PAPER • OPEN ACCESS

Efficient data acquisition and training of collisional-radiative model artificial neural network surrogates through adaptive parameter space sampling

To cite this article: Nathan A Garland *et al* 2022 *Mach. Learn.: Sci. Technol.* **3** 045003

View the article online for updates and enhancements.

MACHINE LEARNING
Science and Technology

PAPER

Efficient data acquisition and training of collisional-radiative model artificial neural network surrogates through adaptive parameter space sampling

Nathan A Garland[1,2,3,5,*] ID, Romit Maulik[4,5] ID, Qi Tang[1] ID, Xian-Zhu Tang[1] and Prasanna Balaprakash[4,5]

1   Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, United States of America
2   Faculty of Science and Engineering, Southern Cross University, Lismore, NSW 2480, Australia
3   School of Environment and Science, Griffith University, Nathan, QLD 4111, Australia
4   Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL 60439, United States of America
5   Argonne Leadership Computing Facility, Argonne National Laboratory, Lemont, IL 60439, United States of America
*   Author to whom any correspondence should be addressed.

E-mail: n.garland@griffith.edu.au
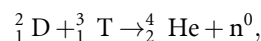
Abstract

Effective plasma transport modeling of magnetically confined fusion devices relies on having an accurate understanding of the ion composition and radiative power losses of the plasma. Generally, these quantities can be obtained from solutions of a collisional-radiative (CR) model at each time step within a plasma transport simulation. However, even compact, approximate CR models can be computationally onerous to evaluate, and in-situ evaluation of these models within a larger plasma transport code can lead to a rigid bottleneck. As a way to bypass this bottleneck, we propose deploying artificial neural network (ANN) surrogates to allow rapid evaluation of the necessary plasma quantities. However, one issue with training an accurate ANN surrogate is the reliance on a sufficiently large and representative training and validation data set, which can be time-consuming to generate. In this work we explore a data-driven active learning and training routine to allow autonomous adaptive sampling of the problem parameter space to ensure a sufficiently large and meaningful set of training data is assembled for the network training. As a result, we can demonstrate approximately order-of-magnitude savings in required training data samples to produce an accurate surrogate.

1. Introduction

The construction of the International Thermonuclear Experimental Reactor (ITER), and its components, is now well underway and being performed over the world, with the final assembly and operation to take place in Southern France. ITER will be the world's largest tokamak, which is a device to magnetically confine a hot plasma in the shape of a torus with the goal of demonstrating a path to viable fusion energy [1, 2]. With anticipated experiments to begin in the mid 2020s, there is currently a significant worldwide research effort taking place to better understand how to efficiently, and safely, operate ITER. Modeling and simulations are a major tool being used to perform these studies, with many plasma simulation codes coupling multiple modules to better describe the wide array of physics phenomena taking place within the plasmas being considered [3–7]. One key component in the plasma simulation cycle is to accurately describe the collisional-radiative (CR) balance of the plasma, in order to determine impurity ion populations, and the radiation being emitted by the excited ion populations within the plasma at a given time and location [8–11].

This information is crucial to accurate modeling of a device like ITER given that impurities, other than the hydrogen-like fuel (H, D, T), will be an inevitable presence in the plasma. Situations of interest include

alpha particles (helium nuclei, that may recombine to He^+ or atomic He) produced via the D-T fusion reaction [1]

$$_1^2 D +_1^3 T \rightarrow_2^4 He + n^0,$$

containment vessel impurities, such as beryllium or tungsten sputtering into the plasma [12], or purposefully injected impurities, such as nitrogen [13, 14] or neon [7, 15], to facilitate radiative cooling of regions of the plasma to control or terminate a plasma following a disruption event [15].

Presently, a major threat to ITER's safe operation are plasma disruptions [15–17]. A disruption is a sudden loss of plasma control and stability, and can lead to extreme heat load deposition to the reactor walls, excessive electromagnetic forces exerted on the reactor structure, and/or relativistic electron beam formation that could penetrate through reactor walls [16, 17]. In either case, the end result is predicted to be unacceptable hardware damage, costing exorbitant time and financial resources. Presently, impurity injection is the reference ITER mitigation strategy, with neon being a likely candidate for injection into the hydrogenic plasma [17].

Thus, given the motivation for modeling and simulating complex fusion plasma discharges, it is then paramount to have accurate ion population and radiative power dissipation quantities when simulating the effects of impurities in the plasma. A general approach to determining these quantities is through CR modeling, which will be further discussed in section 2. While solution of a CR model will provide accurate plasma properties as input to subsequent coupled stages of a plasma simulation code, the computational cost of CR models can be onerous, even when considering relatively simplified models.

Relatively compact approximate CR models, with state vectors $\mathcal{O}(10^1–10^2)$ such as that used in this present work [9, 18], can take up to dozens of seconds to solve. While state-of-the-art, fine structure $\mathcal{O}(10^6–10^9)$ CR models can take minutes or hours to obtain a solution [11, 19]. Presently, in order to obtain these quantities within plasma transport codes the fusion community has resorted to simplified models, in so called coronal equilibrium or coronal-like limits, that can limit the fidelity of physics retained in the model [11, 20–22].

In this study we seek an alternative approach to allow rapid evaluation of necessary CR quantities in the form of an artificial neural network (ANN) surrogate, without sacrificing the fidelity of the physics built into the CR model, or severely limiting the domain of reasonable input parameters that may be required for model evaluation. Once trained, given the typical rapid evaluation time of an ANN surrogate, we hope that coupling a well-trained, accurate ANN surrogate within a plasma simulation framework, in lieu of a CR model solve, will allow accurate atomic and plasma physics information to be passed to the simulation platform, without adding the computational burden of numerically solving an unnecessary large non-linear problem at each time step or iteration.

It is instructive to note that artificial intelligence methods applied to plasma physics problems has a rich history, dating back to the wave of neural network research in the 1990s [23–25], through to more recent applications [26–30]. With focus on specific application of ANN towards CR properties as discussed in this study, pioneering data-driven methods in plasma spectroscopy synthesis can be found from Goldstein, Larsen, Morgan *et al* in the early 1990s [31–33]. Notably, this series of studies demonstrated success in applying relatively simple ANN techniques with limited computational resources available, when compared to the present day, to streamline the use of fundamental atomic and molecular physics models into a larger plasma physics workflow. More recently, ANN(s) have been shown to be beneficial in learning surrogates for non-LTE radiation models in order to aide analysis and direct interpretation of experimental spectroscopy measurements [34, 35]. Lastly, the recent work of Kluth *et al* [36] is a notable demonstration of the utility of employing ANN surrogates of CR models within plasma simulation codes to provide accurate physics quantities at a fraction of the computational cost of the forward-pass CR model. Generally, previous applications of ANN surrogates towards the CR problem have not focused on the problem of optimal data set generation and the associated time required for this. In an effort to address this, we now introduce and discuss a novel adaptive data generation and training framework that demonstrates approximate order of magnitude time savings in data acquisition and training, but also provides physics assurances by systematically sampling representative and meaningful training data.

A standard approach to training an ANN in deep learning is to essentially assemble a large set of data for training and validation purposes. This generally enables robust learning of the potential mappings between input variables and output variables. If forward evaluations of the original CR model are quite fast, this is not a terribly time consuming task. However, depending on the fidelity of the electron energy level description employed in the CR model, as well as the atomic number of the impurity being considered, the forward pass evaluation time can become computationally expensive, as previously mentioned. Given this observation, it

is straightforward to then imagine how training data generation could become a very time consuming part of constructing a sufficiently accurate surrogate for a CR model.

To address this potential pitfall, beginning in section 3, we present a method to adaptively sample over the parameter space to autonomously determine where additional sample points are required, generate those samples, and re-train (and re-sample) the ANN surrogate until a pre-determined accuracy criteria is met. In this way, we find a time-efficient method to produce a physically representative set of training data, train the ANN surrogate, and ultimately yield an accurate surrogate, which is shown in section 4 to outperform one trained on an order of magnitude more data obtained by standard Latin hypercube sampling.

2. CR modeling and its application

Accurate modeling of fusion plasmas with added impurities is an indispensable resource to understanding the physics and mitigation plan of tokamak disruptions [16]. In the simplest scenario, a typical tokamak fusion plasma will be created from hydrogen (deuterium/tritium) gas, which is subject to tremendous amounts of energy to strip electrons from atoms and produce a fully ionized plasma of electrons, positively charged ions with a charge of $+1$, and a minute fraction of neutral atoms.

For simplicity in the remainder of this study we will just refer to a deuterium plasma as intended for eventual operation of ITER, where the number density of deuterium species is n_D, often on the order of 10^{13}–10^{15} m^{-3} depending on a given device. Densities of the singly charged ion and atom may be referred to as n_D^{+1}, n_D^0, where their sum is equal to the total density of deuterium species i.e. $n_D = n_D^{+1} + n_D^0$. Additional descriptive variables of a basic fusion relevant discharge are often the density of electrons in the plasma, n_e, and the average temperature of the plasma electrons, T_e, which characterizes a Maxwell–Boltzmann equilibrium energy distribution

$$f_M(E) = \frac{2}{T_e^{3/2}} \sqrt{\frac{E}{\pi}} \exp\left(\frac{-E}{T_e}\right), \tag{1}$$

where E is electron energy in electronvolts (eV) [1 eV $\approx 1.602 \times 10^{-19}$ J], and T_e is the electron average temperature in eV. Typically fusion plasma temperatures are on the order of 10^3–10^4 eV.

When a discharge is composed of purely hydrogenic atoms or ions (with atomic number $Z = 1$), the atomic physics considerations are generally much simpler, and one may defer to physically and computationally simple models to describe the system with reasonable accuracy [20]. However, in recent times the fusion community has been trialing the addition of impurity gases into the plasma to help control the stability of the plasma, and act as a safeguard against deposition of intolerably large levels of heat against the device's inside walls that contain the plasma [6, 16, 37]. For the purposes of this study, we can characterize the amount of impurity gas added via a number density, n_I. Typical gases in these scenarios include helium ($Z = 2$), nitrogen ($Z = 7$), neon ($Z = 10$), and argon ($Z = 18$) and knowledge of the populations of these atoms and associated ions is crucial to accurately modeling a discharge.

We note that a further consideration of adding the aforementioned impurities to a high temperature hydrogenic plasma around 10^3–10^4 eV is that it will result in cooling of the plasma down to 10^0–10^1 eV, and so a wide range of electron temperatures, T_e, must be considered.

