NSTX-U

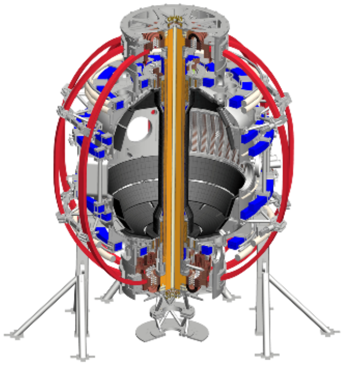
NSTX-U is sponsored by the
U.S. Department of Energy Office of Science
Fusion Energy Sciences

NubeamNet: Accelerated predictive modeling of NSTX-U beam deposition for optimization and control

M.D. Boyer[1], K. Erickson[1], S. Kaye[1], V. Gajaraj[3], J. Kunimune[2], M. Zarnstorff[1]

60th Annual meeting of the APS Division of Plasma Physics
November 5-9, 2018
Portland, Oregon

1 PPPL
2 Olin College of Engineering
3 New York University



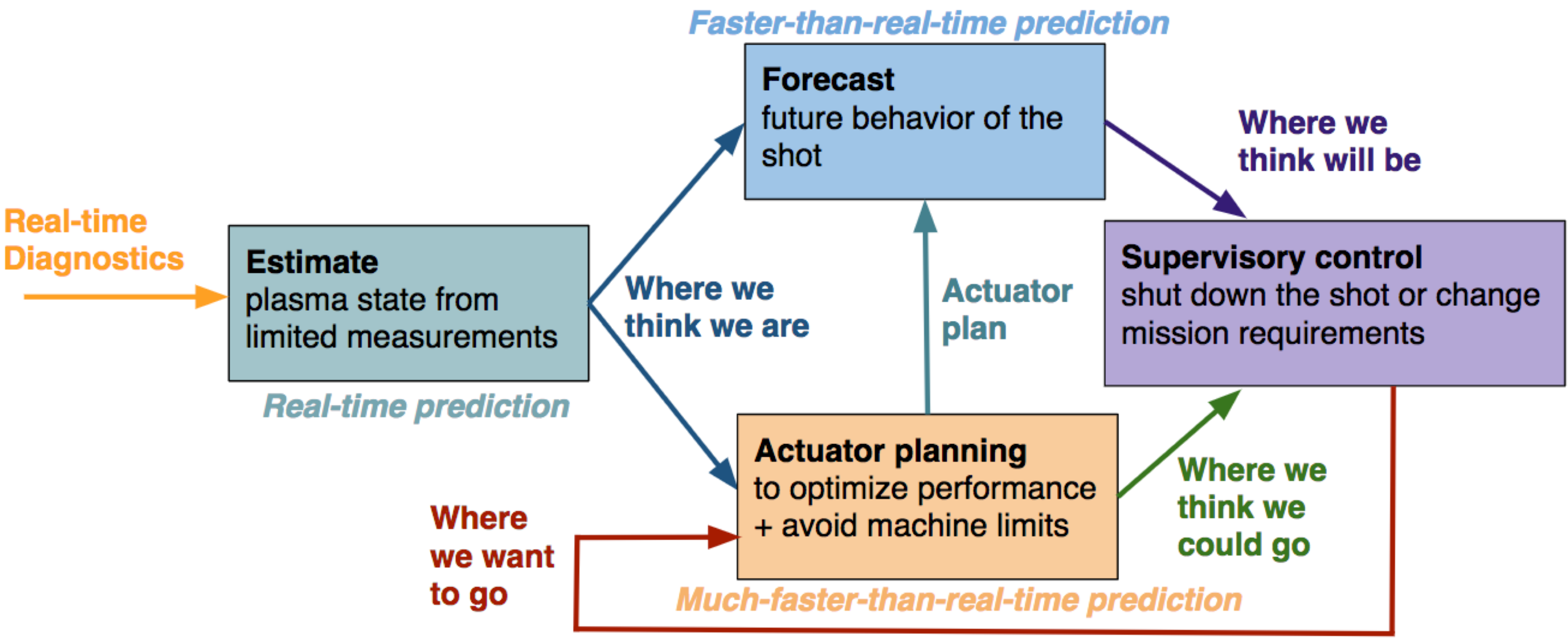
Overview

- Accelerated predictive modeling will enable more sophisticated model-based control of tokamak plasmas
- To enable rapid beam deposition prediction, a neural network model trained on NUBEAM results has been generated
- Dimensionality reduction and input augmentation used to overcome challenges of the problem (spatially varying profile data, time-history dependence)
  - Avoids need for recurrent, convolutional neural network, though this option will be studied as an alternative
- Initial scans of model topology completed for accuracy and real-time evaluation time
- Initial applications demonstrated
  - Current profile observer with Zeff and fast ion diffusivity estimation

This research was supported by the U.S. Department of Energy under contract number DE-AC02-09CH11466.

Advanced control capabilities can enable online shot planning to optimize experimental operations and avoid machine limits

Will require reduced model based control and optimization techniques



Can we make our models fast enough?

Accelerated predictive modeling using neural networks can enable more sophisticated real-time algorithms

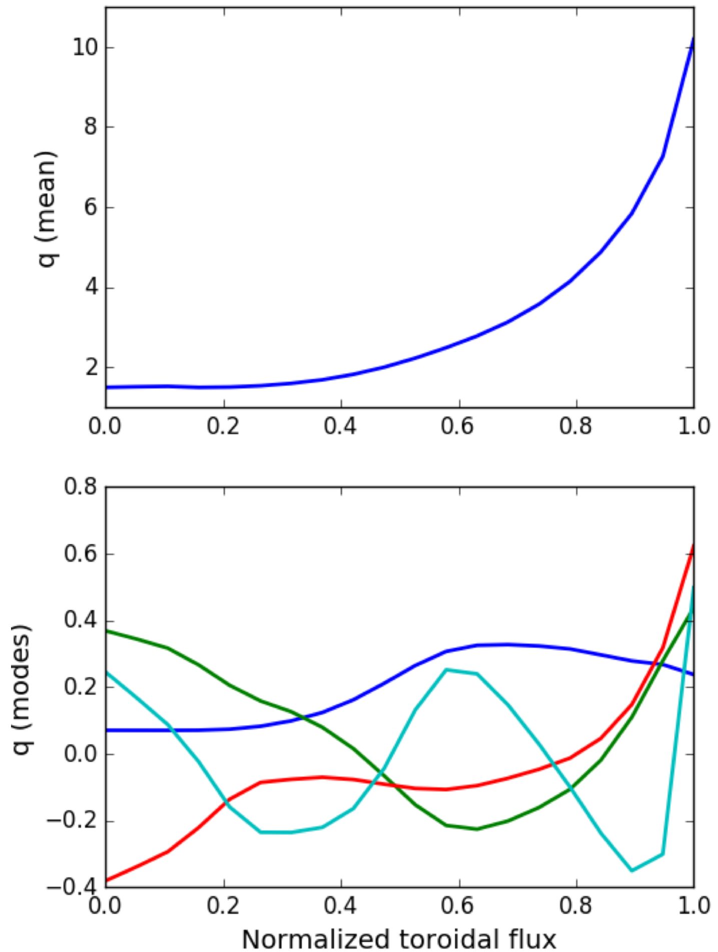
- Neural networks have recently been developed for approximating the results of computationally intensive calculations
  - Meneghini NF 2017, 2014 (TGLF, EPED), Citrin NF 2015 (QuaLiKiz)
- NUBEAM often takes 30% or more of TRANSP time
  - Lower fidelity settings can speed up results but results become noisy
- **Can a neural network be trained to reproduce the result of NUBEAM?**
- Potential applications
  - Fast but realistic beam calculations for control-oriented simulations or use in real-time predictive control algorithms
  - Fast predictions to optimize neutron rate matching in TRANSP runs
  - Prediction of fast ion pressure profile for real-time kinetic EFITs
  - Control room tools to explore beam timing options prior to shot

