Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

RESEARCH ARTICLES

Multimodal convolutional neural networks for predicting evolution of gyrokinetic simulations

Mitsuru Honda*[1] | Emi Narita[2] | Shinya Maeyama[3] | Tomo-Hiko Watanabe[3]

[1]Graduate School of Engineering, Kyoto University, Kyoto, Japan
[2]Naka Fusion Institute, National Institutes for Quantum Science and Technology, Ibaraki, Japan
[3]Department of Physics, Nagoya University, Aichi, Japan

Correspondence
*Mitsuru Honda, Graduate School of Engineering, Kyoto University, Kyotodaigaku-katsura, Nishikyo, Kyoto 615-8530, Japan. Email: honda.mitsuru.5c@kyoto-u.ac.jp

Funding Information
This research was supported by the JSPS, KAKENHI Grant Number 22K03574 and 20K14450, and MEXT as "Program for Promoting Researches on the Supercomputer Fugaku" (Exploration of burning plasma confinement physics: Project IDs: JPMXP1020200103, hp200127, hp210178 and hp220165).

Abstract

Gyrokinetic simulations are required for the quantitative calculation of fluxes by turbulence, which dominates over other transport mechanisms in tokamaks. However, nonlinear gyrokinetic simulations are computationally expensive. A multimodal convolutional neural network model that reads images and values generated by nonlinear gyrokinetic simulations and predicts electrostatic turbulent heat fluxes was developed to support efficient runs. The model was extended to account for squared electrostatic potential fluctuations, which are proportional to the fluxes in the quasilinear model, as well as images containing fluctuating electron and ion distribution functions and fluctuating electrostatic potentials in wavenumber space. This multimodal model can predict the time and electron and ion turbulent heat fluxes corresponding to the input data. The model trained on the Cyclone base case data successfully predicted time and fluxes not only for its test data, but also for the completely different and unknown JT-60U case, with high accuracy. The predictive performance of the model depended on the similarity of the linear stability of the case used to train the model to the case being predicted.

KEYWORDS:
GKV gyrokinetic simulation, turbulent heat flux, deep learning, convolutional neural network, multimodal model

1 | INTRODUCTION

In magnetically confined plasma devices, the transport of particles, heat, and momentum should be reduced to achieve good plasma confinement. Experiments and simulations showed that in tokamaks, turbulent transport, which is driven by turbulence, predominates among transport mechanisms in most cases in tokamaks. Over the years, research has been conducted to elucidate the detailed characteristics of turbulent transport and its reduction mechanisms. One of the powerful tools for tackling this problem is a local flux-tube gyrokinetic simulation code, which solves Vlasov-Maxwell equations with respect to the perturbed distribution function $\tilde{f}$ in the phase space with the assumption that the equilibrium distribution function f_0 is fixed. Numerous local flux-tube gyrokinetic codes have been developed, including GS2[1], GYRO[2], CGYRO[3], GENE[4], GKW[5], and GKV[6]. In this study, we used the GyroKinetic Vlasov simulation code (GKV). GKV has been extensively employed to investigate turbulent transport physics and quantify turbulent fluxes: finite-β dependence of electromagnetic turbulent transport[7], the ion-temperature-gradient (ITG) driven turbulent tungsten transport[8], and cross-scale interaction between small- to large-scale turbulence[9–11].

Similar to other codes, GKV tracks the nonlinear evolution of a perturbed distribution function $\tilde{f}$ at a single spatial location as an initial value problem by solving[6]

$$\frac{\partial \tilde{f}}{\partial t} + v_{\parallel}\boldsymbol{b}\cdot\nabla\tilde{f} + \frac{c}{B_0}\{\Phi,\tilde{f}\} + \boldsymbol{v}_d\cdot\nabla\tilde{f} - \mu(\boldsymbol{b}\cdot\nabla\Omega_i)\frac{\partial\tilde{f}}{\partial v_{\parallel}} = (\boldsymbol{v}_* - \boldsymbol{v}_d - v_{\parallel}\boldsymbol{b})\cdot\frac{e\nabla\Phi}{T_i}f_0 + C(\tilde{f}), \tag{1}$$

where $\boldsymbol{b}$, B_0, c, Φ, e, T_i, and Ω_i are the unit vector parallel to the magnetic field, the magnetic field strength on the magnetic axis, the speed of light, the electrostatic potential averaged over the gyromotion, the elementary charge, the ion temperature and the ion cyclotron frequency, respectively. Here $v_{\parallel}$, $\boldsymbol{v}_d$, $\boldsymbol{v}_*$, and μ are the parallel speed, the magnetic drift velocity, the diamagnetic drift velocity, and the magnetic moment, respectively. Furthermore, $\{\dots\}$ is the Poisson brackets, C is the linearized collision term, and f_0 is the local Maxwellian. Note that the subscripts s denoting the particle species are dropped for the sake of simplicity. Even though the numerical cost of a local flux-tube gyrokinetic simulation is less than that of a full-f global gyrokinetic simulation dealing with the evolution of $f(= f_0 + \tilde{f})$ in an entire spatial domain, it generally takes a few to several dozens of hours to complete a nonlinear calculation using thousands of central processing units (CPUs). Furthermore, in the studies described above, a multiscale gyrokinetic simulation simultaneously covering turbulence from low to high wavenumber spectra takes from tens of hours to weeks to complete even when using tens of thousands of CPUs in a modern powerful large-scale supercomputer like Fugaku. Specifically, a GKV multiscale simulation shown in[11] used 2,560 nodes in Fugaku and took 250 h, i.e., 640 thousand nodehours, wherein each node in Fugaku had 48 computation cores. The vast amount of required computational resources hinders the systematic surveying of input parameter sets and from performing multiple cases for model validation activities[12]. The rapid prediction of turbulent fluxes is also required from the aspect of predicting plasma profiles with a transport code. Even though frameworks, such as TGYRO[13] and TRINITY[14], have been proposed, the prediction of turbulent fluxes at all radii by nonlinear gyrokinetic simulations is too costly. Rapid computation will enable the direct use of nonlinear simulation results on a transport code and improve the reliability of profile predictions. Ways to accelerate computation to overcome the time-consuming calculation problem have been explored. Relying on rapid improvements in computer performance alone for further reductions in computation time is unrealistic. One approach is to develop a faster code. A GPU-native local flux-tube gyrokinetic code that uses pseudo-spectral methods in configuration and velocity space has been recently proposed[15]. The code rapidly predicts fluxes in the Cyclone base case (CBC)[15]. Our alternative approach is to utilize deep learning techniques with gyrokinetic simulation data.

