Neural Network Model of the Multi-Mode Anomalous Transport Module: Real-Time Model for Simulations

Shira Morosohk, Andres Pajares, Tariq Rafiq, and Eugenio Schuster

Lehigh University, Bethlehem, Pennsylvania 18015, USA
E-mail: morosohk@lehigh.edu

Control-Oriented Models Needed in Nuclear Fusion

- Tokamaks must operate stably for competitive energy generation
  - Need to be carefully controlled for performance while avoiding instabilities
- Many control schemes require models of plasma response to actuation
  - Many physics-oriented models are too computationally intensive
  - Need models with faster calculation times for control applications
- Approaches to control-oriented modeling:
  - Analytical models
    - Simple enough models are not always available
  - Empirical scaling laws
    - May only be valid for specific scenarios
  - Machine learning
    - Can replicate the outputs of a function with extremely fast calculation times
    - If trained well, calculation speed does not come with a large drop in accuracy
    - Only valid within the range of the training data

Neural Networks Enable Faster Predictive Modeling for Control and Real-Time Applications

- Multi-Mode Model (MMM) is a physics-based turbulent transport model
  - Verified against first principle simulations
  - Extensively validated against experimental data
  - Takes too long to run for control applications
- Existing control-oriented transport models can run quickly, but may sacrifice some prediction accuracy
- Neural networks versions of computationally-intensive models have recently been developed, e.g.:
  - TGLF, EPED: Meneghini NF 2017, 2014
  - QuaLiKiz: Citrin NF 2015
  - NUBEAM: Boyer NF 2019, Morosohk FED 2021
- **Can a neural network (MMMnet) reproduce the results of MMM?**

Some Neural Network Terminology

- Principle Component Analysis:
  - Project each profile onto a set of basis functions
  - Profile reduced to the coefficients of a linear combination of basis functions
  - Limits the number of data points necessary to describe spatially varying data
- Hyperparameter:
  - Any parameter of the neural network that is assigned as opposed to learned
- Grid search:
  - Testing different combinations of hyperparameters by:
    1. Training the network
    2. Evaluating its performance
- Epoch:
  - Using each observation in the training data set once during training
  - Training usually takes multiple epochs, but too many can cause overfitting
- Overfitting:
  - Neural network learns the training data better than the underlying function
  - Neural network performs worse on validation and testing data

MMMnet was Developed Using PCA, Grid Searches

- Accelerated predictive modeling will enable more sophisticated model-based scenario planning and control of tokamak plasmas
- The neural network model of MMM enables rapid turbulent ion and electron thermal and momentum diffusivity predictions
- Spatially-varying profile data has been simplified using Principle Component Analysis (PCA) to reduce the complexity of the network
- Grid searches used to determine optimal hyperparameters
  - Network architecture chosen to:
    1. Maximize prediction accuracy
    2. Minimize calculation time per prediction
  - Training parameters chosen to maximize accuracy
- Initial predictions made to test for accuracy and evaluation speed

Inputs and Outputs for MMMnet Taken from MMM

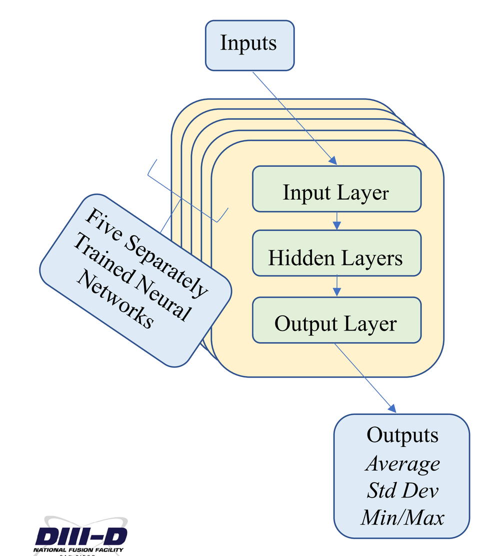
- Inputs to the NN are the same as the inputs to MMM as a standalone

	Symbol	Explanation	Gradient also used
Inputs	Z_{imp}	Mean charge of impurities	
	A_{imp}	Mean mass of impurities	
	R	Major radius	
	a	Minor radius	
	B_{tor}	Toroidal magnetic field	
	n_e	Electron density	×
	n_i	Ion density	×
	n_h	Hydrogenic thermal particle density	×
	n_{imp}	Impurity ion density	×
	n_{fast}	Fast ion density	
	T_e	Electron temperature	×
	T_i	Thermal ion temperature	×
	q	Safety factor	×
	$\Omega_{E\times B}$	ExB shearing rate	
	v_ϕ	Toroidal velocity	×
	v_θ	Poloidal velocity	×
Outputs	χ_i	Turbulent ion thermal diffusivity	
	χ_e	Turbulent electron thermal diffusivity	
	χ_ϕ	Turbulent toroidal momentum diffusivity	

