

PAPER

Self-consistent core-pedestal transport simulations with neural network accelerated models

To cite this article: O. Meneghini *et al* 2017 *Nucl. Fusion* **57** 086034

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

You may also like

- [Simulation prediction of micro-instability transition and associated particle transport in tokamak plasmas](#)

H. Li, J.Q. Li, Y.L. Fu et al.

- [Al₂O₃-Al doped ZnO and SnO₂ encapsulation of randomly oriented ZnO nanowire networks for high performance and stable electrical devices](#)

F Morisot, V H Nguyen, C Montemont et al.

- [Condition-number-based measurement configuration optimization for nanostructure reconstruction by optical scatterometry](#)

Tianjuan Yang, Xiuguo Chen, Shuo Liu et al.

Self-consistent core-pedestal transport simulations with neural network accelerated models

O. Meneghini¹, S.P. Smith¹, P.B. Snyder¹, G.M. Staebler¹, J. Candy¹, E. Belli¹, L. Lao¹, M. Kostuk¹, T. Luce¹, T. Luda², J.M. Park³ and F. Poli⁴

¹ General Atomics, San Diego CA, United States of America

² Politecnico di Torino, Torino, Italy

³ Oak Ridge National Laboratory, Oak Ridge TN, United States of America

⁴ Princeton Plasma Physics Laboratory, Princeton NY, United States of America

E-mail: meneghini@fusion.gat.com

Received 11 January 2017, revised 11 April 2017

Accepted for publication 6 June 2017

Published 12 July 2017



Abstract

Fusion whole device modeling simulations require comprehensive models that are simultaneously physically accurate, fast, robust, and predictive. In this paper we describe the development of two neural-network (NN) based models as a means to perform a non-linear multivariate regression of theory-based models for the core turbulent transport fluxes, and the pedestal structure. Specifically, we find that a NN-based approach can be used to consistently reproduce the results of the TGLF and EPED1 theory-based models over a broad range of plasma regimes, and with a computational speedup of several orders of magnitudes. These models are then integrated into a predictive workflow that allows prediction with self-consistent core-pedestal coupling of the kinetic profiles within the last closed flux surface of the plasma. The NN paradigm is capable of breaking the speed-accuracy trade-off that is expected of traditional numerical physics models, and can provide the missing link towards self-consistent coupled core-pedestal whole device modeling simulations that are physically accurate and yet take only seconds to run.

Keywords: neural networks, transport, pedestal, tokamak

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

1. Background and research motivation

The physics models that are used for practical whole device modeling (WDM) applications must strike a challenging balance between accuracy and speed, depending on the degree of physical fidelity that is possible given the physics understanding, the amount of available computational resources, and the accuracy and time constraints of each individual study. As shown in figure 1, on the one end of the simulations spectrum there are high-fidelity physics models (typically with high computational demands), whose use is mostly limited to the interpretation of a single time-slice, and is often aimed at validating our physics understanding of specific plasma processes. On the other end of the spectrum, models that are used

in experimental feedback control systems have loose fidelity, but stringent real-time simulation requirements. Between these extremes are predictive simulations, and sensitivity and optimization studies, which form the basis of strategic and experimental planning. These simulations usually require a high level of integration across different plasma regions with widely varying spatial and temporal scales—hence the terminology WDM—and rely on reduced-fidelity components to provide reasonable turnaround time for simulations. The struggle between high-fidelity and fast turnaround is a major hurdle towards the accomplishment of WDM goals. This issue was recently highlighted in the 2015 US DOE Report on the workshop on integrated simulations for magnetic fusion [1], where it is stated that *a single first-principles simulation of*

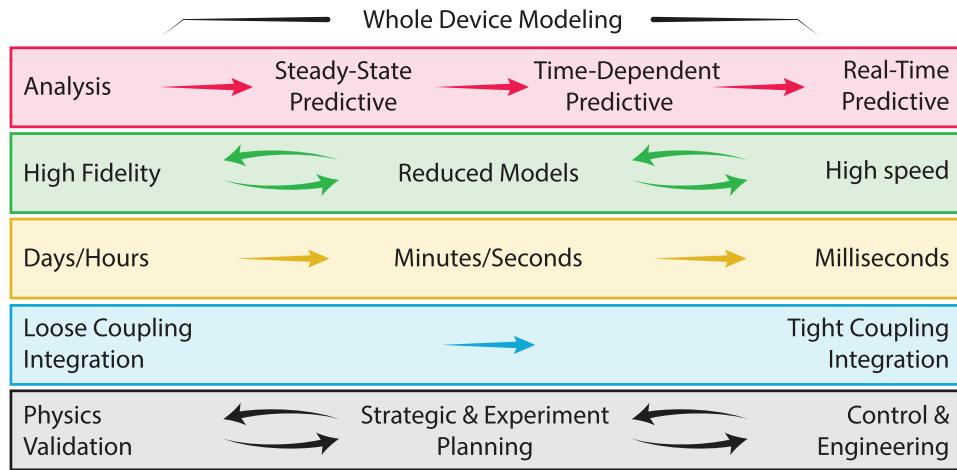


Figure 1. Fusion integrated modeling simulations employ a hierarchy of physics models, encompassing a broad range of fidelity and speed capabilities.

a whole device would require a staggering amount of computational resources. As such, one of the great challenges of integrated simulation is the development of a validated set of models of varying fidelity that can be employed to optimize the use of computational resources to produce simulation outcomes with a required level of accuracy.

This issue is particularly critical in the prediction of the kinetic profiles in the confined region of a tokamak plasma, whose precise yet fast solution is key to the success of the WDM concept itself. The confined region of the plasma is where fusion processes take place. There the fusion performance is highly sensitive to the boundary constraint that is imposed by the pedestal structure of H-mode plasmas. To complicate things further, the physical processes that govern the core and pedestal regions of tokamak plasmas are to be non-linearly coupled to one another. These considerations enforce tight constraints on the fidelity requirements of the models that aim to capture the behavior of these two regions of the plasma, as well as on their coupling scheme.

In this paper we describe the development of two neural-network (NN) based models as a means to perform a non-linear multivariate regression of theory-based simulations for the core turbulent transport fluxes (section 3), and the pedestal structure section 4. Specifically, we find that a NN-based approach can be used to consistently reproduce the results of the TGLF [2] and EPED1 [3] theory-based models over a broad range of plasma regimes, and with a computational speedup of several orders of magnitudes. Each of these sections describes the NN topology and training scheme, benchmark of the NN-based model compared to the original one, and error and sensitivity analysis. Section 5 shows the results of using the models developed in the paper as part of a predictive workflow within the TGYRO transport solver [4], which allows prediction with self-consistent core-pedestal coupling of the kinetic profiles within the last closed flux surface of the plasma. The result of the NN based simulations are then compared against the simulation results using the full models for a set of DIII-D experiments, and have been detailed with convergence and sensitivity analysis.

We stress that no experimental data has been used for training the NN models, and that these models are trained only on the results of the full models simulations, the goal being to reproduce the results of the original TGLF and EPED1 calculations at a fraction of the computational cost. If the full TGLF and EPED1 calculations are a good representation of the experiments, and the NN models are a good representation of the full models, by the transitive property of equality one can expect that the NN models should also be a good representation of the experiment. Nonetheless, validating the physics of the NN or of the original models against experiments, or verifying their outcome against first principles simulations (e.g. non-linear gyro-kinetic simulations) is beyond the scope of this paper. Extensive validations and verifications studies of the full EPED and TGLF models have already been carried out and documented in separate studies, as referenced in the relevant sections of this paper.

2. NN model and infrastructure for data collection

For our study we make use of multi-layer feed-forward neural-networks (NN) as an efficient tool for non-linear regression of multi-dimensional data. In a feed-forward NN the information travels from a layer of ‘input’ neurons, through multiple layers of ‘hidden’ neurons, and out of a layer of ‘output’ neurons. The topology of the NN described in section 3 for the turbulent transport, and in section 4 for the pedestal structure, have been obtained by heuristic optimization of the regularity and accuracy of the regression, the rate of the training convergence, and the training time.

The training of the NNs was accomplished by means of supervised learning with the backward-propagation algorithm. The formalism behind this type of NN is well established [5, 6]. For all cases presented in this paper, 50% of the supervised learning database (randomly sampled) was used for training and the remaining 50% for validation. The training set is used to adjust the weights on the NN. The validation set is used to verify that any increase in accuracy over the training data set actually yields an increase in accuracy over a data set that

the network has not trained on. If throughout the training process the training error decreases, but the validation error stays the same or increases, then the NN model is over-fitting and the training process should be stopped. To further prevent over-fitting of the training data, and ensure good generalization properties, the NN was regularized by means of a weight decay algorithm [7].

Evaluating whether the NN prediction is accurate (or not) is central for its deployment to real world applications. To improve the physics fidelity of our models and obtain a reliable estimate of their error, we average the outcome of multiple NN regressions V_α into a single prediction \bar{V} [8], so that

$$\bar{V} = \sum_{\alpha} w_{\alpha} V_{\alpha} \quad (1)$$

where α is an index that identifies the NN. Each of the NNs in the ensemble is initialized with random connection weights, and is independently trained and validated on a randomly chosen subset of points from the training and validation databases, respectively. The generalization error of the NNs ensemble can be shown to be bounded by the variance in the outputs of the NNs

$$\bar{a} = \sum_{\alpha} w_{\alpha} (V_{\alpha} - \bar{V}_{\alpha})^2, \quad (2)$$

a quantity that is generally referred to as *ensemble ambiguity* [8].

From a technical standpoint, effective data collection, management, and dissemination was achieved by the development of a remote data collection infrastructure named HARVEST. The availability of C, FORTRAN, and Python APIs, allow this system to be conveniently embedded within scientific codes, and data to be transferred at the run-time to a remote server. On the server side, HARVEST leverages the functionalities of no-SQL databases to support rapidly changing schemas of unstructured data with scalable performance. Figure 2 shows the HARVEST data flow for the case of the NN-based model for the pedestal structure, as it is orchestrated by the one modeling framework for integrated tasks (OMFIT) modeling framework [9]. This solution is now being shared by numerous projects requiring a seamless run-time collection of simulation data. For the purpose of this study, the OMFIT framework provides the infrastructure that is used to (1) concurrently submit large sets of simulations on high-performance computing clusters to efficiently build the training database (2) manage the entire data pipeline, from accessing the data stored on the remote HARVEST data collection system to the training and benchmarking of the ensemble of NNs; and finally (3) execute the integrated workflows that are needed for the analysis, verification, and validation of the new models (figure 2).

3. Accelerated trapped gyro Landau fluid model

State of the art turbulent transport models are based on gyrokinetic (GK) theory. Traditionally GK simulations are designed to capture either the ion-scale or the electron-scale turbulent fluctuations, and requires thousands of CPU-hour on high performance computing systems. Recently, the importance

of simulating the ion and electron scales simultaneously has been highlighted [10]. The computational requirements for such multi-scale simulations are extreme, of the order of ~ 10 million CPU-hour.

It is in this context that reduced GK models have been developed. Models such as trapped Gyro Landau fluid (TGLF) [2, 11] are in fact capable of reproducing the particle, energy, and momentum fluxes of GK simulations with a high degree of efficiency, requiring in the order of 10 CPU-second per data point. Nonetheless, even these reduced models are slow considering that typical transport simulations require thousands of evaluations. Furthermore, thorough scientific inference typically requires hundreds of simulations and that real-time control applications require models that are capable of providing accurate solutions within milliseconds.

TGLF is a quasilinear transport model that uses the same general methods as the Weiland [12], GLF23 [13] and Multi-Mode [14] models to predict with high fidelity the turbulent transport fluxes that would be obtained from first-principles GK simulations. In TGLF the linear eigenvalues and eigenfunctions of a system of trapped gyro-Landau fluid equations [15] are used to evaluate the quasilinear weights of the transport fluxes. The linear growth rate spectrum is used to compute a model amplitude for the saturated turbulence. TGLF extends its predecessor GLF23 by having a more accurate system of equations valid at both electron and ion scale, a realistic plasma geometry, and a more accurate trapped particle treatment. The linear eigenvalues of TGLF have been extensively benchmarked with gyrokinetic calculations [15]. The calculation of the linear growth rate spectrum requires in the order of 1 to 10 CPU-second (depending on the number of ion species in the plasma).

