



Development of an Artificial Intelligence Model to Predict Combustion Properties, With a Focus on Auto-Ignition Delay

Roda Bounaceur, Romain Heymes, Pierre-Alexandre Glaude, Baptiste Sirjean, René Fournet, Pierre Montagne, Alexandre Auvray, Eric Impellizzeri, Pierre Biehler, Alexandre Picard, et al.

► To cite this version:

Roda Bounaceur, Romain Heymes, Pierre-Alexandre Glaude, Baptiste Sirjean, René Fournet, et al.. Development of an Artificial Intelligence Model to Predict Combustion Properties, With a Focus on Auto-Ignition Delay. *Journal of Engineering for Gas Turbines and Power*, 2024, 146, pp.061011. 10.1115/1.4063774 . hal-04406508

HAL Id: hal-04406508

<https://hal.science/hal-04406508>

Submitted on 19 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

**DEVELOPMENT OF AN ARTIFICIAL INTELLIGENCE MODEL TO PREDICT
COMBUSTION PROPERTIES, WITH A FOCUS ON AUTO-IGNITION DELAY**

Roda Bounaceur, Romain Heymes, Pierre-Alexandre Glaude, Baptiste Sirjean,

René Fournet,

Université de Lorraine-CNRS, Laboratoire Réactions et Génie des Procédés, Nancy,
France

Pierre Montagne, Alexandre Auvray, Eric Impellizzeri, Pierre Biehler, Alexandre Picard,

Bastien Prieur-Garrouste,

GE Gas Power France, Belfort, France

Michel Molière

Université de Technologie de Belfort Montbéliard, IRTES-LERMPS, Belfort, France

Corresponding author :

Pierre-Alexandre Glaude, pierre-alexandre.glaude@univ-lorraine.fr

ABSTRACT

Hydrogen-compatible gas turbines are one way to decarbonize electricity production. However, burning and handling hydrogen is not trivial because of its high reactivity and tendency to detonate. Mandatory safety parameters, such as auto-ignition delay times, can be estimated thanks to predictive detailed kinetic models, but with significant calculation times that limit coupling with fluid mechanic codes. An auto-ignition prediction tool was developed based on an artificial intelligence (AI) model for fast computations and an implementation into an explosion model. A dataset of ignition delay times was generated automatically using a recent detailed kinetic model from National University of Galway (NUIG) selected from the literature. Generated data covers a wide operating range and different compositions of fuels. Clustering problems in sample points were avoided by a quasi-random Sobol sequence, which covers uniformly the entire input parameter space. The different algorithms were trained, cross-validated and tested using a database of more than 70'000 ignitions cases of Natural Gas/Hydrogen blends calculated with the full kinetic model by using a common split of 70/30 for training, testing. The AI model shows a high degree of robustness. For both the training and testing datasets, the average value of the correlation coefficient was above 99.91%, the Mean Absolute Error (MAE) and the Mean Square Error (MSE) around 0.03 and lower than 0.04 respectively. Tests showed the robustness of the AI model outside the ranges of pressure, temperature, and equivalence ratio of the data set. A deterioration is however observed with increasing amounts of large alkanes in the natural gas.

Keywords: hydrogen, AI model, ignition delay times.

1. INTRODUCTION

The development of hydrogen compatible gas turbines is aimed at playing a role in enabling the decarbonization of the power sector [1,2]. In fact, direct greenhouse gas (GHG) emissions generated by the gas turbine reflect the carbon content in the fuel [3]. This fuel switch from hydrocarbon containing fuels such as natural gas and light distillate oil to part of or pure hydrogen is going to displace CO₂ emissions by H₂O. This latter has a negligible global warming potential. Assuming hydrogen is produced upfront by low carbon process, it would enable minimizing the indirect GHG emissions as well.

Now, burning hydrogen in the combustor and even handling hydrogen in the gas delivery system is not trivial [4,5] and comes with challenges associated with the high reactivity (ignitability) and tendency to detonate [6,7], hence it is important to understand the mechanisms under consideration and develop tools to estimate the “reactivity parameters” [8,9]. Auto-ignition delay is one characteristic of the reactivity of a fuel-air mixture under given conditions of composition, pressure, and temperature, and an important parameter for the design of safe and reliable systems. A detailed kinetic model may be able to simulate these combustion parameters in simple reactors, but it is too expensive for coupled calculations with mass and heat transfer under gas turbine conditions [10]. On the contrary, simplified kinetics may well represent the characteristics of the flame front but not the self-ignition at moderate temperatures [11,12]. To get past these two pitfalls, this paper describes a method based on chemical kinetics as well as artificial intelligence (AI) models which enables its fine prediction as a function of pressure, temperature, equivalence ratio as well as hydrogen-natural gas blends composition. AI-based methods have shown their potential to reproduce the complex kinetics and couplings in combustion [13], whether for systems such as coal combustion [14], engine optimization [15] or large-scale forest fire simulation [16]. Several authors published different methodology based on ML algorithm to estimate directly auto-ignition delay times of hydrocarbon/air mixtures. The Table 1 presents some studies using ML algorithm to predict ignition delay times. Considering the importance of the field of natural gas and hydrogen combustion, the objective of this study is to present a new mathematical approach based on machine learning to estimate the ignition delay for a wide range of conditions, in terms of pressure,

temperature and equivalence ratio, and for a large number of natural gas/hydrogen mixtures under typical gas turbine operating conditions.

Table 1: Some studies on the estimation of IDT using ML algorithms

Type of ML Algorithm	Fuel mixtures	Domain of validity			Authors
		P [bar]	T [K]	ϕ [-]	
Adaptive Nonlinear Least-Squares Algorithm	C ₁ -C ₃ / H ₂	1 - 200	900 - 2500	0.9 – 2.5	Kuppa et al., (2018) [17]
Genetic Algorithm + Artificial Neural Network (GA-ANN)	nC ₄ / H ₂	20 - 25	722 - 987	0.5 – 1.5	Cui et al., (2020) [18]
Convolutional Neural Network (CNN)	DME-CH ₄	5 - 50	600 - 1500	0.5	Han et al., (2021) [19]
Back Propagation Neural Network (BPAN)	C ₇ / iC ₈ / Toluene	1 - 60	700 - 1000	1.0	Cui et al. 2022 [20]
Polynomial chaos expansion (PCE)	C ₁ – C ₇	20 - 40	700 - 900	1.0	Yousefian et al., (2022) [21]

2. MATERIALS AND METHODS

Regardless of the type of machine or deep learning project, the main steps to follow to ensure a high level of quality of an AI model are presented in Figure 1. The first step is the construction of the dataset. Depending on the problem under consideration, the recorded data could come for instance from experimental measurements, from monitored data from a process or from data generated thanks to a comprehensive physical or chemical model. It is of the utmost importance to collect reliable data. The quality of the data provided to the AI model will determine the accuracy of the final model. The second step consists in the analysis and the preparation of the data to be modelled: for instance, filling in missing values or normalizing the data.