A data set was prepared based on the TRANSP runs performed between NSTX-U shots (BEAST)

- Expanded the dataset with a **scan of Z_{eff}, anomalous fast ion diffusivity, and edge neutral density**
  - Randomly selected **~1000 cases** from the grid scan to actually run for initial testing
  - Used **10000 particles, 5ms NUBEAM time step**
- Assigned 10% of ~300 shots in the dataset to the **'testing' data set,** another 10% to **'validation' data set**
  - No data from any simulations of the test shots is used in training the model
  - Data from the validation shots used to compare performance of different model parameters
- Total of **~100k time slices**

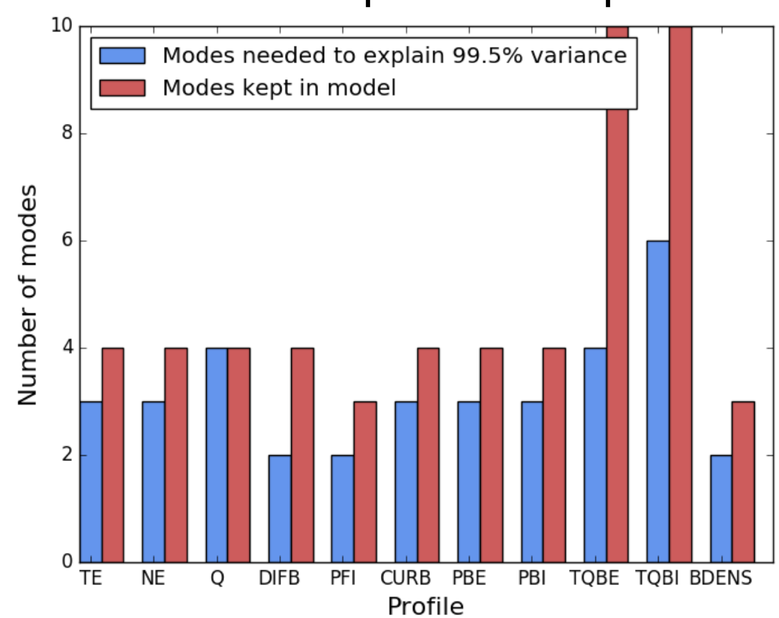
To reduce dimensionality, spatial profiles in dataset were projected onto a reduced set of modes

Example mean and modes:
Safety factor profile



- Principle component analysis of training dataset used to identify most significant modes of each profile
  - Projected profile data onto reduced set of modes, keeping enough modes to describe >99.5% of data variance

modes kept for each profile

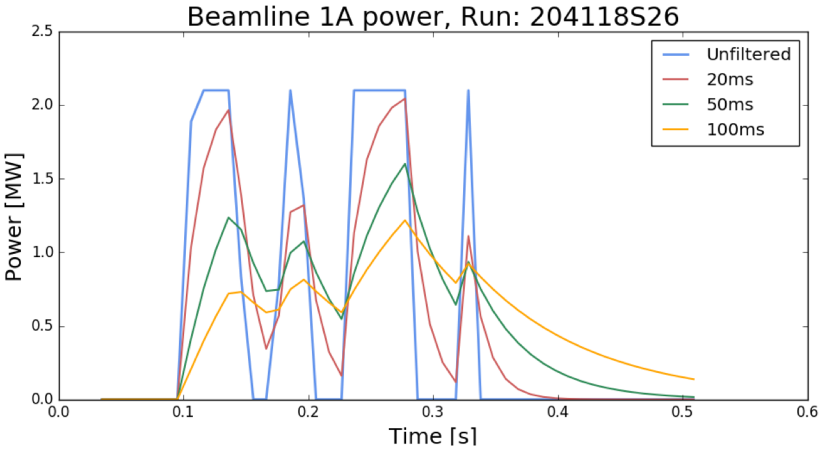


The beam slowing down time causes NUBEAM results to depend on time history

- **Simplest** approach to modeling:
  - Ignore time history, assume **steady-state**, only use instantaneous values of inputs
  - Probably not always suitable for planned applications
    - e.g., Beam modulation during control

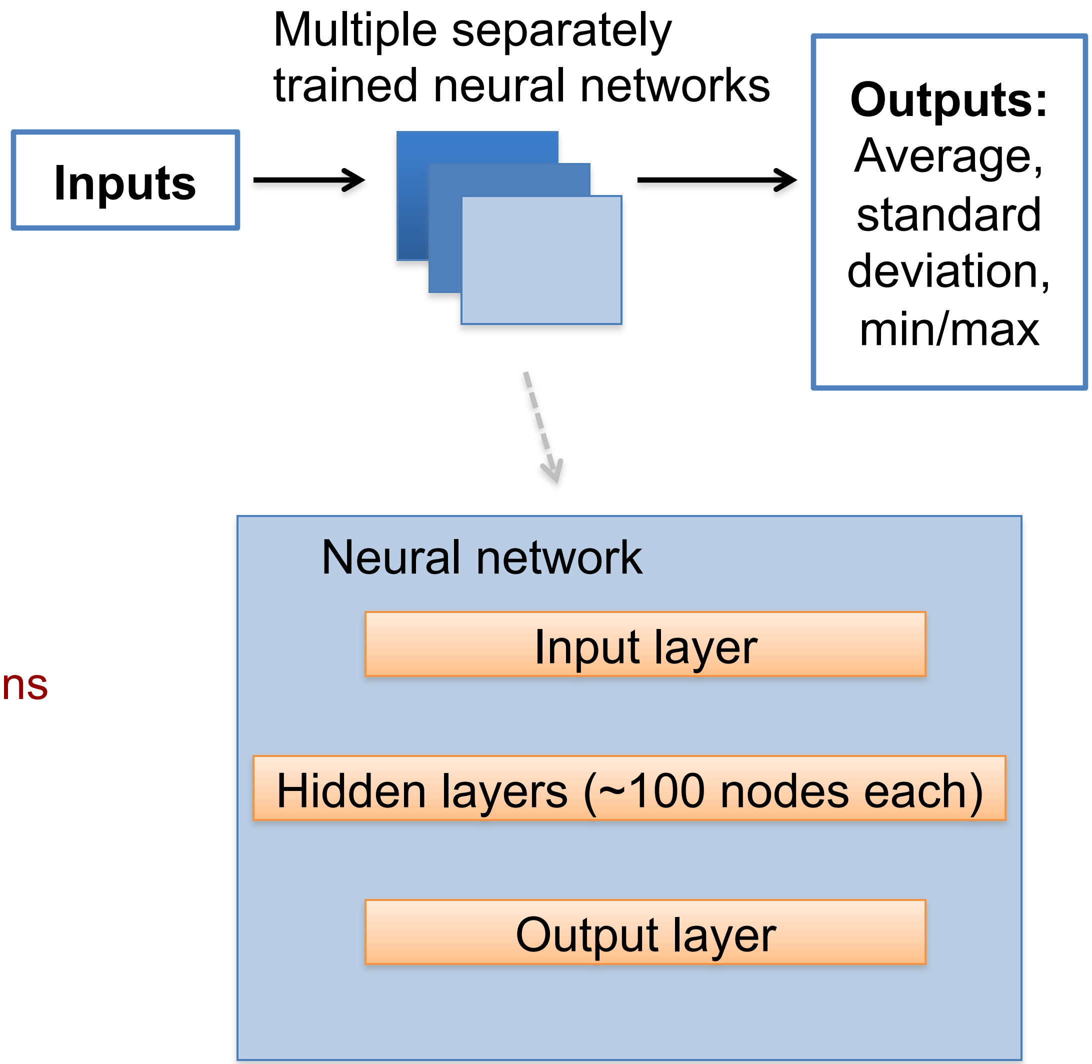
- The **next simplest** approach:
  - Expand inputs with **filtered beam powers**
    - **Multiple time constants** to account for changes in slowing down time
  - Not accounting for time history of plasma parameters
    - Fewer inputs, fewer nodes to train on
    - Plasma parameters evolve fairly slowly compared to slowing down time and beam modulation time