For our study we generated a database of TGLF simulations for the turbulent particle, energy, and momentum transport fluxes as a function of the relevant plasma parameters, and trained a NN to reproduce the same results with a computational cost of a few CPU-microsecond.

TGLF's underlying equations are designed to capture the physics of turbulent transport fluxes in magnetized plasmas. As such, a small number of free parameters in the model—only four—are needed to reproduce the behavior of the gyrokinetic transport fluxes, and these parameters can be determined with a relatively limited database of nonlinear gyrokinetic turbulence simulations (a database of 82 non-linear GYRO simulations was used to calibrate the TGLF model). In contrast, NN-based regression models do not make any assumption about the functional form of the underlying physics, and thus must rely on a significantly larger number of free parameters (the weights in the synaptic connections between neurons) to approximate the functional form of the physical processes of interest. Constraining such a large number of degrees of freedom poses requirements on the number and distribution of samples in the NN training database.

We point out that although the transport simulations database used to train the NN could in principle be based directly on the results of non-linear GK simulations, the computational cost to generate a sufficiently large database with GK simulations would be excessive, even for today's most advanced

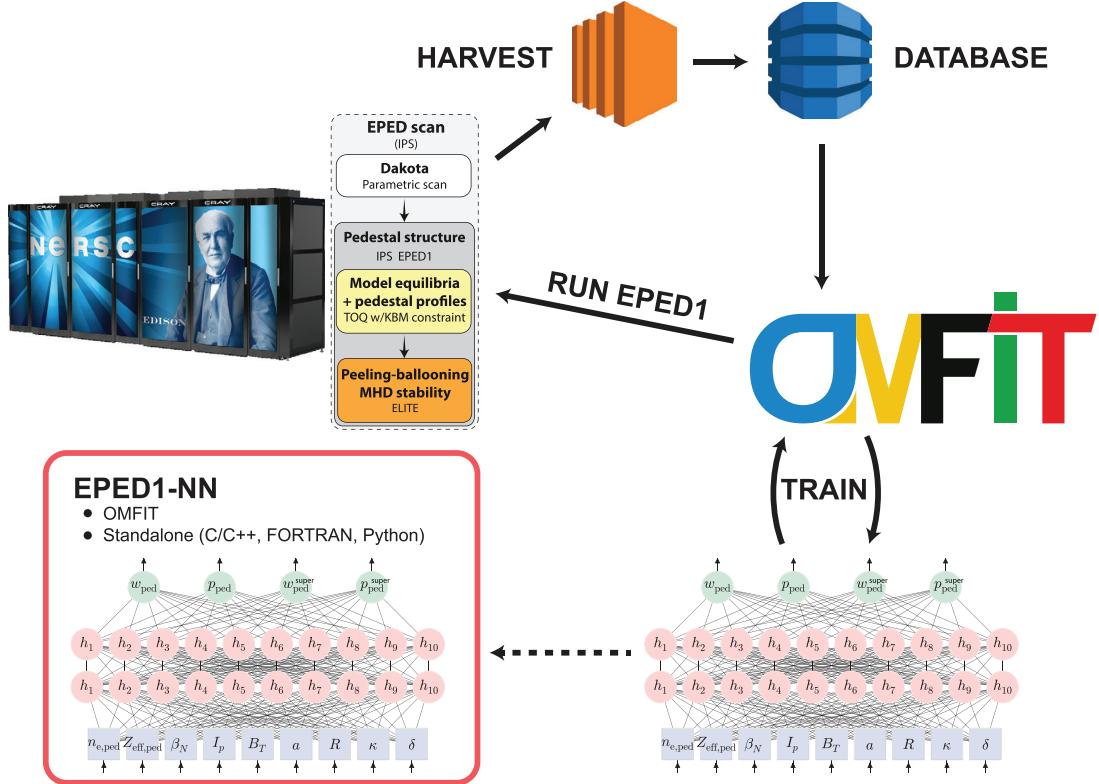


Figure 2. The OMFIT framework provides a convenient and flexible working environment, that is used to launch concurrent sets of simulations to build the training database, interface to the HARVEST data collection system, oversee the NN training, and finally enable the analysis, verification, and validation of the new NN-based models.

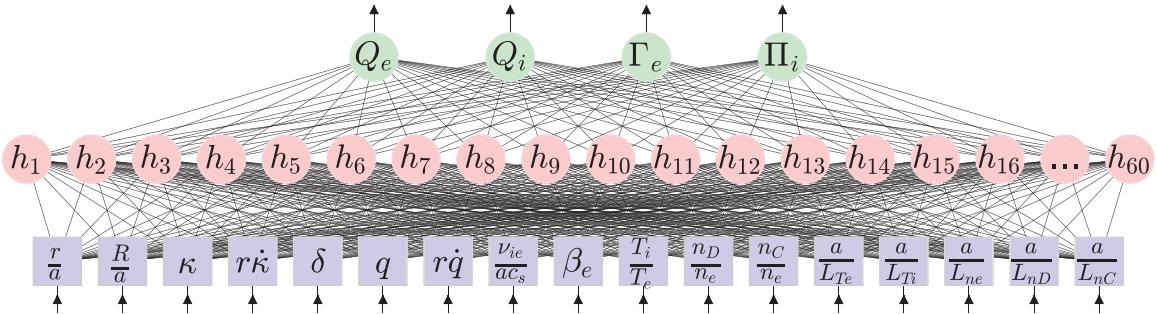


Figure 3. Schematic of the NN topology (simplified in the number of hidden neurons) that is at the basis of the TGLF-NN model. The TGLF-NN model is evaluated as the average of an ensemble of 32 NNs, each fed with the same input parameters values. The standard deviation of the ensemble provides an estimate of the uncertainty in the TGLF-NN model.

computing resources. To complicate things further, the turbulent fluxes of GK simulations are notoriously intermittent near threshold (i.e. zero flux condition). These considerations further highlight the importance of reduced models, such as TGLF, to build a training database for machine learning applications.

3.1. TGLF-NN topology

The NN topology used to model the TGLF turbulent fluxes is schematically shown in figure 3. The results presented in this paper were obtained with an ensemble of 32 NNs, each with 1 hidden layer of 60 neurons each. The choice of the TGLF-NN input and output parameters was dictated by the requirements and common practices that are used in predictive transport simulations with the TGYRO [4] transport code.

In addition to its capability as a transport code, TGYRO is also a parallel transport manager with the ability to call multiple concurrent instances of the TGLF [2] transport model, or more generally, massively-parallel GYRO [16] simulations. TGYRO can simultaneously manage the parallel execution of kinetic NEO [17–19] simulations. The total transport fluxes are given by a sum of the NEO neoclassical fluxes and the turbulent fluxes (via TGLF in this paper). The TGYRO approach (combining TGLF and NEO) has been previously applied with great success for the prediction of kinetic profiles in the core region of tokamak plasmas [4, 11, 20–24].

The NN inputs are the same set of local dimensionless plasma parameters which are input to the TGLF model, as listed in table 1. In the table $L_x = -\frac{1}{x} \frac{\partial x}{\partial r}$ is the scale length of a quantity x ; c_s is the ion sound speed; r and a are the plasma

Table 1. Local dimensionless plasma parameters which are input to the TGLF-NN model.

Variable	Definition
r/a	Normalized minor radius
R/a	Normalized major radius
κ	Elongation
$r \frac{\partial \kappa}{\partial r}$	Elongation shear
δ	Triangularity
$\frac{\partial R}{\partial r}$	Shafranov shift
q	Safety factor
$\frac{q^2 a^2 \partial q}{r^2 \partial r}$	Safety factor shear
β_e	Kinetic to magnetic pressure ratio
ν_{ie}/ac_s	Collision frequency
T_i/T_e	Ion to electron temperature ratio
n_D/n_e	Deuterium to electron density ratio
n_C/n_e	Carbon to electron density ratio
Z_{eff}	Effective ion charge
a/L_{Te}	Electron temperature scale length
a/L_{Ti}	Ion temperature scale length
a/L_{ne}	Electron density scale length
a/L_{nD}	Deuterium density scale length
a/L_{nC}	Carbon density scale length
$\frac{qa^2}{rB_{\text{unit}}} \frac{\partial p}{\partial r}$	Total pressure gradient
$\text{sign}(I_p)R\omega_{\text{tor}} \frac{a}{c_s}$	Parallel velocity
$-\text{sign}(I_p)R \frac{\partial \omega_{\text{tor}}}{\partial r} \frac{a}{c_s}$	Parallel velocity gradient
$-\text{sign}(I_p) \frac{r}{q} \frac{\partial V_{E \times B}}{\partial r} \frac{a}{c_s}$	$E \times B$ velocity shear

minor radius and its value at the last closed flux surface, respectively. The minor radius is defined as $r = \sqrt{\Phi/\pi B_{\text{unit}}}$, where Φ is the toroidal magnetic flux, and $B_{\text{unit}} = \frac{\partial \Phi}{\partial r}/(2\pi r)$. The plasma elongation κ and triangularity δ are defined as by Miller [25]; $r \frac{\partial \kappa}{\partial r}$ is the normalized elongation shear, q is the safety factor, $r \frac{\partial q}{\partial r}$ is the normalized shear, β_e is the kinetic to magnetic pressure ratio, and ν_{ie}/ac_s is the normalized electron-ion collision frequency. The plasma triangularity shear $r \frac{\partial \delta}{\partial r}$ was not input to the TGLF-NN model as it was found to have minimal impact on the output of TGLF. We assume that the temperatures T_i and toroidal velocities V_{tor} are the same for all of the thermal ion species. The TGLF-NN model used in this paper considers Deuterium and Carbon as ion species. Additional thermal ion species could be included in the model with the addition of their density scale-length, and ratio to the electron density to the NN inputs.

The outputs of the NN are the electron and ion turbulent heat fluxes $Q_{e,i}$, the electron turbulent particle flux Γ_e , and the ion turbulent momentum flux Π_i . The ion's particle flux is neglected because TGYRO internally evolves the density equation for the electron species alone. The electron turbulent momentum flux Π_e is not an output of the NN since it is negligible with respect to the ion turbulent momentum flux (by the electron to ion mass ratio). All fluxes are evaluated as the sum

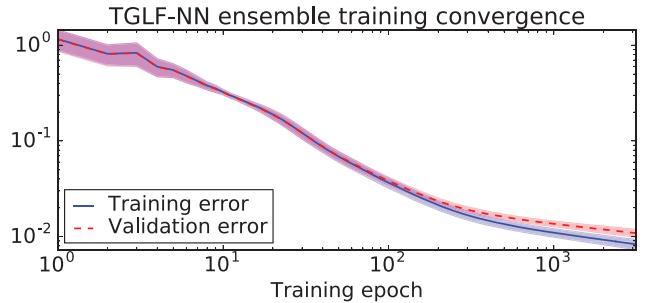


Figure 4. Convergence of the root-mean-square training and validation errors for the ensemble of 30 NN composing the TGLF-NN model. The average and standard deviation of the NN ensemble is shown as a solid line and shaded area, respectively. Both the training and validation error drop by two orders of magnitude, an indication that the training process is effective. It is normal that the training error slightly underestimates the validation error, since that is the data upon which the NN is trained on. More importantly, the validation error does not increase, which indicates that the TGLF-NN is not been over-trained.

of the electrostatic and the electromagnetic contributions with non-adiabatic electrons, and the ion contributions are summed over all ion species. As for the TGLF-NN outputs, the fluxes are normalized to the gyro-Bohm units of the primary ion species (deuterium in our case):

$$Q_{\text{GB}} = n_e c_s T_e (\rho_{D,\text{unit}}/a)^2$$

$$\Gamma_{\text{GB}} = n_e c_s (\rho_{D,\text{unit}}/a)^2$$

$$\Pi_{\text{GB}} = n_e a T_e (\rho_{D,\text{unit}}/a)^2$$

The Gyro-Bohm normalization factors out the scaling of turbulent fluxes with respect to the $\rho^* = \rho_{D,\text{unit}}/a$, where $\rho_{D,\text{unit}}$ is the ion gyroradius normalized by the tokamak minor radius a and evaluated with respect to the B_{unit} magnetic field. As a result of this normalization, the same training database could in principle be used for predicting the fluxes of experiments from different machines, as long as they have similar local non-dimensional parameters as per table 1. The value of the fluxes in physical units is then obtained by scaling the non-dimensional prediction by the Gyro-Bohm normalization of the corresponding machine.