In a typical nonlinear gyrokinetic simulation, the nonlinearly growing phase of fluctuations appears after the linearly growing phase and lasts until the fluctuations at all wavenumbers are saturated. The linearly growing phase is defined as a state in which fluctuations are exponentially growing. In other words, the nonlinear terms in the gyrokinetic equation have not yet taken significant effect. Fluctuation evolution begins diverging from exponential growth when the simulation enters the nonlinearly growing phase because nonlinear terms are effective. The fluctuations stop growing continuously at some point and level off on average afterward even though they continue fluctuating to some degree. This phase is called the saturation phase. Turbulent fluxes are estimated by averaging them over a certain window of time in the saturation phase. Thus, the simulation must be continued for some time after the saturation phase appears. This requirement accounts for the considerable amount of time needed to estimate the turbulent fluxes by gyrokinetic simulations. Given that the saturation phase is the phase involved in flux calculation, the computation leading up to it is a period of endurance, and thus far, the vast amount of data generated prior to the saturation phase has been negligibly utilized. Ultimately, computational costs would be significantly reduced if a model that properly predicts the final saturation level of fluxes or the path to saturation at an early state of the simulation could be developed.

Deep learning techniques have the potential to build such a model. However, if such techniques were to be used, reducing the large amount of data generated by gyrokinetic simulations to a manageable volume remains necessary. A gyrokinetic simulation yields the time-series perturbed distribution function for each species in the five-dimensional space, $\tilde{f}(k_x, k_y, z, v_{\parallel}, \mu)$, where k_x and k_y denote radial and poloidal wavenumbers, respectively, z is the poloidal angle, $v_{\parallel}$ is the parallel velocity and μ is the magnetic moment representing the perpendicular velocity $v_{\perp}$. The raw output data are the table of numerical values and are too massive to handle. Once the numerical data are converted into images, the data size can be reduced by any excellent image compression algorithm then easily handled by deep learning techniques, such as convolutional neural networks (CNNs). However, if the five-dimensional data are to be imaged, 10 different combinations of images must be produced. Given that handling all 10 different images at one time is difficult from the standpoint of numerical costs, selecting one type of images is favorable that best describes the time evolution of fluctuations as a representative is favorable. By scrutinizing the pattern of fluctuation evolution in gyrokinetic simulations from various aspects, a type of the image depicting $|\tilde{f}|^2(k_x, k_y)$, i.e., the intensity of $\tilde{f}$ in the wavenumber space, was selected as the best type of image. As will be discussed again in Section 2.1, $|\tilde{f}|^2(k_x, k_y)$ images are created for multiple $v_{\perp}$ at $z = 0$ and $v_{\parallel} = 0$. In our previous work[16], we developed a model based

on a state-of-the-art CNN model, EfficientNet[17], which can efficiently read images and extract features. EfficientNet has been optimized for computer vision tasks, such as image classification and object detection. It achieves a balance between accuracy and computational efficiency through its design, which employs both model scaling and network architecture search. EfficientNet is a generic name, and EfficientNet-B0 to B7, depending on the complexity of the network, exist. Among these models, we used EfficientNet-B4 as a basis of our model. Transfer learning and fine-tuning techniques were employed to fit the pretrained EfficientNet to our input data set, i.e., images of $|\tilde{f}|^2(k_x, k_y)$. Transfer learning is a technique wherein a pretrained model, that is, EfficientNet-B4 in this context, is repurposed for a different but similar task. By diverting some knowledge learned in the previous task, the model can be trained faster and acquire better accuracy for a new task. Fine-tuning refers to the process of partly readjusting the hyperparameters of some top layers or updating the model architecture of a pretrained model by adding new layers for a new task. Our model can accurately classify the phase and predict the simulation time corresponding to the input data[16]. It is a new method that could estimate the simulation time from images in the wavenumber space. The model is a time predictor and outputs the time t corresponding to the input data. The actual output is scaled to the [0, 1] range, $\bar{t}$, where $\bar{t} = 0$ corresponds to the initial time of a simulation and $\bar{t} = 1$ is the onset time of turbulent flux saturation. Here, t is in R/v_{tp} units, where R is the major radius at the axis and v_{tp} is the proton thermal speed.

Despite its ability to predict time with high accuracy, predicting turbulent heat fluxes with the time predictor at that time is virtually impossible. In actuality, images of $|\tilde{f}|^2(k_x, k_y)$ used as the input of the time predictor are normalized with the maximum value of $|\tilde{f}|^2$ at each time. Each input image is therefore only a relative intensity distribution in the (k_x, k_y) space and does not contain information associated with any kind of absolute value. Two main reasons for normalizing images exist. The first is to recognize a change in the fluctuation intensity map over time in the wavenumber space. In gyrokinetic simulations, fluctuation intensity varies by several orders of magnitude from the beginning of the simulation to just before saturation. If the image is generated on a scale that can represent the maximum value throughout the simulation, the pattern would be completely unreadable from an image with lower intensity. The images should be normalized to extract meaningful information, even from images in the early stage. The second reason is to apply the trained model to a case that is different from the data on which it was trained. A maximum value is different in each case. Without normalization, the same color would correspond to different intensities in each case because gradation varies within the range of values contained in the data. Hence, having the images include the magnitude of fluctuations to predict fluxes is inadequate.

In the prediction of turbulent heat fluxes, the model input should include information on the absolute value of some quantities, such as the intensity of potential fluctuations $\tilde{\phi}$ or $\tilde{f}$. Here, $\tilde{\phi}$ can be determined by the quasi-neutrality condition with $\tilde{f}$[6], and the definition of the turbulent heat fluxes in GKV is consulted in[6]. We therefore develop a multimodal model. A multimodal model generally learns representations from different types of modalities, such as images, texts, sounds, and numerics, within the same model. The model being developed should be able to read and digest two types of modalities: an image and a numerical value. Our multimodal model is thus anticipated to gain the ability to predict fluxes even for an unknown case. That is, the model can predict fluxes by acquiring generalization to flux predictions to previously unseen data sets, where they have similar enough structure in the distribution functions to the training data sets. This paper describes the development of a multimodal model that can properly predict turbulent heat fluxes in growing phases. This model is a milestonet toward the ultimate goal of predicting the final saturated fluxes from data in an early phase of a simulation even for an unseen case.

The rest of this paper is organized as follows. Section 2 details the architecture of a multimodal model. In Section 3, the model trained on the GKV simulation data for the CBC[18], hereafter dubbed CBC data, was examined to assess the performance of the model in flux prediction. Section 4 describes the application of the CBC-trained model will be applied to unknown cases, that is, GKV simulation data for JT-60U plasma. The differences between the cases in which it had good or poor prediction performance were also investigated. Finally, the conclusions and discussion are presented in Section 5.

