J. Plasma Phys. (2026), *vol*. 92, E52
 doi:10.1017/S0022377826101561

Machine learning surrogates for ion energy–angle distributions in thermal and RF plasma sheaths

Mohammad Mustafa[1] **and Davide Curreli**[1]

[1]Nuclear, Plasma & Radiological Engineering, University of Illinois at Urbana Champaign, Urbana, IL 61801, USA

Corresponding author: Mohammad Mustafa, mustafa5@illinois.edu

(Received 13 September 2025; revision received 10 December 2025; accepted 2 March 2026)

Ion energy–angle distributions (IEADs) at material surfaces are a critical input for plasma–material interaction (PMI) studies in fusion devices, yet they are computationally expensive to obtain using particle-in-cell (PIC) simulations. In this work, we develop a machine learning surrogate based on a deep deconvolutional neural network (DDeCNN) trained on large databases generated with the hPIC2 code. The surrogate is capable of reconstructing IEADs from sheath parameters for both thermal and radio-frequency (RF) plasmas, including cases with multiple ion species. Across thousands of test cases, the model achieves high accuracy, with over 97 % of predictions classified as good or average based on standard error metrics (MAE, MSE, L2). Even in the more challenging RF and multi-species regimes, the surrogate reliably captures the multi-peak structure of PIC results. Once trained, the surrogate produces IEADs in milliseconds on a common workstation, yielding speedups of six to seven orders of magnitude compared with running a full PIC simulation. This computational gain enables dense parameter scans and direct coupling of IEAD predictions with PMI and erosion models on whole-device scales in fusion-relevant conditions.

Key words: plasma sheaths, fusion plasma

1. Introduction

In this work, we develop a machine learning (ML) surrogate model for predicting ion energy–angle distributions (IEADs) relevant to plasma–material interaction (PMI) in magnetic confinement fusion (MCF). IEADs determine how ions strike material surfaces, directly influencing sputtering, erosion and surface modification. Since the sputtering yield depends on both the ion impact energy E_i (eV) and the impact angle θ relative to the surface normal, accurate knowledge of IEADs at the material boundary is essential for estimating erosion fluxes, surface evolution and component lifetimes.

In MCF conditions, ions leave the quasi-neutral plasma after being accelerated by the magnetic presheath (Chodura 1982) and the electrostatic Debye sheath

(Tonks & Langmuir 1929); when radio-frequency (RF) heating systems are also present, an additional RF sheath forms (Butler & Kino 1963; Myra 2021). In conventional magnetised plasma sheaths, ions are accelerated by the sheath electric field, with energies and trajectories determined by plasma temperature, charge state, ion-to-electron mass ratio, and the strength and inclination angle of the magnetic field (Khaziev & Curreli 2015). While analytical models provide useful insights (Tonks & Langmuir 1929; Bohm 1949; Riemann 1991, 1994; Ahedo 1997; Stangeby *et al*. 2000; Lieberman & Lichtenberg 2005; Chen 2016; Moseev & Salewski 2019), they typically oversimplify ion dynamics and nonlinear sheath behaviour. RF sheaths introduce additional complexity; the rectified RF-driven sheath potential can significantly exceed the thermal sheath drop and the sheath itself oscillates with the applied RF fields, dynamically altering ion trajectories. This produces more structured, anisotropic IEADs, which can enhance sputtering and modify the surface response. The time-dependent RF potential also introduces temporal variations in ion impact conditions, further complicating ion-material interactions.

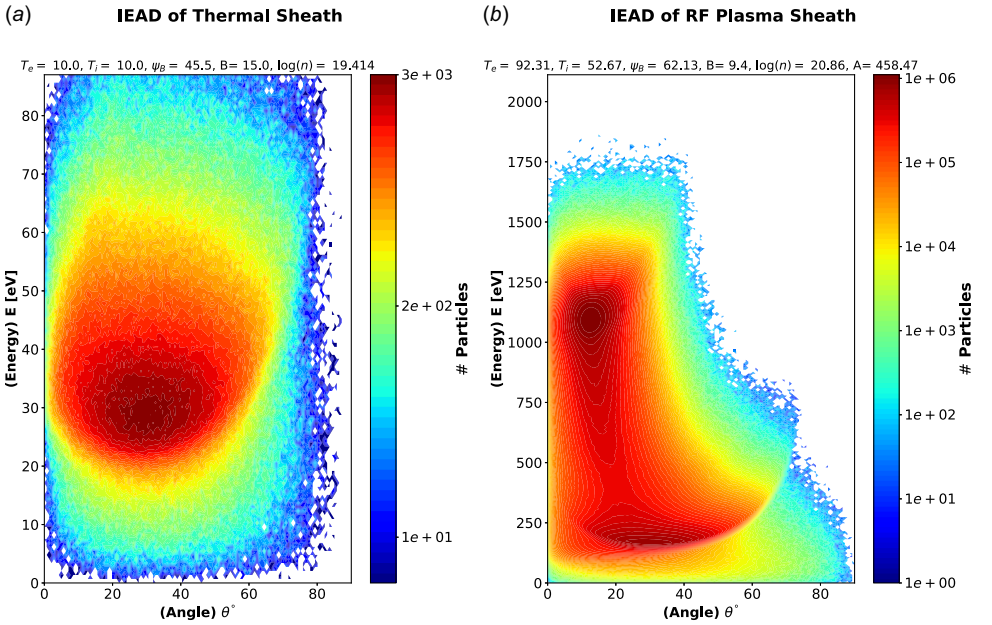
High-fidelity particle-in-cell (PIC) simulations are the most reliable tool for modelling IEADs, as they track ion dynamics with self-consistent fields through thermal and RF sheaths. By resolving ion trajectories and RF-induced oscillations, PIC simulations provide kinetic-level accuracy of IEADs. However, their computational expense makes them impractical for device-scale applications where IEADs must be evaluated across entire first-wall and antenna structures. For example, the codes hPIC (Khaziev & Curreli 2018) and hPIC2 (Meredith *et al*. 2023) have been used to resolve IEADs of thermal sheaths (Khaziev & Curreli 2015) and RF sheaths (Myra *et al*. 2020; Elias, Curreli & Myra 2021) with kinetic accuracy. Figure 1 shows two representative IEADs obtained from hPIC2 simulations for thermal (panel *a*) and RF sheath (panel *b*) conditions. Extending PIC simulations across an entire tokamak surface is prohibitively expensive, motivating the need for reduced-order surrogate models.

Machine learning (ML) based reduced order models are increasingly used in plasma fusion to accelerate simulations and improve predictive capabilities. They enable fast modelling of plasma behaviour, disruption forecasting and real-time control optimisation (Parsons 2017; Kates-Harbeck, Svyatkovskiy & Tang 2019; Fu *et al*. 2020). ML also helps uncover patterns in plasma dynamics and supports studies of material erosion, sputtering and impurity transport (Kino *et al*. 2021; Lin *et al*. 2024). Additional applications include accelerating turbulence simulations, enhancing diagnostic interpretation, and aiding in the design of reactor components and materials (Piccione *et al*. 2020; Chang *et al*. 2021; Dong *et al*. 2021; Lao *et al*. 2022).

Machine learning and surrogate models are revolutionising RF plasma heating research by drastically reducing computational complexity while maintaining high predictive accuracy. Multiple studies highlight this potential: Sánchez-Villar *et al*. (2024, 2025) developed real-time ion cyclotron range of frequencies (ICRF) heating models using random forest and neural networks, achieving speedups of six to seven orders of magnitude and high regression scores. Wallace *et al*. (2022) similarly accelerated lower hybrid current drive simulations from minutes to milliseconds.

Reduced-order modelling approaches for the determination of IEADs have been previously explored. Seleson *et al*. (2022) explored a least-squares fitting, sparse grid, coordinate transformation method for single-species thermal sheaths with up to 4 input parameters. While successful in reconstructing IEADs with 6–7 orders of

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press



IEAD Under Thermal plasma Condition IEAD Under RF plasma Condition

FIGURE 1. Examples of ion energy–angle distributions (IEADs) as obtained from hPIC simulations under (*a*) thermal sheath and (*b*) radio-frequency (RF) sheath conditions. Those two distributions are for illustrative purposes only, to show the nature of the two-dimensional distributions that the ML model will have to be able to reconstruct.

magnitude speedup compared with hPIC simulations, these approaches are difficult to extend to larger parameter spaces or to RF sheaths. For example, with seven input parameters, over 19 000 simulations are required. Additionally, RF-induced IEADs often exhibit multi-peaked structures that are not well captured by coordinate transformation methods. Artificial Neural Networks (ANNs) offer a more scalable and flexible alternative. By training neural networks on large high-fidelity datasets of PIC simulations, surrogate models can reproduce IEADs with near PIC-level accuracy at a fraction of the computational cost.

Several neural network architectures are suitable for the current modelling task, each with distinct advantages and limitations. A fully connected multilayer perceptron (MLP) can directly map input vectors to the unrolled IEAD, but demands substantial memory for high-resolution outputs. Diffusion models (Sohl-Dickstein *et al*. 2015) offer high-quality distribution reconstruction, but require extensive training data, significant computational resources, and exhibit slower training and inference. Transformers with two-dimensional (2-D) positional encoding (Wu *et al*. 2021; Vaswani *et al*. 2023) effectively capture global spatial dependencies, making them well suited for structured outputs, though they are computationally and memory intensive. FiLM-conditioned U-Nets (Meseguer-Brocal & Peeters 2019) provide a flexible mechanism for incorporating conditioning information and are efficient for structured generation tasks, but may be less effective in modelling long-range dependencies.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

In this paper, we present a machine learning surrogate model for reconstructing IEADs with PIC-level fidelity across a wide parameter space. The model combines multilayer perceptrons (MLPs) with convolutional neural networks (CNNs) in a hybrid deep deconvolutional neural network (DDeCNN) architecture, enabling fast and accurate emulation of IEADs under both thermal and RF sheath conditions. The motivations behind this architecture include its computational efficiency, support of multi-output design, spatial coupling and space invariance. The training datasets were generated using large-scale hPIC2 simulations covering a wide range of physical parameters (temperature, density, RF voltage, magnetic field angle and strength) and numerical parameters (time step, grid size, particles per cell). The multi-dimensional input parameter space was sampled using a Latin hypercube design (McKay, Conover & Whiteman 1976) to ensure broad coverage. Testing and validation demonstrate that our ML surrogate model reproduces PIC-level IEADs at a fraction of the computational cost. The surrogate is constructed for both thermal and RF sheath conditions, including single- and multi-species plasmas, making it applicable to divertor, first-wall and RF-enhanced plasma–surface interaction scenarios. We systematically evaluate model accuracy using multiple error metrics (MAE, MSE, relative L2 norm) and demonstrate substantial computational speedups. The remainder of this paper is structured as follows. Section 2 details the problem set-up, datasets and neural network architecture. Section 3 presents results for thermal, RF and multi-species plasmas. Section 4 summarises conclusions and outlines future directions.

2. Methodology

This section describes the methodology used to construct and validate the machine learning surrogate model of ion energy–angle distributions (IEADs). We first summarise the physical problem (§ 2.1), then define the input parameter space (§ 2.2) and outline the data generation process using hPIC2 simulations (§ 2.3). Subsequent steps of data preprocessing, network design and error evaluation are discussed in §§ 2.4–2.7.

2.1. *Physics problem description*

The IEADs were obtained using the hPIC2 PIC code (Khaziev & Curreli 2018; Meredith *et al*. 2023), which self-consistently resolves ion impact distribution functions in magnetised plasma sheaths. A sketch of the simulations domain is shown in figure 2. The simulation domain consists of a plasma slab bounded by two grounded plates ($V = 0$) in the presence of a magnetic field inclined at an angle ψ_B relative to the surface normal, following the classic Chodura sheath configuration (Chodura 1982; DeWald, Bailey & Brooks 1987; Devaux & Manfredi 2006; Khaziev & Curreli 2015). A volumetric particle source maintains quasi-neutrality by replenishing particles lost to the walls. Ion motion is integrated in three spatial dimensions, while the electrostatic potential is solved along the direction perpendicular to the material wall. Thermal sheath cases were evolved to steady state, and the energy and impact angle of ions at the walls were recorded to construct the IEADs. For RF sheath simulations, boundary conditions were that we imposed antisymmetric sinusoidal plate voltages, $V_1 = V_{\text{pp}} \sin(\omega t)$ and $V_2 = V_{\text{pp}} \sin(\omega t - \pi)$, where ω is the RF frequency and V_{pp} is the peak-to-peak applied voltage. This configuration produces rectified RF sheath potentials and reproduces the physics of RF rectification (Elias *et al*. 2019; Myra *et al*. 2020; Elias *et al*. 2021; Rezazadeh, Myra & Curreli 2023).

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

	Units	Physical parameters		Units	Numerical parameters
T_e	eV	Electron temperature	dx	m	Cell length
T_i	eV	Ion temperature	N_{part}		Computational particles
B	T	Magnetic field strength	t	s	Simulation time
ψ_B	deg	Magnetic field angle	dt	s	Time step
n	m^{-3}	Plasma density	N_{steps}		Number of time steps
V_{pp}	V	RF voltage (peak-to-peak)	x	m	Domain length

TABLE 1. List of input parameters used in the ML model.

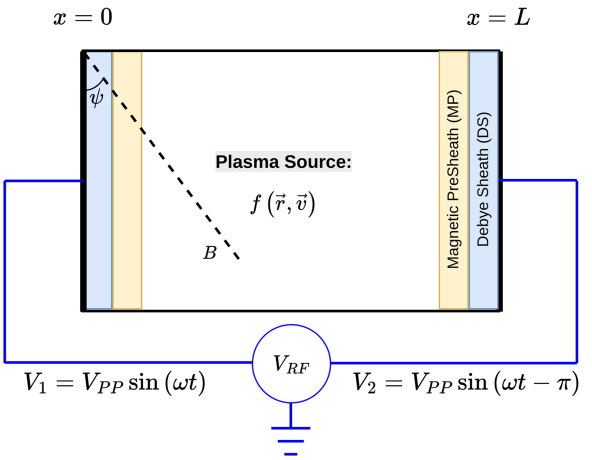


FIGURE 2. Sketch of the hPIC2 simulation domain under RF plasma sheath conditions. The figure illustrates a dual-plate structure driven by antisymmetric sinusoidal voltages, $V_1 = V_{\text{pp}} \sin(\omega t)$ and $V_2 = V_{\text{pp}} \sin(\omega t - \pi)$, along with the volumetric plasma source and resulting sheath regions. For thermal sheath conditions, both boundaries are grounded ($V_1 = V_2 = 0$).

