Real-time capable first principle based modelling of tokamak turbulent transport

J. Citrin[1,2], S. Breton[2], F. Felici[3], F. Imbeaux[2]
T. Aniel[2], J.F. Artaud[2], B. Baiocchi[4], C. Bourdelle[2], Y. Camenen[5], J. Garcia[2]
[1]*FOM Institute DIFFER – Dutch Institute for Fundamental Energy Research, P.O. box 6336, 5600 HH Eindhoven, The Netherlands*
[2]*CEA, IRFM, F-13108 Saint Paul Lez Durance, France*
[3]*Eindhoven University of Technology, Department of Mechanical Engineering, Control Systems Technology Group, P.O. Box 513, 5600MB, Eindhoven, The Netherlands*
[4]*Istituto di Fisica del Plasma "P. Caldirola", Associazione Euratom-ENEA-CNR, Milano, Italy*
[5]*Aix-Marseille Université, CNRS, PIIM UMR 7345, 13397 Marseille, France*

An accurate predictive model for turbulent transport fluxes driven by microinstabilities is vital for the interpretation and optimization of present-day experiments, and extrapolation to and control of future machines. However, the computational cost of direct numerical simulation with massively parallel nonlinear gyrokinetic codes, $10^4 - 10^5$ CPUh for fluxes at a single radius, precludes their use for integrated tokamak transport simulations.

Increased tractability is gained by applying transport models based on the quasilinear approximation, largely valid in the core of tokamak plasmas [1, 2]. These are validated by comparison to nonlinear simulations, and have proven successful in reproducing experimental profiles in many cases. Compared to nonlinear simulations, a $\sim$6 orders of magnitude speedup is gained. However, their computational speed is still insufficient for applications such as convenient large-scale scenario development, trajectory optimization, and simulations for developing real-time controllers.

We suggest an approach to overcome these challenges. The central point is to emulate the original transport model with a neural network (NN) nonlinear regression of quasilinear fluxes previously compiled in a database. The neural network emulator is then orders of magnitude faster than original flux calculation, and can be used for real-time applications. Neural networks have been previously applied for regression of DIII-D heat fluxes from experimental power balance databases [3]. In this work, we suggest to apply the same methodology for code output.

The QuaLiKiz quasilinear gyrokinetic transport model [4, 5] was employed in this work. The computational time for the QuaLiKiz eigenvalue solver at a single wavenumber is $\sim$1 s.

A database of QuaLiKiz solutions was constructed, in the ion temperature gradient (ITG) instability regime. The code was run with adiabatic electrons for simplicity. The database covers

Table 1: Summary of input parameters for the QuaLiKiz adiabatic electron ITG database employed in this work

Parameter	Min value	Max value	No. of points
R/L_{Ti}	2	12	30
T_i/T_e	0.3	3	20
q	1	5	20
$\hat{s}$	0.1	3	20
$k_\theta \rho_s$	0.05	0.8	16
Total no. of points			3 840 000

five input parameters: the driving normalized logarithmic ion temperature gradient R/L_{Ti}, the ion to electron temperature ratio T_i/T_e, the safety-factor q, the magnetic shear $\hat{s} \equiv \frac{r}{q}\frac{dq}{dr}$, and the normalized wavenumber $k_\theta \rho_s$, where $\rho_s \equiv \sqrt{T_e m_i}/(Z_i q_e B)$. The database content is summarized in table 1. The training sets for the neural network were sifted from this database. Outputs include growth rates, frequencies, and ion heat flux. We concentrate on the analysis of the ion heat flux output, which involves a summation over the wavenumbers in the construction of the quasilinear saturation rule.

