

PAPER • OPEN ACCESS

## Proof of concept of a fast surrogate model of the VMEC code via neural networks in Wendelstein 7-X scenarios

To cite this article: Andrea Merlo *et al* 2021 *Nucl. Fusion* **61** 096039

View the [article online](#) for updates and enhancements.



**IOP | ebooks™**

Bringing together innovative digital publishing with leading authors from the global scientific community.

Start exploring the collection—download the first chapter of every title for free.

# Proof of concept of a fast surrogate model of the VMEC code via neural networks in Wendelstein 7-X scenarios

Andrea Merlo<sup>\*</sup>, Daniel Böckenhoff, Jonathan Schilling, Udo Höfel, Sehyun Kwak, Jakob Svensson, Andrea Pavone, Samuel Aaron Lazerson, Thomas Sunn Pedersen and the W7-X Team<sup>a</sup>

Max-Planck-Institute for Plasma Physics, 17491 Greifswald, Germany

E-mail: [andrea.merlo@ipp.mpg.de](mailto:andrea.merlo@ipp.mpg.de)

Received 13 May 2021, revised 2 July 2021

Accepted for publication 3 August 2021

Published 24 August 2021



## Abstract

In magnetic confinement fusion research, the achievement of high plasma pressure is key to reaching the goal of net energy production. The magnetohydrodynamic (MHD) model is used to self-consistently calculate the effects the plasma pressure induces on the magnetic field used to confine the plasma. Such MHD calculations—usually done computationally—serve as input for the assessment of a number of important physics questions. The variational moments equilibrium code (VMEC) is the most widely used to evaluate 3D ideal-MHD equilibria, as prominently present in stellarators. However, considering the computational cost, it is rarely used in large-scale or online applications (e.g. Bayesian scientific modeling, real-time plasma control). Access to fast MHD equilibria is a challenging problem in fusion research, one which machine learning could effectively address. In this paper, we present artificial neural network (NN) models able to quickly compute the equilibrium magnetic field of Wendelstein 7-X. Magnetic configurations that extensively cover the device operational space, and plasma profiles with volume-averaged normalized plasma pressure  $\langle\beta\rangle$  ( $\beta = \frac{2\mu_0 p}{B^2}$ ) up to 5% and non-zero net toroidal current are included in the data set. By using convolutional layers, the spectral representation of the magnetic flux surfaces can be efficiently computed with a single network. To discover better models, a Bayesian hyper-parameter search is carried out, and 3D convolutional NNs are found to outperform feed-forward fully-connected NNs. The achieved normalized root-mean-squared error, the ratio between the regression error and the spread of the data, ranges from 1% to 20% across the different scenarios. The model inference time for a single equilibrium is on the order of milliseconds. Finally, this work shows the feasibility of a fast NN drop-in surrogate model for VMEC, and it opens up new operational scenarios where target applications could make use of magnetic equilibria at unprecedented scales.

Keywords: neural networks, surrogate models, Wendelstein 7-X, ideal-MHD

(Some figures may appear in colour only in the online journal)

\* Author to whom any correspondence should be addressed.

<sup>a</sup> See Klinger *et al* 2019 (<https://doi.org/10.1088/1741-4326/ab03a7>) for the W7-X Team.



Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](https://creativecommons.org/licenses/by/4.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

## 1. Introduction

The computation of magnetohydrodynamic (MHD) equilibria is central in magnetic confinement fusion, where it represents the core component of most modeling and experimental applications. In the stellarator community, the 3D ideal-MHD variational moments equilibrium code (VMEC) [1] is the most widely used, e.g. to infer plasma parameters [2, 3], to reconstruct magnetic equilibria [4–8], and to design future devices [9–11]. VMEC is also employed for equilibrium studies in perturbed, and hence non-2D, axisymmetric configurations [12–22]. However, a single VMEC equilibrium evaluation can take up to  $\mathcal{O}(10)$  minutes<sup>1</sup> even on a high-performance computing facility, especially for a reactor-relevant high- $\beta$  plasma configuration. Table 1 reports the orders of magnitude of VMEC total iterations and wall-clock time typically encountered in target applications. The high computational cost limits an exhaustive exploration of the use case input space. A parallel version of VMEC has recently been developed [23], however, for example, the wall-clock time of a single free boundary equilibrium reconstruction, both in the case of a stellarator and a tokamak scenario, is still on the order of hours [24, 25].

In this paper, we use artificial neural networks (NNs) (see section 2.3) as function approximators to build a fast surrogate model for VMEC. A reduction in run times of up to 6 orders of magnitude can be achieved. The models are trained on VMEC runs from two independent data sets:  $\mathbb{D}_{\text{config}}$  and  $\mathbb{D}_{\beta}$  (see section 2.2.2).  $\mathbb{D}_{\text{config}}$  includes a wide range of vacuum magnetic configurations, while  $\mathbb{D}_{\beta}$  covers a distribution of plasma profiles for a fixed magnetic configuration. To find better models, and to take the human out of the loop, a Bayesian hyper-parameter (HP) search is performed (see section 2.4).

Since NNs poorly extrapolate beyond the expressiveness of training data, a large and experimentally relevant data set is essential for good out-of-sample performance (see section 2.2). Training runs are sampled as employed in the Bayesian scientific modeling framework Minerva [28, 29], aiming to reduce the covariate shift between the training and test data set. The magnetic configurations are sampled from a large hyper-rectangle around the nine Wendelstein 7-X (W7-X) reference configurations [30], while the plasma profiles are modeled as Gaussian processes (GPs) [31], and domain knowledge is embedded in the training data through *virtual observations* [32–35].

Since VMEC assumes nested magnetic flux surfaces, magnetic islands in the equilibrium field are not included by design. Furthermore, an ideal coil geometry (i.e. no coil misalignment or electro-magnetic deformations) is considered, while ideal coil currents and plasma profiles (i.e. error-free measurements) are assumed. The relaxation of these assumptions is not in the scope of this paper.

In the past, Sengupta *et al* successfully regressed single Fourier coefficients (FCs) of the VMEC output magnetic

**Table 1.** Order of magnitude of VMEC iterations and wall-clock time in target applications. Fixed-boundary equilibria are considered in stellarator optimization, while the inference of plasma parameters and equilibrium reconstruction usually requires free-boundary equilibria. VMEC computation time strongly depends on the run requirements (e.g. radial resolution, Fourier resolution, field periodicity, convergence tolerance), thus the  $10^1$ – $10^4$  s range has been considered in this table.

Application	Iterations	Time (s)
Bayesian inference [26]	$10^4$	$10^5$ – $10^8$
Equilibrium reconstruction [4, 25]	$10^0$	$10^1$ – $10^4$
Stellarator optimization [27]	$10^3$	$10^4$ – $10^7$

field, using function parameterization (FP) with quadratic or cubic polynomials for vacuum [36] and finite beta [37] magnetic configurations. The regression of the full VMEC output was broken down into subproblems, where an FP model was derived for each FC, leading to many free parameters to learn. In this work, on top of the previously mentioned components (i.e. physics-like plasma profiles and HP search), the learning task is to infer the full magnetic field geometry with a single multiple-input multiple-output model, where all the VMEC output FCs are regressed at once (see section 2.2.2). Using a single model drastically reduces the number of free parameters to learn, and it forces the NN to efficiently share them among the outputs. Contrary to FP, it is well-known that sufficiently wide or deep NNs can approximate a broad class of functions [38–42]. In addition, convolutional neural networks (CNNs), as powerful tools of current deep learning methods, are better suited to extract and reproduce translation-invariant spatial features from grid data, and to share their free parameters between the features while reducing overfitting. Furthermore, from the user standpoint, a single model can more easily be improved, adapted, and deployed.

For real-time plasma control, having access to low-cost magnetic equilibria can improve traditional strategies, and enable completely new data-driven approaches (e.g. reinforcement learning (RL) based control). In fusion research, the use of NN models to compute the plasma topology [43–45] and to speed up slow workflows [46–50] is not a novel idea, nevertheless, to our knowledge this paper represents the first which effectively addresses the 3D MHD physics in W7-X scenarios.

## 2. Methods

In the following relevant concepts and employed methodologies are described.

### 2.1. The VMEC code

The equilibrium problem under the ideal-MHD model is characterized by the force balance equation, Ampere’s and Gauss’s law

$$\vec{J} \times \vec{B} = \vec{\nabla} p \quad (1)$$

$$\vec{\nabla} \times \vec{B} = \mu_0 \vec{J} \quad (2)$$

$$\vec{\nabla} \cdot \vec{B} = 0. \quad (3)$$

<sup>1</sup> Run time on the Max Planck computing and data facility (MPCDF) cluster ‘DRACO’, using the *small* partition and 16 cores.

VMEC uses a variational principle to solve the *inverse* formulation, which computes the mapping  $f: \vec{\zeta} \rightarrow \vec{x}$  between flux coordinates  $\vec{\zeta} = (s, \theta, \varphi)$ , normalized toroidal flux ( $s = \frac{\Phi}{\Phi_{\text{edge}}}$ , where  $\Phi(s)$  is the toroidal magnetic flux enclosed between the magnetic axis and the flux surface labeled  $s$ ), poloidal and toroidal angle, respectively, and real space cylindrical coordinates  $\vec{x} = (R, \varphi, Z)$ , major radius, azimuth and height above mid-plane, respectively. VMEC adopts a spectral representation of  $\vec{x}$  along the poloidal and toroidal angles. Assuming stellarator symmetry, the cylindrical coordinates can be expressed as

$$R(s, \theta, \varphi) = \sum_{mn} R_{mn}(s) \cos(m\theta - nN_{\text{fp}}\varphi), \quad (4)$$

$$Z(s, \theta, \varphi) = \sum_{mn} Z_{mn}(s) \sin(m\theta - nN_{\text{fp}}\varphi), \quad (5)$$

where  $N_{\text{fp}} \in \mathbb{N}$  is the number of field periods. Furthermore,

$$\lambda(s, \theta, \varphi) = \sum_{mn} \lambda_{mn}(s) \sin(m\theta - nN_{\text{fp}}\varphi) \quad (6)$$

is an angle renormalization parameter such that  $\theta^* = \theta + \lambda(s, \theta, \varphi)$  represents the poloidal angle for which magnetic field lines are straight in  $(s, \theta^*, \varphi)$  [1]. The equilibrium magnetic field  $\vec{B}$  can be written in contravariant form

$$\vec{B} = B^s \hat{e}_s + B^\theta \hat{e}_\theta + B^\varphi \hat{e}_\varphi = B^\theta \hat{e}_\theta + B^\varphi \hat{e}_\varphi, \quad (7)$$

where  $\vec{B} \cdot \vec{\nabla} p = B^s = 0$  under the assumption of nested magnetic flux surfaces. The non-zero components are given by [1]

$$B^\theta = \frac{1}{\sqrt{g}} \Phi' \left( t - \frac{\partial \lambda}{\partial \varphi} \right), \quad (8)$$

$$B^\varphi = \frac{1}{\sqrt{g}} \Phi' \left( 1 + \frac{\partial \lambda}{\partial \theta} \right), \quad (9)$$

where  $t$  is the rotational transform, the prime denotes  $\partial/\partial s$ , and  $\sqrt{g} = (\vec{\nabla} s \cdot \vec{\nabla} \theta \times \vec{\nabla} \varphi)^{-1}$  is the Jacobian of the coordinate transformation  $f$ .

The covariant representation of  $\vec{B}$  can be obtained from equations (8) and (9) and the metric tensor  $g_{ij} = \hat{e}_i \cdot \hat{e}_j = \frac{\partial \vec{x}}{\partial \zeta_i} \cdot \frac{\partial \vec{x}}{\partial \zeta_j}$  as follows

$$B_\theta = \vec{B} \cdot \hat{e}_\theta = B^\theta g_{\theta\theta} + B^\varphi g_{\varphi\theta}, \quad (10)$$

$$B_\varphi = \vec{B} \cdot \hat{e}_\varphi = B^\theta g_{\theta\varphi} + B^\varphi g_{\varphi\varphi}. \quad (11)$$

Finally, the magnetic field vector strength is given by

$$B^2 = \sum_i B^i B_i = (B^\theta)^2 g_{\theta\theta} + 2B^\theta B^\varphi g_{\theta\varphi} + (B^\varphi)^2 g_{\varphi\varphi}. \quad (12)$$

As in case of  $\vec{x}$ , the magnetic field strength is described by VMEC using a spectral representation:

$$B(s, \theta, \varphi) = \sum_{mn} B_{mn}(s) \cos(m\theta - nN_{\text{fp}}\varphi). \quad (13)$$

Like in [1],  $\vec{x}$  is redefined as  $\vec{x} = (R, \lambda, Z)$ , where the angle renormalization parameter  $\lambda$  replaces the toroidal angle  $\varphi$ .

## 2.2. Data set generation

To generate a large and W7-X relevant data set of magnetic configurations and plasma profiles, Minerva [28, 29] is used. Within Minerva, models are described as directed, acyclic graphs. Each node can be deterministic (e.g. a diagnostic model or a physics code) or probabilistic (e.g. plasma parameters or diagnostic observed quantities). The edges define the dependencies between nodes. Model free parameters can be described via probabilistic nodes, where the node *a priori* distribution encodes the domain knowledge on the parameter. In the *forward mode*, observed quantities can be computed, while in the *inverse mode*, the model free parameters can be inferred with different inversion techniques (e.g. maximum *a posteriori* and Markov chain Monte Carlo methods).

Using Minerva to generate physics relevant samples for NNs training has already been explored [51]. Here, a VMEC node is included in a Minerva model. Free parameters are represented by the magnetic configuration and the plasma profiles. The model is relatively simple and can be built as a stand-alone object, yet Minerva allows embedding domain knowledge (i.e. the prior distribution of the model free parameters) in the NN surrogate by reducing the covariate shift between the training data set  $\mathcal{D}$  and the target application data set  $\mathcal{D}_{\text{target}}$ . This approach is similar to that described in [52], where experimental data have been used to populate the training data set. However, in this work, experimental data are not used directly, but simulated data drawn from experimentally validated distributions are used instead. This allows a dense coverage of the input parameter space, while restricting its extension to physically relevant regions only.

W7-X possesses a  $N_{\text{fp}} = 5$ -fold stellarator symmetry, i.e. the main coil system comprises five identical modules, each of which is point symmetric toward the module center (see section 2.1). The resulting magnetic field has a five-fold symmetry along the toroidal direction. Each half module includes five different non-planar and two planar coils. The vacuum field depends only on the currents  $I_{1\dots 5}$  and  $I_{A,B}$ , respectively, the currents in the non-planar and planar coils. Except for a scaling of the magnetic field strength, the vacuum magnetic configuration does not depend on the absolute values of the coil currents but only on their ratios with respect to  $I_1$ ,  $i_{2\dots 5}$  and  $i_{A,B}$ . The current ratios are uniformly sampled from a hyper-rectangle whose boundaries are provided in table 2. These boundaries cover the nine reference configurations of W7-X [30], while extending to a larger set of conceivable configurations. To obtain a magnetic field strength of approximately 2.5 T on axis, at  $\varphi = 0$  and for the standard configuration, the normalization coil current  $I_1$  is set to  $I_1 = 13\,770$  A.