So, given some impurity of atomic number Z and density n_I, added to a plasma with deuterium density n_D described by an electron temperature T_e, the modeling goal is to determine the population densities of the deuterium atom and ion, as well as the impurity atom and ions, in an equilibrium plasma environment (i.e. spatially and temporally homogeneous). Given the wide range of possible ion states that can occur in a plasma, we henceforth will refer to any generic ion level population as n_j, and the set of all ion populations, n_j, in a given plasma as **n**. An additional output variable we seek is the total radiative power loss (RPL), $P_{rad}(\mathbf{n})$, of the plasma given a certain T_e and **n**. This scalar quantity is a function of **n** and describes contributions to radiation being emitted by photons from radiatively decaying excited ion states, electrons radiatively recombining with ions, and electron Bremsstrahlung radiation [11].

To access the desired populations of all ions in the plasma, **n**, and the radiation power loss, $P_{rad}(\mathbf{n})$, a CR model is used to solve for the populations of ions in various excited energy states [11, 18, 38, 39]. A CR model is formulated as a set of ordinary differential equations for each ion population, n_j, that describe transitions into $(i \to j)$ or out $(j \to i)$ from all other possible ion populations, n_i, where $1 \leqslant i, j \leqslant N$ and N is the total number of ion populations in the state vector, **n**. This rate problem can be expressed as

$$\frac{dn_j}{dt} = \sum_{i=0, i \neq j}^{N} R_{i \to j} n_i - \left(\sum_{i=0, i \neq j}^{N} R_{j \to i}\right) n_j, \tag{2}$$

where each $R_{i \rightarrow j}$ is a sum over the possible collisional and radiative transitions that can occur in a plasma, such as excitation or ionization of ions via electron impact collisions, or radiative decay of excited states or radiative recombination of ions [11, 40]. The rates of these atomic physics events are dependent on the electron energy distribution present in the plasma, and can add another degree of difficulty to the problem. As alluded to previously, the Maxwell–Boltzmann equilibrium electron energy distribution (1) is assumed for the purposes of this study, meaning most rates must be computed for a given plasma temperature, T_e.

Given the large number of possible species populations it is natural to form a non-linear rate matrix problem

$$\frac{d\mathbf{n}}{dt} = \mathsf{R}(\mathbf{n})\mathbf{n}, \tag{3}$$

where $\mathsf{R}(\mathbf{n})$ is the rate matrix composed of elements $R_{i \rightarrow j}$ and $\mathbf{n}$ is the state vector previously discussed.

If the CR relaxation timescale to an equilibrium is faster than plasma transport effects being simulated in a parent code then it is reasonable to assume a steady-state scenario to determine plasma properties within such a simulation. In this limit we seek to solve

$$\mathsf{R}(\mathbf{n})\mathbf{n} = 0. \tag{4}$$

In the implementation of most CR models, a range of possible excited quantum states for each ion stage are specified, $n_{Z_j}^{\alpha_j}$ where the index Z_j denotes an ion charge state, and α_j denotes the excited state level of the jth species in $\mathbf{n}$. Generally, plasma transport models will only require the population of each ion stage, irrelevant of what excited quantum states are occupied, and so in post-processing the output of a CR model we will sum over all excited states, α_j, to find the population of a given ion stage Z_j

$$n_{Z_j} = \sum_{\alpha_j} n_{Z_j}^{\alpha_j}. \tag{5}$$

In the application of a CR model, there are many quantities determined by solution of the model that have use for immediate interpretation of a plasma and its properties, and also use for coupling within a larger plasma simulation framework, such as a plasma particle transport simulation code. The fundamental output quantities of a CR model are the ion population densities, n_j, that compose the state vector of the CR system, $\mathbf{n}$. When these ion populations are known, the CR model can then compute the radiation potential of the plasma, $P_{\text{rad}}(\mathbf{n})$.

An important quantity is the free electron density of the plasma, n_e, formed as a result of ionization of atoms and ions in the plasma. In later sections we will use this quantity to compare output performance of potential surrogates. In this work, we are considering a deuterium plasma combined with some higher Z impurity, so the electron density is given by

$$n_e = n_D^{+1} + \sum_{Z_j=0}^{Z} Z_j n_{Z_j}, \tag{6}$$

where n_D^{+1} is the number density of deuterium ions, n_{Z_j} is the number density of the impurity ion with charge Z_j, and Z is the atomic number of the impurity species.

An additional output quantity of a CR model used to characterize a plasma is the average ion charge state, $\bar{Z}$. A weighted average of ion charge with ion population density, we can write $\bar{Z}$ for an impurity species as

$$\bar{Z} = \frac{\sum_{Z_j=0}^{Z} Z_j n_{Z_j}}{\sum_{Z_j=0}^{Z} n_{Z_j}}. \tag{7}$$

Solution of (4) can be computationally challenging when $\mathbf{n}$, and thus R, is large, or R is near singular. Solution of the non-linear rate-matrix problem (4) is usually performed with a standard linear algebra package. Once $\mathbf{n}$ is known, the sum of radiation power loss due to each excited ion state is then computed by the CR model, and P_{rad} is then passed back to the main execution loop of a plasma model [5, 6, 41].

Ideally, one would simply solve a CR model at each time-step or iteration of a plasma code performing a desired simulation, however as outlined in the previous section the computational bottleneck this would create is unsatisfactory. As such, in the following Sections we present an efficient method to assemble physically meaningful training data to allow the creation of an effective ANN surrogate. Essentially, such a surrogate will provide the required mapping of input plasma conditions to output quantities normally computed from a CR model in milliseconds, rather than seconds or often longer.

3. Surrogate modeling strategy and demonstration

With the forward pass CR model detailed, and the requirements of a potential surrogate outlined, we now detail the formulation of the surrogate modeling strategy. We begin with consideration of the input and output quantities required for the surrogate mapping, before then detailing the two-tiered surrogate forms and the adaptive sampling strategy.

When considering the input quantity of an impurity species' atomic number, Z, we must note the fundamentally different nature of plasma particle interactions with targets of different atomic numbers, Z. For example, electron scattering from small targets like hydrogen or helium ($Z = 1, 2$) is significantly different to scattering from larger targets such as neon or argon ($Z = 10, 18$). The subsequent impacts this can have on plasma properties are not trivial due to each integer value of Z essentially posing a discrete quantum mechanical scattering problem, which could be very different from the problem for adjacent elements on the periodic table, i.e. $Z - 1$ or $Z + 1$. As such, we believe it is preferable to choose not to treat Z as a continuous variable to be fed as an input, and rather defer to generation of fit-for-purpose surrogates for each required value of Z. Given the application of these surrogates is limited to a small number of tokamak fusion relevant elements in the periodic table the option of fit-for-purpose surrogates specific to each atomic number is a sensible treatment.

Therefore, in this work, for a given impurity under consideration, with atomic number Z, any surrogate we seek will provide a map for an input field of: electron temperature, impurity density, deuterium density, $\{T_e, n_I, n_D\}$, to an output field of the total RPL and ion stage populations for deuterium and the added impurity, $\{P_{rad}, n_D^0, n_D^{+1}, n_I^0, n_I^{+1}, \ldots, n_I^{+(Z-1)}, n_I^{+Z}\}$.

To generate sample data for training, we must specify bounds on the input fields. Within typical parameters expected in the application of a CR surrogate to tokamak disruption mitigation plasma modeling, we limit the input field quantities to the domains of $1\,\text{eV} \leqslant T_e \leqslant 1000\,\text{eV}$, $10^{13}\,\text{cm}^{-3} \leqslant n_I \leqslant 10^{15}\,\text{cm}^{-3}$, and $10^{13}\,\text{cm}^{-3} \leqslant n_D \leqslant 10^{15}\,\text{cm}^{-3}$. In order to regularize the input field being mapped to output, each variable is normalized to the domain $[0, 1]$ via

$$\hat{x} = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{8}$$

where x and $\hat{x}$ are raw and normalized quantities. We choose this input scaling to allow uniform sampling over $[0, 1]$ as we have no *a priori* expectation of possible input quantity values. For example, essentially any one value of deuterium and/or added impurity atom densities composing the plasma is just as likely as another.

The ion populations in the output field are normalized against the input impurity number density, n_I, and deuterium, n_D, densities, such that we henceforth assume to be dealing with fractional populations such that

$$n_D^0 + n_D^{+1} = 1, \tag{9}$$

$$n_I^0 + n_I^{+1} + \cdots + n_I^{+(Z-1)} + n_I^{+Z} = 1. \tag{10}$$

Given the standard unit of P_{rad} may vary by up to approximately ten orders of magnitude on a logarithmic scale, depending on multiple input factors, we choose to operate with $\log_{10}(P_{rad})$ in order to bring this element of the output field to be on a similar order of magnitude as the fractional ion populations.

In the following section, we evaluate the construction of surrogate models for our CR dataset so that inexpensive forward models may be obtained using ANN. Notably, our goal is to not just obtain a cheap surrogate but to also balance the offline cost of representative data set generation for training this surrogate. This is obtained via an adaptive sampling explained below. We use two surrogate models in this research: a low fidelity (LF) and high fidelity (HF) surrogate. The two models differ in the ability to accurately represent the CR input-output relationship and are used for two distinct purposes in our surrogate development campaign.

3.1. LF surrogate: random forest regressor (RFR)

The first of the fundamental surrogate models employed in our work is an LF RFR [42]. RFR is an ensemble machine learning technique popularly used for regression and classification tasks [43]. It is built around the utilization of many decision trees (thus leading to the 'forest' terminology). A decision tree is an efficient algorithm for splitting a data set according to its various independent variables leading to a branch-like structure with each 'node' corresponding to a splitting based on one variable. The decision to split according

to one independent variable as against the others is taken by measuring the effectiveness of the split through metrics such as impurity or variance reduction in the split. For a regression task, impurity is defined as

$$H(Q_m) = \frac{1}{N_m} \sum_{i=0}^{N_m} (y_i - \bar{y}_m)^2, \tag{11}$$

with $\bar{y}_m$ is the mean of the dependent variable y_i given by

$$\bar{y}_m = \frac{1}{N_m} \sum_{i=0}^{N_m} y_i, \tag{12}$$

and where N_m is the number of data points at a particular node m. The set Q_m represents the data that resides at a node. The total impurity at this node may then be expressed as

$$G(Q_m, \theta) = \frac{n_{\text{left}}}{N_m} H(Q_{\text{left}}(\theta)) + \frac{n_{\text{right}}}{N_m} H(Q_{\text{right}}(\theta)), \tag{13}$$

where n_{left} and n_{right} correspond to the number of children data points in the left and right branches arising from the node m and

$$Q_{\text{left}}(\theta) = (x, y) | x_j <= t_m,$$
$$Q_{\text{right}}(\theta) = (x, y) | x_j > t_m \tag{14}$$

are the left and right split data sets respectively. Note how the splitting of Q_m depends on an independent variable x_j and a threshold t_m at each node. The choice of t_m is generally given by the median value of the attribute x_j i.e. the left data set is comprised of all samples with independent variable x_j less than t_m. The decision tree splits the data Q_m in a manner that minimizes $G(Q_m, \theta)$ by choosing an optimal dimension j for the independent variable. This branching is performed recursively for every node until certain user-defined criteria are met such as $N_m <= N_{\text{tol}}$ at which point the node is denoted a leaf. A branching may also be terminated if a certain maximum depth is reached. A new data point (i.e. an unseen sample) can then follow the branching trajectory (by tracking the order of splits by j and equation (13) and placement into left or right branches by t_m) and reach a leaf. In case a N_{tol} criteria is set, the prediction of the decision tree for a sample is the average prediction value of the dependent variables at this leaf.