A Dataset Was Prepared Based on TRANSP Runs Executed for Recent DIII-D Experiments

- Called 1000 new TRANSP runs
  - Based on 83 existing between-shots TRANSP runs
  - Uniformly varied Z_{eff} from 1.5 to 5, edge neutral density from 5.0×10^{10} to 1.0×10^{13} cm^{-3} using random number generator
  - Assigned fast ion diffusivity profile to be either zero, flat, or peaked with a maximum ranging from 1 to $50,000$ cm^2/s
- **Training** data set (80% of data) used to train neural network model
- **Validation** data set (10% of data) used to determine optimal values of hyperparameters, e.g. network architecture
- **Testing** data set (10% of data) used to assess network prediction accuracy and calculation time on data it has not trained on

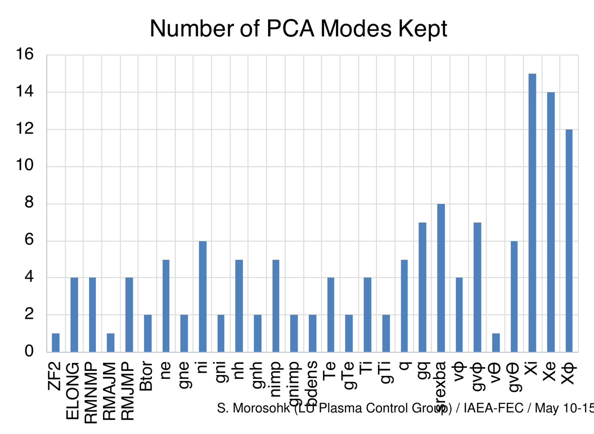
Multi-Layer Perceptron Chosen as Network Structure



- Multi-Layer Perceptrons (MLPs) are a simple type of feedforward neural network with at least one hidden layer
- Hyperparameters determined by trial and error (see slides 11-13)
- Final Hyperparameters:
  - Hidden layers: 3
  - Nodes/hidden layer: 100
  - Batch size: 9
  - Epochs: 16
  - Solver: adam
  - Loss: mean squared error
  - Metrics: accuracy
  - Hidden layer activation function: relu
  - Output layer activation function: linear

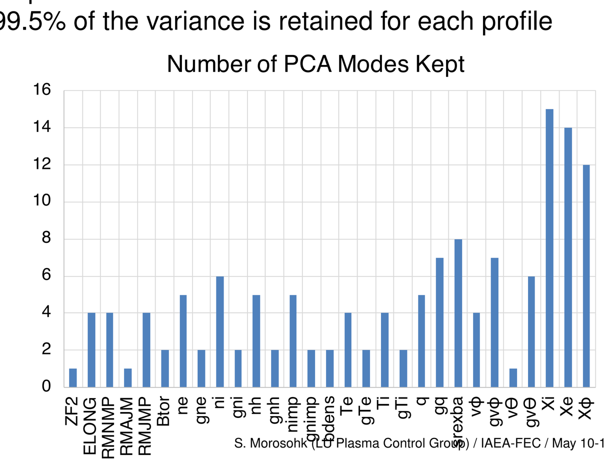
Principle Component Analysis Used to Reduce Profiles to Scalar Quantities

- MLPs are not designed to handle spatially-varying data
- Reduce each profile to the coefficients of basis functions
- Keep modes that explain at least 0.1% of the variance
- Ensures at least 99.5% of the variance is retained for each profile



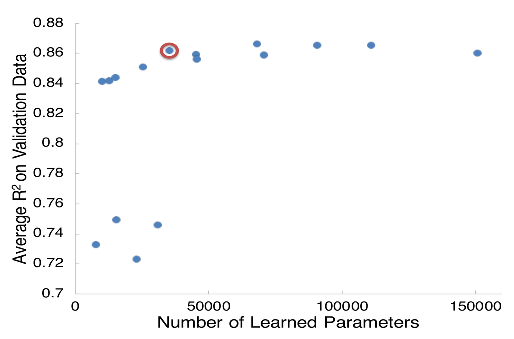
Principle Component Analysis Used to Reduce Profiles to Scalar Quantities