3.2. Database assembly and training

Accounting for realistic plasma geometry, rotation, and two ion species (deuterium and carbon) requires input of 23 local plasma parameters to the TGLF model. Because the number of samples in the training database scales exponentially with the number of parameters that are input to the NN model, it is not possible to uniformly sample the input parameters domain to build the training database. Instead, a non-uniform, sparse database must be adopted.

Restricting the sampling of the domain to fewer dimensions is a possible option, but one that severely limits the applicability of the resulting NN model. For example, in [26] only four dimensionless plasma parameters were varied

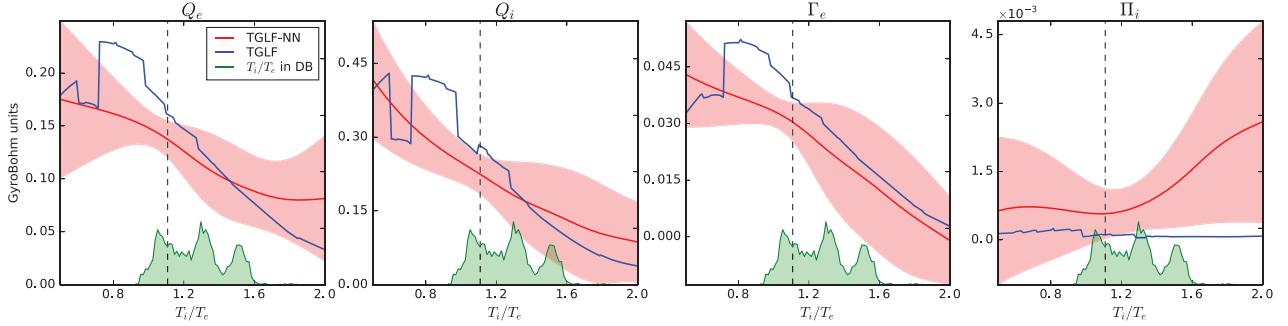


Figure 5. Sensitivity scan of the fluxes evaluated by TGLF and TGLF-NN as a function of T_i/T_e starting from a random point in the database at $\rho = 0.2$ (black dashed line). In the core region of the plasma the transport fluxes tend to be smaller, and the discontinuities in the TGLF solution are relatively more prominent. The green histogram represents the distribution of points (arbitrary units) in the database for T_i/T_e . The shadowed red area represent the ensemble ambiguity of the TGLF-NN model. Note that for this case, although the absolute magnitude of the NN momentum flux is higher than the TGLF momentum flux, its value is also very small (of the order of 10^{-3} gyro-Bohm units).

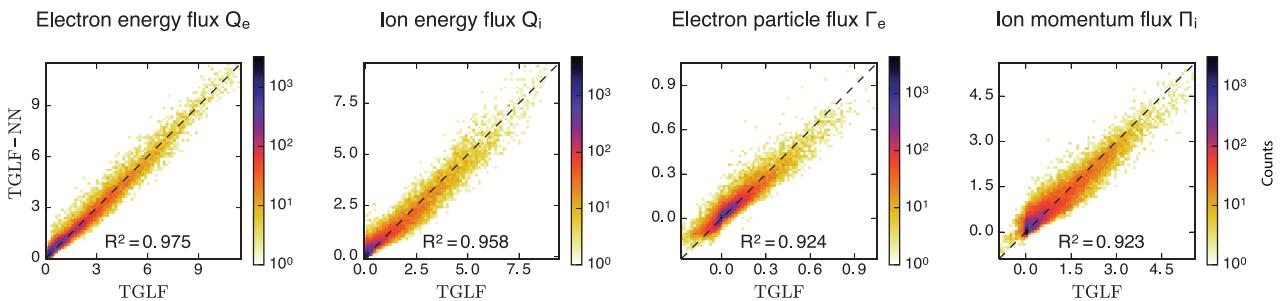


Figure 6. Two-dimensional regression histograms comparing each of the outputs of the TGLF-NN model with the full TGLF calculations. The color-bar indicates the counts in each of the in the two-dimensional histogram bins. The goodness of NN regression is quantitatively captured by the high coefficient of determination metric R^2 .

$(a/L_T i, T_i/T_e, q, r \frac{\partial q}{\partial r})$ to generate a database of ~ 4 million data points of Qualifier quasi-linear gyrokinetic transport simulations in the ITG instability regime. The plasma geometry was restricted to be circular, of fixed aspect ratio, at a prescribed radial location and without Shafranov shift. The effects of magnetic shear and rotation were also excluded from the database. Finally, the code was run with a single ion species.

For our purposes, initial sampling of the 23 dimensional input parameters space was constructed based upon measurements of actual DIII-D experiments, which aimed at studying the effect of energy transport stiffness in the ion channel. This database, which consists of ~ 4000 time-slices from 24 experimental discharges, was chosen because the experimental equilibrium reconstructions, the profiles fitting, and transport analyses were already carried out as part of the study published in [27]. The accurate particle, energy, and momentum balance analyses which were carried out for such study provides a strong basis for the quantitative comparison between the theoretical and the experimental fluxes discussed later in the paper. To efficiently build the TGLF database we leveraged the capability of the TGYRO transport code to execute multiple TGLF runs in parallel for those experimental conditions.

The training procedure described in section 2 resulted in the training convergence shown in figure 4. From the plot it can be seen that the root-mean-square validation error closely follows the training error, thus indicating that the resulting

NNs are not over-fitting the training data. This is an important aspect since it directly correlates with the ability of the trained ensemble to feature good generalization properties.

On a 3.1 GHz processor workstation the time required to train each of the NNs is about 2 min, while the evaluation of the NN outputs is in the order of $\sim 20 \mu s$. Compared to the 2 CPU-second required by the full TGLF model, the NN-based model provides a computational efficiency of over five orders of magnitude. From a technical standpoint, the software interface of TGLF-NN was designed from the beginning as a drop-in replacement, so that it could be seamlessly included into any transport code that already integrates TGLF. This aspect also guarantees that the same data is fed into the two models, thus ensuring their proper benchmark.

3.3. Smoothness

Convergence of transport codes with TGLF has been a long standing issue, which has been often attributed to the stiffness of the TGLF model itself, and has spurred a lot of effort towards the development of robust numerical algorithms in transport codes [28–30]. Although transport stiffness may indeed play a role, we have identified irregularities in the TGLF solution as a function of the input parameters that can strongly affect the evaluation of the Jacobian matrix in transport calculations, and thus negatively impact their convergence.

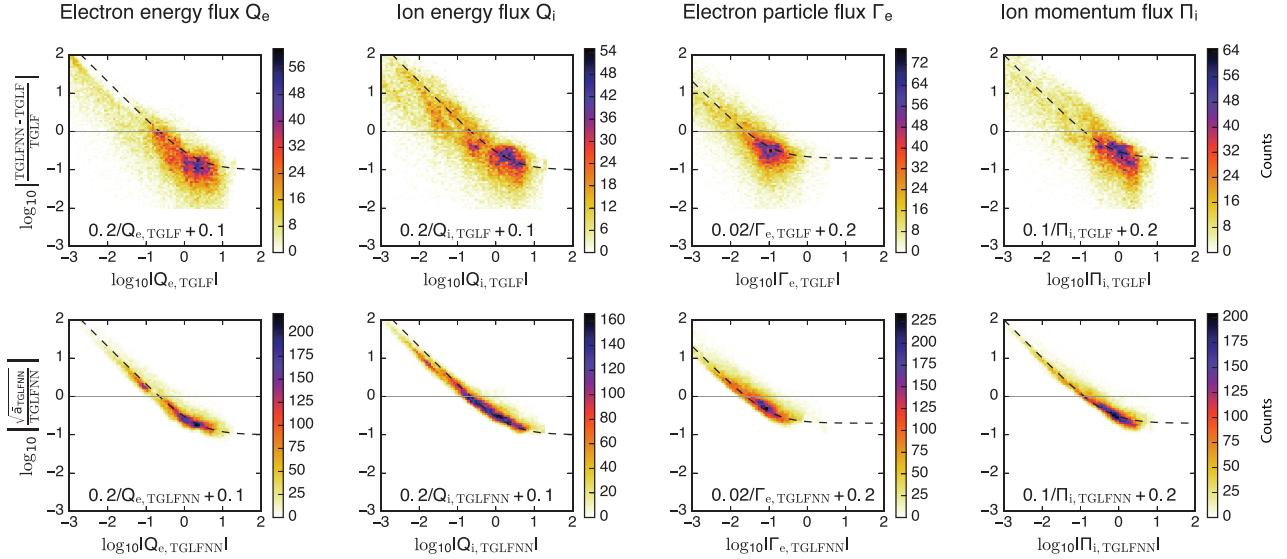


Figure 7. Plotted in logarithmic scales: the first row shows the relative error of the TGLF-NN model compared to TGLF; the second row shows the relative error estimate based on the NN ensemble average and ambiguity information. The dashed line represents the fit to the relative error estimate, based on which a threshold can be set to deduce if the NN predictions can be considered accurate or not.

This issue is illustrated in figure 5, which compares a scan in the ion to electron temperature ratio (T_i/T_e) performed with TGLF and TGLF-NN starting from a randomly selected point in the database. The TGLF solution exhibits discontinuities that, although relatively small in magnitude, can become comparable to the actual value of the fluxes when these are also small. On the other hand, the fluxes evaluated by the NN are a smooth function of the input parameters, which explains the excellent convergence properties of the transport solver when using the TGLF-NN model.

The smoothness of the TGLF-NN solution with respect to the model's input parameters is attributed to the fact that the NN approach, when used as a non-linear regression tool, performs a regularization that effectively smooths out the solution. Moreover, the output of the TGLF-NN model is evaluated as the ensemble average of multiple NNs, which further improves the smoothness of the results compared to those of the individual NN regressions.

Figure 5 also addresses another major concern for NN-based regression models: their range of applicability. Neural networks are known to be capable of providing very accurate representation of the data within the training domain, but their accuracy extrapolating outside of the domain can not be guaranteed. In general, we find the TGLF-NN model closely follows the trend of TGLF, even outside of the training domain. The histogram in the lower part of the figure represents the distribution of points in the database for T_i/T_e (plotted in arbitrary units) for the radius considered ($\rho = 0.2$ in this example). Intuitively one can expect that where there is a higher density of points, there is also a higher likelihood of being within the training domain. As expected, close to the starting condition (which we know is part of the training database) the TGLF-NN ensemble ambiguity is small, and it increases as the T_i/T_e is scanned above or below the starting condition. Remarkably, the TGLF-NN ensemble ambiguity

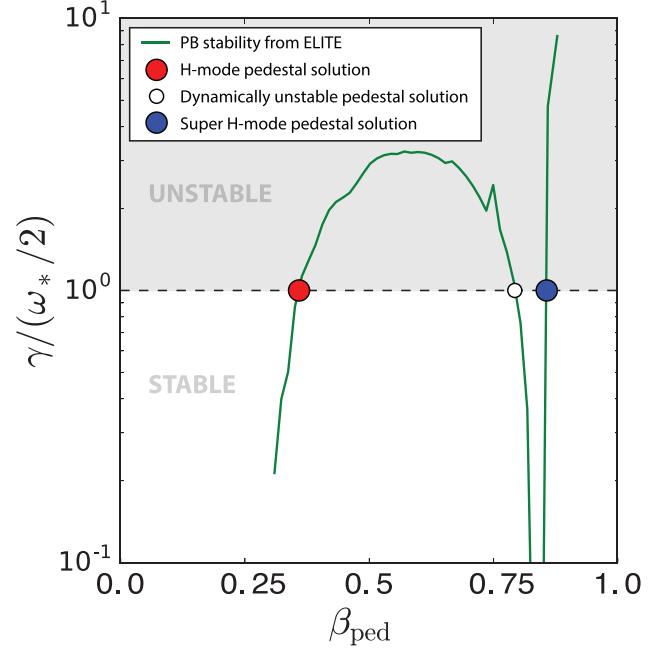


Figure 8. For each of the 100 self-consistent model equilibria of increasing pedestal height, the PB stability is evaluated for moderate to high toroidal mode numbers with the ELITE MHD stability code. The EPED1 solution is defined at the transition between a region of stability to one where the plasma becomes MHD unstable for one of the toroidal modes in the ELITE calculation. For strongly shaped plasmas above a critical density the EPED1 model predicts two stable solutions. This figure shows the growth-rates of the most unstable toroidal mode number as calculated by ELITE, for each of the model equilibria. The evaluation of the PB stability by the ELITE code can in some cases be affected by spurious non-physical crossings of the stability boundary (for the EPED1 simulation shown here this occurs for $\beta_n \geq 0.875$ and is not shown), which poses difficulties for the reliability of automated root finding routines. A newer version of ELITE that is under development aims at limiting such problematic behavior, along with other improvements.

