AI-based prediction and control of tokamaks: Combining simulations and experimental data

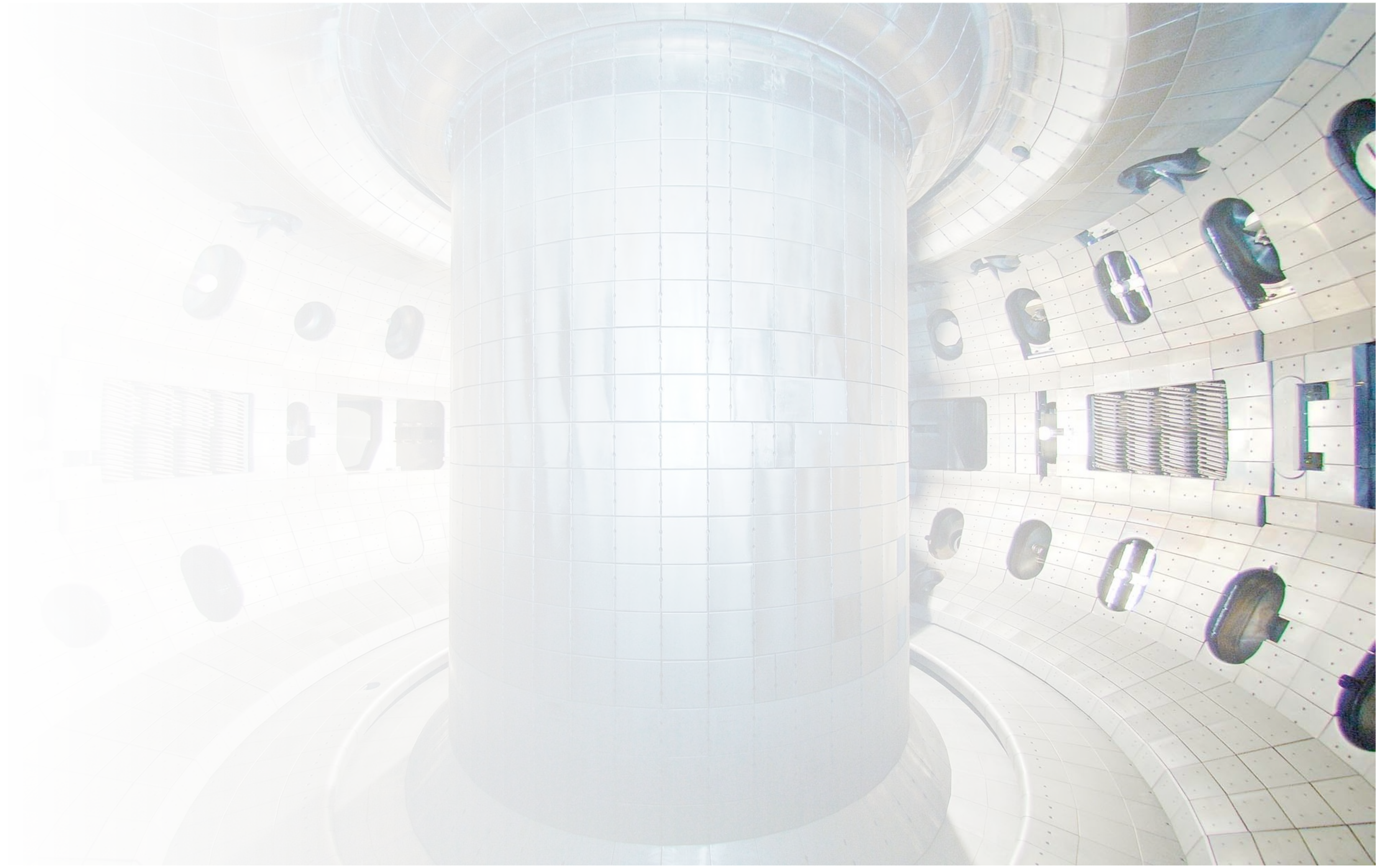
Joe Abbate[1,2]

with R. Conlin[2], I. Char[3]
G. Tardini[4], E. Fable[4], A. Rothstein[2],
K. Erickson[1], R. Shousha[1], H. Farre[1],
A. Pankin[1], V. Mehta[3], B. Grierson[5],
A. Jalalvand[2], E. Kolemen[1,2]

[1]PPPL,
[2]Princeton University,
[3]Carnegie Mellon,
[4]Max-Planck-Institut,
[5]General Atomics

Dissertation Defense

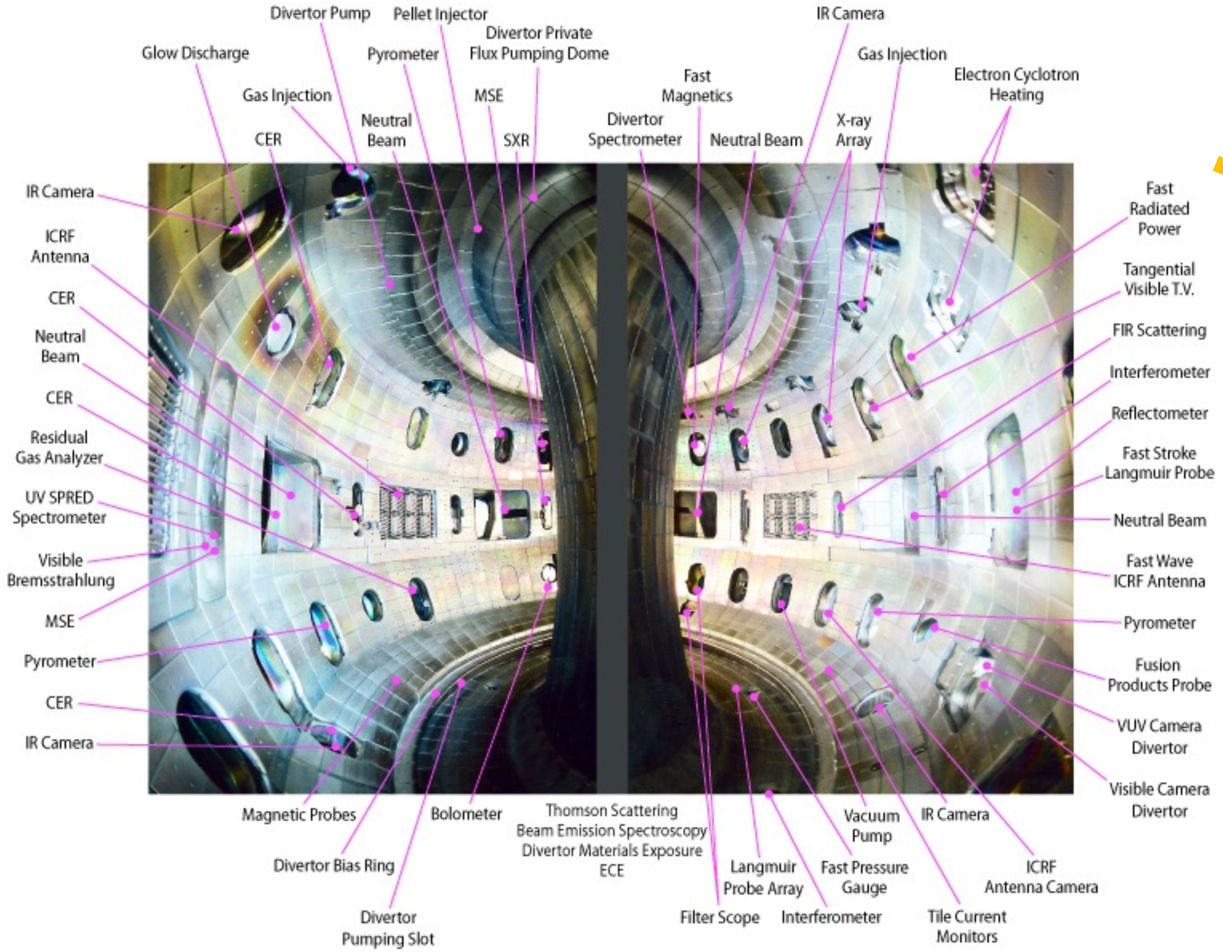
March 2024

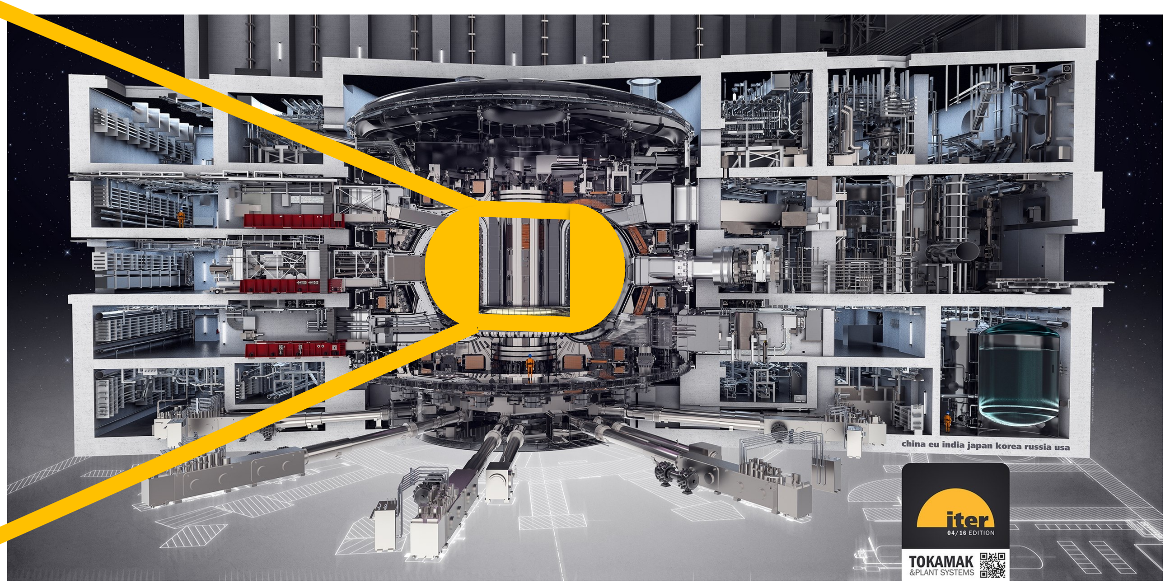


Building and "driving" a tokamak reactor is wildly complex

- **$B to build, $M/day to operate**
- **~hundreds of control knobs, ~thousands of diagnostics (comparable to aircraft)**

Humphreys 2015, PoP 22 021806



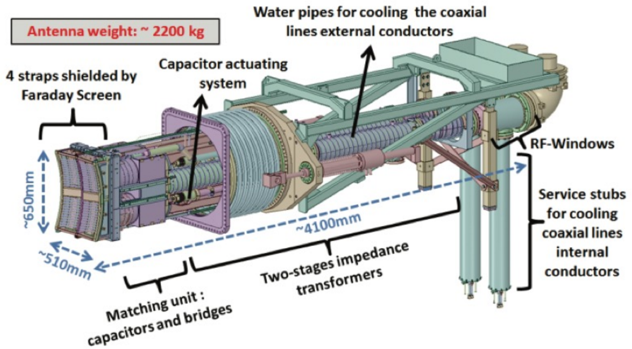


Plasma Control

Tokamak actuators (the knobs operators turn)

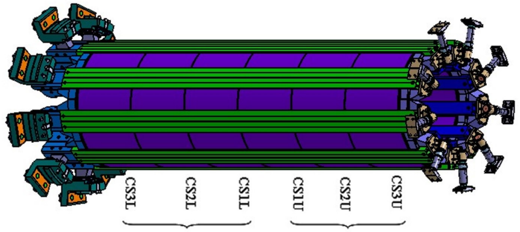
Power: 1kW → 1,000kW

1. Add microwaves

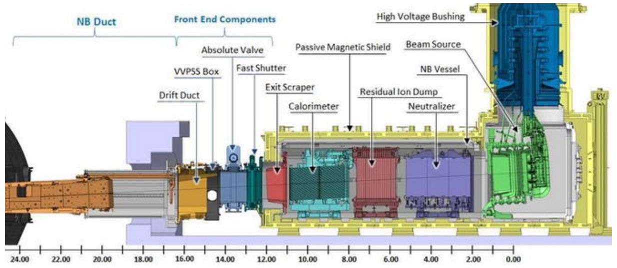


Electron Cyclotron Heating (ECH)

2. Apply electrical current



Ohmic transformer



Neutral Beams

3. Add hot particles

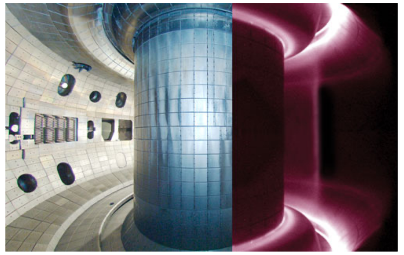
Reproducing and improving a discharge by trial-and-error

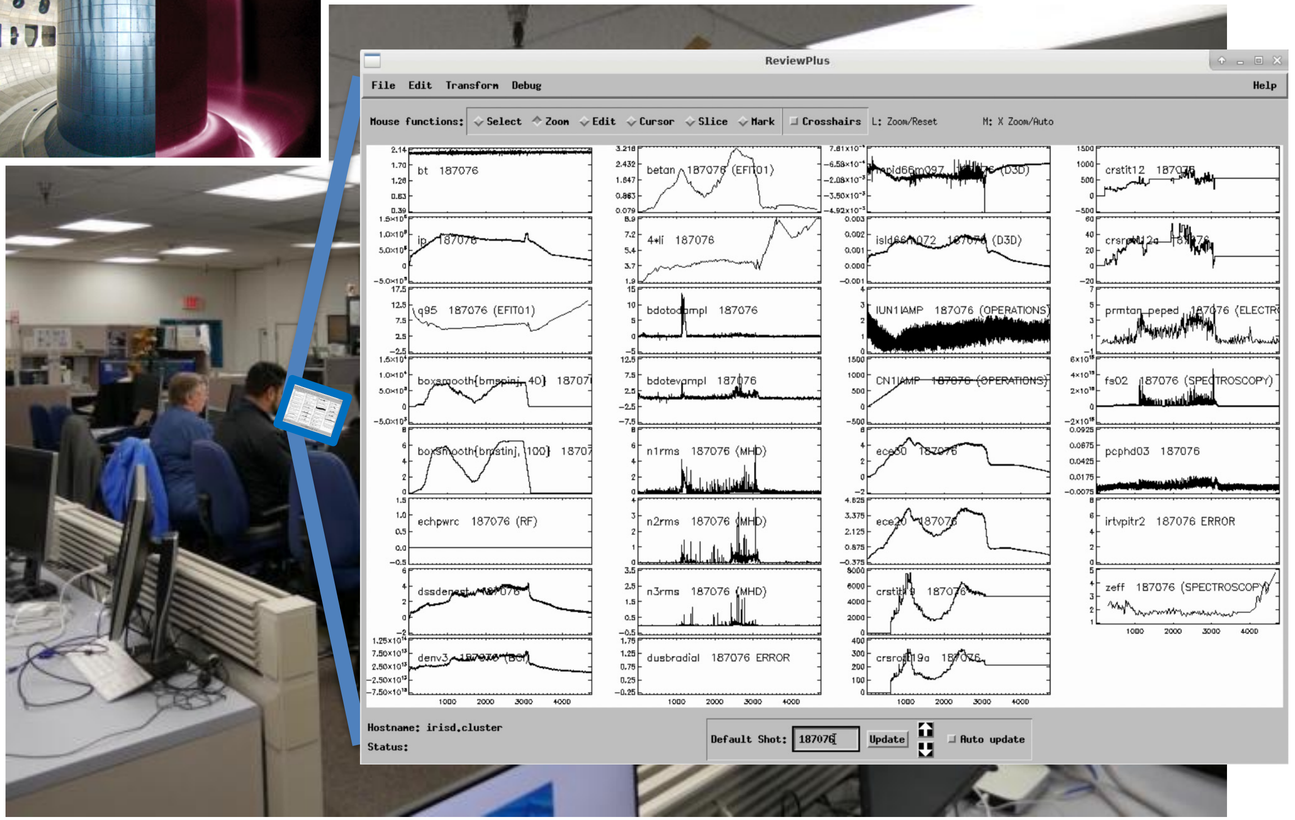
Experiment goal: reproduce and improve an experiment from 10 years before

"Many shots had MHD modes at 3 s… to try to improve that we changed Electron Cyclotron Heating deposition (180639-180642), and go to lower (180643-180646) and higher (180647) plasma current… none of which were successful.

We also tried lowering the voltage on the off-axis beams (180645) to get rid of the bursty modes and moving the BetaN ramp earlier (180646.)"