The third step consists in the choice of a model. Whether it is supervised or unsupervised, based on regression or classification, different algorithms can be used [22,23]. In our case, K-Nearest Neighbors (KNN), Support Vector Regressor (SVR), Decision Trees (DT), Random Forest (RF), XGBoost Regressor (XGBR) and Artificial Neural Network (ANN) algorithms have been tested [24,25]. Each AI model has different hyperparameters, which characterize the model. Thus, the next step is to tune these hyperparameters and see which ones works best. After training the model, it is important to verify its performance. Eventually, the last step is the evaluation of the model using

different metric functions, typically the mean absolute error (MAE), the mean square error (MSE) or the correlation coefficient (R^2).

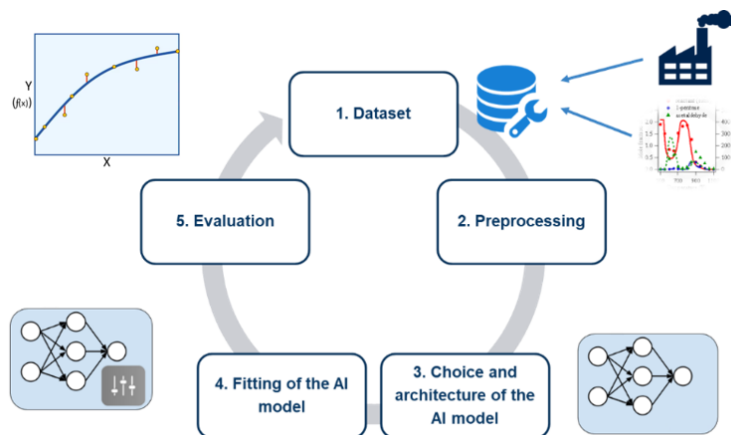


Figure 1: Workflow of a machine or deep learning project

2.1 Generation of the dataset

To obtain reliable data from the detailed kinetic simulations, different kinetic mechanisms were tested on experimental data from the literature. Five different models representing the combustion of hydrogen and alkane/air mixtures have been selected in the literature: the model of Burke et al. [26], which contains alkanes up to propane, the AramcoMech3.0 model [27], which contains alkanes up to n-butane, the model of Mehl et al. from LLNL [28], which contains alkanes up to n-heptane, the model of Bounaceur et al. [9], which contains alkanes up to n-pentane, and the most recent version of the model from University of Galway [29], which contains alkanes up to n-heptane. Several comparison between experiments and simulated results have been made and eventually the hierarchical and comprehensive combustion model developed by NUIG [29] (NUIGMech1.1) was selected based on a lower deviation between simulation and a large set of experimental ignition delay times from the literature. The model was thereafter used to calculate the ignition delay time (IDT) of different hydrogen-natural gas mixtures as a function of pressure, temperature, and equivalence ratio using ChemkinPro software [30]. Donohoe et al. [31] has studied the ignition delay times for different blends of natural gas and hydrogen blends at elevated pressures, as presented in Figure 2. The 19 input parameters for IDT estimation are presented in Table 2. The compositional range was considered by taking the average composition of usual natural gases [9]. One specification

needs to be considered: the heavier the hydrocarbon, the less it should be present in a natural gas. Thus, methane should be the major component of the fuel and NC_7H_{16} the smallest.

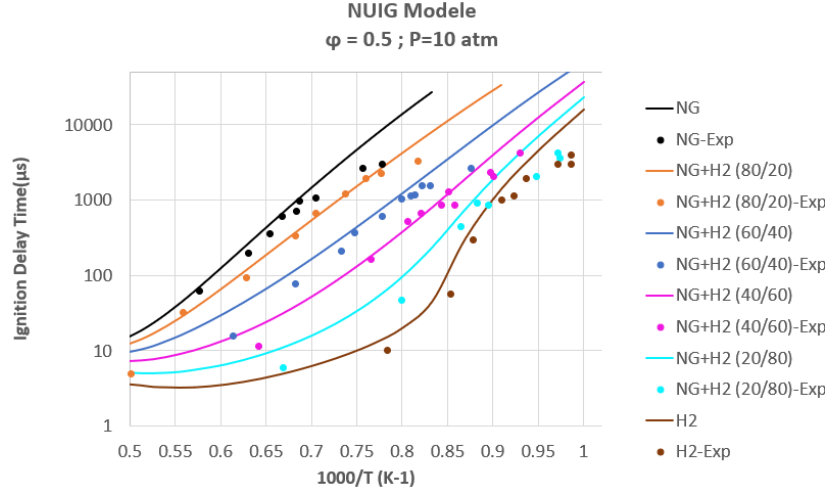


Figure 2: Characteristics of mechanisms considered and comparison experiments [31] vs simulated results with the NUIG model. NG used for experiments has the molar composition: 62.5% CH_4 / 20% C_2H_6 / 10% C_3H_8 / 5% nC_4H_{10} / 2.5% C_5H_{12} .

