région Occitanie, with the MODAR project (grant number: 20007323/ALDOCT-001030) and CERFACS for the financing of Jose Felix Zapata Usandivaras' PhD.

References

[1] Martin, J., Armbruster, W., Stützer, R., General, S., Knapp, B., Suslov, D., and Hardi, J., "Flame Characteristics of a High-Pressure LOX/H2 Rocket Combustor With Large Optical Access," *Case Studies in Thermal Engineering*, Vol. 28, 2020, p. 101546. https://doi.org/https://doi.org/10.1016/j.csite.2021.101546.

[2] Urbano, A., Selle, L., Staffelbach, G., Cuenot, B., Schmitt, T., Ducruix, S., and Candel, S., "Exploration of Combustion Instability Triggering Using Large Eddy Simulation of a Multiple Injector Liquid Rocket Engine," *Combustion and Flame*, Vol. 169, 2016, pp. 129–140. https://doi.org/http://dx.doi.org/10.1016/j.combustflame.2016.03.020.

[3] Urbano, A., and Selle, L., "Driving and Damping Mechanisms for Transverse Combustion Instabilities in Liquid Rocket Engines," *Journal of Fluid Mechanics*, Vol. 820, No. R2, 2017, pp. 1–12. https://doi.org/http://dx.doi.org/10.1017/jfm.2017.227.

[4] Maestro, D., Cuenot, B., and Selle, L., "Large Eddy Simulation of Combustion and Heat Transfer in a Single Element GCH4/GOx Rocket Combustor," *Flow, Turbulence and Combustion*, Vol. 103, No. 3, 2019, pp. 699–730. https://doi.org/10.1007/s10494-019-00036-w.

[5] Huo, H., and Yang, V., "Large-Eddy Simulation of Supercritical Combustion: Model Validation Against Gaseous H2–O2 Injector," *Journal of Propulsion and Power*, Vol. 33, No. 5, 2017, pp. 1272–1284. https://doi.org/10.2514/1.B36368.

[6] Riedmann, H., Kniesner, B., Frey, M., and Munz, C.-D., "Modeling of Combustion and Flow in a Single Element GH2/GO2 Combustor," *CEAS Space Journal*, Vol. 6, 2014, pp. 47–59. https://doi.org/10.1007/s12567-013-0056-3.

[7] Perakis, N., Celano, M. P., and Haidn, O., "Heat Flux and Temperature Evaluation in a Rectangular Multi-Element GOX/GCH4 Combustion Chamber Using an Inverse Heat Conduction Method," *Proceedings of the 7th European Conference for Aeronautics and Space Sciences*, 2017. https://doi.org/10.13009/EUCASS2017-366.

[8] Lapenna, P. E., Indelicato, G., Lamioni, R., and Creta, F., "Modeling the Equations of State Using a Flamelet Approach in LRE-Like Conditions," *Acta Astronautica*, Vol. 159, 2019, pp. 460–469. https://doi.org/10.1016/j.actaastro.2018.07.025.

[9] Perakis, N., and Haidn, O. J., *Experimental and Numerical Investigation of CH4/O2 Rocket Combustors*, Springer International Publishing, Cham, 2021, pp. 359–379. https://doi.org/10.1007/978-3-030-53847-7_23.

[10] Lempke, M., Gerlinger, P., Seidl, M. J., and Aigner, M., "Unsteady High-Order Simulation of a Liquid Oxygen/Gaseous Hydrogen Rocket Combustor," *Journal of Propulsion and Power*, Vol. 31, No. 6, 2015. https://doi.org/10.2514/1.B35726.

[11] Wang, X., Yeh, S.-T., Chang, Y.-H., and Yang, V., "A High-Fidelity Design Methodology Using LES-Based Simulation and POD-Based Emulation: A Case Study of Swirl Injectors," *Chinese Journal of Aeronautics*, Vol. 31, No. 9, 2018, pp. 1855–1869. https://doi.org/10.1016/j.cja.2018.07.004.

[12] Mak, S., Sung, C.-L., Wang, X., Yeh, S.-T., Chang, Y., Joseph, V. R., Yang, V., and Wu, C. F. J., "An Efficient Surrogate Model for Emulation and Physics Extraction of Large Eddy Simulations," *Journal of American and Statistical Associtation*, Vol. 113, No. 524, 2018, pp. 1443–1456. https://doi.org/10.1080/01621459.2017.1409123.

[13] Gray, J., Chin, J., Hearn, T., Hendricks, E., Lavelle, T., and Martins, J. R. R. A., "Chemical-Equilibrium Analysis with Adjoint Derivatives for Propulsion Cycle Analysis," *Journal of Propulsion and Power*, Vol. 33, No. 5, 2017. https://doi.org/10.2514/1.B36215.