- MLPs are not designed to handle spatially-varying data
- Reduce each profile to the coefficients of basis functions
- Keep modes that explain at least 0.1% of the variance
- Ensures at least 99.5% of the variance is retained for each profile

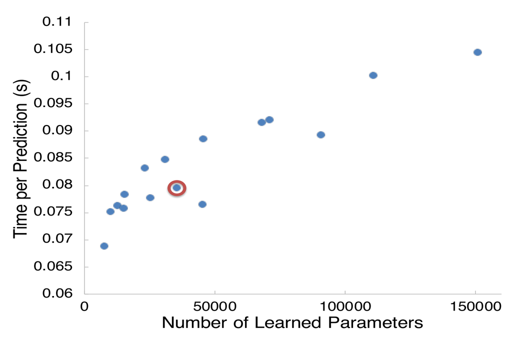


Network Architecture Was Chosen By Grid Search

- The correlation (R^2) between MMMnet and MMM, as well as the MMMnet calculation time, are shown for different numbers of learned parameters
- The red circled points represent the chosen architecture of 3 hidden layers with 100 nodes per layer, or 35,500 learned parameters



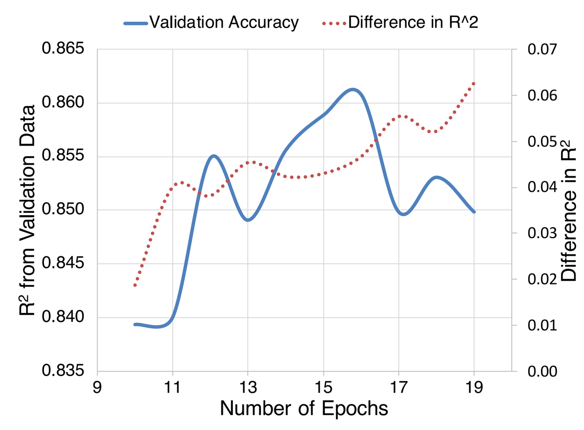
Average R^2 for validation data



Time per prediction

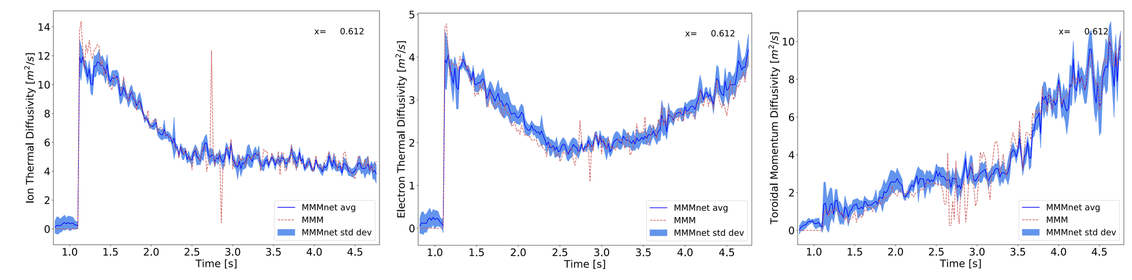
Number of Epochs Was Chosen By Grid Search

- Choose number of epochs to maximize accuracy and minimize difference in accuracy between training and validation data
- Increasing difference in correlation with more epochs indicates overfitting
- Final model uses 16 epochs



Time Traces of Neural Network Predictions Match MMM Closely for Shots in Testing Data Set

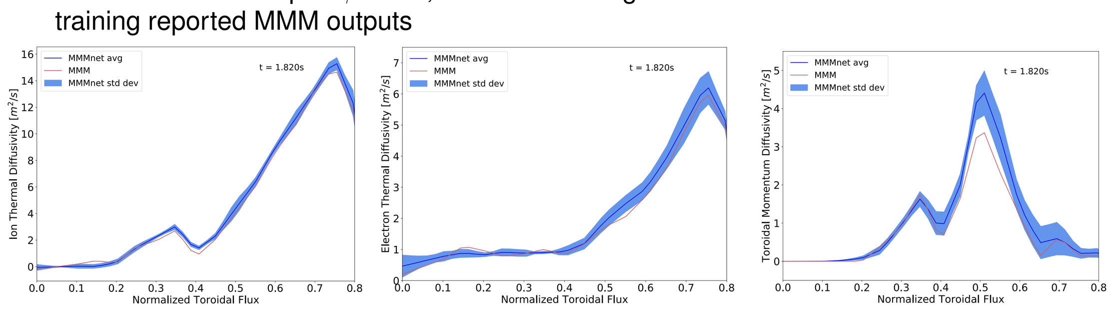
- Prediction of TRANSP run 176052T05 from the testing dataset
- MMMnet is able to follow changes throughout the course of the shot
- Shaded blue area is one standard deviation from the mean prediction
- Uncertainty is higher when data is fluctuating more



