Tokamak prediction and control

J. Abbate[1],

R. Conlin[2], A. Jalalvand[2,3], R. Shousha[2],
I. Char[4], Y. Chung[4], V. Mehta[4],
E. Fable[5], G. Tardini[5], E. Kolemen[1,2]

[1] Princeton Plasma Physics Laboratory

[2] Princeton University

[3] Ghent University

[4] Carnegie Mellon University

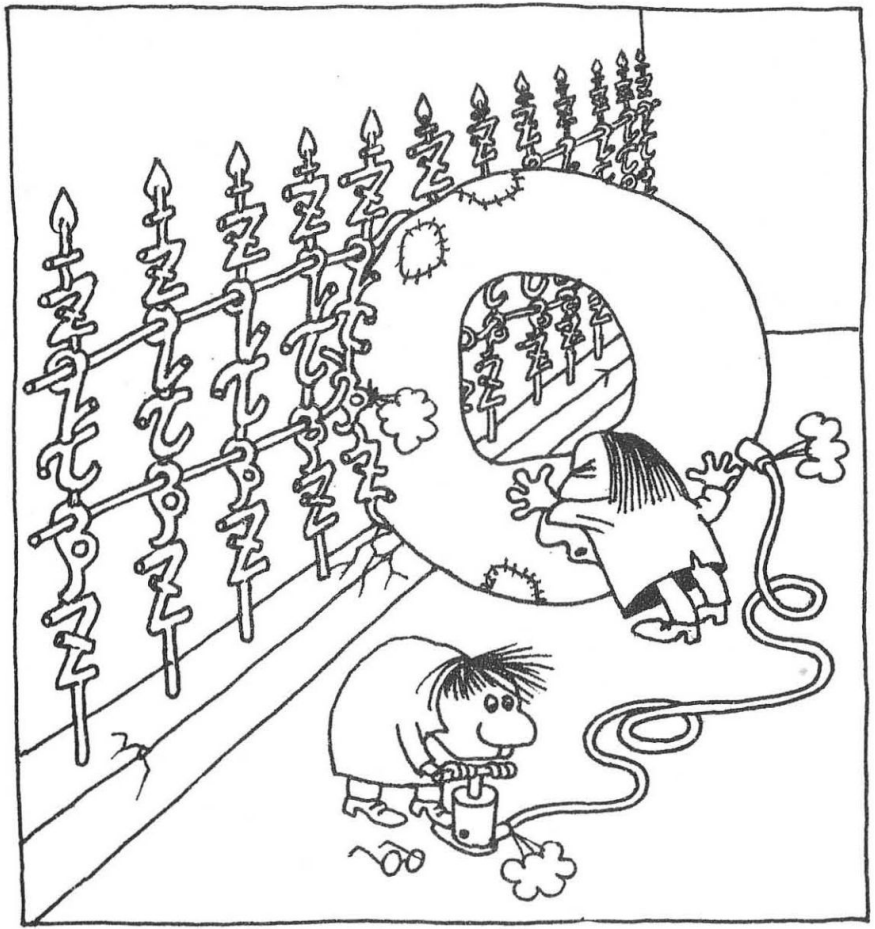
[5] Max Planck Institute for Plasma Physics

MHD Workshop

October 14, 2022

Princeton Plasma Control

control.princeton.edu



Cartoon:
EPS 1981
Kadomtsev?

Reproducing and improving a plasma by trial-and-error

“[after a previous full-day experiment we were] able to reproduce 133103 in 180636, 180643 and 180644.”

Reproducing and improving a plasma by trial-and-error

"[after a previous full-day experiment we were] able to reproduce 133103 in 180636, 180643 and 180644.

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."

Reproducing and improving a plasma by trial-and-error

"[after a previous full-day experiment we were] able to reproduce 133103 in 180636, 180643 and 180644.

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.)"

Reproducing and improving a plasma by trial-and-error

"[after a previous full-day experiment we were] able to reproduce 133103 in 180636, 180643 and 180644.