RFRs utilize ensembles of decision trees by selecting random subsets of the total data for each tree to obtain a consensus on regression predictions. Each tree utilizes a random subset of the data through sampling with replacement and may therefore obtain a completely different branching structure depending on the distribution of that particular data set. The generation of multiple decision trees as estimators also encourages generalization. The underlying mechanism of our adaptive sampling strategy uses the ensemble property of the RFR wherein each tree in the forest may be queried for a prediction of a scalar metric, to estimate prediction variances in the multidimensional parameter space. If the different trees in an RFR show a wide range of predictions (i.e. there is widespread disagreement in the predictions of trees in a forest) in some region of the finely sampled space, we run full-order CR evaluations in that region, append this to our master training data set and retrain the RFR as well as the HF surrogate. This allows us to adaptively sample higher dimensional spaces which may be challenging to cover due to the curse of dimensionality. To initialize this adaptive algorithm, a first (relatively coarse) sampling of the input space is constructed to train our RFR following which the adaptive sampling is performed iteratively. The end of an adaptive sampling iteration is accompanied by a re-training of the HF surrogate on the expanded training data set. The validation performance of the HF surrogate is recorded for the purpose of *a posteriori* model selection. We note that the sampling at each iteration of the LF surrogate (and at the very start of the adaptive training data augmentation) utilizes the Latin Hypercube sampling (LHS) method for experimental design [44]. We note that a choice of scalar metric for the LF surrogate must be made in order to guide resampling. In this study we trial physical quantities in this role, one being the average charge state of the impurity ions $\bar{Z}$ and the other the total RPL, P_{rad}. Both scalar quantities are physically meaningful and important quantities, but represent different physical aspects of a plasma. The total radiative loss, P_{rad}, is the first element of the output layer, while $\bar{Z}$ is computed as a weighted mean via (7). Given *a priori* neither of them trump the other in terms of which should be the more useful guide, we trial both.

3.2. HF surrogate: ANN

Given a LF surrogate used to rapidly inform re-sampling, the second fundamental surrogate model employed in our work is a HF ANN that may act as a universal approximator [45] trained by backpropagation [46]. One technique to obtain a high-fidelity function approximation is through the use of a multilayered perceptron (MLP) architecture, which is a subclass of feedforward ANN. A general MLP consists of several neurons arranged in multiple layers. These layers consist of one input and one output layer along with several hidden layers. Each layer (with the exception of an input layer) represents a linear operation followed by a nonlinear activation that allows for great flexibility in representing complicated nonlinear mappings. This may be expressed as

$$\mathcal{L}^l\left(\boldsymbol{t}^{l-1}\right) := \boldsymbol{w}^l \boldsymbol{t}^{l-1} + \boldsymbol{b}^l, \tag{15}$$

where $\boldsymbol{t}^{l-1}$ is the output of the previous layer, and $\boldsymbol{w}^l, \boldsymbol{b}^l$ are the weights and biases associated with that layer. The output $\boldsymbol{t}^l$, for each layer may then be transformed by a nonlinear activation, η, such as rectified linear activation:

$$\eta(a) = \mathrm{ReLU}(a) = \max(a, 0). \tag{16}$$

For our experiments, the inputs $\boldsymbol{t}^0$ are in $\mathbb{R}^d$ (i.e. d is the number of inputs) and the outputs $\boldsymbol{t}^K$ are in $\mathbb{R}^k$ (i.e. k is the number of outputs). The final map is given by

$$F : \mathbb{R}^d \mapsto \mathbb{R}^k, \quad \boldsymbol{t}^0 \mapsto \boldsymbol{t}^K = F(\boldsymbol{t}^0; (\boldsymbol{w}, \boldsymbol{b})), \tag{17}$$

where

$$F(\boldsymbol{t}^0; \boldsymbol{w}, \boldsymbol{b}) = \eta^K \left(\mathcal{L}^K \circ \eta_{\mathrm{act}}^{K-1} \circ \mathcal{L}^{K-1} \circ \ldots \circ \eta_{\mathrm{act}}^1 \circ \mathcal{L}^1 \right) (\boldsymbol{t}^0) \tag{18}$$

is a complete representation of the neural network and where **w** and **b** are a collection of all the weights and the biases of the neural network. These weights and biases, lumped together as $\phi = \{\mathbf{w}, \mathbf{b}\}$, are trainable parameters of our map, which can be optimized by examples obtained from a training set. The supervised learning framework requires for this set to have examples of inputs in $\mathbb{R}^{N_{\mathrm{ip}}}$ and their corresponding outputs $\mathbb{R}^{N_{\mathrm{op}}}$. This is coupled with a cost function $\mathcal{C}$, which is a measure of the error of the prediction of the network and the ground truth. Our cost function is given by

$$\mathcal{C} = \frac{1}{|T|} \sum_{(\boldsymbol{\theta}, \tilde{\mathbf{x}}) \in T} \| \tilde{\mathbf{x}} - F(\boldsymbol{\theta}; \phi) \|^2 \tag{19}$$

where $|T|$ indicates the cardinality of the training data set, consisting input, θ, and output fields, $\tilde{\mathbf{x}}$, given by

$$T = \{ (\boldsymbol{\theta}_i, \tilde{\mathbf{x}}_i) : \tilde{\mathbf{x}}_i = f(\boldsymbol{\theta}_i) \} \tag{20}$$

and where $f(\boldsymbol{\theta}_i)$ are examples of the true targets obtained from the adaptively generated training data set. The trained MLP may be used for uniformly approximating any continuous function on compact domains [47, 48], provided $\eta(x)$ is not polynomial in nature.

3.3. Training and adaptive sampling of training data

In this section, we introduce a methodology to iteratively use two surrogate models—the RFR and ANN for reducing computational costs for balancing accuracy and training data generation costs. The high-level idea is to use the RFR to identify suitable regions for sampling training data and fitting an ANN regressor with the improved training data set. By coupling the two surrogates for this approach—accurate surrogate models may be obtained that leverage the expressiveness of the ANN but rely on the ease of training of the RFR for identifying a superior dataset. Note that one may also use ANNs that provide uncertainty estimates for identifying regions of high variance for adaptive sampling [49]. However, training such networks is generally more expensive that regular networks and imprecise in the limit of small data. We note, however, that by choosing surrogates from two different classes, the RFR and ANN, we utilize an assumption (i.e. an inductive bias) that the predictive variance from the predictions of different decision trees imply in inability to learn a consistent function approximation in an ensemble sense. It follows then, that the predictive variance is an appropriate region to target for sampling to improve any surrogate model. Such inductive biases have proven effective for various function approximation tasks [50].

    The hyperparameters that need to be selected *a priori* for this framework include the number of samples required for the first training iteration, the number of new samples for each data set augmentation, the total

---

Algorithm 1. Adaptive CR surrogate training sampling.

Generate N_{init} initial samples of forward CR model evaluation;
Train HF surrogate;
while $N_{\text{samples}} < N_{\text{budget}}$ **do**
  Train LF surrogate to map input field to scalar metric;
  Evaluate fine sample space grid using LF surrogate. Sort list by variance;
  for $i = 1 : (N_{\text{samples}}/10)$ **do**
    Take input field of highest variance LF prediction. Generate new sample;
  end
  Train HF surrogate on expanded sample set. Evaluate surrogate metrics;
  if $R^2 > R^2_{\text{goal}}$ **then**
    STOP
  end
end