Ultimately, got "good reproduction of 133103, but no significant improvement"





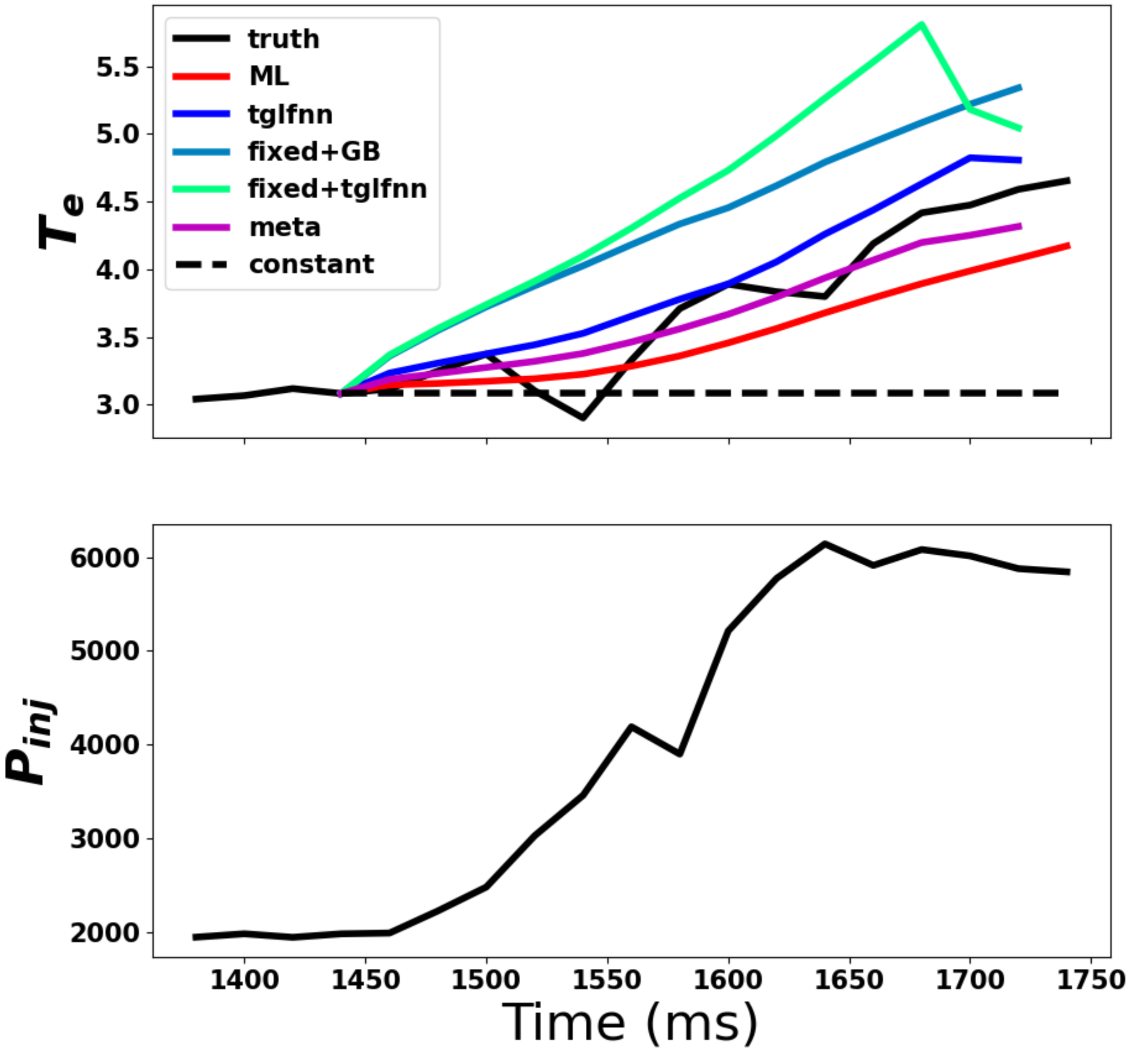
DIII-D Tokamak Control Room

Nik Logan's "overview" scope

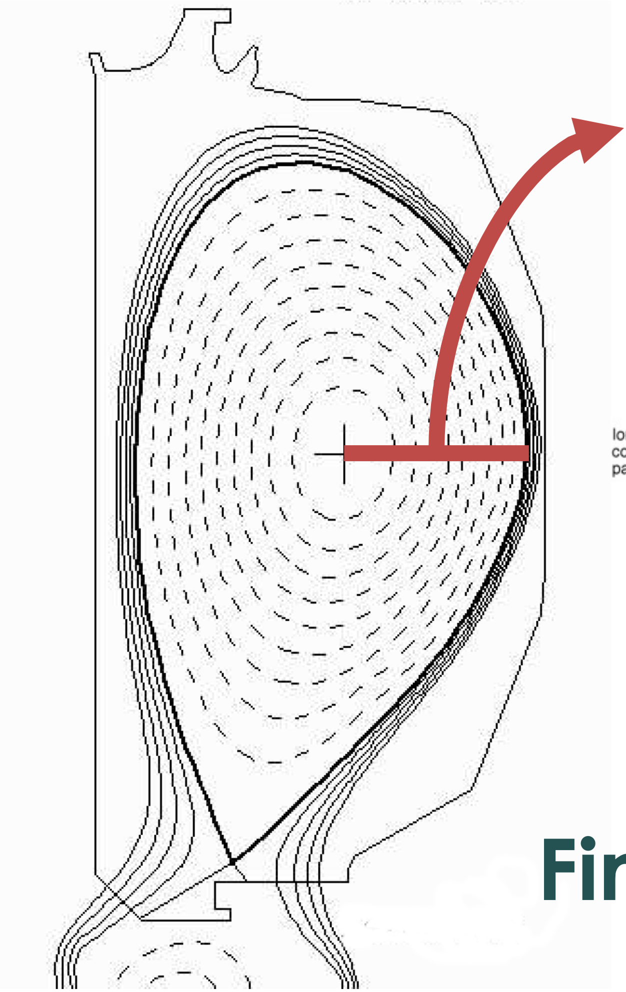
Human operators combine simulations, heuristics, and experience to achieve desired state by trial-and-error

Combining simulations and experimental data for tokamak dynamics

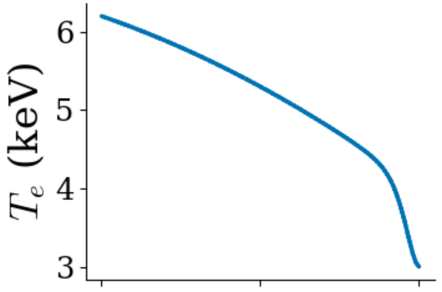
- **Data-driven tokamak dynamics + control**
- **Validating analogous physics simulations**
- **Combining experimental data + simulations**



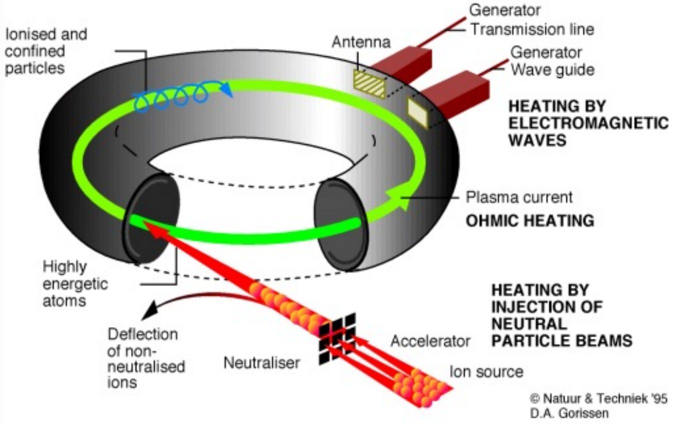
Kinetic plasma profile dynamics model



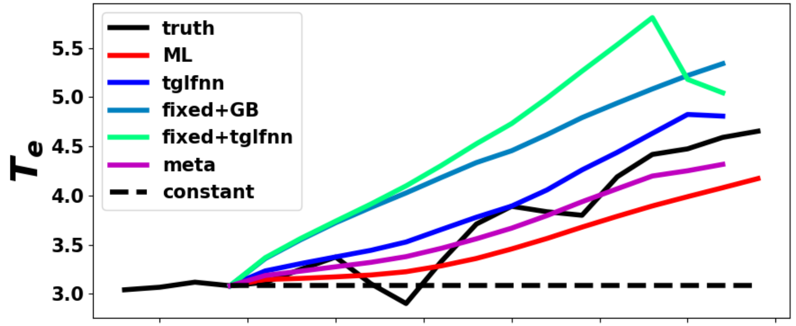
x = state
(1D profiles+shape)

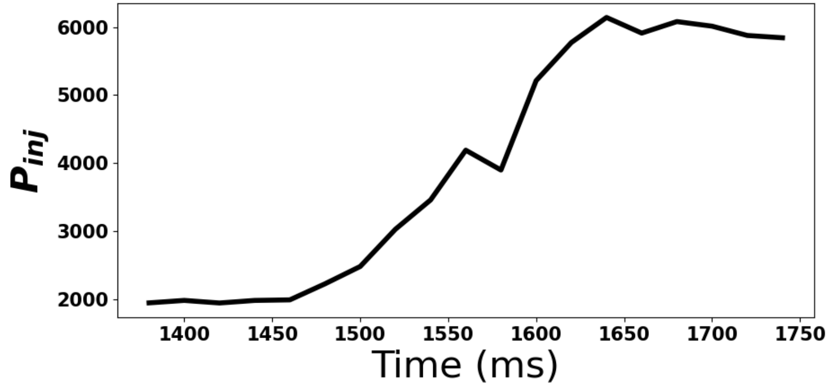


u = actuators



f = mapping

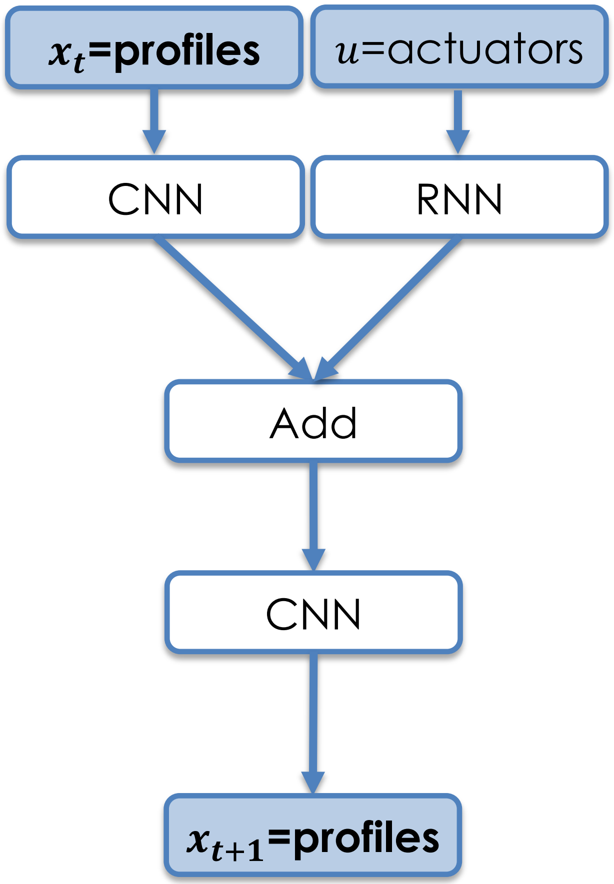


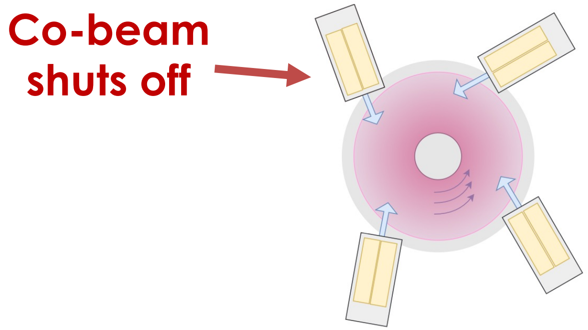


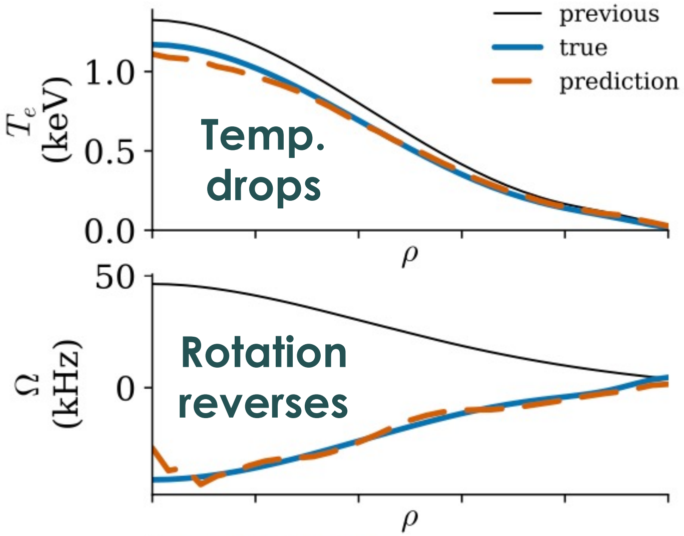
Find mapping f s.t. $x_{t+1} = f(x_t, u_t, u_{t+1})$

Built data-driven (neural net) profile predictor

Find mapping f s.t. $x_{t+1} = f(x_t, u_t, u_{t+1})$







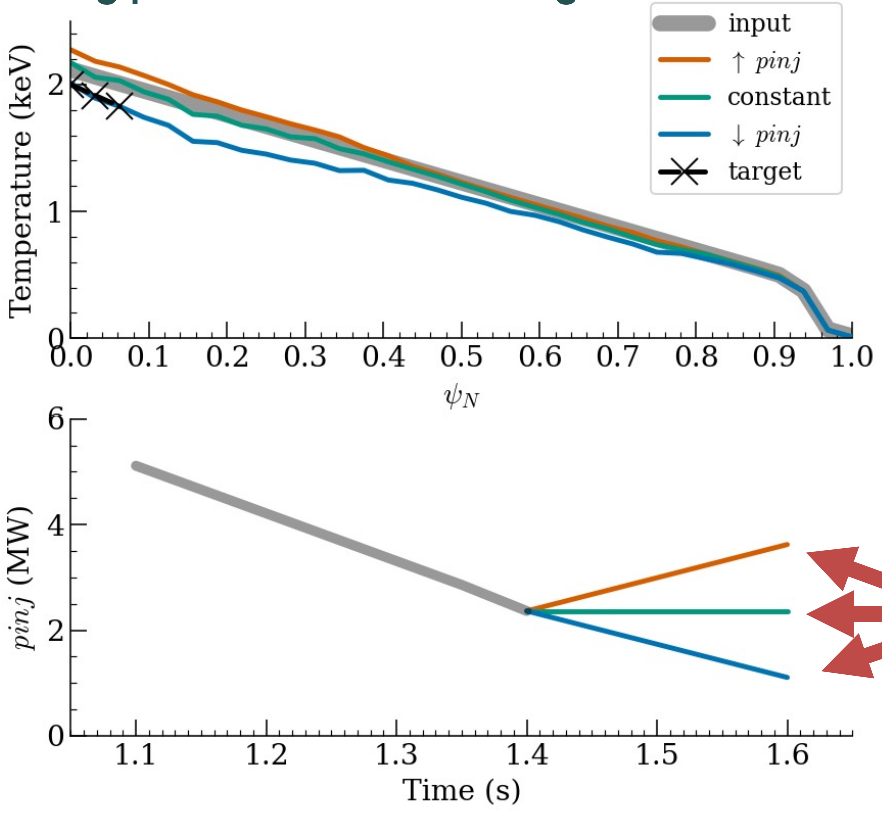
Abbate, Conlin Nuclear Fusion 2021

Profiles (x)	Source
Electron density	Thomson
Electron Temperature	Thomson
Ion temperature	CER
Ion velocity	CER
q	EFIT01
Impurity density	CER

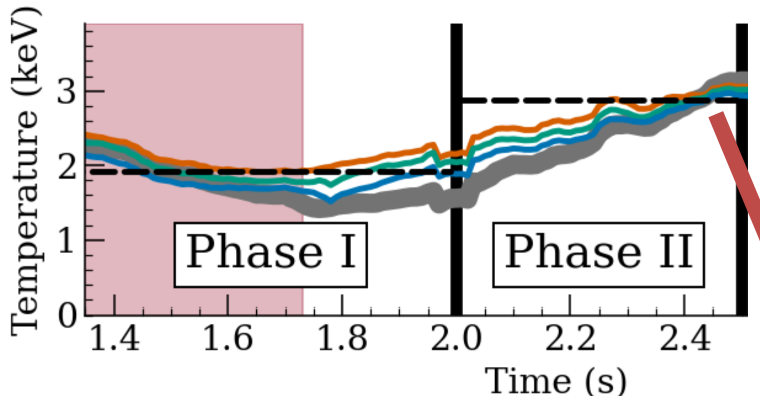
Actuators (u)
NB Power
NB Torque
Target I_p
Gas flow rates
[Shape parameters]
[ECH]

Neural network profile control in PCS (starting w/ finite set MPC)

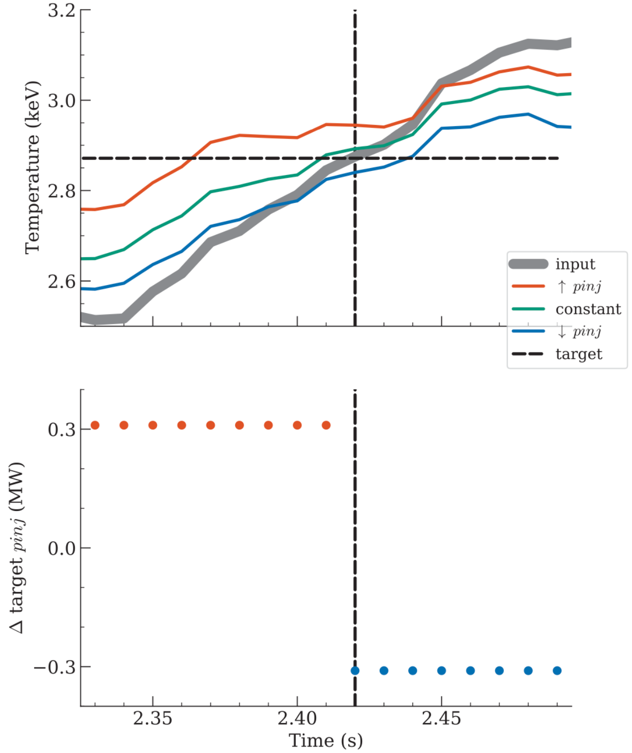
Controller chooses control option yielding prediction closest to target



Finite set of proposed control options



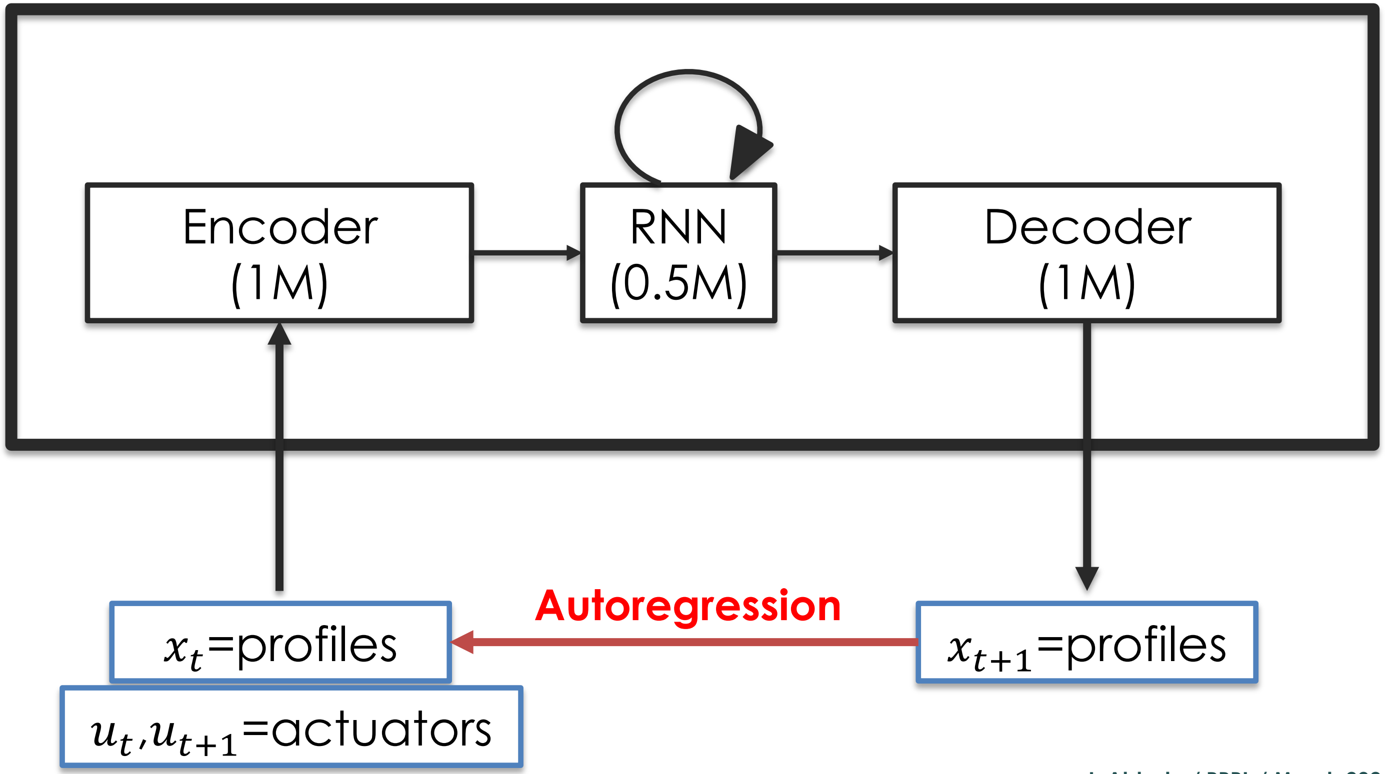
MPC yields derivative-like (lookahead) control without tuning