2.2. *Definition of the input parameter space*

The surrogate model must reproduce IEADs across a broad range of plasma and numerical conditions. The input vector therefore includes both physical parameters (governing sheath dynamics) and numerical parameters (governing PIC discretisation), summarised in table 1.

For thermal sheath cases, four physical parameters were varied: electron-to-ion temperature ratio (T_e/T_i), plasma density n, magnetic field magnitude B and field angle ψ_B. This choice follows from physical considerations and sensitivity studies in earlier work (Seleson *et al*. 2022; Mustafa 2023).

For RF sheath cases, an additional parameter, the RF peak-to-peak voltage V_{pp}, was included. While using temperature ratio T_e/T_i is adequate for describing normalised ion energy–angle distributions (IEADs) under steady thermal sheath conditions, it becomes insufficient in RF sheaths. The time-varying RF potential causes ions to experience phase-dependent acceleration, making the IEAD depend not only on T_e/T_i, but also on the normalised RF amplitude eV_{pp}/T_e and the ratio of the RF period to the ion transit time $\omega_{\text{RF}}\tau_i$. Accordingly, in our model, T_e and

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

T_i were treated as separate parameters, and an additional parameter V_{pp} (the RF peak-to-peak voltage) was introduced to capture these RF-induced effects.

Additionally, up to six numerical parameters (e.g. time step, cell length, particles per cell, domain length) were included as additional inputs. While not strictly 'free' physical parameters, including them ensures the ML surrogate captures subtle dependencies on discretisation and avoids misinterpreting numerical artefacts as physical trends. Constraints on these parameters are detailed in § 2.3 and Appendix A.

2.3. *Data generation*

Training and testing datasets were generated using high-resolution simulations run with the hPIC2 code (Meredith *et al*. 2023). Additionally, we were also able to use a portion of training data from earlier studies (Seleson *et al*. 2022; Mustafa 2023), which were obtained from the hPIC code (Khaziev & Curreli 2015). The two PIC codes share the same physics kernel and for the same inputs, provide identical output, the only difference being hardware (GPU) acceleration implemented in hPIC2, not present in hPIC. Output benchmarks between the two codes were run to guarantee consistency among datasets. The data generation process followed four steps.

(i) Sampling of input parameters. Ranges of physical parameters were selected to match conditions relevant to fusion edge plasmas (table 2) (Keilhacker *et al*. 1982; Stangeby *et al*. 2000; Porter *et al*. 2010; Wesson & Campbell 2011; Ciraolo *et al*. 2019; Pitts *et al*. 2019). Sampling employed a Latin hypercube sampling (LHS) design with corner points explicitly included, ensuring uniform coverage and good representation of extreme cases (McKay *et al*. 1976, 1979), including Lloyd algorithm (Lloyd 1982).

(ii) Evaluation of numerical parameters. For each sampled point, numerical resolution parameters (e.g. grid size, time step, particles per cell) were determined by enforcing resolution constraints to guarantee physical fidelity. Details are given in Appendix A.

(iii) Execution of hPIC2 simulations. Input files were automatically generated for each parameter set and run on the Campus Cluster at the University of Illinois Urbana Champaign. At the wall, ion impact energies and angles were recorded to build the IEADs.

(iv) Dataset organisation. Each database was divided into training, validation and testing subsets. Testing datasets were generated independently (not subsets of training data) to preserve statistical independence and provide unbiased performance evaluation.

Separate databases were created for thermal and RF sheaths, and for single- and multi-species plasmas (see table 2). The range of each free parameter is reported in table 2. Each dataset was first divided into two main subsets, training set and testing set. A portion of the training dataset was further split off as a validation set for hyperparameters tuning and model evaluation during training. The testing set was reserved exclusively for evaluating the model performance after training was complete. The model remains agnostic to the testing set, as it is not used at any point during the training or validation process. After selecting the hyperparameters, the validation dataset is merged back into the training set and the model is retrained using the

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

Databases		1	2	3	4	5	6	7	8
Sheath		Thermal	Thermal	RF	RF	RF	RF	RF	RF
Species		H^+	H^+	D^+	D^+ T^+ O^{8+} W^{40+} He_3^{2+}	D^+	D^+ T^+ W^{40+}	D^+ T^+	He_4^{2+} O^{8+} W^{40+}
No. free parameters		2	4	6	7	6	6	6	5
Input ranges	T_e	$T_e/T_i =$	$T_e/T_i =$	[1, 200]	[1, 500]	[1, 200]	[1, 500]	[1, 500]	$T_e = T_i =$
	T_i	$[10^{-2}, 10^2]$	$[10^{-2}, 10^2]$	[1, 200]	[1, 500]	[1, 200]	[1, 500]	[1, 500]	[1, 500]
	ψ_B	[0, 87]	[0, 87]	[0, 89]	[0, 89]	[0, 89]	[0, 89]	[0, 89]	[0, 89]
	B	—	[7, 15]	[7, 15]	[7, 15]	[7, 15]	[7, 15]	[7, 15]	[7, 15]
	$\log_{10}(n)$	—	[17, 21]	[17, 21]	[17, 21]	[17, 21]	[17, 21]	[17, 21]	[17, 21]
	V_{pp}	—	—	[1, 200]	[1, 200]	[1, 500]	[1, 500]	[1, 500]	[1, 500]
	Misc.	—	—	—	T_{He_3}/T_i = [1, 10]	—	—	—	—
Sampling method	Training	Uniform	Uniform	LHS + corners	LHS + corners	LHS	LHS	LHS	LHS
	Testing	Val split	Val split	Random	Random	—	random	random	random
No. samples	Training	360	500	1580	1995	4995	2000	1000	2000
	Testing	40	125	100	85	—	100	100	100
Notes	——	—	$f = 56$ MHz†	$f = 120$ MHz	$f = 56$ MHz‡	$f = 120$ MHz	f = 120 MHz	f = 56 MHz	

TABLE 2. Summary of the databases generated using hPIC2 and used to train the ML model.

†This database was also used for the data augmented (DA) model and served as the high-fidelity (HF) database for the multi-fidelity (MF) model, as discussed in § 3.2.1. After data augmentation, the total number of data points increases to 38 732.

‡This database was used as the low-fidelity (LF) database for the MF model described in § 3.2.1. When the MF and DA models were combined, a total of 124 812 data points were augmented from this database.

combined data. To generate the training set, a total of N_{sample} samples are drawn from the input parameter space. To ensure that these samples adequately cover the space, first we selected 2^{N_D} corner points of the input hypercube, where N_D is the number of free parameters in the model. The remaining $N_{\text{sample}} - 2^{N_D}$ points are then generated using LHS (McKay *et al*. 1976, 1979), with Lloyd algorithm (Lloyd 1982), which iteratively relocates samples towards the centroids of their Voronoi cells to improve space-filling uniformity and minimise clustering. Furthermore, N_{test} testing points are generated independently and randomly, separate from the training set. Although it is common to select testing data as a subset of the training dataset, doing so would reduce the number of available training points in what is already a sparsely sampled space. A routine was implemented to ensure that the testing data points do not overlap with, or closely resemble, the training or validation data points. Validation datasets were randomly sampled as a subset from the training dataset. In some cases, additional data augmentation was performed to increase temporal and spatial diversity (discussed in § 2.6).

2.4. *Data normalisation*

The input data span several orders of magnitude, which can negatively affect ML training by slowing convergence and biasing optimisation. Large variations in input scales can significantly slow down the convergence of gradient descent algorithms (Ioffe & Szegedy 2015). Moreover, applying feature scaling is especially important when regularisation is used as part of the loss function, as unscaled features can lead to biased or unstable optimisation. To mitigate this, input features were normalised using min–max scaling (Patro & Sahu 2015),

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}, \tag{2.1}$$

where x is the raw value, and $x_{\min}$, $x_{\max}$ are the limits of the sampled range. However, direct min–max scaling was not suitable for our output data (IEADs). This is because IEADs span many orders of magnitude: counts near the distribution boundaries may be very small, while the peak can be several orders larger. Using min–max would (i) obscure small but physically meaningful values at the tails and (ii) risk data leakage, since the maximum particle count is only known after the simulation. Instead, IEADs were preprocessed in two steps.

(i) Conversion to computational units. The accumulated physical particles were converted to computational counts using the physical-to-computational (P2C) ratio defined in hPIC2:

$$\text{P2C} = \frac{nx}{N_{\text{PPC}} N_{\text{cells}}}, \tag{2.2}$$

where n is particle density per unit volume, x is the domain size in [m], N_{PPC} is the number of particles per cell and N_{cells} is the number of cells.

(ii) Logarithmic scaling. The IEADs values were then transformed using a base-10 logarithm, which compresses the dynamic range while preserving the distribution shape.

Data preparation of the IEADs is shown schematically in figure 16 reported in the Appendix. In the case of multi-species, IEADs are arranged as a four-dimensional (4-D) matrix [three-dimensional (3-D) matrix for each data point] as shown in figure 16(*b*).

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

2.5. *Machine learning models*

The two neural network paradigms considered in this work were the following.

(i) MultiLayer perceptron (MLP/ANN). Classical fully connected feed-forward networks (Rumelhart, Hinton & Williams 1986; Bishop 1995) with input, hidden and output layers. While straightforward, MLPs require large parameter counts to represent high-resolution distributions.

(ii) Convolutional neural networks (CNNs). Architectures that exploit spatial structure by applying learnable filters across input data (LeCun *et al*. 1989, 2002; Alzubaidi *et al*. 2021). CNNs reduce parameter count, improve generalisation and are particularly well suited to 2-D outputs such as IEADs (treated as 'images').

MLP consists of interconnected nodes, known as neurons, organised in layers. These layers typically include an input layer, one or more hidden layers and an output layer. Information flows through the network from the input layer, where data are fed into the model, through the hidden layers, where computations are performed, to the output layer, which produces the model predictions or classifications. Each connection between neurons is associated with a weight that determines the strength of the connection. During the training process, these weights are adjusted iteratively to minimise the difference between predictions and outcomes.

Because IEADs are inherently 2-D distributions with strong spatial correlations, CNN-based methods were deemed essential. CNNs achieve regularisation by using fewer connections than MLPs and regularised weights. For instance, while a fully connected layer would demand an extensive number of weights to process a high-resolution image, CNNs efficiently reduce this requirement. Moreover, CNNs efficiently learn feature representations through filter optimisation, overcoming challenges like vanishing and exploding gradients, while simultaneously reducing computational complexity through hierarchical feature extraction.

2.6. *Network architecture*

For the reconstruction of IEADs, we developed a hybrid deep deconvolutional neural network (DDeCNN) that combines MLP and CNN elements (schematic in figure 3) structured as follows.

(i) MLP block. The input parameter vector passes through fully connected layers, progressively expanding the feature space and capturing nonlinear dependencies.

(ii) Reshape layer. The MLP output is reshaped into a low-dimensional matrix suitable for convolutional operations.

(iii) Deconvolutional block. A sequence of transposed 2-D convolutional layers (or informally, 'deconv' layers) progressively upsamples the matrix, doubling its height and width at each stage while reducing depth. This reconstructs the 2-D structure of the IEAD.

(iv) Final convolution. A 2-D convolutional layer reduces the output to one channel (single-species) or multiple channels (multi-species), matching the number of IEADs to be reconstructed.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

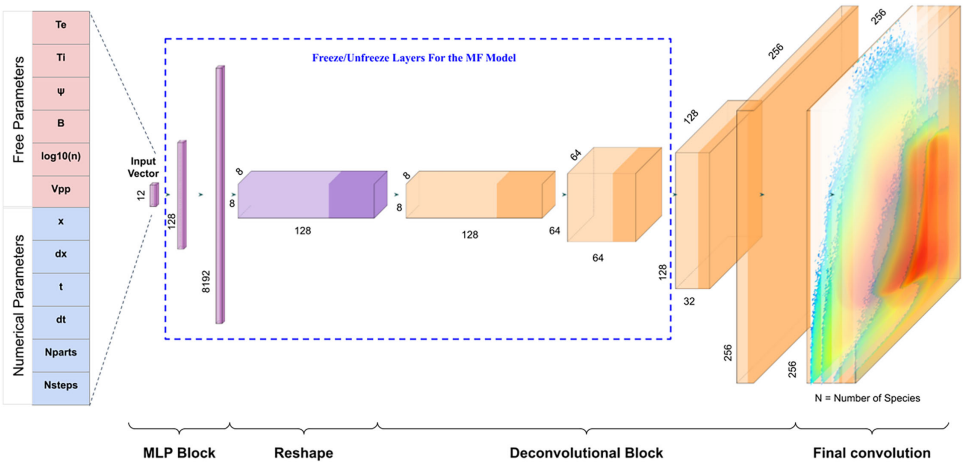


FIGURE 3. Hybrid deep deconvolutional neural network (DDeCNN) architecture used to construct the machine learning surrogate model of ion energy–angle distributions (IEADs) from hPIC2 data. The input vector includes up to 13 parameters (up to seven physical and up to six numerical, listed in tables 1 and 2). These parameters are first processed through a multilayer perceptron (MLP), reshaped and passed into a sequence of transposed 2-D convolutional ('deconv') layers that progressively upsample the feature maps to reconstruct the 2-D IEADs. A final 2-D convolutional layer produces the output channels corresponding to the number of ion species. ReLU activations (Fukushima 1969) are used throughout. The dashed blue box highlights the subset of layers that are frozen and unfrozen during multi-fidelity (MF) training, enabling knowledge transfer from low-fidelity to high-fidelity datasets.