- Future work: recurrent NN
  - More difficult to train, but may be better suited to handling time variation of all inputs (without greatly expanding the number of inputs through filtering)

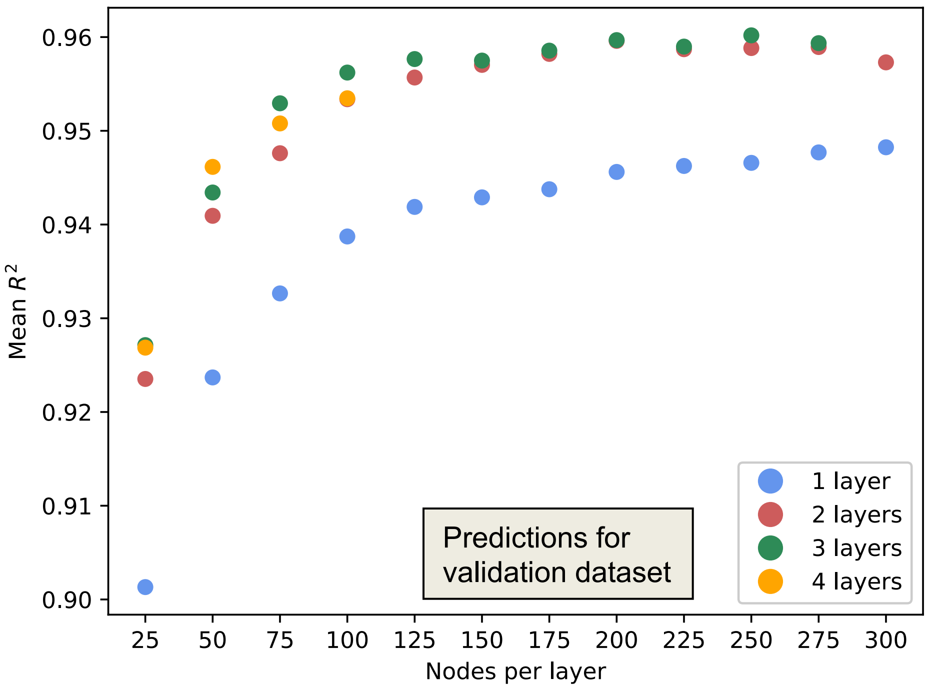


Inputs, outputs, and topology of the neural network model

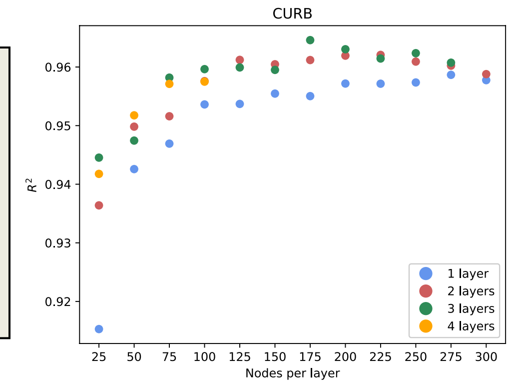
- Inputs:
  - Profiles:
    - T_e, n_e, q
    - *fast ion diffusivity*
  - Scalars:
    - Beam powers
    - *Edge neutral density*
    - Z_{eff}
    - Shape parameters
- Outputs:
  - Profiles:
    - Beam heating to ions/electrons
    - Beam driven current
    - Beam torque
    - Fast ion pressure
  - Scalars:
    - Neutron rate
    - Shine through
    - Charge-exchange and orbit loss



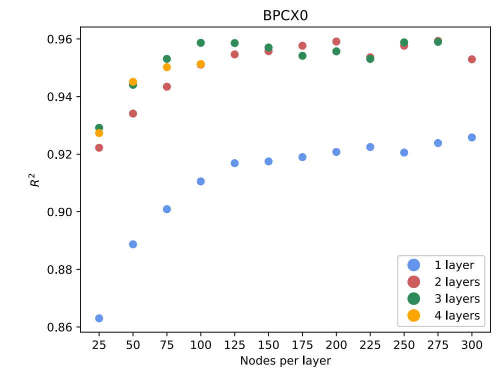
Initial scan of neural network topology used to assess accuracy vs. complexity trade-off



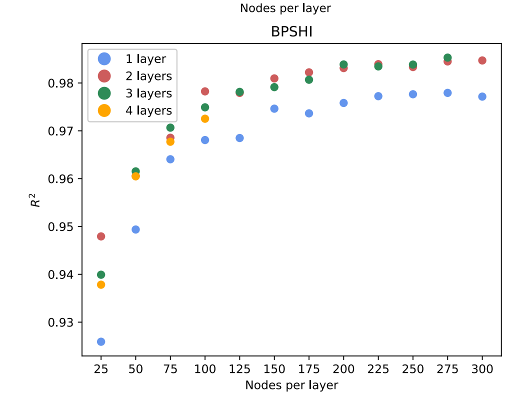
Very sensitive to # of layers at small # of nodes,
But much less sensitive at larger # nodes