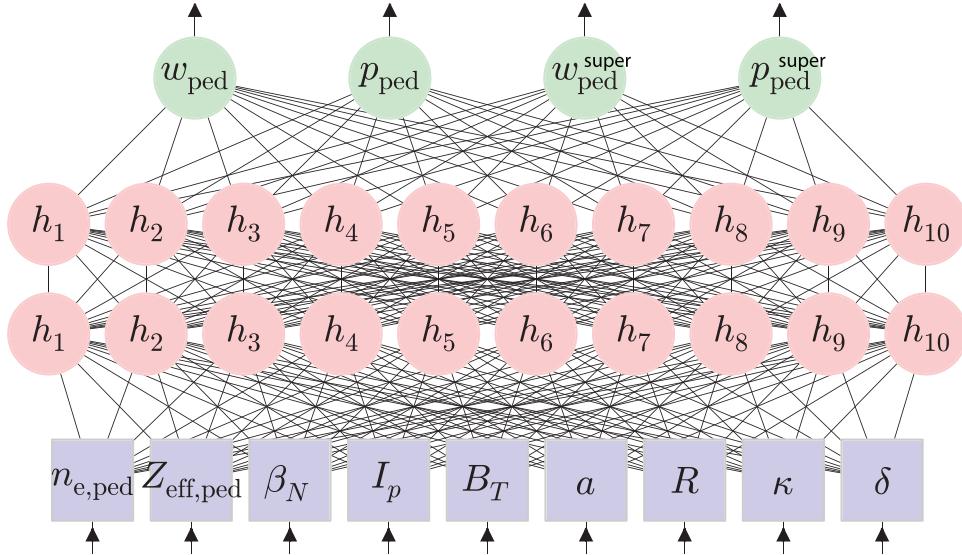


Figure 9. Visual representation of the NN topology (simplified in the number of hidden neurons) that is at the basis of the EPED1-NN model. The output of the EPED1-NN model is the average of an ensemble of 30 NN, each fed with the same input parameters values. The standard deviation of the ensemble—so called *ambiguity*—provides an estimate of the uncertainty in the EPED1-NN prediction.

encompasses the TGLF calculation, even well beyond the range of values of T_i/T_e that are experimentally observed on DIII-D at $\rho = 0.2$. The favorable extrapolation properties of the TGLF-NN model are accredited to the solution regularization and the ensemble averaging of multiple NNs.

3.4. Error analysis

Evaluating when the NN prediction is a reliable representation of the original data (or not) is of the utmost importance for the deployment of its applications. Figure 6 shows the regression plot for each of the outputs of the NN ensemble. The regression plot is obtained by feeding the validation data-set to the TGLF-NN regression and the TGLF models. The distribution of points along the line with slope unity shows that the NN-based model can accurately predict results of the full TGLF calculation.

The domain of the training database can have a very complicated topology in 23 dimensions, and looking at the training range in each of the input parameter dimensions is not enough to evaluate whether a point lays inside or outside of it. Definition of a multi-dimensional norm, and evaluation of the minimum distance of a point of interest from entries in the training database could be a possible strategy to provide an estimate on the accuracy of the NN prediction. However, this is not a computationally efficient strategy, both in terms of number of operations, as well as in terms of memory usage, since this would require access to the original training database.

Instead, we leverage the ambiguity information from the NN ensemble to efficiently estimate the error in the TGLF-NN regression model. The idea behind such estimation is to take advantage of the fact that NNs tend to become inaccurate outside the training range, and using the consensus among the NNs in the ensemble as a reflection of the goodness of their interpolation, or at least of their robust extrapolation.

Table 2. Plasma parameters which are input to the EPED1 and EPED1-NN models. Unlike TGLF, EPED1 uses dimensional parameters as inputs because the model is non-local and it includes the plasma separatrix: a region of plasma whose physics properties do not scale with the normalized gyro-radius, normalized collision frequency, and β as they do in the core.

Variable	Definition
$n_{e,\text{ped}}$	Pedestal electron density
$Z_{\text{eff,ped}}$	Pedestal effective ion charge
β_n	Volume-averaged normalized plasma pressure
I_p	Plasma current
B_T	Vacuum toroidal magnetic field at R
R	Geometric major radius
a	Geometric minor radius
κ	Geometric elongation
δ	Geometric triangularity
m_i	Main ion mass

The estimate of the relative error $\tilde{\epsilon}$ of the TGLF-NN model compared to the full TGLF calculation is obtained as the ratio between the average value of the ensemble \bar{V} normalized to the square root of the standard deviation of the ensemble outputs $\sqrt{\bar{a}}$ (the ensemble ambiguity of equation (2)) so that, for example, for the electron energy channel:

$$\tilde{\epsilon}_{Qe} = \frac{\sqrt{\bar{a}_{Qe}}}{|\bar{V}_{Qe}|}. \quad (3)$$

The plots in figure 7 compare the value of the actual relative error and its estimate. The following thresholds, above which the NN prediction is deemed to be inaccurate, are then defined based on analytic fits of the estimated NN regression relative errors:

$$\tilde{\epsilon}_{Qe} < \left(\frac{0.2}{\bar{V}_{Qe}} + 0.1 \right) K, \quad (4)$$

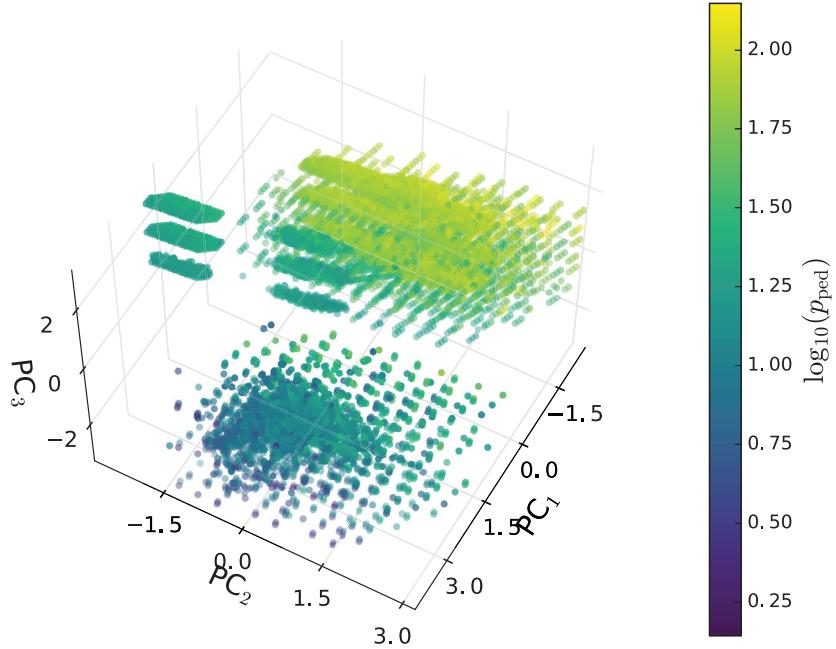


Figure 10. Three-dimensional visualization of the EPED1-NN training database, obtained by projecting the database on the three eigenvectors (PC_1 , PC_2 , and PC_3) which have the largest eigenvalues of a principal component analysis. The color of the data points represents the H-mode pedestal height.

$$\tilde{\epsilon}_{Qi} < \left(\frac{0.2}{\bar{V}_{Qi}} + 0.1 \right) K, \quad (5)$$

$$\tilde{\epsilon}_{Te} < \left(\frac{0.02}{\bar{V}_{Te}} + 0.2 \right) K, \quad (6)$$

$$\tilde{\epsilon}_{H\!i} < \left(\frac{0.1}{\bar{V}_{H\!i}} + 0.2 \right) K, \quad (7)$$

where we generally set the parameter K to 3.0 for realistic transport simulations.

4. Accelerated pedestal model

In the pedestal region, the very short radial scales associated with the equilibrium present a substantial challenge to traditional theoretical approaches. Because the observed fluctuation scales overlap strongly with the equilibrium scale, strictly speaking it is not possible to rigorously apply traditional transport theory. Despite the theoretical challenge, there has been substantial progress in understanding mechanisms which constrain the structure of the pedestal. In particular, the EPED model was developed to predict the pedestal height and width in H-mode plasmas [3, 31, 32]. The predictive capabilities of this model were highlighted by its use to discover an enhanced pedestal stability regime named Super H-mode [33].

The EPED model, even in its most simple incarnation—the EPED1 model [3]—, requires on the order of 10 CPU-hour to be executed and is thus too computationally intensive to be employed for routine WDM simulations. The fact that

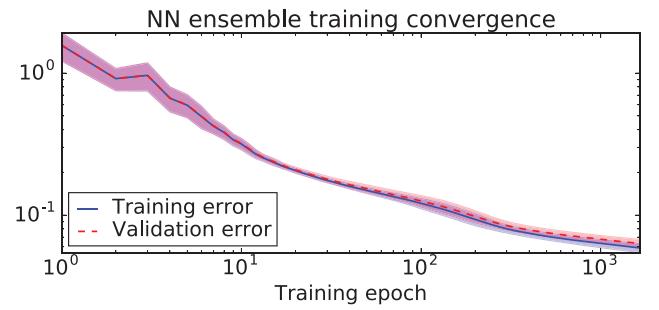


Figure 11. Convergence of the root-mean-square training and validation errors for the EPED1-NN model NN ensemble. The average and standard deviation of the NN ensemble is shown as shaded. As discussed in figure 4, the training and validation error closely follow one another. It is normal that the training error slightly underestimates the validation error, since that is the data upon which the NN is trained on. Also in this case, the validation error does not increase, which indicates that the EPED1-NN is not over-trained.

only recently, a few self-consistent first-principles coupled core-pedestal simulations have been carried out is a testament to the difficulty of the computational task at hand [34]. EPED combines two physics constraints to predict the pedestal height and width. Specifically, the stability of peeling ballooning (PB) modes provides a relation on the height as a function of the width of the pedestal. In addition, EPED1 operates under the assumption that because of the very strong $\mathbf{E} \times \mathbf{B}$ shear and generally strong particle and heat sources in the pedestal transport barrier, the pressure gradient generally rises until it is finally limited by the onset of the nearly-local kinetic ballooning modes (KBM).