[14] Chen, L., Cakal, B., Hu, X., and Thuerey, N., "Numerical Investigation of Minimum Drag Profiles in Laminar Flow Using Deep Learning Surrogates." *Journal of Fluid Mechanics*, Vol. 919, No. A34, 2021. https://doi.org/10.1017/jfm.2021.398.

[15] Simpson, T. W., Mauery, T. M., Korte, J. J., and Mistree, F., "Kriging Models for Global Approximation in Simulation-Based Multidisciplinary Design Optimization," *AIAA Journal*, Vol. 39, No. 21, 2001, pp. 2233–2241. https://doi.org/https://doi.org/10.2514/2.1234.

[16] Shyy, W., Tucker, P. K., and Vaidyanathan, R., "Response Surface and Neural Network Techniques for Rocket Engine Injector Optimization," *Journal of Propulsion and Power*, Vol. 17, No. 2, 2001, pp. 391–401. https://doi.org/https://doi.org/10.2514/2.5755.

[17] Vaidyanathan, R., Tucker, P. K., Papila, N., and Shyy, W., "Computational-Fluid-Dynamics-Based Design Optimization for Single-Element Rocket Injector," *Journal of Propulsion and Power*, Vol. 20, No. 4, 2004, pp. 705–716. https://doi.org/https://doi.org/10.2514/1.11464.

[18] Hwang, J. T., and Martins, J. R. R. A., "A Fast-Prediction Surrogate Model for Large Datasets," *Aerospace Science and Technology*, Vol. 75, 2018, pp. 74–87. https://doi.org/10.1016/j.ast.2017.12.030.

[19] Brunton, S., Noack, B., and Koumoutsakos, P., "Machine Learning for Fluid Mechanics," *Annual Review of Fluid Mechanics*, Vol. 52, No. 1, 2020, pp. 477–508. https://doi.org/10.1146/annurev-fluid-010719-060214.

[20] Baqué, P., Remelli, E., Fleuret, F., and Fua, P., "Geodesic Convolutional Shape Optimization," *Proceedings of the 35th International Conference on Machine Learning*, Vol. 80, PMLR, 2018, pp. 472–481. URL http://proceedings.mlr.press/v80/baque18a/baque18a.pdf.

[21] Tompson, J., Schlachter, K., Sprechmann, P., and Perlin, K., "Accelerating Eulerian Fluid Simulation With Convolutional Networks," *Proceedings of the 34th International Conference on Machine Learning*, Proceedings of Machine Learning Research, Vol. 70, edited by D. Precup and Y. W. Teh, PMLR, 2017, pp. 3424–3433. URL https://proceedings.mlr.press/v70/tompson17a.html.

[22] Ajuria, E., Alguacil, A., Bauerheim, M., Misdaris, A., Cuenot, B., and Benazera, E., "Towards an Hybrid Computational Strategy Based on Deep Learning for Incompressible Flows," *AIAA Aviation Forum*, edited by AIAA-2020-3058, 2020. https://doi.org/10.2514/6.2020-3058.

[23] Alguacil, A., Bauerheim, M., Jacob, M. C., and Moreau, S., "Predicting the Propagation of Acoustic Waves Using Deep Convolutional Neural Networks," *Journal of Sound and Vibration*, Vol. 512, 2021. https://doi.org/10.1016/j.jsv.2021.116285.

[24] Ma, H., xuan Zhang, Y., Haidn, O. J., Thuerey, N., and yu Hu, X., "Supervised Learning Mixing Characteristics of Film Cooling in a Rocket Combustor Using Convolutional Neural Networks," *Acta Astronautica*, Vol. 175, 2020, pp. 11–18. https://doi.org/10.1016/j.actaastro.2020.05.021.

[25] Shadram, Z., Nguyen, T. M., Sideris, A., and Sirignano, W. A., "Neural Network Flame Closure for a Turbulent Combustor with Unsteady Pressure," *AIAA Journal*, Vol. 59, No. 2, 2021, pp. 621–635.

[26] Thuerey, N., Weissenow, K., Prantl, L., and Hu, X., "Deep Learning Methods for Reynolds-Averaged Navier-Stokes Simulations of Airfoil Flows," *AIAA Journal* , Vol. 58, No. 1, 2020, pp. 25–36. https://doi.org/10.2514/1.J058291.

[27] McQuarrie, S. A., Huang, C., and Willcox, K. E., "Data-Driven Reduced-Order Models Via Regularised Operator Inference for a Single-Injector Combustion Process," *Journal of the Royal Society of New Zealand*, Vol. 51, No. 2, 2021, pp. 194–211. https://doi.org/10.1080/03036758.2020.1863237.

[28] Haemisch, J., Suslov, D., Waxenegger-Wilfing, G., Dresia, K., and Oschwald, M., "LUMEN – Design of the Regenerative Cooling System for an Expander Bleed Cycle Engine Using Methane," *Space Propulsion Conference*, 2020.