The rationale for this choice is driven by the 2-D nature of the IEADs. Pure MLPs are inefficient for large 2-D outputs, requiring excessive memory. Diffusion models (Sohl-Dickstein *et al*. 2015) and transformers (Wu *et al*. 2021; Vaswani *et al*. 2023), while powerful, are computationally costly and require much larger training datasets. Deconvolutional CNNs exploit the structured nature of IEADs, enabling accurate and efficient reconstruction with fewer trainable parameters. All layers use the ReLU activation function (Fukushima 1969), which mitigates vanishing gradients, introduces sparsity and preserves spatial continuity. Hyperparameters (learning rate, number of layers, filter sizes, etc.) were tuned via random search (Feurer & Hutter 2019). Dataset-specific hyperparameter selections are reported in Appendix C.

Architecture enhancements. To extend the number of training data points in a given dataset without running more expensive PIC simulations, data augmentation was performed by extracting IEADs at different time steps and from different boundaries. This allowed expanding the dataset diversity at negligible cost and enabled the reconstruction of time-dependent IEADs necessary for RF sheath problems.

We also tested multi-fidelity (MF) learning. Since the number of high-fidelity (HF) data points available for training is limited, as shown in table 2, additional data can be generated at a reduced computational cost using the MF framework. To reduce reliance on expensive HF data, coarse-resolution hPIC2 simulations were used as low-fidelity (LF) data. Training proceeded in three stages: (i) pre-train on LF data; (ii) freeze early layers, retrain on HF data; and (iii) unfreeze all layers and fine-tune on HF data with reduced learning rate. This approach efficiently transferred trends from LF to HF, reducing overfitting and improving generalisation.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

2.7. *Error evaluation*

No single metric can fully characterise surrogate accuracy. Therefore, three complementary error measures were used: the mean squared error (MSE), relative L_2 norm error (RL_2E) and the mean absolute error per pixel (MAE). Each of these metrics captures different aspects of model performance. MSE emphasises larger errors due to its squared nature, making it sensitive to outliers and useful for identifying occasional large deviations. RL_2E provides a normalised measure of error relative to the magnitude of the true data, which is helpful for comparing performance across datasets or scales. MAE per pixel offers an intuitive average error per prediction unit, giving a straightforward sense of overall accuracy.

$$\mathrm{MSE} = \frac{1}{N \times M} \sum_{i=1}^{N} \sum_{j=1}^{M} \left(y_{ij} - \hat{y}_{ij}\right)^2, \tag{2.3}$$

$$\mathrm{RL_2E} = \frac{\left[\sum_{i=1}^{N} \sum_{j=1}^{M} \left(y_{ij} - \hat{y}_{ij}\right)^2\right]^{1/2}}{\left[\sum_{i=1}^{N} \sum_{j=1}^{M} \left(y_{ij}\right)^2\right]^{1/2}}, \tag{2.4}$$

$$\mathrm{MAE} = \frac{1}{N \times M} \sum_{i=1}^{N} \sum_{j=1}^{M} \left|y_{ij} - \hat{y}_{ij}\right|, \tag{2.5}$$

where N and M are the dimensions of the 2-D output, y_{ij} is the actual data value of the ith angle and jth energy pixel indices, normalised between $[0, y_{ij}^{\max}]$ (the number of particles accumulated on each pixel, as simulated by hPIC2), and $\hat{y}_{ij}$ is the ML-predicted value. For consistency, predicted IEADs were first denormalised (§ 2.4) and then rescaled by dividing by their maximum particle count, constraining all values between 0 and 1.

2.8. *Computational resources*

The generation of training and testing datasets required large-scale computational resources. All hPIC2 simulations were carried out on the University of Illinois Urbana–Champaign (UIUC) Campus Cluster, where up to 40 nodes were employed. Each node consisted of two Intel Xeon Processor E5-2670 v2 2.5 GHz with 10 cores each (20 cores per node) and 256 GB of RAM. On average, the generation of one database listed in table 2 required approximately five days using 40 nodes (∼100 000 CPU core-hours). Post-processing of the IEADs was performed using a local workstation equipped with dual Intel Xeon Processor E3 v3 and 96 GB of RAM.

Training and testing of the ML surrogate models were performed on GPU-powered systems. The primary training runs were conducted on the MIT Plasma Science and Fusion Center (PSFC) GPU cluster, where each node is equipped with four NVIDIA V100 GPUs, while additional experiments were performed on consumer-level equipment (NVIDIA GTX 1080 GPU) and industry-level high-end clusters (NVIDIA H100 Tensor Core GPUs). All workflows, including preparation of hPIC2 input files, post-processing of simulation outputs and ML model training, were implemented in Python. Neural network architectures were constructed and trained using the open-source libraries Keras (Chollet *et al*. 2015) and TensorFlow (Abadi *et al*. 2015), with supplementary data handling and preprocessing performed using scikit-learn (Pedregosa *et al*. 2011).

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

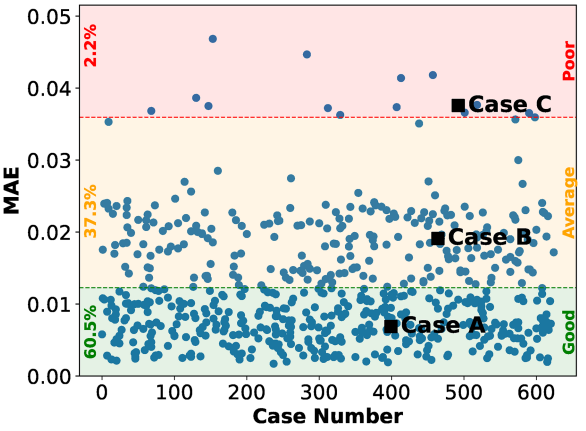


FIGURE 4. Distribution of prediction errors for the thermal sheath database using the MAE metric. Errors are classified into three categories: good ($E < \bar{E}$), average ($\bar{E} < E \leqslant \bar{E} + 3\sigma_E$) and poor ($E > \bar{E} + 3\sigma_E$). The majority of test cases fall in the good or average categories, with fewer than 3 % identified as poor. Representative cases from each category (labelled *a*, *b*, *c*) are highlighted and shown in detail in figure 5.

3. Results

3.1. *Thermal plasma sheaths*

The surrogate model was first evaluated under thermal sheath conditions for a single-species hydrogen plasma. This case was chosen as an initial benchmark because the resulting IEADs are relatively simple. The distributions can be described by a small set of parameters and they typically exhibit a single peak.

The training dataset consisted of 625 data points sampled uniformly across four free parameters: electron-to-ion temperature ratio (T_e/T_i), magnetic field strength B, plasma density n_e and magnetic field inclination angle ψ_B. Approximately 10 % validation split was applied, yielding 560 training samples and 65 testing samples. Hyperparameters for the DDeCNN model were selected through random search, with details provided in table 8 (Appendix C). Model performance was assessed using the three error metrics defined in § 2.7 (MAE, MSE and relative L_2 norm). For each metric, prediction errors were sorted and grouped into three categories: good ($E < \bar{E}$), average ($\bar{E} < E \leqslant \bar{E} + 3\sigma_E$) and poor ($E > \bar{E} + 3\sigma_E$), where $\bar{E}$ is the mean error and σ_E its standard deviation. Figure 4 illustrates this classification using MAE as the error measure, with representative cases from each category (labelled *a*, *b*, *c*) shown in figure 5.

Overall, the ML surrogate reproduced the IEADs with high fidelity. Approximately 60 % of the test cases fell into the 'good' category, while fewer than 3 % were classified as 'poor' predictions (table 3). The small number of poor cases was primarily associated with simulations where the number of accumulated particles was too low, leading to noisy IEADs. In these situations, the ML model tended to smooth the noise because convolutional kernels inherently impose spatial coherence due to their shared-weight architecture, producing smoother but slightly biased reconstructions. Figure 5 compares three representative examples of ML predictions with the corresponding hPIC2 distributions. Case A ('good') demonstrates excellent agreement in both the peak and the tails of the IEAD. Case B ('average') shows moderate deviation but still reproduces the overall distribution shape. Case C

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

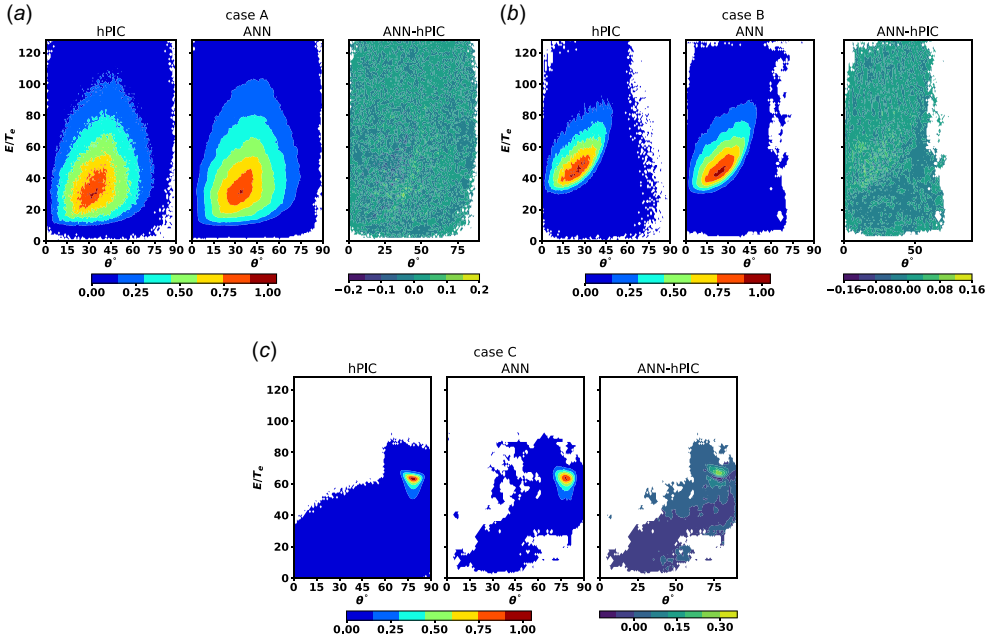


FIGURE 5. Comparison between hPIC2 IEADs (left column), ML surrogate predictions (middle column) and map of the difference between the IEADs of hPIC2 -ANN (right column) for three representative thermal sheath cases selected from figure 4. Case A ('good') illustrates excellent agreement, with the surrogate accurately reproducing both the peak and tails of the distribution. Case B ('average') demonstrates moderate deviations while maintaining the overall distribution shape. Case C ('poor') corresponds to a poor prediction. The associated MAE for examples A, B and C are 0.0067, 0.0191 and 0.0376 respectively.

('poor') highlights the defects of a poor prediction; here, the original hPIC2 result was dominated by noise, which the ML model partially smoothed, but could not fully recover.

Additional evaluation using the MSE and relative L_2 norm confirmed the robustness of these results. According to the MSE metric, 63 % of predictions were classified as good, 33 % as average and 3 % as poor. Using the L_2 norm, the corresponding values were 70 %, 28 % and 2 %, respectively (table 3). These consistent statistics across multiple metrics confirm that the DDeCNN surrogate is able to reliably reproduce thermal sheath IEADs at a fraction of the computational cost of PIC. Table 3 shows a summary of error brackets for different metrics. More details on error statistics can be found in table 6 in Appendix B.

3.2. *RF plasma sheaths*

3.2.1. Single-species RF sheaths

The surrogate model was first tested for single-species deuterium plasmas in the presence of RF sheaths. Compared with the thermal sheath case, RF sheaths present a significantly more challenging prediction task. The oscillatory sheath potential introduces strong temporal modulation of ion acceleration, resulting in structured IEADs with multiple characteristic features. Instead of a single, nearly symmetric distribution, the RF IEADs exhibit broad energy spreads, RF-phase-dependent

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

Model	Error metric	% Good cases	% Average cases	% Poor cases
Thermal	MAE	60.5	37.3	2.2
	MSE	63.2	33.4	3.36
	L_2	69.7	28.32	1.92
RF (DA + MF)	MAE	63	36	1
	MSE	68	30	2
	L_2	66	33	1
Five species RF (hybrid SSIM loss)	MAE	65.97	32.65	1.47
	MSE	67.43	30.87	1.70
	L_2	65.59	33.35	1.06

TABLE 3. Summary of the distribution of error brackets across all points in the thermal sheath dataset and RF sheath testing datasets based on various error metrics.

angular anisotropy and, in many cases, multi-peaked structures associated with ion mobility in the oscillating sheath. These features are particularly important for PMI studies, as they directly influence sputtering yields and surface modification. To address the added complexity of RF sheaths, we investigated more advanced surrogate modelling strategies, namely data augmentation (DA), multi-fidelity (MF) and a combined DA + MF approach.

Construction of the RF database for DA and MF models. The RF database was constructed by sampling six physical parameters (T_e, T_i, n, B, ψ_B and V_{pp}) and six numerical parameters (dx, x, dt, t, N_{part}, N_{steps}), as summarised in table 1. Although the numerical parameters are not strictly free inputs, their inclusion significantly improved prediction accuracy (see Appendix A). Additional training data were generated through augmentation of existing simulations, leading to the DA model. An MF learning strategy was also employed, using two datasets: an LF dataset (dataset 5, table 2) and an HF dataset (dataset 3, table 2). The LF dataset, generated with coarse discretisations, initially contained 4995 cases, expanded via augmentation to 124 812. The HF dataset, generated with fine discretisations, contained 1579 cases, expanded to 38 732. For testing, a separate HF-resolution dataset of 100 cases was generated and augmented to 2475 cases.

Training the DA and MF models. Training the MF model required careful hyperparameter optimisation due to its multi-stage design: (i) training on LF data; (ii) transfer learning on HF data; and (iii) hyperparameter fine-tuning. Additional hyperparameters specific to MF, such as the number and location of frozen/unfrozen layers, were optimised during this process. When DA was combined with MF, a new set of hyperparameters was required. The DA model suffered from overfitting due to redundancy in the augmented data, and thus required fewer training epochs. A random search strategy was used for hyperparameter tuning of both DA and MF + DA models. The final selected hyperparameters are summarised in table 9. After tuning, each model was retrained, and figure 6 shows how the MF model systematically outperformed the baseline by reducing the loss (MAE) across all training stages.