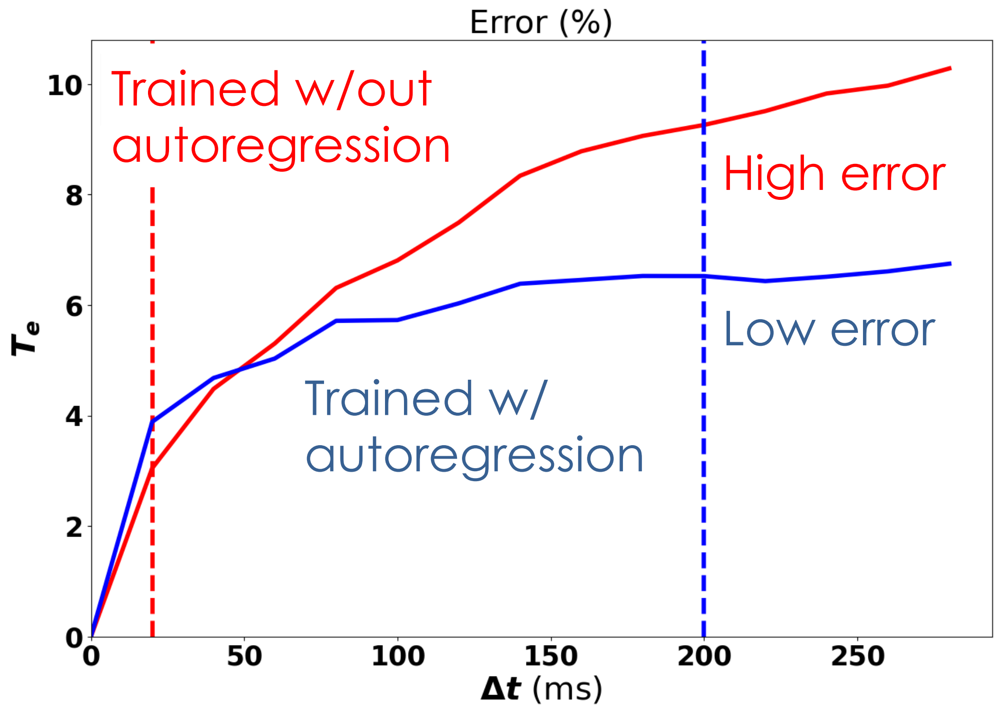


Abbate, Conlin Journal of Plasma Physics 2023

New architecture and training methodology to predict autoregressively

- **New architecture allows continuous predictions any step in future (Char, CMU), via autoregression**
- **Additionally training model with autoregression allows tuning model for any time horizon**



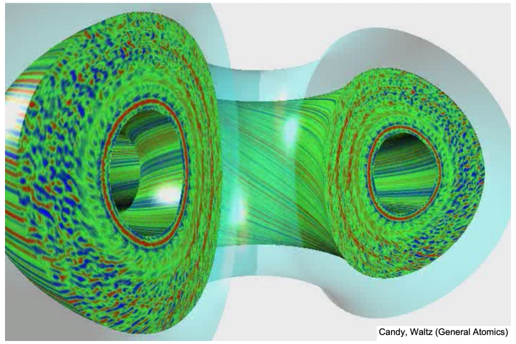


Find mapping f s.t.

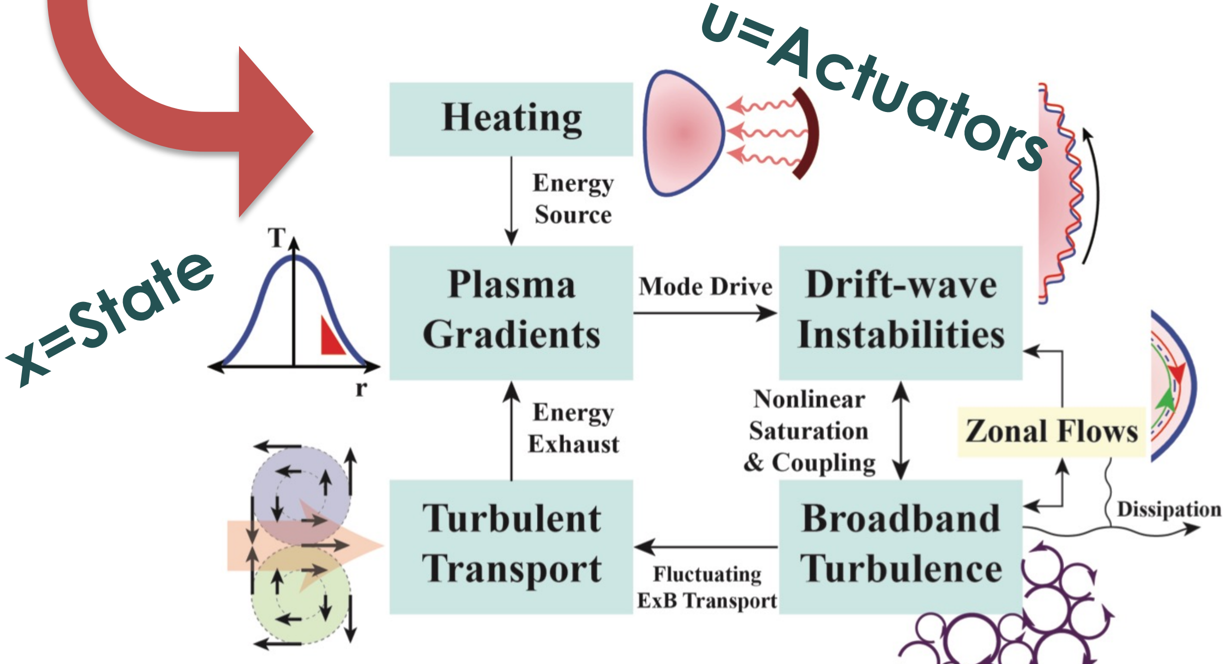
$$x_{t+1} = f(x_t, u_t, u_{t+1})$$

Our approximate plasma state: 1D profiles

- **Goal: track ~10^{23} particles through 6D phase space w/ nonlocal interactions**
  - **Massive progress made (over ~70yrs + ~1,000 careers)**
  - **Find approximations that fit data, try to avoid overfitting**
- **Simplified plasma state given by 1D profiles → "integrated modeling"**



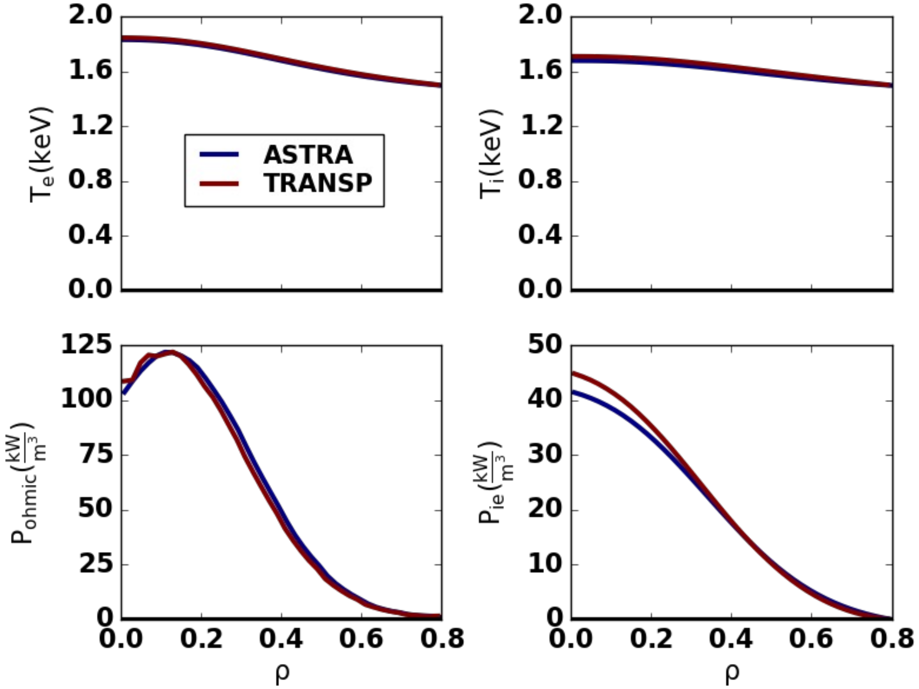
(From Jeff Candy via Greg Hammett)



(From P. Rodriguez-Fernandez MIT thesis 2019)

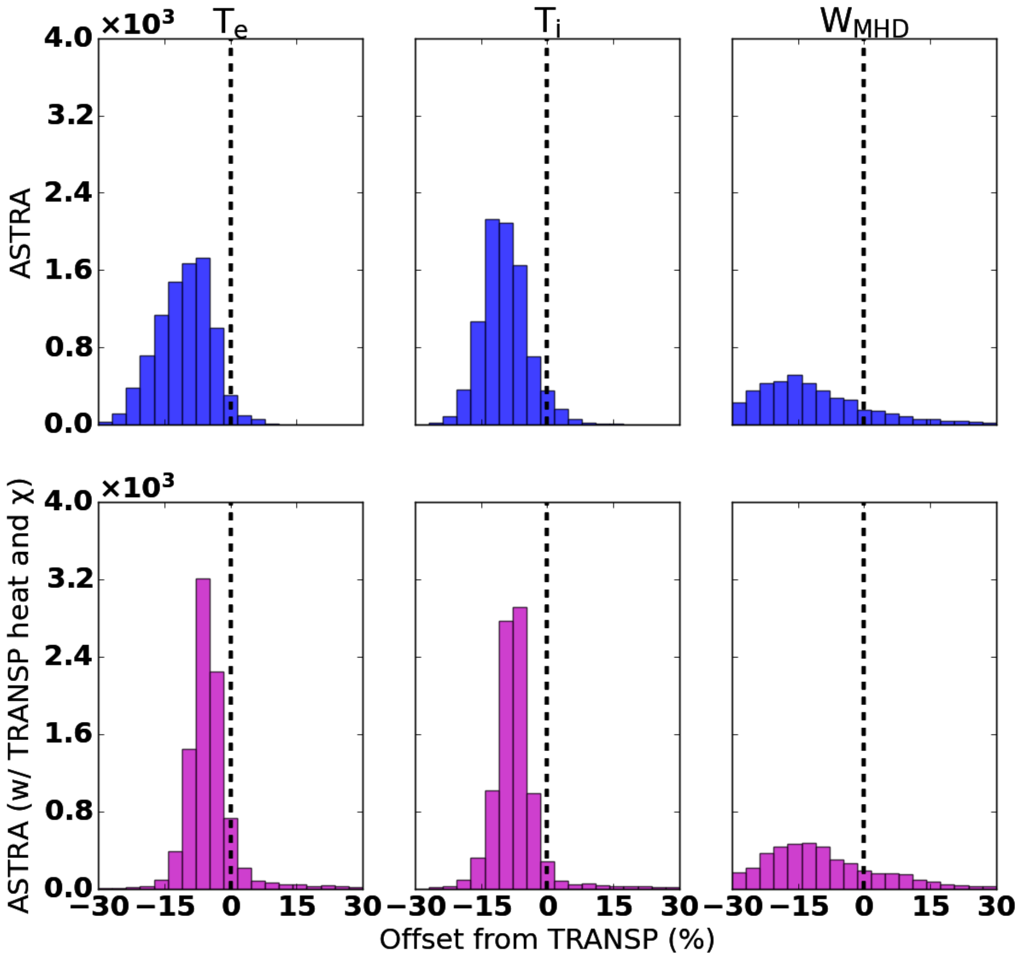
Integrated modeling verification: TRANSP and ASTRA yield different answers for realistic discharges

- **Verify predictions of core Te/Ti and Wmhd**
- **Simple case: ASTRA and TRANSP yield same result**
- **Database comparison: ASTRA and TRANSP differ**
  - **seemingly the solver?**



(ITER Physics Expert Group on Confinement and Transport, 1999)

Bias $\longrightarrow$ $$f = \frac{< T_{prediction} - T_{truth} >}{< T_{truth} >_{RMS}}$$



Integrated modeling validation: simple baselines of comparison

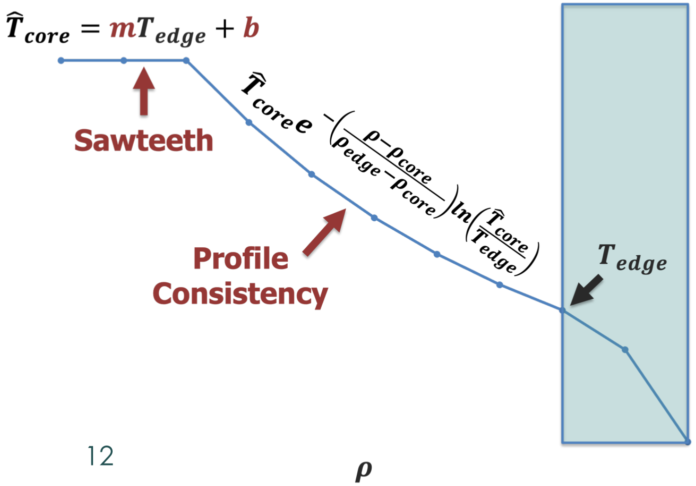
- **Validate predictions of core Te/Ti and Wmhd**
  - **Use multiple independent transport solvers (TRANSP + ASTRA)**
  - **Run on ~hundreds of cases automatically**
- **Compare to empirical baselines:**

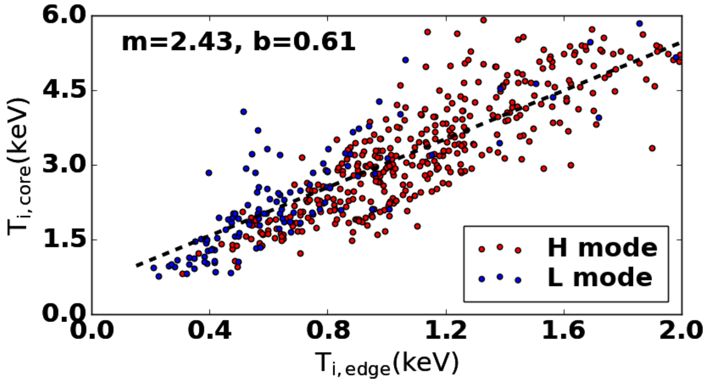
Bias
$$f = \frac{< T_{prediction} - T_{truth} >}{< T_{truth} >_{RMS}}$$

Error
$$\sigma = \frac{< |T_{prediction} - T_{truth}| >_{RMS}}{< T_{truth} >_{RMS}}$$

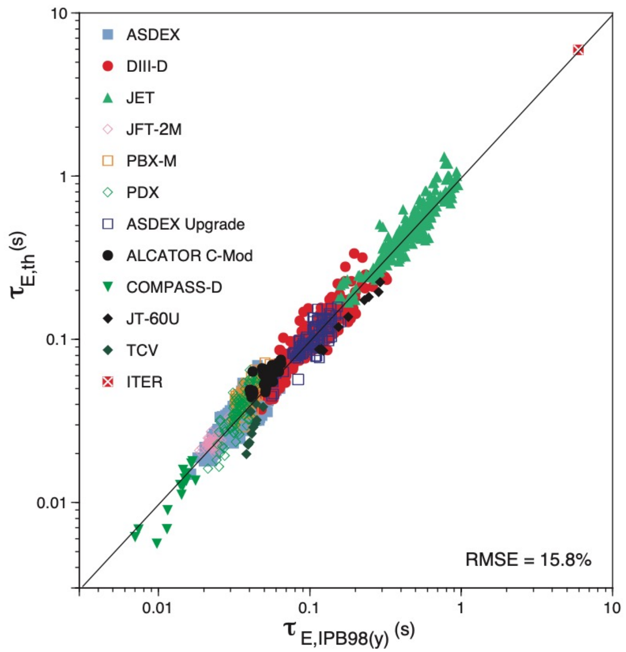
(ITER Physics Expert Group on Confinement and Transport, 1999)

Te/Ti baseline: "Profile consistency"