2 | MULTIMODAL MODEL

The model was implemented in Python by using TensorFlow 2.8.0. The multimodal model was developed by extending the input of the model introduced in a previous paper[16] This model uses images as input to infer simulation time, as explained in Section 1. Given that in principle, the CNN part of the multimodal model is similar to the previous one, one may consult the basis of the CNN in Section 3 of[16]. Here, we will explain the parts of the model that have been extended to handle multimodality.

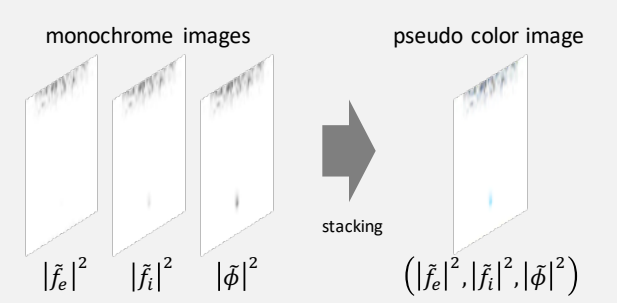

Figure 1 Scheme for processing a pseudo color image stacking the monochrome images of $|\tilde{f}_\mathrm{e}|^2$, $|\tilde{f}_\mathrm{i}|^2$ and $|\tilde{\phi}|^2$ in the (k_x, k_y) space.

2.1 | Input and label data

As in[16], the data in the linearly and nonlinearly growing phases were used in this study and those in the saturation phase were not used. The output of the multimodal model was the simulation time t and the electron and ion turbulent heat fluxes Q_e and Q_i, respectively. Note that the heat flux Q_s for species s used throughout this work is the one in $n_\mathrm{e}T_\mathrm{i}v_\mathrm{tp}(\rho_\mathrm{tp}/R)^2$ unit and $\tilde{\phi}$ in $\rho_\mathrm{tp}T_\mathrm{i}/(eR)$ unit. Here, n_e is the electron density, T_i is the ion temperature and ρ_tp is the proton Larmor radius. Adequately choosing which sort of images and which numerical data will be used as the input of the multimodal model is necessary. The quasilinear formulation in principle leads to the expression of turbulent fluxes in proportion to the square of the fluctuating electrostatic potential (see e.g.,[19]): $Q_\mathrm{Es} \sim |\tilde{\phi}|^2$, where Q_Es is the turbulent heat flux driven by the electrostatic fluctuations. Hereafter, the electrostatic part of the fluxes is the sole focus because it usually predominates over the electromagnetic part. $|\tilde{\phi}|^2$ is naturally chosen as the numerical data input. Given that $|\tilde{\phi}(k_x, k_y)|^2$ is apparently a two-dimensional quantity in the wavenumber space, numerous ways to reduce the dimension of $|\tilde{\phi}|^2$ to a zero-dimension data are available. Here, we define $|\tilde{\phi}|^2$ as $\sum_{k_x,k_y}\langle|\tilde{\phi}(k_x, k_y)|^2\rangle$, where $\langle\cdot\rangle$ is the flux-surface average.

Although the amplitude of the heat fluxes can now be roughly inferred from $|\tilde{\phi}|^2$, predicting the heat flux for each species without information that helps distinguish between electron and ion heat fluxes remains difficult. In a previous work, images of ion $|\tilde{f}|^2$, i.e., $|\tilde{f}_\mathrm{i}|^2$, were solely fed to the model. However, they were apparently insufficient for the individual prediction of the electron and ion turbulent heat fluxes by the model because they did not include information on electrons. Not only images of $|\tilde{f}_\mathrm{i}|^2$, as was done in the previous work but also those of $|\tilde{f}_\mathrm{e}|^2$ and $|\tilde{\phi}|^2$, should be used as the input of the multimodal model. The images of $|\tilde{\phi}|^2$ and $|\tilde{f}_\mathrm{e}|^2$ are also normalized in the same manner as those of $|\tilde{f}_\mathrm{i}|^2$. However, this situation does not directly mean that we can exploit three CNNs for three kinds of images. Even the relatively compact EfficientNet-B4 is still a large model with tens of millions trainable parameters[17], and a single model equipped with three CNNs would be too large and complex to handle. The red-toned image used for the previous model contained only information on the normalized $|\tilde{f}_\mathrm{i}|^2$ distribution[16]. We see that the image needs not have red-based tones; only black and white shade information is sufficient. A red, green, and blue image contains three color channels, and each color channel is stacked to create a natural-colored image. In other words, an image is a three-dimensional (3D) tensor, and each color channel of the image is a two-dimensional (2D) tensor with information on intensity. Therefore, the 2D tensors of $|\tilde{f}_\mathrm{e}|^2$, $|\tilde{f}_\mathrm{i}|^2$, and $|\tilde{\phi}|^2$ can be stacked to form a 3D tensor as a three-color channel image, as shown in Figure 1. This 3D tensor is not literally an image, but given that it has the same format as an image in the program, we call it a "pseudo" color image and use it as input for the multimodal model. It must have information on $|\tilde{f}_\mathrm{e}|^2$, $|\tilde{f}_\mathrm{i}|^2$, and $|\tilde{\phi}|^2$. Thus, feeding a single pseudo color image to a CNN is enough to take in three kinds of information.

Finally, as explained in Section 3 of[16] in detail, at each time the multiple images corresponding to the different perpendicular velocities $v_\perp$ are generated with the parallel velocity $v_\parallel = 0$ and the poloidal angle $z = 0$ fixed. $v_\parallel = 0$ and $z = 0$ were selected because fluctuation amplitudes are usually large there regardless of time. Multiple values are allowed for $v_\perp$ because increasing the number of images used for training is advantageous, and a representative value of $v_\perp$ does not seem to exist. This fact should be kept in mind when examining Figure 4, 6 and 7.

The label data, called the output data in inference mode, are the normalized time $\bar{t}$, as explained in Section 1, and the electron and ion turbulent heat fluxes Q_e and Q_i. Q_e and Q_i are individually standardized by rescaling their distributions such that their means are zero and standard deviations are unity, respectively, prior to feeding them to the model.

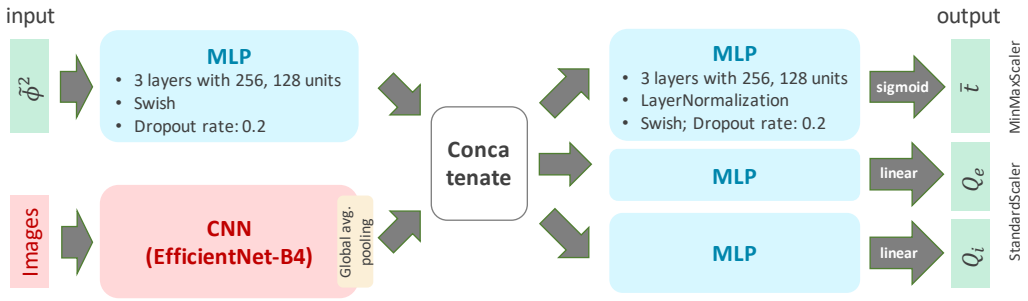

Figure 2 Schematic explanation of the model architecture.