Table 2: Input parameters. Composition for hydrogen-natural gas blends and operating conditions

Molecules	Composition ranges [vol%]	Molecules	Composition ranges [vol%]
CH_4	[0, 100]	IC_5H_{12}	[0, 3]
C_2H_4	[0, 2.5]	$\text{NEOC}_5\text{H}_{12}$	[0, 3]
C_2H_6	[0, 16.5]	NC_6H_{14}	[0, 2.5]
C_3H_6	[0, 3]	NC_7H_{16}	[0, 1.5]
C_3H_8	[0, 10]	N_2	[0, 11.5]
C_4H_{10}	[0, 6]	CO_2	[0, 12]
IC_4H_{10}	[0, 6]	CO	[0, 2.5]
NC_5H_{12}	[0, 3]	H_2	[0, 100]
Pressure		$1 < P \text{ [atm]} < 50$	
Temperature		$200 < T \text{ [}^\circ\text{C]} < 700$	
Equivalence ratio		$0.2 < \varphi \text{ [-]} < 5$	

A numerical procedure has been developed to automatically generate the set of ignition delay times covering the range of input parameters in Table 2. This numerical procedure consists on a mixed language program (Fortran and Python) which follows these sequences:

1. Random selection of a gas composition based on Table 2

2. Random selection of a pressure, temperature and equivalence ratio based on Table 2
3. Estimation of the ignition delay time using the ChemkinPro software [15]. The chemical and thermodynamic model are taken from NUIG [14]
4. Recording of simulation data: composition, pressure, temperature, equivalence ratio, ignition delay time
5. Loop to point 1

Following this procedure, almost two months were necessary to generate the complete database.

The main risk of this approach is creating groups of data points that are similar to each other. This can be a problem because it might lead us to miss important distinctions among the data points. For instance, it is important to cover uniformly the whole domain of temperature, from 200 to 700°C, to ensure a robust model. A simple random generation of data points might lead us to create a lot of data around an average value of 450°C and less around the limits of the range of temperature, that what we name a clustering problem.

In order to avoid any clustering problem in the different successive sample points a Sobol procedure [32] has been applied. The quasi-random Sobol sequence ensures that the entire possible space of input parameters is covered uniformly. Figure 3 displays an example of samples obtained from a uniform pseudo-random distribution and a Sobol sequence for a 2-parameter sample where the sample size $N = 1000$. The Sobol sequence provides a more uniform coverage compared to the random sample. Clustering and gaps are visible for the random sample even within two dimensions. For small sample sizes, they can become quite extreme in a 19-dimensional space like the one studied here. The Sobol technique of mapping allows, with the same number of points, to have a more efficient scanning of the space to generate the dataset.

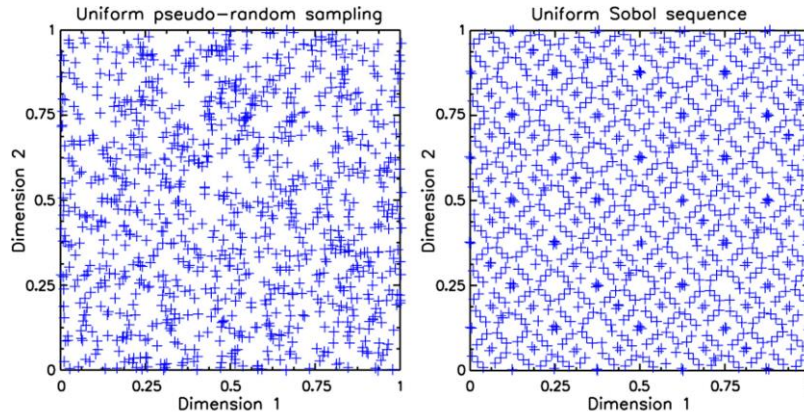


Figure 3: Comparison of distributions for different sampling strategies for a 2-parameters sample with $n = 1000$. uniform pseudo-random sample (left), sobol's quasi-random sequence sample (right)

For constant operating conditions, i.e. constant pressure, temperature and equivalence ratio, it is known that the reactivity decreases, and the IDT increases, in the order H_2 /air mixtures, natural gas/ H_2 /air, natural gas/air mixtures, CH_4 /air mixtures [31]. Another technique has then been used to avoid any clustering problem by classifying all simulated points into four different categories according to the input variables. The created categories are based on the nature of the predominant species in the compositions of the different mixtures considered (Table 3).

Table 3: Organization of the dataset

Category	Definition	Number of points
1-NG	All features and $X_{H_2}=0$	22 925
2- H_2	Only H_2	14 601
3- CH_4	Only CH_4	15 975
4-NG+ H_2	All features and $X_{H_2}\neq 0$	22 473
Full	All the dataset	75 974

The complete data set consists of a table with 75'974 rows and 20 columns. Each row corresponds to a single calculation. In each line, the first 16 columns give the composition of the gas followed by 3 columns for temperature, pressure and equivalence ratio, respectively. The last column gives the corresponding value of the ignition delay time.

Figure 4 present the distribution of the IDT for all categories. As a uniform mapping of the values is recommended to perform a better prediction, $\log(IDT)$ was used instead of IDT. In the same way, we introduced $1000/T$ instead of T , $\log(P)$ instead of P and $\log(\phi)$ instead of ϕ . Eventually, the whole dataset is composed by more than 75 000 points

involving in 19 features (16 fuel components, $1000/T$, $\log(P)$, $\log(\phi)$) and 1 target ($\log(IDT)$).

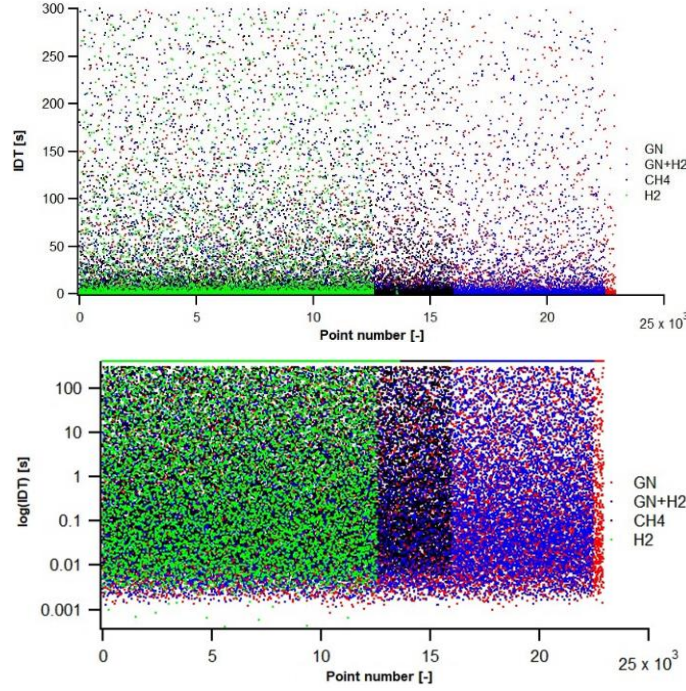


Figure 4: Distribution of the IDT. top: IDT in linear scale. bottom: $\log(IDT)$ in logarithm scale

