



HAL
open science

Solving the population balance equation for non-inertial particles dynamics using probability density function and neural networks: Application to a sooting flame

Andréa Seltz, Pascale Domingo, Luc Vervisch

► To cite this version:

Andréa Seltz, Pascale Domingo, Luc Vervisch. Solving the population balance equation for non-inertial particles dynamics using probability density function and neural networks: Application to a sooting flame. *Physics of Fluids*, 2021, 33 (1), pp.013311. 10.1063/5.0031144 . hal-03116162

HAL Id: hal-03116162

<https://normandie-univ.hal.science/hal-03116162>

Submitted on 20 Jan 2021

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.

Neural networks for solving population balance equation

1 **Solving the population balance equation for non-inertial particles dynamics using**
2 **probability density function and neural networks: Application to a sooting flame**

3 Andrea Seltz,¹ Pascale Domingo,¹ and Luc Vervisch^{1, a)}

4 *CORIA - CNRS, Normandie Université, INSA de Rouen, Technopole du Madrillet, BP 8,*
5 *76801 Saint-Etienne-du-Rouvray, France*

6 Numerical modeling of non-inertial particles dynamics is usually addressed by solving
7 a population balance equation (PBE). In addition to space and time, a discretisation is
8 required also in the particle-size space, covering a large range of variation controlled
9 by strongly nonlinear phenomena. A novel approach is presented in which a hybrid
10 stochastic/fixed-sectional method solving the PBE is used to train a combination of an arti-
11 ficial neural network (ANN) with a convolutional neural network (CNN) and recurrent long
12 short-term memory artificial neural layers (LSTM). The hybrid stochastic/fixed-sectional
13 method decomposes the problem into the total number density and the probability density
14 function (PDF) of sizes, allowing for an accurate treatment of surface growth/loss. After
15 solving for the transport of species and temperature, the input of the ANN is composed
16 of the thermochemical parameters controlling the particle physics and of the increment in
17 time. The input of the CNN is the shape of the particle size distribution (PSD) discretised
18 in sections of size. From these inputs, in a flow simulation the ANN-CNN returns the PSD
19 shape for the subsequent time step or a source term for the Eulerian transport of the particle
20 size density. The method is evaluated in a canonical laminar premixed sooting flame of the
21 literature and for a given level of accuracy (i.e., a given discretisation of the size space),
22 a significant computing cost reduction is achieved (6 times faster compared to a sectional
23 method with 10 sections and 30 times faster for 100 sections).

^{a)}Electronic mail: vervisch@coria.fr

Neural networks for solving population balance equation

24 I. INTRODUCTION

25 Laminar or turbulent flows carrying non-inertial particles are at the core of many chemical and
26 mechanical engineering processes. On the numerical modeling side, the solving of a population
27 balance equation (PBE) for simulating the time evolution of the particle size distribution (PSD) is
28 usually the starting point to tackle particle dynamics.¹⁻⁴

29 Three major difficulties arise when dealing with population balance equations. The first one
30 lies in the physical modeling of nucleation, surface growth/loss, agglomeration/coagulation and
31 sometimes breakage of the particles. Depending on the problem addressed, these phenomena can
32 involve quite complex thermophysical properties of the solid and liquid or/and gaseous phases at
33 play.

34 The second difficulty results from the large range of sizes covered by the particles, making
35 the discretisation in size space uneasy to predict both the distribution of size and its moments
36 accurately.

37 The third difficulty comes from the numerical solving of the particle size distribution itself.
38 The physics behind the terms of the PSD equation can be strongly non-linear, with a PSD time
39 evolution prone to spurious oscillations and divergence. For instance, the growth/loss of particle
40 surface takes the form of a non-linear convective effect of the PSD in size space (i.e., surface
41 growth/loss transport the PSD towards larger/smaller sizes without modification of its shape).
42 Thereby, because of growth/loss, it may be challenging to secure stability of the solution keeping
43 a minimum level of spurious diffusion of the PSD in size space.⁵ Similarly, properly predicting
44 the moments of the particle size distribution in an agglomeration/coagulation process may require
45 a specific numerical treatment.⁶

46 A large variety of methods for simulating the time evolution of PSD have been discussed in the
47 literature and applied with success, including advanced numerical strategies proposed recently to
48 address the issues mentioned above.⁷⁻¹³ However for many of these approaches, securing accu-
49 racy in the PSD shape prediction goes with a large computing effort, jeopardising the systematic
50 application of the most precise methods to three-dimensional and unsteady simulations for vir-
51 tual prototyping of real systems. Advanced methods for solving the PSD have been applied to
52 canonical problems featuring low or moderate Reynolds numbers studied with direct numerical
53 simulation (DNS),¹⁴⁻¹⁶ while simplified approaches are usually developed to deal with complex
54 three-dimensional flows, specifically in the context of particulate emissions from burners and com-

Neural networks for solving population balance equation

55 bustion systems.^{17–19}

56 To progress in this direction, this paper reports an attempt in which the effort made in grid
57 size resolution and precision of PSD solving, serves to train a set of neural networks for solving
58 the population balance equation. Neural networks are subsequently coupled with the flow solu-
59 tion, thus allowing for a significant reduction of the computing cost, still preserving the accuracy
60 provided by the numerical method generating the training database.

61 The PBE takes the form of a balance equation for $n(v^*; \underline{x}, t)$, the number of particles per unit of
62 flow volume and per unit size (or particles number density), where the characteristic particle size
63 is denoted v^* . The generic form of functionals used in flow physics was first studied by Hopf^{20,21}
64 in the 50's. Then in the 70's, O'Brien and his team popularised the concept of probability density
65 functions (PDF) for turbulent scalar mixing studies.^{22–24} Recently, Bouaniche et al.⁵ have dis-
66 cussed how solving the PBE can benefit from the knowledge on PDF as discussed by O'Brien,
67 when he pioneered theoretical and computational approaches in line with PDF for scalars (temper-
68 ature, mass fractions, etc.). Indeed, $n(v^*; \underline{x}, t)$ may be decomposed into $N_T(\underline{x}, t)$, the total number
69 of particles per unit of volume and $\bar{P}(v^*; \underline{x}, t)$, the PDF of particle size

$$70 \quad n(v^*; \underline{x}, t) = N_T(\underline{x}, t) \bar{P}(v^*; \underline{x}, t). \quad (1)$$

71 Then, instead of solving for $n(v^*; \underline{x}, t)$, it is preferred to solve for both $N_T(\underline{x}, t)$ and $\bar{P}(v^*; \underline{x}, t)$, to
72 ease the treatment of the non-linear surface growth/loss term with a Monte Carlo solution for the
73 PDF.

74 The training of the neural networks is performed along these lines with the recently proposed
75 hybrid stochastic-sectional approach.⁵ This hybrid technique combines Monte Carlo and fixed-
76 sectional methods, to minimise the discretisation errors when solving the surface growth/loss term
77 of the population balance equation, and to compute agglomeration using a sectional formalism.
78 It relies on a fixed number of stochastic particles and sections, with a numerical algorithm min-
79 imising errors, even for a moderate number of stochastic particles. This method was previously
80 validated against canonical PSD evolutions for which analytical solution exist.⁵ It was also applied
81 to a sooting flame, in order to further analyse the relation between the mobility diameter, measured
82 in the experiments, and the equivalent sphere diameter, introduced in the modeling. The influence
83 of the fractal particle shape on the simulated particle size distribution was also explored.²⁵

84 The introduction of artificial neural networks in reactive flow simulation is a rapidly growing
85 field, to reduce both computing cost and memory requirements. The optimisation of networks

Neural networks for solving population balance equation

86 has received attention in chemistry tabulation^{26–32} and also to develop data driven modeling^{33–40}
87 or again to analyse experimental measurements^{34,41} and to propose digital-twins for process
88 control.⁴² Works have been specifically devoted to the application of neural networks to study
89 and to model the dynamics of particles in flows and their size distributions. In this context, the
90 improvement of physical models driving the PBE through ANN have been discussed,⁴³ control
91 particle size strategies using ANN have been implemented,⁴⁴ particle size distributions have been
92 approximated with ANN,⁴⁵ measurements of particle size distributions have been coupled with
93 ANN,⁴⁶ memory effects were introduced in networks with feedback connections to progress in
94 Gaussian moment methods⁴⁷ and dynamic modeling of component formation was coupled with a
95 special class of ANN (regularisation networks), in order to predict the solid-liquid equilibrium in
96 complex multi-phase flows.⁴⁸

97 Here, a combination of artificial neural networks (ANN) and convolutional neural networks
98 (CNN)^{49–51} is discussed. These ANN and CNN are both augmented with one additional recur-
99 rent long short-term memory artificial neural layer (LSTM).^{52–54} The balance equations for ther-
100 mochemistry are solved in a regular manner. Then on one hand, the ANN part deals with the
101 thermochemical parameters of the PSD evolution (chemical species mass fractions, temperature,
102 etc.), i.e., the inputs of the physical models driving particle nucleation, surface growth/loss and
103 agglomeration/coagulation. On the other hand, the shape of the PSD is analysed by the CNN,
104 which is specialised for image segmentations tasks, thus well adapted to predict a distribution. To
105 better capture the path followed by the solution of the non-linear problem, the LSTM allows for
106 storing information from previous time-steps in order to facilitate the optimization of the ANN-
107 CNN. Once ANN and CNN concatenated, the vector of thermochemical parameters, the particle
108 size distribution and the increment in time δt , serve as input to the neural networks, which returns
109 the PSD at time $t+\delta t$, thus saving a large amount of the computing time usually devoted to solving
110 for the homogeneous carrier-phase part of the population balance equation (i.e., nucleation, sur-
111 face growth/loss and agglomeration/coagulation of the particles). In practice, the number density
112 of particles is discretised over a set of sections, each section is transported by the turbulent flow
113 solving for convection in the eulerian context, while the neural networks solve for the rest of the
114 particle physics.

115 In the present work, a canonical laminar one-dimensional rich sooting premixed ethylene flame
116 is considered for training the neural networks to predict carbon particulate emissions. Both a priori
117 and a posteriori tests are reported. Well-established physical models are applied, the objective

This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.

PLEASE CITE THIS ARTICLE AS DOI:10.1063/1.50031144

Neural networks for solving population balance equation

118 being to demonstrate the potential of the approach under a given physical/chemical modeling
 119 framework.

120 The population balance equation formalism is recalled in the next section along with the main
 121 lines of the hybrid stochastic fixed-sectional method for solving the PBE, before summarising the
 122 various physical models introduced in this work for sooting flames. In the subsequent section, the
 123 machine learning procedure and its training over the laminar premixed sooting flame is discussed.
 124 Finally, the results of a posteriori tests are presented.