2.2 | Model architecture

The model reads two kinds of inputs, viz., numerical values of $|\tilde{\phi}|^2$ and images, and should therefore be equipped with two sets of layers corresponding to these inputs. The array of numerical values can be appropriately processed by a fully connected feedforward neural network model, namely, a multilayer perceptron (MLP). The input layer for the $|\tilde{\phi}|^2$ data is the dense layer with a sole unit, followed by three hidden layers with 256, 256 and 128 units. The layers of the Swish activation function are sandwiched between each hidden layer and the dropout layer with a rate of 0.2 is added after the final hidden layer. The images are fed into the input layer of EfficientNet-B4 pretrained on ImageNet, and the processed feature vector is output through the global average pooling layer. Here, ImageNet is a large image database with adequate labels and includes approximately a million images. Then, the feature vectors generated by the MLP and CNN are concatenated and subsequently fed to three MLPs, each of which predicts a different quantity, i.e., $\bar{t}$, Q_e and Q_i. The architecture of the MLPs for $\bar{t}$, Q_e and Q_i is almost the same as that for the $|\tilde{\phi}|^2$ input except that the additional layers normalizing the activations of the previous layer, i.e., the LayerNormalization layers. The activation layers settled right before the output layers exploit the sigmoid and linear activation functions for time and fluxes, respectively. The selection of these activation functions depends on the normalization method of label data, as described in Section 2.1. Figure 2 provides a schematic explanation of the architecture of the whole multimodal model.

The image data set is augmented on the fly in a similar manner that is explained in Section 3 of[16], except that the empty areas that emerge when the image is rotated or shifted are filled in with zeros other than with the nearest pixel values. This choice was made through trial and error, which revealed that prediction performance is clearly improved when filling with the constant values, say, zeros, than with the neighboring values.

EfficientNet-B4 consists of seven major blocks[17]. In contrast to those in a previous study[16], the weights and biases pretrained on ImageNet are frozen up to the first *five* blocks and those on the remaining blocks are unfrozen such that they can be updated by training in this work. The number of frozen blocks is determined by parameter survey. The RMSprop optimizer with a learning rate of 0.0001 and a momentum of 0.9 is used. The learning rate should generally be kept low for transfer learning. The LogCosh function is adopted as a loss function for all model outputs. It behaves like the mean squared error function but is less affected by an occasional largely incorrect prediction. We confirmed that using the root mean squared logarithmic error function instead also works well in our model. Note that the loss contributions of the flux outputs are weighted 10-fold against that of the time output to attain better flux prediction performance.

3 | TRAINING USING CBC DATA AND PREDICTION FOR ITS TEST DATA SUBSET

The model developed is trained on CBC data to examine prediction performance. The typical plasma parameters used to generate this data and those used in Section 4 are the same as parameters in[16]. These are also summarized in Table 1. For obtaining accurate predictions with relatively less training data, transfer learning and fine-tuning techniques are employed when training the EfficientNet-B4 part of the model. The data are randomly split into training, validation, and test data sets using the train_test_split function twice with a given random seed. The numbers of training, validation, and test data are 2,635, 765, and 391, respectively, and the batch size is accordingly set to 64. As a result of efficient input pipelining with TensorFlow dataset and autotune APIs, the training itself ended in less than 15 min on GeForce RTX 3090 depending on the number of epochs to be finished by early stopping. Figure 3 shows the smooth reduction in the training and validation losses as the number of epochs increases. The

Table 1 Plasma parameters used for gyrokinetic simulations. The JT-60U plasma parameters are based on the plasma of discharge number #45072.

	case	R/L_{T_e}	R/L_{T_i}	R/L_{n_e}	T_e/T_i	q	s
CBC		6.92	6.92	2.22	1.00	1.40	0.780
JT-60U	(A) $\rho = 0.50$	7.67	4.70	3.92	1.33	1.52	0.813
JT-60U	(B) $\rho = 0.26$	4.25	4.64	2.03	1.42	1.09	0.280

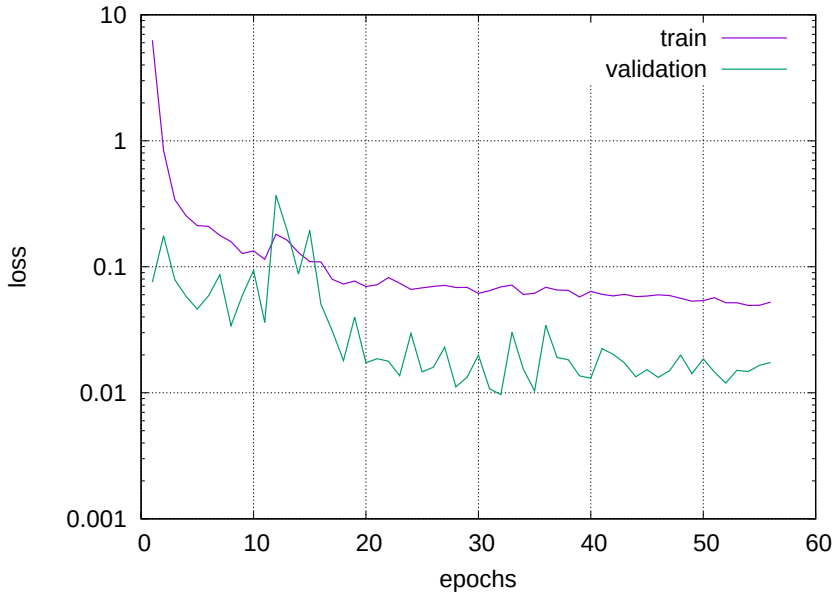

Figure 3 Logarithmic plot of the training and validation losses as a function of epochs.

training is stopped at 56 epochs due to early stopping, which terminates the training when the validation loss does not decrease for 24 consecutive times. As seen from Figure 3, the minimum validation loss is achieved at 32 epochs. The learning rate is reduced by $\sqrt{0.1}$ from 10^{-4} to 10^{-6} when the validation loss does not decrease for seven consecutive times.