---

computational budget for training the surrogate, and the traditional hyperparameters of the MLP (e.g. the learning rate, early stopping criterion, architecture) and the RFR (the number of trees, maximum depth of the forest). Our sampling strategy is outlined in algorithm 1 where the hyperparameters of the associated components are as follows: $N_{\text{init}} = 50$ initial samples, $N_{\text{resample}} = 0.1 N_{\text{samples}}$ re-samples per data set augmentation iteration given a current number of acquired samples N_{samples}, and a budget of $N_{\text{budget}} = 1950$ samples to make 2000 samples the maximum possible. The MLP was chosen to have two hidden layers, 50 neurons wide each, between the input and output layers of dimension three and 14 respectively. The ReLU activation function was employed in this network. The learning rate was fixed as 10^{-3} with the Adam algorithm [51] used for optimization. An early stopping criteria was set to halt learning after 100 epoch of failing to reduce validation loss. Hyperparameters of the RFR were chosen as 50 trees in the forest, with a maximum depth of eight. Stopping criteria for the overall adaptive sampling framework was set to be when a validation R^2 result of 0.95 or greater is obtained.
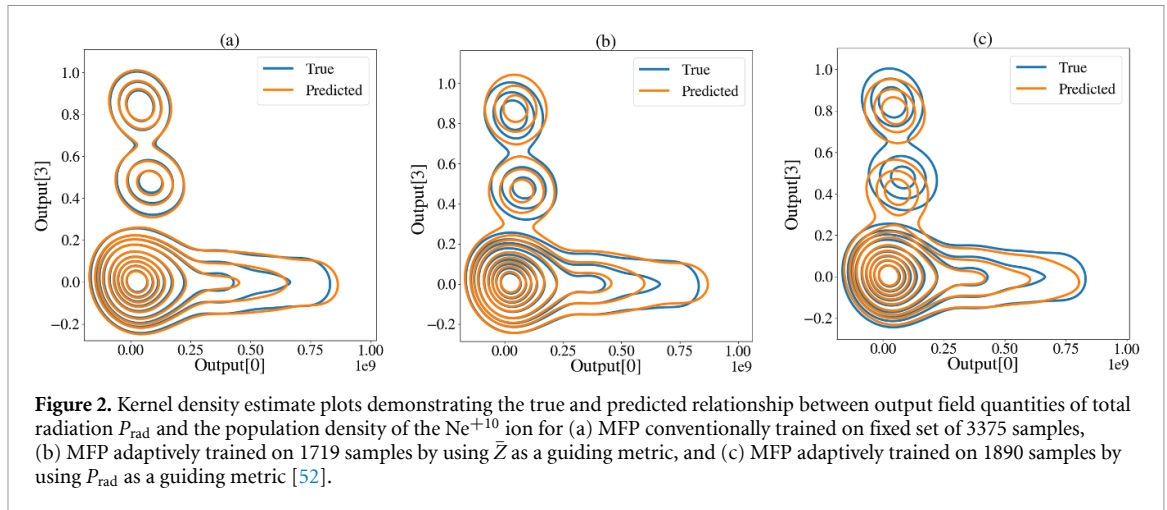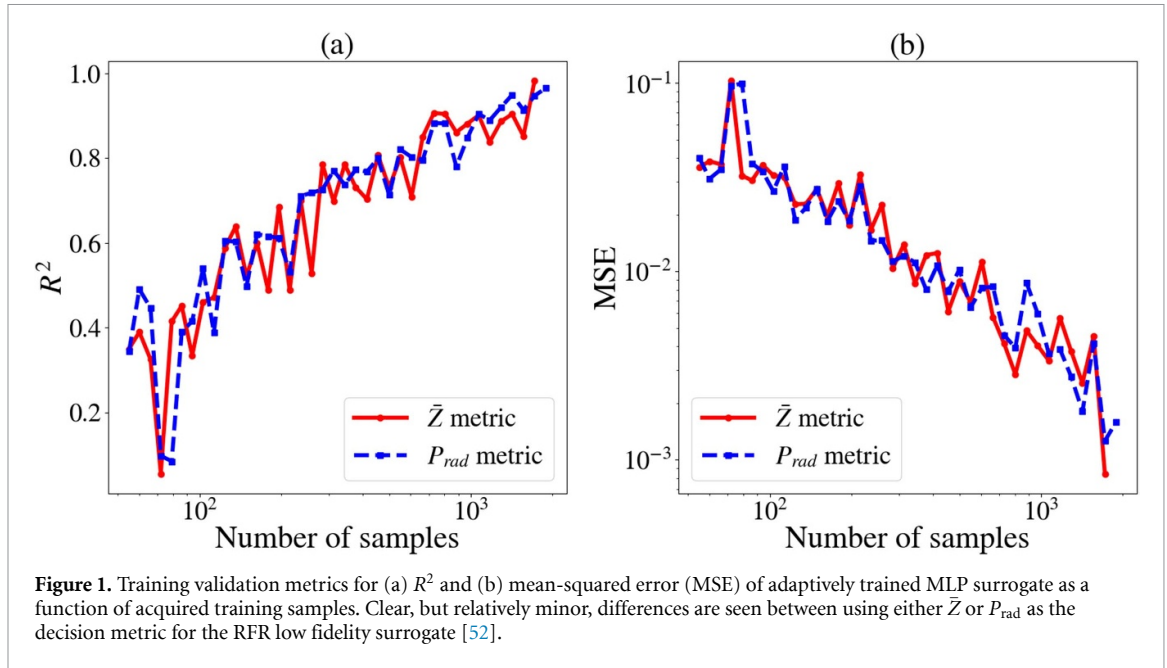
3.4. Experiments and discussion

Given the proposed adaptive sampling method, we now examine the performance of the adaptive sampling routine and compare the performance of a resulting surrogate versus a traditional approach to data generation and training of an ANN. All assessments in this study were conducted with Tensorflow 2.0.0 under Python 3.7.9 on a 2.9 GHz Intel Core i9 CPU with 32GB of RAM. In this section we choose to specify $Z = 10$, which corresponds to a deuterium and neon plasma mixture, which is crucial to planned ITER disruption mitigation strategy [15, 17].

Employing the network architectures and hyperparameter choices outlined in the previous section, we now present some representative results from data generation and training exercises to examine the ability of the adaptive sampling framework to produce an ANN surrogate trained on autonomously gathered training data. Employing either of the two scalar metrics of $\bar{Z}$ and P_{rad} in RFR decision making, we found both cases produced a steady convergence in validation R^2 performance and drop in validation mean-squared error (MSE) as more samples were acquired, as shown in figure 1. While non-monotonic behavior is indeed seen in both quantities with increasing samples added, we can observe that these fluctuations follow an alternating prediction-correction behavior, which is an expected result of the designed sampling routine. Ultimately the adaptively trained multilayer feedforward perceptron (MFP) surrogates passed the R^2 threshold of 0.95 and final surrogate models were obtained. We found that the performance difference between $\bar{Z}$ and P_{rad} scalar metrics is usually comparable, however it was observed that the $\bar{Z}$ option was generally more reliable in reaching the training goal earlier with fewer forward pass model evaluations required. By example, for the case shown in figure 1 the $\bar{Z}$ metric case yielded a validation R^2 of 0.983 and MSE of 8.422×10^{-4}, after 1719 samples were acquired in training. The case of the P_{rad} metric yielded a validation R^2 of 0.965 and MSE of 1.587×10^{-3} using 1890 samples.

As a point of reference for the performance of the adaptive sampling framework, an ANN surrogate was simply trained on a fixed data set of size 3375, which is the result of choosing a space of $15 \times 15 \times 15$ samples determined via LHS. The hyperparameters outlined in the previous section were employed in training this surrogate, and a test R^2 of 0.999 was obtained against an unseen test set. To demonstrate the ability of this ANN, from this unseen test set, a kernel density estimate (KDE) plot showing the predicted relationship between two output field quantities is shown in figure 2.

Testing the adaptively trained MLP surrogates on an unseen data set, held back from training or validation data, we present KDE plots in figure 2 showing the relationship between the output field

Figure 1. Training validation metrics for (a) R^2 and (b) mean-squared error (MSE) of adaptively trained MLP surrogate as a function of acquired training samples. Clear, but relatively minor, differences are seen between using either $\bar{Z}$ or P_{rad} as the decision metric for the RFR low fidelity surrogate [52].

Figure 2. Kernel density estimate plots demonstrating the true and predicted relationship between output field quantities of total radiation P_{rad} and the population density of the Ne^{+10} ion for (a) MFP conventionally trained on fixed set of 3375 samples, (b) MFP adaptively trained on 1719 samples by using $\bar{Z}$ as a guiding metric, and (c) MFP adaptively trained on 1890 samples by using P_{rad} as a guiding metric [52].

quantities of total radiation P_{rad} and the population density of the Ne^{+10} ion as a way to demonstrate the performance of predicted quantities versus the true values. Here we see, compared to the ANN trained on a larger training set, the $\bar{Z}$ guided surrogate performs reasonably well, with the exception of some translation of the prediction over the two upper islands in the plot. Similarly, the surrogate guided by the P_{rad} metric does not accurately reproduce the upper two islands of the KDE. We note that testing R^2 values of 0.982 and 0.967 were obtained for the adaptively trained networks informed by $\bar{Z}$ and P_{rad} respectively.

Given the primary motivator of constructing a framework for surrogate training was for rapid *in situ* evaluation, a timing comparison for 1000 forward pass evaluations of the ANN surrogate and original CR model executable was performed to highlight the compute time savings. The mean surrogate execution time was approximately 4.2 ms, while the mean CR model forward pass execution was 2.7 s. While the direct comparison between adaptively trained surrogates discussed in this section and a traditional ANN surrogate trained on a large pre-assembled data set is at best inconclusive, we have demonstrated the success of the sampling and training method as a proof of concept. To extend upon this basis, in the next section, we refine some of the strategies used in adaptive sampling and demonstrate that a comparable, or often better, ANN surrogate may be produced through automated sampling acquisition.

4. Improving adaptive sampling in training and data acquisition

In the prior section, we examined a brief demonstration of the feasibility in producing a sufficiently accurate surrogate, within the error tolerances forgivable for in-situ coupling in plasma transport codes. Production

of this surrogate via an adaptive training-sampling method was shown to be quicker than assembling an arbitrarily large training data set, while still ensuring enough representative sample points were taken over the parameter space. The proof-of-concept example demonstrated in the previous section was achieved while employing a very simple sample acquisition method that gathered from high variance regions for a scalar output of the HF surrogate.

In this section, we present a more effective sample acquisition method that produces an accurate ANN surrogate with very few samples required to match or exceed an ANN surrogate trained on a very large LHS sampled training set. In lieu of the prior choices for the LF surrogate scalar metric, being the average ion charge state or the total radiation power loss, we now use the LF RFR surrogate to predict the sample error of the HF surrogate

$$\mathrm{MSE} = \frac{1}{N} \sum_j^N (y_j - \hat{y}_j)^2, \tag{21}$$

where N is the number of data points in the training set, y_j is the true value of the jth output, and $\hat{y}_j$ is the corresponding output predicted by the ANN output node. With a focus to rapidly evaluate an estimation of the HF surrogate's error over a finely grained input parameter space, we now propose an improved training sample acquisition function.

4.1. Balancing exploitation vs. exploration

With the use of a LF model that predicts the sample error of validation data for the HF surrogate, we may utilize acquisition function ideas from Bayesian optimization literature to adaptively select new points. First, we sample a large number of unevaluated input parameters. At each of these locations, the low-fidelity model is utilized to compute a mean (μ) and standard-deviation (σ) estimate for the error. Evaluating points with large values of μ indicates that this point can potentially result in reduction of validation error of the HF model, which we denote exploitation. Large values of σ indicate that the low-fidelity model has not adequately learned to predict the error at these locations and therefore these regions may add to the diversity of the overall training data set through a process of exploration. Several acquisition functions have been developed for Bayesian optimization that attempt to balance exploration and exploitation for various settings. In this paper we focus on the lower confidence bound, a simple and robust acquisition function defined by

$$\mathcal{A}_{LCB}(x) = \mu(x) + \lambda \sigma(x), \tag{22}$$

where the parameter $\lambda \geqslant 0$ controls the balance of exploration and exploitation. When set to zero, this parameter performs pure exploitation.

In our work we have examined the effects of varying this λ parameter to control the balance between acquiring new training data samples from parameter space regions of high mean error (exploitation) and regions of high variance (exploration). As a result, we will present a brief comparative study in subsequent sections outlining the benefits of a balanced approach between exploitation and exploration in assembling a representative and meaningful training data set.