125 II. POPULATION BALANCE EQUATION MODELING

126 Let us denote $n(v; \underline{x}, t)$ the number of particles of characteristic size v (in terms of volume or
 127 mass, v is a continuous independent variable), per unit of flow volume and per unit of characteris-
 128 tic size, of an aerosol submitted to simultaneous nucleation, surface variation and agglomeration.
 129 $n(v; \underline{x}, t)$ follows a population balance equation, an integro-partial-differential equation of the hy-
 130 perbolic type:⁴

$$131 \quad \frac{\partial n(v; \underline{x}, t)}{\partial t} + \underline{u} \cdot \nabla n(v; \underline{x}, t) = - \frac{\partial}{\partial v} [G(v; \underline{x}, t)n(v; \underline{x}, t)] + \dot{A}(v; \underline{x}, t), \quad (2)$$

132 with \underline{u} the velocity vector. $G(v; \underline{x}, t) > 0$ is the surface growth rate or $G(v; \underline{x}, t) < 0$ the surface loss
 133 rate. This growth/loss rate may depend on both v , the particle size, and (\underline{x}, t) , the position and
 134 time, because the surface kinetics is usually controlled by the local gaseous/liquid environment.
 135 The RHS contains a source term $\dot{A}(v; \underline{x}, t)$, which is composed of $\dot{h}(v) > 0$ (respectively $\dot{h}(v) < 0$)
 136 the nucleation (respectively disappearance) rate and an integral which accounts for agglomeration
 137 following the continuous counterpart of the so-called Smoluchowski equation, with $\beta(v, v^+)$ the
 138 collision kernel for two particles of volume v and v^+ , hence

$$139 \quad \begin{aligned} \dot{A}(v; \underline{x}, t) = & \dot{h}(v; \underline{x}, t) + \frac{1}{2} \int_0^v \beta(v - v^+, v^+) n(v - v^+; \underline{x}, t) n(v^+; \underline{x}, t) dv^+ \\ & - n(v; \underline{x}, t) \int_0^\infty \beta(v, v^+) n(v^+; \underline{x}, t) dv^+. \end{aligned} \quad (3)$$

141 From Eq. (2), the total number of particles per unit of volume

$$142 \quad N_T(\underline{x}, t) = \int_0^\infty n(v; \underline{x}, t) dv, \quad (4)$$

Neural networks for solving population balance equation

143 evolves according to

$$144 \quad \frac{\partial N_T(\underline{x}, t)}{\partial t} + \underline{u} \cdot \nabla N_T(\underline{x}, t) = \int_0^{\infty} \dot{A}(v; \underline{x}, t) dv. \quad (5)$$

145 A. Hybrid stochastic/sectional PBE solving with PDF

146 From the relation $n(v^*; \underline{x}, t) = N_T(\underline{x}, t) \bar{P}(v^*; \underline{x}, t)$, between the particle size density, the total
147 particle density and $\bar{P}(v^*; \underline{x}, t)$ the PDF of size, using Eqs. (2) and (5), the balance equation for
148 $\bar{P}(v^*; \underline{x}, t)$ is derived, where v^* denotes the sample space (or the stochastic variable) associated to
149 the physical particle size v ,

$$150 \quad \frac{\partial \bar{P}(v^*; \underline{x}, t)}{\partial t} + \underline{u}(\underline{x}, t) \cdot \nabla \bar{P}(v^*; \underline{x}, t) = - \overbrace{\frac{\partial}{\partial v^*} [G(v^*; \underline{x}, t) \bar{P}(v^*; \underline{x}, t)]}^{(i)} \\ 151 \quad + \underbrace{\dot{B}(v^*; \underline{x}, t) - \bar{P}(v^*; \underline{x}, t) \int_0^{\infty} \dot{B}(v^+; \underline{x}, t) dv^+}_{(ii)}. \quad (6)$$

152 The term (i) is the convection of the PDF in size space according to surface growth or loss (Fig. 1(a))
153 illustrates the PDF evolution with $G > 0$). The term (ii) is a source/sink whose amplitude is
154 driven by nucleation and agglomeration of the particles according to the rate given by $\dot{B}(v^*; \underline{x}, t) =$
155 $\dot{A}(v^*; \underline{x}, t)/N_T(\underline{x}, t)$ (Fig. 1(b)). This second term is decomposed in two parts: a local decrease or
156 increase of the probability according to the sign of $\dot{B}(v^*; \underline{x}, t)$ and a redistribution, proportional to
157 the probability of finding size v^* , of the sum of $\dot{B}(v^*; \underline{x}, t)$ over all sizes (last term in Eq. (6)), so that
158 this RHS vanishes when integrated over all particle sizes, thereby preserving the normalisation of
159 the PDF.

160 In the PDF studies devoted to scalar transport and micro-mixing by O'Brien²² and others,⁵⁵⁻⁵⁷
161 the statistical behavior of conserved quantities was examined, thus in cases where $\dot{B}(v^*; \underline{x}, t) = 0$
162 in Eq. (6). Considering physical particles, which can be created by nucleation or can disappear
163 by agglomeration, breakage or oxidation, this source becomes non-zero and it requires a specific
164 treatment when the PDF is decomposed into a set of stochastic particles for simulating its time
165 evolution. Here, when training the neural networks, the agglomeration source terms in $\dot{B}(v^*; \underline{x}, t)$
166 are computed following the approach by Kumar and D. Ramkrishna,⁵⁸ combined with a Monte
167 Carlo procedure to properly reassign the stochastic particles sizes following the physical models
168 used for agglomeration/coagulation.⁵ The roundoff error between the exact continuous PSD and

Neural networks for solving population balance equation

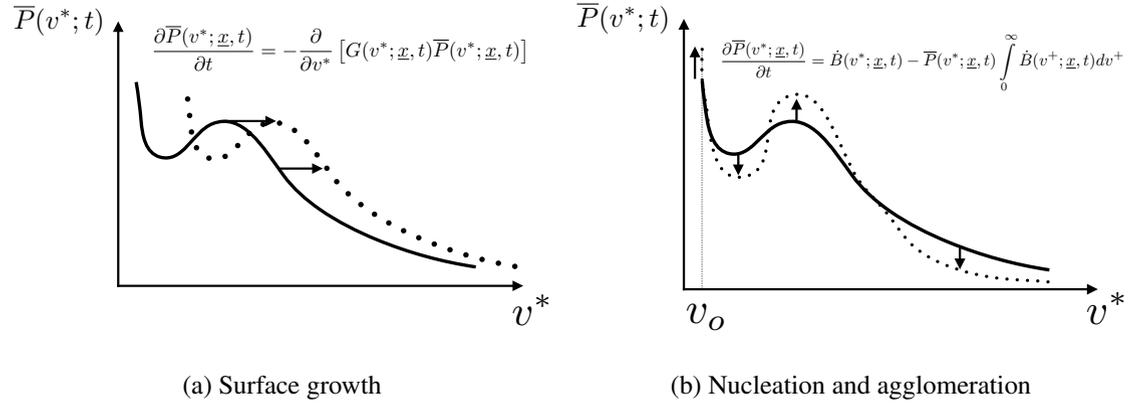


FIG. 1. Sketch of the PDF evolution according to Eq. (6). (a): Surface growth. (b): Nucleation and agglomeration.

169 the distribution reconstructed from the PDF discretised over the stochastic particles is corrected
 170 by transporting residuals with a fixed-sectional 2-point method.⁵⁹ All the details concerning this
 171 numerical method for solving the PBE using the PDF concept and its careful validation against
 172 analytical solutions may be found in Bouaniche et al.⁵ and are not repeated here for sake of brevity.

173 During the PDF solving, the number of particles per unit of flow volume within a section of
 174 size may be written

$$175 \quad N_i(\underline{x}, t) = \int_{I_{v_i}} n(v^*; \underline{x}, t) dv^* = N_T(\underline{x}, t) \int_{I_{v_i}} \bar{P}(v^*; \underline{x}, t) dv^*, \quad (7)$$

176 where the interval $I_{v_i} \equiv [v_i^{inf}, v_i^{sup}]$ defines the i -th fixed-section of size. M sections are considered
 177 for training the CNN. The training is performed from a database of correlated $N_i(t)$ and $N_i(t + \delta t)$
 178 (for $i = 1, \dots, M$) applying surface growth/loss, nucleation and agglomeration of the particles, i.e.,
 179 only terms (i) and (ii) of Eq. (6). Transport in physical space is not included in the neural networks
 180 as it will be handled separately by the flow solver. From the relations (2) and (7), the M scalar
 181 balance equations for $N_i(\underline{x}, t)$ that need to be considered together with other usual aerothermo-
 182 chemistry equations may be written

$$183 \quad \frac{\partial N_i(\underline{x}, t)}{\partial t} + \underline{u} \cdot \nabla N_i(\underline{x}, t) = \int_{I_{v_i}} \left(-\frac{\partial}{\partial v^*} [G(v^*; \underline{x}, t) n(v^*; \underline{x}, t)] + \dot{A}(v^*; \underline{x}) \right) dv^* = \dot{\Omega}_i(\underline{x}, t). \quad (8)$$

184 It is therefore $\dot{\Omega}_i(\underline{x}, t)$ that will be computed from the ANN-CNN as discussed thereafter.

Neural networks for solving population balance equation

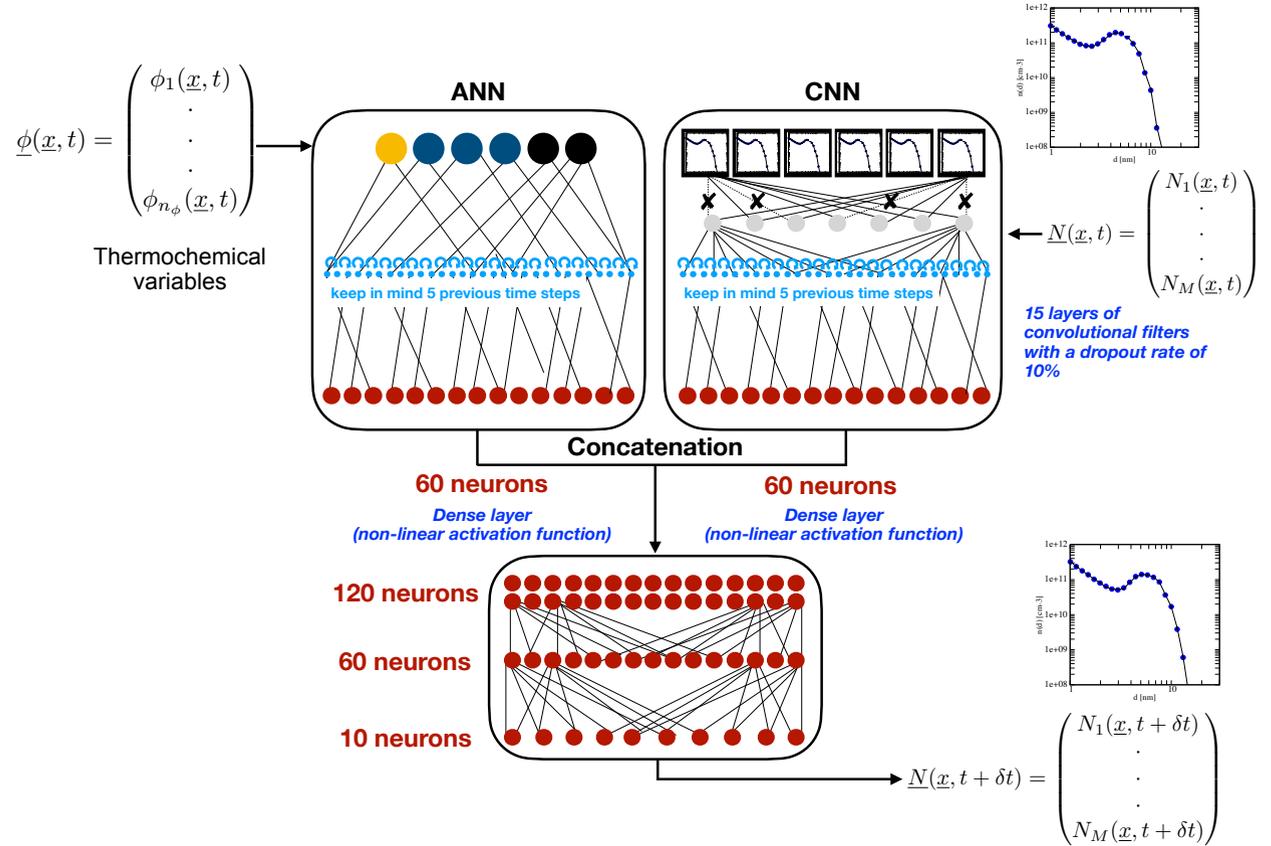


FIG. 2. Neural networks architecture. ANN inputs: Thermochemical variables and δt , time step. CNN input: Particle size distribution at time t . Concatenated networks output: Particle size distribution at time $t + \delta t$. Information of the 5 previous time-steps are stored in the LSTM.