From an algorithm perspective, the EPED1 model starts from a series of roughly 100 model equilibria constructed

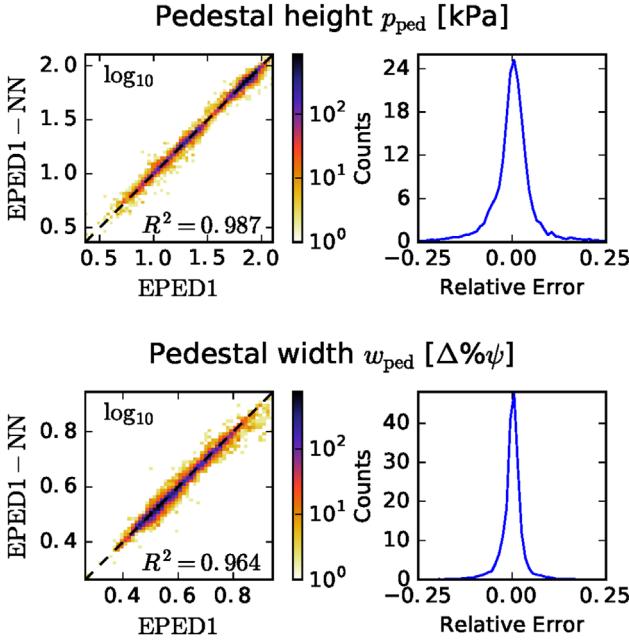


Figure 12. On the left, histogram of the correlations comparing the H-mode p_{ped} and w_{ped} as evaluated by the EPED1-NN model and the full EPED1 calculations. On the right, histograms of the relative error (101 bins in the ± 0.5 range) show that the relative error on the regression of the pedestal widths is smaller than the one of the pedestal heights.

using the TOQ [35] equilibrium solver. These equilibria have increasing pedestal height, and a width that is proportional to the square-root of the pedestal pressure, based on an analytic model for the KBM constraint [3, 31]. For each of these equilibria, the PB criticality is evaluated for a set of moderate to high toroidal mode numbers with the ELITE code [36–38] (typically 7 toroidal modes ranging from 5 to 30). The EPED1 solution is defined as the equilibrium (and the corresponding pressure profile) at which the PB stability reaches a critical condition. Figure 8 shows the EPED1 PB stability plot for equilibria of increasing pedestal height that satisfy the KBM constraint.

For strongly shaped plasmas, above a critical density, the EPED1 model predicts two solutions for which the pedestal is limited by the PB and KBM constraints [33]. The solution at the lowest pedestal pressure is the standard H-Mode pedestal, while the solution at the highest pedestal pressure is the so called super-H mode. The existence of such bifurcation in the allowed pedestal structure has been confirmed experimentally on DIII-D [39]. Recent simulations have shown that the Super H-mode pedestal can substantially improve the ITER fusion performance, and can also be advantageous for a steady-state power plant, by providing a path to increasing the bootstrap current while simultaneously reducing the demands on the core physics performance [39].

Assembly of the training database is made possible by the translation as part of the SciDAC AToM project [40] of the original IDL based EPED1 workflow [3, 31] to high performance computing (HPC) capable workflow driven by the integrated plasma simulator (IPS) [41, 42] framework. This IPS-EPED1 workflow replicates the original EPED1

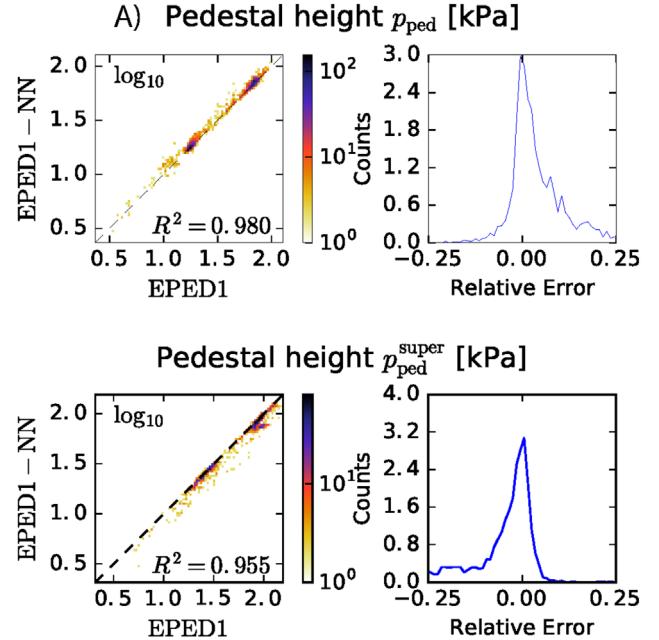


Figure 13. Regression for the (A) H-mode p_{ped} and (B) Super H-mode $p_{\text{ped}}^{\text{super}}$ outputs, only for the data points where the two roots are not degenerate.

capability, but at a fraction of the wall clock time. A typical IPS-EPED1 iteration uses up to 700 cores on the Edison Cray XC30 platform at NERSC [43], and simulation results are obtained in less than 2 min (limited by the time required for one TOQ run plus one ELITE run for the highest mode number being considered). Multiple IPS-EPED1 executions are dispatched concurrently within a single instance of the IPS, by leveraging the integrated IPS-DAKOTA optimization and parameter sweep environment [44]. Seamless data collection of the simulation results is provided by an interface in the IPS-EPED1 workflow to the HARVEST architecture described in section 2.

4.1. EPED1-NN

The NN topology displayed in figure 9 was used to reproduce the pedestal structure results of the full EPED1 model. The results presented in this paper were obtained with an ensemble of 30 NNs, each with 2 hidden layers of 10 neurons each. The outputs of each NN are the pedestal pressure height (p measured in Pa) and width (w measured in percentage of the normalized poloidal flux) for the H-mode (p_{ped} and w_{ped}) and super H-mode roots ($p_{\text{ped}}^{\text{super}}$ and $w_{\text{ped}}^{\text{super}}$). The inputs to the NN are the set of plasma parameters that are input to the original EPED1 model, as listed in table 2. Here the volume-averaged normalized plasma pressure is defined as $\beta_n \doteq \beta_T/I_n$ [45], where

$$\beta_T = \frac{\langle p \rangle}{B_t^2/2\mu_0} \quad \text{and} \quad I_n = \frac{I_p}{aB_t}. \quad (8)$$

The model density and temperature profiles of the TOQ equilibria that are part of the EPED1 workflow have a

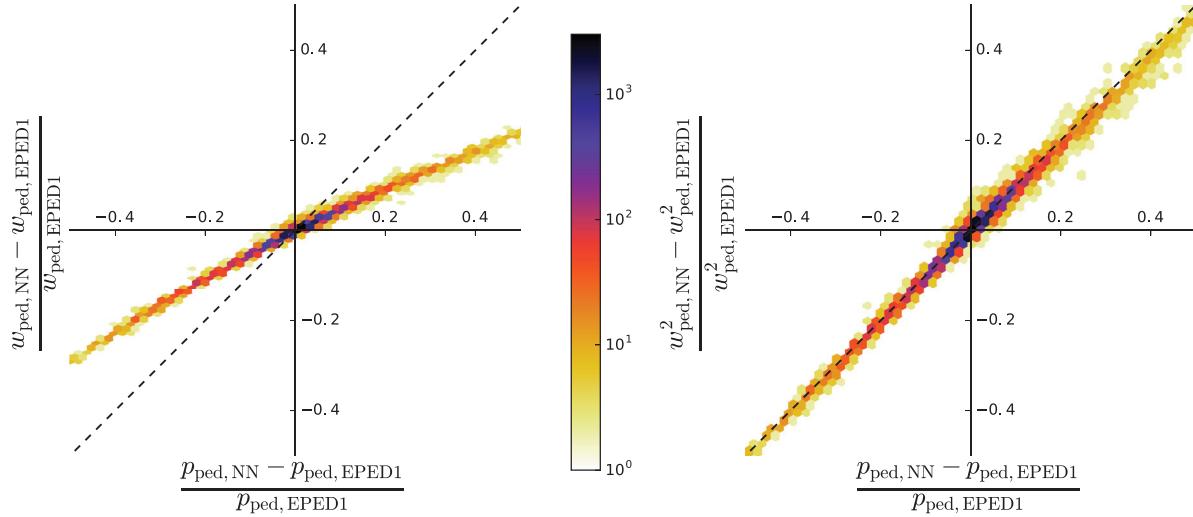


Figure 14. On the left: two-dimensional histogram of the H-mode pedestal width and height relative errors shows that these two quantities are non-linearly related to each other, and that the former is smaller than the latter. On the right: the relative error of the square of the pedestal width is shown to be directly proportional to the relative error of the pedestal height.

functional form of a hyperbolic tangent shape in the pedestal, and a simple polynomial dependence in the core [46]. The density and temperature profiles are obtained from the output of the EPED1-NN model, by using the same functional representation that the TOQ equilibrium uses for EPED1 calculations. The availability of these profiles is required to perform core-pedestal coupled simulations described in section 5.

4.2. Database assembly

Within the operational space of individual devices, the characteristic pedestal height and width do not exhibit sharp structures as a function of the input parameters. Furthermore, the inputs to the EPED1 model are global (as opposed to the local parameters of the TGLF-NN model) and many of these have a relatively small variation over the operational range of interest for different devices.

Based on these observations, the training database was assembled by performing uniform scans in a few parameters with a limited number of samples for each of the scanned parameter. For example, EPED1 scans for the ITER D-D scenario at full-field were performed by varying $n_{e,\text{ped}}$, $Z_{\text{eff,ped}}$, β_n , κ and δ in 5 steps each. This approach was used to generate EPED1 predictions for ranges of input parameters that are characteristics of JET (200 cases), KSTAR (700 cases), and ITER (15 000 cases). For the DIII-D tokamak, a data-set of 3000 cases was assembled by executing the EPED1 model based on the input parameters of the discharges that were carried out during the 2014 experimental campaign, each evaluated at the time of the discharge when the normalized global pressure β_n was maximum. The aggregated database resulted in a total of $\sim 20\,000$ EPED1 runs.

Figure 10 shows the graphical representation of the training database in 3D space. The dimensionality reduction from ten to three dimensions was obtained by projecting the database on the principal component analysis eigenvectors with the highest eigenvalue. The plot highlights the sparse nature of the

database, with regions of space where the points are densely packed, and others that are void. These clusters correspond to the individual ranges of input parameters that were used for each of the devices in the database. The uniform sampling of the domain for some of these clusters is also apparent.

4.3. Training

The ensemble of 30 NNs that compose the EPED1-NN model are trained following the procedure described in section 2. The outputs for the H-Mode and Super H-mode solutions are made to coincide when they are degenerate. The training convergence properties of the ensemble are illustrated in figure 11, and show similar logarithmic-like training convergence rate as observed in figure 4 for the TGLF-NN model (section 3.2), thus suggesting that also in this case good generalization properties ought to be expected.

On a 3.1 GHz processor workstation the time required to train one NN is about 8 min, while the evaluation of the NN outputs is in the order of few μs . Compared to the 25 CPU-hour required by the full EPED1 model, NN-based pedestal model provides a computational saving of over nine orders of magnitude.