4.2. Training and adaptive sampling of training data

As a point of reference for the performance of the adaptive sampling framework, an ANN surrogate was simply trained on a fixed data set of size 8000, spanning a space of $20 \times 20 \times 20$ samples determined via LHS. Here the fixed data set size is increased to 8000 compared to 3375 in the previous section so to raise the bar on the ANN benchmark that we will compare against. The hyperparameters selected for training this iteration of the surrogate model were largely kept the same as the prior training procedure, outlined in the previous section 3.3. Notable changes to these hyperparameters were that we employed $N_{\mathrm{init}} = 100$ initial samples, $N_{\mathrm{resample}} = 0.1 N_{\mathrm{samples}}$ re-samples per data set augmentation iteration, and a budget of $N_{\mathrm{budget}} = 7900$ samples to make 8000 samples the maximum possible to match the total number of data points used in generation of a high-resolution data set for training a benchmark ANN surrogate. Additionally, we increased the threshold for stopping criteria of the overall adaptive sampling framework to be a validation R^2 result of 0.99 or greater. Further, given the oscillatory nature of the validation criteria observed in figure 1 as a function of increasing samples, we add a requirement that to successfully complete training the goal validation R^2 must be passed in sequential iterations of the outer loop. This ensures an improved surrogate is produced, by having to demonstrate robust convergence to the desired criteria and not simply passing once, before potentially failing on the next data set augmentation. Modifications to the sampling-training algorithm are shown in algorithm 2.

Algorithm 2. Adaptive CR surrogate training sampling.

Generate N_{init} initial samples of forward CR model evaluation;
Train HF surrogate;
while $N_{\text{samples}} < N_{\text{budget}}$ **do**
    Train LF surrogate to map input field to HF surrogate MSE;
    Evaluate fine sample space grid using LF surrogate;
    Evaluate lower confidence bound in (22) and sort list;
    for $i = 1 : (N_{\text{samples}}/10)$ **do**
        Take input field of highest variance LF prediction. Generate new sample;
    end
    Train HF surrogate on expanded sample set. Evaluate surrogate metrics;
    if $R^2 > R_{\text{goal}}^2$ **then**
        Increment pass count N_{pass};
        if $N_{\text{pass}} > 2$ **then**
            STOP
        end
    else
        Reset pass count $N_{\text{pass}} = 0$
    end
end

In addition, the activation functions employed in the HF ANN surrogate in this section were modified from the previous iteration in which we essentially demonstrated a proof-of-concept. Hidden layer neurons were assigned the state-of-the-art Swish activation function [53]

$$\text{swish}(x) = x \times \text{sigmoid}(x) = \frac{x}{1 + \exp(-x)}, \tag{23}$$

to make use of the general observations that it outperforms, or matches, the widely used ReLU activation function as a result of its improved smoothness and differentiability properties [53, 54].

To map all output field quantities between [0,1], we now choose to rescale $\log_{10}(P_{\text{rad}})$ between -20 and 20 using (8). This then enables us to employ an output layer of the ANN using a sigmoid activation function, noting that output variable quantities at the output layer are normalized between (0,1). This choice of activation layer was made with the intention of providing a fit-for-purpose architecture to hopefully improve training performance, but to also circumvent non-physical quantities appearing at the output layers, such as negative population densities of ions for example. Finally, we note that the discussed changes to hyperparameters and activation functions used from the first half of this study were employed so to enhance the fit-for-purpose of the surrogate. All subsequent ANNs were trained using the same updated hyperparameters and activation functions so that we may still focus on the differences arising due to active learning in the training routine compared to training using an arbitrarily large fixed data set.

4.3. Experiments and discussion

Employing the proposed modifications to the adaptive sampling routine, we present quantitative and qualitative results of the performance of the sampling method and resulting surrogates. In this section we choose to specify $Z = 2$, which corresponds to a deuterium and helium plasma mixture, which is relevant to the scenario of helium dilution of a burning plasma undergoing DT fusion.

The main change in this evolution of the adaptive sampling framework is the adjustment of the LF surrogate to now learn and predict the mapping of input fields to the MSE of the HF surrogate at the current iteration. Given the current CR problem being studied, there is no obvious *a priori* choice for the value of the hyperparameter, λ, which balances exploitation and exploration of an error surface for a given input field. In our work, multiple values of λ were trialled and tested in order to gain an empirical understanding of this hyperparameter's influence. Here, we present some results representative of the general trends observed in our study and a short discussion on this matter.

Here, we employ three values of $\lambda = 0.1, 1$, and 10 to highlight the influence of this hyperparameter. We recall from (22) that when $\lambda = 1$ there is equal weighting between variance of the MSE (exploration) and the MSE itself (exploitation). When $\lambda > 1$ priority is placed on exploration as training data samples are collected from regions of relatively high variance in the MSE. Conversely, when $\lambda < 1$ the policy priority is on exploiting regions of high MSE itself to augment the training data set with additional samples.

Following execution of the adaptive sampling and training framework for these values of λ, given the conditions previously outlined in this section, we obtain surrogates for determining deuterium-helium

Table 1. Metrics for varying sizes of adaptively chosen training data acquisition methods.

	Training samples	Training R^2	Training MSE
Large LHS set	8000	0.9992	2.6835×10^{-7}
Balanced AS	655	0.9913	7.0304×10^{-6}
Exploitation biased AS	596	0.9962	6.0526×10^{-6}
Exploration biased AS	339	0.9915	2.4972×10^{-5}

Table 2. Testing metrics for different training data acquisition methods.

	Testing R^2	Testing R^2 w/ n_e
Large LHS set	0.9924	0.9931
Balanced AS	0.9923	0.9976
Exploitation biased AS	0.9911	0.9922
Exploration biased AS	0.9826	0.9891

plasma characteristics. The training metrics, as well as the number of required samples for reaching the training termination criteria are shown in table 1.
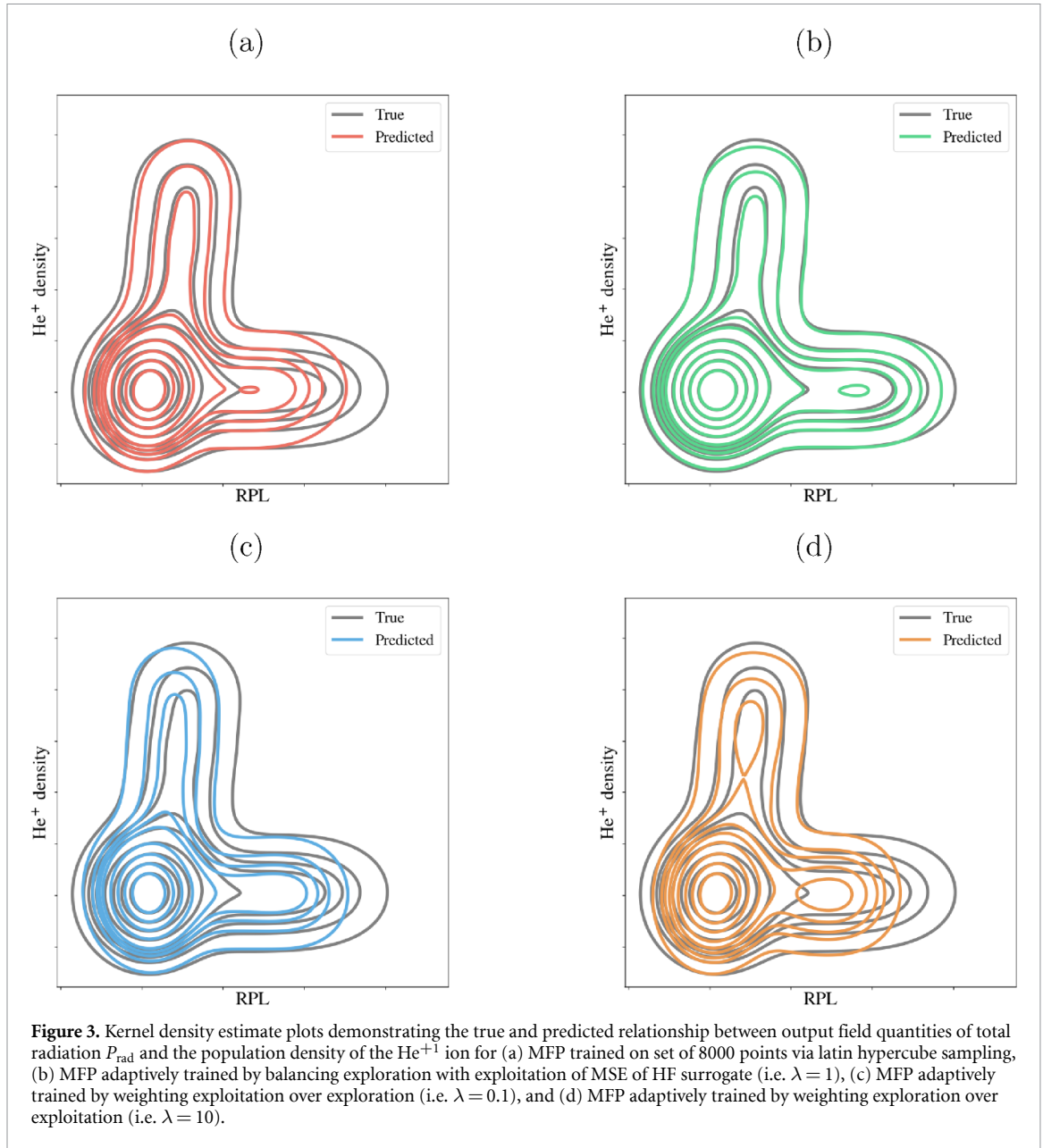
As shown in table 1, using the adaptive sampling framework, with hyperparameters described previously, enables a high training validation goal of $R^2 = 0.99$ to be reached with only a few hundred samples. For a balanced policy, $\lambda = 1$, 655 samples were required, compared to 596 when either exploitation or 339 when exploration were prioritized. For the raw validation metrics, the R^2 and MSE values for $\lambda = 0.1$ are marginally the best performing. However, these metrics are not significantly better than the R^2 and MSE produced for the balanced $\lambda = 1$ case, where both cases yield an MSE on the order of 10^{-6}. We should note that given the capability of single precision operation of the ANN training platform, and for the context of providing fractional ion populations and a transformed RPL quantity, this error is an acceptable outcome for implementation within multiphysics plasma simulation codes. Naturally, the surrogate trained on the very large LHS dataset with 8000 samples performed very well in validation, with an order of magnitude reduction in MSE and R^2 of 0.999.