The plasma profiles cover a broad range of W7-X discharge scenarios, and include plasma pressure on axis up to 200 kPa, corresponding to volume-averaged  $\langle \beta \rangle$  of approximately 5%, and a net toroidal current ranging from  $-10$  kA to  $10$  kA. The profiles are defined as a function of the normalized toroidal flux  $s$ :  $p(s)$  is the pressure of the plasma at the flux surface labeled  $s$ , and  $I(s)$  is the enclosed toroidal current flowing inside the surface  $s$ . With this definition,  $I(s = 1)$  is the total toroidal current in the plasma, which we will refer to as  $I_{\text{tor}}$ .

**Table 2.** Hyper-rectangle boundaries for the vacuum magnetic configurations, pressure and toroidal current profile included in the data set. Each parameter is uniformly sampled.

Magnetic configuration			
Free parameter	Min	Max	Unit
$\Phi_{\text{edge}}$	-2.5	-1.6	Wb
$i_{[1\dots 5]}$	0.6	1.3	—
$i_{[A,B]}$	-1.0	1.0	—
Pressure profile			
$p_0$	0	200	kPa
$\sigma_f$	2.0	4.0	—
$l_{\text{core}}$	2.0	3.0	—
$l_{\text{edge}}$	1.0	2.0	—
$s_0$	0.7	0.9	—
$s_w$	0.3	0.4	—
Toroidal current profile			
$I_{\text{tor}}$	-10	10	kA
$\sigma_f$	2.0	3.0	—
$l_{\text{core}}$	2.0	3.0	—
$l_{\text{edge}}$	3.0	5.0	—
$s_0$	0.1	0.6	—
$s_w$	0.01	0.1	—

Theoretically, all the possible continuous functions for  $s \in [0, 1]$  should be sampled. The exploitation of domain knowledge obtained from experience with W7-X discharges allows us to restrict the function space of the profiles, and to sample the region of interest denser as compared to unconstrained parameterization. The profile shapes are modeled via GPs [53], stochastic processes whose joint distribution of every finite, linear combination of random variables is a multivariate Gaussian. GPs are usually employed in the modeling context, as they can be seen as distributions of functions. For example, a one dimensional function  $f : s \in \mathbb{R} \rightarrow \mathbb{R}$  with a GP prior is

$$f(s) \sim \mathcal{GP}(\mu(s), \Sigma(s, s')), \quad (14)$$

where  $\mu(s)$  is the mean function, and  $\Sigma(s, s')$  is the covariate function. Then, for a set  $S_* := \{s \in \mathbb{R}\}$ , the corresponding  $\mathcal{F}_*$  are distributed as

$$\mathcal{F}_* \sim \mathcal{N}(\mu(S_*), \Sigma(S_*, S_*)), \quad (15)$$

where the covariate function matrix is computed element-wise.

In Minerva, plasma profiles are usually specified as GPs with zero mean, which does not restrict the mean of the posterior process to be zero [53], and squared exponential covariance function [54]. In particular, since profiles can have substantially different gradients in the core and edge regions [55], a non-stationary covariance function [56] is used [57]. Here, the GP mean and covariance functions are

$$\begin{aligned} \mu(s) &= 0, \quad (16) \\ \Sigma(s_i, s_j) &= \sigma_f^2 \sqrt{\frac{2\sigma_x(s_i)\sigma_x(s_j)}{\sigma_x^2(s_i) + \sigma_x^2(s_j)}} \\ &\times \exp\left(-\frac{(s_i - s_j)^2}{\sigma_x^2(s_i) + \sigma_x^2(s_j)}\right) + \sigma_y^2 \delta_{ij}, \quad (17) \end{aligned}$$

where  $\sigma_y$  is usually fixed to  $\sigma_y = 10^{-3}\sigma_f$  [34], and  $\sigma_x$ , which represents the length scale function, is a hyperbolic tangent function

$$\sigma_x(s) = \frac{l_{\text{core}} + l_{\text{edge}}}{2} - \frac{l_{\text{core}} - l_{\text{edge}}}{2} \tanh\left(\frac{s - s_0}{s_w}\right), \quad (18)$$

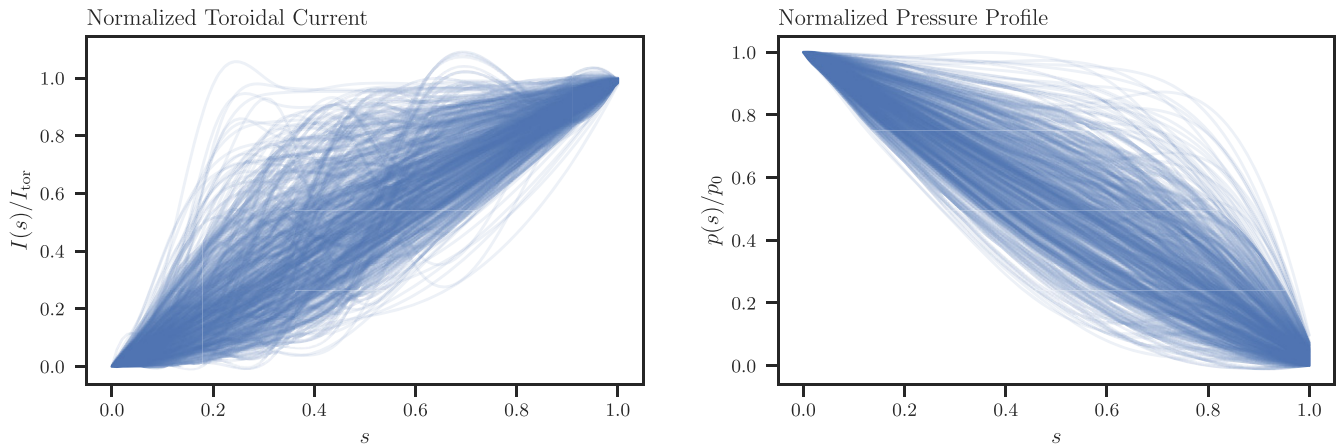
where  $l_{\text{core}}$  and  $l_{\text{edge}}$  are the core and edge length scale, respectively.  $s_0$  is the transition location and  $s_w$  represents the length scale for the transition. The domain knowledge on the plasma profiles is encoded via the HPs of the GP used to represent them, which define the distributions from where the profiles are drawn. The values of the GP HPs are uniformly sampled from a hyper-rectangle, whose boundaries are given in table 2. These values are adapted from previous works where the plasma profiles in W7-X are modeled via GPs [34, 49, 54, 58].

The profiles are further constrained by the use of *virtual observations* [34], such that the GP prior is refined with ‘virtual diagnostic measurements’, described by a normal distribution. As usually observed in W7-X experiments, the electron and ion density and temperature profiles are peaked<sup>2</sup> in the core [59–61]. Therefore, the normalized pressure profile is constrained to 0 at the last closed flux surface (LCFS) and 1 on axis. Contrarily, the normalized toroidal current profile is set to 0 on axis, and 1 at the LCFS. Figure 1 shows a subset of the normalized plasma profiles, which are independently sampled from the two refined GPs. Finally, the profiles are scaled to the desired values: the pressure profile is multiplied by  $p_0$ , the pressure value on axis, and  $I_{\text{tor}}$ , which is the total toroidal current enclosed by the plasma, is provided as input parameter to VMEC.

All VMEC calculations are performed in free boundary mode, where the confined region is characterized with the total enclosed magnetic toroidal flux,  $\Phi_{\text{edge}} = \Phi(s = 1)$ . Given the large input space, VMEC runs which are not relevant for W7-X, e.g. runs which did not converge or exhibit values for the plasma volume and minor radius outside the boundaries given in table 3, are discarded.

**2.2.1. Training scenarios.** To decouple the regression complexity of the 3D ideal-MHD equilibrium from the vacuum field computation, the problem is broken down in two different scenarios: a null and finite- $\langle\beta\rangle$  cases, which lead to two independent data sets,  $\mathbb{D}_{\text{config}}$  and  $\mathbb{D}_{\beta}$ .  $\mathbb{D}_{\text{config}}$  is populated with vacuum magnetic configurations, i.e. pressure and plasma current profiles are constant 0. This scenario targets two applications: discharges with low  $\langle\beta\rangle$  values which could be effectively studied with a vacuum field, and further investigations of the properties of the vacuum configurations of W7-X. In particular, the use of a slightly modified model is envisioned to further explore the richness of the vacuum magnetic configurations of W7-X, searching for optimized equilibria in terms of, e.g. neoclassical transport via the effective helical ripple amplitude  $\epsilon_{\text{eff}}$  [62] or ideal MHD stability via the magnetic well [63]. In  $\mathbb{D}_{\beta}$ , the standard magnetic configuration (EJM + 252) [30] is fixed, and the data set is populated with plasma

<sup>2</sup> A globally decreasing function of the radial profile, not to be confused with a high ‘peaking factor’ as used in the fusion community.



**Figure 1.** Subset of normalized plasma profiles included in the data set as a function of the flux radial coordinate  $s$ . Only plasma profiles which resulted in a valid VMEC equilibrium are depicted.

**Table 3.** Plasma volume and minor radius boundaries of valid VMEC runs included in the data set.

Variable	Min	Max	Unit
$V_p$	22.0	38.0	$\text{m}^3$
$a_{\text{eff}}$	45	60	cm

profiles as described in section 2.2. This scenario covers discharges with volume-averaged  $\langle\beta\rangle$  up to 5% and net toroidal current up to 10 kA.

The number of VMEC simulations for the two scenarios are 11 360 and 11 709, respectively. Of these, only 10 339 and 9675 converged. Finally, after filtering out the equilibria based on the ranges given in table 3, the data sets contain  $|\mathbb{D}_{\text{config}}| = 9589$  and  $|\mathbb{D}_{\beta}| = 9332$  valid runs, respectively.

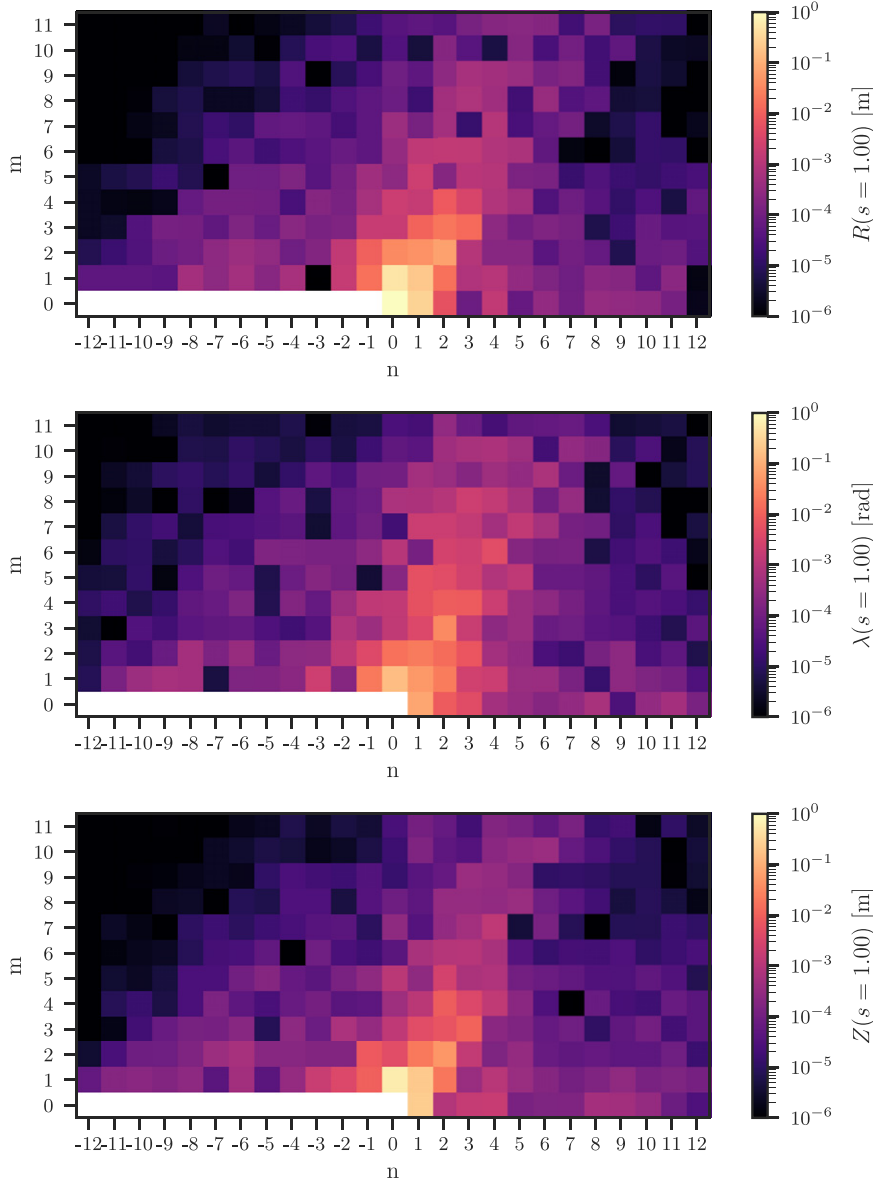
In this work, the two dimensions to characterize a W7-X magnetic configuration, the vacuum field geometry and the plasma profiles, are independently explored in  $\mathbb{D}_{\text{config}}$  and  $\mathbb{D}_{\beta}$ . Given the large vacuum magnetic configuration space probed in  $\mathbb{D}_{\text{config}}$  and the relatively low values of  $\langle\beta\rangle$  included in  $\mathbb{D}_{\beta}$ , the spread of the FCs describing the equilibrium field is expected to be higher in  $\mathbb{D}_{\text{config}}$  than in  $\mathbb{D}_{\beta}$ . Furthermore, W7-X is an optimized stellarator where the plasma influence on the magnetic configuration has been strongly reduced by the minimization of the bootstrap current and the Shafranov shift [64]. Hence, the equilibrium field coefficients are expected to be smooth functions of the main parameters characterizing the plasma,  $p_0$  and  $I_{\text{tor}}$ , in contrast to the flexibility of the vacuum magnetic configurations of W7-X. In the scope of the next steps of this proof of concept, working models in these two extreme cases can give valuable insights on the use of NNs for the regression of the equilibrium magnetic field in an arbitrary finite- $\langle\beta\rangle$  configuration.

**2.2.2. Models inputs and outputs.** In  $\mathbb{D}_{\text{config}}$ , the inputs are represented by  $\Phi_{\text{edge}}$  and the six independent coil current ratios, while in  $\mathbb{D}_{\beta}$ ,  $\Phi_{\text{edge}}$ ,  $p_0$ ,  $I_{\text{tor}}$ , and the normalized pressure and toroidal current profiles are used. In both scenarios the regressed outputs are the iota profile,  $\iota(s)$ , the Fourier series of the flux surface coordinates, represented by  $R_{mn}(s)$ ,  $\lambda_{mn}(s)$

and  $Z_{mn}(s)$  and the Fourier series of magnetic field strength,  $B_{mn}(s)$ . The output FCs are regressed instead of the real space values for the following reasons: first and foremost, the Fourier series profiles are a compressed representation of the magnetic field, thus letting the network learn a reduced number of independent outputs. Furthermore, we seek to replace VMEC with similar input and output signature as the original code such that our application can serve as a drop-in replacement for existing use cases. For example, in the context of the application of this work in the inference of plasma parameters, the flux surface coordinates are needed to map real space diagnostic measurements to flux coordinates [54, 65], and the magnetic field strength plays a crucial role in the analysis of many diagnostics (e.g. electron cyclotron emission [26]).