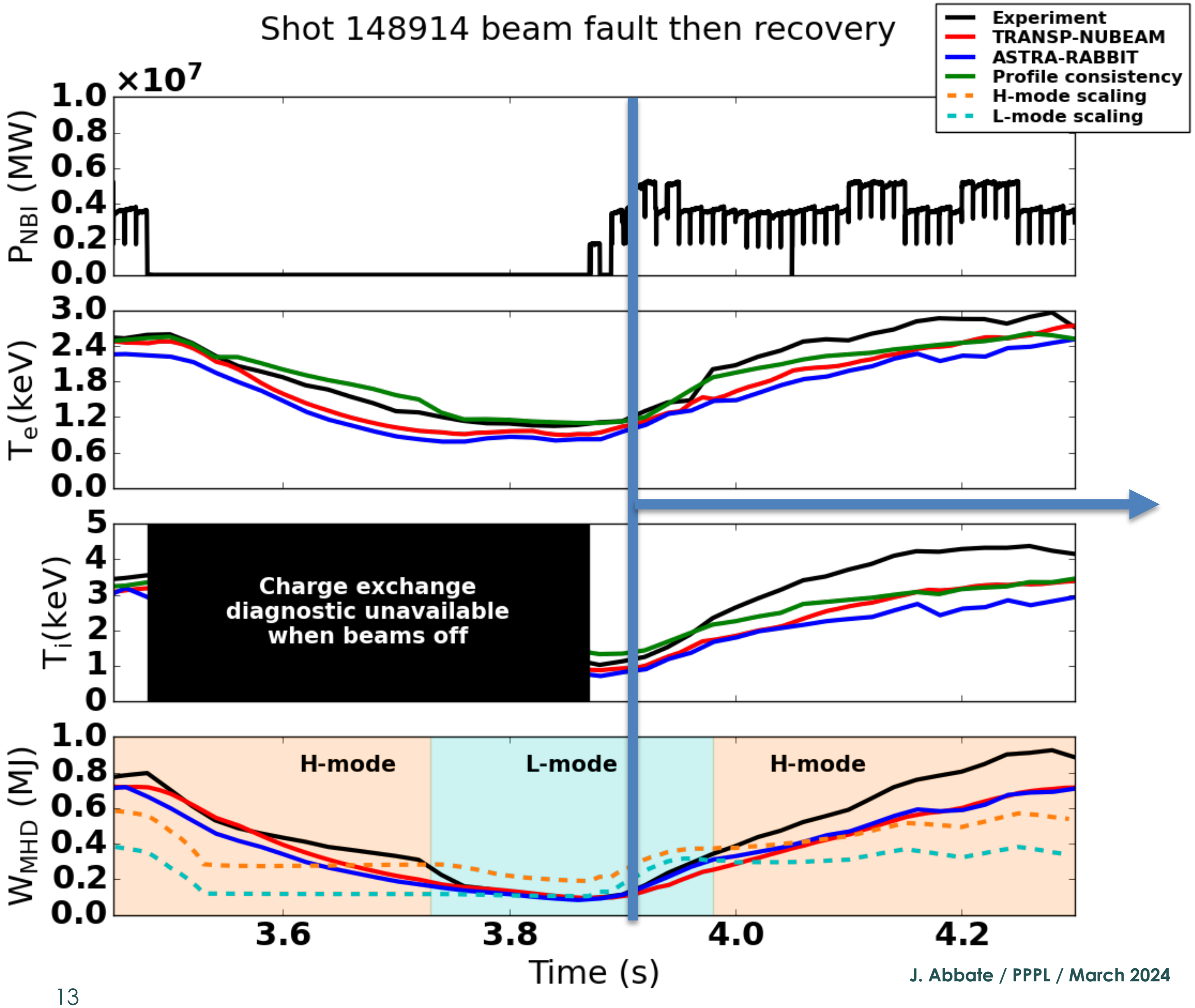


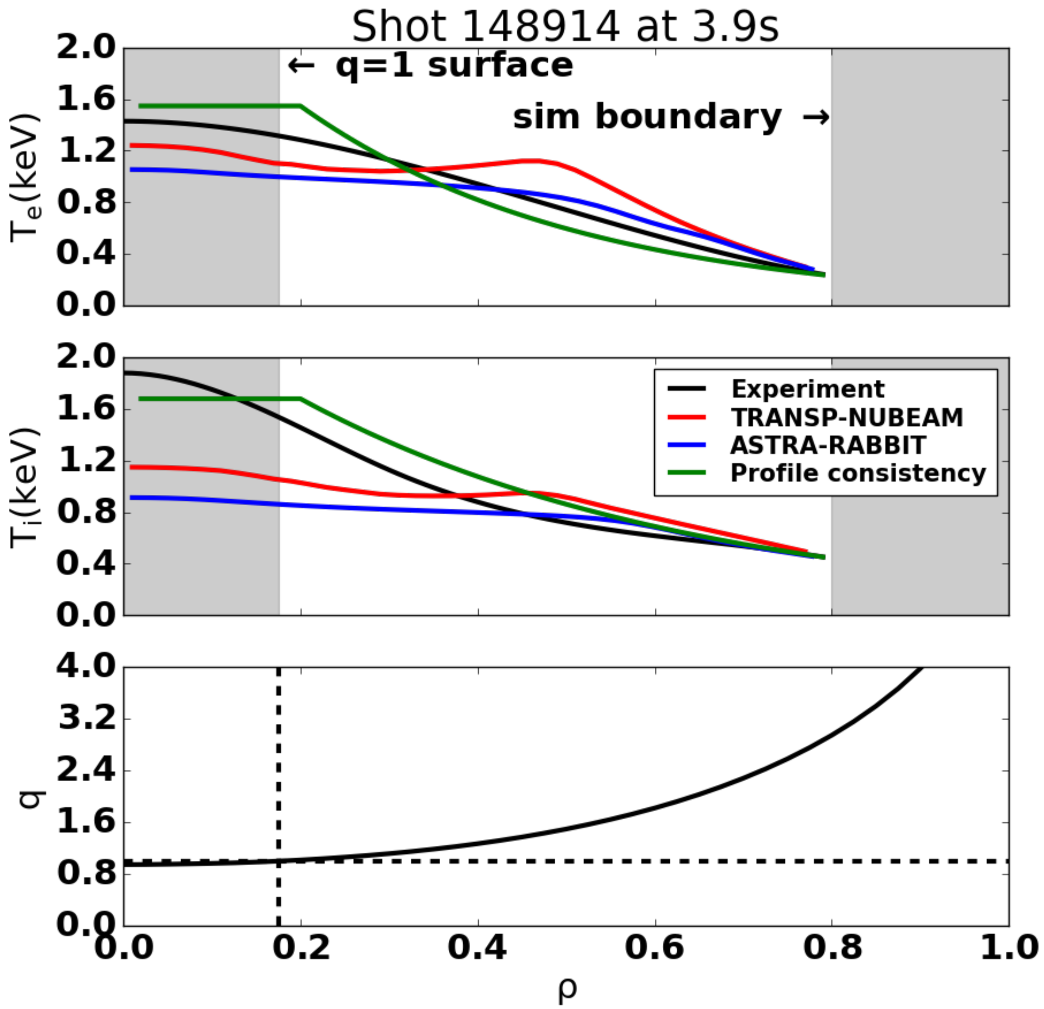
Wmhd baseline: "H scaling"



$$W_{MHD} = \tau_E P_{total}$$

Time-dependent predictions made over 900ms window

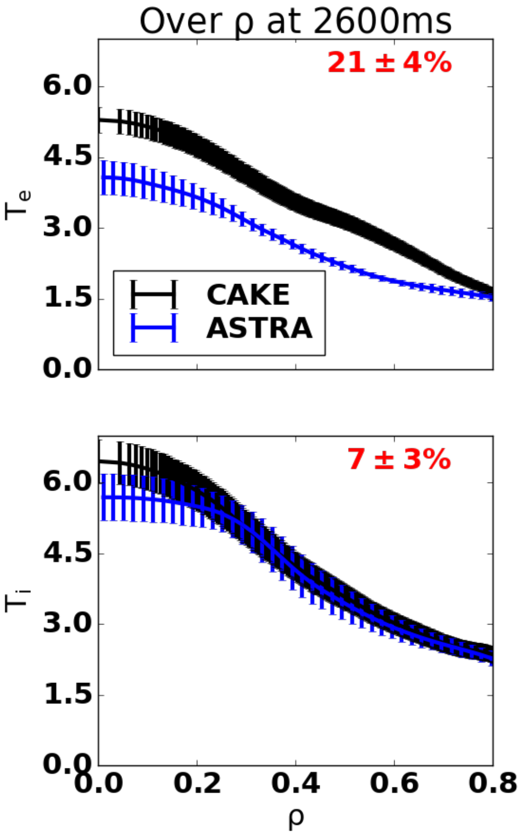




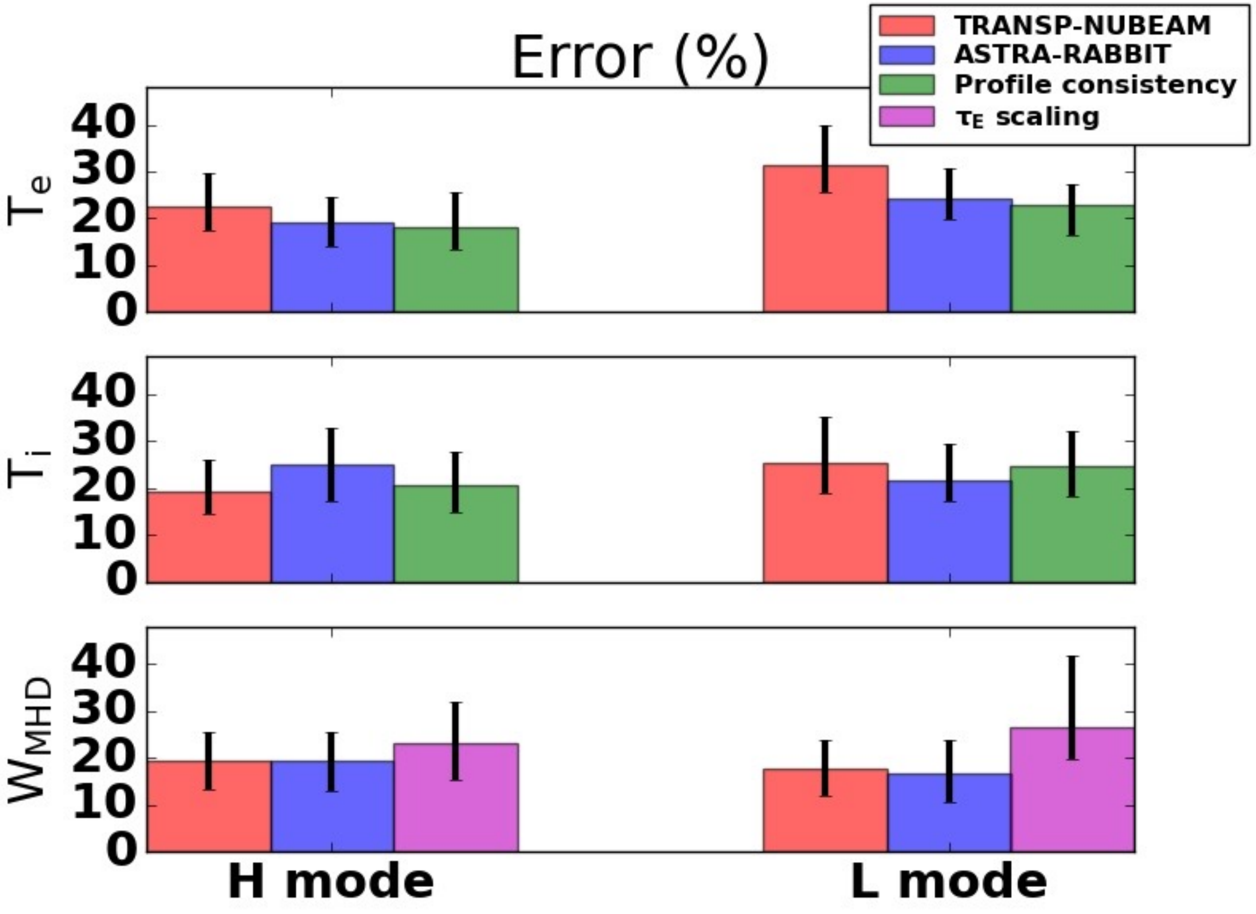
Plasma Control

ASTRA/TRANSP perform no better than empirical baseline, large database with relatively low diagnostic uncertainty propagation gives confidence

- **Hypothesis: ASTRA/TRANSP performs significantly better than baseline**
  - **$p<0.05$ for Wmhd but not Te/Ti**
- **Diagnostic uncertainty ≲5% for single shot; less for mean over database**



	Input	Output
T_e	6%	4%
T_i	9%	3%
n_e	6%	3%
Z_{eff}	5%	1%
Ω	9%	1%
q	10%	4%



Plasma Control

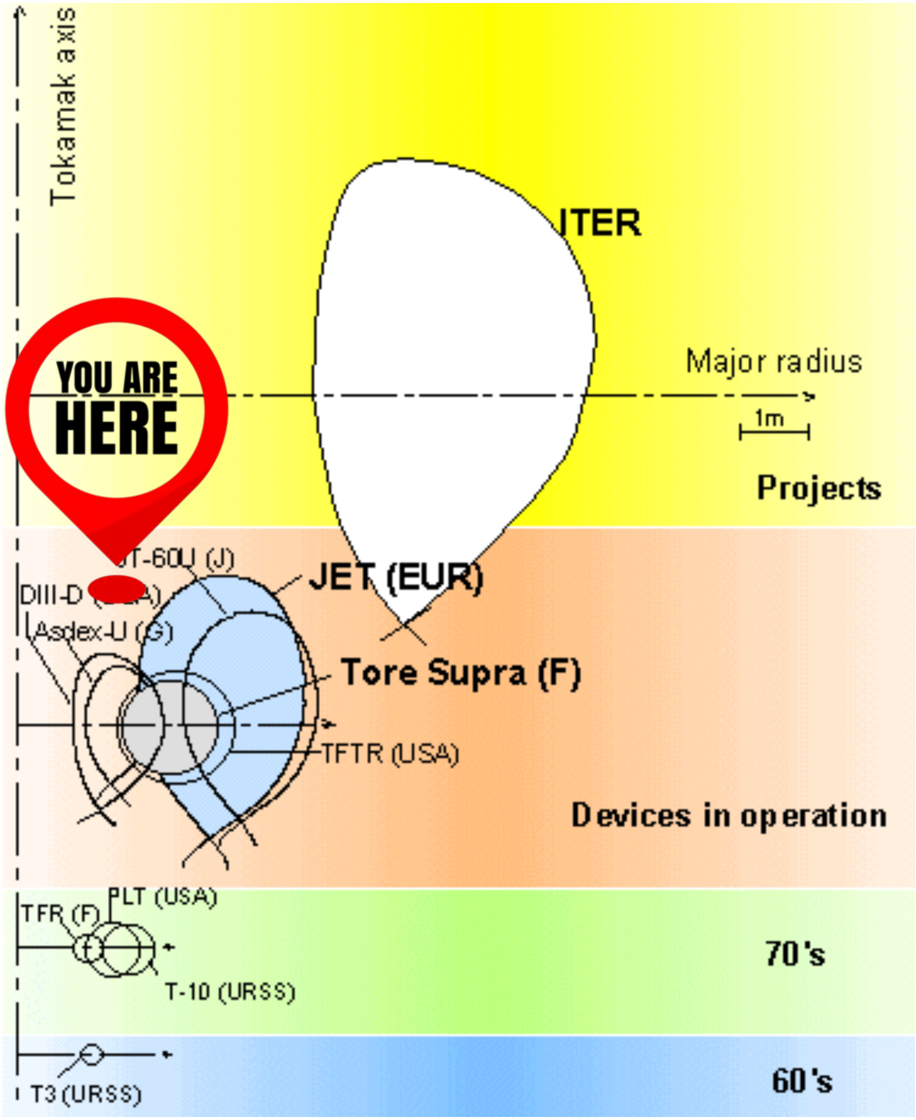
Training a model for the task of extrapolation: 4 methods for data+sim (one of which was already successful)

- **Reactor commissioning: use *all* information until current timestep to predict future evolution**
  - **All timesteps / discharges up to now**
  - **Simulations**
  - **Previous tokamaks**
- **Our emulation: Train on D3D I_p < 0.9 MA, predict on D3D 1.0 MA < I_p < 1.2 MA**
  - **Add AUG data**
  - **Add ASTRA simulation info**

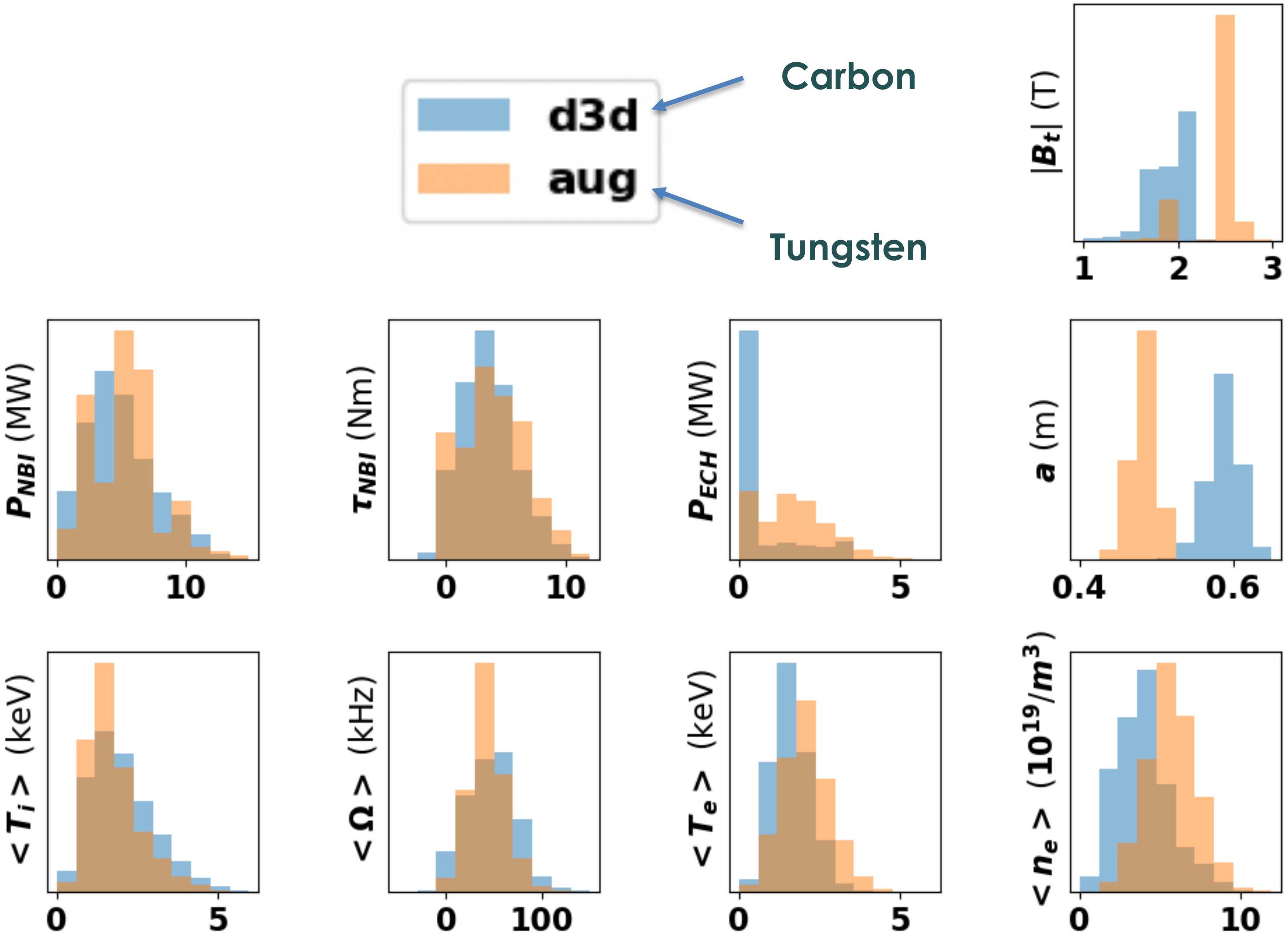
✗ 1. **<u>Add data from more machines</u> using normalization**
✗ 2. **<u>Concatenate</u> simulation context as additional input**
✗ 3. **<u>Transfer learn</u> by training on experimental data, tuning on simulation data**
✓ 4. **<u>Meta-learned model</u> taking output of data-driven and simulation *models***