4.4. Regression and relative error analysis

We find that a single NN can be trained to capture the EPED1 predictions across an input parameter range that spans multiple devices, as shown in the regression plot in figure 12. The H-Mode and Super H-Mode solutions of the pedestal height and width as predicted by the EPED1-NN ensemble and the original EPED1 model are in excellent agreement, as captured by the high value of the coefficient of determination R^2 . Figure 13 illustrates the same regression plot but considering only the data points for which the H-mode and Super H-Mode solutions are distinct. The H-mode solution tends to over-estimate, whereas the Super H-mode solution tends to

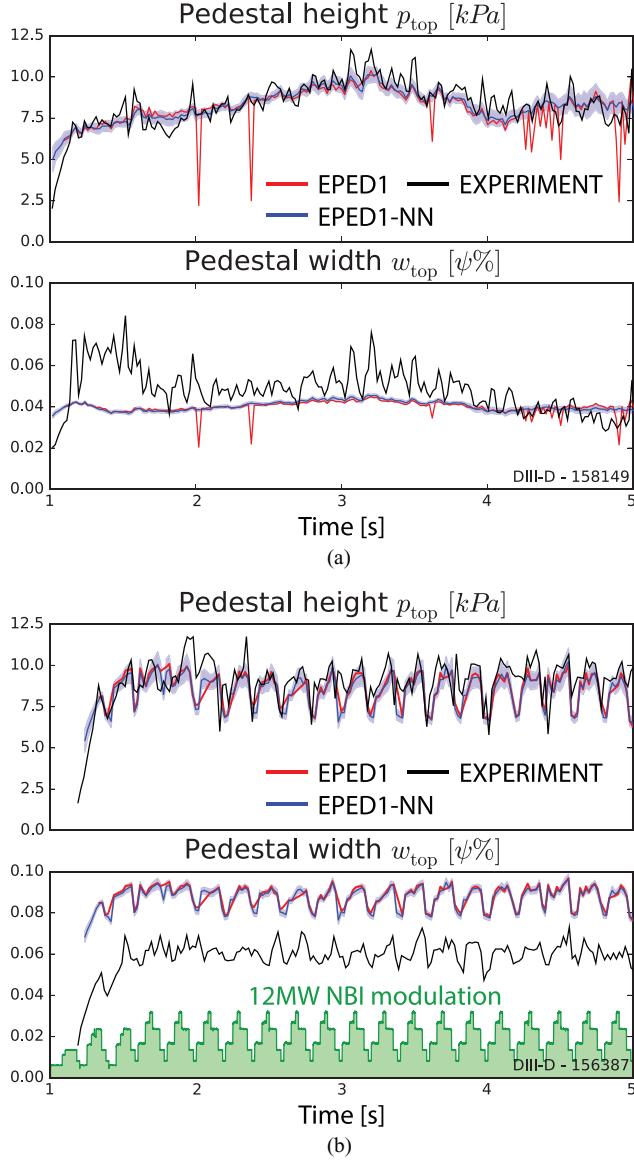


Figure 15. Evolution of the pedestal height and width for DIII-D discharges 158149 (a) and 156387 (b). Both figures illustrate how the EPED1-NN model closely reproduces the results of the full EPED1 calculation. The efficiency of EPED1-NN enables routine comparison with the experiments, and thus (indirect) validation of the EPED1 model. For shot 156387, although the EPED1 model matches the experimental pedestal height very well, the predicted width is 25% lower than the experiment. This result could be improved by more sophisticated EPED models that compute more accurately the pedestal physics [31, 32].

under-estimate. This effect is understood in terms of the NN regularization.

Some of the scatter that is observed in the regression plot is associated with algorithmic uncertainty of the EPED1 model, which is mainly due to the discrete pedestal pressure increments between subsequent TOQ equilibria, and the sparse increments of the toroidal mode numbers used for the ELITE stability calculation.

Interestingly, the histograms in figure 12 show the relative error of the pedestal width is significantly tighter than the one of the pedestal height. This effect can be understood recalling that for the EPED1 model the pedestal height and width are

related to each other via the analytic KBM constraint condition $w_{\text{ped}} = c \beta_{\theta, \text{ped}}^{1/2}$. Here $\beta_{\theta, \text{ped}} = \langle p_{\text{ped}} \rangle / (B_\theta^2 \mu_0)$ is the volume-averaged plasma pressure normalized to the poloidal magnetic pressure, and c , which is generally a weakly varying function of dimensionless parameters, is fixed to a constant in EPED1, using a value $c = 0.076$ that fits KBM constraint calculations for a set of standard aspect ratio cases (DIII-D, JET, and ITER) [31]. The two dimensional histograms in figure 14 verify that the relative error of the pedestal height and widths are related to one another by the same analytic expression. This observation provides an indirect consistency check, which confirms that EPED1-NN captures the simple pedestal height-to-width relationship of the original EPED1 model. Yet, the approach presented here is generic, and could be used to fit more advanced versions of EPED, for which the pedestal height-to-width relationship is evaluated individually for each case by numerical solution of the KBM stability constraint [31].

4.5. Application and validation

To assess the performance of EPED1-NN we compared it against both the original EPED1 model as well as against experimental measurements of the pedestal structure. Figure 15 illustrates the comparison for two DIII-D shots which were not part of the original EPED1-NN training database. The NN model can be seen to follow very closely the results of the full EPED1 calculation. The NN accelerated model makes a prediction for the entirety of the discharge in less than a second, while the full calculation requires 140 000 CPU-hour to run.

The glitches in the EPED1 solution of figure 15(A) arise from inaccurate automatic detection of the pedestal stability threshold within the EPED1 algorithm (see figure 8). These artifacts would require manual intervention to be properly sanitized, which could be problematic for integrated simulations that rely on the robustness of the pedestal structure calculation. In comparison the NN model is robust and smooth. The shaded blue area represents the EPED1-NN ensemble ambiguity, which is relatively small, thus suggesting that the predicted data falls within the domain of the training database.

The pedestal response to a 12 MW neutral beam modulation is observed both in the experimental and simulated time traces of figure 15(B). Throughout the discharge the pedestal density and effective ion charge are relatively constant, suggesting that the response of the pedestal height to the NBI core heating (i.e. β_n) is occurring via global Shafranov shift stabilization of the PB modes. Although the experimental pedestal height is matched very well by the EPED1 model, the predicted width is 25% lower than the experiment. This result could be improved by more sophisticated EPED models that compute more accurately the pedestal physics [31, 32].

Finally, the efficiency of the EPED1-NN model has enabled the development of an interactive tool for the exploration of the pedestal structure, as depicted in figure 16. The graphics shows the EPED1-NN predicted pedestal height and width H-Mode (red) and Super H-mode (blue) solutions for plasma parameters that are compatible with ITER operations.

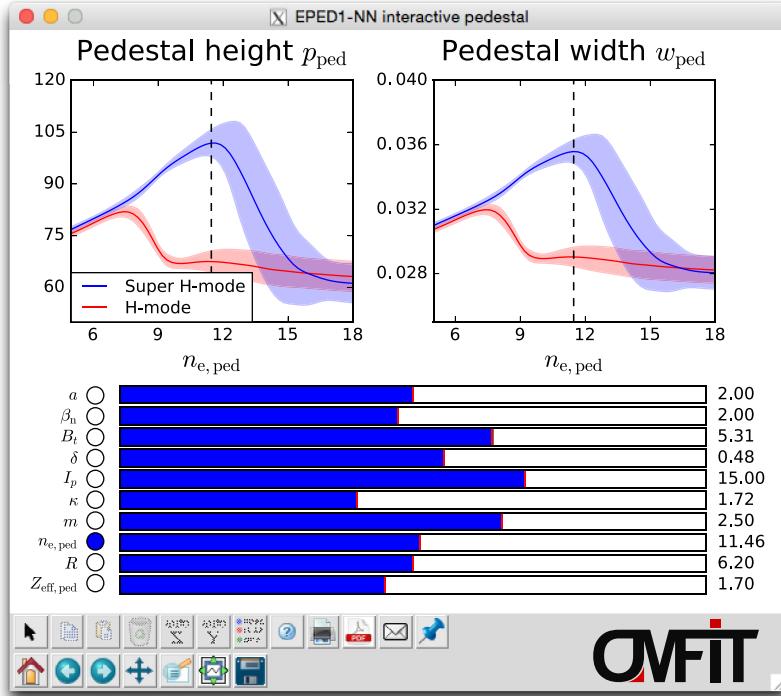


Figure 16. Development of the EPED1-NN model has enabled the interactive display of the pedestal characteristics as a function of the model input parameters. This figure shows the tool that is used within the OMFIT framework for exploring and optimizing the predicted pedestal height and width. For this particular set of ITER-like parameters it can be seen that the H-mode (red) and Super H-mode (blue) roots split at a pedestal density of about $7 \times 10^{19} \text{ m}^{-3}$. The Super H-mode root reaches a peak near $11.5 \times 10^{19} \text{ m}^{-3}$ with a gain of over $\sim 30\%$ over the peak of H-mode solution at lower density. The H-Mode and Super H-mode predictions coincide when the two pedestal solutions are degenerate.

5. Accelerated self-consistent core-peDESTAL coupled simulations

Accurate prediction of fusion performance in present and future tokamaks requires taking into account the strong interplay between core transport, pedestal structure, current profile, and plasma equilibrium. In [34] the authors have reported on the development of a workflow capable of calculating the steady-state self-consistent solution of this strongly-coupled problem. In the original iterative workflow self-consistency is achieved by successive iterations of the full EPED1 pedestal model, the core transport solver TGYRO [4] (running TGLF and NEO for the turbulent and neoclassical fluxes), the energy, particle, and momentum sources calculated via the ONETWO transport code, and the equilibrium solver EFIT. The coupling scheme was shown to closely reproduce the experimental measurements *without* requiring *a priori* knowledge of the temperature and density profiles. Further, it was demonstrated that the self-consistent, interactive solution is independent of the initial guess that is used to initiate the pedestal calculation at the beginning of the iterative process.

The workflow described in [34] is computationally demanding, since it uses the full TGLF model for simulating the core turbulent transport and full EPED1 to model the pedestal stability. By leveraging the TGLF-NN and EPED1-NN models the most significant computational bottlenecks can be removed. Rather than solving the full drift-kinetic problem via

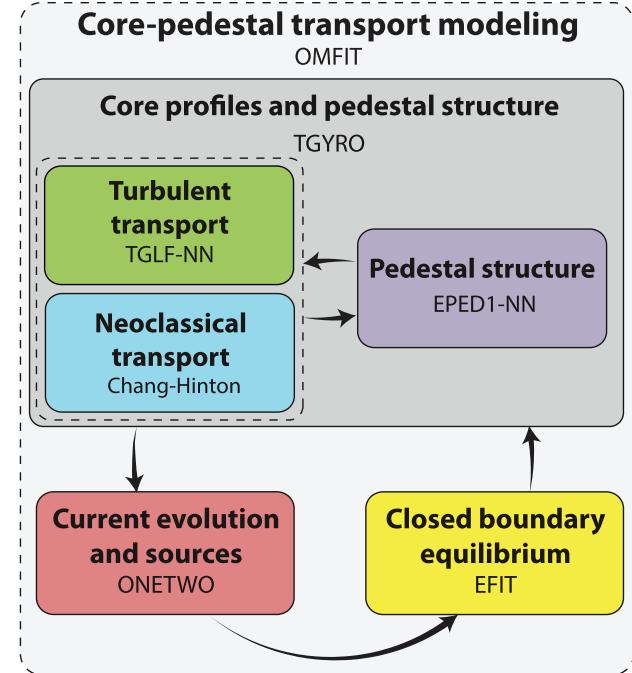


Figure 17. The self-consistently coupled core-peDESTAL workflow accelerated by the NN models mimics the workflow originally implemented in [34]. To reduce the overhead of exchanging data between the core and pedestal solutions, and fully leverage the speed of the NN models, the EPED1-NN model was directly integrated within the TGYRO transport solver.