For the generation of the data set, the resolution of the VMEC output is set to  $N_s = 99$  flux surfaces and  $m_{\text{pol}} = n_{\text{tor}} = 12$ , where  $|m| < m_{\text{pol}}$  and  $|n| \leq n_{\text{tor}}$  are the poloidal and toroidal Fourier modes respectively. Since all the outputs are real quantities,  $o_{mn} = (o_{-m,-n})^*$  for  $o \in \{R, \lambda, Z, B\}$ . This limits the independent FCs to a subplane (usually  $m \geq 0$ ). Despite this symmetry consideration, still 28 512 coefficients remain per output<sup>3</sup>. Figure 2 shows the FCs of the three coordinates for one sample in the data set, evaluated at the LCFS. However, it has been argued that  $m_{\text{pol}} = n_{\text{tor}} = 6$  modes are sufficient to represent the magnetic field in case of W7-X configurations [37]. In this work, the sufficient Fourier resolution is further investigated. For the radial profile, a subset of  $\hat{N}_s$  flux surfaces is selected, while up to  $\hat{m}_{\text{pol}}$  and  $\hat{n}_{\text{tor}}$  poloidal and toroidal modes are used for the FCs. To more densely cover the plasma region near the axis, the flux surfaces are selected such that their radial locations  $s$  follow a quadratic progression in  $[0, 1]$ . To compute the loss of information due to the downscaling, the reduced representation is upscaled to match the full resolution by asserting  $x_{mn} = 0$  for  $x \in \{R, \lambda, Z\}$  if  $m \geq \hat{m}_{\text{pol}}$  or  $|n| > \hat{n}_{\text{tor}}$ . Then, the outputs  $R$ ,  $\lambda$ ,  $Z$  and  $B$  are evaluated with equations (4)–(6) and (13) on a grid along the  $\theta$  and  $\varphi$  angles, using  $N_{\theta} = 18$  poloidal and  $N_{\varphi} = 9$  toroidal points per period. Finally, the full radial resolution is recovered

<sup>3</sup> The number of FCs per coordinates scales as  $\mathcal{O}(N_s \cdot m_{\text{pol}} \cdot n_{\text{tor}})$ .



**Figure 2.** FCs of the cylindrical coordinates evaluated at the LCFS. The Fourier series have poloidal modes  $m < m_{\text{pol}}$  and toroidal modes  $|n| \leq n_{\text{tor}}$ . A logarithmic colormap is used to show the span in orders of magnitude expressed by the data.

by cubic interpolation along  $s$ . Similarly, a reduced resolution of the iota profile is investigated, using  $\hat{N}_s$  flux surfaces (the same as those employed for  $\vec{x}$  and  $B$ ). To compare the reduced to the full resolution, the iota profile is then upscaled via cubic interpolation.

Given a set  $\mathcal{Y} = \{y \in \mathbb{R}^K\}$  of generic quantities  $y$  with true or reference value  $y^*$ , the root-mean-square error (rmse) between  $y$  and  $y^*$  is computed as

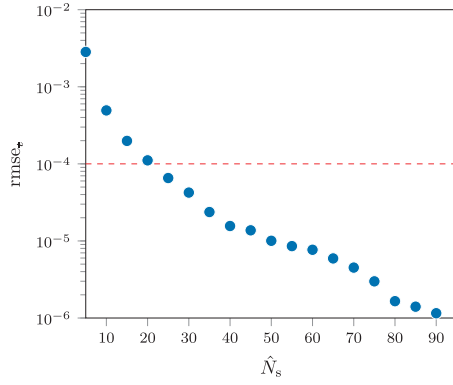
$$\text{rmse}_y = \frac{1}{K} \sum_{k=1}^K \sqrt{\frac{1}{|\mathcal{Y}|} \sum_{i=1}^{|\mathcal{Y}|} (y_{ki} - y_{ki}^*)^2}. \quad (19)$$

Here it is used to compare the two resolutions, where for each output,  $y$  is the reduced output representation of  $y^*$ , and  $K$  is the number of evaluation points:  $K_e = \hat{N}_s$ , and  $K_R = K_\lambda =$

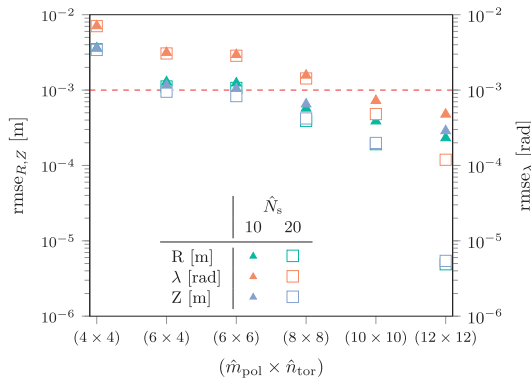
$K_Z = K_B = \hat{N}_s N_\theta N_\varphi$ . Figure 3 shows the rmse for different values of the resolution parameters.

In case of the iota profile,  $\hat{N}_s = 20$  flux surfaces are sufficient for a deviation of approximately  $\text{rmse}_e^* = 10^{-4}$ ,  $\hat{N}_s = 10$ ,  $\hat{m}_{\text{pol}} = 6$  and  $\hat{n}_{\text{tor}} = 4$  are needed for the flux surfaces coordinates to achieve  $\text{rmse}_{R,Z}^* = 10^{-3}$  m and  $\text{rmse}_\lambda^* = 10^{-3}$  rad, and  $\hat{N}_s = 10$ ,  $\hat{m}_{\text{pol}} = 6$  and  $\hat{n}_{\text{tor}} = 12$  are used for  $B$  to obtain  $\text{rmse}_B^* = 10^{-3}$  T. These choices result in 20 locations for the iota profile, while 1500 FCs describe the flux surface coordinates and the magnetic field strength<sup>4</sup>. This resolution represents a practical trade-off between the complexity of the regression task and the reconstruction fidelity. It is important to note that  $\text{rmse}_e^*$ ,  $\text{rmse}_{R,Z}^*$ ,  $\text{rmse}_\lambda^*$  and  $\text{rmse}_B^*$  represent a lower

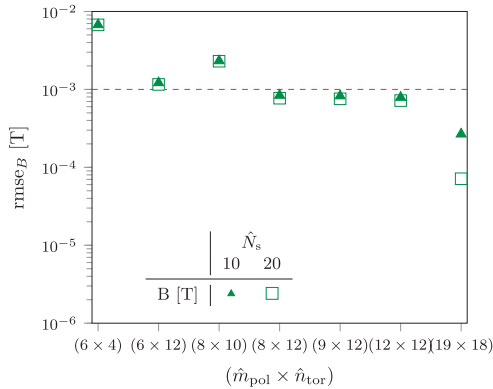
<sup>4</sup> Each output is described by  $N_{\text{FCs}} = N_o \hat{N}_s [\hat{m}_{\text{pol}}(2\hat{n}_{\text{tor}} + 1) - \hat{n}_{\text{tor}}]$  FCs, where  $N_o$  is the output dimension ( $N_o = 3$  for  $\vec{x}$  and  $N_o = 1$  for  $B$ ).



(a) rmse of the reduced representation of the iota profile, for different radial resolutions. The reference value  $\text{rmse}_i^* = 10^{-4}$  is marked by the dashed line.



(b) rmse of the reduced representation of the flux surfaces coordinates, for different radial and Fourier resolutions. The dashed line marks the reference values  $\text{rmse}_{R,Z} = 10^{-3}$  m and  $\text{rmse}_{\lambda} = 10^{-3}$  rad.



(c) rmse of the reduced representation of magnetic field strength, for different radial and Fourier resolutions. The dashed line marks the reference values  $\text{rmse}_B = 10^{-3}$  T.

**Figure 3.** Analysis of the rmse between the full and reduced representation of the iota profile, flux surface coordinates and magnetic field strength. In figures 3(b) and (c) the truncated Fourier resolutions are ordered based on the total number of FCs used, which scales as  $\mathcal{O}(\hat{m}_{\text{pol}} \times \hat{n}_{\text{tor}})$ . In the case of the flux surface coordinates and magnetic field strength, and for the Fourier truncated resolution of interest (i.e.  $\hat{m}_{\text{pol}} \approx 6$  and  $\hat{n}_{\text{tor}} \approx 6$ ), an increased radial resolution of 20 flux surfaces does not significantly differ from using only 10 flux surfaces.

bound of the reconstruction error that can be achieved by using the models presented in this work.

Given the two data sets,  $\mathbb{D}_{\text{config}}$  and  $\mathbb{D}_{\beta}$ , and the three output quantities,  $\iota$ ,  $\vec{x}$  and  $B$ , six independent regression tasks are defined: *config-iota* and *beta-iota*, *config-surfaces* and *beta-surfaces*, and *config-B* and *beta-B*. In the *config-iota* and *beta-iota* tasks, an NN is trained to compute the reduced resolution iota profile, using respectively  $\mathbb{D}_{\text{config}}$  and  $\mathbb{D}_{\beta}$  as data set. Similarly, in the *config-surfaces*, *beta-surfaces*, *config-B* and *beta-B* tasks, an NN is trained to compute the FCs of the reduced resolution  $\vec{x}$  or magnetic field strength  $B$ , using  $\mathbb{D}_{\text{config}}$  and  $\mathbb{D}_{\beta}$ , respectively.

Considering the scope of this paper which attempts to develop a VMEC proof-of-concept surrogate model, it is useful to investigate the performance on independent subproblems. In future works, a single NN could be trained to compute all outputs and to handle both vacuum and finite- $\langle\beta\rangle$  runs.

### 2.3. NN architectures

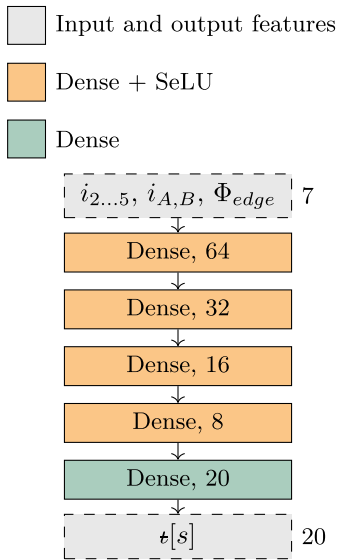
In general, given two quantities  $\vec{\psi} \in \mathbb{R}^K$  and  $\vec{\gamma} \in \mathbb{R}^D$ , and a set of  $N$  observations  $(\vec{\psi}_i, \vec{\gamma}_i)$  sampled from a fixed but unknown distribution  $p(\vec{\psi}, \vec{\gamma})$ , an NN, parameterized with a set of free parameters  $\vec{w}$ , can be employed to learn a mapping  $\tilde{f}: \mathbb{R}^K \rightarrow \mathbb{R}^D$  which minimizes the empirical loss  $\frac{1}{N} \sum_i l(\vec{\gamma}_i, \tilde{f}(\vec{\psi}_i; \vec{w}))$ , where  $l: \mathbb{R}^D \times \mathbb{R}^D \rightarrow \mathbb{R}$  is a given loss function. In this work, the expressive power of NNs is exploited to learn a low-cost approximation of a known function, using observations sampled from a known distribution. An NN usually employs successive layers of *artificial neurons* to create the mapping  $\tilde{f}$ , where each neuron computes a non-linear transformation of the neurons from the previous layer. The NN free parameters  $\vec{w}$  are derived during the training process to minimize the empirical loss on the given training set. For a detailed introduction on NNs please refer to [66].

Two NN architectures are adopted herein. One is a feedforward fully-connected neural network (FF-FC), which is composed of a sequence of dense blocks, each comprising a dense layer with  $L^2$  regularization and a non-linear activation function. The number of hidden units is halved for each successive block. The activation function for the last block is the identity. Figure 4 illustrates the architecture for the *config-iota* task, where a network with five of such blocks is shown.

The FF-FC architecture is used on the iota reconstruction, where the regressed output is composed of only 20 elements. However, its number of free parameters grows linearly with the dimensionality of the output. Thus, more efficient architectures are needed for the surfaces and magnetic field strength reconstruction, where each sample has 1500 output elements. Hence, 3D CNNs [67, 68] and encoder–decoder like architectures [69–71] are explored. In these architectures an encoder processes variable-length input features and generates a fixed-length, flattened representation. Conditioned on the encoded representation, the decoder then builds the required outputs.

For these tasks the  $\vec{x}$  coordinates are stacked. Figure 5 displays an example of such architecture for the *beta-surfaces* task, where a CNN architecture with transposed convolution is used. In the *encoder* tree, high-level features are extracted

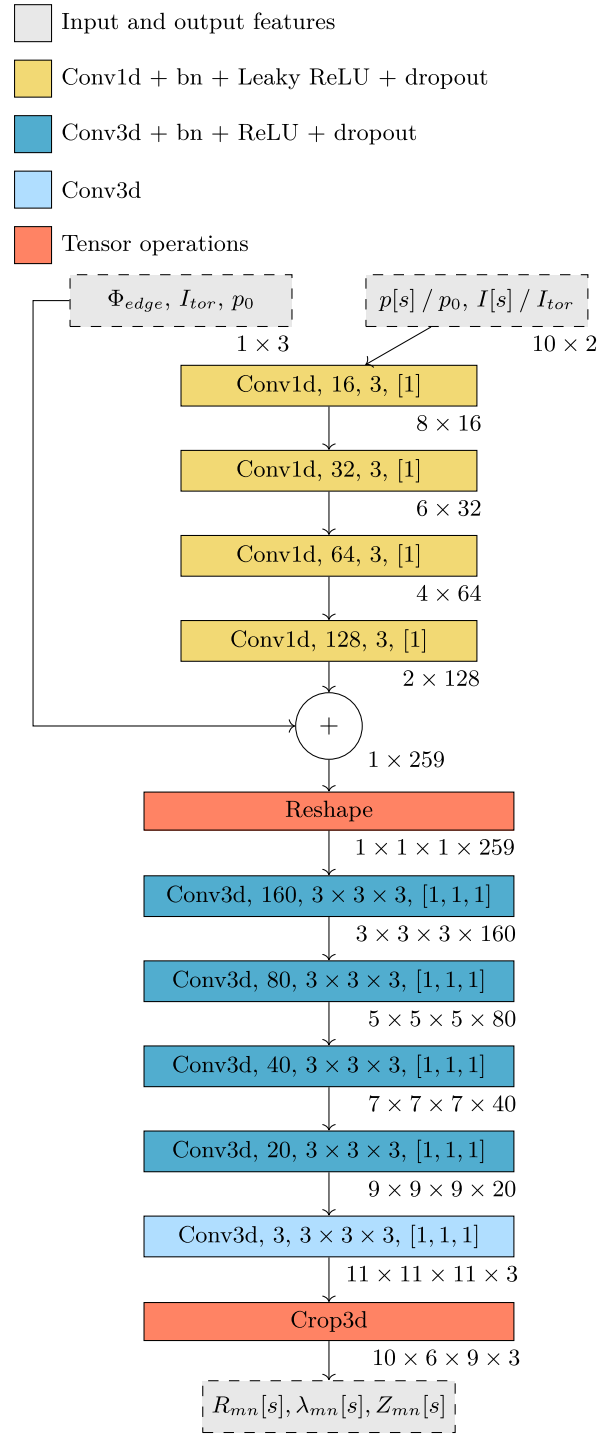




**Figure 4.** FF-FC architecture for the *config-iota* task. The gray blocks represent the input and output features, where the dimension is indicated on the right. A single block is composed of: a ‘dense,  $m$ ’ layer with  $m$  units and a non-linear activation function (e.g. the scaled exponential linear unit (SeLU)). The last block uses the identity function. The values of the HP of the best performing model in the *config-iota* task are shown here (see sections 2.4 and A.1). The use of the SeLU activation function, as discovered by HP search, leads to whitened layer input distributions, which improve the convergence of the training process [79].

from the plasma profiles via consecutive 1D convolutional blocks and concatenated back with the scalar inputs into a flattened representation. Then, a *decoder* tree gradually builds up the output via consecutive transposed convolutional blocks. Finally, the output shape is matched via a 3D cropping operation. Each *encoder* block comprises a 1D convolutional layer, batch normalization [72], and a non-linear activation function with dropout [73]. Similarly, a *decoder* block is composed of a 3D transpose convolutional layer, batch normalization, and a non-linear activation function with dropout. For the last block, batch normalization is not included and the identity activation function is used. For each block the number of filters in the *encoder* tree is doubled, while halved in the *decoder* tree. Convolutional layers with stride are employed over up-sampling operations, as suggested by [74].