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"

Reproducing and improving a plasma by trial-and-error

"[after a previous full-day experiment we were] able to reproduce 133103 in 180636, 180643 and 180644.

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"

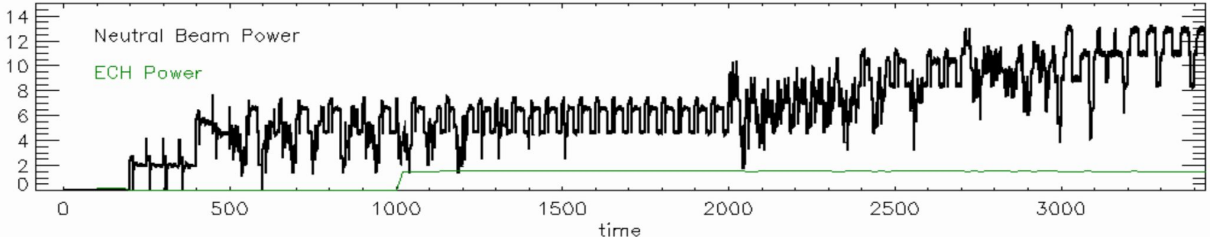
Human operators combine simulations, heuristics, and experience to achieve desired state by trial-and-error

Tokamak prediction and control via machine learning

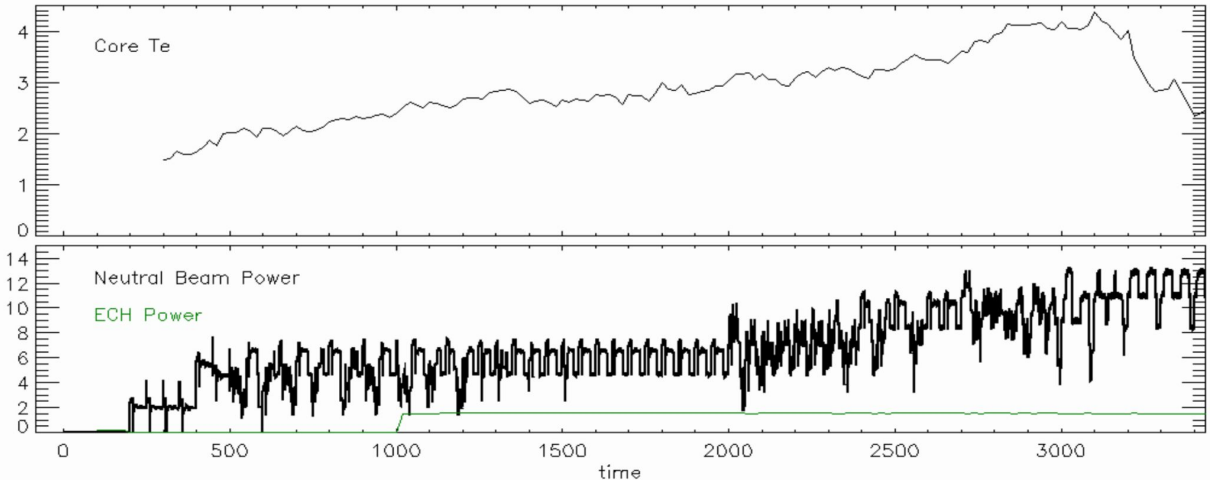
1. Plasma predictions: traditional methods
2. Plasma predictions: data-driven machine learning
3. Tokamak control