Even large # nodes in a single layer barely matches a few nodes in 2 layers

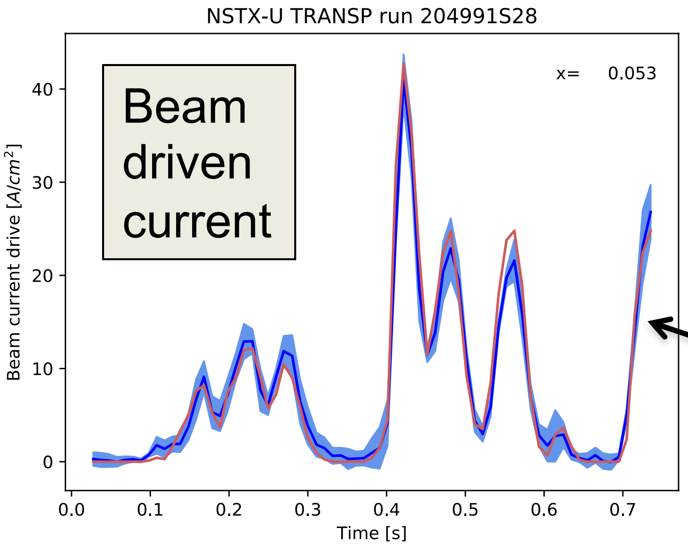


Continued improvement even at large # nodes per layer

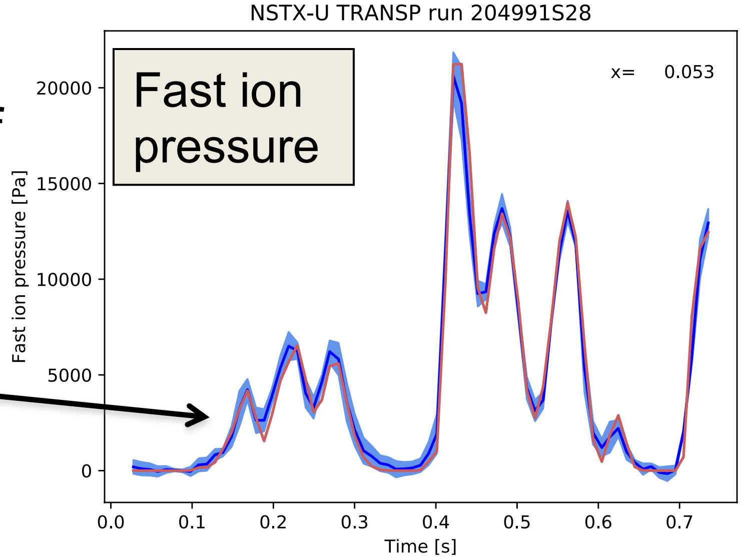


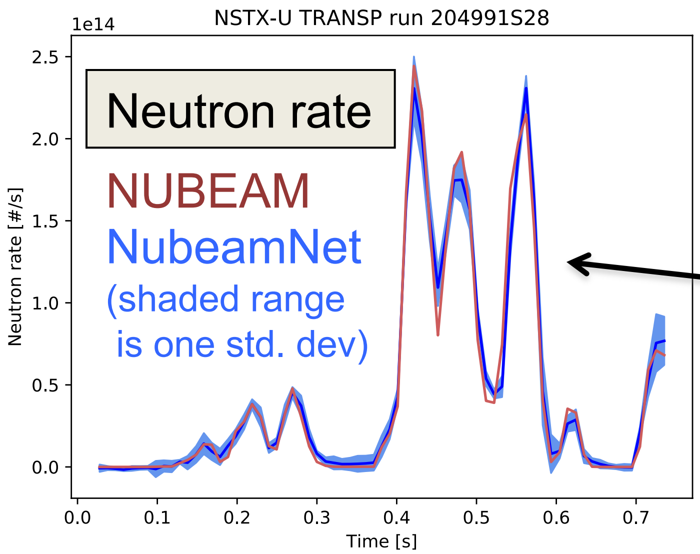
- Clear benefit to having 2-3 layers and 50-100 nodes per layer (diminishing returns beyond this)
- Each quantity has unique response to topology changes
  - May be optimal to have different # of nodes in each layer
  - Optimization of topology will be studied

Time traces of NN compare well with NUBEAM for shots in testing data set

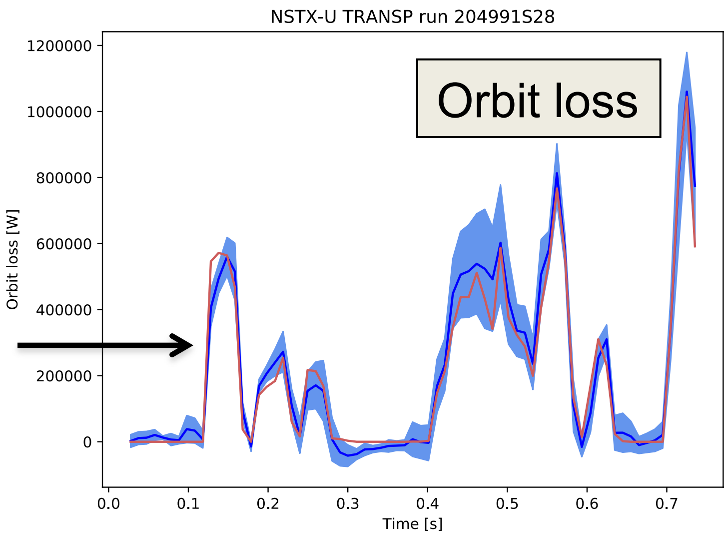


- Good matching of time history of current drive and fast ion pressure near axis

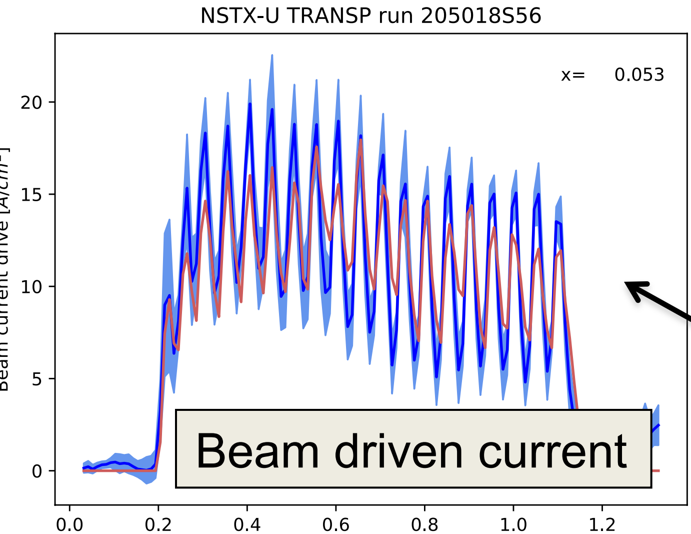




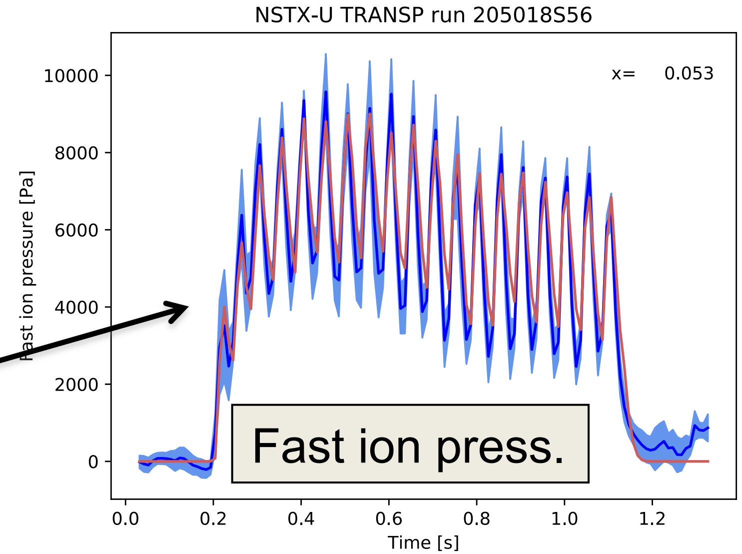
- Also good matching of scalars like neutron rate and orbit loss



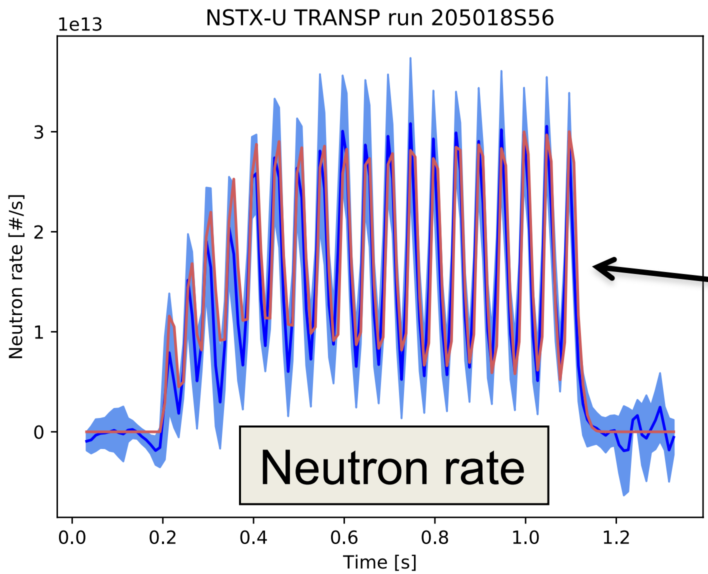
Time traces compare fairly well during beam blip shots in testing data set



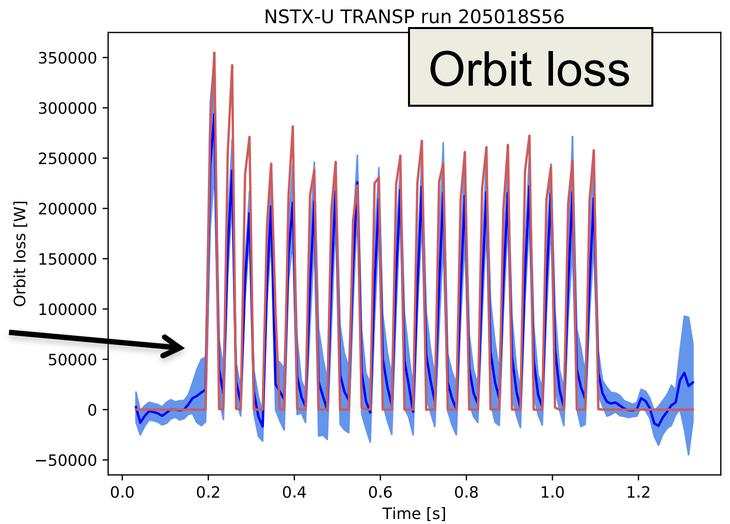
- Good matching of time history of current drive and fast ion pressure near axis