The stacking of consecutive convolutional layers acting on inputs of different length scales, in conjunction with a scaling of the feature channels, is a common approach in modern deep convolutional neural network architectures. This structure decreases the number of free parameters by forcing the model to learn a hierarchical representation of high- and low-level features, while imposing a regularizing effect during training. A subset of the NN architecture HPs is not fixed *a priori*, but optimized via HP search. The lists of the explored HPs (e.g. the layer non-linear activation function) are provided in section A.1.



**Figure 5.** The 3D CNN architecture for the  $\beta$ -surfaces task. The gray blocks represent the input and output features, the yellow ones the 1D convolutions, the blue ones the 3D convolutions, and the red ones tensor operations. For each block, the output dimension is indicated on the bottom right, where the last number is always the number of features, and the antecedent ones the feature dimension (e.g.  $10 \times 6 \times 9 \times 3$  refers to 3 features of size  $10 \times 6 \times 9$ ). For the convolutional blocks, the number of filters, kernel size, and stride (in bracket) are indicated in sequence. The use of batch normalization is indicated via bn. The values of the HP of the best performing model in the  $\beta$ -surfaces task are shown here (see sections 2.4 and A.1).

**Table 4.** Main results across all learning tasks. The  $\text{nmse}$ , training and inference time mean and 95% confidence interval are evaluated with bootstrapping [91]. The inference time is conservatively estimated with a batch size of 1 on a single Intel Xeon Gold 6136 CPU. However, orders of magnitude in inference time can be gained by parallel computation, pre- and post-training optimizations (e.g. model pruning and quantization). The  $\text{nmse}_{\text{best}}$ , which refers to the error on the cross-validation fold used in HP search, is within the 95% of the  $\text{nmse}$  distribution for all tasks, meaning that the model discovered in the HP search is robust across the whole data set.

Task	$\text{nmse} (10^{-2})$	$t_{\text{train}} (10^2 \text{ s})$	$t_{\text{inference}} (10^{-3} \text{ s})$	$\text{NN}_{\text{free parameters}}$	$\text{nmse}_{\text{best}} (10^{-2})$
<i>config-iota</i>	$1.51 \pm 0.19$	$1.45 \pm 0.26$	$4.25 \pm 0.67$	3436	1.4
<i><math>\beta</math>-iota</i>	$4.77 \pm 0.50$	$3.71 \pm 0.87$	$5.51 \pm 0.80$	14 276	4.5
<i>config-surfaces</i>	$14.17 \pm 0.37$	$9.0 \pm 1.9$	$5.93 \pm 0.74$	244 989	14.4
<i><math>\beta</math>-surfaces</i>	$19.16 \pm 0.67$	$13.1 \pm 2.7$	$14.7 \pm 3.0$	1607 535	19.5
<i>config-B</i>	$3.39 \pm 0.27$	$8.2 \pm 1.4$	$7.23 \pm 0.88$	316 193	3.5
<i><math>\beta</math>-B</i>	$9.87 \pm 0.35$	$3.29 \pm 0.59$	$8.7 \pm 1.3$	541 921	10.2

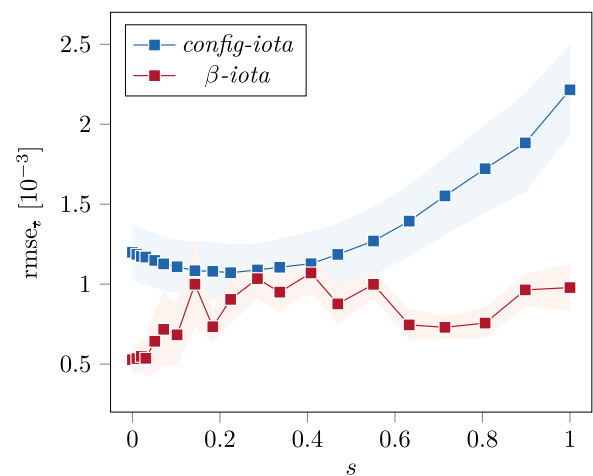
In both architectures all weights are uniformly initialized as suggested by [75], while the bias terms, where present, are initialized to zero. The weights are then optimized via the *Adam* optimizer [76], while reducing the learning rate by a fixed multiplier factor once a validation loss plateau is reached. Early stopping [77] is employed during training. The NN models are built, trained and evaluated via the open source software package *Tensorflow* [78] on a single NVIDIA RTX8000P virtual graphical processing unit (GPU).

#### 2.4. Training and evaluation pipeline

For each task defined in section 2.2, the training and evaluation pipeline includes the following steps:

**Data scaling.** It is known that NN models converge faster during training if the input distributions are whitened [80], i.e. linearly transformed to have zero mean and unit variance. All scalar inputs are mapped to  $[-1, 1]$ , while non-scalar inputs and outputs (plasma profiles,  $\epsilon$  profile and FCs) are scaled to the inter-quartile range. These steps are performed via the open source software package *Scikit-learn* [81].

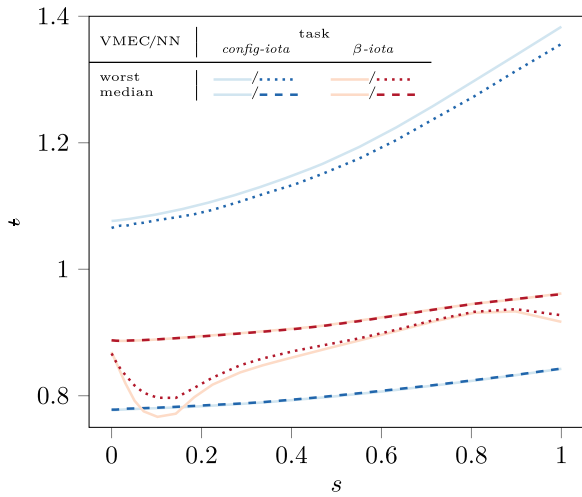
**Bayesian HPs search.** The large number of HPs and the significant training time of the considered NN architectures make a manual model optimization procedure hardly effective. Therefore, to standardize the search of more performing models, an automated approach to HPs search is used in this work. In particular, the tree-structured Parzen estimator (TPE) [82] algorithm, provided via the open source software package *hyperopt* [83], is employed. TPE is a sequential model-based optimization (SMBO) algorithm, where the true fitness function, e.g. the model training and evaluation, is approximated with a low-cost model that is cheaper to evaluate. The proxy model is then numerically optimized to retrieve new configurations to be evaluated. Contrarily to other SMBO strategies where the fitness function is directly learned, TPE models the distribution function of configuration values given classes of optimal and non-optimal fitness function values. It then optimizes the expected improvement criterion [84] with a heuristic procedure. Its main advantage over other HP search approaches is the sampling efficiency on tree-structured configuration spaces [82], i.e. spaces in which not all dimensions



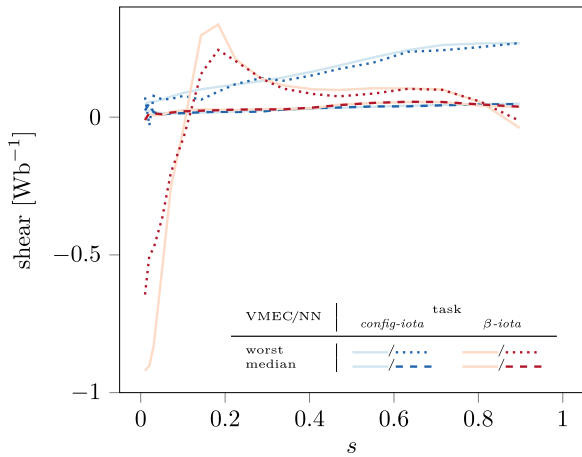
**Figure 6.** Results for the *config-iota* (blue) and  *$\beta$ -iota* (red) tasks. Lines show the  $\text{rmse}$  mean and 95% confidence interval as a function of the radial coordinate  $s$ . While the  $\text{rmse}$  in the  *$\beta$ -iota* task is generally lower than  $10^{-3}$ , the  $\text{rmse}$  in the *config-iota* scenario increases along the radial profile, following the characteristic vacuum W7-X  $\epsilon$  profile.

are well-defined for all the configurations (e.g. number of hidden units in the second layer of a single-layer FF-FC model). For a detailed description of the algorithm, please refer to [82]. On each learning task, 30 search iterations are performed. The data set is split in 20% for testing, 10% for validation and 70% for training. For each search iteration, the training data is used to train the model, while the validation data is used to assess the model regression error and inform the search strategy. The best performing model is then adopted in the cross-validation scheme. To ease the computational cost of the search, a simple mean-squared error (mse) loss is used for training and HP validation.

**Repeated k-fold cross-validation.** To estimate the regression error on out-of-sample data, a five-fold cross-validation evaluation is repeated 10 times. In a  $k$ -fold cross validation scheme [85, 86], the data set is partitioned into  $k$ -folds of equal cardinality. Then, for each fold, the training process is repeated  $k$ -times, using the selected fold as test set, and the remain-



**Figure 7.** Worst and median predicted samples for the *config-iota* (blue) and *beta-iota* (red) tasks. The solid lines represent the true  $t$  profiles as evaluated by VMEC, while the dotted (worst) and dashed (median) lines show the predicted profiles by the model. The results from the worst performing cross-validation fold are shown.



**Figure 8.** Shear profiles of the worst and median predicted samples in the *config-iota* (blue) and *beta-iota* (red) tasks (same samples as in figure 7). The solid lines represent the true shear profiles as evaluated from the VMEC  $t$  profiles, while the dotted (worst) and dashed (median) lines show the shear profiles derived from the model  $t$  predicted profiles. The results from the worst performing cross-validation fold are shown. The shear profile is computed as  $d\epsilon/d\Phi$ . Even in the case of the worst predicted samples, which feature a particular sheared profile, the  $t$  shear is qualitatively regressed.

ing folds for the training and validation sets. The estimate of the regression error is the average of the test error on each fold. However, the cross-validation estimate of the regression error can be highly variable due to the single partition of the data set into the  $k$ -folds [87]. To overcome this limitation, in the repeated  $k$ -fold cross-validation scheme, the  $k$ -fold cross-validation scheme is repeated  $n$ -times, partitioning the data set into a different  $k$ -fold each time. The average of the test error on each fold is then used as the final estimate.

### 3. Results

The results achieved on each task are now presented. It is important to remember that  $\mathbb{D}_\beta$  includes plasma profile for a fixed magnetic configuration (the standard configuration), while  $\mathbb{D}_{\text{config}}$  explores the rich space of W7-X vacuum magnetic configurations. The changes in  $\mathbb{D}_\beta$ , induced by finite-beta effects, are then small compared to those in  $\mathbb{D}_{\text{config}}$ , induced by coil currents (i.e. finite-beta effects span a space that only slightly expands the vacuum solution). Therefore, the spread of the output data in the finite-beta cases is smaller than in the vacuum scenarios: the coil system of W7-X has been designed to allow a large flexibility in the vacuum magnetic configuration space [88, 89], while the W7-X optimization explicitly targeted robustness against changes in plasma profiles, in particular pressure profiles [64, 90]. These features are expected to make the output data in the finite- $\langle\beta\rangle$  tasks more difficult to resolve because of the smaller spread. Therefore, to quantitatively compare the results across all tasks, the normalized root-mean-squared error (nrmse) is used instead.

Given  $\mathcal{Y} = \{y \in \mathbb{R}^K\}$  (see section 2.2.2), the nrmse between the predicted  $y$  and the true or reference  $y^*$  is computed as:

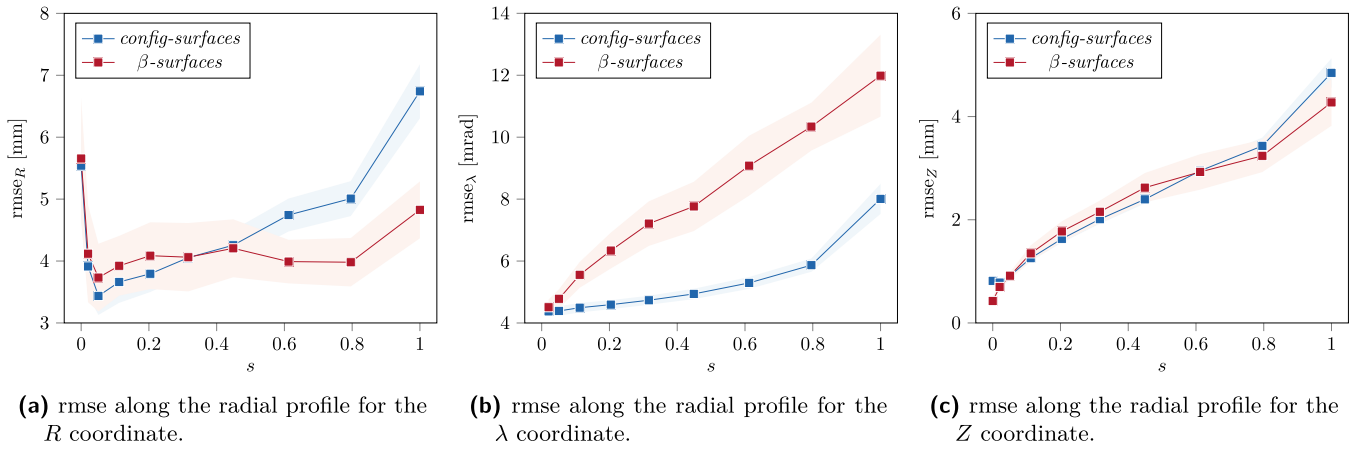
$$\text{nrmse}_y = \frac{1}{K} \sum_{k=1}^K \sqrt{\frac{\sum_{i=1}^{|\mathcal{Y}|} (y_{ki} - y_{ki}^*)^2}{\sum_{i=1}^{|\mathcal{Y}|} (y_{ki} - \bar{y}_k)^2}}, \quad (20)$$

where  $\bar{y}_k = \frac{1}{|\mathcal{Y}|} \sum_{i=1}^{|\mathcal{Y}|} y_{ki}$ . The  $t$  profile is evaluated along the radial profile with  $\hat{N}_s$  flux surfaces, so  $K_t = 20$ . As employed in section 2.2.2, an evaluation grid with  $N_\theta = 18$  and  $N_\varphi = 9$  is used for the flux surface coordinates and the magnetic field strength. The use of the nrmse allows us to aggregate the regression error on the three flux surface coordinates, and to compare the results across all outputs and scenarios.