A multilayer perceptron neural network is used, which is a nonlinear function with tunable variables (weights and biases), with the property of universal approximation [6]. Linear combinations of the inputs and biases are propagated through a series of nonlinear transfer function vectors (named 'hidden layers'), until eventually linearly combined to an output layer. With two hidden layers and a single output value (as used in this work), this is represented as:

$$y = b_3 + \sum_i^N w_i^2 g\left(b_i^2 + \sum_j^M w_{ij}^1 g\left(b_j^1 + \sum_k^I w_{jk}^{in} x_k\right)\right) \tag{1}$$

Where y is the output 'neuron' containing the output value (e.g. ion heat flux), x_k the vector of input values, b^n the bias vectors, w^{in} the M×I weight matrix connecting the input vector to the 1st hidden layer, w^1 the N×M weight matrix connecting the two hidden layers, and w^2 the weight vector connecting the 2nd hidden layer to the output neuron. g is the nonlinear transfer function, defined as a sigmoid in this work:

$$g(x) = \frac{2}{1+e^{-2x}} - 1 \tag{2}$$

Following a series of optimization tests, two hidden layers, as shown in equation 1, were employed here. The hidden layer sizes M and N were set to 40. The input layer size, I, is 4 for ion heat fluxes, and 5 for growth rates and frequencies.

The key stage is the determination of the optimized values of the weights and biases through fitting a 'training set' of predetermined mappings, taken from a subset of 35000 QuaLiKiz database outputs. Following training, the network output then emulates the original model within the database input parameter envelope. This is validated by comparison to validation sets sifted from the database, chosen different from the training set. To avoid overfitting the data, regularization techniques were used in the optimization.

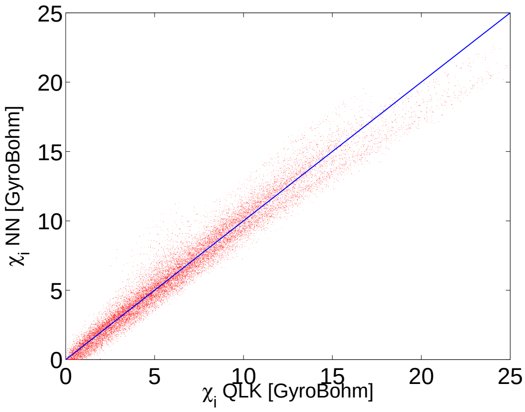

Figure 1: *Comparison between normalized ion heat fluxes obtained directly from QuaLiKiz (x-axis) and those from its NN regression (y-axis)*

A comparison between the regression NN and QuaLiKiz outputs for a validation set of 10000 unstable cases is shown in figure 1. The regression network has an RMS error of 0.77 in gyroBohm units ($\chi_{iGB} = \frac{T_i^{3/2} m_i^{1/2}}{(Z_i q_e B)^2 r}$) when compared to the validation set. This RMS error is primarily due to the regularization constraint. The impact of this error on the simulated profiles is minor, due to stiffness, and corresponds to a $\Delta(R/L_{Ti}) = 0.29$.

The typical quality of the fits can be seen in figure 2, displaying scans of the 4 separate input parameters while the others remained fixed. Each NN output is calculated on a sub 10 μs timescale in MATLAB on a Intel(R) Xeon(R) E5450 CPU @ 3.00GHz. This is a 5 order of magnitude speedup compared to the original QuaLiKiz calculations.

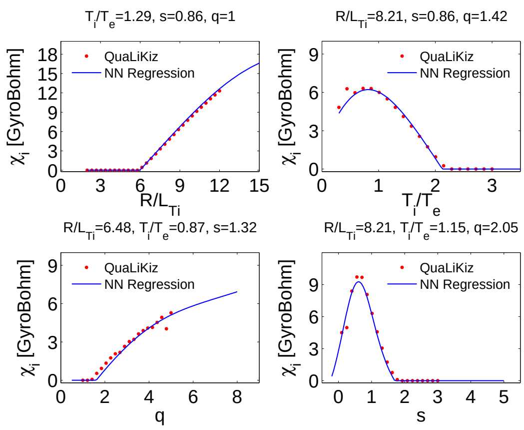

Figure 2: *Comparison of NN parameter scans (blue solid lines) vs the original QuaLiKiz ion heat flux calculations (red dots). The scans are in R/L_{Ti} (top left panel), T_i/T_e (top right panel), q (bottom left panel) and $\hat{s}$ (bottom right panel)*

A transport model based on the trained neural network was implemented both in the CRONOS [7] and RAPTOR [8] integrated modelling codes. We focus here on the real-time simulation capabilities offered in RAPTOR.