The coefficient of determination is defined as $R^2 = 1 - \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 / \sum_{i=1}^{n}(y_i - \bar{y}_i)$ with the number of data points n, the true value y_i, the predicted $\hat{y}_i$, and the mean $\bar{y}_i$ to measure model prediction performance. The R^2 values of $\bar{t}$, Q_e and Q_i predictions for the test data are 0.972, 0.9995, and 0.9995, respectively. The regression plots shown in Figure 4 clearly show that the predicted values, which are presented as red dots in the figure, are almost aligned with the line of slope unity. In Figure 4 (b) and (c), the number of dots in the regression plots of the fluxes appears small relative to the amount of test data because the fluxes are very small in the linear phase and remain small in most of the nonlinear growing phase, as seen in their logarithmic graphs in Figure 4 (d) and (e). The model does not predict the true value well in the range of the true flux values less than about unity. We do not emphasize accurately predicting excessively small fluxes because our aim is to predict fluxes near the saturation phase as accurately as possible.

This result reveals that the multimodal model can attain the ability to reproduce sufficiently the fluxes and time for the test data. However, our preliminary work has ensured that the previous model using only images as input could reproduce fluxes to a certain degree if it is applied to the test data subset. The true value of the multimodal model can be tested only when it is

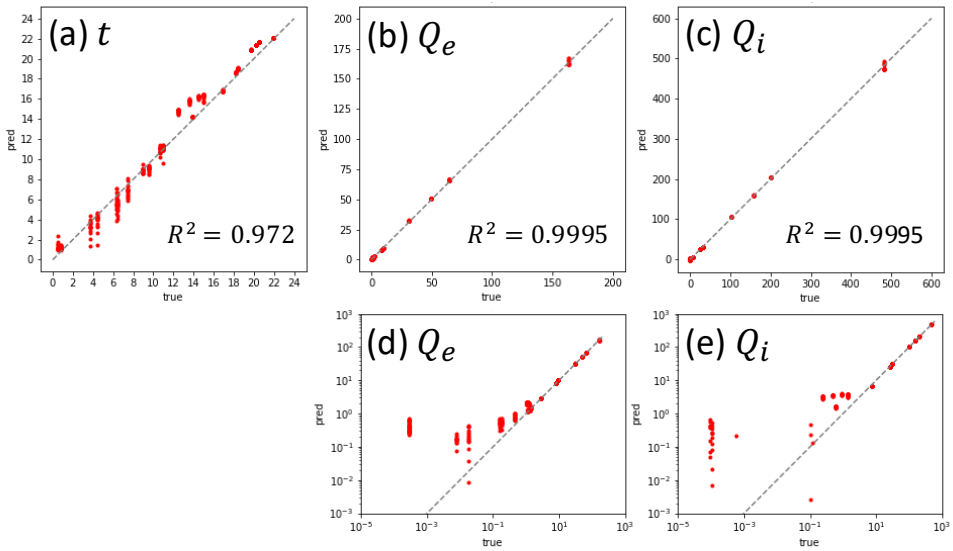

Figure 4 Regression plots of the predicted (a) t, (b) Q_{e}, and (c) Q_{i} versus the true ones for the test data with the corresponding R^2. The actual model output $\bar{t}$ is rescaled to t for visibility. The plots in (d) and (e) are the logarithmic graphs of (b) and (c), respectively. The number of data points is 391.

applied to data that are not associated with the subset of the data used for training. Hereafter, the multimodal model trained in this section will be used as is.

4 | PREDICTION PERFORMANCE FOR UNKNOWN DATA

As reported in this section, the multimodal model trained on CBC data is applied to the two data sets of JT-60U #45072 at different radial positions. One data set is the plasma parameter set at $\rho = 0.50$, namely, JT-60U (A), and the other data set is the set at $\rho = 0.26$, namely, JT-60U (B). Their parameters are summarized in Table 1. Here, ρ is the normalized radial coordinate. For all cases, the evolution of heat fluxes and the linear stability diagram showing the normalized linear growth rate γ and the normalized real frequency ω as a function of the normalized k_y are shown in Figure 5 (a) and (b), respectively. Knowing the appearance of the distributions of fluxes in the CBC data set used to train the model before applying it to other data sets is meaningful. The histogram of Q_{e} and Q_{i} is shown on the right panel in Figure 5 (a3). The horizontal axis represents flux values and the vertical axis represents the number of data included in each bin. As expected, the distributions of the data do not look like normal distributions. For electrons and ions, the skewness and kurtosis, which are measures of the deviation from the normal distribution, are $[3.20, 3.14]$ and $[9.57, 9.22]$, respectively. These values clearly indicate non-normal distributions. The histograms are almost flat over the entire range and can be called log-uniform distributions, except for the peak observed only for Q_{e}, which corresponds to the plateau of Q_{e} at approximately $t = 10$ shown in Figure 5 (a1). The mean and median of the data set are $\mu = [13.5, 39.7]$ and $\mathrm{Mdn} = [1.17, 0.131]$ for electrons and ions, respectively, but in this kind of distributions wherein values vary over many orders of magnitude, the mean does not make sense.

Prior to application to unknown cases, we first discuss the results of the model predictions for the CBC data in comparison with Figure 5 (a3). In gyrokinetic simulations, fluxes take time to rise to near-saturation levels. Therefore, most of the flux values contained in the data set are inevitably quite small, as shown in Figure 5 (a3), and such small flux values are not predicted very well by the model, as seen in Figure 4 (b)-(e). On the basis of this prediction result, input data with small flux values may not have contributed to the prediction of the flux values; However, they may act as a sign of an early phase of a simulation. Thus, the data with small flux values have helped predict time accurately.