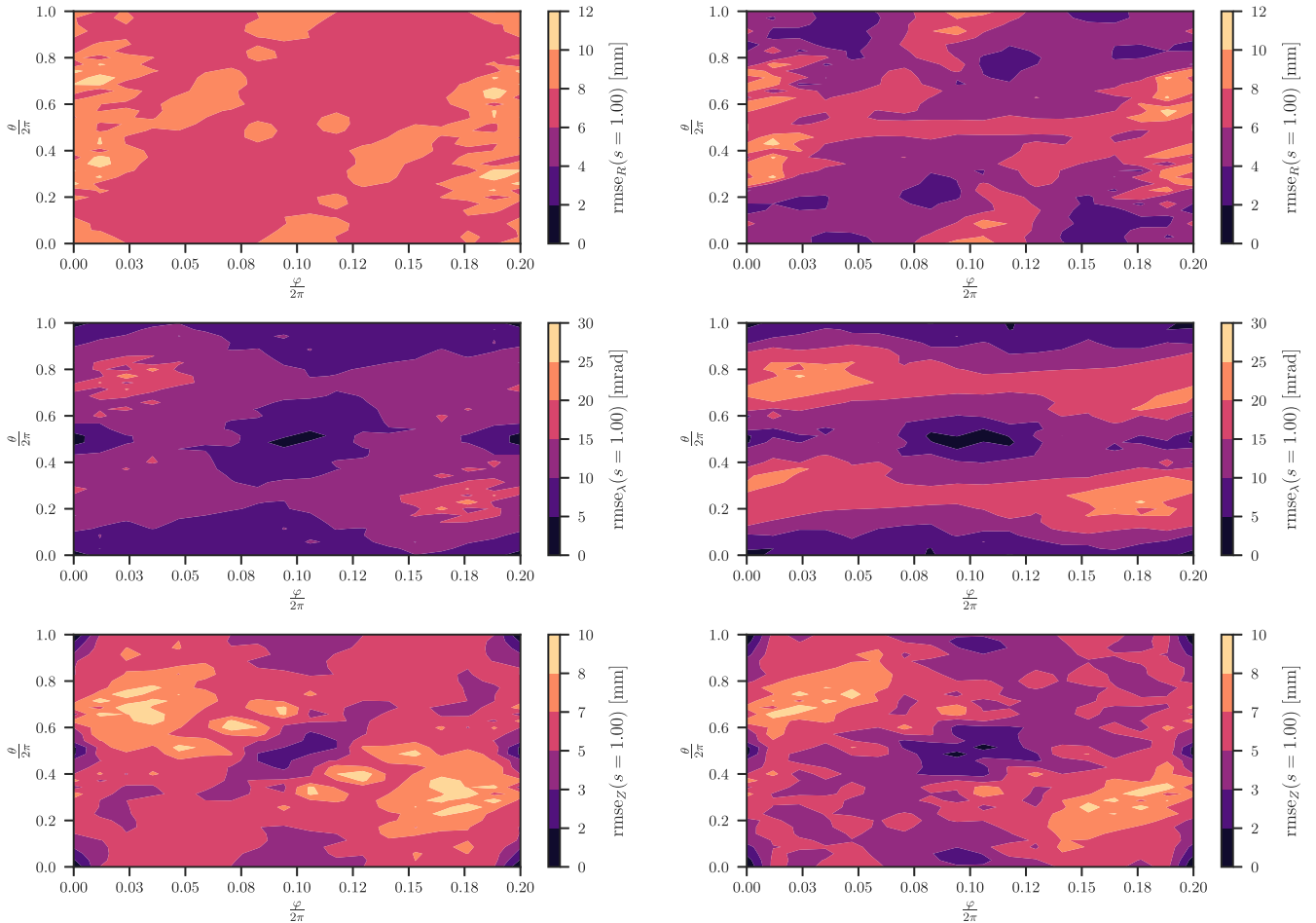
Table 4 summarizes the results for all tasks. As expected, on each output, the nrmse in the vacuum scenario is lower as compared to the finite- $\langle\beta\rangle$  case. Moreover, a nrmse below 10% is consistently achieved for the  $t$  profile and the magnetic field strength. In the flux surface coordinates tasks, nrmse values between 14% and 20% are achieved instead.

Given the relative small size of the data sets and of the NNs, the model training time is on the order of magnitude of minutes but less than an hour. More importantly, the inference time, even in the most conservative evaluation (i.e. with a single thread on 1 CPU core with a batch size of 1) is on the order of few milliseconds. However, parallel computation (e.g. batched inference and GPU deployment), pre- and post-training optimizations (e.g. model pruning and quantization), are expected to deliver consistent orders of magnitude speed-up [92, 93]. These optimizations are out of the scope of this paper.

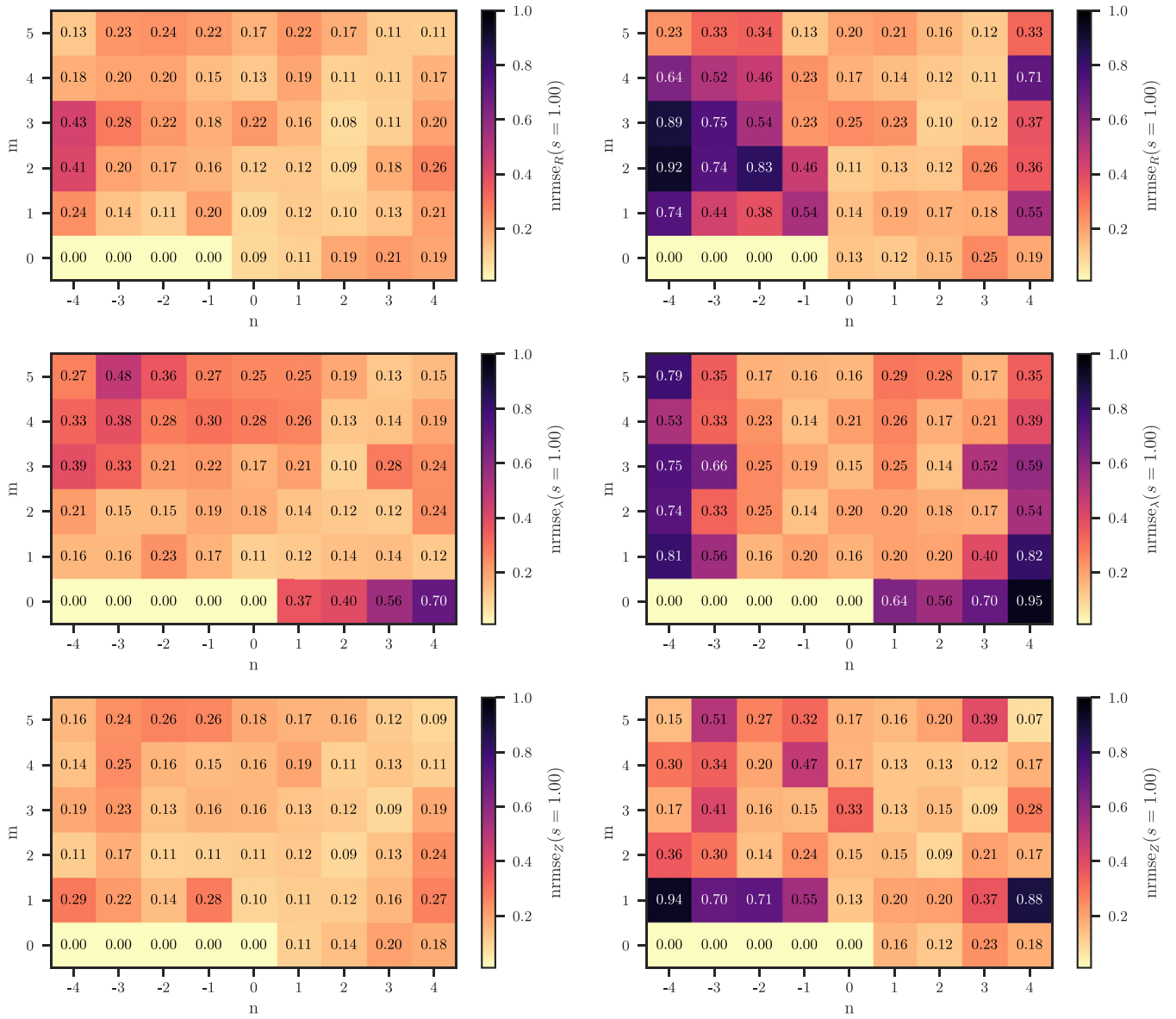
Tables 5–10 list the HP values for the best performing models discovered via HP search (see section 2.4). As reported in table 4, the nrmse obtained during search is compatible with



**Figure 9.** rmse along the radial profile for flux surface coordinates in the *config-surfaces* and *beta-surfaces* tasks. The plotted values are the poloidal and toroidal average over each flux surface. The solid lines show the mean values for the cross-validation folds, while the shaded area the 95% confidence interval. The rmse generally increases from the magnetic axis toward the edge on all tasks, apart for  $R$  near the axis.



**Figure 10.** rmse for the surfaces tasks evaluated at the LCFS on a grid with  $N_\theta = 36$  poloidal and  $N_\varphi = 18$  toroidal points per period. The results for the worst performing cross-validation fold are shown. In case of  $R$ , the bean-shape ( $\varphi \approx 0$  rad) cross section exhibits the largest regression error. While for  $\lambda$  and  $Z$ , the  $\varphi \approx 0.03$  cross section has the largest rmse.



**Figure 11.** nrmse for the regressed FCs in the *config-surfaces* and  $\beta$ -surfaces tasks. For each FC, the nrmse value is annotated at the  $(m, n)$  location. The worst performing cross validation fold is shown.

the nrmse estimated via cross-validation (i.e. its value is within the 95% interval of the distribution). This means that the HP search procedure did not overfit<sup>5</sup> to the validation data, but HP values which perform well on the whole data set were found.

In the following, the fidelity of the different NNs is inspected in closer detail and the major influences on the regression error are identified.

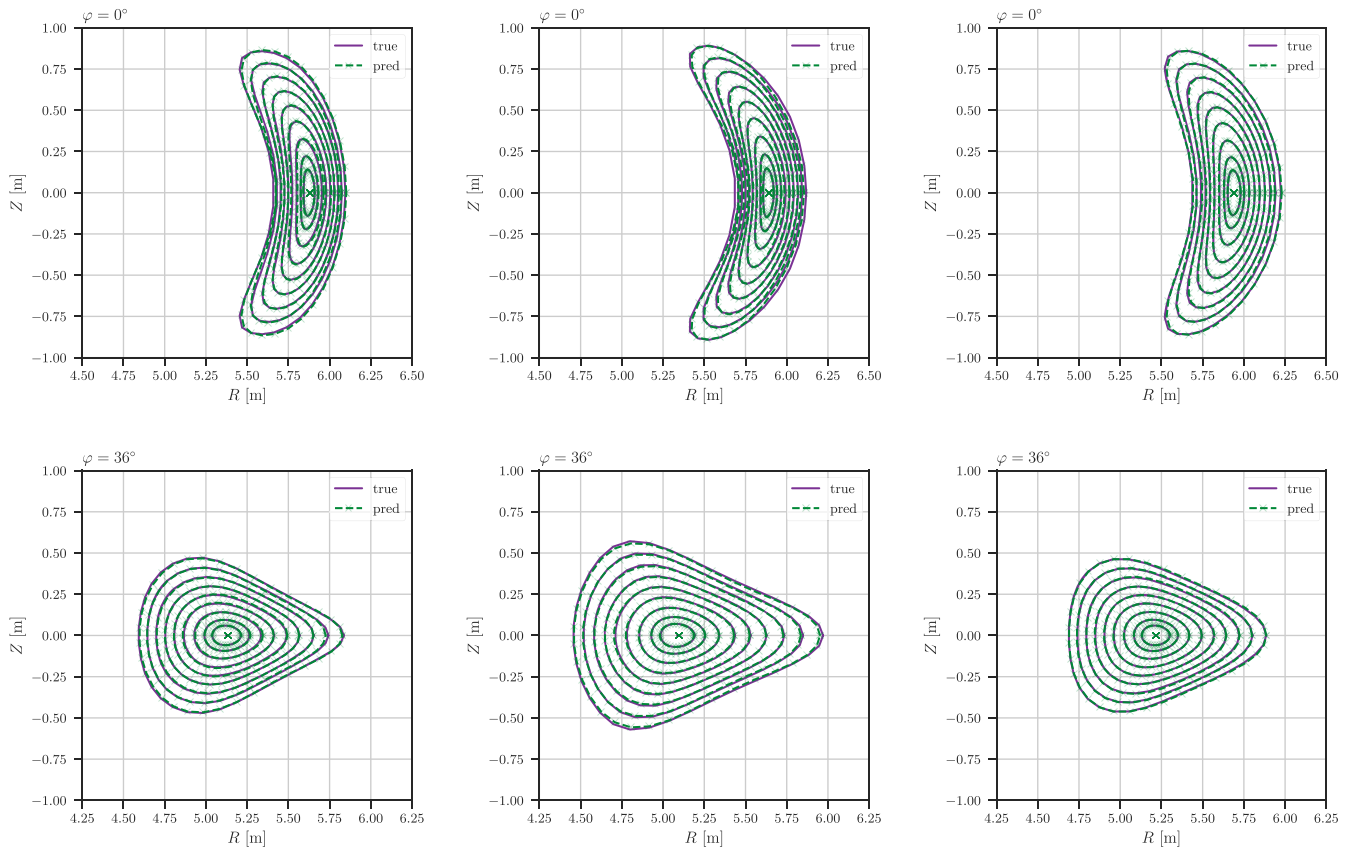
### 3.1. *iota* regression

Figure 6 shows the rmse profile along the radial flux coordinates for the *config-iota* and  $\beta$ -*iota* tasks. Although the average

nrmse in the  $\beta$ -*iota* case is higher than in the *config-iota* case, the rmse is on the order of  $10^{-3}$  for both. In the *config-iota* scenario, the rmse increases from the axis to the edge. This may be caused by the characteristic shear profile of W7-X magnetic configurations and the hence increasing spread of  $\iota$  profile in the data from the axis to the edge. Instead, in the  $\beta$ -*iota* task, the toroidal current (and partially the pressure) profile is the main parameter affecting  $\iota$ . By data set construction, these have a larger spread at mid-radius (see figure 1). The larger spread is reflected in the maximum at  $s \approx 0.4$ . In both cases, this work shows that even shallow, FF-FC NNs can effectively regress the  $\iota$  profile with high accuracy.