185 B. Neural networks for PBE solving

186 The problem is decomposed into the thermochemical quantities (species concentrations, tem-
 187 perature, etc.), collected in a vector $\underline{\phi}(\underline{x}, t)$ of dimension n_ϕ controlling $G(v^*; \underline{x}, t)$, $\dot{A}(v^*; \underline{x}, t)$ and
 188 $\dot{B}(v^*; \underline{x}, t)$ in Eqs. (5) and (6), and the particle size distribution, expressed in terms of $N_i(\underline{x}, t)$ for the
 189 i -th section of size, organised into a vector $\underline{N}(\underline{x}, t)$ of dimension M , the number of sections of size
 190 considered. The neural networks are trained to solve for nucleation/breakage, surface growth/loss
 191 and agglomeration/coagulation so that for an input $(\underline{\phi}(\underline{x}, t), \underline{N}(\underline{x}, t), \delta t)$, the output is $\underline{N}(\underline{x}, t + \delta t)$.

192 The architecture of the numerical model, summarised in Fig. 2, is a multi-headed neural net-
 193 work which takes the form of a directed graph of layers. It is a combination of an artificial neural
 194 network tracking the evolution of the thermochemical vector $\underline{\phi}(\underline{x}, t)$, while a convolutional neural
 195 network is used for the particle number density $\underline{N}(\underline{x}, t)$. Both of these neural networks include

Neural networks for solving population balance equation

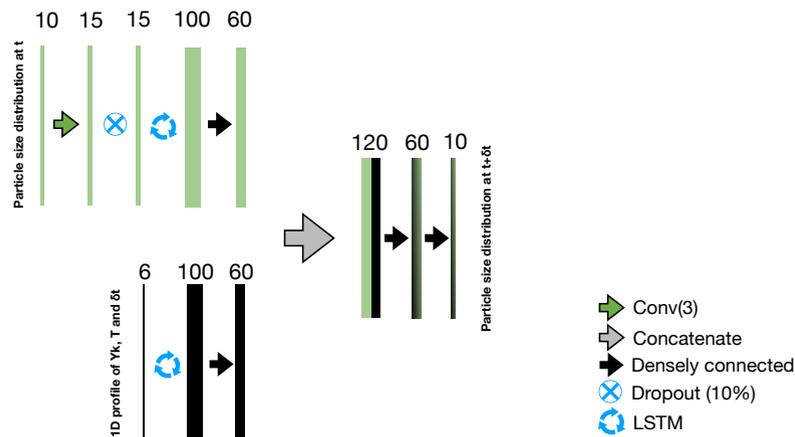


FIG. 3. Structure of the ANN-CNN with LSTM for PBE solving.

196 feedback connections,^{53,54} to account for the time history of the signals.

197 Neural networks allow for modeling strongly non-linear phenomena by organising in layers a
198 large number of linear functions of the form,

$$199 \quad O_i^k = f \left[w_o^k + \sum_{j=1}^n w_{ij}^k x_j^\ell \right], \quad (9)$$

200 where O_i^k is the output of the i -th neuron of the k -th layer, x_j^ℓ is the input from the j -th neuron
201 of the ℓ -th layer, w_{ij}^k is the weight attributed to that input, w_o^k is the bias and f is the activation
202 function. The w_{ij}^k are optimised to match a given set of outputs for a given set of inputs. In the
203 CNN, additional layers are exploring the input data (mostly filtering and max pooling) to extract
204 features associated to the control parameters. Then, the neural networks are trained to recognise
205 these features, which are associated to output values.^{49,50,60}

206 The evolutions of the thermochemical parameters $\underline{\phi}(\underline{x}, t)$ (solved separately from neural net-
207 works) and of δt , which may vary from iteration to iteration, both enter an ANN featuring a
208 recurrent long short-term memory neural layer (LSTM)⁵² composed of 100 neurons, to which a
209 60-neuron layer is connected. The feedback connections of this neural network are calibrated over
210 a duration characterised by 5 iterations in time. During training, each LSTM cell (100 neurons)
211 uses the solution at 't' as a training label (or target value), whereas the solutions at the 5 previ-
212 ous iterations serve as training data. Increasing this number of stored iterations was found not to
213 improve accuracy significantly, while below 4 iterations stored, accuracy cannot be secured. As
214 usually done in machine learning based approaches, every component of $\underline{\phi}(\underline{x}, t)$ is normalised by

Neural networks for solving population balance equation

215 its maximum level seen in the training database, these maximum must be stored to subsequently
 216 use the networks.

217 Because of the large range covered by the particle sizes during their evolution, before entering
 218 the CNN devoted to the PSD analysis, its signal is normalised by $\langle N \rangle$, the mean computed over all
 219 sections for all positions and times of the training database,

$$220 \quad N_i^*(\underline{x}, t) = \frac{N_i(\underline{x}, t) - \langle N \rangle}{\sigma_N}. \quad (10)$$

221 $\langle N \rangle$ and $\sigma_N = \sqrt{\langle [N_i(\underline{x}, t) - \langle N \rangle]^2 \rangle}$ computed from the training database, are stored to reconstruct
 222 $N_i(\underline{x}, t) = \sigma_N N_i^*(\underline{x}, t) + \langle N \rangle$ during the simulation from the dimensionless CNN output.

223 The normalised input $N_i^*(\underline{x}, t)$ goes through 15 layers of convolutional filters with a dropout
 224 rate of 10% (i.e., 10% of the data do not proceed to the next step). To account for the time history
 225 of the PSD, the 100 neurons LSTM layer used in the CNN is connected to a 60-neuron dense
 226 layer (a dense layer is driven by the same formulas as the linear layers, but the end result is passed
 227 through a non-linear activation function). As mentioned above, the time history is collected over
 228 5 iterations.

229 The 60 neurons of each branch (thermochemistry and particle sizes) are then concatenated
 230 (Fig. 2), leading to a 120-neuron layer, in which the concatenation operation is channel-wise.
 231 Finally, two dense connected layers return the prediction $\underline{N}^*(\underline{x}, t + \delta t)$, which is then transformed
 232 back into $\underline{N}(\underline{x}, t + \delta t)$ applying Eq. (10). Hence, the ANN-CNN is cast so that

$$233 \quad N_i(\underline{x}, t + \delta t) = \mathcal{F}_i \left[\underline{\phi}(\underline{x}, t), \underline{N}(\underline{x}, t), \delta t \right]. \quad (11)$$

234 Two options exist to implement such ANN-CNN in a flow solver. If a fractional step method is
 235 used with splitting⁶¹ of the fluid mechanics operators contributing to the evolution of the $N_i(\underline{x}, t)$,
 236 flow transport is first applied leading to the intermediate values $N_i(\underline{x}, t^*)$, which enter the ANN-
 237 CNN to return $N_i(\underline{x}, t + \delta t)$. In the second option of a fully explicit and direct integration of the
 238 RHS of the $N_i(\underline{x}, t)$ balance equations, the source $\dot{\Omega}_i(\underline{x}, t)$ in Eq. (8) may be expressed as

$$239 \quad \dot{\Omega}_i(\underline{x}, t) = \frac{\mathcal{F}_i[\underline{\phi}(\underline{x}, t), \underline{N}(\underline{x}, t), \delta t] - N_i(\underline{x}, t)}{\delta t}. \quad (12)$$

240 With both options, the time-step values seen in the flow solver must be contained within the range
 241 of time steps considered during the training of the neural networks. (Indeed, the highly non-linear
 242 interpolations by neural networks are prone to divergence when inputs values are not strictly within
 243 their training bounds.)

Neural networks for solving population balance equation

244 The ANN-CNN is built using the Keras and TensorFlow Python library with GPU support
 245 (www.tensorflow.org).

246 C. Soot physical modeling

247 The objective of this work is to isolate and study the potential of data driven approaches for
 248 solving the PBE, well established hypothesis and physical modeling are therefore introduced to
 249 generate the training database, so that the results obtained on a canonical problem will be similar
 250 to those previously reported in the literature. The novelty is the drastic reduction in computing
 251 time provided by the method, without loss in accuracy.

252 The soot particles are modeled as spherical and in the hybrid stochastic/fixed-sectional method,
 253 each stochastic particle discretising $\bar{P}(v^*; \underline{x}, t)$, the PDF of sizes, carries information on a character-
 254 istic volume v^k , with $k = 1, \dots, N_P$, where N_P is the total number of stochastic particles considered.

255 Nucleation is assumed to be driven by the collision of two pyrenes ($C_{16}H_{10}$) molecules,⁶²

$$256 \quad \dot{H}(\underline{x}, t) = 0.5\beta_{py}N_{py}^2, \quad (13)$$

257 with

$$258 \quad N_{py} = [C_{16}H_{10}] \mathcal{N}_A, \quad (14)$$

259 the volume number of pyrene molecules and where \mathcal{N}_A is the Avogadro constant.

260 The surface rate of change G is simplified in condensation of pyrene molecules on soot particles
 261 (G_{Cond}), C_2H_2 (acetylene) addition by the so-called HACA mechanism and surface oxidation by
 262 O_2 and OH ($G_{\text{HACA, Oxi}}$),⁶² i.e.,

$$263 \quad G(v^k, t) = G_{\text{Cond}}(v^k, t) + G_{\text{HACA, Oxi}}(v^k, t) \quad (15)$$

264 is applied to control the surface growth/loss of each stochastic particle v^k ($dv^k(t)/dt = G(v^k, t)$).

265 The condensation source term reads:

$$266 \quad G_{\text{Cond}}(v^k, t) = m_{py} \dot{H}_{\text{Cond}}(v^k, t) / N_i(t), \quad (16)$$

267 (for $v^k \in I_{v_i}$), with m_{py} the mass of one pyrene molecule and

$$268 \quad \dot{H}_{\text{Cond}}(v^k, t) = \beta_{v^k, py} N_i(t) N_{py}. \quad (17)$$

Neural networks for solving population balance equation

269 The surface reaction source term is cast in:

$$270 \quad G_{\text{HACA,Oxi}}(v^k; t) = (\dot{\omega}_{\text{C}_2\text{H}_2} + \dot{\omega}_{\text{O}_2} + \dot{\omega}_{\text{OH}})/N_i(t), \quad (18)$$

271 with the chemical sources,

$$272 \quad \dot{\omega}_{\text{C}_2\text{H}_2} = 2W_{\text{C}}k_{\text{C}_2\text{H}_2}[R_{v^k}][\text{C}_2\text{H}_2], \quad (19)$$

$$273 \quad \dot{\omega}_{\text{O}_2} = -2W_{\text{C}}k_{\text{O}_2}[R_{v^k}][\text{O}_2], \quad (20)$$

$$274 \quad \dot{\omega}_{\text{OH}} = -W_{\text{C}}k_{\text{OH}}[S_{v^k}][\text{OH}], \quad (21)$$

275 where W_{C} is the molar mass of Carbon, and $k_{\text{C}_2\text{H}_2}$, k_{O_2} , k_{OH} are calculated as in previous works.⁶²
 276 $[S_{v^k}]$ and $[R_{v^k}]$ read^{62,63}

$$277 \quad [S_{v^k}] + [R_{v^k}] = s^k \chi \alpha_{\text{HACA}}(v^k) N_i(t) / \mathcal{N}_A, \quad (22)$$

278 s^k is the particle surface and χ is the number of sites per unit surface of soot. Mature particles are
 279 expected to provide a lower proportion of active sites per unit of surface and this is accounted for
 280 using^{25,62}

$$281 \quad \alpha_{\text{HACA}}(v^k) = \tanh \left(\frac{a}{\log(\rho_{\text{soot}} \times v^k / (W_{\text{C}} / \mathcal{N}_A))} + b \right), \quad (23)$$

282 with $a = 12.65 - 0.00563T$ and $b = -1.38 + 0.00068T$.