Plasma Control

Add data from more machines: AUG data to enhance D3D predictions



The main tokamaks in the world,
French Atomic Energy Commission (CEA)



Hypothesize normalization aids cross-machine learning

- **Physics: nondimensionalizing Vlasov eq**
  - **BUT no quantitative accuracy in most cases (atomic physics, 3D fields, boundary effects…)**
- **Physics (simple):**

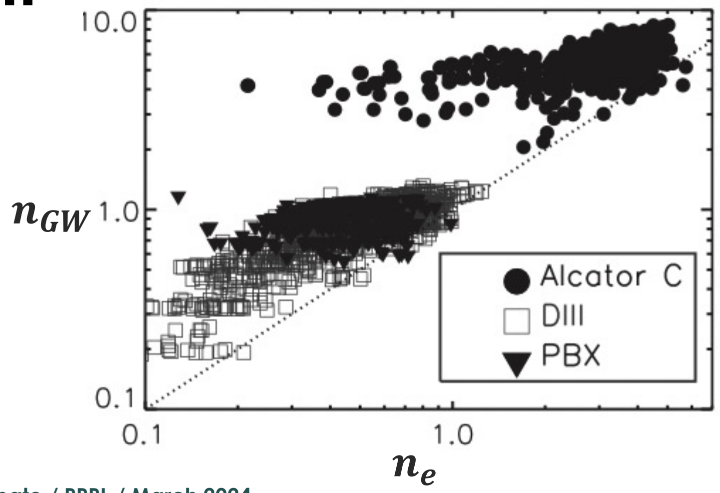
$$\Omega \rightarrow R^2 \Omega \int n_e dV$$

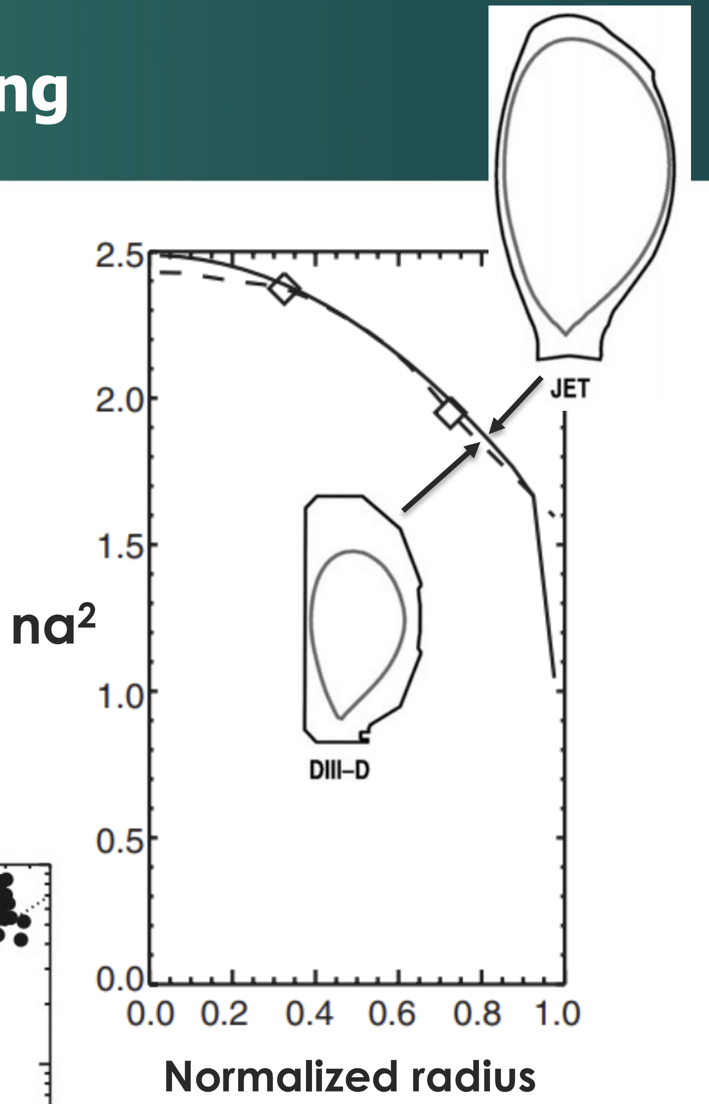
$$P \rightarrow \frac{P}{V}$$

- **Empirical: operators observe degradation as density approaches** $n_{GW} = \frac{I_p}{\pi a^2}$

$$n_e \rightarrow \frac{n_e}{n_{GW}}$$

M. Greenwald 2002 Plasma Phys. Controlled Fusion 44

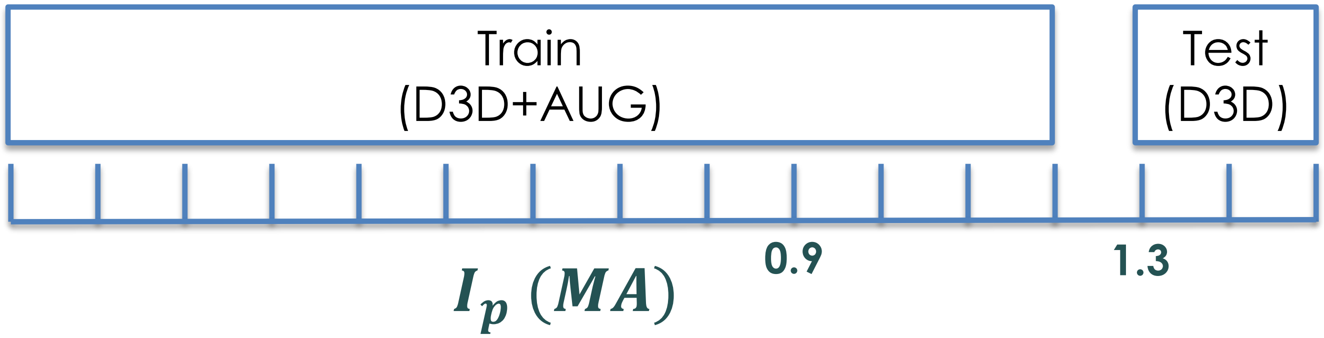


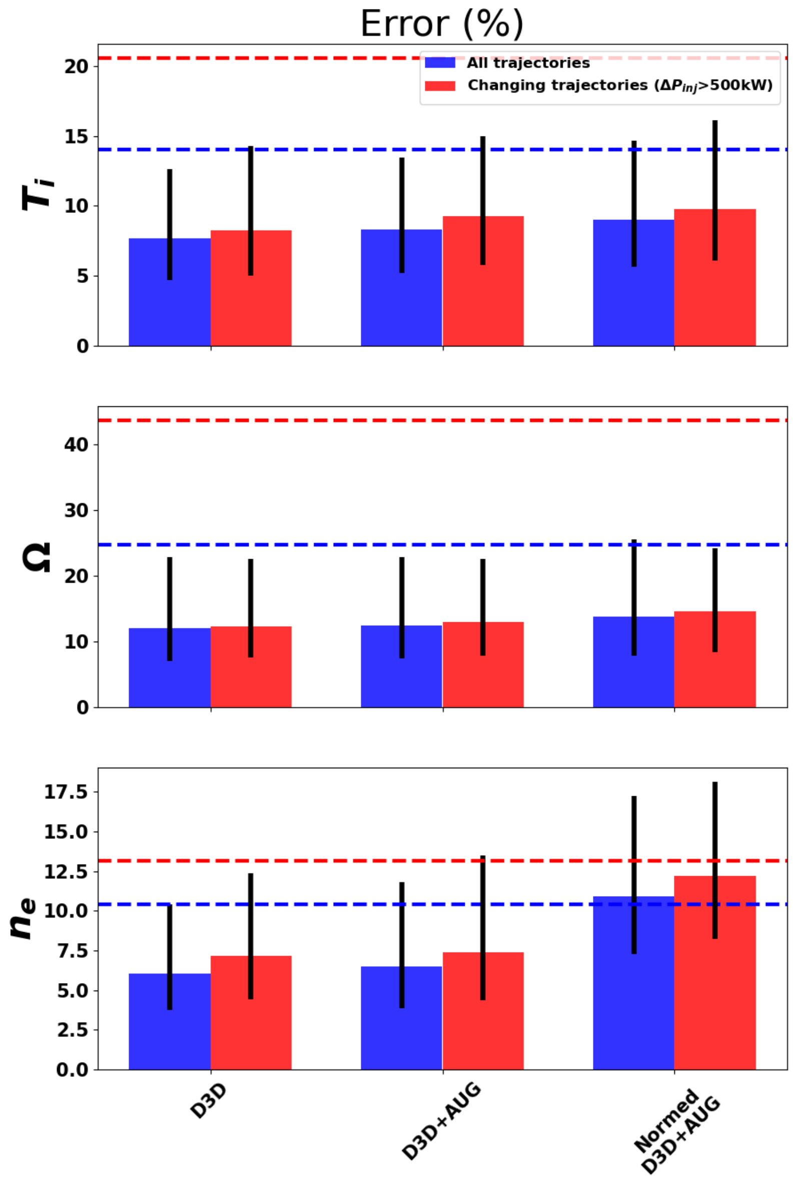


T.C. Luce et al 2002 Nucl. Fusion 42 1193

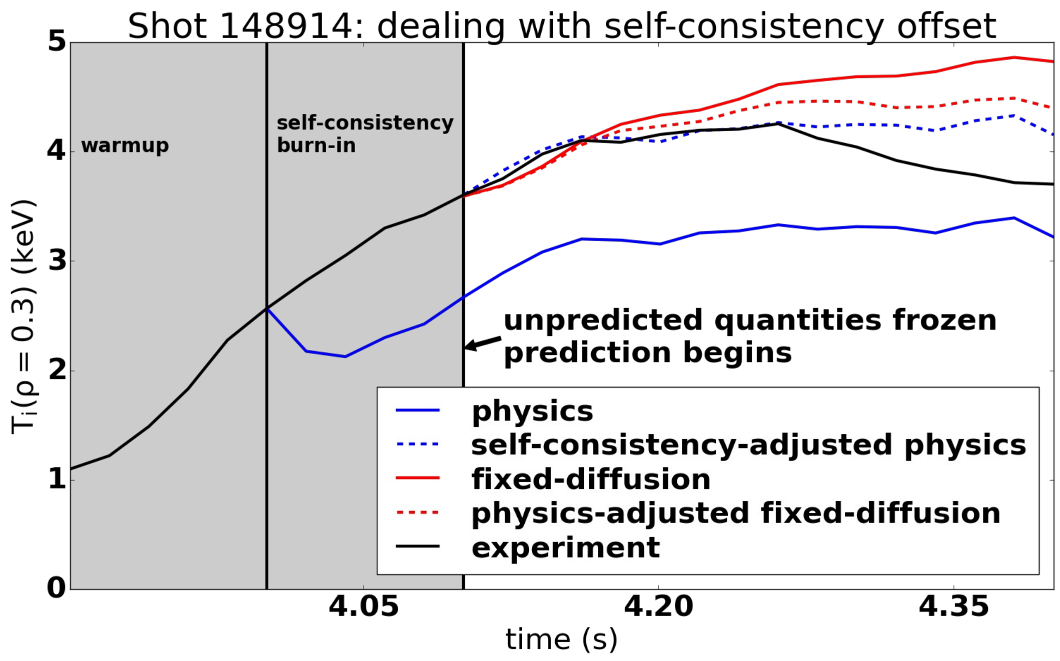
Add data from more machines: does not improve performance

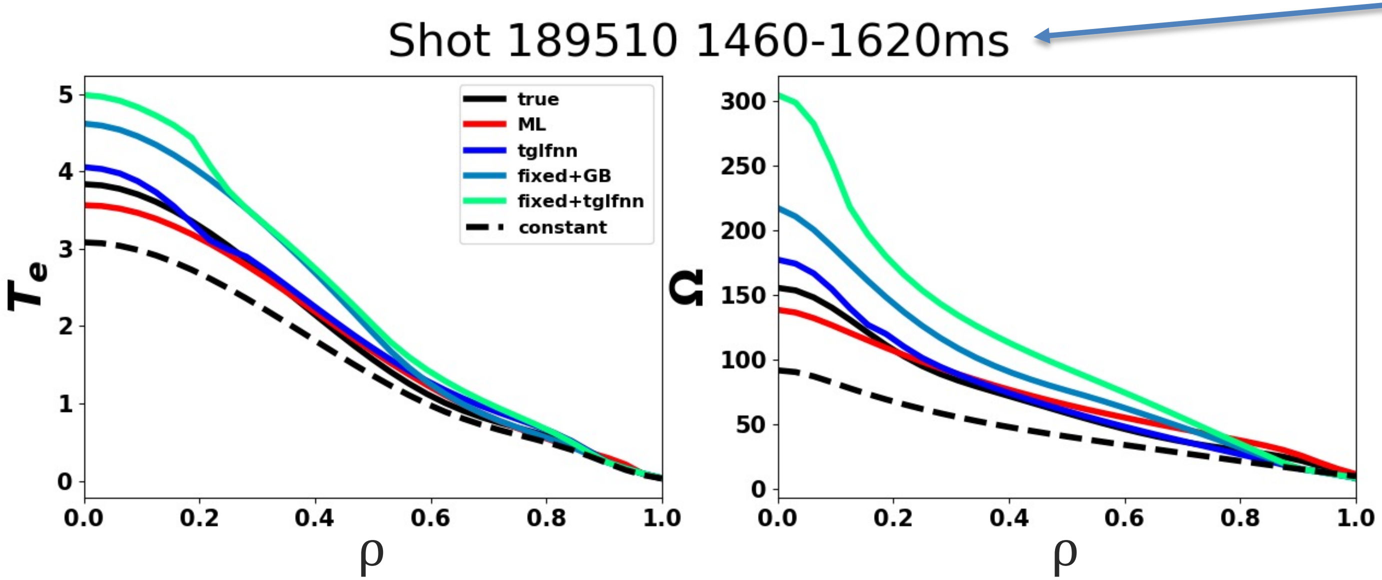
- **Consider ITER commissioning: using data up to now, predict next shot**
  - **No improvement from including AUG data**
- **"Constant prediction" for reference**
- **Time-dependent vs steady-state trajectories**
  - **Based on $\Delta \mathrm{P}_{\mathrm{inj}} > 500\mathrm{kW}$**
- **UPSHOT: don't skimp on ITER pre-operation phase**
  - **Maybe more machines helps learn own normalization?**