2.2 Machine learning algorithms

Machine learning (ML) methods can be divided into two broad categories, supervised and unsupervised learning. Supervised learning requires labeled data for the training stage, i.e. data that contains both input features and desired outputs. Instead, unsupervised learning infers natural pattern within unlabeled data. All algorithms applied in this study fall into the first category, suitable for regression tasks. Some ML algorithms need a normalization. For such cases, we used a StandardScaler method [22,25] based on this formula (1):

$$\forall z \in D, \quad z^* = \frac{(z - u)}{s} \quad (1)$$

with D the data set of a column of a dataset, u the mean of the whole column and s the standard deviation, z^* is the normalized data. Different algorithms have been used to develop the AI model and are briefly described below (more information in [25]).

KNeighbors Regressor (KNN)

The k-nearest neighbor model is one of the simplest and is often used to quickly obtain results that allow to evaluate if the generation of the dataset has been done correctly: i.e. the database has enough points, no clustering problem, etc. Despite its simplicity and its fast computation time, this algorithm offers quite often good results [10]. The symbol k represents the number of neighbors used to do the prediction. This is a distance-based approach, considering that a point should “look like” its neighbors. KNN does not require any normalization.

Support Vector Regressor (SVR)

Support Vector Regression is a machine learning algorithm that seeks to identify the optimal hyperplane (or plane), which separates two sets of points in a multi-dimensional space, where each point is defined by a set of attributes. SVR does require normalization.

Decision Tree Regressor (DTR)

Decision Tree Regressor splits a large dataset into smaller ones to build a decision tree with the features and finally reach the target of a model. It is one of the most frequently used Machine Learning algorithms for solving regression as well as classification problems. DTR does not require any normalization.

Random Forest Regressor (RFR)

Random Forest is an ensemble technique capable of performing regression tasks using multiple decision trees and a technique commonly referred to as "bagging" [25]. The basic idea is to combine multiple decision trees to determine the final result rather than relying on individual decision trees. Random Forest has multiple decision trees as its basic learning models, making it a very accurate model compared to a single decision tree used in DTR. With this method, the tuning of hyperparameters could be a very long task compared to DTR. RFR does not require any normalization.

XGBoost Regressor (XGBR)

XGBoost Regressor (<https://xgboost.readthedocs.io/>) is a library, developed by community members, of methods based on gradient boosting and self-hyperparameter tuning. XGBoost provides a parallel tree boosting that solve many data science problems in a fast and accurate way. XGBR does require normalization.

Artificial Neural Network (ANN)

The Multi-Layer Perceptron is based on a construction of a neural network capable of generating outputs with a set of inputs. Links are created between neurons to symbolize relationships and influences between the features and the targets. The principle of backpropagation is used. It means that the link between the neurons and their weights in the final contribution is updated until reaching a stopping criterion. MLPRegressor does require a normalization. As a reminder, for each case where normalization was required, the StandardScaler method was used.

2.3 Performance Indicators

The predictive ability of an ML model is commonly evaluated in regression problems thanks to three parameters used to compare predicted ignition delay times and expected ignition delay in the testing set. The first parameter is the correlation coefficient (R^2) described in Eq (2). A value of R^2 close to 1 indicates a strong accuracy of the model. The second and third parameter are the Mean Absolute Error (MAE) and the Mean Square Error (MSE) (eq (3) and (4)), which must be as close as possible to zero to indicate a good prediction of the model.

$$R^2 = 1 - \frac{\sum (y_i - \hat{y}_i)^2}{\sum (y_i - \bar{y})^2} \text{ with } \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i \quad (2)$$

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (3)$$

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (4)$$

where N is the total amount of data in the dataset, y_i is the i 'th expected value in the dataset and $\hat{y}_i$ is the corresponding predicted value.

Another common performance indicator is the parity plots, which consist in the representation of target values predicted by the AI model as a function of target values coming from the dataset. This representation is very useful because it gives a quick visual representation of the discrepancy of the data. Last, as the IDT of a mixture depends on the temperature, the pressure and the equivalence ratio, we have compared for specified mixture at a fixed pressure and equivalence ratio the variation of IDT versus temperature

estimated using the original NUIG chemical model compared to values coming from the AI model.

3. RESULTS AND DISCUSSION

The following work was realized using the language Python 3.8.13. and the ML library Sklearn 1.0.2 [25]. Each ML models are characterized by different hyperparameters. To guarantee the best performance of a model, it is highly recommended to tune its hyperparameters. The library Sklearn provides a large number of models with hyperparameters set to their default value, and depending on the type of ML algorithm, different kind of hyperparameter could to be tuned. For the KNN model, the number of neighbors k is one of the possible hyperparameters to optimize. For the ANN model, the number of layers, number of neurons per layer and the activation function are hyperparameter, which can to be tuned. Since the AI algorithms are black boxes, it is not easy to immediately know the effect of a parameter on the final prediction. For this reason, the only way to have access to the optimized parameters for a model is to test for every discrete combination of the value of the parameter, which can be a very time-consuming task. Different Python libraries for serial and parallel optimization over awkward search spaces, which may include real-valued, discrete, and conditional dimensions have been developed during the last decade to tuned “automatically” a large set of hyperparameters. Hypertopt [33] and BayesSearchCV [34] are libraries for hyperparameter tuning based on Bayesian optimization. Optuna [35] is another automatic hyperparameter optimization software, especially designed for machine learning. All three are complex to implement and, to the best of our knowledge, are difficult to couple with an ANN model. To overcome this problem, it is more interesting to grid the parameters by testing many parameters at the same time with a lot of values. It is possible to automate this by using the library python Sklearn GridSearchCV. In addition to testing all the different value of the hyperparameter, this function proceeds with a cross validation to avoid an overfitting of the AI model. For all AI models tested, we used a dataset split factor of 70% for the training set (which corresponds to 53'182 points) and 30% for the test set (which correspond to 22'792 points) and a cross validation of five folds as represented by Figure 5. In this figure, each fold contains around 10'636 points.