Following validation, given an unseen data set witheld from training and validation, we can assess the performance of the four ANN surrogates on mapping a set of input fields unseen in training. For this case, the testing R^2 results are shown in table 2. For these assessments testing metrics are computed from the output fields, but in addition a R^2 value is computed when the output field is augmented by an additional element of the electron density, n_e, which can be computed in post-processing from the output field and then appended. Though the n_e quantity is a derived quantity given an output field, it is a very important variable when describing a plasma, and so we include this in our assessment of our surrogates to further test their ability to reproduce physically important values.

Here we can see that performance between the large LHS trained surrogate is comparable to the performance of the adaptively trained surrogate that was exposed to over an order of magnitude less data (8000 vs. 655). In fact, the adaptively trained case with $\lambda = 1$ could be argued to slightly outperform the rest with an R^2 of 0.9976 compared to the surrogate trained on the very large LHS set. This result is encouraging, in that over an order of magnitude less sample points can be used to train an effective surrogate trained using traditional data acquisition methods.

While comparing R^2 and MSE metrics is a reasonable thing to do when evaluating surrogates, we can also visually compare the performance of potential surrogates by projecting the relationships between output field quantities onto KDE plots to provide easy graphical representation of the predicted outputs compared to the known output relationships in the testing set. These KDE plots are shown in figure 3 for the four surrogates being examined in this section.

Interestingly, comparison of subplots (a) and (b) in figure 3 highlights that the MFP trained with a balance of exploitation and exploration better captures the relationship between the two output field quantities considered; He$^+$ ion density and $\log_{10}(P_{rad})$ in this case. Specifically, we see that the adaptively trained surrogate is able to better predict the features in the lower half of the probability distribution than any of the other surrogates. Ultimately, this indicates that the predictions capable of the adaptively trained surrogate are grounded in a training set with more representative samples compared to the very large set of 8000 instances generated via LHS. This outcome is an encouraging demonstration of the intended nature of the adaptive sampling framework to identify regions of parameter space where predictions break down, and assign more sampling resources to those regions.

Figure 3. Kernel density estimate plots demonstrating the true and predicted relationship between output field quantities of total radiation $P_{\rm rad}$ and the population density of the He^{+1} ion for (a) MFP trained on set of 8000 points via latin hypercube sampling, (b) MFP adaptively trained by balancing exploration with exploitation of MSE of HF surrogate (i.e. $\lambda = 1$), (c) MFP adaptively trained by weighting exploitation over exploration (i.e. $\lambda = 0.1$), and (d) MFP adaptively trained by weighting exploration over exploitation (i.e. $\lambda = 10$).

To provide a quantitative comparison to complement the KDE plot shown in figure 3, we can compute a relative entropy measure, specifically the Kullback–Leibler divergence [55, 56], between the distributions of output field quantities of a proposed surrogate and the known truth distribution. In the context of comparing an approximate distribution, Q, trying to emulate a true distribution, P, the Kullback–Leibler divergence, $D_{\rm KL}(P||Q)$, is computed as
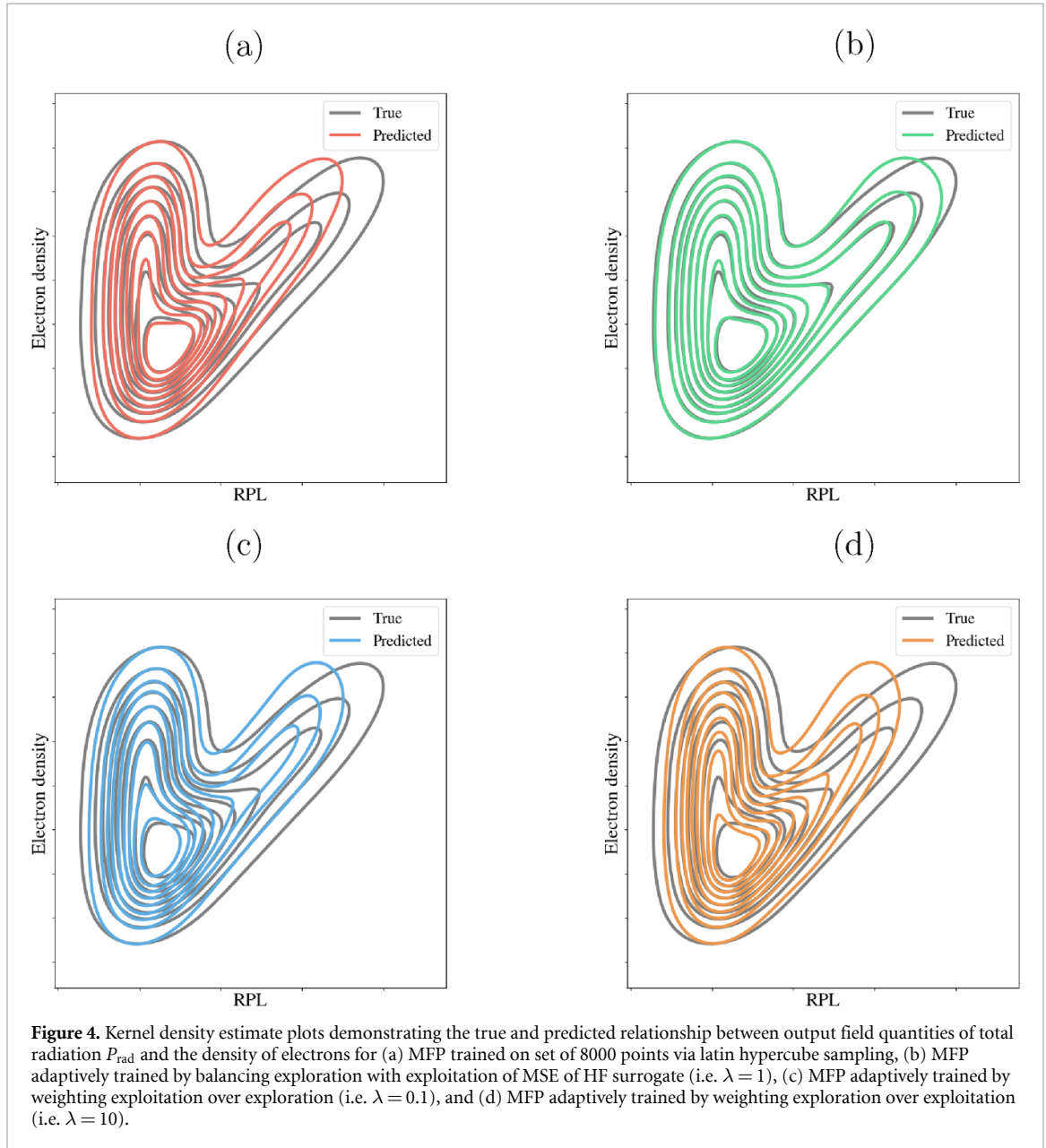
$$D_{\rm KL}(P||Q) = \sum_i P_i \log \frac{P_i}{Q_i}, \qquad (24)$$

and is a measure of entropy increase, or loss of information, due to the use of an approximation, Q, compared to the true distribution itself, P. When $D_{\rm KL}(P||Q) = 0$ it implies that P and Q have identical amounts of information. Thus a smaller value of $D_{\rm KL}(P||Q)$ indicates a better surrogate performance. For the RPL and He$^+$ ion density variables visualized in figure 3 we show $D_{\rm KL}$ values in table 3. Inspecting the numerical values shown in table 3 we see that the improved visual similarity observed in the KDE plots when adaptively sampling with a balanced policy in exploitation and exploration is indeed reflected in the relative entropy measures between the candidate surrogates.

As reinforcement of the observations made from the previous KDE visualization, we present a second KDE figure showing the relationship between electron density and RPL, which are arguably the two most

Table 3. Kullback–Leibler divergence measures for radiation loss and He$^+$ ion density variables visualized in figure 3.
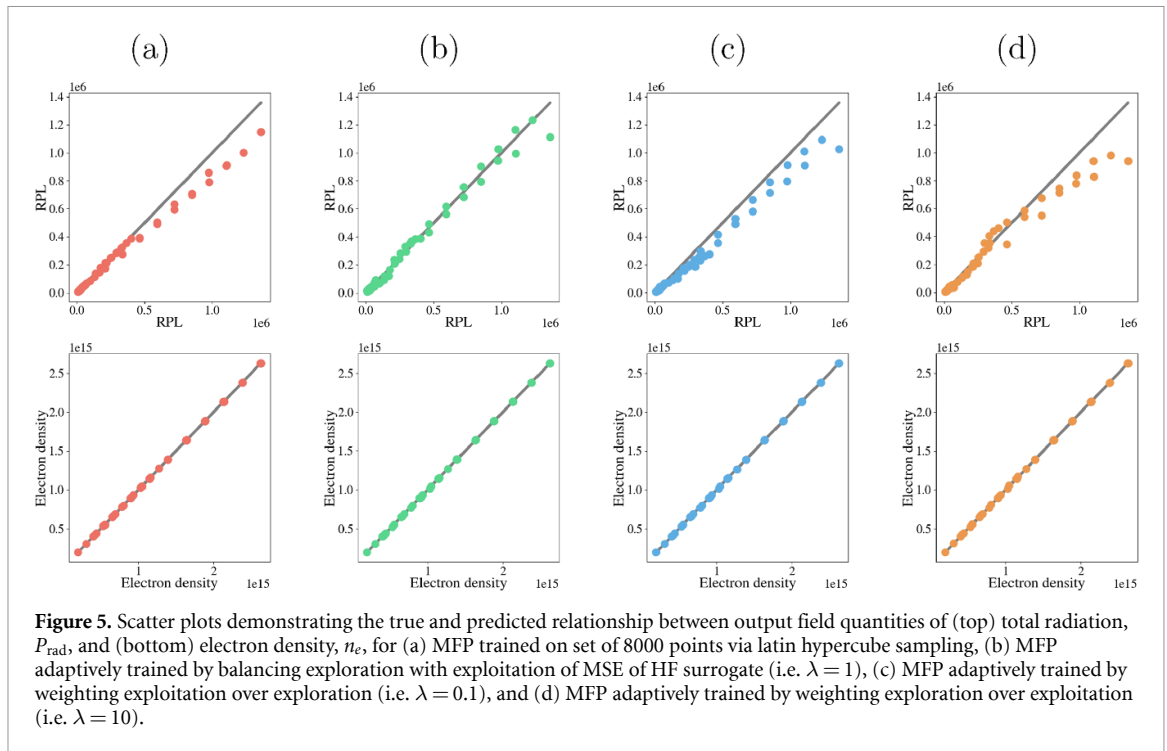
Sampling model	D_{KL} for RPL	D_{KL} for n_e
Large LHS set	5.13×10^{-3}	5.28×10^{-4}
Balanced AS	3.03×10^{-3}	2.81×10^{-4}
Exploitation biased AS	5.81×10^{-3}	1.66×10^{-3}
Exploration biased AS	1.19×10^{-2}	1.40×10^{-1}

Figure 4. Kernel density estimate plots demonstrating the true and predicted relationship between output field quantities of total radiation P_{rad} and the density of electrons for (a) MFP trained on set of 8000 points via latin hypercube sampling, (b) MFP adaptively trained by balancing exploration with exploitation of MSE of HF surrogate (i.e. $\lambda = 1$), (c) MFP adaptively trained by weighting exploitation over exploration (i.e. $\lambda = 0.1$), and (d) MFP adaptively trained by weighting exploration over exploitation (i.e. $\lambda = 10$).

physically meaningful quantities in plasma characterization, in figure 4. In addition to this, we also present scatter plots of these two output variables in figure 5 to assess the performance of the four surrogates in a different light.