Comparison across models. Figure 7 presents a systematic comparison of model accuracy using box-and-whisker plots of MAE. From left to right, the figure compares: the baseline model (no DA/MF), the DA model, the MF model and the

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

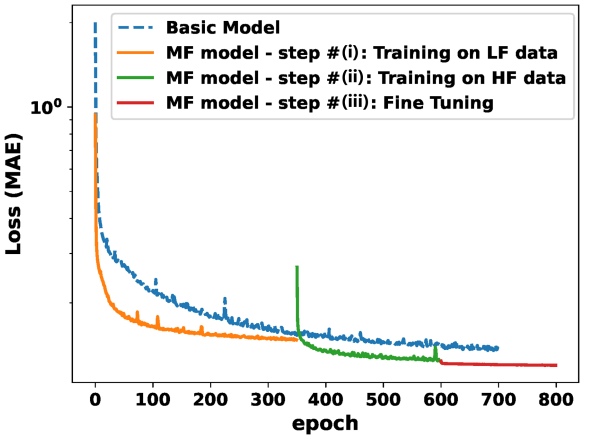


FIGURE 6. Evolution of the training loss function (MAE) during multi-fidelity (MF) training, compared with the baseline model. The three MF stages are shown: (i) initial training on the low-fidelity (LF) dataset; (ii) transfer learning on the high-fidelity (HF) dataset with partial layer freezing; and (iii) fine-tuning on HF data with all layers unfrozen. Each step yields systematic improvement over the baseline, demonstrating the effectiveness of the MF strategy.

combined MF + DA model, with 'last step' variants[1] also shown for DA and MF + DA. While the baseline model performed reasonably well, both DA and MF reduced the mean and median MAE by approximately 50 % compared with the baseline model, with MF achieving slightly better results. Combining DA and MF further reduced the error by another 50 % relative to the already improved result. Variance was moderately reduced in the DA model and strongly reduced in MF and MF + DA. Interestingly, MF + DA did not substantially outperform MF alone in terms of mean and variance, but produced a more stable error distribution. Figure 9 shows a representative example where the surrogate accurately reproduced double-peak features caused by RF rectification. In most cases, the MF approach provided sufficient accuracy for practical applications, though both DA and MF introduced additional computational overhead: LF data generation increases simulation cost for MF, while DA greatly enlarges the training dataset, requiring efficient data handling. Notably, DA can emulate IEADs across multiple time steps, making it particularly valuable when transient behaviour is of interest. Trends observed in the MAE were consistent with those for MSE and L_2 error, as shown in figure 8. More details on error statistics can be found in table 7 in Appendix B.

Time dependence. Some applications, such as RF-phase-resolved sputtering, turbulence studies and other plasma transient processes, require time-resolved IEADs. To evaluate model performance across time, we analysed the temporal evolution of MAE in the test dataset (figure 10). Combining MF with DA consistently reduced overall prediction error compared with DA alone, though the relative trends in error were preserved. High-error cases proved intrinsically more difficult to emulate, independent of the model. Elevated errors at early time steps were attributed to: (1) low particle statistics, which increase noise in the reference PIC data; and

[1]The last step variant refers to samples taken from the final time step of the data point, without data augmentation, matching the baseline model data point.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

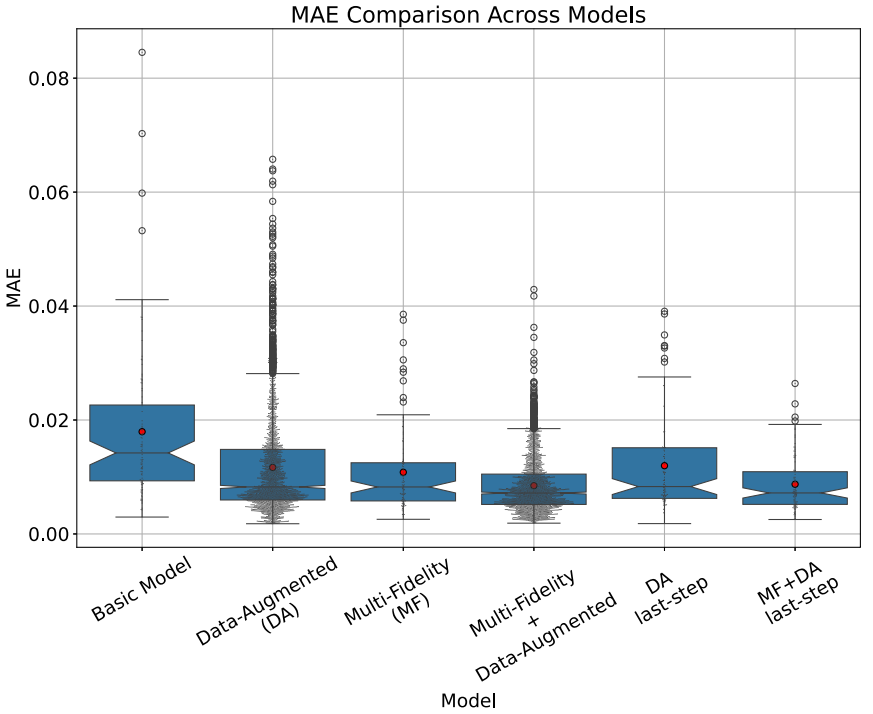


FIGURE 7. Box-and-whisker comparison of the mean absolute error (MAE) for IEAD predictions for single-species RF sheaths across different surrogate models: baseline, data-augmented (DA), multi-fidelity (MF) and the combined DA + MF approach. Models trained on augmented datasets were also evaluated on the original data points ('last step' cases). In each plot, the central line marks the median, the red dot indicates the mean, the blue box spans the interquartile range (50 % of the data), whiskers extend to the minimum and maximum values, and black circles denote outliers. Both DA and MF substantially reduce error relative to the baseline, with MF + DA providing the most stable error distribution.

(2) transient plasma states, which reduce the representativeness of training samples. In some cases, errors increased at later stages, coinciding with the appearance of secondary peaks in the IEADs due to sheath rectification, which added structural complexity.

The surrogate models for RF sheaths also demonstrated strong predictive capability for single-species RF sheaths. Data augmentation and multi-fidelity strategies each improved accuracy relative to the baseline, with MF providing the most consistent gains and DA offering advantages for time-dependent cases. The combined MF + DA approach yielded the most stable error distributions, though at the cost of additional data preparation and training complexity.

To address whether it is more beneficial to increase the number of HF simulations or to adopt an MF modelling approach, we considered both structure and computational cost of our datasets. The MF approach offers clear advantages in our context. Although HF simulations provide highly accurate results, they are computationally expensive; each HF run takes approximately three times longer than an LF simulation. By leveraging both LF and HF data, the MF model efficiently uses the broader coverage of the LF dataset while preserving the precision of the HF data.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

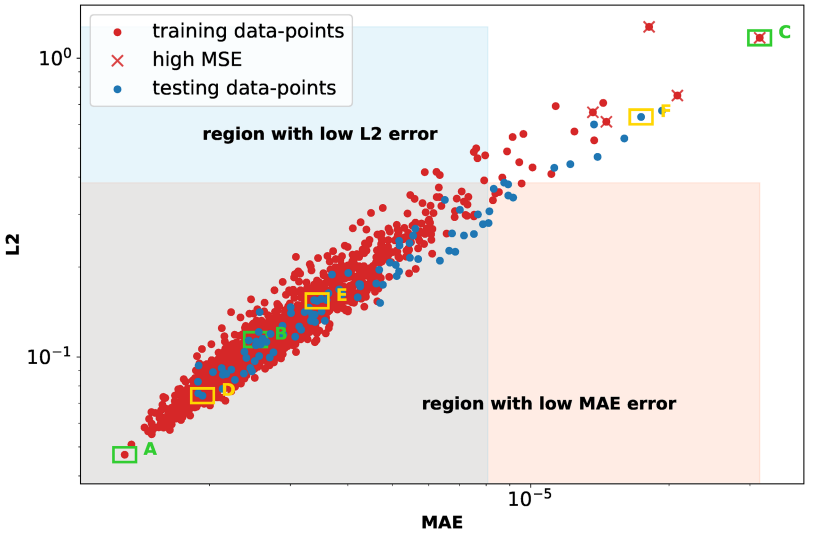


FIGURE 8. Distribution of error metrics for the training (red) and testing (blue) datasets. Data points within three standard deviations of the mean are shown for both MAE and L_2 error ($E_{\mathrm{MAE}}(i) \leqslant 3\sigma_{\mathrm{MAE}}$, $E_{L_2}(i) \leqslant 3\sigma_{L_2}$), while cases exceeding 3σ in MSE are marked with an 'X'. Representative cases were identified using a weighted average of all three error metrics: the best, average and worst predictions in the training set are labelled A, B and C, respectively, while those in the testing set are labelled D, E and F. Detailed comparisons for these cases are provided in figure 9.

Importantly, both datasets are generated using the same code (hPIC2) and resolve the physics accurately, with LF simulations already meeting high-quality standards (∼2000 particles per cell) and HF simulations further reducing PIC noise (∼10 000 particles per cell). The strong correlation between LF and HF data enables the MF model to learn effectively from both sources. Additionally, the MF framework supports transfer learning, allowing the model to generalise across a wider parametric space or adapt to related tasks with minimal additional cost. These factors make the MF approach a more scalable and resource-efficient strategy compared with simply increasing HF samples.

To further evaluate the performance of the surrogate models, the sputtering yields computed using IEADs generated by hPIC are compared with those reconstructed using the surrogate models, as shown in figure 11. The comparison includes both the training and testing datasets, and considers two variations of the surrogate model: the basic model and the multi-fidelity model. The sputtering yield at each point in the IEAD is calculated using the methods described by Yamamura & Shindo (1984) and Yamamura & Tawara (1996), demonstrating the robustness of the surrogate models in accurately predicting sputtering yields.

3.2.2. Multi-species RF sheaths

A more demanding test of the surrogate model involved plasmas containing multiple ion species in the presence of RF sheaths. Multi-species effects are ubiquitous in fusion-relevant plasmas, arising from fuel composition (D–T), fusion products (He_4), minority heating (He_3), intrinsic impurities (W, B, etc.), wall recycling and impurity

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

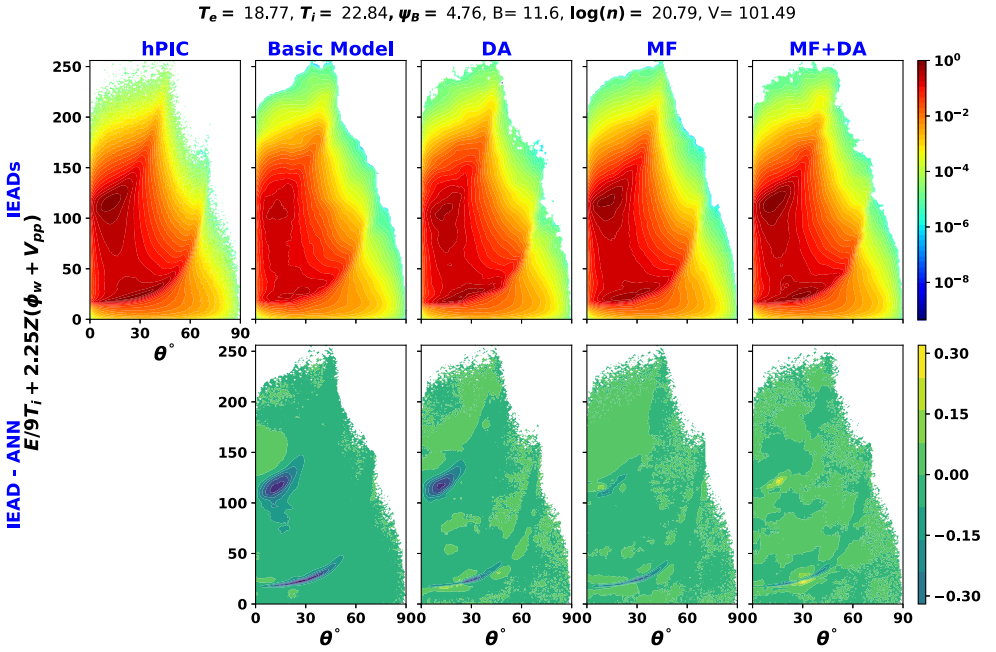


FIGURE 9. Example of IEAD predictions for a single-species RF sheath. The first row compares the reference distribution from hPIC2 (left) with the surrogate predictions from the baseline, data-augmented (DA), multi-fidelity (MF) and combined DA + MF models. The second row shows the corresponding residual error distributions for each model. The MF and DA + MF approaches most accurately reproduce the reference IEAD, including key features such as RF-induced broadening and multi-peaked structure, while also yielding smaller residuals. (Additional examples in figure 17.)

seeding. Under RF sheath conditions, these mixtures generate particularly complex IEADs: each ion species responds differently to the sheath potential depending on its charge-to-mass ratio, leading to species-dependent broadening, overlapping distributions and modified angular anisotropy.

For this problem, we adopted dataset 4 in table 2, corresponding to a case with helium-3 minority heating. The surrogate was trained to emulate IEADs for five species simultaneously: [D, T, O, He_3, W]. Because the minority He_3 population can reach temperatures well above the bulk ions, the helium-3 temperature T_{He_3} was included as an additional input parameter, defined as a multiple of the bulk ion temperature in the range $T_{He_3} = [1-10]T_i$. Unlike the single-species case, the outputs consisted of multiple IEADs, one per species, stored in separate output channels. This required a multi-output formulation of the surrogate, with the final convolutional layer producing multiple channels in parallel. Such a design ensured that correlations among species (e.g. common broadening due to the same sheath potential) were captured.