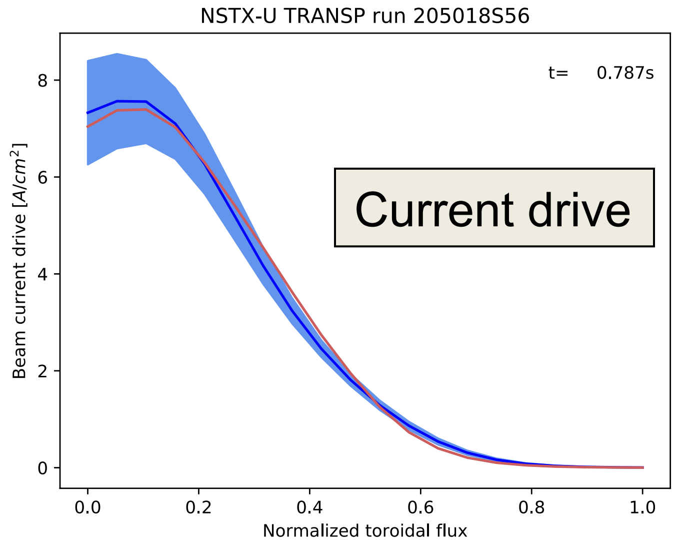
NUBEAM
NubeamNet



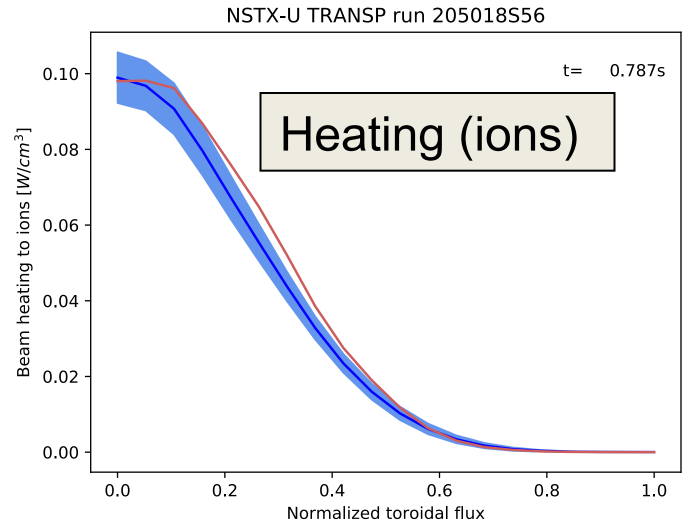
- Also good matching of scalars like neutron rate and orbit loss

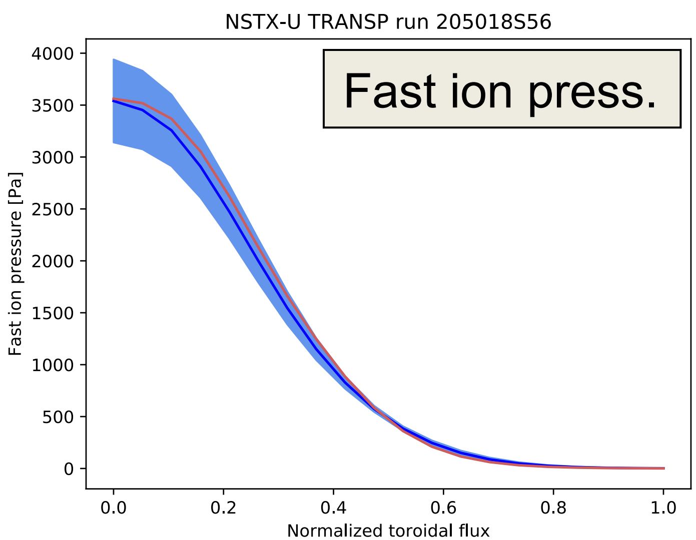


Profiles show good agreement between NUBEAM and neural network prediction

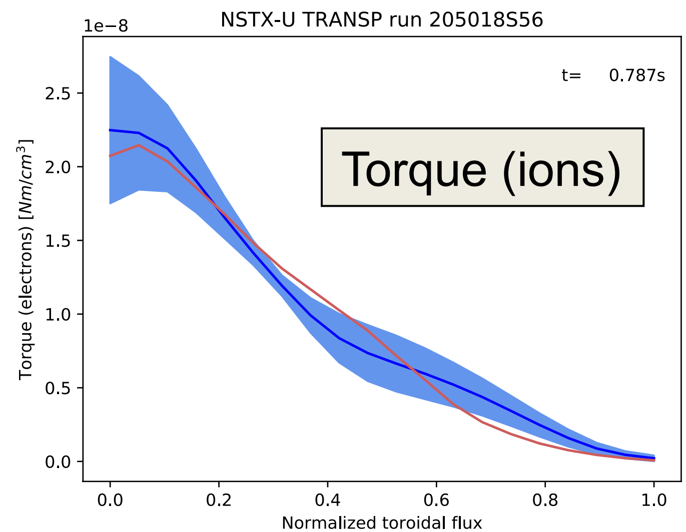


- Generally good matching, smooth profile predictions
- Torque from NUBEAM for some runs is spatially noisy – NN smoothes this out





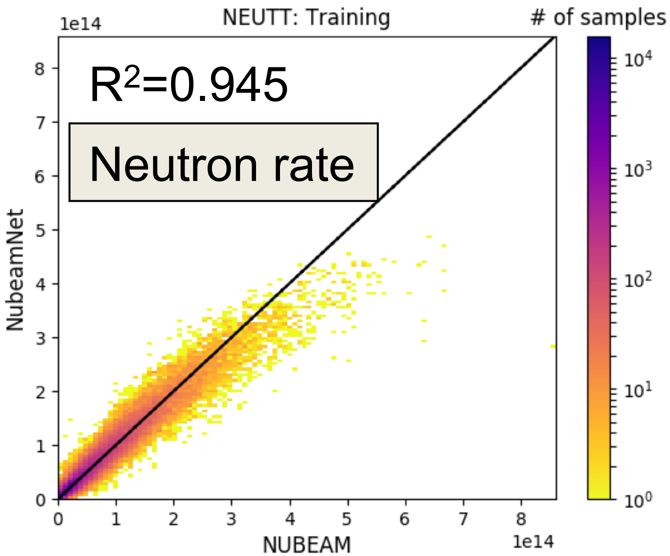
NUBEAM
NubeamNet

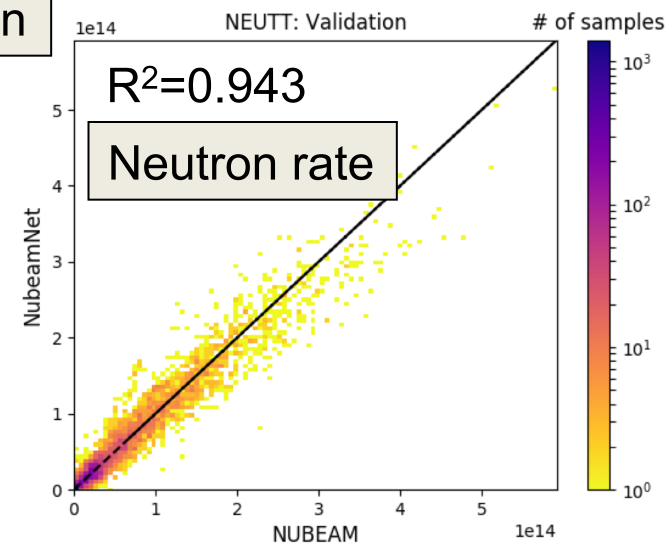


Regression plots for training and validation data set show good fitting and generalization