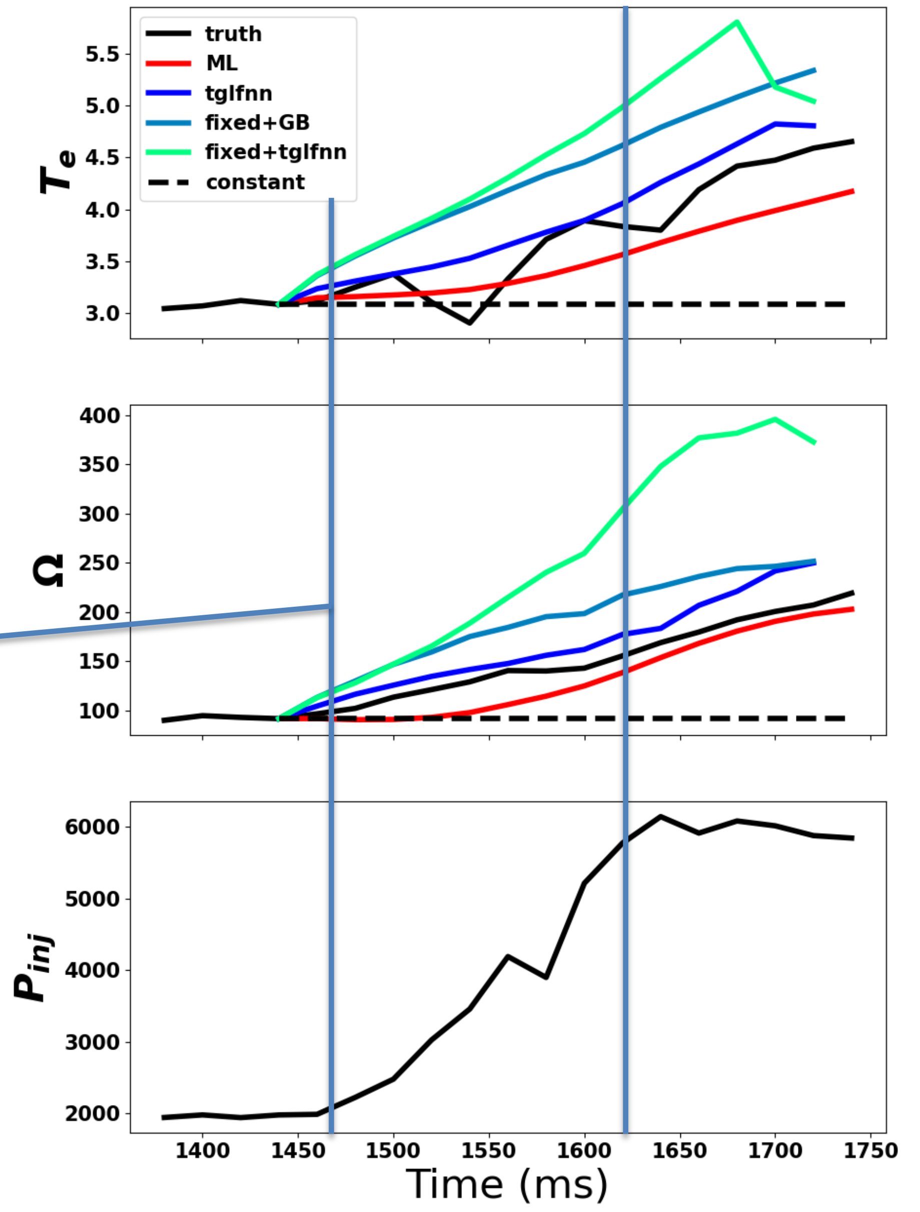




The ensemble of simulators

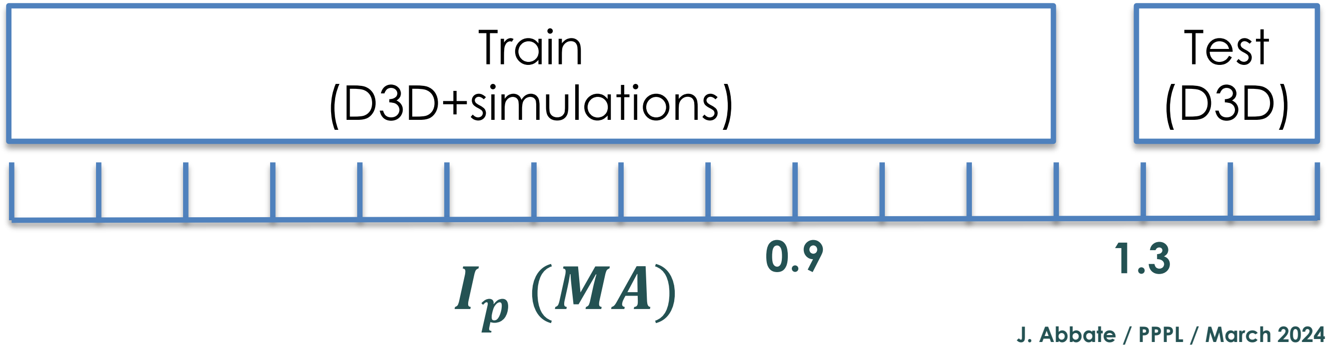


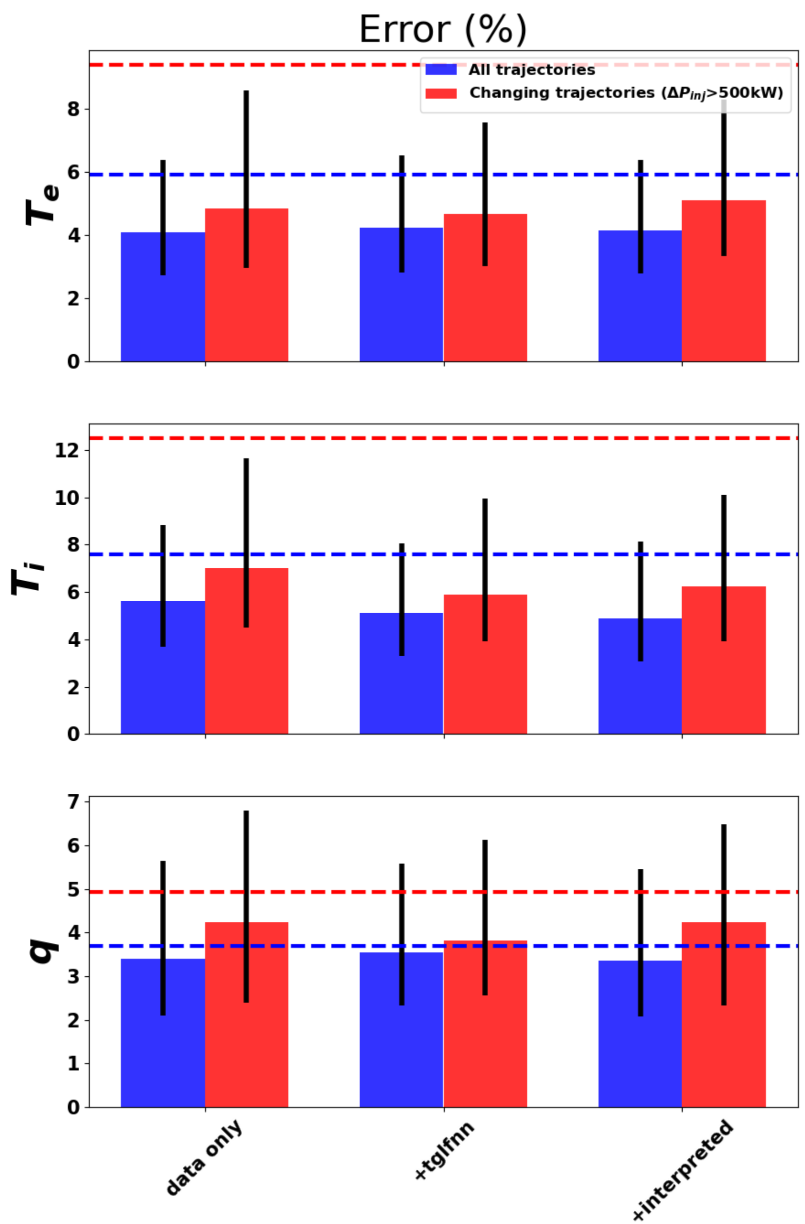




Concatenate simulation context as additional input

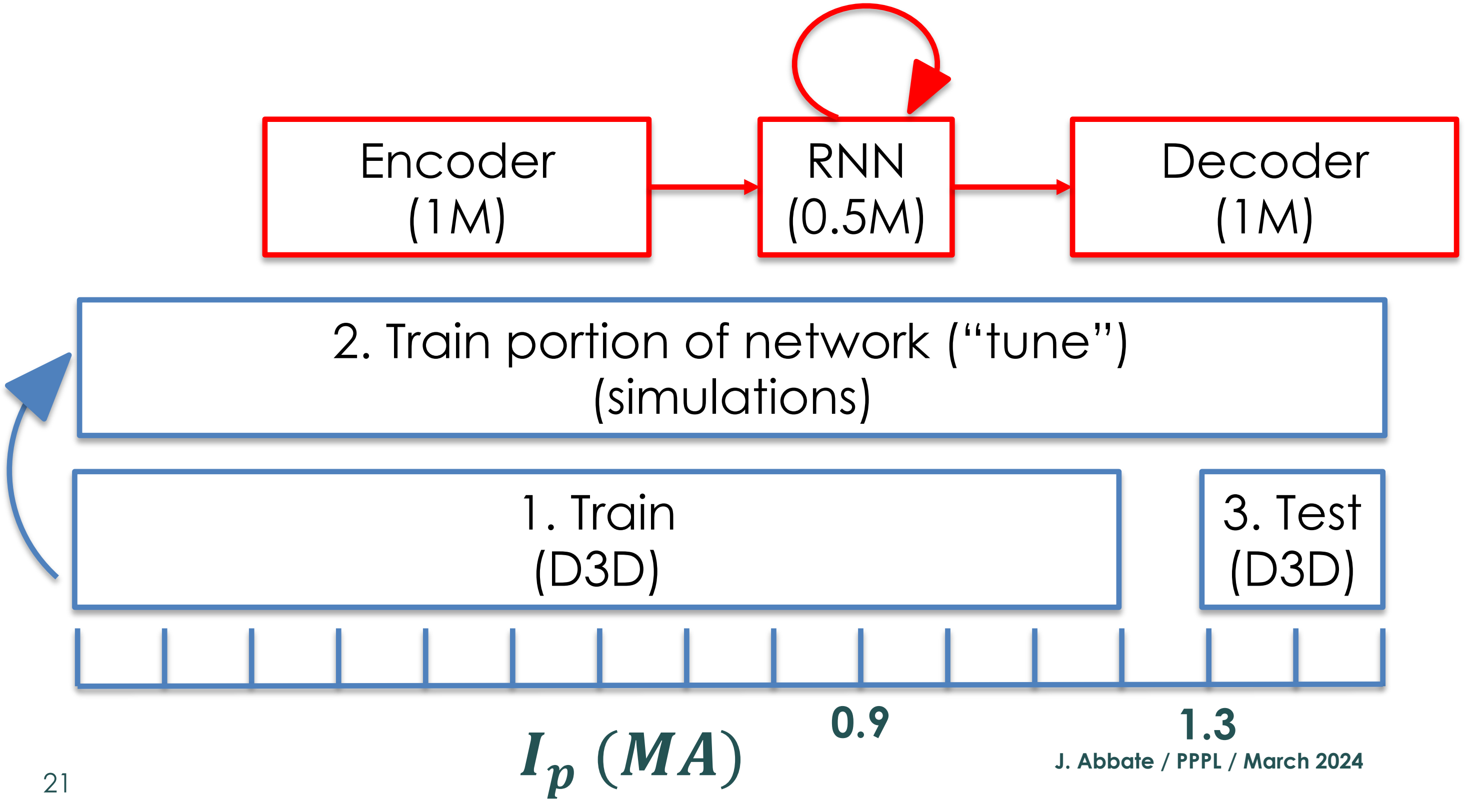
- **Predicted quantities**
  - **Core electron, ion temperature (TGLF-nn)**
- **Interpreted quantities**
  - **Total heat to electrons, ions**
  - **Driven current**
- **No statistically significant improvement over data-driven model for any signal**
- **UPSHOT: details of heat and current deposition do not seem important (~profile consistency)**
  - **Maybe more / better simulations?**

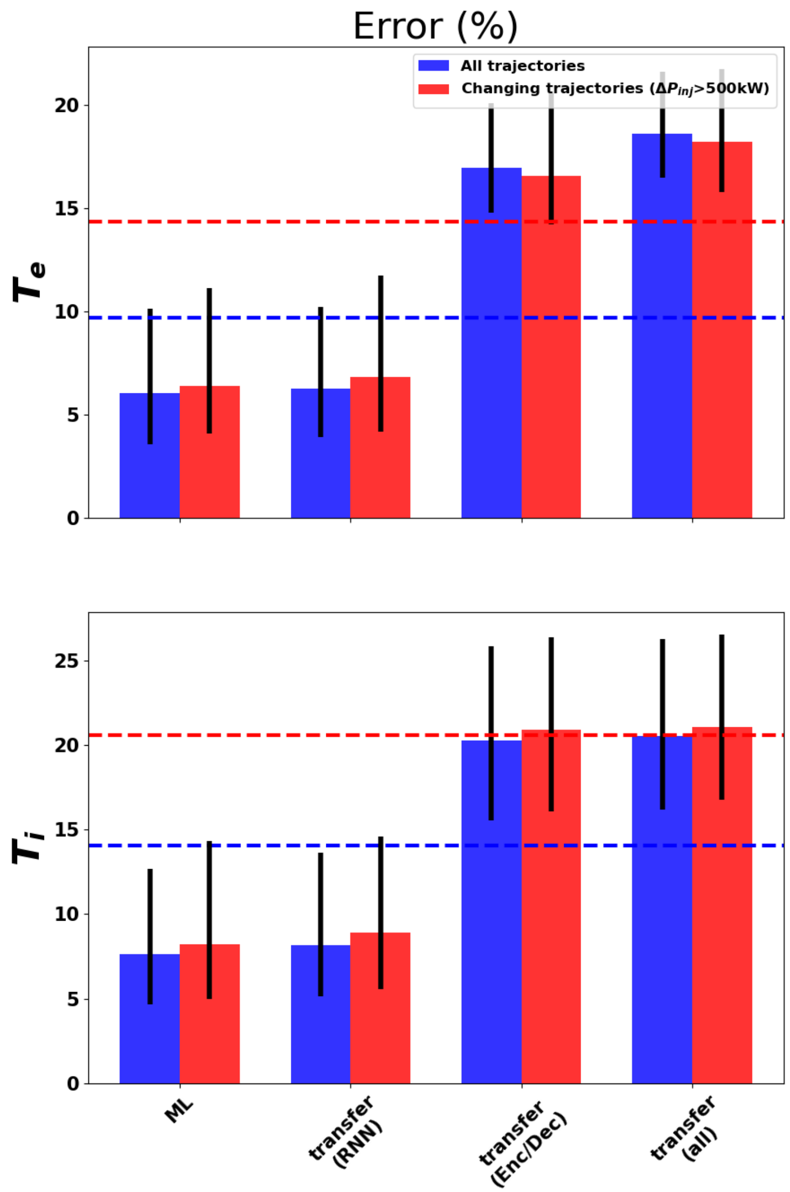




Transfer learn by training on experiment data, tuning on simulation

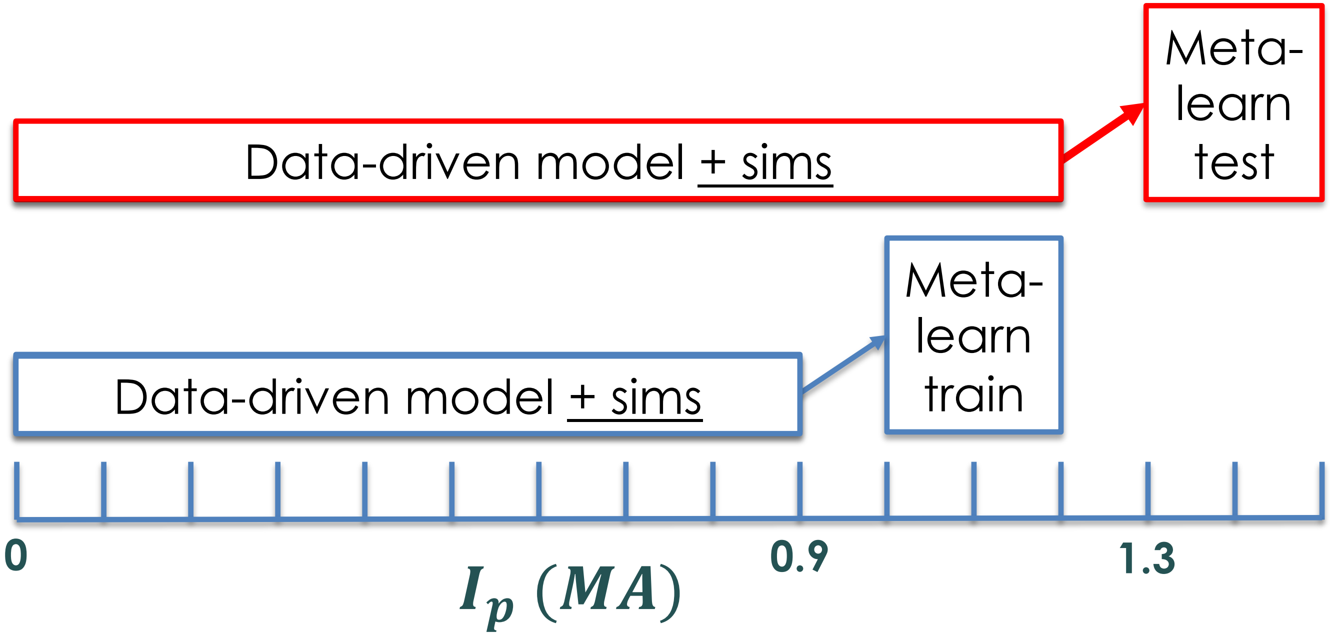
- **Pretend simulations are reality in regimes we haven't yet seen**
- **Once again: no significant improvement**
- **UPSHOT: simulations are not quantitatively accurate**
  - **Again, maybe more/better simulations?**





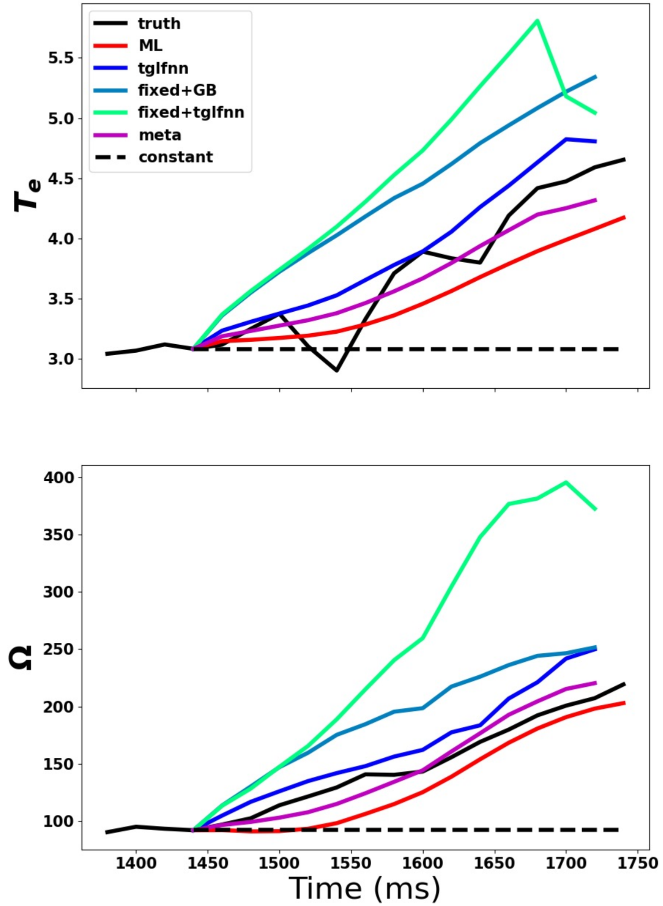
Meta-learned model ensembling output of data-driven and physics models

- **Explicitly train model for *task* of extrapolation**
- **Train data-driven model(s) on one dataset**
- **Build meta-learning model that predicts on extrapolated dataset given model output**
- **Also consider simulations on equal footing**