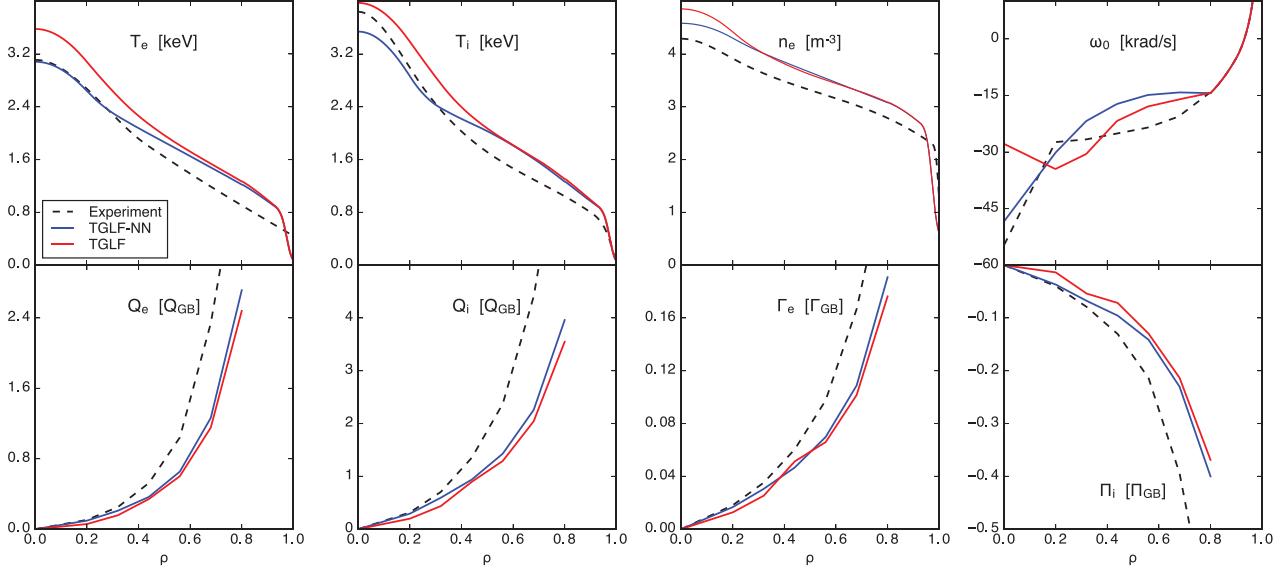


Figure 18. Comparison of the predicted electron and ion temperatures, the density, rotation profiles and their respective fluxes for DIII-D shot 145452 at 1245 ms using TGLF-NN (blue) and TGLF (red) as turbulent transport models. Comparison with the experimental profiles (dashed) shows good agreement. It is normal that, although both the TGLF-NN and TGLF have well converged, their fluxes differ with one another and with the experimentally inferred fluxes, since the different density and temperature profiles result in a different gyro-Bohm normalization factor. The energy exchange between ions and electrons also contributes to the differences in the energy fluxes.

the NEO model [17] (the NEO calculation requires about the same amount of resources as TGLF), the neoclassical transport calculation is significantly sped up by using the Hinton-Hazeltine [47] model for the electron energy flux and the electron and ions particle flux, and the Chang-Hinton [48] model for the neoclassical ion energy flux. For the purpose of the class of DIII-D discharges analyzed in this paper, it was found that the neoclassical transport fluxes are negligible compared to the turbulent fluxes, independently of the model used. The details of the neoclassical models as implemented in TGYRO are described in [17].

To obtain further speedup, we modified the original iterative workflow of [34] by self-consistently coupling the core and pedestal solutions within the internal iteration loop of the TGYRO transport solver itself, as shown in figure 17. According to this scheme, the pedestal solution (that is the boundary condition for the core solution) is updated after each step of the TGYRO flux-matching calculation. As a result, the core-profiles and pedestal structure that is output by TGYRO are now self consistent with one another, before being iterated with the sources and equilibrium calculations. This approach is advantageous because the convergence of the core-pedestal profiles is now decoupled from the convergence of the larger sources-equilibrium-profiles system. The modified iteration scheme trades more pedestal structure calculations—which have virtually no computational cost using the EPED1-NN model—for fewer calculations of the sources and equilibrium, thus resulting in significant computational savings. We point out that the core-pedestal coupling scheme within TGYRO has been achieved with a transition region that allows the gradient scale lengths that are consistent with the transport calculation in the core to smoothly transition to the values that

are consistent with the dynamics of the pedestal. This is the same technique that was first introduced in [34].

5.1. Verification and validation

The updated TGYRO coupled core-pedestal workflow has been validated against the experimental measurements of 200 randomly sampled time-slices within the DIII-D ion-stiffness database, with very positive results. All TGYRO simulations were carried out evolving the temperatures, density, and rotation profiles. The core flux matching calculations were carried out at six radial locations equally distributed in ρ from 0.2 to 0.8. The pedestal was also dynamically evolved with the EPED1-NN model. The results are evaluated after 30 iterations of the internal TGYRO Newton solver.

We found that convergence of the TGYRO solution using the TGLF model to be significantly more difficult to achieve than with its NN-based counterpart. This result is attributed to the fact that TGLF-NN provides a smooth representation of the TGLF fluxes. As a result, of 200 times-slices, 165 (82%) converged to a residual lower than 10^{-2} using the TGLF-NN model, whereas only one (0.5%) did with TGLF. This is an important aspect, as problems for which we could not find a solution before now converge. In view of the limited availability of converged simulations with TGLF, we were limited to comparing TGLF-NN versus TGLF, only for the time-slice for which TGYRO converged with both models (DIII-D shot 145452 at 1245 ms).

Figure 18 compares the predicted electron and ion temperatures, the density, rotation profiles and their respective fluxes using TGLF-NN (blue) and TGLF (red) as turbulent transport models. We find that the absolute value of the fluxes are within 0.01 gyro-Bohm units between the two models, which

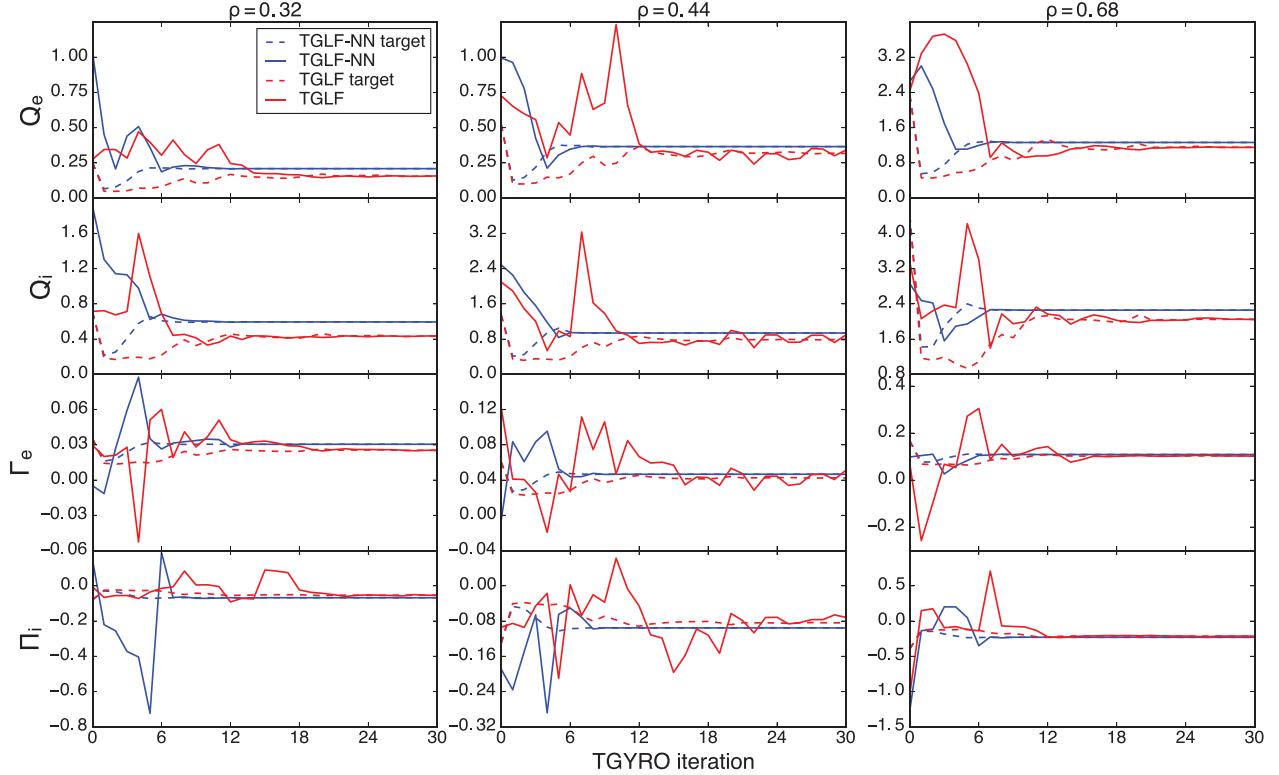


Figure 19. Convergence of the TGYRO solution for the fluxes of DIII-D shot 145452 at 1245 ms at 3 radial locations ($\rho = 0.32, 0.44$, and 0.68). The TGLF-NN fluxes in TGYRO match their target value in fewer iterations than with TGLF. This plot highlights the excellent properties of TGLF-NN, which converges steadily and with a final residual that is two orders of magnitude lower than with the TGLF model.

when integrated result in small differences in the core profiles. Overall the TGLF-NN fluxes appear to be smoother than the ones calculated by TGLF, which is a further indirect indication that the output of the NN model is a smoother function of the local plasma parameters than TGLF itself. Perhaps not surprisingly, the profiles are nearly identical in the pedestal region, in view of the fact that the volume averaged β_n is very similar between the two TGYRO simulations, and thus are also the input parameters to the EPED1-NN model.

To better analyze the different convergence behavior between the two models, the evolution of the turbulent fluxes as function of iteration is depicted in figure 19 for the electron energy flux Q_e at the core, mid, and edge radial locations. Here convergence is achieved when the energy turbulent transport fluxes from either the TGLF or TGLF-NN models match the target flux, that is the value of the flux that matches the level of the energy sources and sinks in the plasma. The TGYRO solution converges differently for the two models, with residuals of 4×10^{-5} for TGLF-NN, and 7×10^{-3} for TGLF. The figure also shows that the TGLF-NN fluxes in TGYRO match their target value in fewer iterations than with TGLF.

The robustness of the transport solution of the same DIII-D discharge was further investigated by perturbing the initial ion temperature profile that is input to TGYRO. In particular, the ion temperature was multiplied and divided by a function

$$(1 - \rho/0.8)^2 + 1$$

for $\rho < 0.8$. Such function was chosen so to ensure smooth profiles while maintaining the boundary condition at $\rho = 0.8$ of the TGYRO transport simulation fixed. Figure 20 shows the evolution of the ion temperature as a function of the TGYRO iterations for the different initial conditions. This analysis highlights how the TGYRO solution with the TGLF-NN model is independent of the initial ion temperature profile. By contrast, TGYRO with the TGLF transport model takes longer to converge, and the ion temperature profile at the last iteration are significantly different depending on the initial condition of the simulation.

A comprehensive statistical analysis of the 165 discharges that converged with the TGLF-NN model shows that TGYRO simulations with the self-consistently coupled TGLF-NN and EPED1-NN models predict a solution that is consistent with the experimental measurements. Figure 21 shows the median of the relative error as a function of radius between the predicted electron and ion temperatures, the density and rotation profiles, and the experimentally measured ones. The majority of the NN-based simulations converged to a residual on the order of 10^{-4} . Statistically, relatively small but systematic errors are observed: a 25% under-prediction of the electron temperature at $\rho = 0.8$; a 15% under-prediction of the ion temperature near the axis; and a 15% over-prediction of the rotation at mid-radius. Globally the β_N is on average 90% of the experimental value.

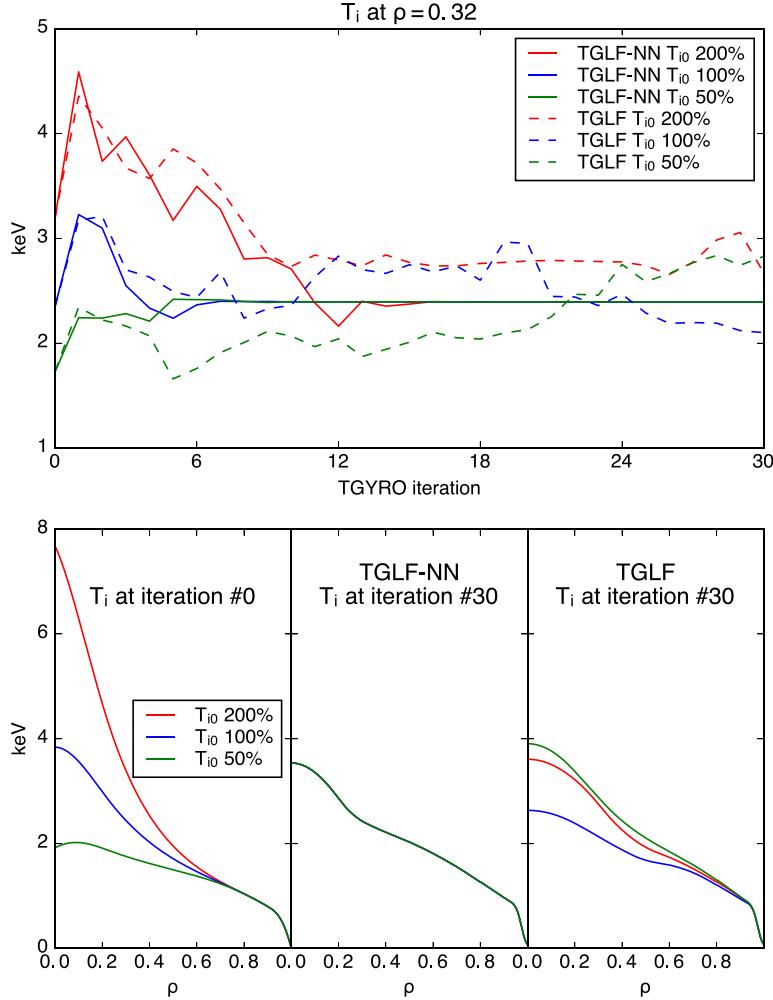


Figure 20. Comparison of the TGYRO evolution of the ion temperature using the TGLF and TGLF-NN turbulent transport models, starting from initial profiles that on axis are 200% (red) and 50% (green) of the original experimental profile (blue). This perturbation analysis shows that not only TGLF-NN converges faster, but also that the solution found by TGYRO is independent of the initial ion temperature profile.