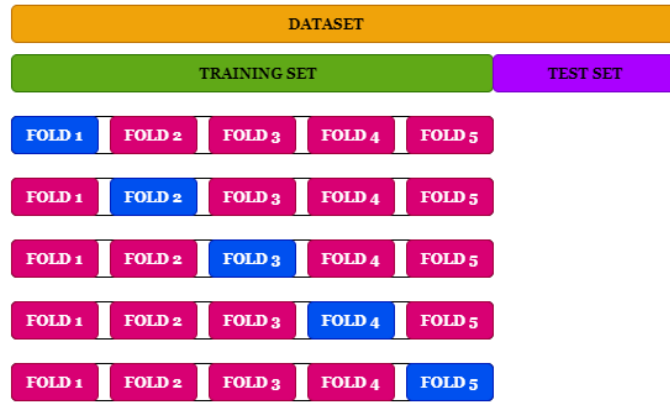


Figure 5: Cross validation representation with 5 folds

All the algorithms presented above have been tested with the dataset to determine the best model which represent correctly values of predicted ignition delay time compare with those estimated through the detailed kinetic model and ChemkinPro. To make a decision, the metrics are not the only parameters: the capacity of reproducing the physical behavior for selected configuration have been also tested. Table 4 show the values of the different metrics obtained for all the AI model after several runs. Although the correlation coefficient of all models (except the KNN model) is close to 1, it seems that parity plots show a poor representation of data by the models, with the exception of the ANN model. Then, we have decided to focus on the ANN model and develop numerical strategy to improve the accuracy of the ANN model.

Table 4: Metrics of all the models

	R^2 train	R^2 test	MSE train	MSE Test
KNN	0.6141	0.8286	0.5588	1.0249
SVR	0.9943	0.9901	0.1005	0.1334
DT	0.9994	0.9812	0.0832	0.1837
RF	0.9991	0.9941	0.1026	0.0667
XGBoost	0.9976	0.9952	0.0645	0.0660
ANN	0.9982	0.9979	0.0570	0.9979

3.1 Optimization of the model

The function GridsearchCV with a cross validation of 5 folds was used to determine the best architecture for the ANN model by testing several hidden layers (from 1 to 10) and several number of neurons per layer (from 2^5 to 2^7). The number of iteration (max_iter),

the activation function (activation), the solver (solver) and the learning rate (learning_rate) have also to be tuned. The number of folds in a cross-validation test can also be optimized. A value of five folds is commonly used in the literature. Some tests with a cross-validation of 10 folds have been performed without affecting the accuracy of the model. Due to confidentiality reasons, this information could not be disclosed in detail. The representation of a tested ANN model is representing in Figure 6 with the list of hyperparameters tuned given in Table 5.

Table 5: List of tuned hyperparameters for ANN model

Hyperparameter
hidden_layer_sizes
max_iter
Activation function
solver
Learning_rate

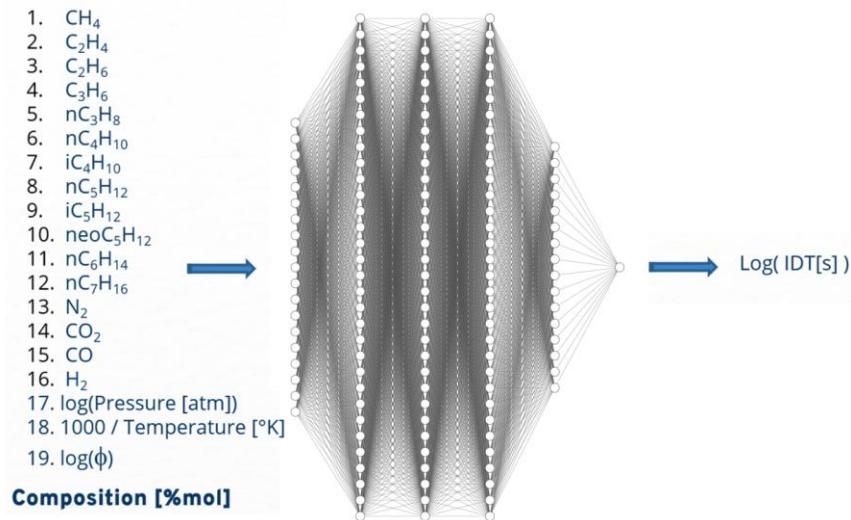


Figure 6: Representation of an ANN model tested in this study

Table 6 summarizes the performance indicators of the ML model. Note that the correlation coefficient R^2 of both trained and test sets are close to each other and to 1. It means that the model is providing a good prediction and very limited over-fitting. The MSE and MAE are very low, which is of course expected and desirable.

Table 6: Performance Indicator

Dataset	R²	MSE	MAE
<i>Train</i>	0.9994	0.0339	0.0234
<i>Test</i>	0.9993	0.0361	0.0241

The overall performance of the ANN model to predict the IDT is displayed in the parity plot in Figure 7. On the whole domain, the predicted ignition delay times matches very well with the reference ignition delay with a deviation lower than 2%. The performance of the ANN model is also validated for the prediction of the Negative Temperature Coefficient (NTC) behavior for natural gases containing a higher amount of alkanes larger than ethane. Figure 8 shows the results of the validation for different natural gas/H₂ blends and inlet pressures over the range of inlet temperature at different equivalence ratios. The simulated dataset from the detailed kinetic model is well reflected by the AI model results. In fact, the data displayed in Figure 8 has a maximum of 2% inaccuracy. Among the cases selected for display are pure CH₄ and H₂ (respectively on lower left and lower right) as the first pillars for validation. Next, the results of the AI model were compared to natural gas and hydrogen blended in natural gas in order to represent typical gas turbine fuel use.

By following the ASTM E 659-78, mixtures, which do not ignite within 5 min, are considered to be below the minimum ignition temperature. The simulation considers a volume of 200 cm³ with a heat loss of 19 W/m². The red line in Figure 8 represents this zone with values above 300 s, which correspond to a value of $\log(300) \approx 2.48$. Thus, if the ANN model predicts a IDT value greater than 300 s, the reacting mixture is in the non-ignition region and the result should not be considered.