The qualitative fitness of the model can be visualized in figure 7, which shows the worst and median predicted  $\iota$  profiles

<sup>5</sup> High variance of the model error on unseen data.



**Figure 12.** True (pink) and predicted (green) flux surfaces for the bean-shape (upper) and triangular (bottom) cross sections on the *config-surfaces* task (■). The worst (left), median (center) and best (right) regressed samples are shown from the worst performing cross-validation fold.

for the worst performing cross-validation fold. In addition, as highlighted in figure 8, even in case of the worst predicted sample in the worst performing cross-validation fold (i.e. the worst possible scenario included in the data set), the model is still able to capture the main features of the  $\iota$  profile (e.g. the  $\iota$  shear).

### 3.2. Flux surfaces regression

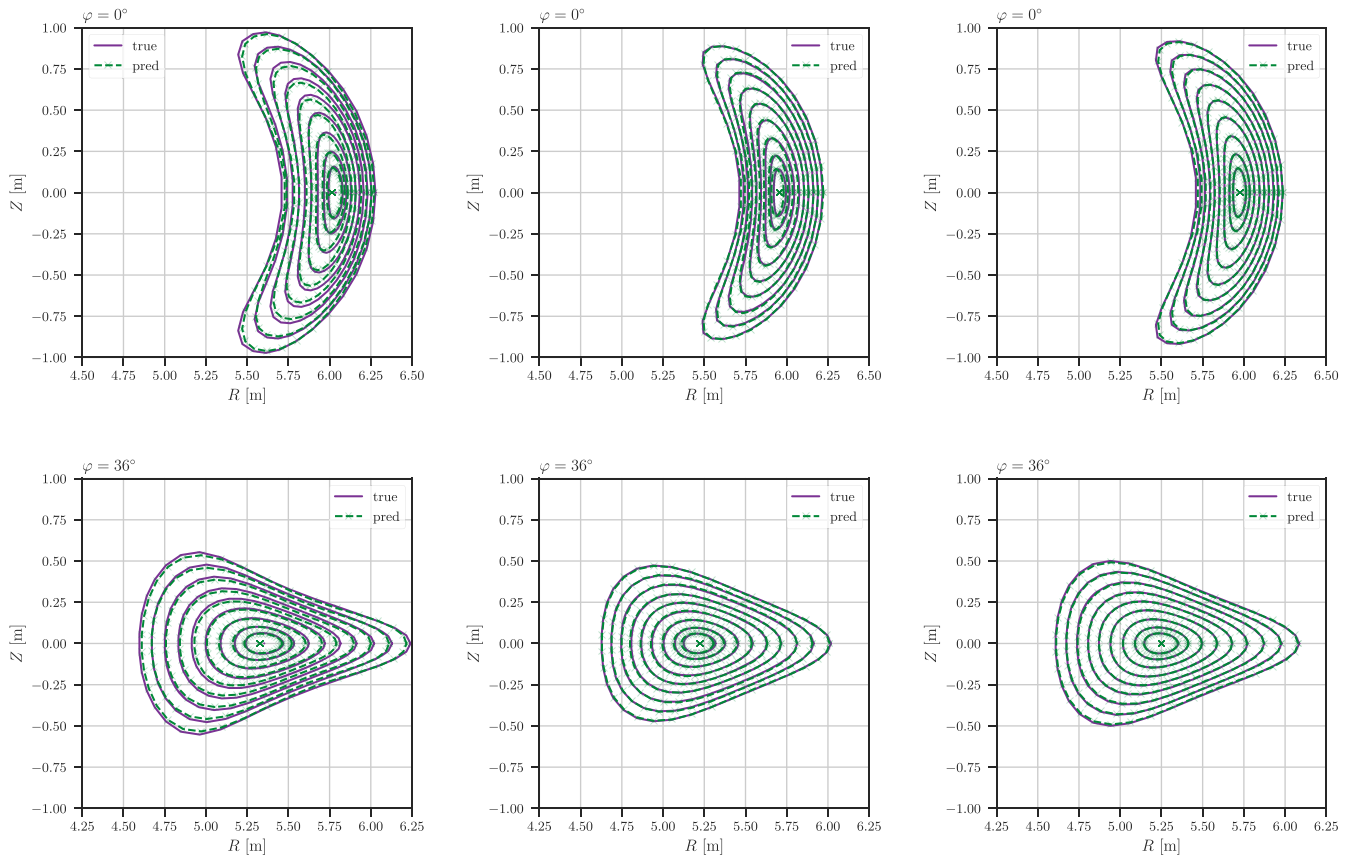
Figure 9 shows the rmse broken down by flux surface coordinate along the radial profile. The reported rmse values are the poloidal and toroidal average on each flux surface, on a grid as employed in section 2.2.2. A solid line depicts the mean on the cross-validation folds, while the shaded area represents the 95% confidence interval. An initial decreasing rmse from the magnetic axis till  $s \approx 0.1$ , a plateau, and a steep increase toward the edge can be observed in  $R$  (see figure 9(a)). Contrarily, the rmse for both  $\lambda$  and  $Z$  monotonically increases from the axis till the edge (see figures 9(b) and (c)).

In all coordinates, apart from  $R$  in the  $\beta$ -surfaces task, the rmse is higher at  $s = 1$ , i.e. the LCFS. We find it worth investigating this in more detail, and hence examine the poloidal and toroidal dependency of the rmse specifically at the LCFS with figure 10. In order to emphasize the error, the worst performing cross-validation fold is shown, and a grid with  $N_\theta = 36$  poloidal and  $N_\varphi = 18$  toroidal points per period has been used.

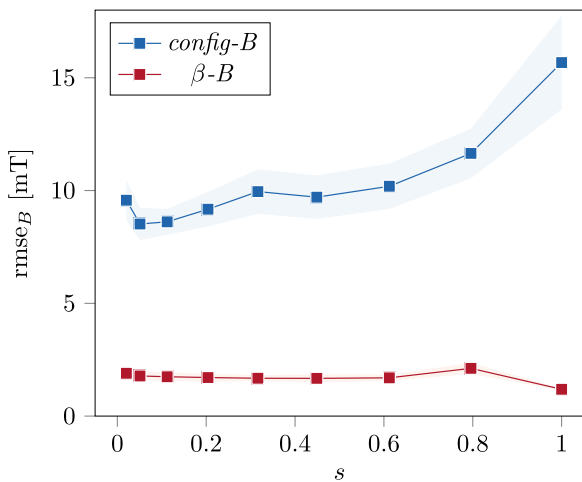
The error for  $R$  is almost flat on the surface, with maxima at  $\varphi \approx 0$  rad, representing the tips of the bean-shaped cross section. On the other hand, the error for  $Z$  and  $\lambda$  shows an  $m = 1$ ,  $n = 1$  dependency. In the *config-surfaces* scenario, at  $\frac{\varphi}{2\pi} \approx 0.03$  (and at  $\frac{\varphi}{2\pi} \approx 0.17$  following the symmetry) a higher rmse is observed. In the  $\beta$ -surfaces task, while the rmse for  $Z$  still shows a poloidal and toroidal dependency similar to that observed in the vacuum case, the dominant rmse factor for  $\lambda$  is a poloidal  $m = 1$  term.

To further investigate the rmse poloidal and toroidal dependency, figure 11 shows the regression error on the FCs of  $(R, \lambda, Z)$  evaluated at the LCFS. In this figure, to effectively compare the error on both low-order and high-order modes (see figure 2), the nrmse is used. Again, the worst performing cross-validation fold is shown. It is important to note that the FCs are the actual quantities which the NN learned. In both cases, the leading FCs are regressed with a nrmse below 20%. However, there are some regions in the  $(m, n)$  space which the model struggles to reconstruct, in particular in the  $\beta$ -surfaces task (see figure 11(b)).

The regression performance is visualized in figures 12 and 13, where the true and regressed flux surfaces at the bean-shape ( $\varphi = 0$  rad) and triangular ( $\frac{\varphi}{2\pi} = 0.1$ ) cross sections are represented. Worst (left), median (center), and best (right) regressed samples from the worst performing cross-validation fold are shown. The LCFS shows the largest



**Figure 13.** True (pink) and predicted (green) flux surfaces for the bean-shape (upper) and triangular (bottom) cross sections on the  $\beta$ -surfaces task (■). The worst (left), median (center) and best (right) regressed samples are shown from the worst performing cross-validation fold.



**Figure 14.** rmse along the radial profile for the *config-B* and  $\beta$ -*B* tasks. Lines show the rmse mean and 95% confidence interval. The rmse in the  $\beta$ -*B* task is considerable lower than in *config-B* due to the smaller spread of  $B$  in the data set.

inconsistency (as already observed in figure 9), and in particular the  $R$  coordinate of the high-field side of the bean-shaped cross section (i.e.  $\theta = \pi$ ) appears to be the most complicated feature to resolve (as previously observed in figure 11).

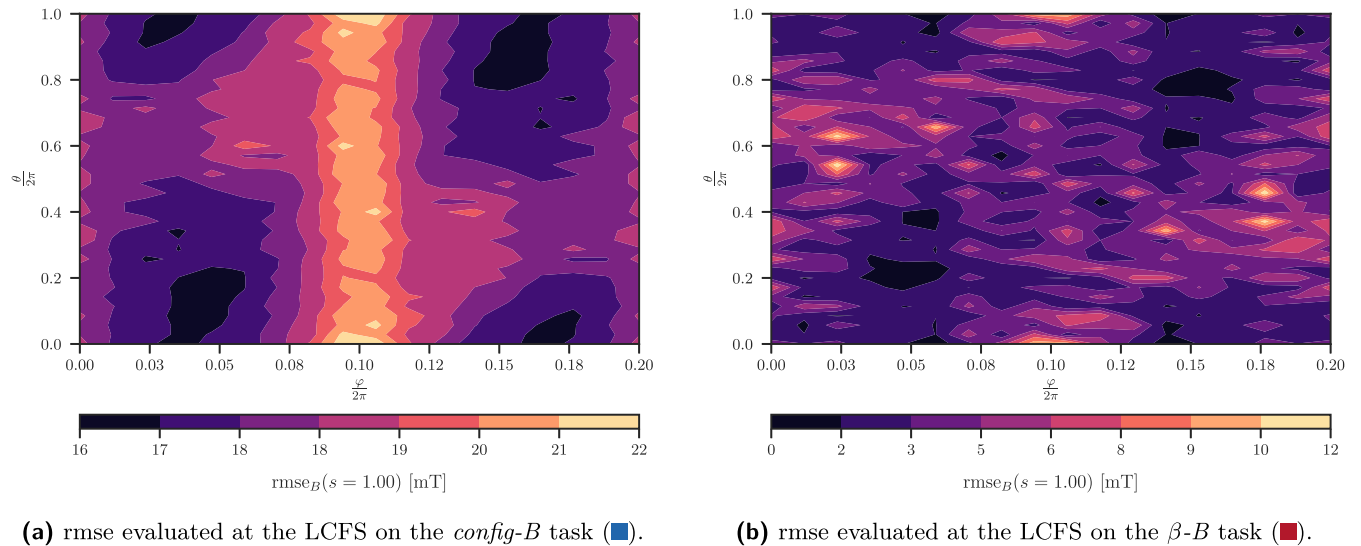
Of the three flux surface coordinates,  $\lambda$  is the most arduous to reconstruct. Although not needed to compute the

location of the flux surfaces, it gives information on the direction of the magnetic field lines. In particular, the  $\lambda_{0n}$  FCs are hardly regressed in both flux surface tasks. Earlier works have encountered similar challenges and the lack of spectral minimization for  $\lambda$  in VMEC is presumed to cause such difficulties [37].

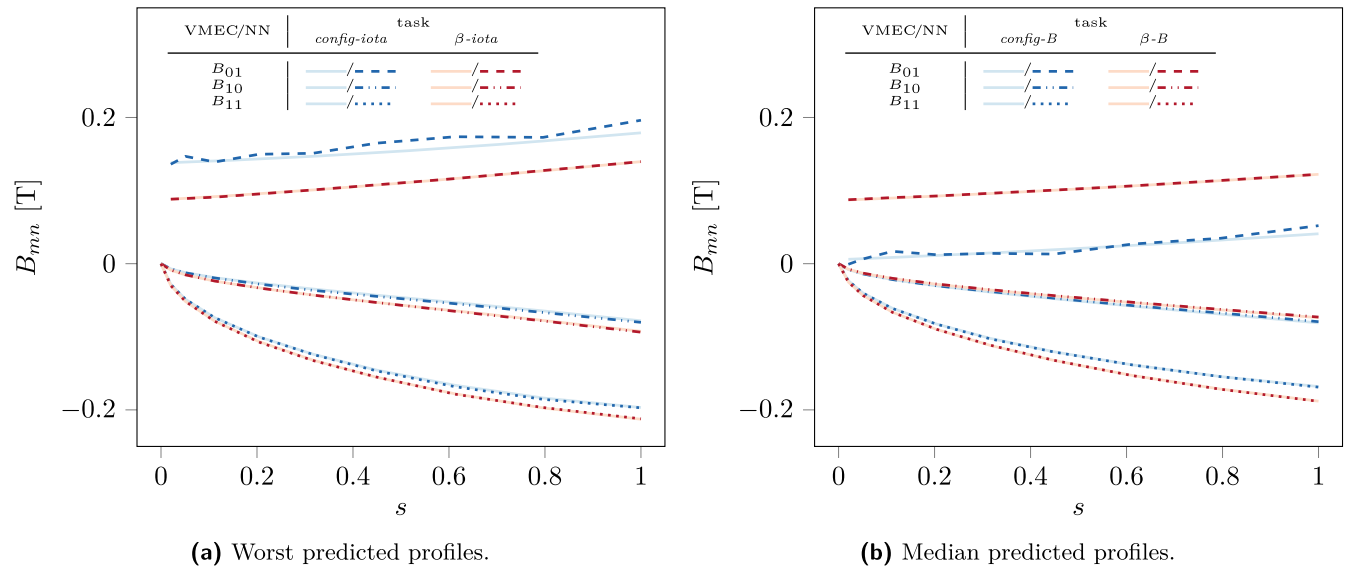
### 3.3. Magnetic field strength regression

The variance of the magnetic field strength contained in the vacuum and finite- $\langle\beta\rangle$  scenario data sets are on different orders of magnitude. In *config-B*, the magnetic field strength exhibits an average spread of  $\sigma_{\text{config-B}} = 252$  mT, while in  $\beta$ -*B*, the spread is only  $\sigma_{\beta-B} = 28$  mT. This mainly derives from the rich vacuum magnetic configuration space of W7-X and the low impact of pressure and toroidal current on the equilibrium field [89]. Therefore, the magnetic field strength in the  $\beta$ -*B* task is more difficult to resolve. Indeed, even though the achieved nrmse for the  $\beta$ -*B* task is higher than in *config-B*, the rmse in  $\beta$ -*B* is considerably lower than in *config-B*, as figure 14 shows, due to the smaller spread in the data set. Additionally, in  $\beta$ -*B*, the rmse does not seem to have any radial dependency, while the regression error increases from the magnetic axis toward the edge for *config-B*.