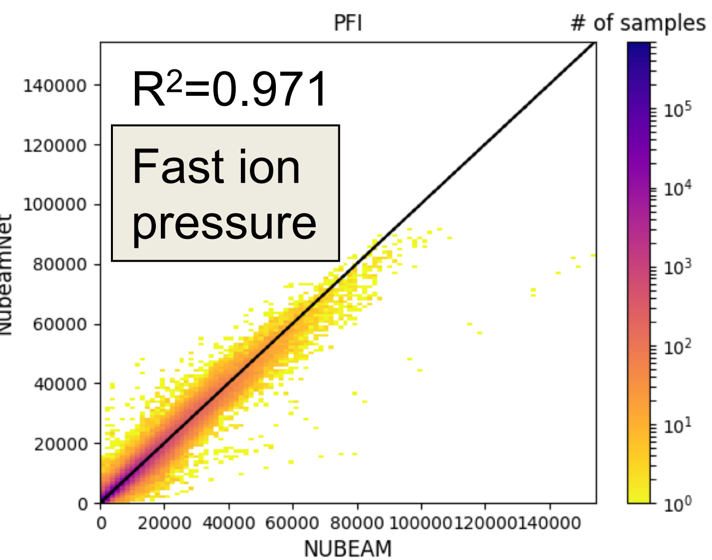
Training

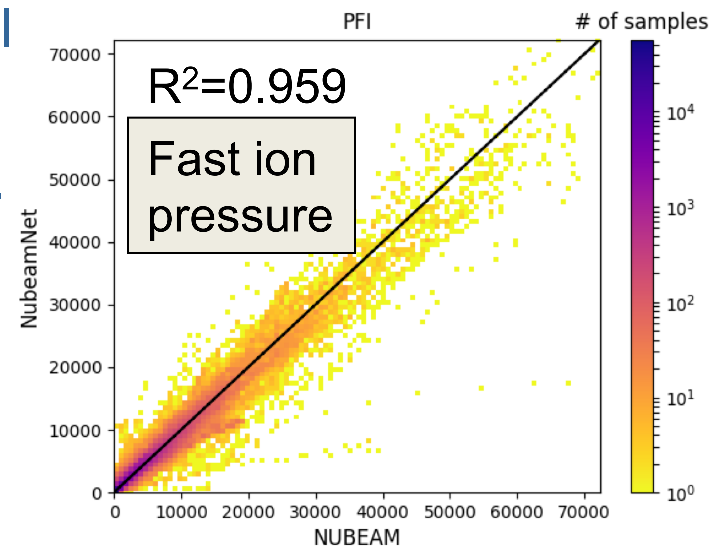
Validation





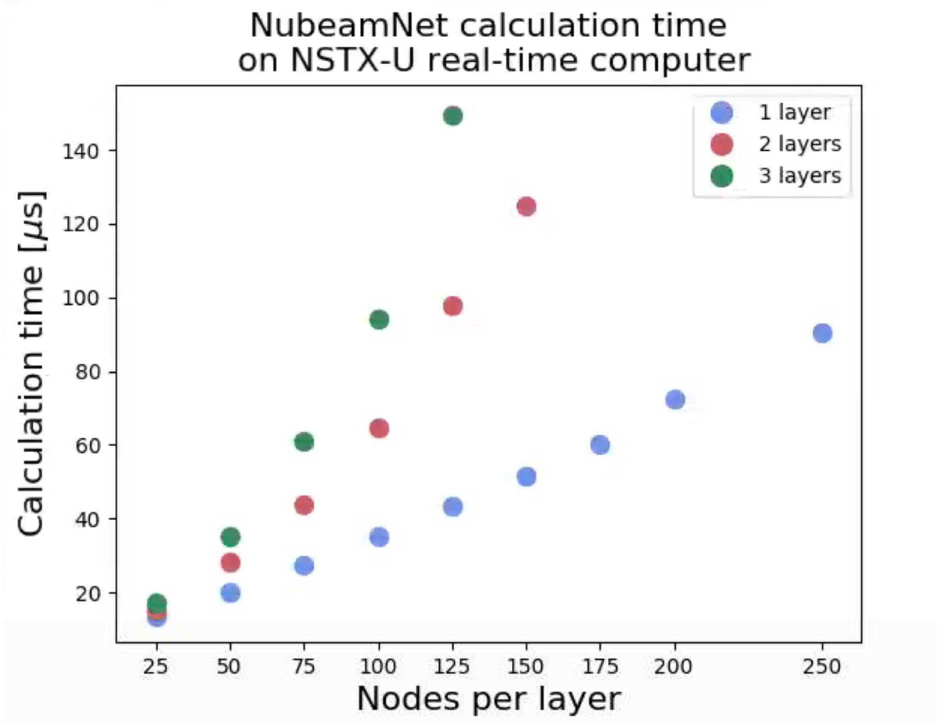
- **Log scale histograms**
- R^2 drops in testing data set but not too bad
  - Will continue to optimize neural network topology, add more data, etc. to improve generalization



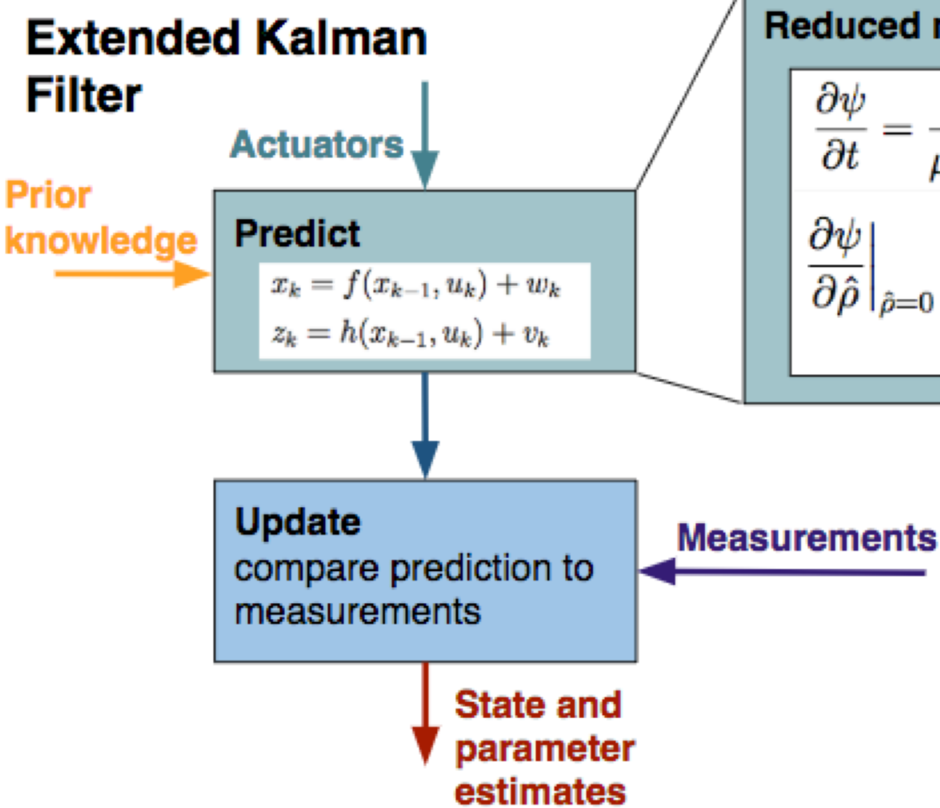


Timing tests of real-time implementation demonstrate sub-ms prediction time

- Model implemented in C++ and tested on NSTX-U real-time computer (64 cores, 2.8Ghz, real-time kernel)
  - Test included dimensionality reduction, normalization, neural network evaluation, and projection of outputs
- Scan of number of layers and nodes per layer
- Uncertainty quantification and sensitivity analysis:
  - Parallel model prediction exploiting multiple cores and advances in internodal communication
- Sub-ms timing results well-suited to faster-than-real-time prediction goals
  - E.g., nonlinear model-predictive current profile control