[29] Celano, M. P., Silvestri, S., Schlieben, G., Kirchberger, C., Haidn, O. J., and Knab, O., "Injector Characterization for a Gaseous Oxygen-Methane Single Element Combustion Chamber," *Progress in Propulsion Physics*, edited by M. Calabro, L. DeLuca, S. Frolov, L. Galfetti, and O. Haidn, 2016, pp. 145–164. https://doi.org/10.1051/eucass/201608145.

[30] Chemnitz, A., Sattelmayer, T., Roth, C., Haidn, O., Daimon, Y., Keller, R., Gerlinger, P., Zips, J., and Pfitzner, M., "Numerical Investigation of Reacting Flow in a Methane Rocket Combustor: Turbulence Modeling," *Journal of Propulsion and Power*, Vol. 34, No. 4, 2018, pp. 864–877. https://doi.org/10.2514/1.B36565.

[31] Pope, S. B., *Turbulent Flow*, 1st ed., Cambridge University Press, 2000.

[32] Poinsot, T., and Veynante, D., *Theoretical and Numerical Combustion*, 2nd ed., Edwards, Philadelphia, 2005.

[33] McBride, B., and Gordon, S., "Computer Program for Calculation of Complex Chemical Equilibrium Compositions and Applications: 1. Analysis," Tech. Rep. NASA RP 1311, NASA, October 1994.

[34] Launder, B. E., and Spalding, D. B., *Lectures in Mathematical Models of Turbulence*, Academic Press London, New York, London, New York, 1972.

[35] Peters, N., "Laminar Diffusion Flamelet Models in Non-Premixed Turbulent Combustion," *Progress in Energy and Combustion Science*, Vol. 10, No. 3, 1984, pp. 319–339. https://doi.org/10.1016/0360-1285(84)90114-X.

[36] Kim, T., Kim, Y., and Kim, S.-K., "Effects of Pressure and Inlet Temperature on Coaxial Gaseous Methane/Liquid Oxygen Turbulent Jet Flame under Transcritical Conditions," *The Journal of Supercritical Fluids*, Vol. 81, 2013, pp. 164–174. https://doi.org/10.1016/j.supflu.2013.05.011.

[37] Roche, P. J., "Verification of Codes and Calculations," *AIAA Journal* , Vol. 36, No. 5, 1998, p. 696. https://doi.org/10.2514/2.457.

[38] McKay, M. D., Beckman, R. J., and Conover, W. J., "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code," *Technometrics*, Vol. 21, No. 2, 1979, p. 239. https://doi.org/10.2307/1268522.

[39] Jin, R., Chen, W., and Sudjianto, A., "An Efficient Algorithm for Constructing Optimal Design of Computer Experiments," *Journal of Statistical Planning and Inference*, Vol. 134, No. 1, 2005, pp. 268–287. https://doi.org/10.1016/j.jspi.2004.02.014.

[40] Bouhlel, M. A., Hwang, J. T., Bartoli, N., Lafage, R., Morlier, J., and Martins, J. R., "A Python Surrogate Modeling Framework with Derivatives," *Advances in Engineering Software*, Vol. 135, 2019, p. 102662. https://doi.org/10.1016/j.advengsoft.2019.03.005.

[41] Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S., "PyTorch: An Imperative Style, High-Performance Deep Learning Library," *Advances in Neural Information Processing Systems 32*, Vol. 32, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, Curran Associates, Inc., 2019, pp. 8024–8035. URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[42] Chollet, F., *Deep Learning with Python*, Manning Publications Co, Shelter Island, New York, 2017.

[43] Ronneberger, O., Fischer, P., and Brox, T., "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, edited by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Springer International Publishing, Cham, 2015, pp. 234–241.

[44] Khan, T. W., and Qamar, I., "Optimum characteristic length of gas generator for liquid propellant rocket engine," *Acta Astronautica*, Vol. 176, 2020, pp. 1–12.

[45] Blank, J., and Deb, K., "Pymoo: Multi-Objective Optimization in Python," *IEEE Access*, Vol. 8, 2020, pp. 89497–89509. https://doi.org/10.1109/ACCESS.2020.2990567.

[46] Degenève, A., Vicquelin, R., Mirat, C., Labegorre, B., Jourdaine, P., Caudal, J., and Schuller, T., "Scaling Relations for the Length of Coaxial Oxy-Flames with and without Swirl," *Proceedings of the Combustion Institute*, Vol. 37, No. 4, 2019, pp. 4563–4570. https://doi.org/https://doi.org/10.1016/j.proci.2018.06.032.

[47] Sutton, G. P., and Biblarz, O., *Rocket Propulsion Elements*, 8th ed., John Wiley and Sons, INC., Hoboken, New Jersey, 2000.