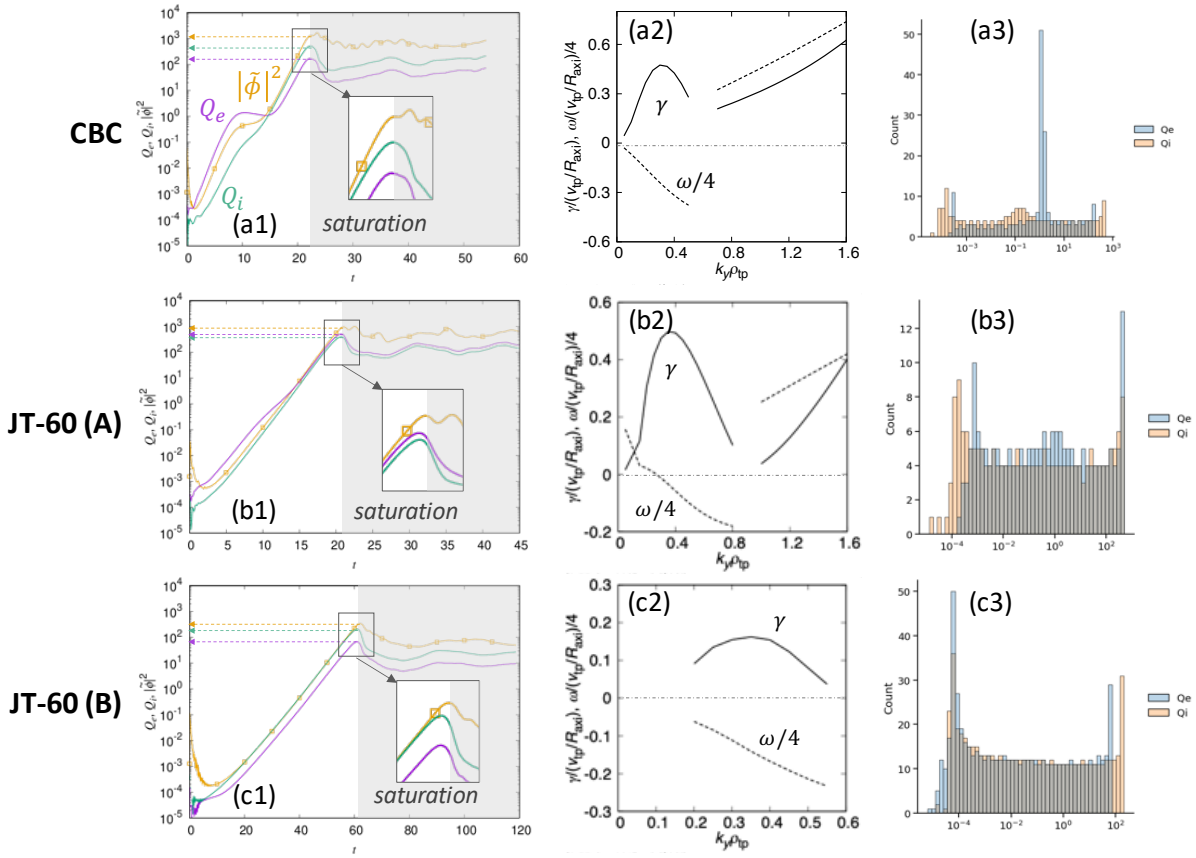

Figure 5 Plots in the left panel show the evolution of $|\tilde{\phi}|^2$ (yellow with symbols), Q_e (purple), and Q_i (green) as a function of time; those in the middle show the linear stability diagrams; and those in the right show the histograms of Q_e and Q_i for CBC and the JT-60U (A) and (B) cases. The subfigures embedded in the left panels are magnified plots near saturation, and the dashed arrows are drawn to facilitate finding peak value magnitudes.

4.1 | JT-60U (A) case

Prior to checking prediction performance, the difference between the characteristics of CBC and the JT-60U (A) case shown in Figure 5 must be examined. From the statistics point of view, $\mu = [50.9, 39.5]$ and $\mathrm{Mdn} = [0.405, 0.117]$ with the skewness of $[2.72, 2.74]$ and kurtosis of $[6.40, 6.50]$ for electrons and ions, respectively. In Figure 5 (b1), Q_e is larger than Q_i in the JT-60U (A) case and the similar is true for CBC until the middle of the nonlinearly growing phase. This implies that the pattern of flux evolution in the JT-60U (A) case is somewhat different from that in CBC. Immediately before the saturation, all flux peak values, except for Q_e in CBC, which is approximately 170, are close to 500. This finding indicates that the model trained on CBC data has not experienced $Q_e \gtrsim 170$ but will have to predict such a situation when it is applied to JT-60U (A) data. Therefore, building a multimodal model has improved the prediction performance for unknown data if the model can predict Q_e well even when $Q_e \gtrsim 170$. Both the linear stability diagrams in general show typical ITG/TEM behavior, wherein TEM denotes a trapped electron mode, despite some difference in ω behavior in the low-k_y region and in the γ magnitude.

The predictive performance of the CBC-trained model in the JT-60U (A) case is scrutinized on the basis of differences and similarities. Regression plots are shown in Figure 6. Despite some variance in t predictions in the early and middle phases, all three model outputs agree very well with the true values with high R^2 values of 0.915, 0.987, and 0.989. It should be noted that the prediction accuracy decreases for small flux values as the model predictions seem to have a cutoff around 1 to 10. Even in the region where Q_e exceeds approximately 170, Q_e can be successfully predicted. Interestingly, the multimodal model trained on CBC data, wherein Q_i is approximately three times larger than Q_e at their maximum, successfully reproduced the fluxes in the JT-60U (A) case, wherein Q_i and Q_e are comparable. Among the inputs for the model, $|\tilde{\phi}|^2$ is the only information that can be used to infer flux amplitude, which is of course independent of species. Nevertheless, the successful prediction of the flux of

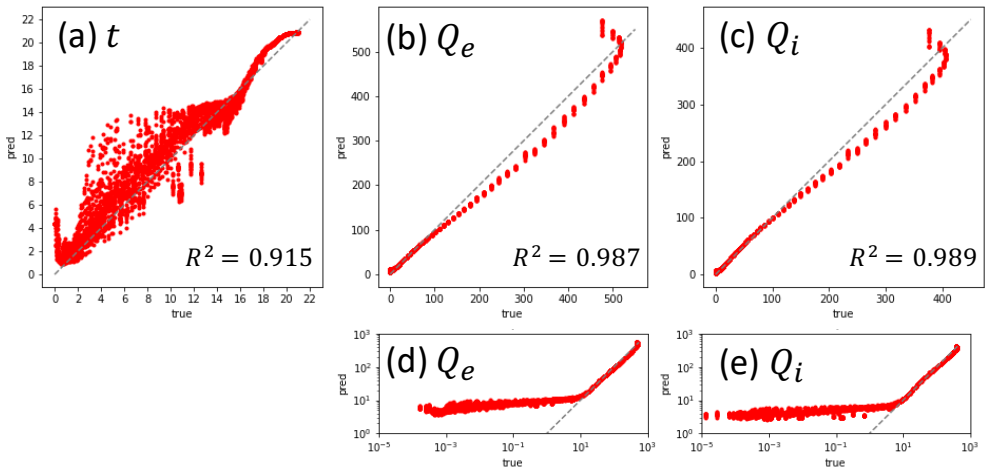

Figure 6 Regression plots of (a) t, (b) Q_e, and (c) Q_i predicted by the CBC-trained model versus the true ones for JT-60U (A) data with the corresponding R^2. The plots in (d) and (e) are the logarithmic graphs of (b) and (c), respectively. The number of data points is 3,587.

each species means that the model can read the difference between Q_e and Q_i from the image input. In other words, the method using a pseudo image overlaying $|\tilde{f}_e|^2$, $|\tilde{f}_i|^2$, and $|\tilde{\phi}|^2$ images is working effectively. The multimodal model trained on CBC data can predict t, Q_e, and Q_i with high accuracy for the unknown JT-60U data when provided with only a set of images and $|\tilde{\phi}|^2$ values.