NubeamNet enables real-time current profile observer with estimation of Zeff and fast ion diffusivity



Reduced magnetic diffusion equation + NubeamNet

$$\frac{\partial\psi}{\partial t} = \frac{\eta(T_e)}{\mu_0\rho_b^2\hat{F}^2}\frac{1}{\hat{\rho}}\frac{\partial}{\partial\hat{\rho}}\left(\hat{\rho}\hat{F}\hat{G}\hat{H}\frac{\partial\psi}{\partial\hat{\rho}}\right) + R_0\hat{H}\eta(T_e)\frac{\langle\bar{j}_{NI}\cdot\bar{B}\rangle}{B_{\phi,0}}$$

$$\left.\frac{\partial\psi}{\partial\hat{\rho}}\right|_{\hat{\rho}=0} = 0, \qquad \left.\frac{\partial\psi}{\partial\hat{\rho}}\right|_{\hat{\rho}=1} = -\frac{\mu_0}{2\pi}\frac{R_0}{\hat{G}\big|_{\hat{\rho}=1}\hat{H}\big|_{\hat{\rho}=1}}I(t),$$

Measurements
Neutron rate, plasma current, and in-domain poloidal flux gradient (opt.)

States and parameters
Poloidal flux +
Zeff, and fast ion diffusivity (assumed flat)

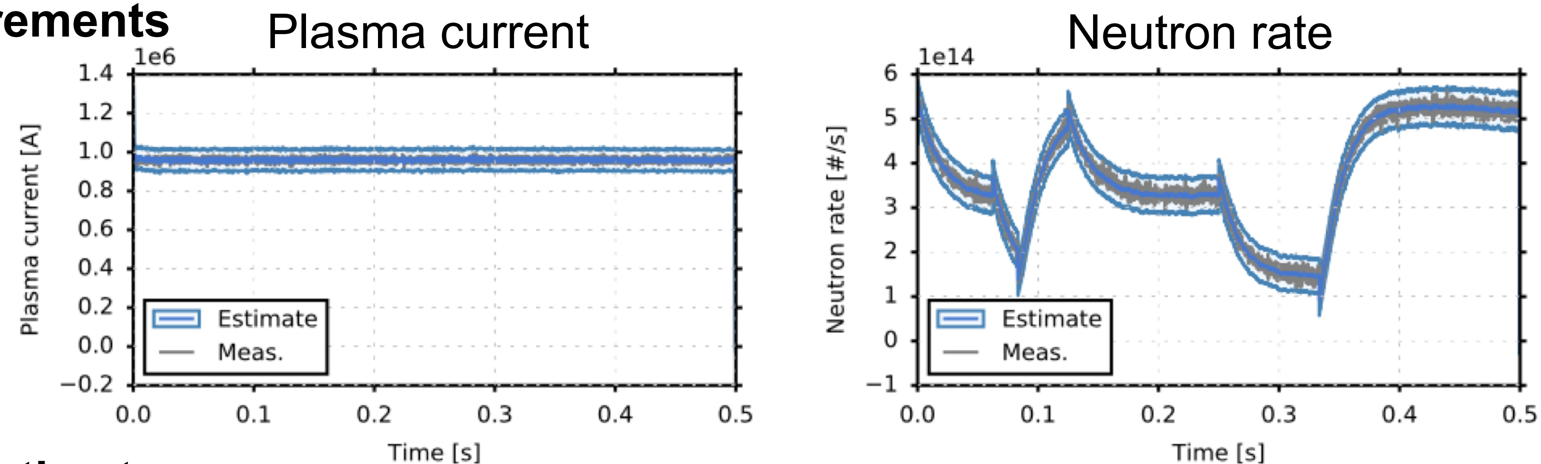
Assumptions made for initial simulations:

T_e, bootstrap current, and geometric parameters fixed and known

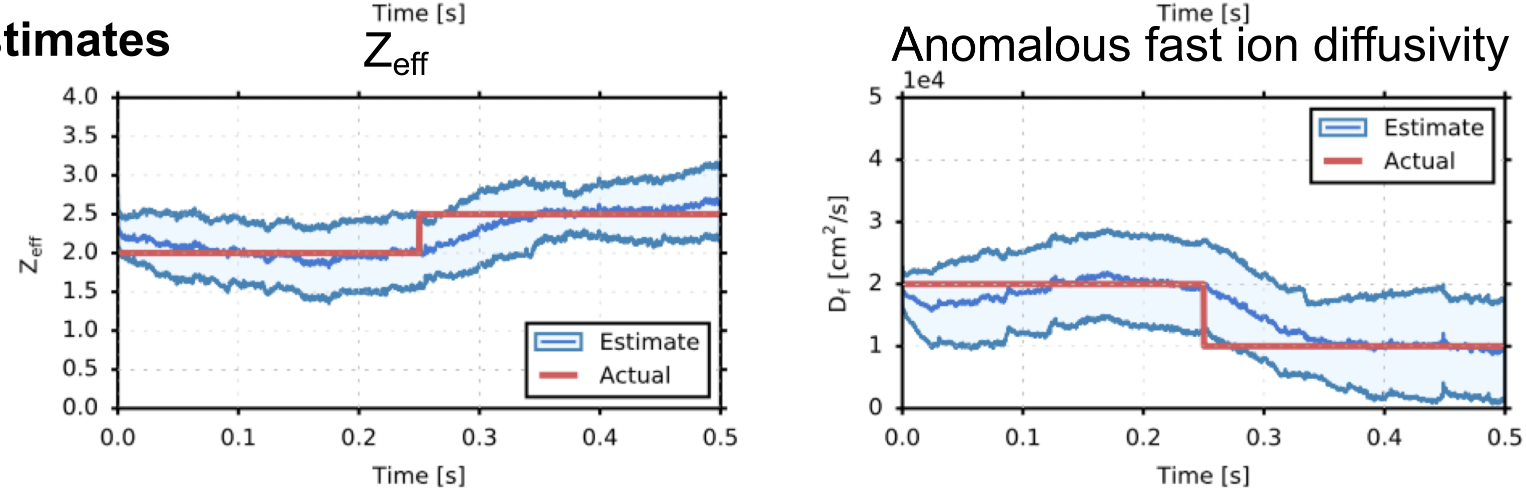
Spitzer resistivity

NubeamNet enables real-time current profile observer with estimation of Zeff and fast ion diffusivity

Measurements



State estimates

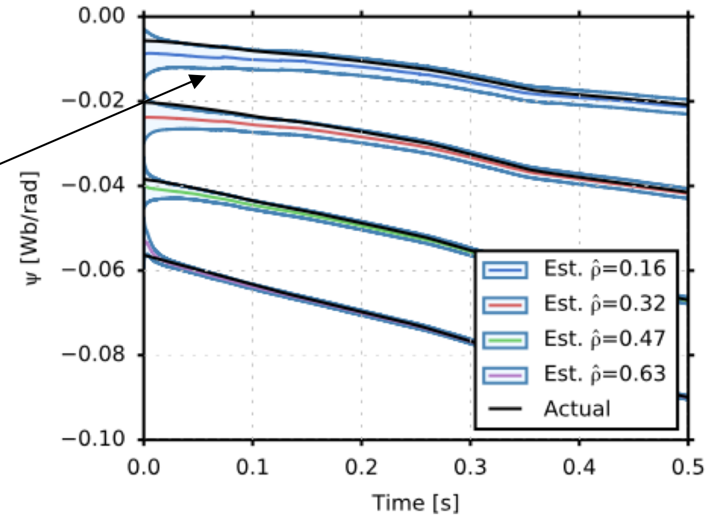


- Mismatch between predictions and measurements drives state estimate update
- NubeamNet enables **real-time calculation of sensitivity** of current drive and neutron rate to parameter changes