Plasma predictions: traditional methods
MHD stability

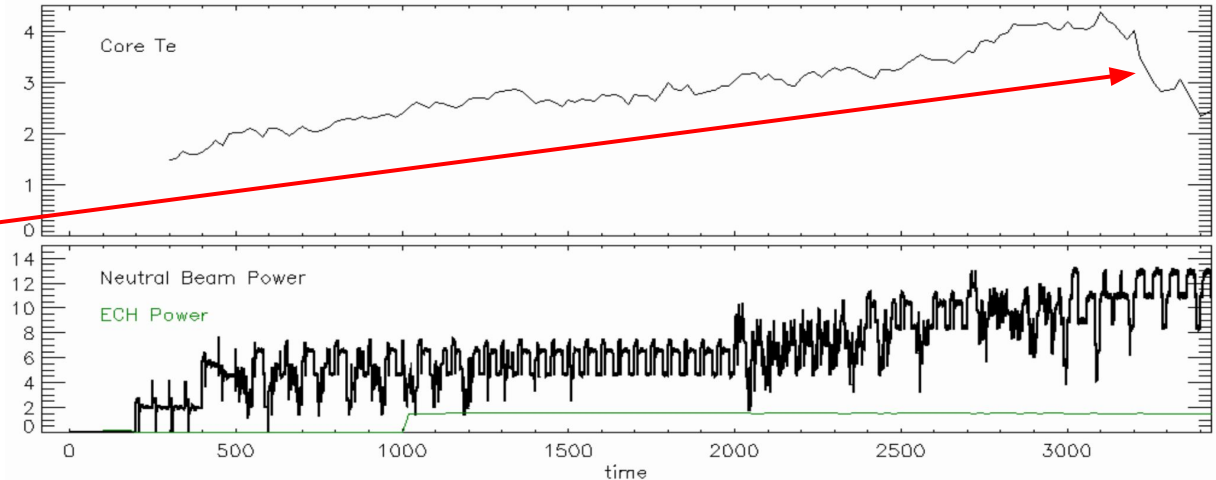
Simple observations



Simple observations