As shown in Figure 5 (a3) and (b3), the CBC data contain Q_e peaks at approximately unity, and the model trained on that data should naturally incorporate information on that peak, whereas the JT-60U (A) data do not contain such peaks there. Figure 6 (b)-(e) show that when applying the CBC-trained model to the JT-60U (A) case, R^2 and the predictive tendency between Q_e and Q_i are almost the same. This finding indicates that the Q_e peaks in CBC data may not affect the predictive performance of the model on JT-60U (A) data, which do not contain such peaks.

4.2 | JT-60U (B) case

We now proceed to the next case, JT-60U (B), where $\mu = [5.87, 17.7]$ and $\mathrm{Mdn} = [0.0116, 0.0322]$ with the skewness of $[2.94, 2.98]$ and kurtosis of $[7.73, 7.98]$ for electrons and ions, respectively. Figure 5 (c1) shows that Q_i always hovers above Q_e during the entire period before saturation, which is different from that in the JT-60U (A) case and partly different from that in CBC. In general, the amplitudes of Q_e and Q_i in the JT-60U (B) case are less than half those in CBC, indicating that the range of the fluxes in the JT-60U (B) case is well within the range in CBC. Thus, the CBC-trained model has already known more or less the range in the JT-60U (B) case. The linear stability diagram in Figure 5 (c2) clearly shows that the plasma is stable in the $k_y \lesssim 0.2$ and $k_y \gtrsim 0.55$ regions and elsewhere the ITG mode solely resides. The relatively smaller growth rate may be associated with the relatively smaller fluxes in the nonlinear simulation.

Regression plots measuring the prediction performance of the model are shown in Figure 7. These plots indicate poor performance. The R^2 values are 0.643, 0.518, and 0.555. In time prediction, the model predicts time farther into the future than it actually is in the early phase but has better time prediction in the later phase. A similar pattern of predictions is seen in Figure 12 (c) of[16]. For fluxes, as seen in Figure 5 (c1), the amplitudes of the fluxes are at most less than or equal to 10 until $t = 50$. This result indicates that most of the flux prediction points visible in Figure 7 (b) and (c) correspond to $t > 50$. They tend to be underestimated by a factor of three. Figure 5 (c1) reveals that the fluxes begin to decrease before entering the saturation phase, whereas $|\tilde{\phi}|^2$ continues to increase. As a result, the series of predictions appear if they are folding back in Figure 7 (b)-(e), and the predicted values immediately before saturation agree well with the true values. This finding shows that the values close to their maximum are well reproduced, but those en route to saturation are poorly reproduced. Figure 8 also clearly depicts the time series of the true and predicted heat fluxes for electrons and ions. The heat fluxes are not successfully reproduced in the early and middle growing phases wherein the fluxes are approximately less than 10. Where $Q_e, Q_i \gtrsim 10$, they can be predicted to some extent, but are less than the true values. Just before saturation, both heat fluxes are close to the true values.

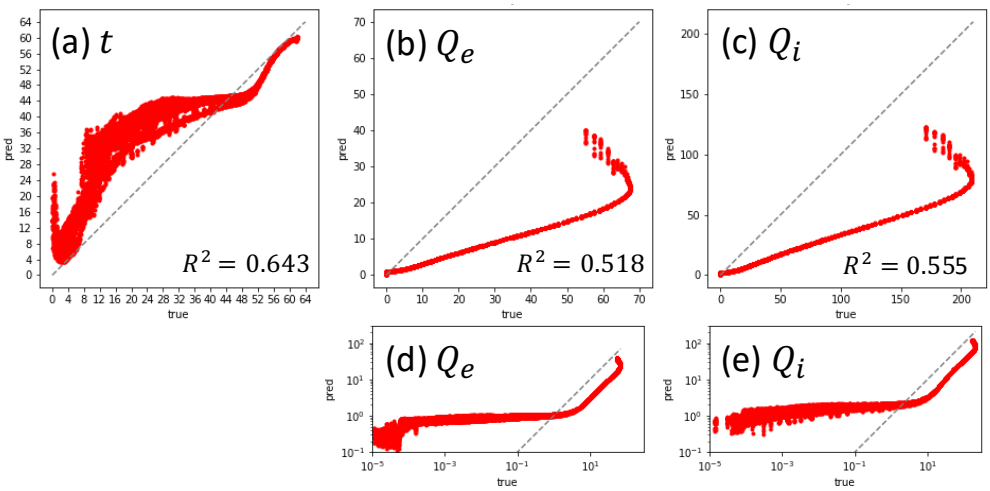

Figure 7 Regression plots of (a) t, (b) Q_e, and (c) Q_i predicted by the CBC-trained model versus the true ones for JT-60U (B) data with the corresponding R^2. The plots i (d) and (e) are the logarithmic graphs of (b) and (c), respectively. The number of data points is 10,557.

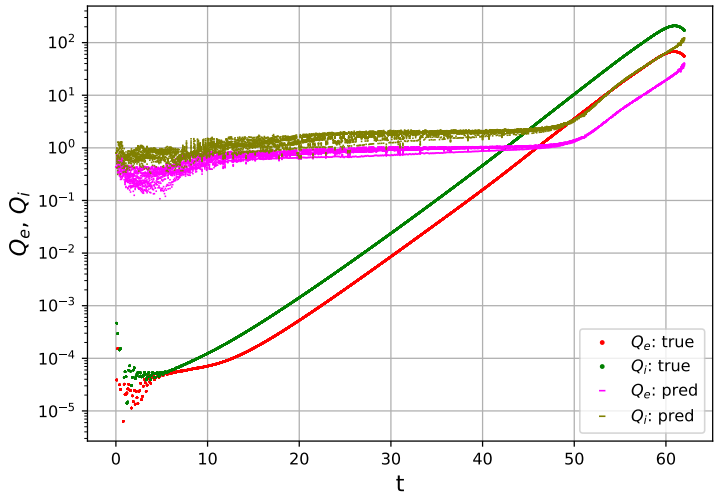

Figure 8 Time series graph of true heat fluxes for electrons (red) and ions (green) and predicted fluxes for electrons (magenta) and ions (olive).

The patterns of evolution that emerge in the linear and nonlinear growing phases vary depending on the dominant instability. The model does not reproduce the JT-60U (B) case well likely because the dominant instability of the CBC data used to train the model, viz., ITG/TEM, is different from this case, viz., pure ITG, as already suggested in a previous work[16].