283 The collision rates entering the Smoluchowski equation expressing the agglomeration, are cal-
 284 culated depending on the Knudsen number.^{64,65} Collision frequencies are calculated in the same
 285 manner for collisions between pyrene molecules (nucleation) or between pyrene molecules and
 286 soot particles (condensation). The numerical implementation of the discretisation of the Smolu-
 287 chowski equation⁵⁸ follows the now standard method by Eberle et al.⁶⁶ to account for the source
 288 term \dot{B}_i (Eq. (6)) in each I_{v_i} section considered.

289 III. ANN-CNN TRAINING AND VERIFICATION

290 A. One-dimensional sooting flame

291 A one-dimensional fuel rich (equivalence ratio of 2.07) laminar premixed ethylene-argon-
 292 oxygen flame studied both experimentally and numerically in the literature⁶⁷, is first considered
 293 to evaluate the proposed approach. The fresh gases flowing at a temperature of 300 K stabilise a
 294 flame propagating at the velocity of 8.26 cm/s. In mole fractions, the composition of the injected
 295 mixture is $X_{\text{C}_2\text{H}_4} = 0.133$, $X_{\text{O}_2} = 0.193$ and $X_{\text{Ar}} = 0.674$.

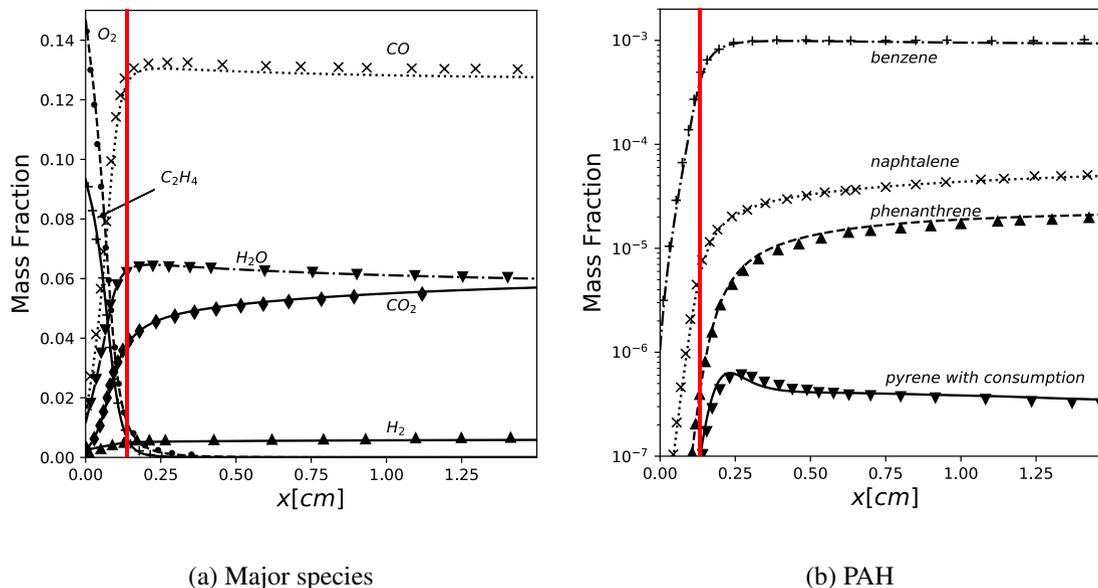


FIG. 4. (a) Major species mass fractions and (b) PAH across the sooting one-dimensional flame. Lines: present simulation. Symbols: reference simulation.⁶⁷ Vertical red line: position of the maximum heat release rate.

296 The complex molecular transport and the detailed chemical scheme by Appel et al.⁶² (101
 297 species and 544 elementary reactions) is used to simulate the gaseous part of the problem (ther-
 298 mochemical parameters that will enter the ANN) with the specialised CANTERA reacting flow
 299 solver.⁶⁸ The length of the one-dimensional computational domain is 0.02 m and the mesh is
 300 composed of 2906 points, it is refined at the reaction zone down to a mesh cell of $0.607\mu\text{m}$.
 301 To minimise error compensation between chemistry and heat transfer, as usually done in the nu-
 302 merical study of one-dimensional sooting flame, the measured temperature profile is imposed.⁶⁷
 303 This reference one-dimensional flame of the literature was already simulated using the hybrid
 304 stochastic/fixed-sectional approach solving for the PDF of size, the comparisons of the results
 305 against measurements and previous simulations may be found in Bouaniche et al.²⁵

306 The PBE one-dimensional simulations are performed in a moving reference frame, so that the
 307 derivative versus the Lagrangian residence time $\tau(x)$ relates to the position through the flame

$$308 \quad \tau(x) = \int_0^x \frac{1}{u(x^+)} dx^+, \quad (24)$$

309 where $u(x)$ results from the flame simulation in physical space.

310 During its training, the artificial neural network tracks the evolution of T , the temperature, and
 311 of the mass fractions of C_2H_2 , the fuel, of O_2 , the oxidizer, of OH, an intermediate radical, and of

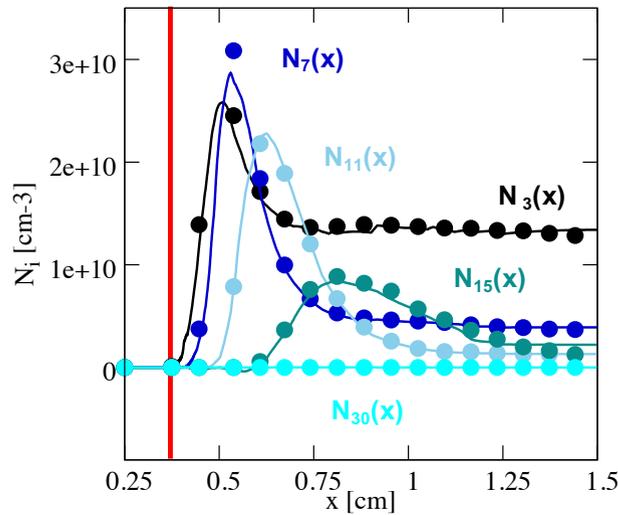


FIG. 5. Distribution across the one-dimensional premixed flame of particle number density for representative sections of the training database with $M = 30$. Red line: position of the maximum heat release rate. Section mid-diameter $0.5(v_i^{inf} + v_i^{sup})$, N_3 : 1.15 nm. N_7 : 1.97 nm. N_{11} : 3.3 nm. N_{15} : 5.82 nm. N_{30} : 44.22 nm.

312 C_6H_6 , the pyrene, therefore $\underline{\phi} = (T, Y_{C_2H_2}, Y_{O_2}, Y_{OH}, Y_{C_6H_6})$. These species were selected because
 313 of their dual impact on both flame and soot chemistry. C_6H_6 controls the modeling the particle
 314 surface growth due to condensation. The temperature, the mass fractions of C_2H_2 , O_2 and OH
 315 allow for quantifying the overall progress of combustion, to characterise the gaseous environment
 316 in which the soot particles are traveling. Adding more chemical species in the input vector of the
 317 ANN was found not to significantly improve the predictions by the ANN-CNN.

318 The consumption of pyrene by nucleation is accounted for by coupling the solving of this
 319 species with the PBE simulation (i.e., a sink term is added to the balance equation for the pyrene
 320 mass fraction). Figure 4 shows the comparison against major and PAH (polycyclic aromatic hydrocarbon) species obtained in this work and previous results reported in the literature.⁶⁷ The
 321 chemical one-dimensional flame structure obtained follows expected results and can be used for
 322 training the neural networks.
 323

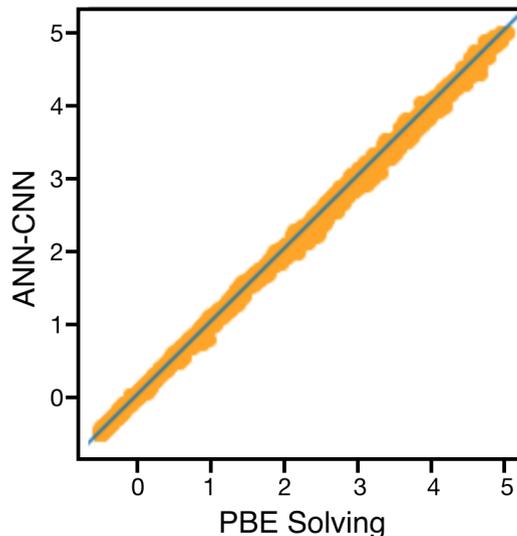


FIG. 6. Scatter plot (orange) of normalised particle number densities as predicted by the ANN-CNN versus the values from the PBE solution with the hybrid stochastic-fixed sectional method used for training.

324 B. Training and testing of the ANN-CNN

325 The soot number density $N_i(x)$ (Eq. (7)) for this steady flow problem is discretised over a num-
 326 ber M of sections, $I_{v_i} \equiv [v_i^{inf}, v_i^{sup}]$, defined from a geometric grid $v_i^{inf} = v_o F_s^i$, with $F_s = 1.5$ and
 327 $v_o = m_o \times \rho_{soot}$ for $m_o = 2m_{py}$, representative of the mass of nascent soot particles from the colli-
 328 sion of two pyrene molecules, corresponding to a diameter of 0.88 nm. The larger particle diameter
 329 to be considered is 51 nm. The ANN-CNN prediction is evaluated for $M = 10$ and 30 sections. The
 330 training is performed over a solution database obtained with the hybrid stochastic/fixed-sectional
 331 method using $N_P = 1000$ stochastic particles and a fixed number of 30 sections. The 10 sections
 332 ANN-CNN is trained interpolating the 30 sections database over 10 sections. The impact of us-
 333 ing a smaller number of sections in the ANN-CNN simulation than for the solution providing the
 334 training database is then evaluated, as a strategy to reduce the computing time.

335 Figure 5 shows the distributions through the one-dimensional flame, i.e., from fresh to burnt
 336 gases, of number densities of the training database for representative sections. The simulations
 337 starts with all the sections set at zero ($N_i(x) = 0, \forall i$), then particles nucleate and undergo surface
 338 growths. Some number densities of particles, as for N_3 (diameter of 1.15 nm), reach a peak
 339 value before decreasing to a plateau, these particles will be those found in the burnt gases. For
 340 other characteristic sizes, such as 3.3 nm (N_{11} in Fig. 5), N_i is produced to then vanish in burnt

Neural networks for solving population balance equation

341 gases, because of thermophysical conditions favouring both agglomeration and oxidation of these
 342 particles.