Presently, RAPTOR only models electron heat transport. The NN model output was thus modified to roughly approximate ITG regime electron heat transport by setting a constant $q_i/q_e = 3$. This is based on typical nonlinear and quasilinear observations in the ITG regime [9, 2].

In figure 3, we compare a RAPTOR simulation of an ITER hybrid scenario, using the QuaLiKiz NN model for electron heat transport, with a simulation of the same case originally carried out [10] using CRONOS and the GLF23 [11] transport model. A RAPTOR simulation of an entire 300 s ITER discharge took 10 s on a single CPU. This combination of simulation speed and first-principle modelling is unprecedented. With CRONOS/GLF23, the simulation took 24 hours.

This computational speed opens up many new possibilities for real-time controller design and validation, scenario preparation and optimization, and real-time discharge supervision. This initial QuaLiKiz neural network emulator is a proof-of-principle, and there remains much scope for expanding the number of input dimensions in the databases used for the fits, as well as employing slower yet more complete linear gyrokinetic codes for populating the database. This work is ongoing.

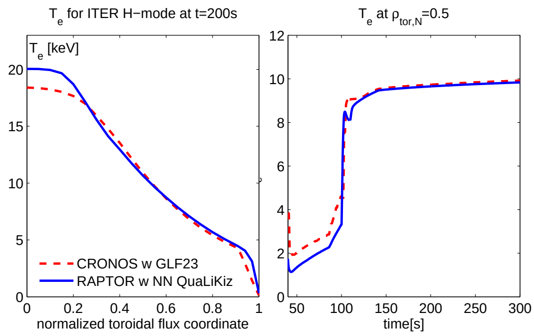

Figure 3: *Comparison between T_e predictions for an ITER hybrid discharge carried out with CRONOS/GLF23 [10] (red curve) and a RAPTOR simulation using the QuaLiKiz NN transport model (blue curve). A typical H-mode profile (left panel) and time dependence at mid-radius (right panel) are shown. The LH transition was set at 100 s.*

Acknowledgments.– This work is part of the research programme 'Fellowships for Young Energy Scientists' (YES!) of the Foundation for Fundamental Research on Matter (FOM), which is financially supported by the Netherlands Organisation for Scientific Research (NWO). The authors greatly thank O. Meneghini, S. Smith, U. von Toussaint and P. Xanthopolous for inspiring discussions.

References

[1] T. Dannert and Frank Jenko *Phys. Plasmas* **12** 072309 (2005)

[2] A. Casati *et al. Nucl. Fusion* **49** 085012 (2009)

[3] O. Meneghini, C. J. Luna, S. P. Smith, and L. L. Lao *Phys. Plasmas* **21** 060702 (2014)

[4] C. Bourdelle *et al. Phys. Plasmas* **14** 112501 (2007)

[5] C. Bourdelle. Turbulent Transport in Tokamak Plasmas: bridging theory and experiment. Physics. Aix Marseille Université, (2015). https://tel.archives-ouvertes.fr/tel-01113299

[6] Christopher M Bishop *Neural networks for pattern recognition.* Clarendon press Oxford (1995)

[7] J.F. Artaud *et al. Nucl. Fusion* **50** 043001 (2010)

[8] F. Felici and O. Sauter *Plasma Phys. Control. Fusion* **54** 025002 (2012)

[9] J.E Kinsey, R.E. Waltz and J. Candy *Phys. Plasmas* **12** 062302 (2005)

[10] J. Citrin *et al., Nucl. Fusion* **50** 115007 (2010)

[11] R.E. Waltz, G.M. Staebler, W. Dorland and G.W. Hammett *Phys. Plasmas* **4** 2482 (1997)