5 | CONCLUSIONS AND DISCUSSION

A previously developed CNN-based model was extended to a multimodal model to predict the turbulent heat fluxes of electrons and ions and time. In addition to images, this model can read $|\tilde{\phi}|^2$ values as the absolute value inputs, which are directly linked to the heat flux in quasilinear modeling. The developed multimodal model was trained on CBC data, and it showed good performance for not only the CBC test data but also for the data based on the parameters taken from JT-60U #45072 at $\rho = 0.50$,

i.e., the JT-60U (A) case. Notably, the model was able to predict fluxes with high accuracy despite having no knowledge of the JT-60U (A) case. This result demonstrates the applicability of the multimodal model for supporting gyrokinetic simulations. However, the model performed poorly on the data which had dominant instability different from the CBC data, i.e., the JT-60U (B) case. In a previous work [16], pure ITG CBC and pure TEM CBC were additionally prepared and tests were done by applying the model to these cases. In this work, the multimodal model was not applied to these cases, but it can be attempted in future. However, we now understand that to obtain better prediction performance, multiple models trained on various data with different dominant instabilities must be prepared as representatives, and the best one must be employed for prediction based on a linear stability analysis performed in advance. The similarity of the linear stability between the data used for model training and for prediction may be a certain measure of the *distance* between data sets and hints at predictive performance.

A considerable amount of work remains for the future. As stated in Section 1, our ultimate goal is to develop a predictor of saturated fluxes based on a limited time-series of early-phase nonlinear simulation data. We are developing a combined recurrent neural network and CNN model to predict future flux on the basis of early-phase data by making use of the knowledge gained from the developement of the CNN-based multimodal model.

A gyrokinetic simulation outputs not only heat flux but also particle and momentum fluxes when the kinetic electron response is included. This study solely focused on the prediction of the heat flux for a proof-of-principle experiment, but in principle the predictions of the particle and momentum fluxes are possible in the same manner.

In this work, the model was trained solely on the well-known CBC data. However, this choice of data may not be always the best for this purpose. First, training on a variety of data sets may result in a more robust model. We will attempt to test this approach in the near future. Second, the time of the $|\tilde{\phi}|^2$ peak coincides with that of the Q_e and Q_i peaks in CBC, whereas in the other two cases the fluxes decrease before $|\tilde{\phi}|^2$ culminates. Deviations of the peak positions between $|\tilde{\phi}|^2$ and fluxes must influence prediction performance because folding back in flux prediction has been observed in both JT-60U cases, as seen in figure 6 and figure 7. Whether the selection of $|\tilde{\phi}|^2$ as input is appropriate to reproduce fluxes is also an issue. Some velocity space moments of the distribution functions can be used as input because the fluxes to be predicted are the quantities integrated over the distribution functions. Furthermore, the linear and nonlinear cross-phases have been suggested to agree well, and cross-phases could be a good quantity for the comparison of linear gyrokinetic simulations with experimental measurements [20]. Recall that the electrostatic turbulent heat flux could be written as $Q_{\mathrm{Es,k_y}} \sim k_y \sum_{k_x} \langle |\tilde{p}||\tilde{\phi}| \sin \delta_{\tilde{p},\tilde{\phi}} \rangle$, where the brackets denote the flux-surface average and $\delta_{\tilde{p},\tilde{\phi}}$ is defined as the cross-phase between the pressure fluctuation $\tilde{p}$ and $\tilde{\phi}$ [20]. Therefore, it is possible to develop a multimodal model, that reads the numerical values $|\tilde{p}|, |\tilde{\phi}|$ and the images of their cross-phases as input, to predict fluxes. Such a model is under development, and the results will be reported in the near future. Moreover, the outcome of a previous study [20] showing that cross-phases obtained through linear and nonlinear simulations are in agreement may help lead to the development of a model that can properly estimate saturated fluxes with information obtained by linear calculation.

Finally, we should mention the existence of long-time-scale dynamics in the saturation phase in gyrokinetic simulations [21]. The literature shows the cases in which the turbulence state essentially changes after saturation. Thus far, we have attempted to predict the fluxes in the saturation phase utilizing only the data before the saturation. Therefore, we anticipate that this phenomenon cannot be recovered by our approach. Developing a model that incorporates the data generated in the saturation phase to predict the final turbulence level is a future challenge.

ACKNOWLEDGMENTS

One of the authors (MH) would like to thank the anonymous referees for their valuable comments. This work was partly supported by JSPS KAKENHI Grant Number 22K03574 and 20K14450 and by MEXT as “Program for Promoting Researches on the Supercomputer Fugaku” (Exploration of burning plasma confinement physics: Project IDs: JPMXP1020200103, hp200127, hp210178 and hp220165) and used computational resources of ITO provided by Kyushu University.

References

[1] M. Kotschenreuther, G. Rewoldt and W.M. Tang, *Comput. Phys. Commun.* **1995**, 88, 128.

[2] J. Candy and R.E. Waltz, *J. Comput. Phys.* **2003**, 186, 545.

[3] J. Candy, E.A. Belli and R.V. Bravenec, *J. Comput. Phys.* **2016**, 324, 73.

[4] F. Jenko et al., *Phys. Plasmas* **2000**, 7, 1904.

[5] A.G. Peeters et al., *Comput. Phys. Commun.* **2009**, 180, 2650.

[6] T.-H. Watanabe and H. Sugama, *Nucl. Fusion* **2006**, 46, 24.

[7] A. Ishizawa et al., *Phys. Rev. Lett.* **2019**, 123, 025003.

[8] S. Xu, S. Maeyama and T.-H. Watanabe, *Nucl. Fusion* **2022**, 62, 064003.

[9] S. Maeyama et al., *Phys. Rev. Lett.* **2015**, 114, 255002.

[10] S. Maeyama, T.-H. Watanabe and A. Ishizawa, *Phys. Rev. Lett.* **2017**, 119, 195002.

[11] S. Maeyama et al., *Nat. Commun.* **2022**, 13, 3166.

[12] J. Citrin et al., *Nucl. Fusion* **2022**, 62, 086025.

[13] J. Candy et al., *Phys. Plasmas* **2009** 16, 060704.

[14] M. Barnes et al., *Phys. Plasmas* **2010** 17, 056109.

[15] N.R. Mandell et al., submitted to *J. Plasma Phys.*

[16] E. Narita, M. Honda, S. Maeyama and T.-H. Watanabe, *Nucl. Fusion* **2022**, 62, 086037.

[17] M. Tan and Q.V. Le, *arXiv:1905.11946* **2019**

[18] A.M. Dimits et al., *Phys. Plasmas* **2000**, 7, 969.

[19] C. Bourdelle et al., *Phys. Plasmas* **2007**, 14, 112501.

[20] A. Bañón Navarro et al., *Phys. Plasmas* **2015**, 22, 042513.

[21] A. Di Siena et al., *Nucl. Fusion* **2019**, 59, 124001.

How to cite this article: M. Honda, E. Narita, S. Maeyama, and T.-H. Watanabe (2023), Multimodal convolutional neural networks for predicting evolution of gyrokinetic simulations, *Contrib. Plasma Phys.*, *2023;00:1–6*.