χ_i χ_e χ_ϕ

Profile Outputs Show Good Agreement Between MMM and Neural Network Prediction

- Prediction of TRANSP run 176052T05 from the testing dataset
- MMMnet is capable of reconstructing complex profile shapes by using reduced number of modes arising from Principle Component Analysis
- Predictions shown up to $\hat{\psi}=0.8$, which is the range for which the TRANSP runs used for training reported MMM outputs



χ_i χ_e χ_ϕ

Neural Network has Capability to Generate Accurate Predictions at Speed Useful for Control Applications

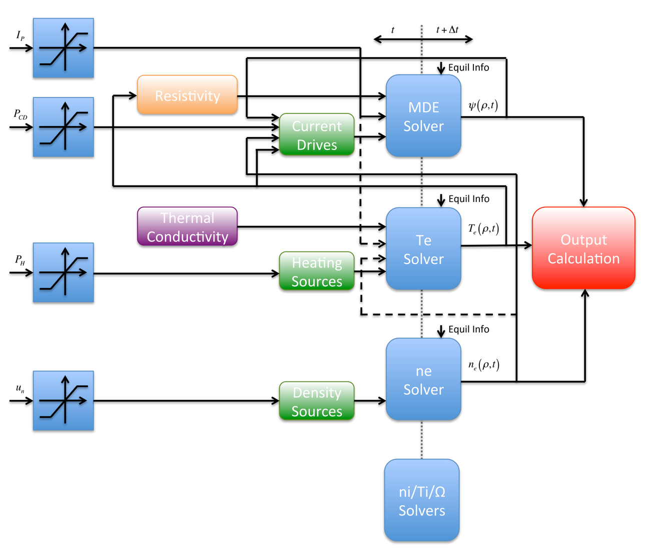
- In **1.35 ms** (using Cython), network calculates for one time step:
  - Data preprocessing and postprocessing
  - Prediction of all outputs 5 times
    - 5 separate neural networks account for randomness in weight initialization
  - Average and standard deviation of the 5 values
- Average predictions correlate to MMM data with good accuracy:

	R^2 values: training data	R^2 values: testing data
χ_i	0.961	0.883
χ_e	0.941	0.843
χ_ϕ	0.928	0.878

COTSIM Predicts Evolution of Critical Plasma States

- Goals:
  - Simulate full shots fast enough to be useful for control applications
    - Offline: iterative control design, between-shots scenario planning, etc.
    - Real time: state estimation, optimization-based feedback control, etc.
    - Faster than real time: state forecasting, etc.
  - Be able to run with prescribed actuator inputs or test a feedback controller
  - Configured for control of scalars, profiles, or both
- Current capabilities:
  - Solves transport equations for poloidal stream function (ψ), electron temperature (T_e), ion rotation (Ω)
  - Uses lower complexity models for T_i, n_e, n_i
  - Uses a prescribed equilibrium or a fixed-boundary analytical solver
- Current upgrade work:
  - Add transport equations for ion temperature (T_i), electron/ion densities (n_e/n_i)
  - Integrate a free-boundary numerical equilibrium solver

COTSIM's Modularity Allows for User-defined Complexity



MMMnet Predicts Diffusivity Used by COTSIM to Solve Heat Transport Equation

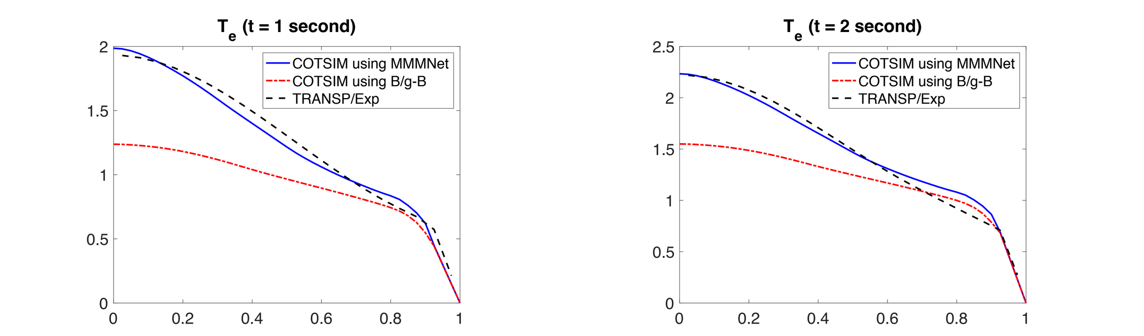
- COTSIM uses a modular configuration
  - Allows the user to choose which model to use for each step in the simulation
  - All available models need to meet speed requirements
- χ_e used in electron heat transport equation