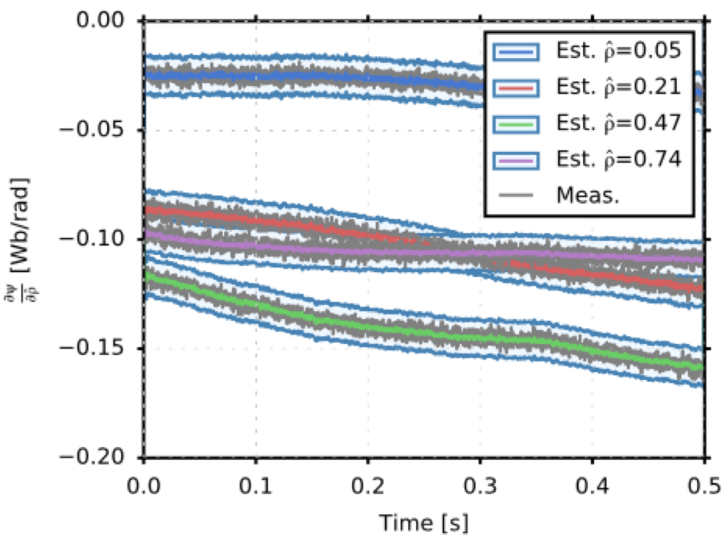
Poloidal flux profile converges to actual values, faster convergence if in-domain flux gradient measurements are included

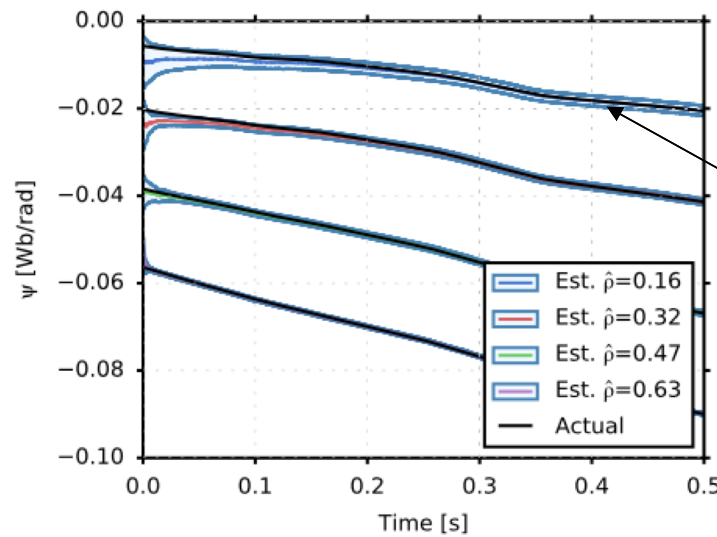
Poloidal flux profile at 4 locations
No in-domain measurements used by observer

Slow convergence in core (far from measurements, low resistivty)



4 in-domain flux gradient measurements used by observer





Faster convergence with in-domain measurements

NUBEAM is just one part of an integrated predictive model, but progress is being made on other modules

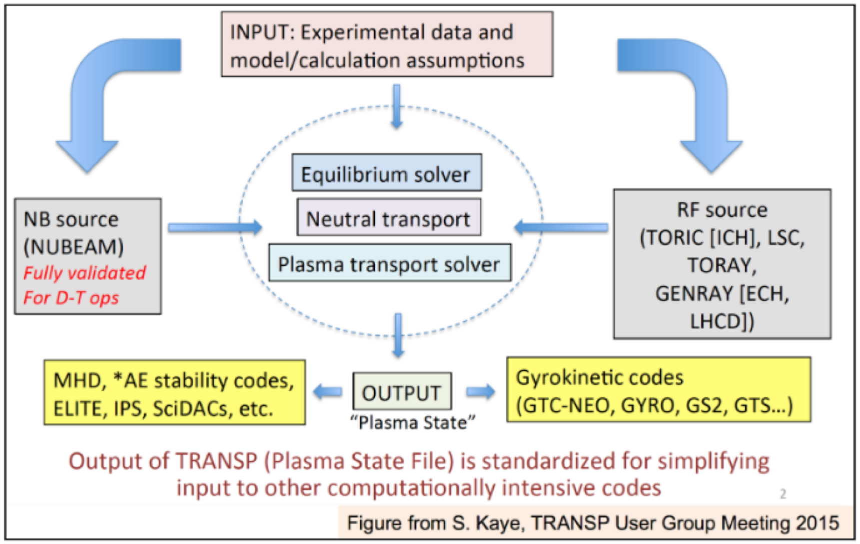
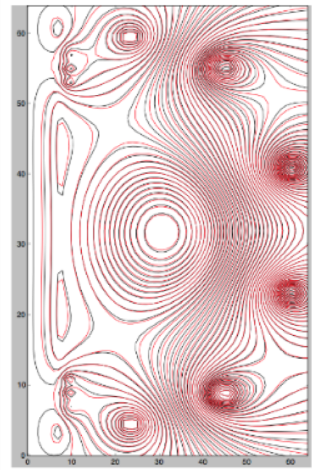


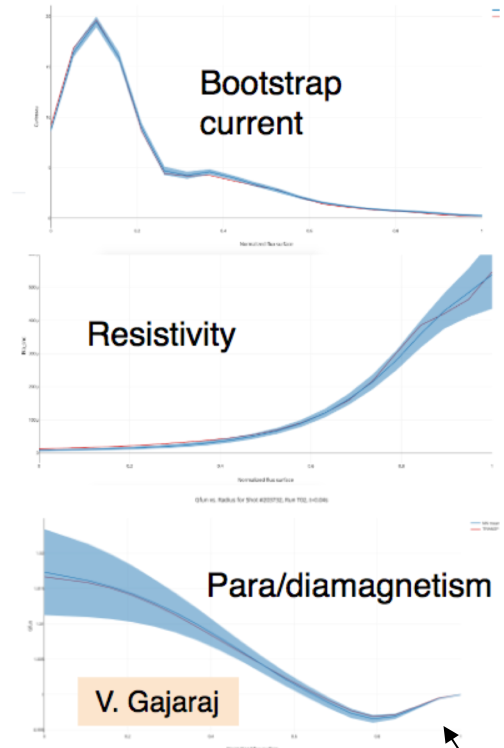
Figure from S. Kaye, TRANSP User Group Meeting 2015

EFIT01

Neural network



D. Boyer

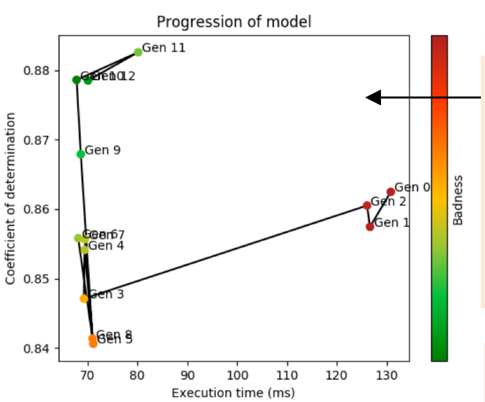


Transport fluxes

Meneghini NF 2017, 2014 (TGLF), Citrin NF 2015 (QuaLiKiz)

Boundary conditions

Meneghini NF 2017 (EPED)



Genetic algorithms being used to choose hyperparameters that optimize trade-off of accuracy and execution time

J. Kunimune

Assumptions made in observer in previous section can be removed once these models are included

Future work

- Compare results to recurrent, convolutional neural networks
  - May require more data, more computationally intensive
- Implement multi-threaded algorithm for evaluating ensemble and generating gradients in real-time
- Implement in TRANSP for routine use of AFID/Zeff fitting option
- Implement beam deposition optimization algorithm for between shots and real-time use
- Expand to other machines
  - Developing models for DIII-D and KSTAR