The topology of the regression error of the magnetic field strength at the LCFS, for the worst performing cross-validation fold, is visualized in figure 15. In the *config-B* task, in addition to a non-zero baseline error (i.e.  $m = 0$  and  $n = 0$ ),  $n = 1$



**Figure 15.** rmse for the magnetic field strength tasks evaluated at the LCFS on a grid with  $N_\theta = 36$  poloidal and  $N_\varphi = 18$  toroidal points per period. The worst performing cross-validation fold is shown. In *config-B*, the errors on the  $B_{00}$ ,  $B_{01}$  and  $B_{10}$  terms are the major contributors to the rmse. In  $\beta$ -*B* instead, the rmse is almost flat with a shallow, high-order structure.



**Figure 16.** The solid lines represent the true  $B_{mn}$  profiles as evaluated by VMEC, while the marks show the profiles as predicted by the model. The results from the *config-B* (blue) and  $\beta$ -*B* (red) tasks, and the worst performing cross-validation fold are shown.

toroidal and  $m = 1$  poloidal terms are visible. This stems from the fact that the main FCs of W7-X, besides  $B_{00}$ , are  $B_{01}$  and  $B_{10}$ , while in general the other  $B_{mn}$  are much smaller [89]. Contrarily, in the  $\beta$ -*B* task, the rmse surface is more indented and higher  $B_{mn}$  error terms become the dominant influence on the regression error.

Figure 16 qualitatively captures the regression of the leading FCs, where the true and predicted FC profiles are plotted in case of the worst and median samples. The worst performing cross-validation fold is shown. As observed in figure 15,  $B_{01}$  shows the largest discrepancy.

#### 4. Summary and outlook

This paper investigates the feasibility of building a fast surrogate NN model of the MHD equilibrium code VMEC in W7X magnetic configurations. It extends earlier works [36, 37] by using physics constrained plasma profiles, modern NN architectures and workflows, and by employing single models to reconstruct multiple output quantities. The decomposition of the problem into a vacuum and finite- $\langle\beta\rangle$  data set allows the independent study of the two limiting cases, of which the viability is necessary for a future VMEC surrogate model.



The reconstruction of the  $\iota$  profile shows a nrmse between 1% and 5%. Regression of flux surface coordinates  $(R, \lambda, Z)$  gives nrmse values between 14% and 20%, where the  $\lambda$  coordinate appears to be the most problematic to regress. For the magnetic field strength  $B$ , nrmse values between 3% and 10% are obtained. In almost all outputs, the regression error increases from the magnetic axis toward the edge. As expected, the regression of the finite- $\langle\beta\rangle$  samples proves to be more challenging than the vacuum cases. However, the observed rmse values were often similar for the two scenarios. Limited to the investigated scenarios, a relatively small data set (e.g. 10 k samples) seems to be adequate.

The promising results of this paper show that NNs can be used to deploy a drop-in surrogate model for VMEC, although additional questions have to be investigated. First, the performance of such models to resolve both the vacuum magnetic configuration and the finite-beta effects on the equilibrium magnetic field has to be assessed by using a data set which comprises both vacuum and finite- $\langle\beta\rangle$  samples. Second, to define a quantitative required accuracy for the models, which strongly depends on the target application, the degree to which physics quantities of interest, such as MHD stability or neoclassical transport rates, are faithfully reproduced has to be characterized. This verification represents a key metric to gauge the use of NN models to provide fast, yet physics-preserving, MHD equilibria.

Given such unexplored application, several paths can still be investigated. First, multiple output quantities (e.g.  $\iota$  and  $\bar{x}$ ) can be regressed at once with a single model, thus exploiting the correlation between those quantities. Second, to obtain self-consistent equilibrium magnetic fields and flux surfaces geometries, the magnetic field strength  $B$  could be computed directly from the model's  $\iota$  and  $\bar{x}$  instead of being regressed (see equations (8)–(12)). Third, domain knowledge and physics constraints could be embedded in both NN architecture and training process [95], and the coil system geometry could be extended to a generic device geometry, thus opening up the possibility to use such a surrogate model in a generic stellarator optimization workflow. Fourth, to reduce the dimensionality of the problem, the radial dependency of the output quantities could be cast as an additional predictor, also gaining the ability to compute analytical derivatives with respect to the radial coordinate.

Furthermore, broader HP search and an ensemble of NNs could further improve the performance over single base learner [94], and optimization techniques, such as pruning and quantization, are expected to deliver improved inference times. Moreover, the results of this work suggest that the full Fourier resolution is in reach if larger NNs and longer training time are accessible.

Finally, the use of MHD fast surrogate models can impact multiple applications: fast Bayesian inference of plasma parameters and equilibrium reconstruction workflows for

**Table 5.** Hyper-parameters values for the best FF-FC model on the *config-iota* task. The float type hyper-parameters are reported with two significant digits.

HP	Value
Dense layers	4
First layer hidden units	64
Activation function	SeLU
Batch size	64
Learning rate	$6.50 \times 10^{-4}$
Learning rate decay rate	$4.00 \times 10^{-1}$
Learning rate decay steps	20
$L^2$ regularization factor	$1.60 \times 10^{-5}$
Early stopping patience epochs	40

**Table 6.** Hyper-parameters values for the best FF-FC model on the  *$\beta$ -iota* task. The float type hyper-parameters are reported with two significant digits.

HP	Value
Dense layers	4
First layer hidden units	128
Activation function	SeLU
Batch size	32
Learning rate	$2.60 \times 10^{-3}$
Learning rate decay rate	$3.40 \times 10^{-1}$
Learning rate decay steps	20
$L^2$ regularization factor	$1.60 \times 10^{-5}$
Early stopping patience epochs	40

intra-shot analysis<sup>6</sup>, access to large and rich optimization spaces for present and future magnetic confinement devices, milliseconds-range MHD equilibrium computations for real-time plasma control, and the generation of very large data sets of equilibrium computations necessary to investigate machine learning control strategies (e.g. RL)<sup>7</sup>.

## Acknowledgments

We wish to acknowledge the helpful discussions on VMEC and MHD equilibrium with J Geiger. Furthermore, we are indebted to the communities behind the multiple open-source software packages on which this work depends. The data sets

<sup>6</sup> If target physics quantities are not adequately reproduced by the surrogate model, a two-stage approach should be pursued: employ the model predictions to extensively explore the target input space, then, switch to a high-fidelity equilibrium computation to refine the solution. This may be applied to provide fast transformations for diagnostics, a broad exploration of the posterior probability distribution in a Bayesian framework, or good initial configurations for a more rapid convergence of equilibrium codes.

<sup>7</sup> It is important to note that when a sufficiently large data set is accessible, given the relative low training time, the proposed NN models could be trained to target specific data distributions expected for a use case application, thus reducing the covariate shift between the training and test set.

**Table 7.** Hyper-parameters values for the best 3D CNN model on the *config-surfaces* task. The float type hyper-parameters are reported with two significant digits.

HP	Value
Decoder layers	5
Decoder first layer filters	112
Decoder kernel size	$3 \times 3 \times 3$
Decoder stride	$1 \times 1 \times 1$
Decoder activation function	SeLU
Decoder dropout	$2.18 \times 10^{-2}$
Batch size	32
Learning rate	$8.90 \times 10^{-4}$
Learning rate decay rate	$2.56 \times 10^{-1}$
Learning rate decay steps	15
$L^2$ regularization factor	$3.35 \times 10^{-6}$
Early stopping patience epochs	35

**Table 8.** Hyper-parameters values for the best 3D CNN model on the  $\beta$ -*surfaces* task. The float type hyper-parameters are reported with two significant digits.

HP	Value
Encoder layers	4
Encoder first layer filters	16
Encoder kernel size	$3 \times 3 \times 3$
Encoder stride	$1 \times 1 \times 1$
Encoder activation function	Leaky ReLU
Encoder dropout	$4.80 \times 10^{-1}$
Decoder layers	5
Decoder first layer filters	160
Decoder kernel size	$3 \times 3 \times 3$
Decoder stride	$1 \times 1 \times 1$
Decoder activation function	ReLU
Decoder dropout	$9.90 \times 10^{-2}$
Batch size	32
Learning rate	$1.00 \times 10^{-3}$
Learning rate decay rate	$3.50 \times 10^{-1}$
Learning rate decay steps	10
$L^2$ regularization factor	$1.50 \times 10^{-4}$
Early stopping patience epochs	45

were generated on the MPCDF cluster ‘DRACO’, Germany. Financial support by the European Social Fund (ID: ESF/14-BM-A55-0007/19) and the Ministry of Education, Science and Culture of Mecklenburg-Vorpommern, Germany via project ‘NEISS’ is gratefully acknowledged. This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training program 2014–2018 and 2019–2020 under Grant Agreement No. 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

### Author statement

The contributions to this paper are described using the CRediT taxonomy [96]:

**Andrea Merlo** conceptualization, ideas, data curation, formal

**Table 9.** Hyper-parameters values for the best 3D CNN model on the *config-B* task. The float type hyper-parameters are reported with two significant digits.

HP	Value
Decoder layers	3
Decoder first layer filters	64
Decoder kernel size	$5 \times 5 \times 5$
Decoder stride	$1 \times 1 \times 2$
Decoder activation function	ReLU
Decoder dropout	$2.80 \times 10^{-4}$
Batch size	32
Learning rate	$5.20 \times 10^{-4}$
Learning rate decay rate	$3.40 \times 10^{-1}$
Learning rate decay steps	25
$L^2$ regularization factor	$8.20 \times 10^{-5}$
Early stopping patience epochs	35

**Table 10.** Hyper-parameters values for the best 3D CNN model on the  $\beta$ -*B* task. The float type hyper-parameters are reported with two significant digits.

HP	Value
Encoder layers	2
Encoder first layer filters	16
Encoder kernel size	$3 \times 3 \times 3$
Encoder stride	$2 \times 2 \times 2$
Encoder activation function	Leaky ReLU
Encoder dropout	$2.16 \times 10^{-1}$
Decoder layers	3
Decoder first layer filters	64
Decoder kernel size	$5 \times 5 \times 5$
Decoder stride	$1 \times 1 \times 2$
Decoder activation function	ReLU
Decoder dropout	$2.10 \times 10^{-2}$
Batch size	96
Learning rate	$9.50 \times 10^{-4}$
Learning rate decay rate	$3.28 \times 10^{-1}$
Learning rate decay steps	15
$L^2$ regularization factor	$4.60 \times 10^{-4}$
Early stopping patience epochs	45

analysis, investigation, methodology, software, visualization, writing—original draft preparation, writing—review & editing.

**Daniel Böckenhoff** ideas, methodology, software, supervision, validation, writing—original draft preparation, writing—review & editing.

**Jonathan schilling** data curation, methodology, software, writing—review & editing. **Udo Höfel** ideas, methodology, software. **Sehyun Kwak** ideas, methodology, software, writing—review & editing. **Jakob Svensson** ideas, methodology, software.

**Andrea Pavone** ideas, formal analysis, methodology, software.

**Samuel Aaron Lazerson** Supervision, writing—review & editing.

**Thomas Sunn Pedersen** conceptualization, writing—review & editing, Supervision.

## Data and code availability








The data sets and code needed to reproduce this work are available at <https://gitlab.com/amerlo94/vmecfastsurrogate>.

## Appendix A

### A.1. Hyper-parameters values

Tables 5–10 report the HP values of the best performing model on each task discovered via HP search.

## ORCID iDs

Andrea Merlo  <https://orcid.org/0000-0001-8359-2731>  
 Daniel Böckenhoff  <https://orcid.org/0000-0003-1033-4648>  
 Jonathan Schilling  <https://orcid.org/0000-0002-6363-6554>  
 Udo Höfel  <https://orcid.org/0000-0003-0971-5937>  
 Sehyun Kwak  <https://orcid.org/0000-0001-7874-7575>  
 Andrea Pavone  <https://orcid.org/0000-0003-2398-966X>  
 Samuel Aaron Lazerson  <https://orcid.org/0000-0001-8002-0121>  
 Thomas Sunn Pedersen  <https://orcid.org/0000-0002-9720-1276>