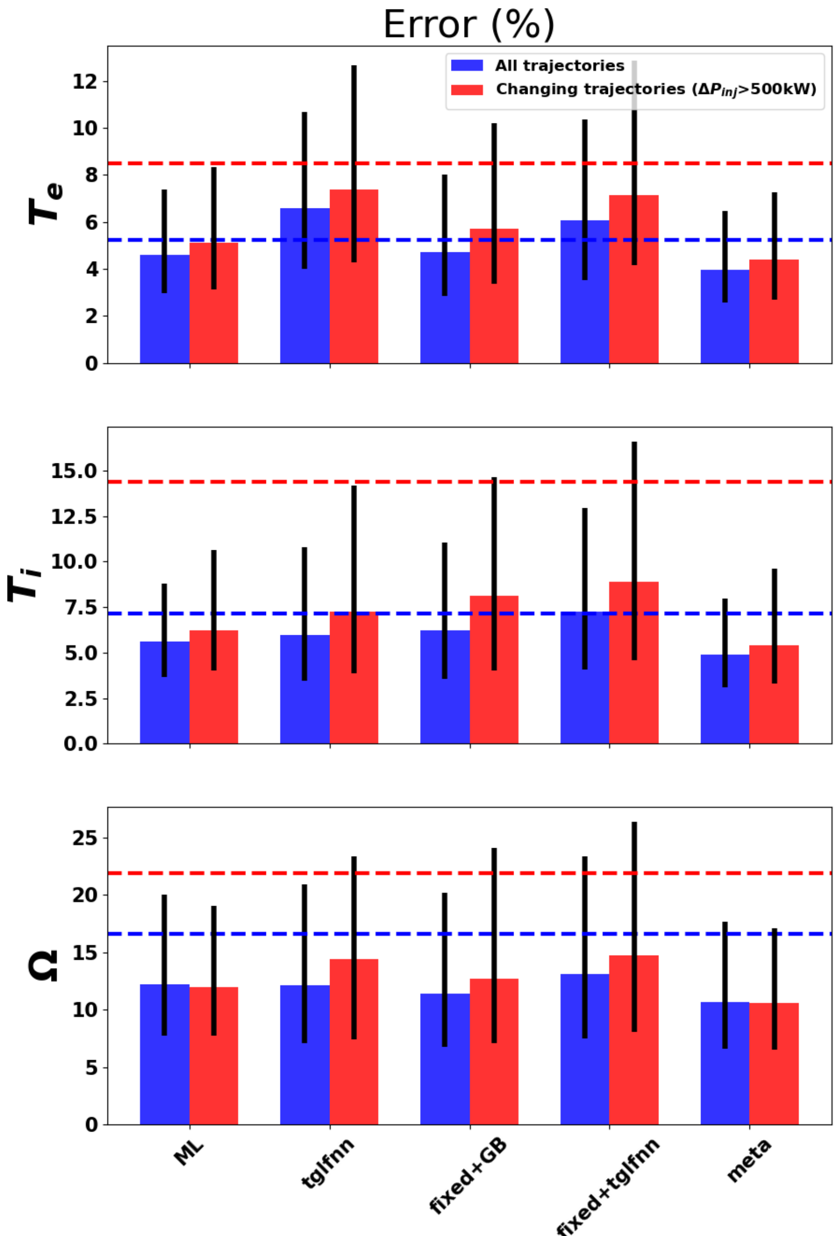
E.g. logistic regression: learn coefficients in

$$T_{meta} = \alpha_{data} T_{data} + \alpha_{TGLFnn} T_{TGLFnn} + \alpha_{fixedGB} T_{fixedGB} + \alpha_{fixedTGLFnn} T_{fixedTGLFnn}$$



Plasma Control

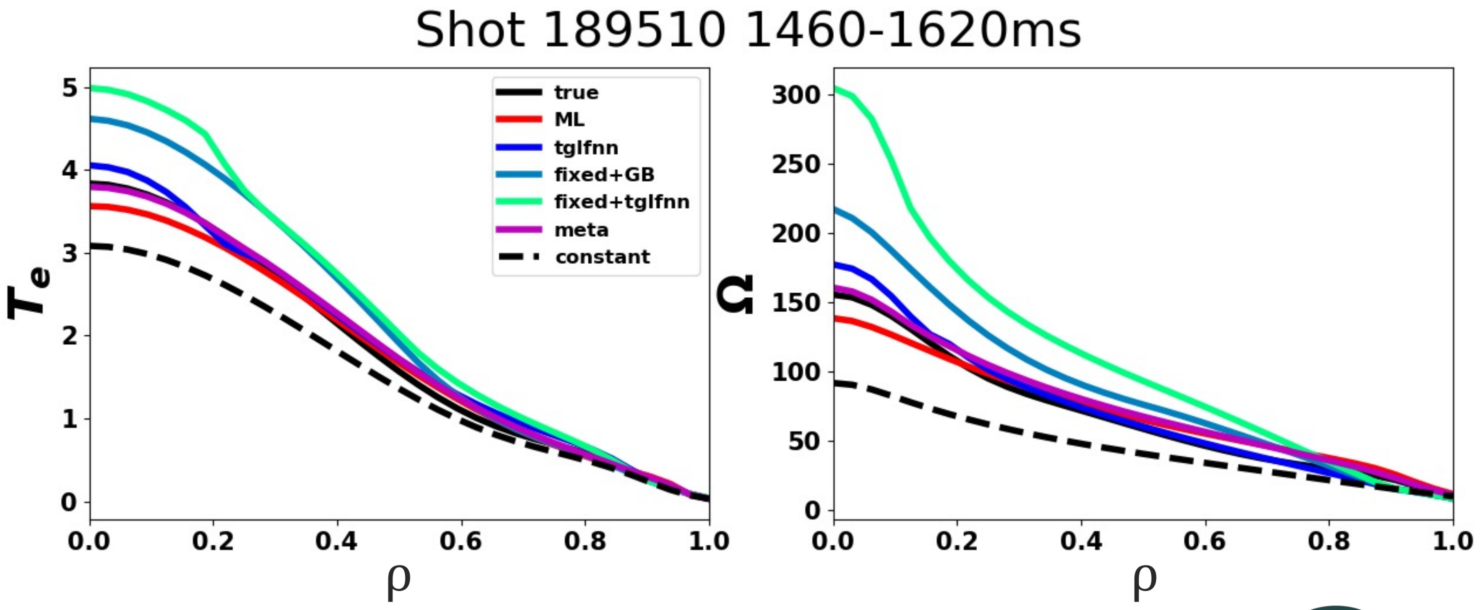
Meta-learned model outperforms data-driven or simulations alone!



$$\alpha_{data} = 64\%$$
$$\alpha_{fixedGB} = 19\%$$
$$\alpha_{TGLFnn} = 15\%$$
$$\alpha_{fixedTGLFnn} = 2\%$$

E.g. logistic regression: learn coefficients in

$$T_{meta} = \alpha_{data} T_{data} + \alpha_{TGLFnn} T_{TGLFnn} + \alpha_{fixedGB} T_{fixedGB} + \alpha_{fixedTGLFnn} T_{fixedTGLFnn}$$



Plasma Control

Thesis contributions: guidance

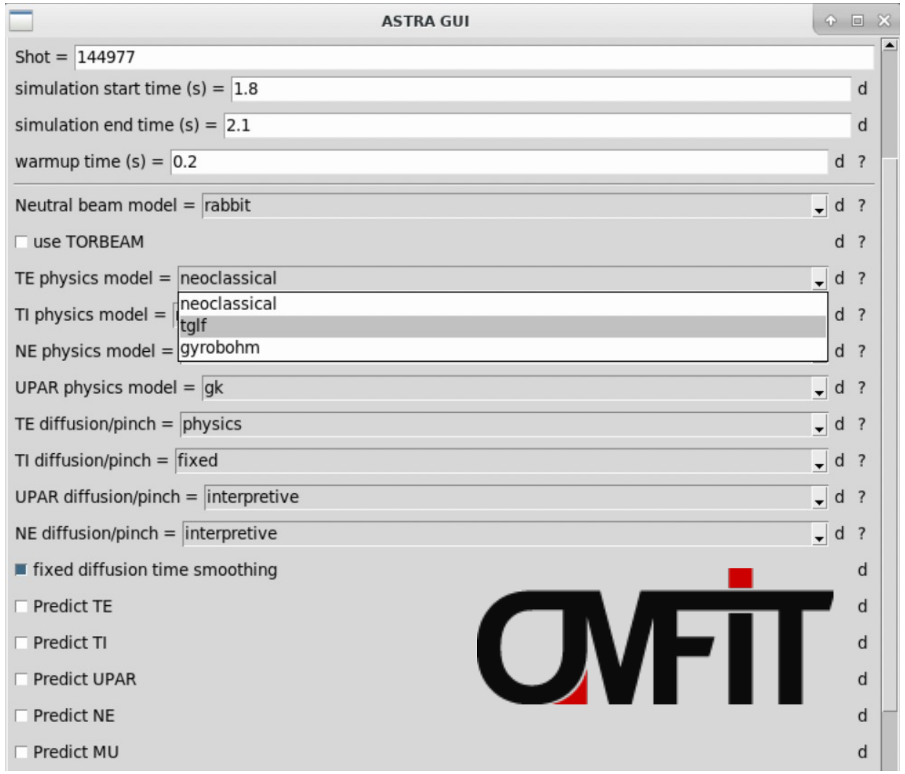
- **“Always have a baseline”: use hypothesis testing for validation**
- **Database scans are useful (less bias, more confidence)**
- **Simple simulation context doesn’t aid predictions**
  - **Like profile consistency: details of heat deposition don’t matter**
- **ITER will need to rely largely on its own data for control development**
  - **Don’t sacrifice pre-operation commissioning time for sake of speed**
- **Take a clear stance on spectrum of “end-to-end” to “interpretable” models**
  - **Setting low-level vs high-level control targets (e.g. gas vs target density)**
  - **Predicting diagnostics directly vs fitted profiles**

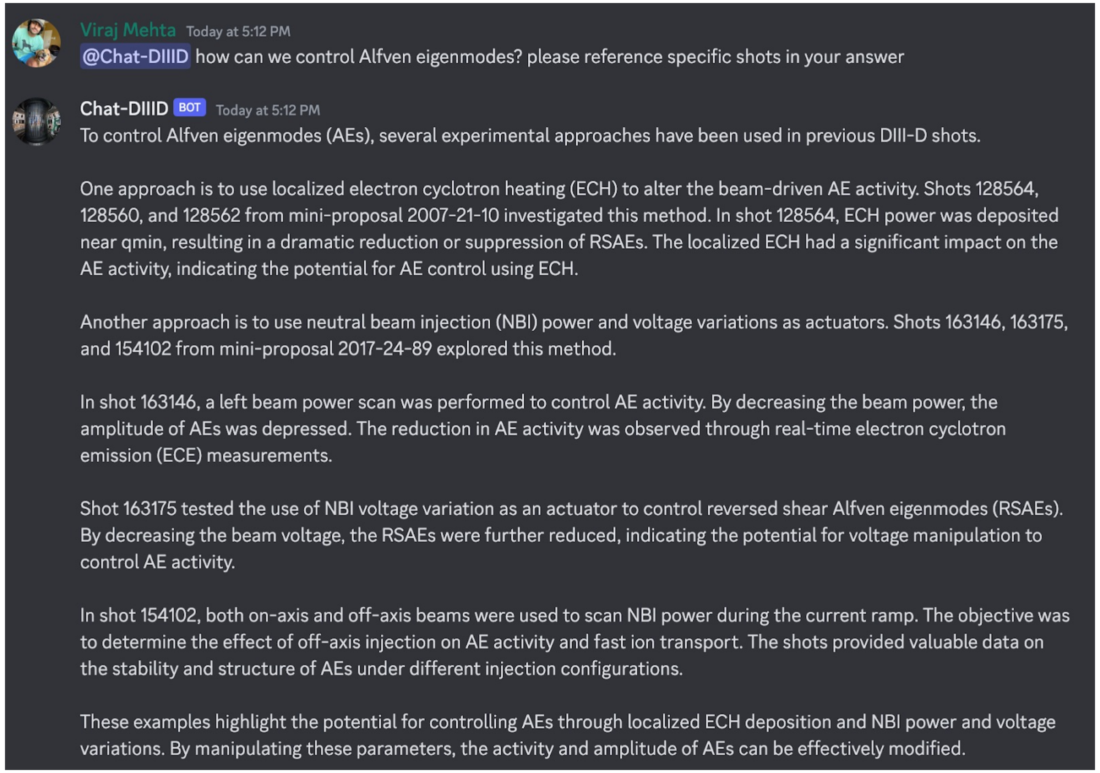
- **Once we have a clear task to deploy, an open competition would be effective**
  - **If data access a problem, then at least within the fusion community**

Thesis contributions: infrastructure

- **With Emiliano/Giovanni, upgraded ASTRA and made OMFIT interface**
- **Pipeline for fetching standardized dataset (feat. Brian Sammuli's toksearch)**
- **With Rory, neural network control in DIII-D PCS**
- **With Ian, arbitrary-timestep profile dynamics models**
- **With Viraj/Allen/Andy/Ian, Chat-D3D**

tinyurl.com/abbate-phd-defense

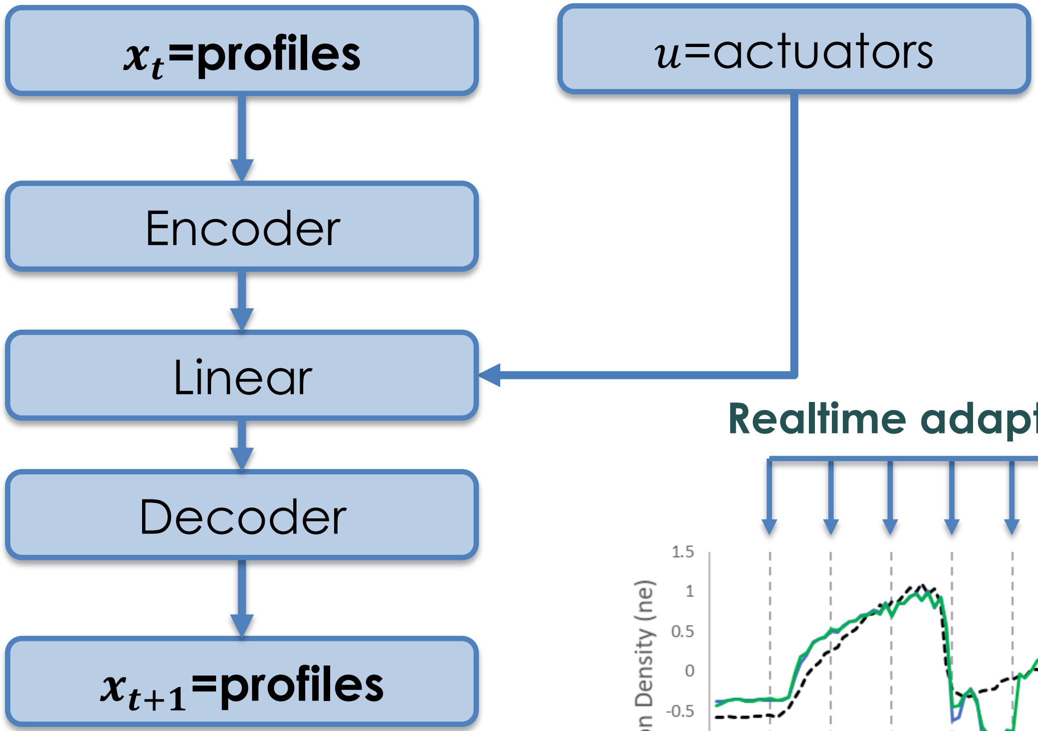




Plasma Control

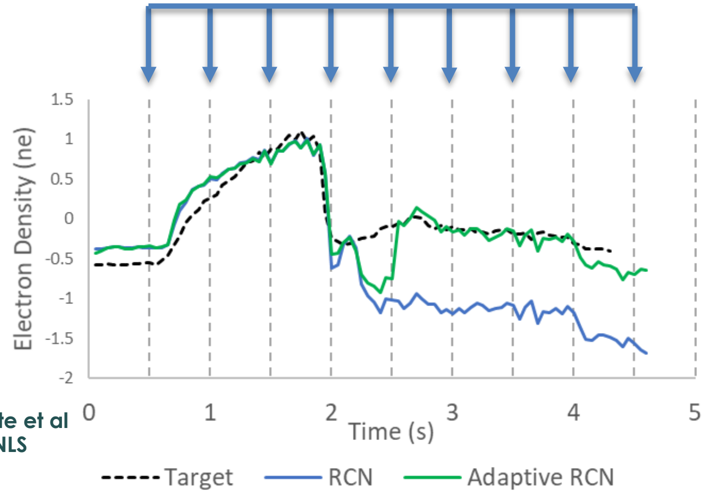
Work in progress: Upgrades for profile prediction

Linear-state-space model



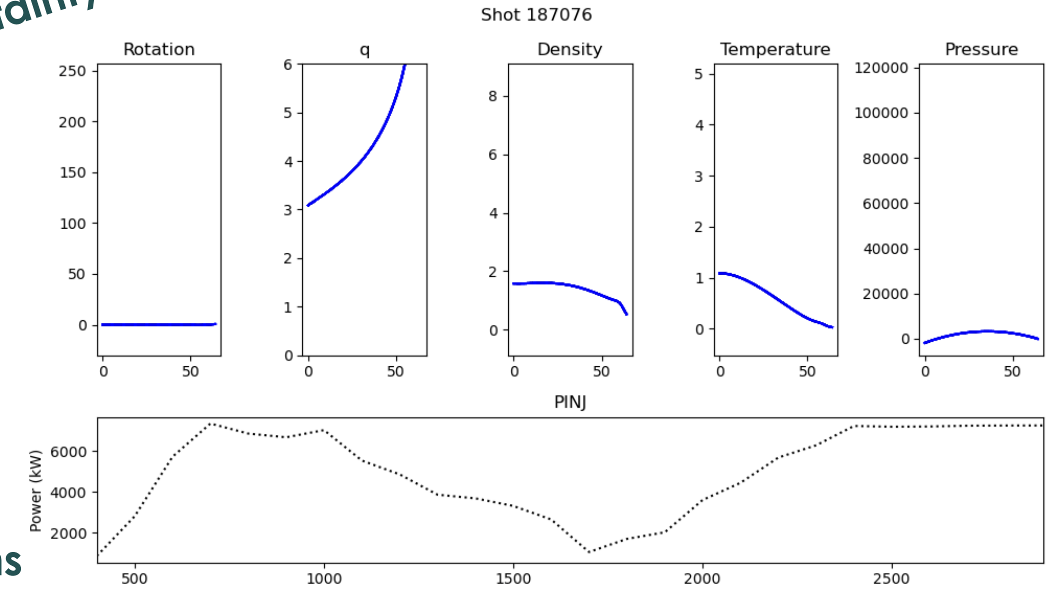
Farre, Abbate et al *in preparation*

Realtime adapt every 500ms



Jalalvand, Abbate et al 2022 IEEE TNNLS

Add uncertainty



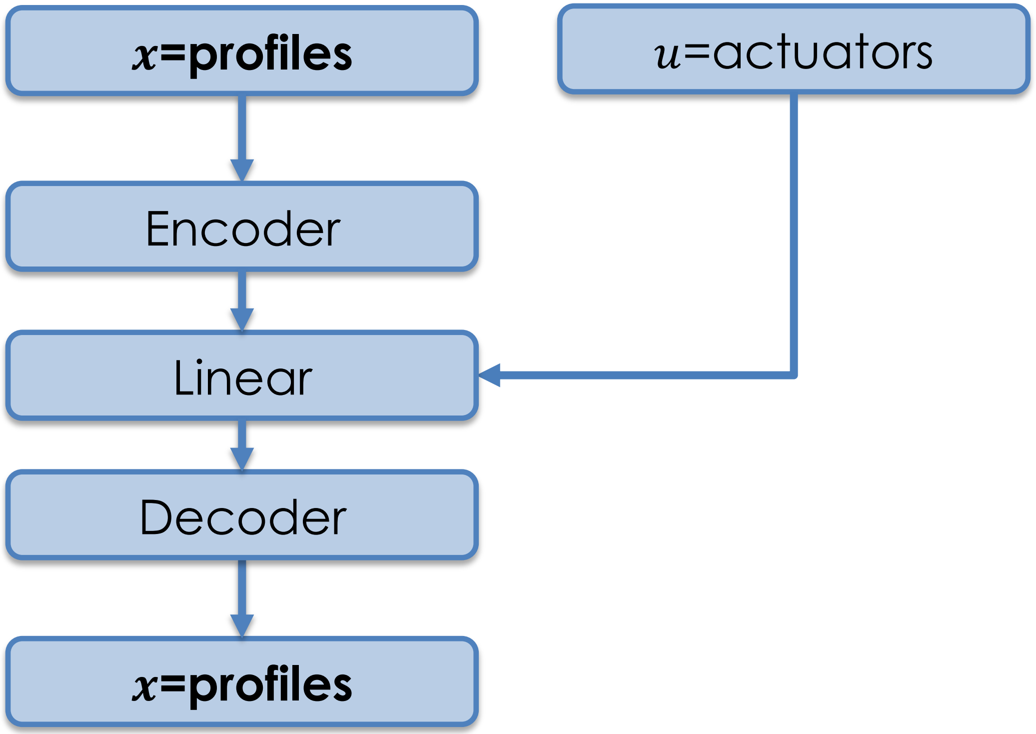
Char, Abbate *in preparation*

Find mapping f s.t.