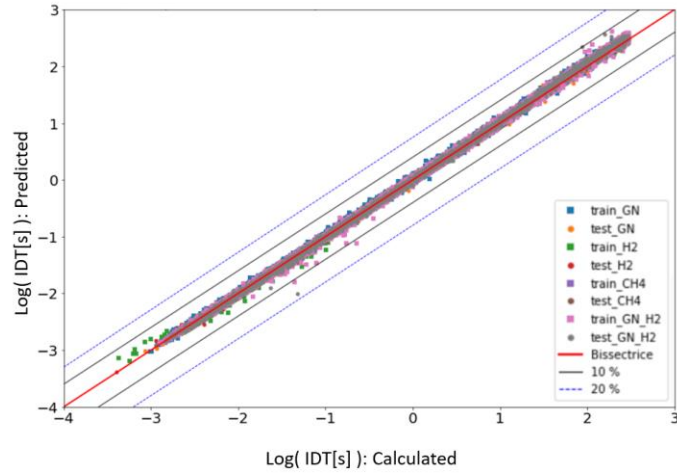
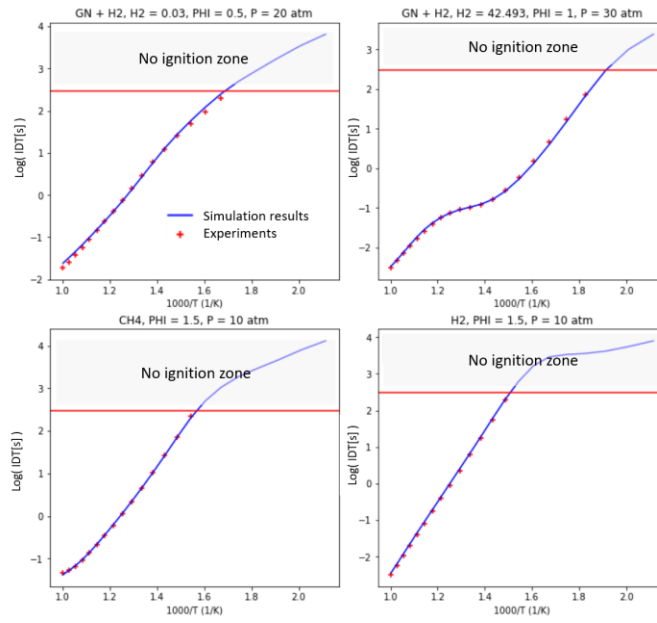


Figure 7: Parity plot for the ANN model



	GN+H2 (%vol)	
	H2=0.03	H2=42.493
CH4	93.73	49.044
C2H4	0	0.555
C2H6	2.75	1.471
C3H6	0	0.33
C3H8	0.77	1.697
C4H10	0.21	0.8
IC4H10	0.09	0.299
NC5H12	0.04	0.208
IC5H12	0.03	0.146
NEOC5H12	0	0.297
NC6H14	0.03	0.581
NC7H16	0.01	0.279
N2	1.41	0.795
CO2	0.87	0.772
CO	0.01	0.234
H2	0.03	42.493

Figure 8: comparison of IDT estimated through the NUIG kinetic model (red points) and IDT predicted by the ANN model (blue line)

3.2 Test of robustness on the ANN model

Various robustness tests were performed to assess the sensitivity of the IDT result when predictions are made outside the validity domain, i.e. out of the ranges of parameters displayed in Table 2. Figure 9 shows some different robustness tests with a composition outside the range, a pressure above the maximum authorized and an equivalence ratio below the limit. In general, the prediction from the ANN model is still applicable when pressure or equivalence ratio are outside the defined domain. This is not the case with the composition. Alkanes involving more than 3 carbon atoms have a strong Negative Temperature Coefficient (NTC) behavior as a function of the temperature, which has not been taken into account in the AI model. The model focuses indeed on natural gas which contains only traces of alkanes heavier than ethane. It is therefore expected that the AI model does not work outside the validation domain. Adding more data in the same range of composition or using deep learning algorithms would not change the observed deviation unless using a reinforcement model approaches, what is out of scope of this study.