$$\frac{3}{2}\frac{\partial}{\partial t}[n_e T_e] = \frac{1}{\rho_b^2 \hat{H}}\frac{1}{\hat{\rho}}\frac{\partial}{\partial \hat{\rho}}\left[\hat{\rho}\frac{\hat{G}\hat{H}^2}{\hat{F}}\left(\chi_e n_e \frac{\partial T_e}{\partial \hat{\rho}}\right)\right] + Q_e \quad (1)$$

- Current options to calculate χ_e include Bohm/gyro-Bohm, Coppi-Tang anomalous models, Chang-Hinton neoclassical model
- Add option to use MMMnet instead
- As more transport equations are added, more diffusivities will be needed

Use of MMMnet Allows COTSIM To Predict T_e Closer to Experimental Profile: Shot 147634

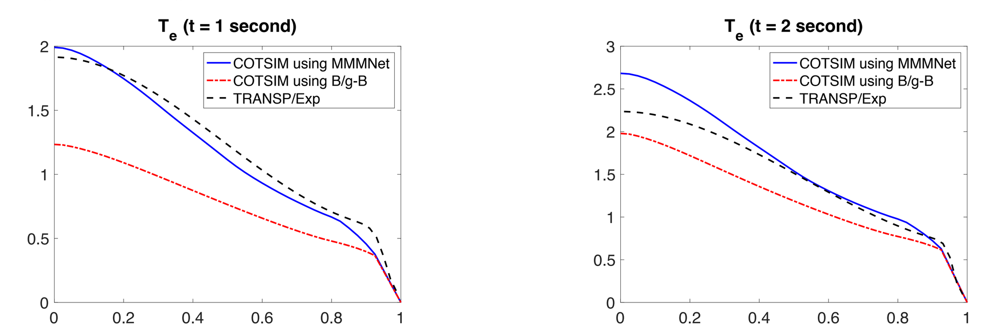
- Both COTSIM simulations use the same pedestal model, so they match each other exactly at the edge
- COTSIM using MMMnet predict a T_e profile that is closer to the experimental profile in both shape and magnitude



Prediction of the T_e profile (keV) Prediction of the T_e profile (keV)

Use of MMMnet Allows COTSIM To Predict T_e Closer to Experimental Profile: Shot 147621

- Choose a shot from the same experiment with the same equilibrium
- Use the same model except for a higher value of B_0 to match the new shot
- Using MMMnet gives a T_e profile closer to the experimental profile



Prediction of the T_e profile (keV) Prediction of the T_e profile (keV)

Future Plans for Improving Neural Network

- Include outer edge of plasma in predictions
  - MMM will have H-mode pedestal capabilities soon
- Add poloidal momentum, impurity, electron particle diffusivities as outputs
- Improve network speed by:
  - Calling network using executable
  - Training separate network for each output
  - Run 5 separate networks using parallel computing
  - Decreasing complexity of networks
    - Sacrifices some accuracy
- Consider using convolutional neural network
  - Better able to handle spatially varying data
  - May require more data, more computationally intensive
  - More complicated architecture means longer prediction times
- Integrate MMMnet into DIII-D Plasma Control System

Future Work Towards NN Model Integration in COTSIM

- Further testing of MMMnet χ_e predictions
  - Simulate shots and make comparison for different plasma scenarios
    - Empirical models have been tuned to this particular scenario
    - MMMnet is trained on data from many different plasma scenarios
    - Evaluate potential benefit of using MMMnet over empirical models
- Use MMMnet χ_ϕ prediction in solving rotation equation
  - Need χ_ϕ prediction across the whole spatial profile
- Add more neural network options to COTSIM
  - NUBEAM (Morosohk FED 2021)
    - Option already available in COTSIM to calculate heating, torque, and current drive from neutral beams
  - TGLF (Meneghini NF 2017)
    - Need transport equations in COTSIM that can handle TGLFNN outputs of flux, including diffusive and convective components
  - GENRAY/CQL3D
    - Neural network models being developed by MIT