343 The size of the database is 2906 (number of mesh points) $\times [M$ (number of sections) $+ 6$
 344 (number of thermochemical quantities & time step)], for example $104\,616$ data for $M = 30$
 345 sections. To train the neural networks, 80% of the points through the flame are randomly selected.
 346 For this database, the time steps entering the ANN are within the interval $[2.5 \cdot 10^{-6}\text{s}, 7.5 \cdot 10^{-5}\text{s}]$
 347 with a mean value centred at $1.0 \cdot 10^{-5}\text{s}$. This time-step distribution results from Eq. (24), thereby
 348 from the velocity profile through the flame and the non-uniform distribution of the mesh points
 349 through the reaction zone.

350 An Adam optimizer and a loss function based on mean absolute percentage error are used.⁶⁹
 351 The 20% remaining data serve to test the ANN-CNN accuracy. The particle number densities, as
 352 predicted by the networks versus those of the hybrid stochastic/fixed-sectional method, are dis-
 353 played in Figure 6. This result is obtained after 1500 epochs (number of times that the learning
 354 algorithm worked through the entire training dataset). The solution of the system converges with
 355 a maximum local error on the PSD of less than 10% and a mean error of less than 1% , which is
 356 sufficient to accurately predict the PSD in a posteriori tests performed, in which the error accumu-
 357 late.

358 C. A posteriori tests

359 To test the method on conditions similar to those of its implementation in a flow solver, the bal-
 360 ance equation for the $N_i(x)$ should now be solved. The equation for the $N_i(x)$ may be transformed
 361 into the moving frame for the steady one-dimensional flame

$$362 \quad u \frac{dN_i(x)}{dx} = \frac{dN_i(x(\tau))}{d\tau} = \hat{\Omega}_i(x(\tau)). \quad (25)$$

363 The derivative versus time, $dN_i(x(\tau))/d\tau$, is discretised using a third-order minimal storage
 364 Runge-Kutta scheme,⁷⁰ with the RHS $\hat{\Omega}_i(x(\tau))$ of every sub-step of the integration reconstructed
 366 from the ANN-CNN (Eq. (11)) using Eq. (12).

367 For a number of sections of $M = 30$, Fig. 7 shows the comparison between the PSD predicted
 368 with the hybrid stochastic/fixed-sectional method to solve for the population balance equation
 369 and the results obtained with Eq. (25). The reference simulation of this flame by direct Monte
 370 Carlo solutions of non-inertial particles dynamics, which aims at performing a direct numerical

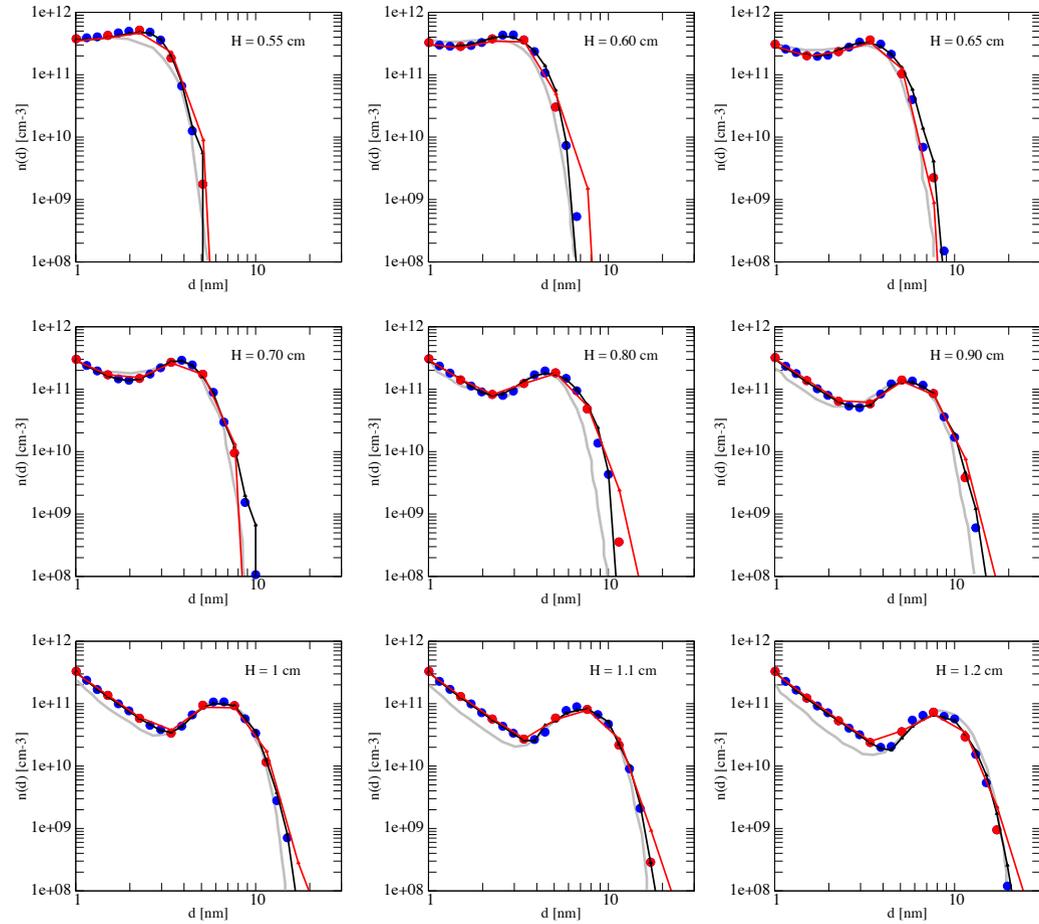


FIG. 7. Particle sizes distribution at different heights H through the one-dimensional premixed flame. Grey line: Reference simulation by Zhao et al.⁶⁷ Symbols: Training simulation with 1000 Monte-Carlo particles.²⁵ Solid line: ANN-CNN. Blue: $M = 30$ sections. Red: $M = 10$ sections.

371 simulation of the elementary physical phenomena acting on the particles,⁶⁷ is also given in these
 372 figures. As usually done for soot, in these plots the PSD is cast in $n(d) = dN(\log(d))/d\log(d)$,
 373 with d the particle diameter in nm. Figure 7 also suggests that the ANN-CNN operating with $M =$
 374 10 sections, but trained by interpolation over 30 sections and 1000 stochastic particles, provides
 375 results with an accuracy similar to the one obtained with the ANN-CNN operating with $M = 30$
 376 sections. This confirms the potential of the approach relying on the training over a large database,
 377 to then reduce the dimensionality of the problem solved and thus minimise the computing cost.

378 The total number density $N_T = \int_{d_0}^{\infty} n(d)d\log(d)$ is compared against measurements in Fig. 8
 379 for a diameter of the primary particle $d_0 = 3$ nm (see²⁵ for a discussion on d_0). As expected from
 380 the results for the PSD, the ANN-CNN reproduces the evolution of $N_T(x)$ across the flame, as

Neural networks for solving population balance equation

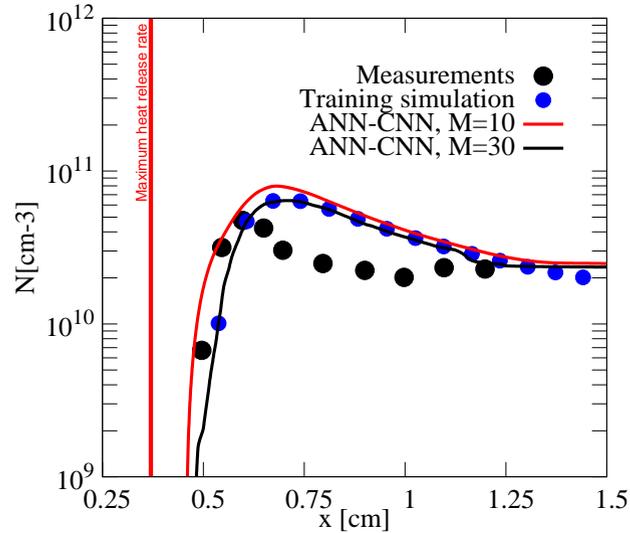


FIG. 8. Number density for particles above 3 nm. Black circle: Measurements.⁶⁷ Symbols: Training simulation with 1000 Monte-Carlo particles.²⁵ Solid lines: ANN-CNN, Black: $M = 30$ sections, Red: $M = 10$ sections.

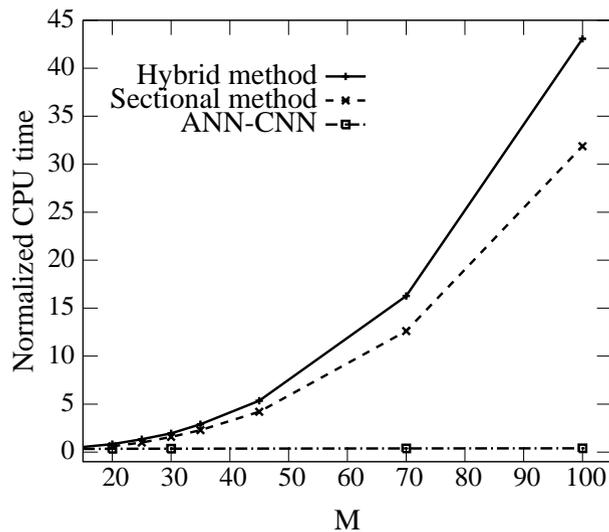


FIG. 9. Normalised computing cost versus the number of sections. Dashed line: Sectional two-point method.⁵⁹ Solid line: Hybrid stochastic/fixed-sectional method. Dashed-dotted line and square: ANN-CNN.

381 previously obtained solving the PBE with the hybrid method.

382 This operation is repeated for various numbers of sections. Figure 9 shows the correspond-
 383 ing computing cost normalised by the cost of a reference simulation with a two-point sectional
 384 method⁵⁹ with 25 sections. Here, the training and the a posteriori simulations are performed with

Neural networks for solving population balance equation

385 the same number of sections. The computing effort with the ANN-CNN varies only with the num-
386 ber of sections transported, i.e., the number of balance equations solved (Eq. (8)), in which the
387 interactions between the sections is cast in a source term provided by the neural networks. Solv-
388 ing for the PBE (Eq. (2)) is much more demanding in terms of computing when the number of
389 sections increases, because of the direct simulation of the nucleation, of the agglomeration, of the
390 surface growth and of the oxidation processes. Thereby, a significant computing cost reduction is
391 observed with the neural networks, up to more than 30 for 100 sections. Such reduction should
392 allow for coupling detailed soot modeling with the flow solution over complex three-dimensional
393 geometries.

394 **IV. CONCLUSION**

395 A strategy has been discussed to train a combination of artificial neural networks (ANN) and
396 convolution neural networks (CNN) to solve for the population balance equation (PBE) of flowing
397 non-inertial particles. The numerical framework to build the training database relies on a hybrid
398 stochastic-fixed sectional approach, based on the probability density function (PDF) concept²² in
399 order to benefit from an accurate treatment of the surface growth/loss of the particles.

400 Applied to a canonical sooting flame, the ANN deals with the thermochemical inputs, driving
401 the physics of the carbon particulate emissions, while the CNN processes the particle size distri-
402 bution (PSD). Both networks are augmented with one additional long short-term memory artificial
403 neural layer, before being concatenated to return the new PSD for a given increment in time, which
404 is then used to construct a source term entering the eulerian transport equation for particle num-
405 ber densities of a few sections of sizes. For a given level of complexity of the thermo-physical
406 modeling of the particles, which is present in the source terms through the neural networks, the
407 computing cost for accurately transporting these sections stays very moderate compared to the
408 usual cost of PBE solving. Therefore results confirm the potential of the approach, both in terms
409 of accuracy and cost reduction.