Training of multi-species problems. Training used 1995 independently generated cases and 100 test cases. Hyperparameter optimisation was performed via random search, with modifications introduced to handle the higher dimensionality of the multi-species problem (see table 10). The architecture was expanded to accommodate the multi-output design and the choice of loss function was carefully examined, since

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

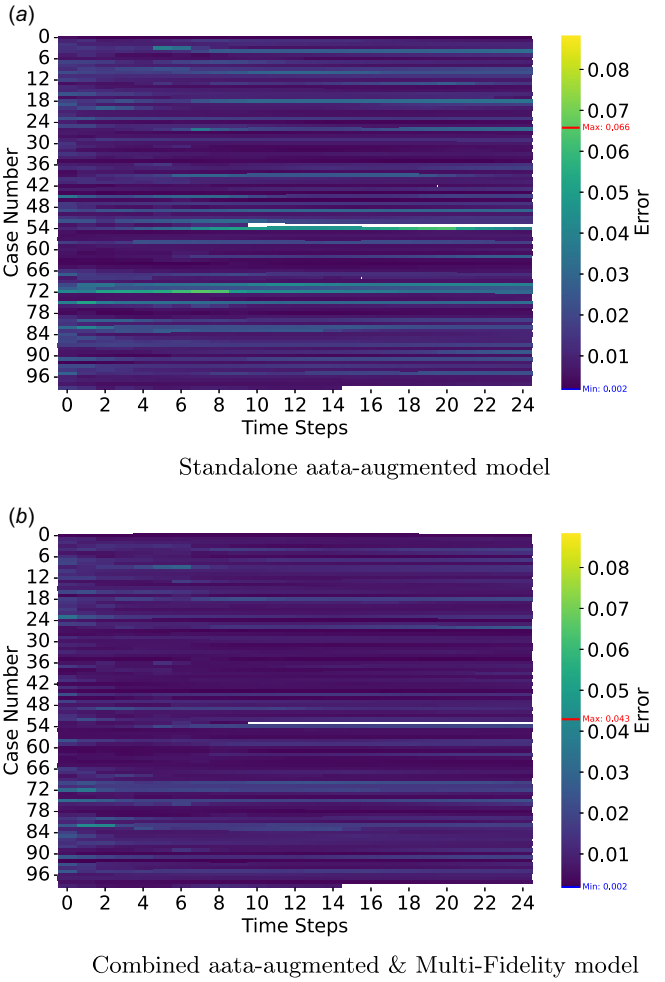


FIGURE 10. Heatmaps of the mean absolute error (MAE) on the testing dataset as a function of time step. Results for (*a*) the stand-alone data-augmented (DA) model and (*b*) the combined DA + MF model. Colour bars indicate the error range, with markers denoting the maximum and minimum values for each model. The DA + MF approach reduces overall errors compared with DA alone, while preserving the temporal trends in model performance.

accurate representation of all species simultaneously required more than standard point-wise error minimisation.

Loss function comparison. Several loss functions were evaluated (figure 12): mean absolute error (MAE), mean squared error (MSE), Wasserstein loss (Frogner *et al.* 2015), Huber loss, structural similarity index measure (SSIM) and a custom hybrid SSIM–MAE loss. The hybrid loss combined pixel-wise accuracy with structural fidelity:

$$\mathcal{L}_{\text{Hybrid}} = \alpha \cdot \mathcal{L}_{\text{SSIM}} + (1-\alpha) \cdot \mathcal{L}_{\text{MAE}}, \quad \alpha = 0.5, \tag{3.1}$$

where $\mathcal{L}_{\text{SSIM}} = C - \text{SSIM}$ and C is a normalisation constant (typically $C = 1$). Neither MAE nor SSIM alone was sufficient. When training with these metrics, the model prioritised certain species while neglecting others, resulting in incomplete

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

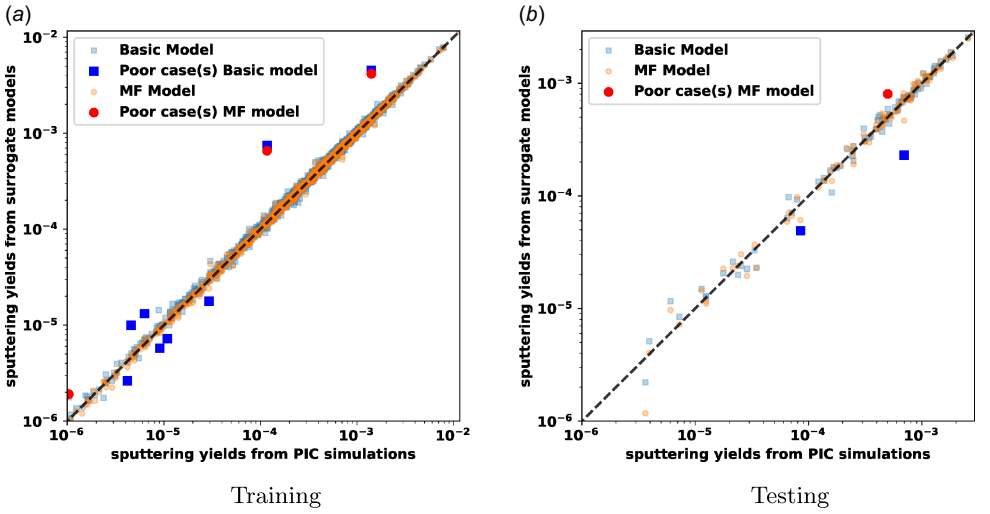


FIGURE 11. Comparison of sputtering yields calculated from distributions obtained using the ML surrogate model (y-axis), against sputtering yields obtained from PIC distributions (x-axis), for (a) training dataset and (b) testing dataset of RF sheath. Agreement between the two predictions is along the black dashed line. In each panel, two variations of the surrogate models are reported, namely basic model (light blue), and multi-fidelity model MF (light orange). Outlier cases classified as 'poor' predictions ($<1.85\,\%$ for the basic model and $<0.49\,\%$ for the MF model) have been highlighted in dark blue and dark red.

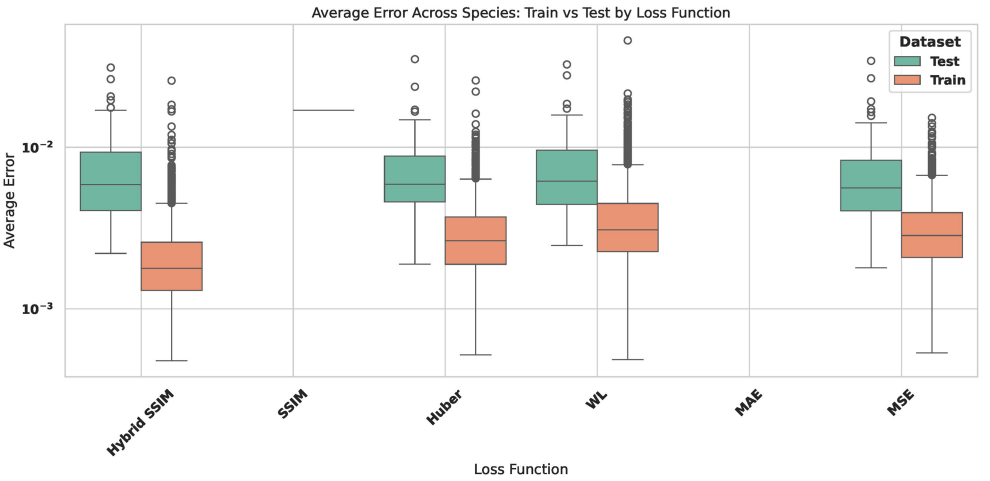


FIGURE 12. Box plot comparing the performance of different loss functions in emulating the IEADs of five species. Performance is evaluated using the mean absolute error (MAE), averaged over all species, showing both training and testing datasets. The loss functions compared, from right to left, are mean squared error (MSE), mean absolute error (MAE), Wasserstein loss, Huber loss, structural similarity index measure (SSIM) and a hybrid SSIM–MAE loss.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

or blank IEADs. By contrast, the hybrid SSIM–MAE consistently outperformed the alternatives, producing lower mean error while preserving distributional structure. Huber and MSE achieved more constrained error distributions (lower variance), but at the expense of structural fidelity. Wasserstein performed worst overall. Figure 13 illustrates a representative case, comparing hPIC2 reference distributions (top row) with surrogate predictions trained under different losses. Models using the hybrid SSIM–MAE loss qualitatively preserved the multi-species features more effectively. Quantitatively, both MAE and L_2 error metrics confirmed the superiority of the hybrid approach.

Species-dependent error behaviour. While figure 12 reports prediction error averaged over all species, figure 14 highlights error distributions for each species. Certain cases (e.g. #58, #67, #76) were consistently more challenging across all species, reflecting higher intrinsic complexity. Errors were generally larger for high-Z species such as oxygen and tungsten, due to their stronger acceleration across the sheath potential. These ions undergo significant adiabatic cooling, producing narrower IEADs that are more sensitive to small deviations, which amplifies error metrics relative to broader distributions.

Overall, the surrogate successfully emulated IEADs for all five species simultaneously, provided that an appropriate architecture and loss function were used. The hybrid SSIM–MAE loss offered the best balance between structural fidelity and low error, while alternative losses performed less consistently. Species with narrow distributions (e.g. tungsten) were more sensitive to prediction deviations, but the surrogate captured their main features reliably.

3.3. *Time gain*

A key advantage of the ML surrogate models lies in their dramatic reduction of computational cost relative to full PIC simulations. Generating a single IEAD with hPIC2 typically requires of the order of 12–24 h of wall-clock time on multiple CPU/GPU nodes, depending on plasma parameters and numerical resolution. In contrast, once trained, the DDeCNN surrogate can reproduce the same IEAD in less than one millisecond on a single GPU or less than a second on a standard CPU workstation. This corresponds to a speedup of approximately six to seven orders of magnitude.

Table 4 shows a summary of the time required for: (i) dataset generation (an indication of total time spent on running PIC simulations); (ii) ML training; and (iii) ML prediction, for each of the databases used in this work. The time needed to generate each dataset can vary dramatically based on the number of data points in the database, numerical parameters (see table 5), number of input parameters and their ranges, and computer architecture used for data generation. Generally speaking, each of the eight databases of table 2 took ∼5 days to be generated using 40 nodes of the UIUC campus cluster (100 000 CPU core-hours). Time required for training also varies dramatically based on the size of data, the complexity of the architecture, number of epochs, etc.; a summary is reported in table 4. Training of the multi-species RF model is slightly more time consuming due to the increased complexity of the architecture used for that database. However, time required to train the DA models is significantly longer, due to the very large size of the database used for that model. Regardless of the specific surrogate architecture, the prediction time per data point remains nearly constant and depends only on the underlying GPU hardware, as illustrated in figure 15.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

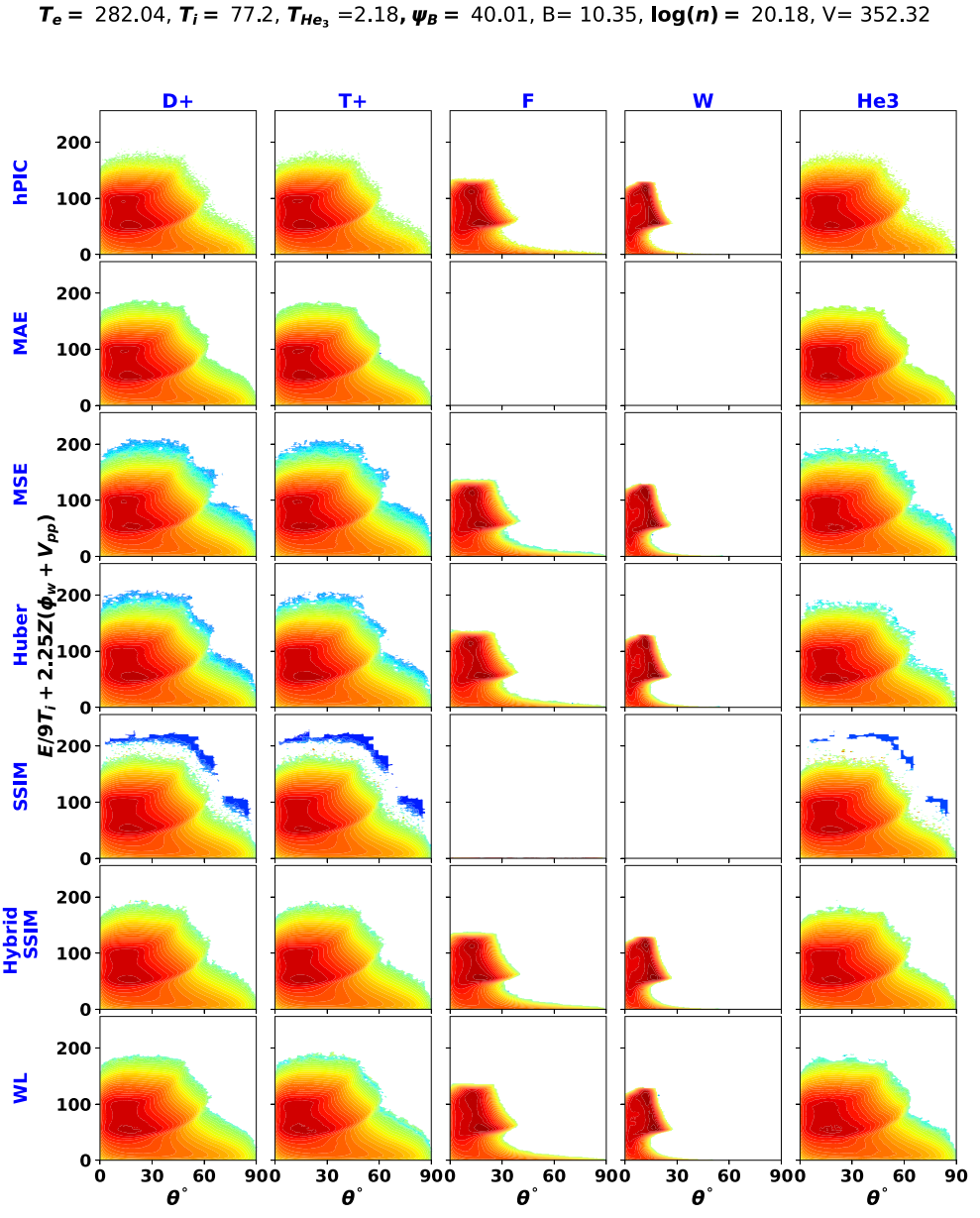


FIGURE 13. Example of IEADs for a five-species plasma, comparing the reference hPIC2 results (top row) with surrogate predictions obtained using different loss functions (subsequent rows), illustrating the predictive performance of each loss function and the overall performance of the ML model for the multi-species RF sheath case. Superior accuracy in multi-species problems is obtained for the hybrid SSIM–MAE loss function.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