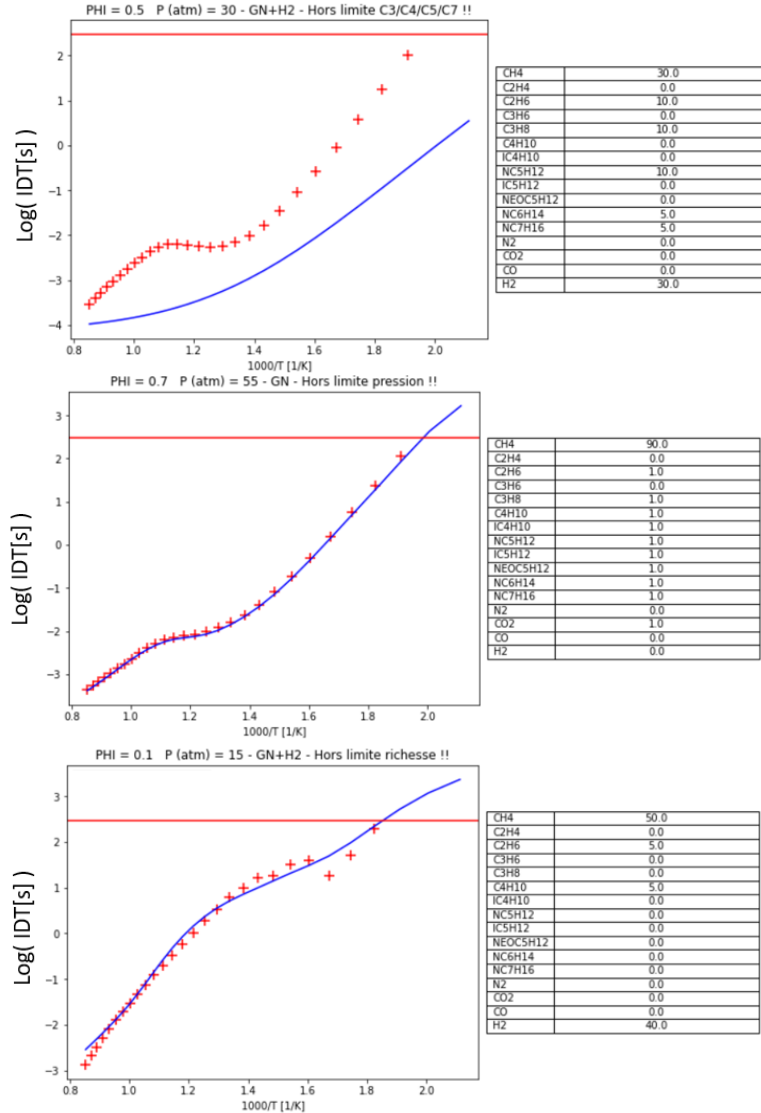


Figure 9: Robustness tests outside the domain. (up) over the range composition (middle) over the pressure (down) over the equivalence ratio. Red points: estimated through the kinetic model. Blue line: estimated with the ANN model

4. CONCLUSION

An AI model based on a Multi-Layer Perceptron Regressor also named Artificial Neural Network has been developed in order to predict the IDT of different natural gas/hydrogen/air blends at different pressures, temperatures and equivalence ratios.

The final AI model shows a high degree of robustness in reproducing the results of the detailed model as shown by different metric functions. For both the train and test dataset, the average value of the correlation coefficient, 99.91%, was found above: the Mean Absolute Error (MAE) around 0.03 and the Root Mean Square Error (RMSE) lower than

0.04. On the whole domain, the predicted ignition delay times match very well with the ignition delay calculated with the detailed kinetic mechanism with a deviation lower than 2%. Several tests have been made to check the robustness of the model by comparing predicted and simulated values outside the ranges of conditions used to train the AI model. The different results show that even with an estimated value outside the pressure, temperature or equivalence ratio range gave correct results. This is not the case when considering a process gas with a quantity of larger alkanes, such as n-hexane or n-heptane, above the authorized limit.

It is interesting to note that one reason of this project was motivated in particular by the idea of reducing the time of computing the IDTs. From 5 min with the NUIGmech1.1 model to at worst 1 s for the ANN model, the estimation of an IDT set is now easier and doable for a large number of configurations. In fact, it eventually opens the way to coupling with broader multi-physical tools, e.g. CFD codes, enabling prediction of flammable mixture and likelihood of ignitability in transient applications of safety scenarios.

The next steps will include testing other methods such as deep learning to improve the simulation at the boundary conditions of the concentration domain and for higher alkane contents. The model will also be extended to simulate other safety criteria such as minimum ignition temperatures, and include other fuels such as ammonia.

REFERENCES

- [1] Chiesa, P., Lozza, G., and Mazzocchi, L., 2005, "Using Hydrogen as Gas Turbine Fuel," *Journal of Engineering for Gas Turbines and Power*, **127**(1), pp. 73–80.
- [2] Bothien, M. R., Ciani, A., Wood, J. P., and Fruechtel, G., 2019, "Toward Decarbonized Power Generation with Gas Turbines by Using Sequential Combustion for Burning Hydrogen," *Journal of Engineering for Gas Turbines and Power*, **141**(12).
- [3] Nemitallah, M. A., Haque, M. D. A., Hussain, M., Abdelhafez, A., and Habib, M. A., 2022, "Stratified and Hydrogen Combustion Techniques for Higher Turndown and Lower Emissions in Gas Turbines," *J. Energy Resour. Technol.-Trans. ASME*, **144**(2), p. 020801.

- [4] Molière, M., 2004, "Hydrogen Fueled Gas Turbines: Status and Prospects," *2nd CAME-GT Conference*, Bled, Slovenia.
- [5] Campbell, A., Goldmeer, J., Healy, T., Washam, R., Molière, M., and Citenò, J., 2009, "Heavy Duty Gas Turbines Fuel Flexibility," *Paper GT2008-51368*, Proceedings of the ASME Turbo Expo 2008: Power for Land, Sea, and Air.
- [6] Chong, C. T., Ng, J.-H., Aris, M. S., Mong, G. R., Shahril, N., Ting, S. T., and Zulkifli, M. F., 2019, "Impact of Gas Composition Variations on Flame Blowout and Spectroscopic Characteristics of Lean Premixed Swirl Flames," *Process Saf. Environ. Protect.*, **128**, pp. 1–13.
- [7] Lu, M., Li, D., Xie, K., Sun, G., and Fu, Z., 2023, "Investigation of Flame Evolution and Stability Characteristics of H₂-Enriched Natural Gas Fuel in an Industrial Gas Turbine Combustor," *Fuel*, **331**, p. 125938.
- [8] Pekalski, A. A., Terli, E., Zevenbergen, J. F., Lemkowitz, S. M., and Pasman, H. J., 2005, "Influence of the Ignition Delay Time on the Explosion Parameters of Hydrocarbon–Air–Oxygen Mixtures at Elevated Pressure and Temperature," *Proceedings of the Combustion Institute*, **30**(2), pp. 1933–1939.
- [9] Bounaceur, R., Glaude, P.-A., Sirjean, B., Fournet, R., Montagne, P., Vierling, M., and Molière, M., 2016, "Prediction of Auto-Ignition Temperatures and Delays for Gas Turbine Applications," *J. Eng. Gas Turbines Power*, **138**(2), p. 021505.
- [10] Strohle, J., and Myhrvold, T., 2007, "An Evaluation of Detailed Reaction Mechanisms for Hydrogen Combustion under Gas Turbine Conditions," *Int. J. Hydrog. Energy*, **32**(1), pp. 125–135.
- [11] Sun, Y., Zhang, Y., Huang, M., Li, Q., Wang, W., Zhao, D., Cheng, S., Deng, H., Du, J., Song, Y., Li, H., and Xu, H., 2022, "Effect of Hydrogen Addition on the Combustion and Emission Characteristics of Methane under Gas Turbine Relevant Operating Condition," *Fuel*, **324**, p. 124707.
- [12] Carpio, J., Li, B., Fernandez-Galisteo, D., Sanchez, A. L., and Williams, F. A., 2023, "Systematically Derived One-Step Kinetics for Hydrogen-Air Gas-Turbine Combustion," *Combust. Flame*, **250**, p. 112633.
- [13] Elliott, L., Ingham, D. B., Kyne, A. G., Mera, N. S., Pourkashanian, M., and Wilson, C. W., 2005, "The Use of Ignition Delay Time in Genetic Algorithms Optimisation of

Chemical Kinetics Reaction Mechanisms,” *Eng. Appl. Artif. Intell.*, **18**(7), pp. 825–831.