A common visualization to assess ANN performance is a scatter graph of predicted output field values against known true values. For the important RPL and electron density variables used in figure 4, we present scatter plots in figure 5. From figure 5 we can observe that all surrogates appear to do a reasonable job at predicting the electron density for a set of unseen input fields. However, as suggested in the KDE plots, the predictive capability for RPL is less than ideal for most surrogates, except the adaptively trained surrogate using $\lambda = 1$.

It is not unsurprising that RPL is a more challenging quantity to reproduce than electron density, or ion charge state populations, given the well known physics of the coronal regime limit [11, 20]. In the coronal
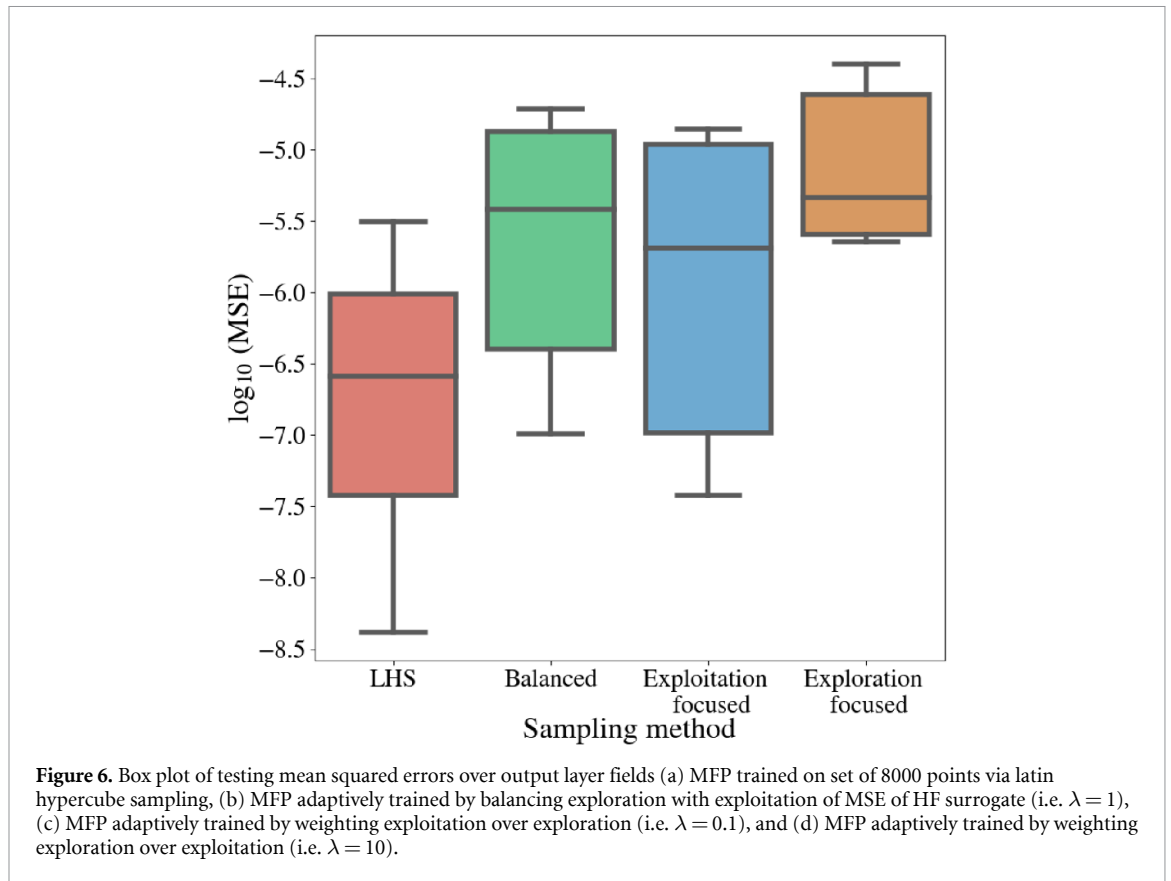
Figure 5. Scatter plots demonstrating the true and predicted relationship between output field quantities of (top) total radiation, P_{rad}, and (bottom) electron density, n_e, for (a) MFP trained on set of 8000 points via latin hypercube sampling, (b) MFP adaptively trained by balancing exploration with exploitation of MSE of HF surrogate (i.e. $\lambda = 1$), (c) MFP adaptively trained by weighting exploitation over exploration (i.e. $\lambda = 0.1$), and (d) MFP adaptively trained by weighting exploration over exploitation (i.e. $\lambda = 10$).

limit plasma densities are assumed to be relatively low, and ion populations are assumed to be in the ground state. Ion population balance is determined between ionization and recombination between ground states of ion charge states, and so very simple models can replace full CR models to determine charge state populations. Where this physical limit breaks down is in the determination of excited states, and thus subsequent RPL quantities, in intermediate regimes where arbitrary populations of ion excited states can occur. This general CR regime is where our current tokamak plasmas appear to sit, though in some scenarios accurate ion charge state populations can be determined from coronal-like models if the conditions are appropriate [7]. Ultimately, from these physics considerations, the total RPL quantity is very sensitive to small changes in excited state populations, and thus makes for a more challenging quantity to learn accurate mappings for when compared to ion charge state populations.

Despite the reduced size of the training data set, this surrogate is able to reasonably predict the RPL much better than the alternatives. Again, this result underscores the virtue of the adaptive sampling framework to not only produce a sufficiently accurate surrogate in a shorter time, with fewer training samples required, but it also ensures a physically meaningful set of data is gathered. By exposing the MLP training infrastructure to a much more representative set of samples we can then take greater confidence that a surrogate we obtain via this method can be used to bypass an original forward pass CR model coupled within a multiphysics plasma simulation. Consequently, with confidence in the fidelity of a CR surrogate, that can be demonstrably produced cheaply, there are now greater opportunities for greatly improved coupling between atomic and plasma physics models at next to no computational expense thanks to the nature of rapidly executable ANN surrogates.

Finally, we now comment on a measurement in which the large LHS data set remains superior: the MSE, as shown in table 1, where a $\mathcal{O}(10^{-7})$ training MSE was reported compared to the $\mathcal{O}(10^{-6})$ training MSE for the cheaper balanced adaptively sampled surrogate. To briefly explore this quantity, we present box plots of the testing MSEs obtained over output layer fields in figure 6. Here, we can see that broadly the ANN surrogate trained on the LHS set is able to maintain a window of approximately an order of magnitude around an MSE $\mathcal{O}(10^{-6.5})$. The balanced adaptively sampled ANN surrogate is able to maintain a similar bandwidth of MSE but is pinned at a higher $\mathcal{O}(10^{-5.5})$ MSE.

We note that while not the best overall candidate in our study, the exploitation biased adaptively sampled NN surrogate was able to yield a lower MSE floor than the balanced sampling case. The latter was able to provide results which were more aligned with the true behavior when comparing KL-divergence metrics. This outcome is a direct result of the objective function seeking to minimize the surrogate MSE in training. Another demonstration of the impact of our objective function policy can be seen in the MSE of the exploration biased surrogate. Here, the bandwidth of MSE error is by far the smallest due to the penalty added to minimize variance of the MSE, but not the MSE itself. As a result, the MSE floor and ceiling are

Figure 6. Box plot of testing mean squared errors over output layer fields (a) MFP trained on set of 8000 points via latin hypercube sampling, (b) MFP adaptively trained by balancing exploration with exploitation of MSE of HF surrogate (i.e. $\lambda = 1$), (c) MFP adaptively trained by weighting exploitation over exploration (i.e. $\lambda = 0.1$), and (d) MFP adaptively trained by weighting exploration over exploitation (i.e. $\lambda = 10$).

higher for this model despite having a narrow spread. Demonstration of these impacts on MSE due to objective function policy is an encouraging result, and will allow further tailoring and improvements on surrogate training depending on potential MSE requirements of a surrogate's application. Furthermore, we may also use a combination of metrics to determine the sampling strategy for new data sets.

5. Conclusion and future work

In this study we have outlined the motivation for employing ANN surrogates of CR models in coupled multiphysics plasma simulation codes, so that vital quantities can be provided to the physics simulation in a computationally efficient manner, without adding a bottleneck at each time-step or iteration of a code. Circumventing this bottleneck with ANN surrogates has previously been successfully demonstrated in other plasma modeling applications [36]. With this in mind, we have introduced and demonstrated the capability of a novel adaptive sampling framework to autonomously generate training data in statistically meaningful areas of the CR problem parameter space, which in turn allows: (a) faster acquisition of training (validation) data for training the ANN surrogate, and (b) some assurance that the data assembled for training (validation) is physically meaningful and representative of the problem parameter space. Further, we have shown that application of this framework towards the creation of an ANN based surrogate for a CR model, that would allow rapid evaluation of ion and radiation properties within a fusion plasma doped with an impurity species, outperforms training an ANN surrogate on more than 10 times the amount of data generated by the often used LHS technique. Future work avenues will explore transfer learning of the ANN (rather than retraining), and parallelized sampling, for example employing DeepHyper [28], as well as implementation of the surrogates presented in this study into plasma fluid simulation research codes. Another direction will be to apply the adaptive sampling framework outlined in this study to more physics rich, but expensive, forward CR models.