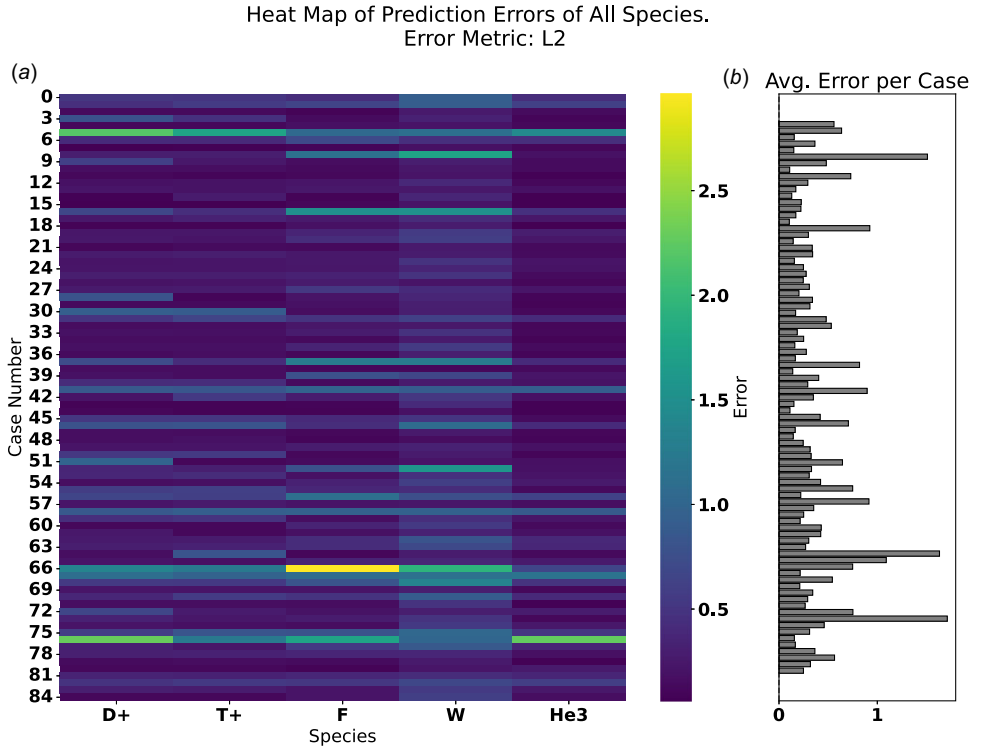


FIGURE 14. (*a*) Heatmap of the L_2 error for each species across all cases in the testing dataset, highlighting species-specific error trends. (*b*) Bar plot of the mean absolute error (MAE) for each case, averaged over all species from the heatmap. Together, these plots show both case-by-case and species-resolved variations in prediction accuracy, emphasising that certain species and cases are intrinsically more challenging to emulate.

To emphasise the impact of this scalability, we considered a realistic application, computing IEADs across a detailed 3-D mesh made of 230 k surface nodes of an Ion Cyclotron Resonance Heating (ICRH) actuator, similar in geometry to the one in use within the WEST tokamak. This example was chosen for the geometrical complexity of the plasma-facing surfaces, due to the presence of RF limiters, Faraday rods, septa and corners within the RF actuator. Using hPIC2, such a calculation would require an estimated ∼1000 days of runtime on 40 nodes of the UIUC Campus Cluster. In contrast, the ML surrogate developed in this work reduces the same task to less than 2 minutes on a single GPU workstation. This result demonstrates the enormous efficiency gain achieved without compromising the fidelity of the original PIC model, enabling practical integration into device-scale PMI studies and design workflows. The implications of this gain are significant. Large-scale PMI studies often require IEADs to be evaluated at thousands of surface locations, under a wide variety of plasma conditions. Using direct PIC simulations for such studies is computationally prohibitive, since generating even 1000 IEADs would require several months of continuous use of a large computing cluster. By contrast, the surrogate can generate the same number of IEADs within minutes,

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

Model			Data generation time [hr case^{-1}]	Training time‡	Prediction time [ms case^{-1}] ¶
RF single-species	Thermal		[2−10]	∼20 [min]	∼0.4
	Basic model			∼20 [min]	∼0.25
	Data-augmented			∼3.5 [hr]	∼0.25
	Multi-fidelity	Step 1: low-fidelity	HF: [6−48]	∼30 [min]	∼0.25
		Step 2: high-fidelity		∼500 [s]	
		Step 3: hyper-tuning		∼100 [s]	
		Total		∼40 [min]	
	Multi-fidelity + Data-augmented	Step 1: low-fidelity	LF: [2−18]	∼3.5 [hr]	∼0.25
		Step 2: high-fidelity		∼2.4 [hr]	
		Step 3: hyper-tuning		∼36 [min]	
		Total		∼6.35 [hr]	
RF multi-species			[6−52]	∼35 [min]	∼0.85

TABLE 4. Summary of the time required to train the ML models discussed in §§ 3.1 and 3.2, including the time required to train each step in the multi-fidelity models. Also showing emulation time for each IEAD following the training of the models.†

†Times listed in the table are based on NVIDIA RTX-1060Ti GPU for both training and prediction.
‡Based on NVIDIA RTX 1060 GPU.
¶For the entire dataset, based on NVIDIA RTX 1060.

Symbol	Meaning	Affected parameters
N_{PPC}	Number of particles per cell	$\boldsymbol{N_{\text{parts}}}$
N_r	Minimum number of Larmor radii in the domain	$\boldsymbol{x}$, $\mathbf{d}\boldsymbol{x}$, N_{cells}, N_{parts}, t, dt
N_{λ_D}	Minimum number of Debye lengths in the domain	$\boldsymbol{x}$, $\mathbf{d}\boldsymbol{x}$, N_{cells}, N_{parts}, t, dt
N_{pp,T_g}	Number of points per gyro-period	$\mathbf{d}\boldsymbol{t}$, $\boldsymbol{N_{\text{steps}}}$
$N_{pp,T_{RF}}$	Number of points per RF cycle	$\mathbf{d}\boldsymbol{t}$, $\boldsymbol{N_{\text{steps}}}$
N_{pp,λ_D}	Number of points per Debye length	$\boldsymbol{N_{cells}}$, N_{parts}
N_{T_i}	Number of ions transient time	$\boldsymbol{t}$, $\boldsymbol{N_{\text{steps}}}$
N_{RF}	Minimum number of RF cycles	$\boldsymbol{N_{\text{steps}}}$, $\boldsymbol{t}$

TABLE 5. User defined parameters that control the constrains. Parameters affected directly are highlighted in bold.

enabling detailed PMI modelling and sensitivity studies that would otherwise be infeasible.

The efficiency gain is further amplified when considering parameter scans and optimisation studies. For example, in designing RF antenna systems for fusion reactors, the RF sheath response must be evaluated across a multi-dimensional parameter space involving density, temperature, magnetic field, RF voltage and geometry. Traditional PIC-based approaches can only afford to sample this space sparsely, whereas the surrogate allows dense sampling and RF-phase-resolved exploration. This makes it possible to perform probabilistic uncertainty quantification, surrogate-based optimisation and real-time coupling of sheath models with system-level codes.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

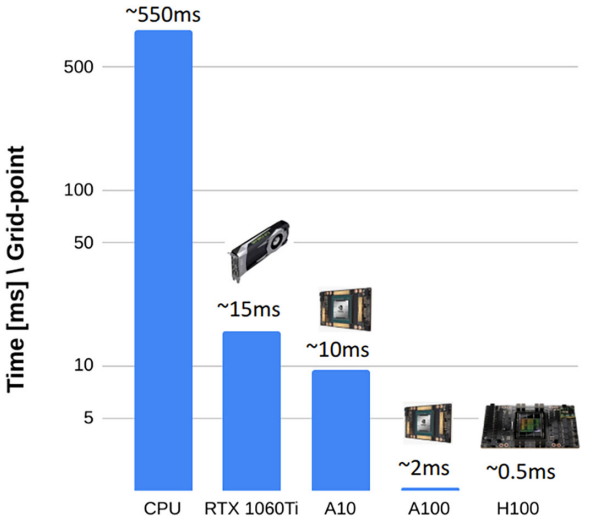


FIGURE 15. Inference time required to emulate multi-species IEADs with the ML surrogate, compared across CPU and several GPU architectures (RTX 1080 Ti, A10, A100 and H100). GPU acceleration reduces prediction time per case from seconds on CPU to milliseconds on modern GPUs, enabling six to seven orders of magnitude speedup relative to full PIC simulations. All time measurements are taken with the GPUs configured to run in double precision.

4. Conclusions

This work has demonstrated the development and application of machine learning surrogates for predicting ion energy–angle distributions (IEADs) in fusion-relevant plasma sheath conditions. Using data generated from the hPIC2 particle-in-cell code, we constructed and trained a deep deconvolutional neural network (DDeCNN) capable of accurately reproducing IEADs for both thermal and RF sheath regimes, including up to 13 input variables and up to five distinct plasma species.

The surrogate achieved high fidelity across a broad parameter space. For thermal sheaths, more than 97 % of test cases were classified as good or average predictions across multiple error metrics, with poor cases (<3 %) occurring only in situations where hPIC2 results were affected by small statistical noise. For RF sheaths, which produce more structured, multi-peak IEADs, prediction accuracy remained robust, with 99 % of single-species and 98 % of multi-species test cases categorised as good or average. We show that the ML surrogate can reliably capture both the phase-space structure and magnitude of RF-driven IEADs.

In our work, we also explored several more advanced techniques, including data augmentation (DA) and multi-fidelity (MF) approaches. These methods enhanced the model's ability to emulate complex distributions and allowed for significant reduction in training data requirements. Despite the limited number of training data points in some cases, the model demonstrated excellent performance, achieving high fidelity in predicting IEADs while maintaining inference times of the order of milliseconds. The DA approach, in particular, proved effective in capturing time-dependent behaviour, necessary for RF sheaths, making it especially promising for future studies involving transient plasma dynamics, such as transients, turbulence, etc.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

Of particular relevance to practical applications, the ML surrogate delivers orders-of-magnitude improvement in computational efficiency. Once trained, the DDeCNN reproduces IEADs in milliseconds on a workstation, compared with hours or days required for PIC simulations on a supercomputing cluster. This corresponds to a speedup of six to seven orders of magnitude, effectively transforming IEAD generation from a computational bottleneck into a routine step that can be seamlessly integrated into larger whole-device frameworks. Such a gain makes it feasible to evaluate IEADs at thousands of wall locations, perform dense parameter scans, and couple sheath predictions directly with other PMI and erosion models. The ML surrogates developed in this study enable the incorporation of ion impact distributions, evaluated with PIC-level accuracy, into multiscale PMI modelling and reactor-relevant design studies.

Acknowledgments

Particle-in-cell simulations were run on the Illinois Campus Cluster, a computing resource operated by the Illinois Campus Cluster Program (ICCP) in conjunction with the National Center for Supercomputing Applications (NCSA) and supported by funds from the University of Illinois Urbana–Champaign (UIUC). Training, validation and testing of the machine learning models were performed on the Illinois Campus Cluster and using computing resources at the Plasma Science and Fusion Center (PSFC) at the Massachusetts Institute of Technology (MIT), Boston, Massachusetts. Proofreading of the final version of the manuscript was performed using GPT-5.0.

Editor Luís O. Silva thanks the referees for their advice in evaluating this article.

Funding

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Fusion Energy Sciences, under award number DE-SC0024369 (SciDAC Center for Advanced Simulation of RF-Plasma-Material Interactions). Additional support was provided under award DE-SC0025853 (FIRE Collaborative: APP-FPP Advanced Profile Prediction for Fusion Pilot Plant Design). Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. Department of Energy.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Quantitative constraints on numerical parameters

After generating the training and testing datasets through structured sampling of the free parameter space, each hPIC2 simulation configuration is evaluated using a set of logical conditions designed to enforce constraints on the numerical parameters. These constraints ensure that the simulations are sufficiently well resolved. The user-defined parameters that control the strictness of these constraints are listed in table 5. The specific numerical requirements are discussed as follows.

The size of the domain x and element size dx are computed based on the Debye length of the multi-species plasma λ_D given by (A1) and the maximum value of Larmor radii r^j of all ion species given by (A2), where ϵ_0 is the permittivity of free

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

space; K_B is Boltzmann constant; e is the elementary charge; n_e is plasma density; T_e is electron temperature; B is the magnetic field strength; and Z^j, n^j and T_i^j are respectively the charge, density and temperature of the jth ion species. Here, $v_\perp^j$ is the speed perpendicular to the magnetic field, which is approximated as the most probable thermal speed of the jth ion species given by $v_\perp^j = \sqrt{2T_i^j/m^j}$. Therefore, dx is always selected as one Debye length λ_D and the domain size x is the largest of 500 λ_D or 3 times the largest value of r_L.

$$\lambda_D = \left(\frac{\epsilon_0 K_B / e^2}{n_e / T_e + \sum_j Z^j n^j T_i^j} \right)^{1/2}, \tag{A1}$$

$$r_{\max} = \max(r^j) = \max\left(\frac{m^j v_\perp^j}{Z^j e B} \right). \tag{A2}$$

Using (A1) and (A2), and the user defined parameters in table 5, the domain size x is computed as follows:

$$x = \max\left[N_{\lambda_D} \times \lambda_D, 3 \times \max\left(r^j\right) \right], \tag{A3}$$

and the element size is calculated using the value of points per Debye length $N_{\mathrm{pp},\lambda_D}$. In the case where $N_{\lambda_D} \times \lambda_D \leqslant 3 \times \max(r^j)$, $N_{\mathrm{pp},\lambda_D} = N_{\lambda_D}$ or

$$\mathrm{d}x = N_{\mathrm{pp},\lambda_D}(x/\lambda_D). \tag{A4}$$

Four constraints are enforced on the time step $\mathrm{d}t$. The constraints are denoted as, thermal time step $\mathrm{d}t_{\mathrm{th}}$, gyro time step $\mathrm{d}t_g$, RF sheath ion transient time $\mathrm{d}t_{\mathrm{RF}}$ and RF period sampling frequency $\mathrm{d}t_T$. The degree at which the constraints are enforced is controlled by the user-defined parameters as shown in table 5.