- [14] Abbas, T., Awais, M. M., and Lockwood, F. C., 2003, “An Artificial Intelligence Treatment of Devolatilization for Pulverized Coal and Biomass in Co-Fired Flames,” *Combust. Flame*, **132**(3), pp. 305–318.
- [15] Bhattacharya, A., and Majumdar, P., “Artificial Intelligence-Machine Learning Algorithms for the Simulation of Combustion Thermal Analysis,” *Heat Transf. Eng.*
- [16] Darwish Ahmad, A., Akafuah, N. K., Forthofer, J., Fuchihata, M., Hirasawa, T., Kuwana, K., Nakamura, Y., Sekimoto, K., Saito, K., and Williams, F. A., 2023, “Large-Scale Fire Whirl and Forest Fire Disasters: Awareness, Implications, and the Need for Developing Preventative Methods,” *Front. Mech. Eng.-Switzerland*, **9**, p. 1045542.
- [17] Kuppa, K., Goldmann, A., and Dinkelacker, F., 2018, “Predicting Ignition Delay Times of C1-C3 Alkanes/Hydrogen Blends at Gas Engine Conditions,” *Fuel*, **222**, pp. 859–869.
- [18] Cui, Y., Wang, Q., Liu, H., Zheng, Z., Wang, H., Yue, Z., and Yao, M., 2020, “Development of the Ignition Delay Prediction Model of N-Butane/Hydrogen Mixtures Based on Artificial Neural Network,” *Energy and AI*, **2**, p. 100033.
- [19] Han, W., Sun, Z., Scholtissek, A., and Hasse, C., 2021, “Machine Learning of Ignition Delay Times under Dual-Fuel Engine Conditions,” *Fuel*, **288**, p. 119650.
- [20] Cui, Y., Liu, H., Wang, Q., Zheng, Z., Wang, H., Yue, Z., Ming, Z., Wen, M., Feng, L., and Yao, M., 2022, “Investigation on the Ignition Delay Prediction Model of Multi-Component Surrogates Based on Back Propagation (BP) Neural Network,” *Combustion and Flame*, **237**, p. 111852.
- [21] Yousefian, S., Bourque, G., and Monaghan, R. F., 2022, “Metamodelling of Ignition Delay Time for Natural Gas Blends Under Gas Turbine Operating Conditions,” *Turbo Expo: Power for Land, Sea, and Air*, American Society of Mechanical Engineers, p. V03BT04A014.
- [22] James, G., Witten, D., Hastie, T., and Tibshirani, R., 2013, *An Introduction to Statistical Learning*, Springer.
- [23] Ng, A., 2015, “What Data Scientists Should Know about Deep Learning,” *Speech Presented at Extract Data Conference, November*.

- [24] Chollet, F., 2018, "Deep Learning with Python (Shelter Island, New York.)"
- [25] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., and Dubourg, V., 2011, "Scikit-Learn: Machine Learning in Python," the Journal of machine Learning research, **12**, pp. 2825–2830.
- [26] Burke, U., Metcalfe, W. K., Burke, S. M., Heufer, K. A., Dagaut, P., and Curran, H. J., 2016, "A Detailed Chemical Kinetic Modeling, Ignition Delay Time and Jet-Stirred Reactor Study of Methanol Oxidation," Combustion and Flame, **165**, pp. 125–136.
- [27] Li, Y., Zhou, C.-W., Somers, K. P., Zhang, K., and Curran, H. J., 2017, "The Oxidation of 2-Butene: A High Pressure Ignition Delay, Kinetic Modeling Study and Reactivity Comparison with Isobutene and 1-Butene," Proceedings of the Combustion Institute, **36**(1), pp. 403–411.
- [28] Mehl, M., Pitz, W. J., Westbrook, C. K., and Curran, H. J., 2011, "Kinetic Modeling of Gasoline Surrogate Components and Mixtures under Engine Conditions," Proceedings of the Combustion Institute, **33**(1), pp. 193–200.
- [29] Mohamed, A. A. E.-S., Panigrahy, S., Sahu, A. B., Bourque, G., and Curran, H. J., 2021, "An Experimental and Kinetic Modeling Study of the Auto-Ignition of Natural Gas Blends Containing C1–C7 Alkanes," Proceedings of the Combustion Institute, **38**(1), pp. 365–373.
- [30] Kee, R. J., Rupley, F. M., and Miller, J. A., 1993, "CHEMKIN-II: A FORTRAN Chemical Kinetics Package for the Analysis of Gas-Phase Chemical Kinetics."
- [31] Donohoe, N., Heufer, A., Metcalfe, W. K., Curran, H. J., Davis, M. L., Mathieu, O., Plichta, D., Morones, A., Petersen, E. L., and Güthe, F., 2014, "Ignition Delay Times, Laminar Flame Speeds, and Mechanism Validation for Natural Gas/Hydrogen Blends at Elevated Pressures," Combustion and Flame, **161**(6), pp. 1432–1443.
- [32] Sobol', I. M., 1967, "On the Distribution of Points in a Cube and the Approximate Evaluation of Integrals," U.S.S.R. Comput. Math. Math. Phys., **7**(4), pp. 86–112.
- [33] Bergstra, J., Yamins, D., and Cox, D., 2013, "Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures," *International Conference on Machine Learning*, PMLR, pp. 115–123.

- [34] “Tuning a Scikit-Learn Estimator with Skopt — Scikit-Optimize 0.8.1 Documentation” [Online]. Available: https://scikit-optimize.github.io/stable/auto_examples/hyperparameter-optimization.html. [Accessed: 20-Feb-2023].
- [35] Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M., 2019, “Optuna: A next-Generation Hyperparameter Optimization [25,” *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2623–2631.