410 Even though this procedure using neural networks seems promising for solving population bal-
411 ance equations, some points still need to be addressed. The major one is the choice of the training
412 database to deal with three-dimensional flows. In the present work, the training, the testing of the
413 neural networks and the subsequent simulation were performed on the very same flow configu-
414 ration. As for any neural network based modeling, it is fully of practical interest only when the

Neural networks for solving population balance equation

415 training is performed on a generic problem, preferably aside from the targeted flow simulation.^{28,71}
416 This was recently achieved for detailed chemistry reduction and pre-integration with ANN using
417 a stochastic micro-mixing problem solving for the PDF of the thermochemical variables, to train
418 the networks without performing any flow simulation.³² Coupling this approach with the present
419 PBE solving with ANN-CNN would be an interesting route to follow in future work, in order to
420 generalise the method to the application of three-dimensional turbulent combustion systems.

421 ACKNOWLEDGMENT

422 This paper in which stochastic methods are combined with neural networks was prepared to
423 honour the memory of Professor Edward E. O'Brien, who pioneered and popularised the PDF
424 concept for the numerical simulation of complex turbulent flows.

425 The PhD of the first author is funded by ANRT (Agence Nationale de la Recherche et de la
426 Technology), SAFRAN-AE under the CIFRE No 1643/2016. Computing time has been provided
427 by CRIANN (Centre Régional Informatique et d'Applications Numériques de Normandie).

428 DATA AVAILABILITY STATEMENT

429 The data that support the findings of this study are available from the corresponding author
430 upon request. The code of the Hybrid Stochastic/Sectional method used for training is available at
431 <https://www.coria-cfd.fr/index.php/File:Nps-master.zip>.

432 REFERENCES

- 433 ¹D. Ramkrishna, *Population balances. Theory and applications to particulate systems in engi-*
434 *neering* (Academic Press San Diego, 2000) p. pp. 356.
- 435 ²D. L. Marchisio and R. O. Fox, *Computational Models for Polydisperse Particulate and Multi-*
436 *phase Systems* (Cambridge University Press, 2013) p. 508.
- 437 ³R. O. Fox, *Computational models for turbulent reacting flows* (Cambridge University Press,
438 2003).
- 439 ⁴J. Solsvik and H. A. Jakobsen, "The foundation of the population balance equation: A review,"
440 *Journal of Dispersion Science and Technology* **36**, 510–520 (2015).

Neural networks for solving population balance equation

- 441 ⁵A. Bouaniche, L. Vervisch, and P. Domingo, “A hybrid stochastic/fixed-sectional method for
442 solving the population balance equation,” Chem. Eng. Sci. **209**, 115198 (2019).
- 443 ⁶A. Liu and S. Rigopoulos, “A conservative method for numerical solution of the population
444 balance equation, and application to soot formation,” Combust. Flame **205**, 506–521 (2019).
- 445 ⁷D. L. Marchisio and R. O. Fox, eds., *Multiphase reacting flows: Modelling and simulation*,
446 CISM International Center for Mechanical Sciences, Vol. 492 (Springer-Verlag, Wien, 2007) p.
447 269.
- 448 ⁸F. Sewerin and S. Rigopoulos, “An explicit adaptive grid approach for the numerical solution of
449 the population balance equation,” Chemical Engineering Science **168**, 250 – 270 (2017).
- 450 ⁹R. Irizarry, “Fast Monte Carlo methodology for multivariate particulate systems-I: Point ensem-
451 ble Monte Carlo,” Chemical Engineering Science **63**, 95–110 (2008).
- 452 ¹⁰R. Irizarry, “Fast Monte Carlo methodology for multivariate particulate systems-II: τ -PEMC,”
453 Chemical Engineering Science **63**, 111–121 (2008).
- 454 ¹¹M. Singh, H. Y. Ismail, T. Matsoukas, A. B. Albadarin, and G. Walker, “Mass-based finite
455 volume scheme for aggregation, growth and nucleation population balance equation,” Proc. R.
456 Soc. A **475**, 20190552 (2019).
- 457 ¹²S. Salenbauch, C. Hasse, M. Vanni, and D. L. Marchisio, “A numerically robust method of
458 moments with number density function reconstruction and its application to soot formation,
459 growth and oxidation,” J. Aerosol Sci. , 34–49 (2019).
- 460 ¹³A. Boje, J. Akroyd, S. Sutcliffe, and M. Kraft, “Study of industrial titania synthesis using a
461 hybrid particle-number and detailed particle model,” Chemical Engineering Science **219**, 115615
462 (2020).
- 463 ¹⁴F. Bisetti, A. Attili, and H. Pitsch, “Advancing predictive models for particulate formation in
464 turbulent flames via massively parallel direct numerical simulations,” Phil. Trans. R. Soc. A **372**,
465 20130324 (2014).
- 466 ¹⁵A. Liu, C. E. Garcia, F. Sewerin, B. A. O. Williams, and S. Rigopoulos, “Population balance
467 modelling and laser diagnostic validation of soot particle evolution in laminar ethylene diffusion
468 flames,” Combust. Flame **222**, 384–400 (2020).
- 469 ¹⁶L. Cifuentes, J. Sellmann, I. Wlokas, and A. Kempf, “Direct numerical simulations of nanoparti-
470 cle formation in premixed and non-premixed flame-vortex interactions,” Phys. Fluids **32**, 093605
471 (2020).

Neural networks for solving population balance equation

- 472 ¹⁷J. Boulanger, W. S. Neill, F. Liu, and G. J. Smallwood, “An improved phenomenological soot
473 formation submodel for three-dimensional Diesel engine simulations: Extension to agglomera-
474 tion of particles into clusters,” *J. Eng. Gas Turbines Power* **130**, 062808 (2008).
- 475 ¹⁸C. G. Moniruzzaman and F. Yu, “A 0D aircraft engine emission model with detailed chemistry
476 and soot microphysics,” *Combust. Flame* **159**, 1670–1686 (2012).
- 477 ¹⁹B. Franzelli, A. Vié, and N. Darabiha, “A three-equation model for the prediction of soot emis-
478 sions in LES of gas turbines,” *Proc. Combust. Inst.* **37**, 5411–5419 (2019).
- 479 ²⁰E. Hopf, “Statistical hydromechanics and functional calculus,” *J. Rational Mech. Anal.* **1**, 87–
480 123 (1952).
- 481 ²¹W. Kollmann, *Navier-Stokes Turbulence, Theory and Analysis* (Springer Nature Switzerland,
482 2019) p. 725.
- 483 ²²C. Dopazo and E. O’Brien, “Functional formulation of nonisothermal turbulent reactive flows,”
484 *Phys. Fluids* **17**, 1968–1975 (1974).
- 485 ²³E. E. O’Brien, “The probability density function (pdf) approach to reacting turbulent flows,” in
486 *Turbulent Reacting Flows* (Academic Press London, 1980) p. 185.
- 487 ²⁴R. E. Meyers and E. E. O’Brien, “The joint PDF of a scalar and its gradient at a point in a
488 turbulent fluid,” *Combust. Sci. Tech.* **26**, 123–134 (1981).
- 489 ²⁵A. Bouaniche, J. Yon, P. Domingo, and L. Vervisch, “Analysis of the soot particle size distri-
490 bution in a laminar premixed flame: A hybrid stochastic/fixed-sectional approach,” *Flow Turbu-
491 lence and Combust.* **104**, 753–775 (2020).
- 492 ²⁶F. Christo, A. Masri, E. Nebot, and S. Pope, “An integrated PDF/neural network approach for
493 simulating turbulent reacting systems,” *Symp. (Int.) Combust.* **26**, 43–48 (1996).
- 494 ²⁷J. A. Blasco, N. Fueyo, C. Dopazo, and J. Ballester, “Modelling the temporal evolution of a
495 reduced combustion chemical system with artificial neural network,” *Combust. Flame* **113**, 38–
496 52 (1998).
- 497 ²⁸L. L. Franke, A. K. Chatzopoulos, and S. Rigopoulos, “Tabulation of combustion chemistry
498 via artificial neural networks (ANNs): Methodology and application to LES-PDF simulation of
499 sydney flame L,” *Combust. Flame* **185**, 245–260 (2017).
- 500 ²⁹F. R. R. Padilha and A. L. D. Bortoli, “Solutions for a laminar jet diffusion flame of methyl for-
501 mate using a skeletal mechanism obtained by applying ANNs,” *Journal of Mathematical Chem-
502 istry* **57**, 2229–2247 (2019).

Neural networks for solving population balance equation

- 503 ³⁰O. Owoyele, P. Kundu, M. M. Ameen, T. Echekeki, and S. Som, “Application of deep artificial
504 neural networks to multi-dimensional flamelet libraries and spray flames,” *International Journal*
505 *of Engine Research* **21**, 151–168 (2020).
- 506 ³¹R. Ranade, S. Alqahtani, A. Farooq, and T. Echekeki, “An ann based hybrid chemistry framework
507 for complex fuels,” *Fuel* **241**, 625–636 (2019).
- 508 ³²K. Wan, C. Barnaud, L. Vervisch, and P. Domingo, “Chemistry reduction using machine learn-
509 ing trained from non-premixed micro-mixing modeling: Application to DNS of a syngas turbu-
510 lent oxy-flame with side-wall effects,” *Combust. Flame* **220**, 119–129 (2020).
- 511 ³³L. V. Lorang, B. Podvin, and P. L. Quéré, “Application of compact neural network for drag
512 reduction in a turbulent channel flow at low Reynolds numbers,” *Phys. Fluids* **20**, 045104 (2008).
- 513 ³⁴R. Ranade and T. Echekeki, “A framework for data-based turbulent combustion closure: A pos-
514 teriori validation,” *Combustion and flame* **210**, 279–291 (2019).
- 515 ³⁵Z. Nikolaou, C. Chrysostomou, L. Vervisch, and R. S. Cant, “Progress variable variance and
516 filtered rate modelling using convolutional neural networks and flamelet methods,” *Flow Turbu-*
517 *lence Combust.* **103**, 485–501 (2019).
- 518 ³⁶C. J. Lapeyre, A. Misdariis, N. Cazard, D. Veynante, and T. Poinso, “Training convolutional
519 neural networks to estimate turbulent sub-grid scale reaction rates,” *Combust. Flame* **203**, 255–
520 264 (2019).
- 521 ³⁷A. Seltz, P. Domingo, L. Vervisch, and Z. M. Nikolaou, “Direct mapping from les resolved
522 scales to filtered-flame generated manifolds using convolutional neural networks,” *Combust.*
523 *Flame* **210**, 71–82 (2019).
- 524 ³⁸X. Zhu, Z. Cai, J. Wu, Y. Cheng, and Q. Huang, “Convolutional neural network based combus-
525 tion mode classification for condition monitoring in the supersonic combustor,” *Acta Astronau-*
526 *tica* **159**, 349–357 (2019).
- 527 ³⁹B. Liu, J. Tang, H. Huang, and X.-Y. Lu, “Deep learning methods for super-resolution recon-
528 struction of turbulent flows,” *Phys. Fluids* **32**, 025105 (2020).
- 529 ⁴⁰Z. M. Nikolaou, C. Chrysostomou, Y. Minamoto, and L. Vervisch, “Evaluation of a neural
530 network-based closure for the unresolved stresses in turbulent premixed v-flames,” *Flow Turbu-*
531 *lence and Combust.* (in press).
- 532 ⁴¹K. Wan, S. Hartl, L. Vervisch, P. Domingo, R. Barlow, and C. Hasse, “Combustion regime
533 identification from machine learning trained by Raman/Rayleigh line measurements,” *Combust.*
534 *Flame* **219**, 168–274 (2020).