Following the determination of $\mathrm{d}x$, $\mathrm{d}t_{\mathrm{th}}$ is computed as follows:

$$\mathrm{d}t_{\mathrm{th}} = \frac{\mathrm{d}x}{3v_{\mathrm{th}}}. \tag{A5}$$

The frequency of gyro-period sampling is defined using N_{pp,T_g}. Then, $\mathrm{d}t_g^j$ is then computed using (A6), such that $\tau_g = 2\pi/\omega_c$ is the gyro-period and ω_c is the cyclotron frequency of the jth species given by (A7),

$$\mathrm{d}t_g^j = \frac{\tau_g}{N_{\mathrm{pp},T_g}}, \tag{A6}$$

$$\omega_c^j = \frac{Z^j e B}{m_i^j}. \tag{A7}$$

RF sheath ion transient time $\mathrm{d}t_{\mathrm{RF}}$ is computed according to (A8), where a^j shown in (A9) is the accelerating of the jth ion species given the peak value of the RF sheath, E_s is the maximum RF electrical field shown in (A10) and L is the estimated length of the RF sheath. The latter is estimated as $L = 10\lambda_D$.

$$\mathrm{d}t_{\mathrm{RF}}^j = \sqrt{\frac{\mathrm{d}x}{2a^j}}, \tag{A8}$$

$$a^j = \frac{eZ^j E_s}{m_i^j}, \tag{A9}$$

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

$$E_s = 2V_{\mathrm{pp}}/L. \tag{A10}$$

RF period sampling frequency $\mathrm{d}t_T$ is computed as follows:

$$\mathrm{d}t_T = \frac{\tau_{\mathrm{RF}}}{N_{\mathrm{pp},T_{\mathrm{RF}}}}. \tag{A11}$$

Finally, $\mathrm{d}t$ to be used in each simulations is computed according to the following equation:

$$\mathrm{d}t = \min\left[\mathrm{d}t_{\mathrm{th}},\ \min\left(\mathrm{d}t_g^j\right),\ \min\left(\mathrm{d}t_{\mathrm{RF}}^j\right),\ \mathrm{d}t_T\right]. \tag{A12}$$

Once $\mathrm{d}t$ is computed via (A12), the total simulation time t can be computed according to another three restrictions; the total number of ion transient time N_{T_i}, total number of RF cycles N_{RF} and a predefined minimum number of time steps.

The ion transit time is computed using (A13), such that ψ_B is the magnetic field inclination angle with respect to the surface normal,

$$T_i = \frac{x}{v_{\mathrm{th}} \cos\psi_B}. \tag{A13}$$

Then, the total simulation time is computed according to the following equation:

$$t = \max\left[N_{T_i} T_i,\ N_{\mathrm{RF}}/f_{\mathrm{RF}},\ N_{\mathrm{steps}}\right] \tag{A14}$$

and the number of time steps is simply

$$N_{\mathrm{steps}} = \mathrm{int}(t/\mathrm{d}t). \tag{A15}$$

A.1. *Normalisation applied on the IEADs*

To use DDeCNN architecture, the IEADs generated using hPIC2 simulations should satisfy the following requirements.

(i) The number of elements in the energy and angle dimensions should be a power of 2.

(ii) The maximum energy bin should be chosen such that all the particles accumulated during the simulation have energy below that value.

(iii) The number of elements should be fixed for all simulations.

To satisfy requirements (i) and (iii), the number of bins in the angle axis was set to 128 bins with range [0,90] degrees. However, the number of energy bins was fixed to 257 bins, where the maximum energy bin acts as a sink and will be removed during the post-processing. It should be noted that the number of bins can be changed during the data generation stage if more resolution is needed. However, as the number of bins increases, the training and sampling computational cost of the ML model increases.

While requirements (i) and (iii) should be fixed for all species, this is not the case for requirement (ii). To satisfy requirement (ii), the maxima energy bin for the jth species in a given input vector is computed according to (A17). The first term of (A17) accounts for the thermal distribution of the ions' source, with an average energy distribution of $\frac{3}{2}T_i$, the latter is multiplied by 3 to account for the high energy portion of the distribution, and then multiplied by 2 as a factor of safety. The last

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

Data type	Error type	Mean	Standard-deviation	Max
Testing data	MAE	0.0224	0.0118	0.0562
	MSE	2.52×10^{-3}	2.71×10^{-3}	0.0107
	RL_2	0.2539	0.1989	0.8423
Training data	MAE	0.0182	0.0110	0.0657
	MSE	1.58×10^{-3}	1.79×10^{-3}	0.0127
	RL_2	0.1764	0.6062	0.7569

TABLE 6. Statistics of reconstruction errors for the ML model trained on 4-D thermal sheath data, detailing the reconstruction errors of IEAD compared with the original data for both testing and training dataset.

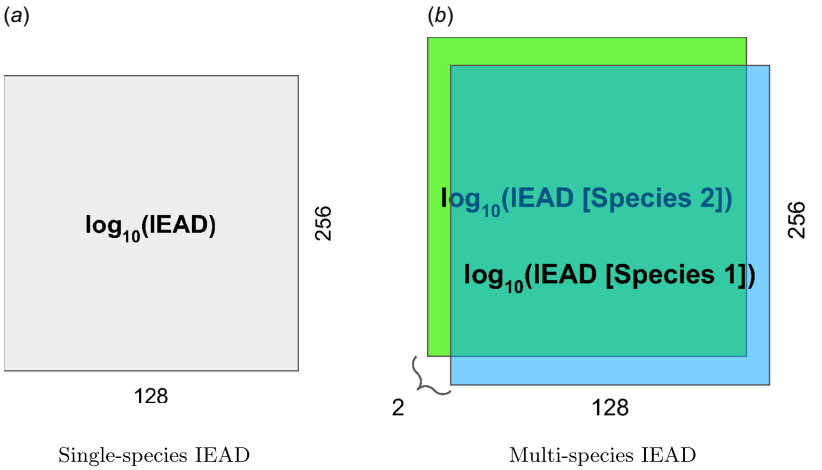


FIGURE 16. Data preparation of the ML model, showing the procedure for (*a*) single-species and (*b*) multi-species.

term of (A17) accounts for the energy gained by an ion located at the centre of the domain as it is accelerated by the RF E-field at its peak voltage V_{pp}, with a value estimated as $2Z^j V_{\mathrm{pp}}$ and multiplied by a safety factor of 0.25.

$$\frac{e\phi_w}{K_B T_e} = -\frac{1}{2}\ln\left(\frac{m_i/m_e}{2\pi(1+T_i/T_e)}\right), \tag{A16}$$

$$E^j_{\mathrm{max}} = 9T_i^j + 2.25Z^j\left(\phi_w^j + V_{\mathrm{pp}}\right). \tag{A17}$$

The middle term of (A17) given by $(2.25Z^j\phi_w^j)$ accounts for the total energy gained by an ion located in the middle of the domain as it is accelerated by the floating wall potential ϕ_w of the thermal plasma sheath given by (A16). A factor of 1.7 was added to account for the plasma pre-sheath. For simplicity, the factor 2.25 was used instead.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

Appendix B. Statistics of prediction errors for the ML model

Dataset	Error metric	Model	Mean	Standard-deviation	Max
Test	MAE	Basic model	1.79×10^{-2}	1.35×10^{-2}	8.45×10^{-2}
		MF	1.08×10^{-2}	7.54×10^{-3}	3.86×10^{-2}
		DA (last step)	1.20×10^{-2}	8.59×10^{-3}	3.91×10^{-2}
		DA + MF (last step)	8.73×10^{-3}	4.88×10^{-3}	2.64×10^{-2}
		DA	1.17×10^{-2}	8.92×10^{-3}	6.58×10^{-2}
		DA + MF	8.48×10^{-3}	4.86×10^{-3}	4.29×10^{-2}
	MSE	Basic model	2.67×10^{-3}	3.49×10^{-3}	2.36×10^{-2}
		MF	1.11×10^{-3}	1.30×10^{-3}	6.54×10^{-3}
		DA (last step)	1.39×10^{-3}	1.76×10^{-3}	9.04×10^{-3}
		DA + MF (last step)	7.29×10^{-4}	7.48×10^{-4}	4.24×10^{-3}
		DA	1.37×10^{-3}	2.08×10^{-3}	2.32×10^{-2}
		DA + MF	7.01×10^{-4}	8.02×10^{-4}	8.57×10^{-3}
	L_2	Basic model	3.04×10^{-1}	2.40×10^{-1}	1.86×10^{0}
		MF	2.00×10^{-1}	1.67×10^{-1}	1.34×10^{0}
		DA (last step)	2.19×10^{-1}	2.07×10^{-1}	1.76×10^{0}
		DA + MF (last step)	1.68×10^{-1}	1.33×10^{-1}	9.60×10^{-1}
		DA	2.12×10^{-1}	1.93×10^{-1}	1.76×10^{0}
		DA + MF	1.62×10^{-1}	1.23×10^{-1}	1.14×10^{0}
Train	MAE	Basic model	1.13×10^{-2}	8.09×10^{-3}	9.64×10^{-2}
		MF	6.55×10^{-3}	5.49×10^{-3}	9.86×10^{-2}
		DA (last step)	4.87×10^{-3}	3.62×10^{-3}	3.78×10^{-2}
		DA + MF (last step)	4.55×10^{-3}	3.40×10^{-3}	3.77×10^{-2}
		DA	4.59×10^{-3}	3.59×10^{-3}	5.40×10^{-2}
		DA + MF	4.34×10^{-3}	3.52×10^{-3}	5.85×10^{-2}
	MSE	Basic model	1.14×10^{-3}	1.78×10^{-3}	4.06×10^{-2}
		MF	4.62×10^{-4}	1.45×10^{-3}	4.44×10^{-2}
		DA (last step)	2.18×10^{-4}	3.07×10^{-4}	5.38×10^{-3}
		DA + MF (last step)	1.94×10^{-4}	2.91×10^{-4}	6.14×10^{-3}
		DA	1.98×10^{-4}	3.27×10^{-4}	1.17×10^{-2}
		DA + MF	1.82×10^{-4}	3.31×10^{-4}	1.26×10^{-2}
	L2	Basic model	2.67×10^{-1}	1.27×10^{0}	3.64×10^{1}
		MF	1.97×10^{-1}	1.38×10^{0}	3.81×10^{1}
		DA (last step)	8.43×10^{-2}	5.04×10^{-2}	7.14×10^{-1}
		DA + MF (last step)	7.95×10^{-2}	4.61×10^{-2}	6.34×10^{-1}
		DA	7.83×10^{-2}	4.60×10^{-2}	9.52×10^{-1}
		DA + MF	7.48×10^{-2}	4.63×10^{-2}	9.22×10^{-1}

TABLE 7. Statistics of reconstruction errors for the ML model trained on the single-species RF sheath dataset. The table reports reconstruction errors of the IEAD compared with the original data for both the training and testing datasets. It includes results for different model variants: the baseline model, the multi-fidelity (MF) model, the data-augmented (DA) model and the combined MF + DA model. The 'last step' variant shows the reconstruction error when using only the original data (without augmentation).

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

Appendix C. ML models hyper-parameters details

Hyper-parameters	Selected value
No. of hidden layers in the MLP part	2
Dimensions of hidden layers in the MLP part	128, 8192
No. of epochs	500
Batch size	64
Stride size (transposed convolutional layers)	2×2
Kernel size (transposed convolutional layers)	4×4
Padding size (transposed convolutional layers)	Same
Stride size (convolutional layers)	1×1
Kernel size (convolutional layers)	4×4
Padding size (convolutional layers)	Same
Learning rate	0.0005
Optimiser	Adam [†]

TABLE 8. Summary of hyper-parameters used to train the ML model on the single-species, thermal sheath dataset.

[†]Kingma & Ba (2014).

Appendix D. Additional results

Hyper-parameters	Training step/selected value			
	LF (+DA)	HF (+DA)	MF (+DA)	Baseline (+DA)
No. of epochs	350 (150)	250 (100)	50 (25)	600 (275)
Batch size	256	128	128	128
Learning rate	5×10^{-4}	5×10^{-4}	5×10^{-5}	5×10^{-4}
Layers to freeze/unfreeze		[1−3]		N/A
No. of hidden layers in the MLP part		2		
Dimensions of hidden layers in the MLP part		128, 16,384		
Stride size (transposed convolutional layers)		2×2		
Kernel size (transposed convolutional layers)		4×4		
Padding size (transposed convolutional layers)		Same		
Stride size (convolutional layers)		1×1		
Kernel size (convolutional layers)		4×4		
Padding size (convolutional layers)		Same		
Optimiser		Adam[‡]		

TABLE 9. Summary of hyper-parameters used to train the multi-fidelity and the data-augmented (DA) models on the single-species, RF sheath dataset. Values inside brackets show the value when DA model is included.

[‡]Kingma & Ba (2014).

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

Hyper-parameters	Selected value
No. of hidden layers in the MLP part	2
Dimensions of hidden layers in the MLP part	128, 16384
No. of epochs	1500
Batch size	64
Stride size (transposed convolutional layers)	2×2
Kernel size (transposed convolutional layers)	4×4
Padding size (transposed convolutional layers)	Same
Stride size (convolutional layers)	2×2
Kernel size (convolutional layers)	4×4
Padding size (convolutional layers)	Same
Learning rate	0.0005
Optimiser	Adam

TABLE 10. Summary of hyper-parameters used to train the multi-species RF sheath dataset.