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

Acknowledgments

ORCID iDs

Nathan A Garland ● https://orcid.org/0000-0003-0343-0199
Romit Maulik ● https://orcid.org/0000-0001-9731-8936
Qi Tang ● https://orcid.org/0000-0001-9614-1075

References

[1] ITER Physics Basis editors, ITER Physics Expert Group Chairs Co-Chairs and ITER Joint Central Team and Physics Integration Unit 1999 Chapter 1: overview and summary *Nucl. Fusion* **39** 2137
[2] Shimada M *et al* 2007 Chapter 1: overview and summary *Nucl. Fusion* **47** S1–S17
[3] Ferraro N M, Lyons B C, Kim C C, Liu Y Q and Jardin S C 2018 3D two-temperature magnetohydrodynamic modeling of fast thermal quenches due to injected impurities in tokamaks *Nucl. Fusion* **59** 016001
[4] Guo Z, McDevitt C J and Tang X-Z 2017 Phase-space dynamics of runaway electrons in magnetic fields *Plasma Phys. Control. Fusion* **59** 044003
[5] McDevitt C J, Guo Z and Tang X-Z 2019 Avalanche mechanism for runaway electron amplification in a tokamak plasma *Plasma Phys. Control. Fusion* **61** 054008
[6] Hollmann E M *et al* 2020 Study of argon expulsion from the post-disruption runaway electron plateau following low-Z massive gas injection in DIII-D *Phys. Plasmas* **27** 042515
[7] Whyte D G *et al* 2003 Disruption mitigation with high-pressure noble gas injection *J. Nucl. Mater.* **313–316** 1239–46
[8] Yu P A and Krasheninnikov S I 1996 Application of the collisional-radiative, atomic-molecular model to the recombining divertor plasma *Phys. Lett.* A **222** 251–7
[9] Garland N A *et al* 2020 Impact of a minority relativistic electron tail interacting with a thermal plasma containing high-atomic-number impurities *Phys. Plasmas* **27** 040702
[10] Fournier K B, May M J, Pacella D, Finkenthal M, Gregory B C and Goldstein W H 2000 Calculation of the radiative cooling coefficient for krypton in a low density plasma *Nucl. Fusion* **40** 847–64
[11] Ralchenko Y 2016 *Modern Methods in Collisional-Radiative Modeling of Plasmas (Springer Series on Atomic, Optical and Plasma Physics)* (Cham: Springer)
[12] Matthews G F *et al* 2016 Melt damage to the JET ITER-like Wall and divertor *Phys. Scr.* **T167** 014070
[13] Kallenbach A *et al* 2013 Impurity seeding for tokamak power exhaust: from present devices via ITER to DEMO *Plasma Phys. Control. Fusion* **55** 124041
[14] Shrish Raj N Bisai V S and Sen A 2020 Effects of nitrogen seeding in a tokamak plasma *Phys. Plasmas* **27** 122302
[15] Strait E J *et al* 2019 Progress in disruption prevention for ITER *Nucl. Fusion* **59** 112012
[16] Lehnen M *et al* 2015 Disruptions in ITER and strategies for their control and mitigation *J. Nucl. Mater.* **463** 39–48
[17] Lehnen M and Maruyama S 2018 Executive report of the disruption mitigation workshop *Technical Report ITR-18-002* (St-Paul-lez-Durance: ITER Organization)
[18] Chung H K, Chen M H, Morgan W L, Ralchenko Y and Lee R W 2005 FLYCHK: Generalized population kinetics and spectral model for rapid spectroscopic analysis for all elements *High Energy Density Phys.* **1** 3–12
[19] Fontes C J *et al* 2015 The Los Alamos suite of relativistic atomic physics codes *J. Phys. B: At. Mol. Opt. Phys.* **48** 144014
[20] Bauche J 2015 *Atomic Properties in hot Plasmas : From Levels to Superconfigurations* (Cham: Springer)
[21] Summers H P and O'Mullane M G 2011 Atomic data and modelling for fusion: the ADAS project *AIP Conf. Proc.* **1344** 179–87
[22] Whyte D G, Evans T E and Mellman A G E 1997 *Proc. 24th European Conf. on Controlled Fusion and Plasma Physics, Berchtesgaden* vol 21A (Petit-Lancy: European Physical Society) p 1137
[23] van Milligen B P, Tribaldos V and Jiménez J A 1995 Neural network differential equation and plasma equilibrium solver *Phys. Rev. Lett.* **75** 3594–7
[24] Allen S L *et al* 1992 A design study for an advanced divertor for DIII-D and ITER: the radiative slot divertor *J. Nucl. Mater.* **196–198** 804–9
[25] Morgan W L 1991 The feasibility of using neural networks to obtain cross sections from electron swarm data *IEEE Trans. Plasma Sci.* **19** 250–5
[26] Kates-Harbeck J, Svyatkovskiy A and Tang W 2019 Predicting disruptive instabilities in controlled fusion plasmas through deep learning *Nature* **568** 526–31
[27] Rea C, Montes K J, Pau A, Granetz R S and Sauter O 2020 Progress toward interpretable machine learning-based disruption predictors across tokamaks *Fusion Sci. Technol.* **76** 912–24

[28] Maulik R, Egele R, Lusch B and Balaprakash P 2020 Recurrent neural network architecture search for geophysical emulation *SC20: Int. Conf. for High Performance Computing, Networking, Storage and Analysis* (IEEE) pp 1–14

[29] Wang L, Xu X Q,Zhu B, Chenhao M and Lei Y 2020 Deep learning surrogate model for kinetic Landau-fluid closure with collision *AIP Adv.* **10** 075108

[30] Humphreys D *et al* 2020 Advancing fusion with machine learning research needs workshop report *J. Fusion Energy* **39** 123–55

[31] Larsen J T, Morgan W L and Goldstein W H 1992 Artificial neural networks for plasma x-ray spectroscopic analysis *Rev. Sci. Instrum.* **63** 4775–7

[32] Morgan W L, Larsen J T and Goldstein W H 1994 The use of artificial neural networks in plasma spectroscopy *J. Quant. Spectrosc. Radiat. Transfer* **51** 247–53

[33] Osterheld A L, Morgan W L, Larsen J T, Young B K F and Goldstein W H 1994 Analysis of spectra from laser produced plasmas using a neural network *Phys. Rev. Lett.* **73** 1505–8

[34] Zhang W-J, Zhu Xi-M, Wang Y-F, Wang G-X, Yan Y, Meng S-F, Jia J-W and Ning Z-X 2022 A neural network model relating extraction current characteristics with optical emission spectra for the purpose of a digital twin of miniaturized ion thrusters *J. Appl. Phys.* **55** 26LT01

[35] Ghosh K *et al* 2019 Deep learning spectroscopy: neural networks for molecular excitation spectra *Adv. Sci.* **6** 1801367

[36] Kluth G, Humbird K D, Spears B K, Peterson J L, Scott H A, Patel M V, Koning J, Marinak M, Divol L and Young C V 2020 Deep learning for NLTE spectral opacities *Phys. Plasmas* **27** 052707

[37] Harvey R W, Chan V S, Chiu S C, Evans T E, Rosenbluth M N and Whyte D G 2000 Runaway electron production in DIII-D killer pellet experiments, calculated with the CQL3D/KPRAD model *Phys. Plasmas* **7** 4590–9

[38] David Robert B, A Kingston E and McWhirter R W P 1962 Recombination between electrons and atomic ions, II. Optically thick plasmas *Proc. R. Soc.* A **270** 155–67

[39] McWhirter R W P and Hearn A G 1963 A calculation of the instantaneous population densities of the excited levels of hydrogen-like ions in a plasma *Proc. Phys. Soc.* **82** 641–54

[40] Tallents G J 2018 *An Introduction to the Atomic and Radiation Physics of Plasmas* (Cambridge: Cambridge University Press)

[41] Hesslow L, Embréus O, Hoppe M, DuBois T C, Papp G, Rahm M and Fülöp T 2018 Generalized collision operator for fast electrons interacting with partially ionized impurities *J. Plasma Phys.* **84** 905840605

[42] Breiman L 2001 Random forests *Mach. Learn.* **45** 5–32

[43] Hastie T, Tibshirani R, Friedman J and Franklin J 2005 The elements of statistical learning: data mining, inference and prediction *Math. Intell.* **27** 83–85

[44] Iman R L, Davenport J M, and Zeigler D K 1980 Latin hypercube sampling (program user's guide) [lhc, in fortran] vol 1 (available at: www.osti.gov/biblio/557163)

[45] Hornik K *et al* 1989 Multilayer feedforward networks are universal approximators *Neural Netw.* **2** 359–66

[46] Hecht-Nielsen R 1992 Theory of the backpropagation neural network *Neural Networks for Perception* (Amsterdam: Elsevier) pp 65–93

[47] Cybenko G 1989 Approximation by superpositions of a sigmoidal function *Math. Control Signals Syst.* **2** 303–14

[48] Barron A R 1993 Universal approximation bounds for superpositions of a sigmoidal function *IEEE Trans. Inf. Theory* **39** 930–45

[49] Mallick T, Balaprakash P and Macfarlane J 2022 Deep-ensemble-based uncertainty quantification in spatiotemporal graph neural networks for traffic forecasting (arXiv:2204.01618)

[50] Balaprakash P, Salim M, Uram T D, Vishwanath V and Wild S M 2018 Deephyper: asynchronous hyperparameter search for deep neural networks *2018 IEEE 25th Int. Conf. on High Performance Computing (HiPC)* (IEEE) pp 42–51

[51] Kingma D P and Jimmy B 2017 Adam: a method for stochastic optimization (arXiv:1412.6980)

[52] Garland N A, Maulik R, Tang Q, Tang X-Z and Balaprakash P 2020 Progress towards high fidelity collisional-radiative model surrogates for rapid *in-situ* evaluation *Third Workshop on Machine Learning and the Physical Sciences (NeurIPS 2020)* (*Vancouver, Canada*) (available at: https://ml4physicalsciences.github.io/2020/files/NeurIPS_ML4PS_2020_79.pdf)

[53] Ramachandran P, Zoph B and Le Q V 2017 Searching for activation functions (arXiv:1710.05941)

[54] Mercioni M A and Holban S 2020 P-swish: activation function with learnable parameters based on swish activation function in deep learning *2020 Int. Symp. on Electronics and Telecommunications (ISETC)* pp 1–4

[55] Shlens J 2014 Notes on Kullback–Leibler divergence and likelihood (arXiv:1404.2000)

[56] Cover T M 2006 *Elements of Information Theory* (Hoboken, NJ: Wiley)