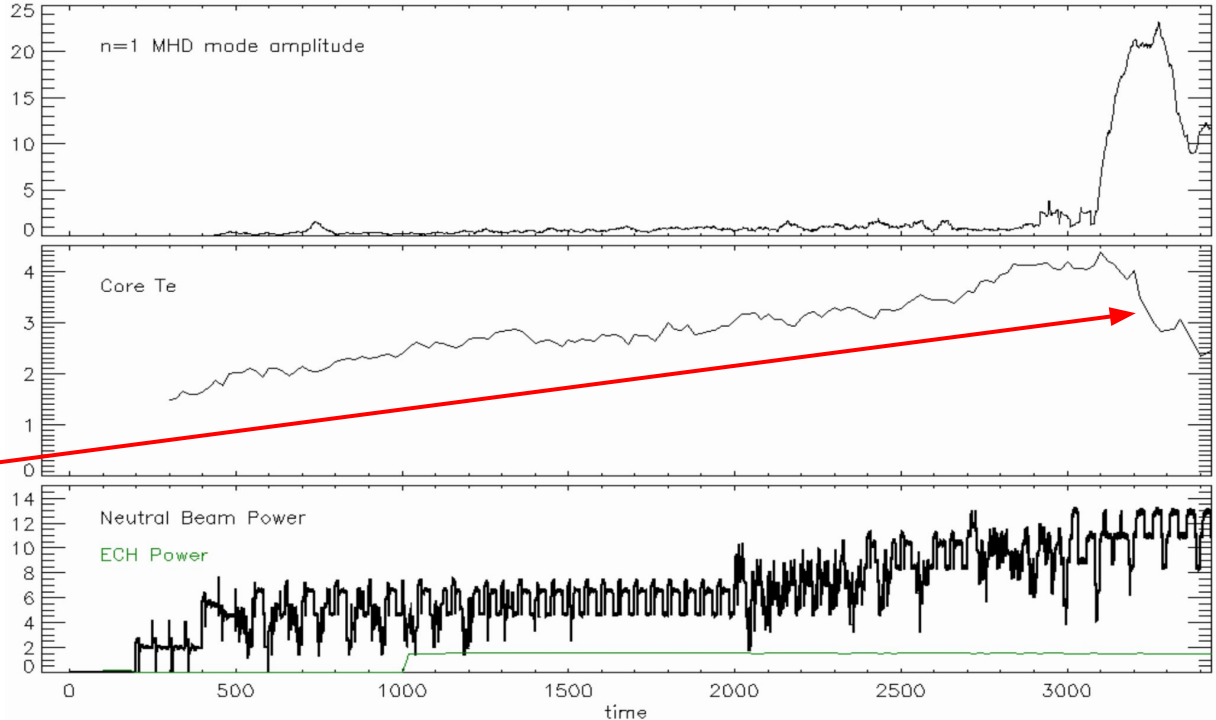


Simple observations



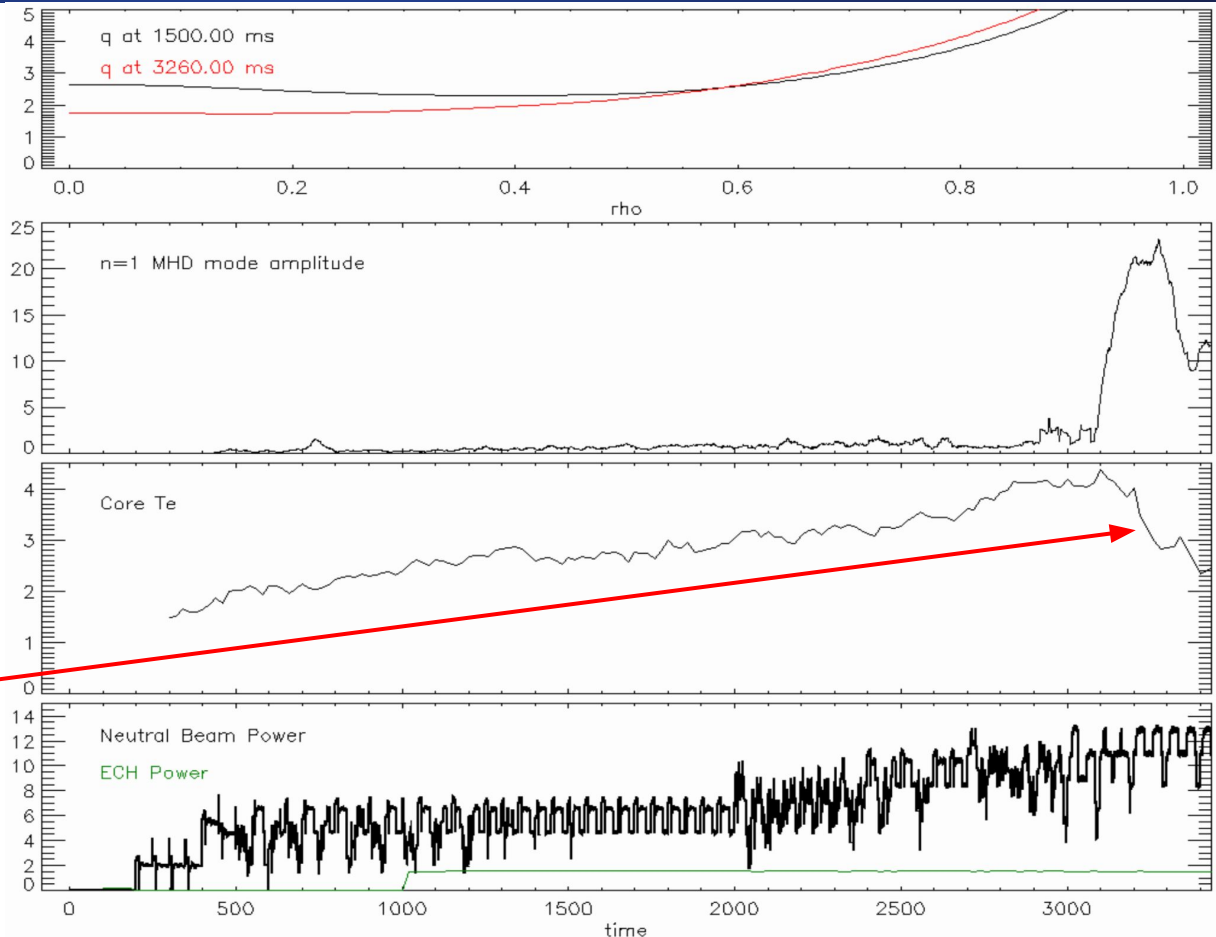
Sudden loss of confinement without any clear change in actuators