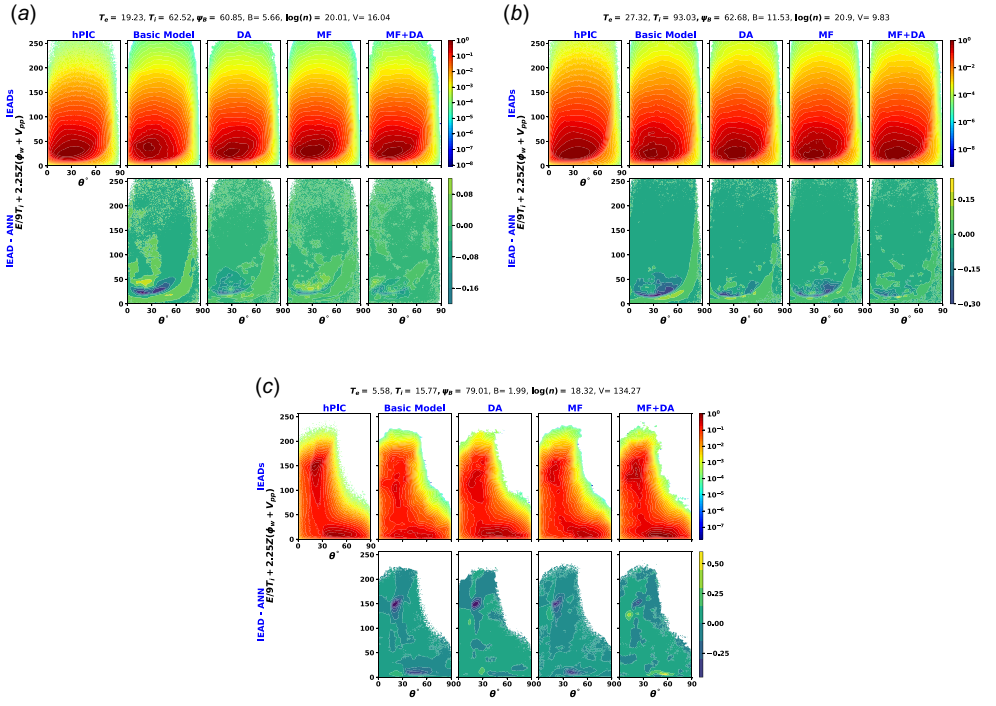


FIGURE 17. Examples of three representative IEAD predictions for a single-species RF sheath. (*a*) Good illustrates excellent agreement. (*b*) Average demonstrates moderate deviations while maintaining the overall distribution shape. (*c*) Poor corresponds to a poor prediction. In each example, the first row compares the reference distribution from hPIC2 (left) with the surrogate predictions from the baseline, data-augmented (DA), multi-fidelity (MF) and combined DA+MF models. The second row shows the corresponding residual error distributions for each model.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

REFERENCES

ABADI, M. *et al.* 2015 TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.

AHEDO, E. 1997 Structure of the plasma-wall interaction in an oblique magnetic field. *Phys. Plasmas* **4**, 4419–4430.

ALZUBAIDI, L., ZHANG, J., HUMAIDI, A.J., AL-DUJAILI, A., DUAN, Y., AL-SHAMMA, O., SANTAMARÍA, J., FADHEL, M.A., AL-AMIDIE, M. & FARHAN, L. 2021 Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. *J. Big Data* **8**, 53.

BISHOP, C.M. 1995 *Neural Networks for Pattern Recognition*. Oxford University Press.

BOHM, D. 1949 *The Characteristics of Electrical Discharges in Magnetic Fields*. McGraw-Hill.

BUTLER, H.S. & KINO, G.S. 1963 Plasma sheath formation by radio-frequency fields. *Phys. Fluids* **6**, 1346–1355.

CHANG, C.S., KU, S., HAGER, R., CHURCHILL, R.M., HUGHES, J., KÖCHL, F., LOARTE, A., PARAIL, V. & PITTS, R.A. 2021 Constructing a new predictive scaling formula for iter's divertor heat-load width informed by a simulation-anchored machine learning. *Phys. Plasmas* **28**, 022501.

CHEN, F.C. 2016 *Introduction to Plasma Physics and Controlled Fusion*. Springer International Publishing.

CHODURA, R. 1982 Plasma–wall transition in an oblique magnetic field. *Phys. Fluids* **25**, 1628–1633.

CHOLLET, F. *et al.* 2015 Keras. Available at: https://keras.io.

CIRAOLO, G. *et al.* 2019 First modeling of strongly radiating west plasmas with soledge-eirene. *Nucl. Mater. Energy* **20**, 100685.

DEVAUX, S. & MANFREDI, G. 2006 Vlasov simulations of plasma-wall interactions in a magnetized and weakly collisional plasma. *Phys. Plasmas* **13**, 083504.

DEWALD, A.B., BAILEY, A.W. & BROOKS, J.N. 1987 Trajectories of charged particles traversing a plasma sheath in an oblique magnetic field. *Phys. Fluids* **30**, 267–269.

DONG, G., WEI, X., BAO, J., BROCHARD, G., LIN, Z. & TANG, W. 2021 Deep learning based surrogate models for first-principles global simulations of fusion plasmas. *Nucl. Fusion* **61**, 126061.

ELIAS, M., CURRELI, D., JENKINS, T.G., MYRA, J.R. & WRIGHT, J. 2019 Numerical model of the radio-frequency magnetic presheath including wall impurities. *Phys. Plasmas* **26**, 092508.

ELIAS, M., CURRELI, D. & MYRA, J.R. 2021 Energy-angle distribution of the ions in the rf sheath of icrh antennas. *Phys. Plasmas* **28**, 052106.

FEURER, M. & HUTTER, F. 2019 Hyperparameter optimization. In *Automated Machine Learning: Methods, Systems, Challenges* (ed. F. Hutter, L. Kotthoff & J. Vanschoren), pp. 3–33. Springer International Publishing.

FROGNER, C., ZHANG, C., MOBAHI, H., ARAYA, M. & POGGIO, T.A. 2015 Learning with a wasserstein loss. In *Advances in Neural Information Processing Systems*, vol. 28.

FU, Y., ELDON, D., ERICKSON, K., KLEIJWEGT, K., LUPIN-JIMENEZ, L., BOYER, M.D., EIDIETIS, N., BARBOUR, N., IZACARD, O. & KOLEMEN, E. 2020 Machine learning control for disruption and tearing mode avoidance. *Phys. Plasmas* **27**, 022501.

FUKUSHIMA, K. 1969 Visual feature extraction by a multilayered network of analog threshold elements. *IEEE Transfer Syst. Sci. Cyb.* **5**, 322–333.

IOFFE, S. & SZEGEDY, C. 2015 Batch normalization: accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pp. 448–456. PMLR.

KATES-HARBECK, J., SVYATKOVSKIY, A. & TANG, W. 2019 Predicting disruptive instabilities in controlled fusion plasmas through deep learning. *Nature* **568**, 526–531.

KEILHACKER, M., LACKNER, K., BEHRINGER, K., MURMANN, H. & NIEDERMEYER, H. 1982 Plasma boundary layer in limiter and divertor tokamaks. *Phys. Scr.* **1982**, 443.

KHAZIEV, R. & CURRELI, D. 2015 Ion energy-angle distribution functions at the plasma-material interface in oblique magnetic fields. *Phys. Plasmas* **22**, 043503.

KHAZIEV, R. & CURRELI, D. 2018 hpic: a scalable electrostatic particle-in-cell for plasma–material interactions. *Comput. Phys. Commun.* **229**, 87–98.

KINGMA, D.P. & BA, J. 2014 Adam: a method for stochastic optimization. arXiv preprint arXiv: 1412.6980.

KINO, H., IKUSE, K., DAM, H.-C. & HAMAGUCHI, S. 2021 Characterization of descriptors in machine learning for data-based sputtering yield prediction. *Phys. Plasmas* **28**, 013504.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

LAO, L.L. *et al.* 2022 Application of machine learning and artificial intelligence to extend efit equilibrium reconstruction. *Plasma Phys. Control. Fusion* **64**, 074001.

LECUN, Y., BOSER, B., DENKER, J.S., HENDERSON, D., HOWARD, R.E., HUBBARD, W. & JACKEL, L.D. 1989 Handwritten digit recognition with a back-propagation network. In *Advances in Neural Information Processing Systems*, vol. 2, pp. 396–404.

LECUN, Y., BOTTOU, L., BENGIO, Y. & HAFFNER, P. 2002 Gradient-based learning applied to document recognition. *Proc. IEEE* **86**, 2278–2324.

LIEBERMAN, M.A. & LICHTENBERG, A.J. 2005 *Principles of Plasma Discharges and Materials Processing*. John Wiley & Sons.

LIN, Z., MAUREL-OUJIA, T., KADOCH, B., KRAH, P., SAURA, N., BENKADDA, S. & SCHNEIDER, K. 2024 Synthesizing impurity clustering in the edge plasma of tokamaks using neural networks. *Phys. Plasmas* **31**, 032505.

LLOYD, S.P. 1982 Least squares quantization in pcm. *IEEE Transfer Inf. Theory* **28**, 129–136.

MCKAY, M.D., BECKMAN, R.J. & CONOVER, W.J. 1979 A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* **21**, 239–245.

MCKAY, M.D., CONOVER, W.J. & WHITEMAN, D.E. 1976 Report on the application of statistical techniques to the analysis of computer codes. *Tech. Rep.* LA-6479-MS. Los Alamos Scientific Lab., N.Mex. (USA).

MEREDITH, L.T., REZAZADEH, M., HUQ, M.F., DROBNY, J., SRINIVASARAGAVAN, V.V., SAHNI, O. & CURRELI, D. 2023 hpic2: a hardware-accelerated, hybrid particle-in-cell code for dynamic plasma-material interactions. *Comput. Phys. Commun.* **283**, 108569.

MESEGUER-BROCAL, G. & PEETERS, G. 2019 Conditioned-u-net: introducing a control mechanism in the u-net for multiple source separations. arXiv preprint arXiv: 1907.01277.

MOSEEV, D. & SALEWSKI, M. 2019 Bi-maxwellian, slowing-down, and ring velocity distributions of fast ions in magnetized plasmas. *Phys. Plasmas* **26**, 020901.

MUSTAFA, M.A.-J.M. 2023 Development of reduced-order models of the ion impact distribution function in magnetized plasma sheaths. PhD thesis, University of Illinois at Urbana–Champaign.

MYRA, J.R., ELIAS, M.T., CURRELI, D. & JENKINS, T.G. 2020 Effect of net direct current on the properties of radio frequency sheaths: simulation and cross-code comparison. *Nucl. Fusion* **61**, 016030.

MYRA, J.R. 2021 A tutorial on radio frequency sheath physics for magnetically confined fusion devices. *J. Plasma Phys.* **87**, 905870504.

PARSONS, M.S. 2017 Interpretation of machine-learning-based disruption models for plasma control. *Plasma Phys. Control. Fusion* **59**, 085001.

PATRO, S.G. & SAHU, K.K. 2015 Normalization: a preprocessing stage. arXiv preprint arXiv: 1503.06462.

PEDREGOSA, F. *et al.* 2011 Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830.

PICCIONE, A., BERKERY, J.W., SABBAGH, S.A. & ANDREOPOULOS, Y. 2020 Physics-guided machine learning approaches to predict the ideal stability properties of fusion plasmas. *Nucl. Fusion* **60**, 046033.

PITTS, R.A. *et al.* 2019 Physics basis for the first iter tungsten divertor. *Nucl. Mater. Energy* **20**, 100696.

PORTER, G.D., PETRIE, T.W., ROGNLIEN, T.D. & RENSINK, M.E. 2010 Uedge simulation of edge plasmas in diii-d double null configurations. *Phys. Plasmas* **17**, 112501.

REZAZADEH, M., MYRA, J.R. & CURRELI, D. 2023 Resonance in radio frequency sheath admittance and enhanced impurity emission near the ion cyclotron frequency. *Nucl. Fusion* **63**, 126024.

RIEMANN, K.-U. 1991 The bohm criterion and sheath formation. *J. Phys. D: Appl. Phys.* **24**, 493.

RIEMANN, K.-U. 1994 Theory of the collisional presheath in an oblique magnetic field. *Phys. Plasmas* **1**, 552–558.

RUMELHART, D.E., HINTON, G.E. & WILLIAMS, R.J. 1986 Learning representations by back-propagating errors. *Nature* **323**, 533–536.

SANCHEZ-VILLAR, A., BAI, Z., BERTELLI, N., BETHEL, E.W., HILLAIRET, J., PERCIANO, T., SHIRAIWA, S., WALLACE, G.M. & WRIGHT, J.C. 2025 Machine learning enhanced predictions of icrf heating: overcoming numerical limitations via data curation. *Phys. Plasmas* **32**, 062504.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press

SÁNCHEZ-VILLAR, Á., BAI, Z., BERTELLI, N., BETHEL, E.W., HILLAIRET, J., PERCIANO, T., SHIRAIWA, S., WALLACE, G.M. & WRIGHT, J.C. 2024 Real-time capable modeling of icrf heating on nstx and west via machine learning approaches. *Nucl. Fusion* **64**, 096039.

SELESON, P. MUSTAFA, M., CURRELI, D., HAUCK, C.D., STOYANOV, M. & BERNHOLDT, D.E. 2022 Data-driven surrogate modeling of hpic ion energy-angle distributions for high-dimensional sensitivity analysis of plasma parameters' uncertainty. *Comput. Phys. Commun.* **279**, 108436.

SOHL-DICKSTEIN, J., WEISS, E., MAHESWARANATHAN, N. & GANGULI, S. 2015 Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pp. 2256–2265. PMLR.

STANGEBY, P.C., *et al.* 2000 *The Plasma Boundary of Magnetic Fusion Devices*, vol. 224. Institute of Physics Publishing.

TONKS, L. & LANGMUIR, I. 1929 A general theory of the plasma of an arc. *Phys. Rev.* **34**, 876–922.

VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A.N., KAISER, L. & POLOSUKHIN, I. 2023 Attention is all you need. arXiv: 1706.03762.

WALLACE, G.M., BAI, Z., SADRE, R., PERCIANO, T., BERTELLI, N., SHIRAIWA, S., BETHEL, E.W. & WRIGHT, J.C. 2022 Towards fast and accurate predictions of radio frequency power deposition and current profile via data-driven modelling: applications to lower hybrid current drive. *J. Plasma Phys.* **88**, 895880401.

WESSON, J. & CAMPBELL, D.J. 2011, *Tokamaks*, vol. 149. Oxford University Press.

WU, K., PENG, H., CHEN, M., FU, J. & CHAO, H. 2021Rethinking and improving relative position encoding for vision transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10033–10041.

YAMAMURA, Y. & SHINDO, S. 1984 An empirical formula for angular dependence of sputtering yields. *Radiat. Effects* **80**, 57–72.

YAMAMURA, Y. & TAWARA, H. 1996 Energy dependence of ion-induced sputtering yields from monatomic solids at normal incidence. *Atom. Data Nucl. Data* **62**, 149–253.

https://doi.org/10.1017/S0022377826101561 Published online by Cambridge University Press