## References

- [1] Hirshman S.P. and Whitson J.C. 1983 Steepest descent moment method for three-dimensional magnetohydrodynamic equilibria *Phys. Fluids* **26** 12
- [2] Langenberg A., Svensson J., Thomsen H., Marchuk O., Pablant N.A., Burhenn R. and Wolf R.C. 2016 Forward modeling of x-ray imaging crystal spectrometers within the Minerva Bayesian analysis framework *Fusion Sci. Technol.* **69** 560–7
- [3] Bozhnikov S.A. et al 2020 High-performance plasmas after pellet injections in Wendelstein 7-X *Nucl. Fusion* **60** 066011
- [4] Hanson J.D. et al 2009 V3FIT: a code for three-dimensional equilibrium reconstruction *Nucl. Fusion* **49** 075031
- [5] Lazerson S.A. (the DIII-D Team) 2015 Three-dimensional equilibrium reconstruction on the DIII-D device *Nucl. Fusion* **55** 023009
- [6] Andreeva T. et al 2019 Equilibrium evaluation for Wendelstein 7-X experiment programs in the first divertor phase *Fusion Eng. Des.* **146** 299–302
- [7] Howell E.C. and Hanson J.D. 2020 Development of a non-parametric Gaussian process model in the three-dimensional equilibrium reconstruction code V3FIT *J. Plasma Phys.* **86** 905860102
- [8] Lazerson S.A. et al 2020 Validation of the BEAMS3D neutral beam deposition model on Wendelstein 7-X *Nucl. Fusion* **60** 076020
- [9] Mynick H.E., Pomphrey N. and Ethier S. 2002 Exploration of stellarator configuration space with global search methods *Phys. Plasmas* **9** 869–76
- [10] Drevlak M., Beidler C.D., Geiger J., Helander P. and Turkin Y. 2019 Optimisation of stellarator equilibria with ROSE *Nucl. Fusion* **59** 016010
- [11] Feng Z., Gates D.A., Lazerson S.A., Landreman M., Pomphrey N. and Fu G. 2020 Optimization of quasi-axisymmetric stellarators with varied elongation *Phys. Plasmas* **27** 022502
- [12] Terranova D., Marrelli L., Hanson J.D., Hirshman S.P., Cianciosa M. and Franz P. 2013 Helical equilibrium reconstruction with V3FIT in the RFX-mod reversed field pinch *Nucl. Fusion* **53** 113014
- [13] Lazerson S.A. and Chapman I.T. 2013 STELLOPT modeling of the 3D diagnostic response in ITER *Plasma Phys. Control. Fusion* **55** 084004
- [14] Chapman I.T. et al 2014 Three-dimensional distortions of the tokamak plasma boundary: boundary displacements in the presence of resonant magnetic perturbations *Nucl. Fusion* **54** 083006
- [15] Lazerson S.A. 2014 The ITER 3D magnetic diagnostic response to applied  $n = 3$  and  $n = 4$  resonant magnetic perturbations *Plasma Phys. Control. Fusion* **56** 095006
- [16] Schmitt J.C., Bialek J., Lazerson S. and Majeski R. 2014 Magnetic diagnostics for equilibrium reconstructions with eddy currents on the lithium tokamak experiment *Rev. Sci. Instrum.* **85** 11E817
- [17] King J.D. et al 2015 Experimental tests of linear and nonlinear three-dimensional equilibrium models in DIII-D *Phys. Plasmas* **22** 072501
- [18] Lazerson S.A., Loizu J., Hirshman S. and Hudson S.R. 2016 Verification of the ideal magnetohydrodynamic response at rational surfaces in the VMEC code *Phys. Plasmas* **23** 012507
- [19] Koliner J.J. et al 2016 Three dimensional equilibrium solutions for a current-carrying reversed-field pinch plasma with a close-fitting conducting shell *Phys. Plasmas* **23** 032508
- [20] Wingen A. et al 2016 Use of reconstructed 3D VMEC equilibria to match effects of toroidally rotating discharges in DIII-D *Nucl. Fusion* **57** 016013
- [21] Cianciosa M. et al 2017 Helical core reconstruction of a DIII-D hybrid scenario tokamak discharge *Nucl. Fusion* **57** 076015
- [22] Cianciosa M., Hirshman S.P., Seal S.K. and Shafer M.W. 2018 3D equilibrium reconstruction with islands *Plasma Phys. Control. Fusion* **60** 044017
- [23] Seal S.K. et al 2016 PARVMEC: an efficient, scalable implementation of the variational moments equilibrium code 2016 45th Int. Conf. Parallel Processing (ICPP) (16–19 August 2016) (Piscataway, NJ: IEEE) pp 618–27
- [24] Seal S.K. et al 2017 Parallel reconstruction of three dimensional magnetohydrodynamic equilibria in plasma confinement devices *Proc. Int. Conf. Parallel Processing* (Bristol, United Kingdom, 14–17 August 2017) (<https://doi.org/10.1109/ICPP.2017.37>)
- [25] Schmitt J. 2021 private communication
- [26] Hoefel U. et al 2019 Bayesian modeling of microwave radiometer calibration on the example of the Wendelstein 7-X electron cyclotron emission diagnostic *Rev. Sci. Instrum.* **90** 043502
- [27] Paul E.J. et al 2018 An adjoint method for gradient based optimization of stellarator coil shapes *Nucl. Fusion* **58** 076015
- [28] Svensson J. and Werner A. 2007 Large scale Bayesian data analysis for nuclear fusion experiments 2007 IEEE Int. Symp. Intelligent Signal Processing, WISP (Alcala de Henares, Spain, 3–5 October 2007) (Piscataway, NJ: IEEE) pp 1–6
- [29] Svensson J. et al 2010 Connecting physics models and diagnostic data using Bayesian graphical models 37th EPS Conf. Plasma Physics 2010, EPS (Dublin, Ireland, 21–25 June 2010) pp 169–72
- [30] Andreeva T. 2002 Vacuum magnetic configurations of Wendelstein 7-X *Technical Report No. IPP III/270* Max-Planck-Institut für Plasmaphysik
- [31] Svensson J. 2011 Non-parametric tomography using Gaussian processes *Technical Report EFDA-JETPR(11)24* JET Internal Report
- [32] Svensson J., Dinklage A., Geiger J., Werner A. and Fischer R. 2004 Integrating diagnostic data analysis for W7-AS using Bayesian graphical models *Rev. Sci. Instrum.* **75** 4219–21

- [33] Ford O.P. 2010 Tokamak plasma analysis through Bayesian diagnostic modelling *PhD Thesis* Imperial College London
- [34] Pavone A. *et al* 2019 Neural network approximation of Bayesian models for the inference of ion and electron temperature profiles at W7-X *Plasma Phys. Control. Fusion* **61** 075012
- [35] Kwak S. 2020 Bayesian modelling of nuclear fusion experiments *Doctoral Thesis* Technische Universität Berlin, Berlin
- [36] Sengupta A., McCarthy P.J., Geiger J. and Werner A. 2004 Fast recovery of vacuum magnetic configuration of the W7-X stellarator using function parameterization and artificial neural networks *Nucl. Fusion* **44** 1176–88
- [37] Sengupta A., Geiger J. and Carthy P.J.M. 2007 Statistical analysis of the equilibrium configurations of the W7-X stellarator *Plasma Phys. Control. Fusion* **49** 649–73
- [38] Cybenko G. 1989 Approximation by superpositions of a sigmoidal function *Math. Control Signal Syst.* **2** 303–14
- [39] Hornik K. 1991 Approximation capabilities of multilayer feedforward networks *Neural Netw.* **4** 251–7
- [40] Pinkus A. 1999 Approximation theory of the MLP model in neural networks *Acta Numer.* **8** 143–95
- [41] Eldan R. and Shamir O. 2016 The power of depth for feedforward neural networks 23–26 June 2016 New-York City, USA *Conf. on Learning Theory* **49** (New-York City, USA, 23–26 June 2016) pp 907–40
- [42] Lu Z. *et al* 2017 The expressive power of neural networks: a view from the width *Advances in Neural Information Processing Systems* **30** (Long Beach, CA, USA, 4–9 December 2017)
- [43] 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
- [44] Tribaldos V. and Van Milligen B.P. 1997 Neural network tool for rapid recovery of plasma topology *Rev. Sci. Instrum.* **68** 931–4
- [45] Joung S. *et al* 2020 Deep neural network Grad–Shafranov solver constrained with measured magnetic signals *Nucl. Fusion* **60** 016034
- [46] Citrin J. *et al* 2015 Real-time capable first principle based modelling of tokamak turbulent transport *Nucl. Fusion* **55** 092001
- [47] Meneghini O. *et al* 2017 Self-consistent core-pedestal transport simulations with neural network accelerated models *Nucl. Fusion* **57** 086034
- [48] van de Plassche K.L., Citrin J., Bourdelle C., Camenen Y., Casson F.J., Dagnelie V.I., Felici F., Ho A. and Van Mulders S. 2020 Fast modelling of turbulent transport in fusion plasmas using neural networks *Phys. Plasmas* **27** 022310
- [49] Pavone A., Svensson J., Kwak S., Brix M. and Wolf R.C. 2020 Neural network approximated Bayesian inference of edge electron density profiles at JET *Plasma Phys. Control. Fusion* **62** 045019
- [50] Piccione A., Berkery J.W., Sabbagh S.A. and Andreopoulos Y. 2020 Physics-guided machine learning approaches to predict the ideal stability properties of fusion plasmas *Nucl. Fusion* **60** 046033
- [51] Pavone A., Svensson J., Langenberg A., Pablant N., Hoefel U., Kwak S. and Wolf R.C. 2018 Bayesian uncertainty calculation in neural network inference of ion and electron temperature profiles at W7-X *Rev. Sci. Instrum.* **89** 10K102
- [52] Ho A., Citrin J., Bourdelle C., Camenen Y., Casson F.J., van de Plassche K.L. and Weisen H. 2021 Neural network surrogate of QualiKiz using JET experimental data to populate training space *Phys. Plasmas* **28** 032305
- [53] Rasmussen C.E. 2004 *Gaussian Processes in Machine Learning (Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics))* vol **3176** (Berlin: Springer) pp 63–71
- [54] Kwak S., Svensson J., Bozhenkov S., Flanagan J., Kempnaars M., Boboc A. and Ghim Y.-C. 2020 Bayesian modelling of Thomson scattering and multichannel interferometer diagnostics using Gaussian processes *Nucl. Fusion* **60** 046009
- [55] ASDEX Team 1989 The H-mode of ASDEX *Nucl. Fusion* **29** 1959–2040
- [56] Higdon D., Swall J., Kern J. 1999 Non-stationary spatial modeling *Bayesian Statistics* **6** 1
- [57] Chilenski M.A., Greenwald M., Marzouk Y., Howard N.T., White A.E., Rice J.E. and Walk J.R. 2015 Improved profile fitting and quantification of uncertainty in experimental measurements of impurity transport coefficients using Gaussian process regression *Nucl. Fusion* **55** 023012
- [58] Kwak S. *et al* 2021 Bayesian inference of spatially resolved  $Z_{\text{eff}}$  profiles from line integrated Bremsstrahlung spectra *Rev. Sci. Instrum.* **92** 043505
- [59] Drews P. *et al* 2019 Edge plasma measurements on the OP 1.2a divertor plasmas at W7-X using the combined probe *Nucl. Mater. Energy* **19** 179–83
- [60] Klinger T. *et al* 2019 Overview of first Wendelstein 7-X high-performance operation *Nucl. Fusion* **59** 112004
- [61] Wolf R.C. *et al* 2019 Performance of Wendelstein 7-X stellarator plasmas during the first divertor operation phase *Phys. Plasmas* **26** 082504
- [62] Nemov V.V. *et al* 1999 Evaluation of  $1/\nu$  neoclassical transport in stellarators *Phys. Plasmas* **6** 12
- [63] Nührenberg C. 2016 Free-boundary ideal MHD stability of W7-X divertor equilibria *Nucl. Fusion* **56** 076010
- [64] Beidler C. *et al* 1990 Physics and engineering design for Wendelstein VII-X *Fusion Technol.* **17** 148–68
- [65] Langenberg A. *et al* 2019 Inference of temperature and density profiles via forward modeling of an x-ray imaging crystal spectrometer within the Minerva Bayesian analysis framework *Rev. Sci. Instrum.* **90** 063505
- [66] Bishop C.M. 1995 *Neural Networks for Pattern Recognition* (Oxford: Oxford University Press)
- [67] Ji S. *et al* 2013 3D convolutional neural networks for human action recognition *IEEE Trans. Pattern Anal. Mach. Intell.* **35** 221–31
- [68] LeCun Y., Bengio Y. and Hinton G. 2015 Deep learning *Nature* **521** 436–44
- [69] Cho K. *et al* 2015 On the properties of neural machine translation: encoder–decoder approaches (arXiv:1409.1259)
- [70] Ronneberger O., Fischer P. and Brox T. 2015 U-net: convolutional networks for biomedical image segmentation (arXiv:1505.04597)
- [71] Badrinarayanan V., Kendall A. and Cipolla R. 2017 SegNet: a deep convolutional encoder–decoder architecture for image segmentation *IEEE Trans. Pattern Anal. Mach. Intell.* **39** 2481–95
- [72] Ioffe S. and Szegedy C. 2015 Batch normalization: accelerating deep network training by reducing internal covariate shift *32nd Int. Conf. Machine Learning, ICML* (Lille, France, 6–11 July 2015) vol **37** pp 448–56
- [73] Srivastava N. *et al* 2014 Dropout: a simple way to prevent neural networks from overfitting *J. Mach. Learn. Res.* **15** 1
- [74] Springenberg J.T. *et al* 2015 Striving for simplicity: the all convolutional net *3rd Int. Conf. Learning Representations, ICLR 2015—Workshop Track Proc.* (San Diego, CA, USA, 7–9 May 2015)
- [75] Glorot X. and Bengio Y. 2010 Understanding the difficulty of training deep feedforward neural networks *International Conference on Artificial Intelligence and Statistics* **9** ed Y.W. Teh and M. Titterton (Sardinia, Italy, 13–15 May 2010) vol 9 pp 249–56
- [76] Kingma D.P. and Ba J.L. 2015 Adam: a method for stochastic optimization *3rd Int. Conf. Learning Representations, ICLR 2015—Conf. Track Proc.* (San Diego, CA, USA, 7–9 May 2015)

- [77] Morgan N. and Bourlard H. 1989 Generalization and parameter estimation in feedforward nets: some experiments *Proc. 2nd Int. Conf. Neural Information Processing Systems* (Denver, Colorado, USA, 27–30 November 1989)
- [78] Abadi M. *et al* 2016 TensorFlow: a system for large-scale machine learning (<https://research.google/pubs/pub45381/>)
- [79] Klambauer G. *et al* 2017 Self-normalizing neural networks (arXiv:1706.02515)
- [80] Wiesler S. and Ney H. 2011 A convergence analysis of log-linear training *Advances in Neural Information Processing Systems 24: 25th Annual Conf. Neural Information Processing Systems 2011 (NIPS)* (Granada, Spain, 12–17 December 2011) pp 657–65
- [81] Pedregosa F. *et al* 2011 Scikit-learn: machine learning in Python *J. Mach. Learn. Res.* **12** 85
- [82] Bergstra J. *et al* 2011 Algorithms for hyper-parameter optimization *Advances in Neural Information Processing Systems 24: 25th Annual Conf. Neural Information Processing Systems 2011 (NIPS)* (Granada, Spain, 12–17 December 2011) pp 2546–554
- [83] Bergstra J., Yamins D. and Cox D.D. 2013 Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures *30th Int. Conf. Machine Learning (ICML)* (Atlanta, USA, 16–21 June 2013) vol **28** p 1
- [84] Jones D.R. 2001 A taxonomy of global optimization methods based on response surfaces *J. Glob. Optim.* **21** 345–83
- [85] Stone M. 1974 Cross-validatory choice and assessment of statistical predictions *J. R. Stat. Soc. B* **36** 111–33
- [86] Geisser S. 1975 The predictive sample reuse method with applications *J. Am. Stat. Assoc.* **70** 320–8
- [87] Efron B. and Tibshirani R. 1997 Improvements on cross-validation: the 632+ Bootstrap method *J. Am. Stat. Assoc.* **92** 548–60
- [88] Renner H. *et al* 2002 Divertor concept for the W7-X stellarator and mode of operation *Plasma Phys. Control. Fusion* **44** 325
- [89] Geiger J., Beidler C.D., Feng Y., Maaßberg H., Marushchenko N.B. and Turkin Y. 2015 Physics in the magnetic configuration space of W7-X *Plasma Phys. Control. Fusion* **57** 014004
- [90] Neuner U. *et al* 2020 Measurements of the parameter dependencies of the bootstrap current in the W7-X stellarator *Nucl. Fusion* **61** 036024
- [91] DiCiccio T.J. and Efron B. 1996 Bootstrap confidence intervals *Stat. Sci.* **11** 189–228
- [92] Krishnamoorthi R. 2018 Quantizing deep convolutional networks for efficient inference: a whitepaper (arXiv:1806.08342)
- [93] Liang T. *et al* 2021 Pruning and quantization for deep neural network acceleration: a survey (arXiv:2101.09671)
- [94] Raissi M., Perdikaris P. and Karniadakis G.E. 2019 Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations *J. Comput. Phys.* **378** 686–707
- [95] Hansen L.K. and Salamon P. 1990 Neural network ensembles *IEEE Trans. Pattern Anal. Machine Intell.* **12** 993–1001
- [96] Brand A., Allen L., Altman M., Hlava M. and Scott J. 2015 *Beyond Authorship: Attribution, Contribution, Collaboration, and Credit* vol **28** (Learned Publishing) pp 151–5