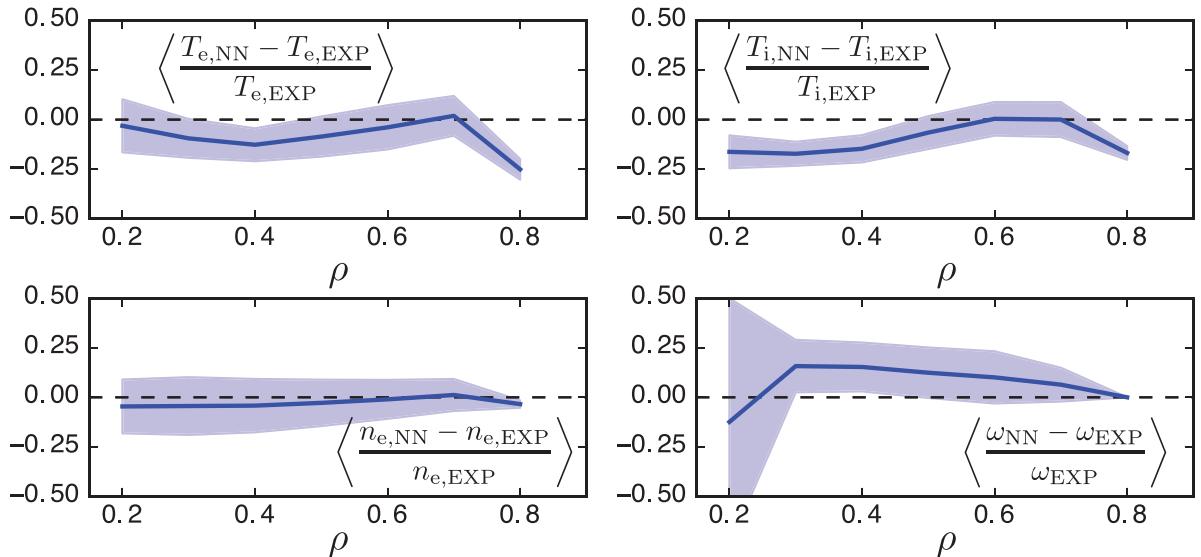


Figure 21. Two hundred randomly sampled time-slices in the DIII-D ion-stiffness database were simulated using TGLF-NN as the turbulence model, and the pedestal predicted by the EPED1-NN model. This figure illustrates the relative error statistics of the evolved profiles for the 165 self-consistent core-pedestal TGYRO simulations that fully converged.

6. Conclusions

We reported on the development of neural-network based models that are designed to accelerate two of the most critical aspects for functional whole device modeling applications: (1) the prediction of the turbulent transport fluxes in the core of the plasma; and (2) the H-mode plasma pedestal structure. Each of these models has been systematically analyzed and verified against their respective first-principles calculation.

Results from the coupled NN-based predictive workflow show that the NN paradigm is capable of breaking the speed-accuracy trade-off that is expected of traditional numerical physics models, and provides the missing link towards self-consistent coupled core-pedestal WDM simulations that are physically accurate, robust, and extremely efficient to run.

Acknowledgments

The authors would like to acknowledge the useful discussions with the *transport and confinement*, the *pedestal and edge physics*, and *integrated operation scenarios* International Tokamak Physics Activity (ITPA) topical groups.

For this study the workflow for fetching the data, creating and training the network and postprocessing the results is implemented as part of the OMFIT [9] integrated modeling framework. The NNs were implemented based on the FANN [49] library.

Work supported by the Office of Science of the US Department of Energy under Contract No. DE-SC0012656 (GA AToM SciDAC), DE-AC05-00OR22725 (ORNL AToM SciDAC), DE-FG02-95ER54309 (GA theory), DE-FC02-06ER54873 (ESL), DE-FC02-04ER54698 (DIII-D). This research used resources of the National Energy Research Scientific Computing Center (NERSC), a DOE Office of Science User Facility supported by the Office of Science of the US Department of Energy under Contract No. DE-AC02-05CH11231.

ORCID

F. Poli  <https://orcid.org/0000-0003-3959-4371>

References

- [1] Bonoli P. and McInnes L. 2015 Workshop on integrated simulations for magnetic fusion *Technical Report*, United States Department of Energy
- [2] Staebler G., Kinsey J. and Waltz R. 2007 *Phys. Plasmas* **14** 055909
- [3] Snyder P., Groebner R., Leonard A., Osborne T. and Wilson H. 2009 *Phys. Plasmas* **16** 056118
- [4] Candy J., Holland C., Waltz R., Fahey M. and Belli E. 2009 *Phys. Plasmas* **16** 060704
- [5] Widrow B. and Lehr M. 1990 *Proc. IEEE* **78** 1415
- [6] MacKay D. 2003 *Information Theory, Inference and Learning Algorithms* (Cambridge: Cambridge University Press)
- [7] Moody J., Hanson S., Krogh A. and Hertz J.A. 1995 *Adv. Neural Inf. Process. Syst.* **4** 950
- [8] Krogh A. et al 1995 Neural network ensembles, cross validation, and active learning *Adv. Neural Inf. Process. Syst.* **7** 231–8
- [9] Meneghini O. et al 2015 *Nucl. Fusion* **55** 083008
- [10] Howard N.T., Holland C., White A.E., Greenwald M., Candy J. and Creely A.J. 2016 *Phys. Plasmas* **23** 056109
- [11] Kinsey J., Staebler G. and Waltz R. 2008 *Phys. Plasmas* **15** 055908
- [12] Bateman G., Kritz A., Kinsey J., Redd A. and Weiland J. 1998 *Phys. Plasmas* **5** 1793
- [13] Waltz R., Staebler G., Dorland W., Hammett G., Kotschenreuther M. and Konings J. 1997 *Phys. Plasmas* **4** 2482
- [14] Rafiq T., Kritz A.H., Weiland J., Pankin A.Y. and Luo L. 2013 *Phys. Plasmas* **20** 032506
- [15] Staebler G., Kinsey J. and Waltz R. 2005 *Phys. Plasmas* **12** 102508
- [16] Candy J. and Waltz R. 2003 *J. Comput. Phys.* **186** 545
- [17] Belli E. and Candy J. 2008 *Plasma Phys. Control. Fusion* **50** 095010
- [18] Belli E. and Candy J. 2009 *Plasma Phys. Control. Fusion* **51** 075018
- [19] Belli E. and Candy J. 2010 *Phys. Plasmas* **17** 112314
- [20] Rhodes T. et al 2011 *Nucl. Fusion* **51** 063022
- [21] Holland C., Petty C., Schmitz L., Burrell K., McKee G., Rhodes T. and Candy J. 2012 *Nucl. Fusion* **52** 114007
- [22] Citrin J. et al and JET-EFDA Contributors 2014 *Nucl. Fusion* **54** 023008
- [23] Navarro A.B., Happel T., Grler T., Jenko F., Abiteboul J., Bustos A., Doerk H., Told D. and ASDEX Team 2015 *Phys. Plasmas* **22** 042513
- [24] White A.E. et al 2015 *Phys. Plasmas* **22** 056109
- [25] Miller R.L., Chu M.S., Greene J.M., Lin-Liu Y.R. and Waltz R.E. 1998 *Phys. Plasmas* **5** 973
- [26] Citrin J., Breton S., Felici F., Imbeaux F., Aniel T., Artaud J., Baiocchi B., Bourdelle C., Camenen Y. and Garcia J. 2015 *Nucl. Fusion* **55** 092001
- [27] Holland C. et al 2013 *Nucl. Fusion* **53** 083027
- [28] Pereverzev G. and Corrigan G. 2008 *Comput. Phys. Commun.* **179** 579
- [29] Fable E. et al 2013 *Plasma Phys. Control. Fusion* **55** 124028
- [30] Sommer F. et al 2015 *Nucl. Fusion* **55** 033006
- [31] Snyder P., Groebner R., Hughes J., Osborne T., Beurskens M., Leonard A., Wilson H. and Xu X. 2011 *Nucl. Fusion* **51** 103016
- [32] Snyder P., Osborne T., Burrell K., Groebner R., Leonard A., Nazikian R., Orlov D., Schmitz O., Wade M. and Wilson H. 2012 *Phys. Plasmas* **19** 056115
- [33] Snyder P., Solomon W., Burrell K., Garofalo A., Grierson B., Groebner R., Leonard A., Nazikian R., Osborne T. and Belli E. 2015 *Nucl. Fusion* **55** 083026
- [34] Meneghini O. et al 2016 *Phys. Plasmas* **23** 042507
- [35] Lao L.L., Hirshman S.P. and Wieland R.M. 1981 *Phys. Fluids* **24** 1431
- [36] Snyder P., Wilson H., Ferron J., Lao L., Leonard A., Osborne T. and Turnbull A. 2002 *Phys. Plasmas* **9** 2037
- [37] Wilson H., Snyder P. and Huysmans G. 2002 *Phys. Plasmas* **9** 1277
- [38] Snyder P. et al 2007 *Nucl. Fusion* **47** 961
- [39] Solomon W.M. et al 2016 *Phys. Plasmas* **23** 056105
- [40] AToM SciDAC project, <http://scidac.github.io/atom/>
- [41] Elwasif W.R., Bernholdt D.E., Shet A.G., Foley S.S., Bramley R., Batchelor D.B. and Berry L.A. 2010 The design and implementation of the SWIM integrated plasma simulator *18th Euromicro Int. Conf. on Parallel, Distributed and Network-Based Processing* (IEEE) pp 419–27

- [42] Elwasif W.R., Bernholdt D.E., Foley S.S., Shet A.G. and Bramley R. 2011 Multi-level concurrency in a framework for integrated loosely coupled plasma simulations *9th IEEE/ACS Int. Conf. on Computer Systems and Applications (December 2011)* pp 188–95
- [43] NERSC National Energy Research Scientific Computing Center, www.nersc.gov
- [44] Elwasif W.R., Bernholdt D.E., Pannala S., Allu S. and Foley S.S. 2012 *IEEE 15th Int. Conf. on Computational Science and Engineering (December 2012)* pp 102–10
- [45] ITER Team and Others 1999 *Nucl. Fusion* **39** 2137
- [46] Snyder P., Wilson H., Ferron J., Lao L., Leonard A., Mossessian D., Murakami M., Osborne T., Turnbull A. and Xu X. 2004 *Nucl. Fusion* **44** 320
- [47] Hinton F. and Hazeltine R. 1976 *Rev. Mod. Phys.* **48** 239
- [48] Chang C. and Hinton F. 1982 *Phys. Fluids* **25** 1493
- [49] Nissen S. 2003 Implementation of a fast artificial neural network library (FANN) *Report*, Department of Computer Science University of Copenhagen (DIKU) vol 31