Neural networks for solving population balance equation

- 535 ⁴²G. Aversano, A. Bellemans, Z. Li, A. Coussement, O. Gicquel, and A. Parente, “Application of
536 reduced-order models based on PCA & Kriging for the development of digital twins of reacting
537 flow applications,” *Computers & Chemical Engineering* **121**, 422–441 (2019).
- 538 ⁴³“Population balance model-based hybrid neural network for a pharmaceutical milling process,”
539 *Journal of Pharmaceutical Innovation* **5**, 161–168 (2010).
- 540 ⁴⁴P. Georgieva and S. F. de Azavedo, “Neural network-based control strategies applied to a fed-
541 batch crystallization process,” *International Journal of Chemical and Molecular Engineering* **1**
542 (2007).
- 543 ⁴⁵G. Cogoni and P.-J. Frawley, “Particle size distribution reconstruction using a finite number of its
544 moments through artificial neural networks: A practical application,” *Crystal Growth & Design*
545 **15**, 239–246 (2015).
- 546 ⁴⁶J. Schäfer, P. Schmitt, M. W. Hlawitschka, and H.-J. Bart, “Measuring particle size distribution
547 in multiphase flows using a convolutional neural network,” *Chemie Ingenieur Technik* **91**, 1688–
548 1695 (2019).
- 549 ⁴⁷S. H. Bryngelson, A. Charalampopoulos, T. P. Sapsis, and T. Colonius, “A gaussian moment
550 method and its augmentation via lstm recurrent neural networks for the statistics of cavitating
551 bubble populations,” *International Journal of Multiphase Flow* **127**, 103262 (2020).
- 552 ⁴⁸“Application of population balance theory for dynamic modeling of methane and ethane hydrate
553 formation processes,” *Energy & Fuels* **32**, 8131–8144 (2018).
- 554 ⁴⁹Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature* **521**, 436–444 (2015).
- 555 ⁵⁰J. Schmidhuber, “Deep learning in neural networks: An overview,” *Neural Networks* **61**, 85–117
556 (2015).
- 557 ⁵¹H. H. Aghdam and E. J. Heravi, *Guide to Convolutional Neural Networks* (Springer, Switzerland
558 Cham, 2017) p. 282.
- 559 ⁵²S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation* **9**, 1735–
560 1780 (1997).
- 561 ⁵³X.-S. Zhang, “Feedback neural networks,” in *Neural Networks in Optimization. Nonconvex Op-
562 timization and its Application*, Vol. 46 (Springer, Boston, MA, 2000) pp. 137–175.
- 563 ⁵⁴S. Herzog, C. Tetzlaff, and F. Wörgötter, “Evolving artificial neural networks with feedback,”
564 *Neural Networks* **123**, 153–162 (2020).
- 565 ⁵⁵S. B. Pope, “Pdf method for turbulent reacting flows,” *Prog. Energy Combust. Sci.* **11**, 119–195
566 (1985).

Neural networks for solving population balance equation

- 567 ⁵⁶W. Kollmann, “The pdf approach to turbulent flow,” *Theor. and Comp. Fluid Dynamics* **1**, 349–
568 285 (1990).
- 569 ⁵⁷D. C. Haworth, “Progress in probability density function methods for turbulent reacting flows,”
570 *Prog. Energy Combust. Sci.* **36**, 168–259 (2010).
- 571 ⁵⁸S. Kumar and D. Ramkrishna, “On the solution of population balance equations by discretization
572 – I: A fixed pivot technique,” *Chemical Engineering Science* **51**, 1311 – 1332 (1996).
- 573 ⁵⁹S. Park and S. Rogak, “A novel fixed-sectional model for the formation and growth of aerosol
574 agglomerates,” *Journal of Aerosol Science* **35**, 1385 – 1404 (2004).
- 575 ⁶⁰R. Wason, “Deep learning: Evolution and expansion,” *Cognitive Systems Research* **52**, 701–708
576 (2018).
- 577 ⁶¹J. H. Ferziger and M. Perić, *Computational Methods for Fluid Dynamics* (Springer, 1996) p.
578 364.
- 579 ⁶²J. Appel, H. Bockhorn, and M. Frenklach, “Kinetic modeling of soot formation with detailed
580 chemistry and physics: laminar premixed flames of C2 hydrocarbons,” *Combust. Flame* **121**,
581 122 – 136 (2000).
- 582 ⁶³D. Aubagnac-Karkar, J.-B. Michel, O. Colin, P. E. Vervisch-Kljakic, and N. Darabiha, “Sec-
583 tional soot model coupled to tabulated chemistry for Diesel RANS simulations,” *Combust.*
584 *Flame* **162**, 3081 – 3099 (2015).
- 585 ⁶⁴A. Kazakov and M. Frenklach, “Dynamic modeling of soot particle coagulation and aggregation:
586 Implementation with the method of moments and application to high-pressure laminar premixed
587 flames,” *Combust. Flame* **114**, 484 – 501 (1998).
- 588 ⁶⁵G. Blanquart and H. Pitsch, “A joint volume-surface-hydrogen multi-variate model for soot
589 formation,” in *Combustion Generated Fine Carbonaceous Particles*, edited by H. Bockhorn,
590 A. D’Anna, A. F. Sarofim, and H. Wang (KIT Scientific Publisher, Karlsruhe, 2009) pp. 437–
591 463.
- 592 ⁶⁶C. Eberle, P. Gerlinger, and M. Aigner, “A sectional PAH model with reversible PAH chemistry
593 for CFD soot simulations,” *Combust. Flame* **179**, 63 – 73 (2017).
- 594 ⁶⁷B. Zhao, Z. Yang, M. V. Johnston, H. Wang, A. S. Wexler, M. Balthasar, and M. Kraft, “Mea-
595 surement and numerical simulation of soot particle size distribution functions in a laminar pre-
596 mixed ethylene-oxygen-argon flame,” *Combust. Flame* **133**, 173 – 188 (2003).
- 597 ⁶⁸D. G. Goodwin, H. K. Moffat, and R. L. Speth, “Cantera: An object-oriented software toolkit
598 for chemical kinetics, thermodynamics, and transport processes,” (2017), version 2.3.0.

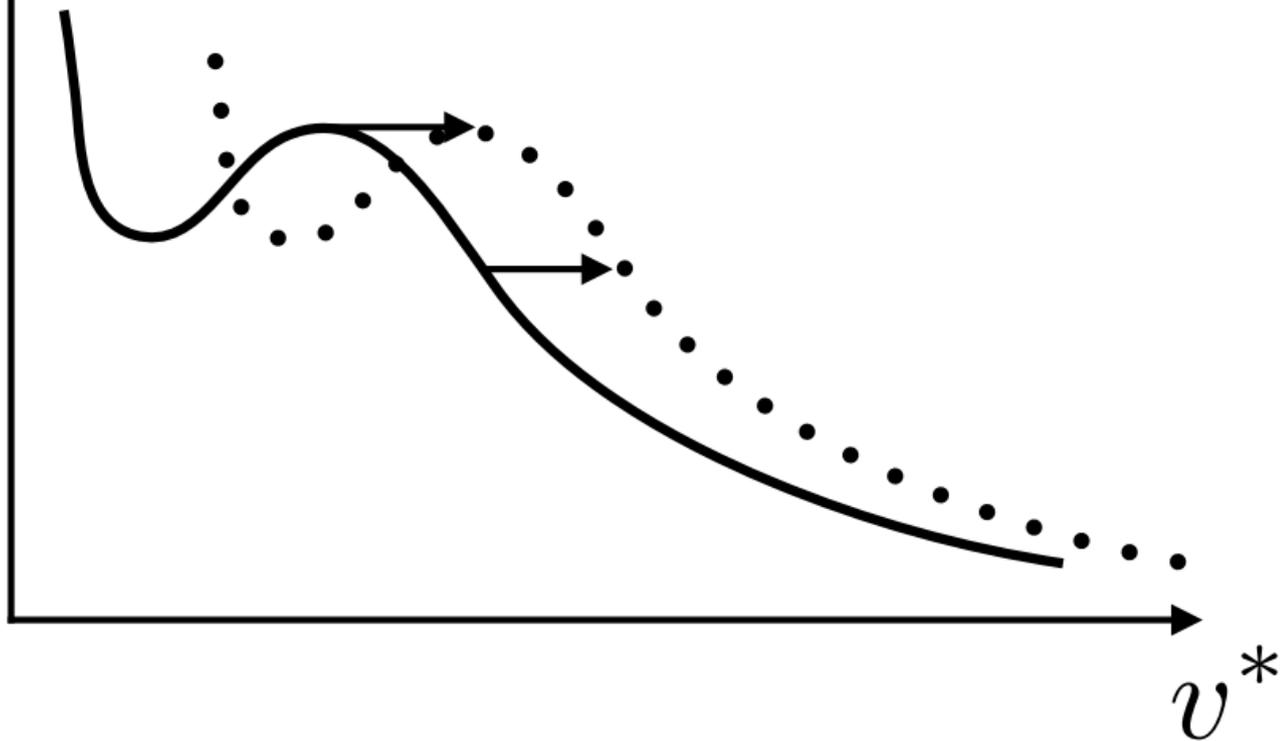
This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.
PLEASE CITE THIS ARTICLE AS DOI:10.1063/1.50031144