$$x_{t+1} = f(x_t, u_t, u_{t+1})$$

Backup slides

“Linear latent space” approach to leverage full Model-Predictive Control



$$z = f(x)$$

$$z_{t+1} = Az_t + Bu_t$$

$$z_{t+2} = Az_{t+1} + Bu_{t+1} = A^2 z_t + ABu_t + Bu_{t+1}$$

$$\dots$$

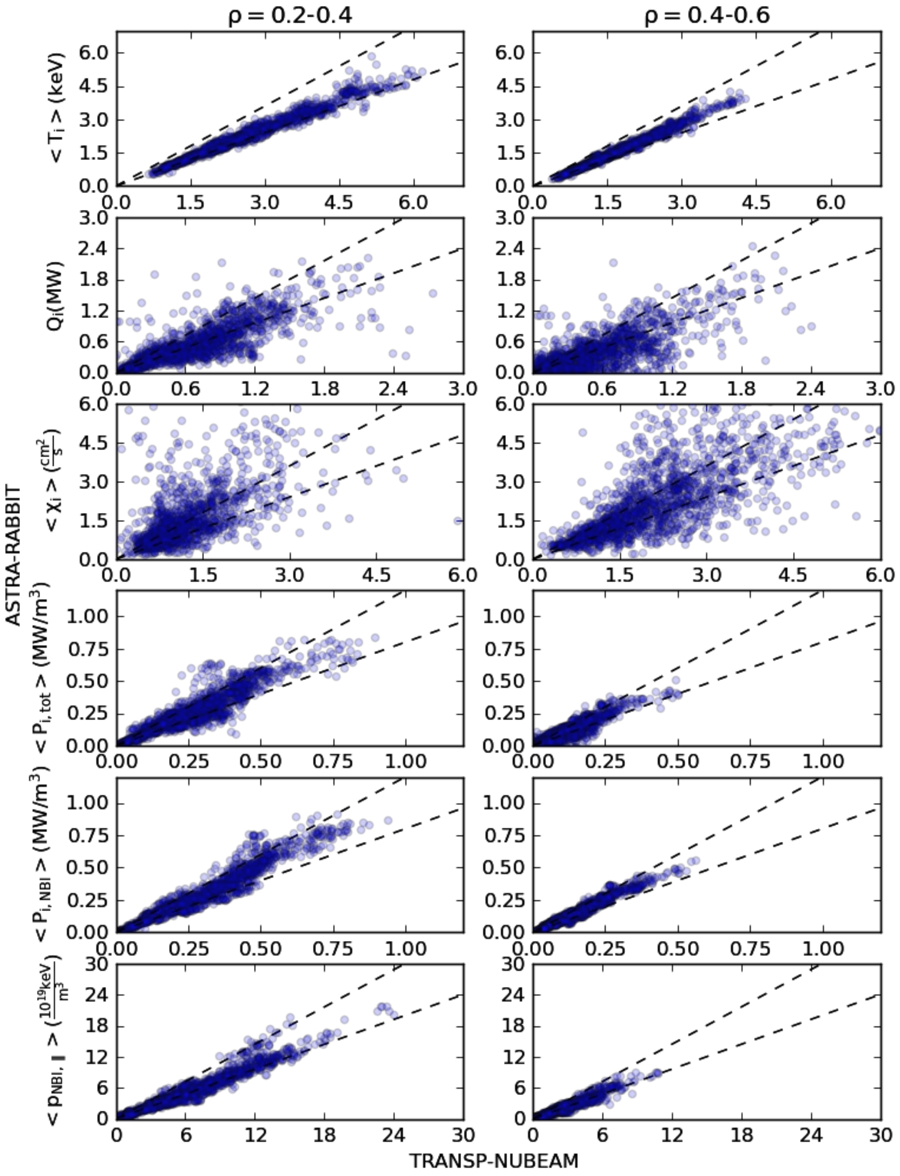
$$z_{t+N} = A^N z_t + A^{N-1} Bu_t + \cdots + Bu_{t+N}$$

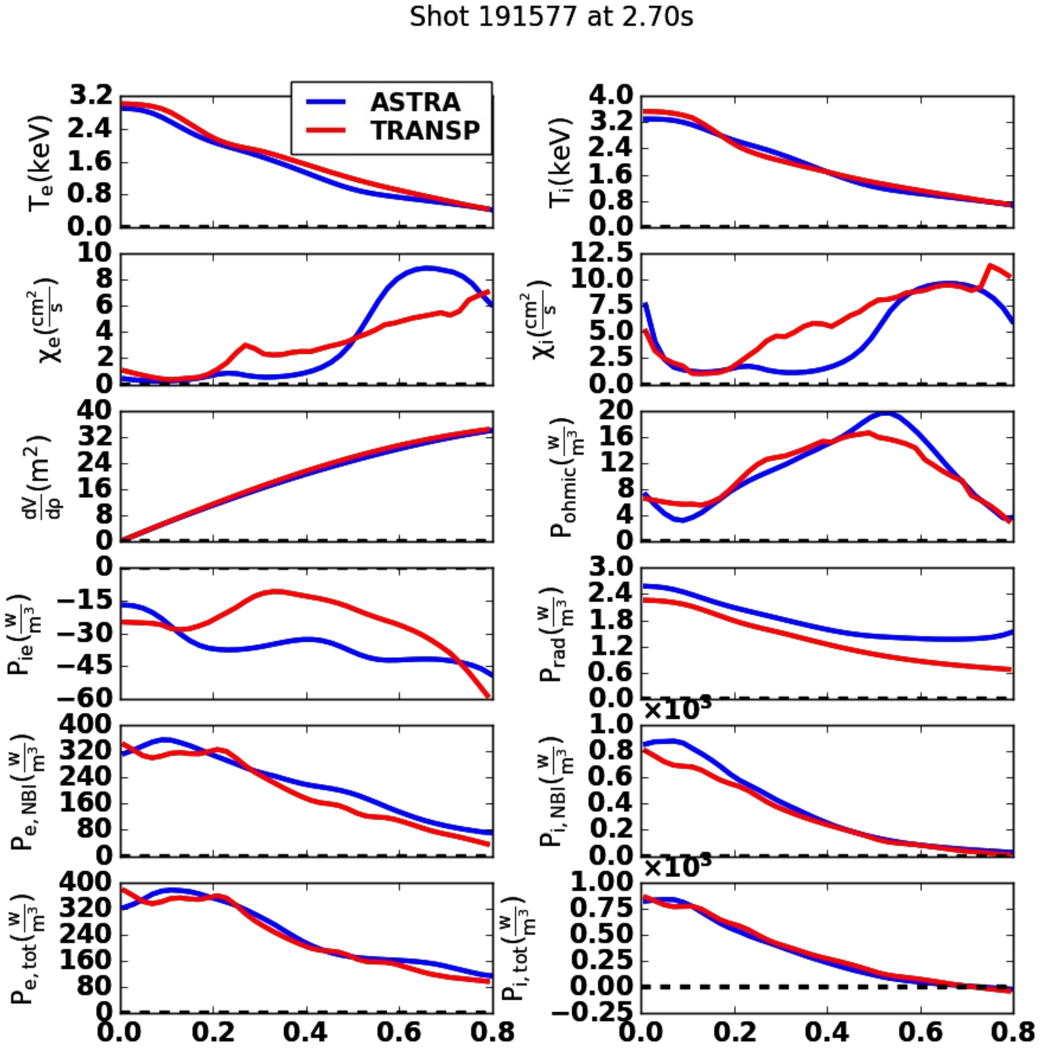
Encoded state at any time in future is linear function of the initial state and the trajectory of actuators

Model-Predictive Control:

Minimize $(z - z_{target})^T Q (z - z_{target}) + u^T R u$
s.t. $Du < d$

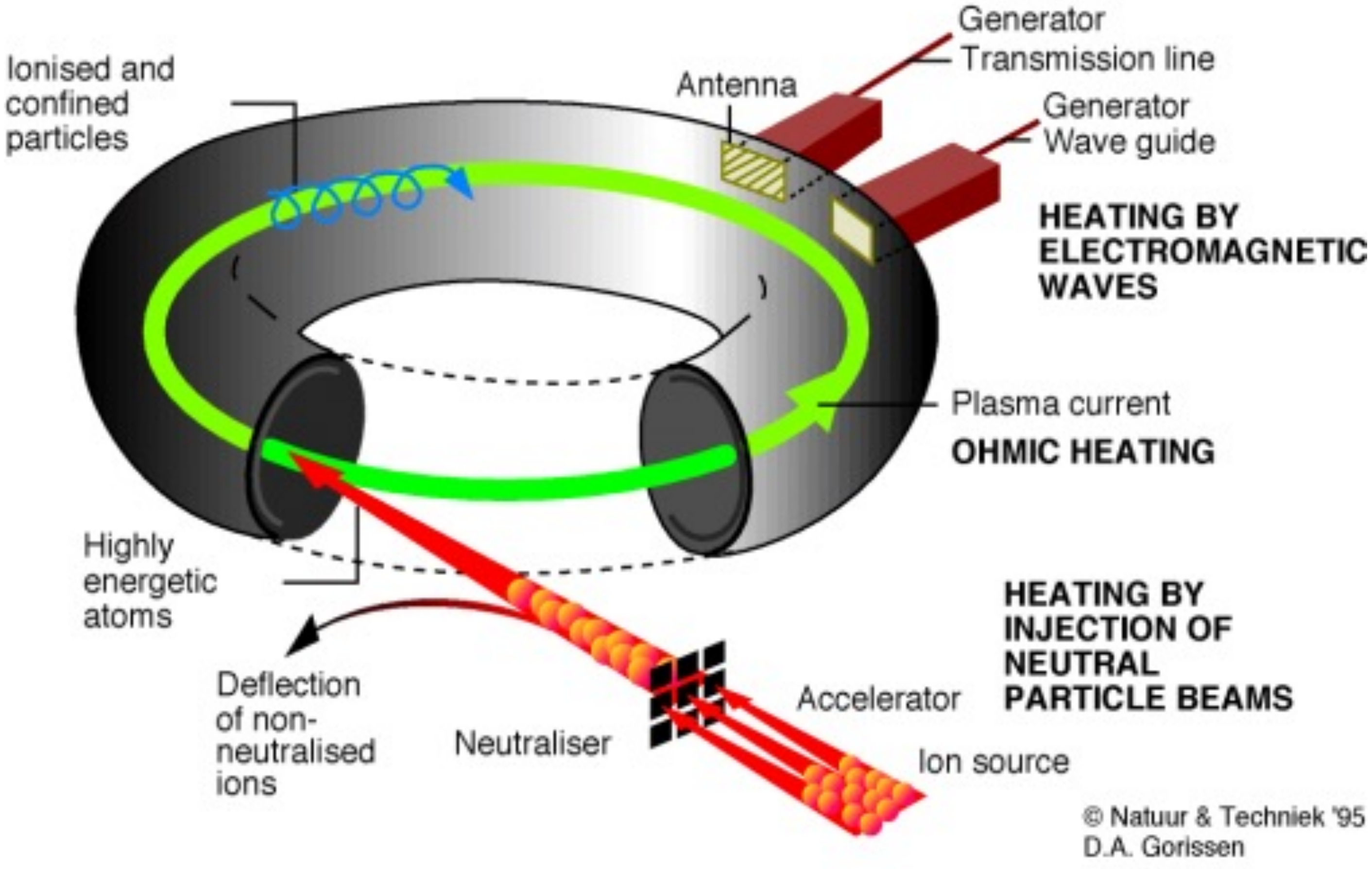
More details on ASTRA/TRANSP comparison



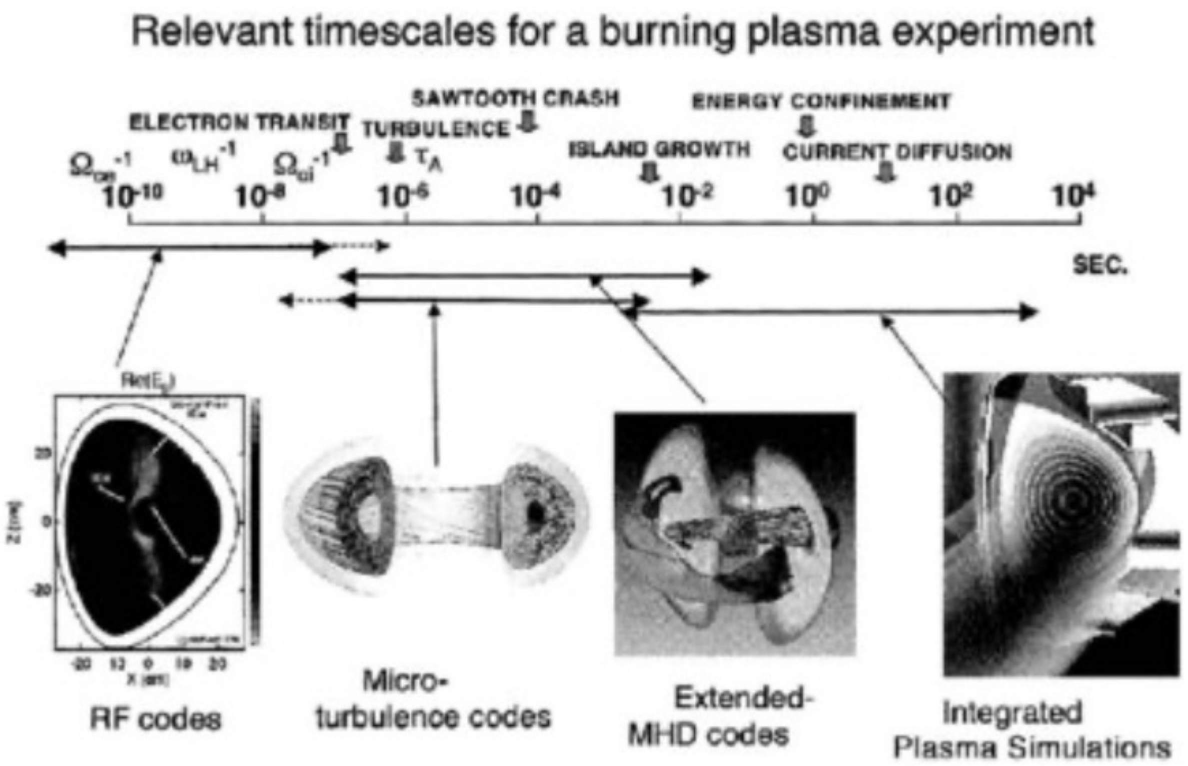


Kinetic plasma profile actuators

- Total neutral beam power
- Total neutral beam torque
- Total ECH power
- Target plasma current
- Total deuterium gas puffed
- Target toroidal magnetic field
- Target plasma shape parameters



Timescales



Running surrogate models: ms
Running simple sims: hour
Running better sims: month
Training data-driven models: <days
Making reduced models: ~decades

~10 Years	~Year	~Months	~Minutes	~Seconds
Design/build device	Upgrade device	Plan experiments	Between-experiment adjustments	Realtime adjustments

Report of the Fusion Simulation Project Steering Committee, 2004

1st+2nd author publications

Cross-verification and validation of predictive capabilities of the ASTRA and TRANSP codes
***Abbate**, Fable et al Physics of Plasmas [in review]*
Created a novel benchmark (an empirical model) for validation and for the first time compared ASTRA 8

Towards LLMs as Operational Copilots for Fusion Reactors
*Mehta, **Abbate** et al 2023 NeurIPS Workshop AI4Science [in review]*
Applied large language models in a retrieval augmented generation system to assist tokamak operators. Deployed as a bot in the DIII-D tokamak operations Discord, and on an MIT server for the Alcator C-Mod tokamak.

A general infrastructure for data-driven control design and implementation in tokamaks
***Abbate**, Conlin et al 2023 Journal of Plasma Physics 89(1) 895890102*
Deployed the model below in a model-predictive framework on DIII-D (using finite set control) to achieve user-specified pressure and temperature profiles by varying injected power and torque.

Offline Model-Based Reinforcement Learning for Tokamak Control
*Char, **Abbate** et al 2023 Learning for Dynamics and Control Conference*
Trained similar model but with uncertainty output and more robust long-time dynamics, deployed in reinforcement learning controller at DIII-D to achieve user-specified betan by varying beam power

Real-Time and Adaptive Reservoir Computing With Application to Profile Prediction in Fusion Plasma
*Jalalvand, **Abbate** et al, 2022 IEEE Transactions on Neural Networks and Learning Systems vol. 33 no. 6*
Rebuilt the model using Reservoir Computing Networks for realtime adaptability and towards a linear statespace

Data-driven profile prediction for DIII-D
***Abbate**, Conlin et al 2021 Nuclear Fusion 61 046027*
Developed a first-of-its-kind fully data-driven machine learning model to simulate plasma dynamics

Plasma Control