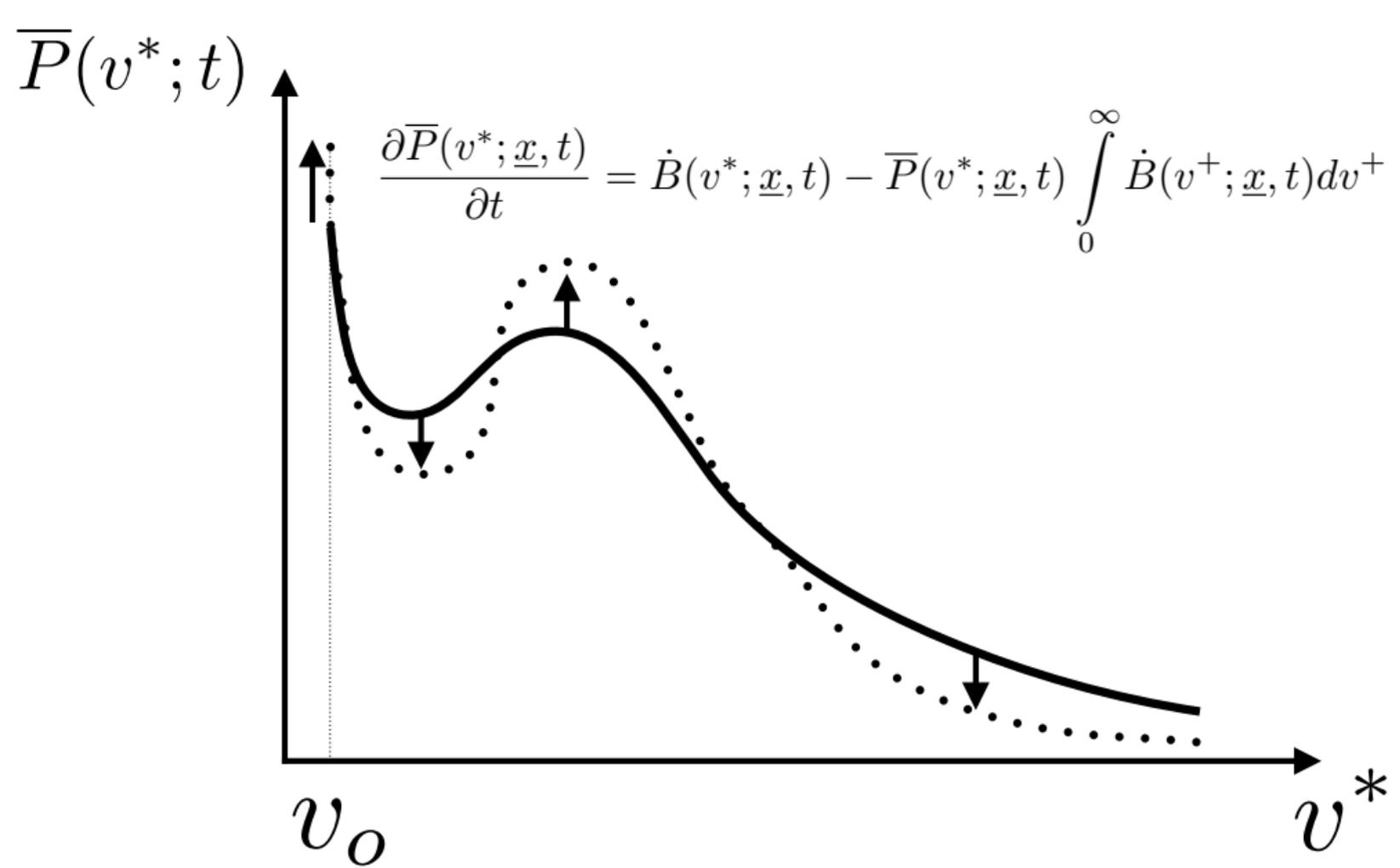
Neural networks for solving population balance equation

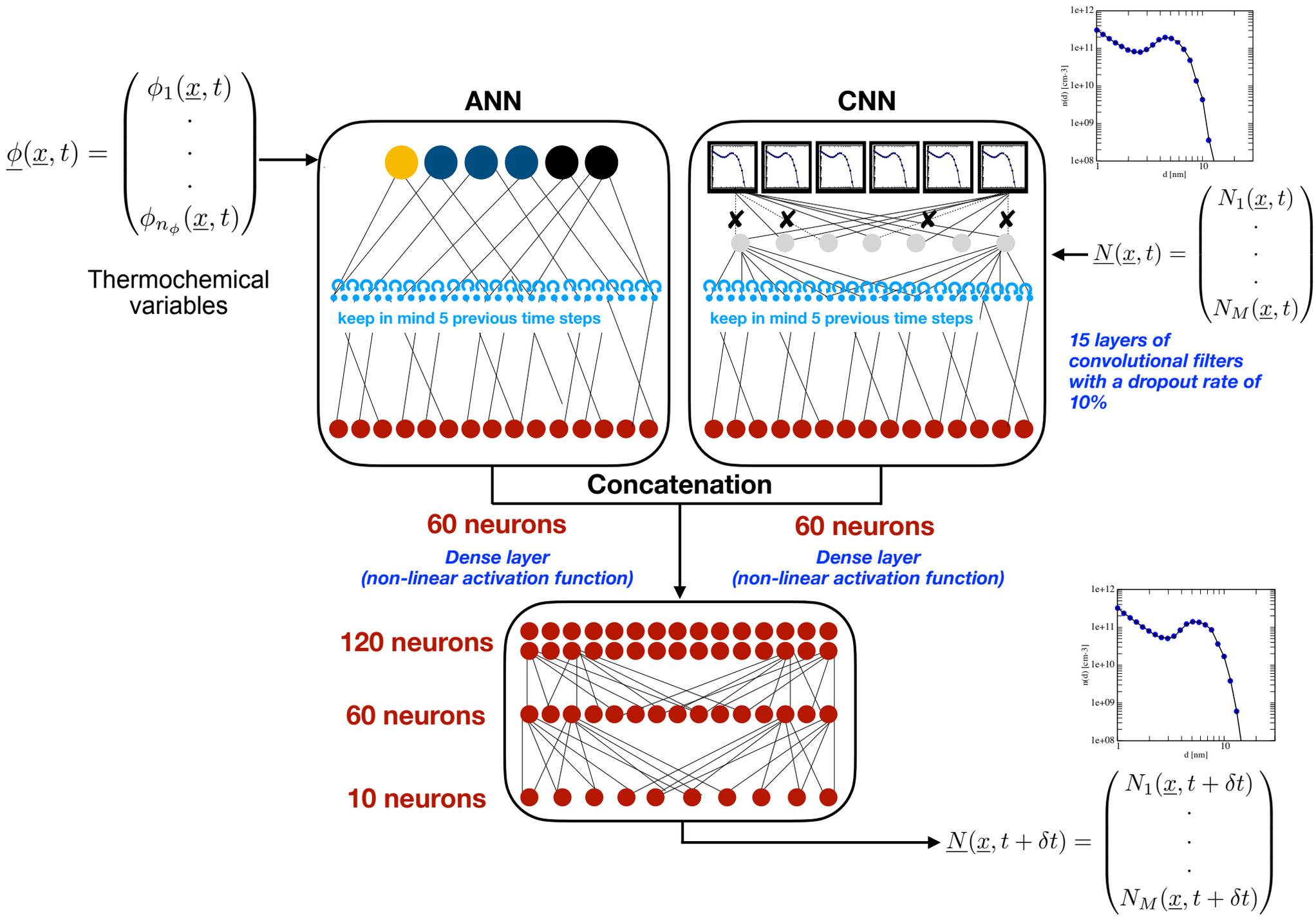
- 599 ⁶⁹D. P. Kingma and J. L. Ba, “ADAM: A method for stochastic optimization,”
600 <https://arxiv.org/pdf/1412.6980> (2017).
- 601 ⁷⁰A. A. Wray, “Minimal storage time-advancement schemes for spectral methods,” Tech. Rep.
602 (Center for turbulence research Report, Stanford University, 1990).
- 603 ⁷¹A. K. Chatzopoulos and S. Rigopoulos, “A chemistry tabulation approach via Rate Controlled
604 Constrained Equilibrium (RCCE) and Artificial Neural Networks (ANNs), with application to
605 turbulent non-premixed CH₄/H₂/N₂ flames,” Proceedings of the Combustion Institute **34**, 1465–
606 1473 (2013).

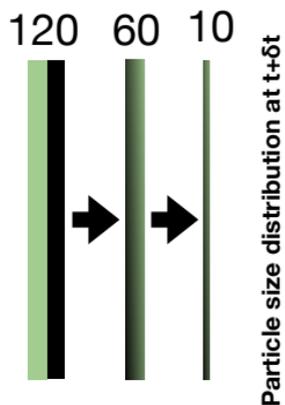
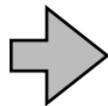
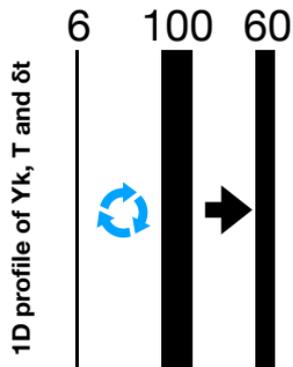
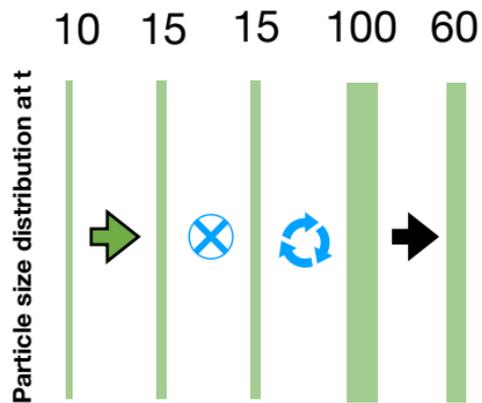
$\bar{P}(v^*; t)$

$$\frac{\partial \bar{P}(v^*; \underline{x}, t)}{\partial t} = -\frac{\partial}{\partial v^*} [G(v^*; \underline{x}, t) \bar{P}(v^*; \underline{x}, t)]$$



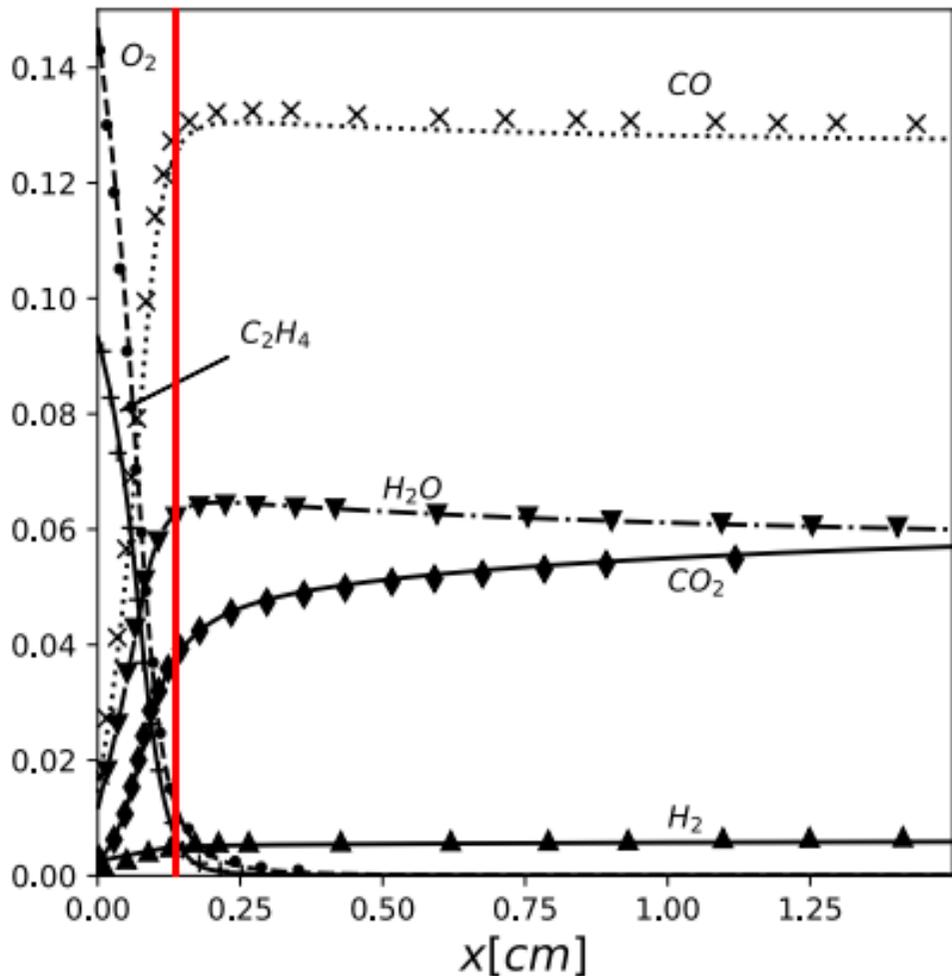






-  Conv(3)
-  Concatenate
-  Densely connected
-  Dropout (10%)
-  LSTM

Mass Fraction



Mass Fraction