Simple observations



Sudden loss of confinement without any clear change in actuators

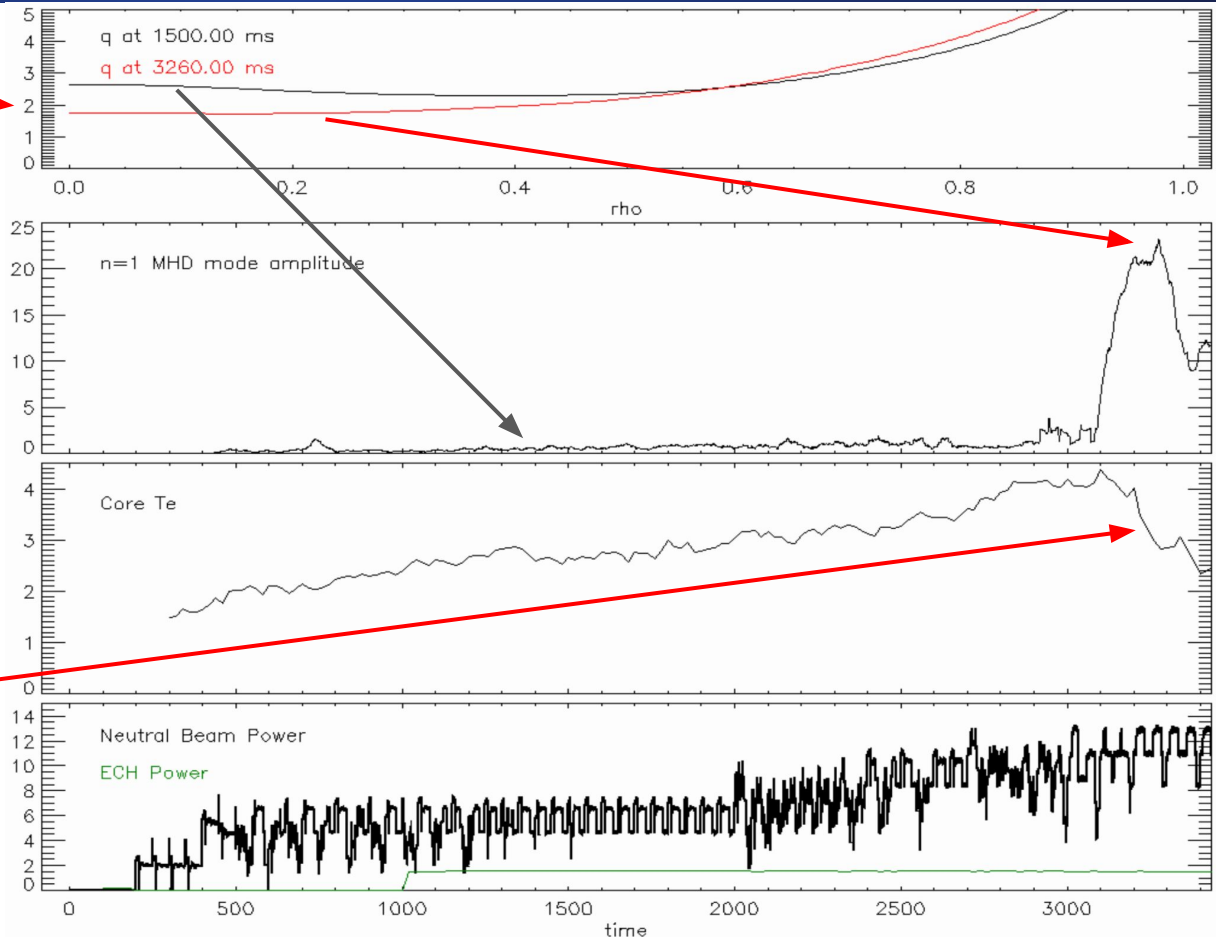
Simple observations



Sudden loss of confinement without any clear change in actuators

Simple observations

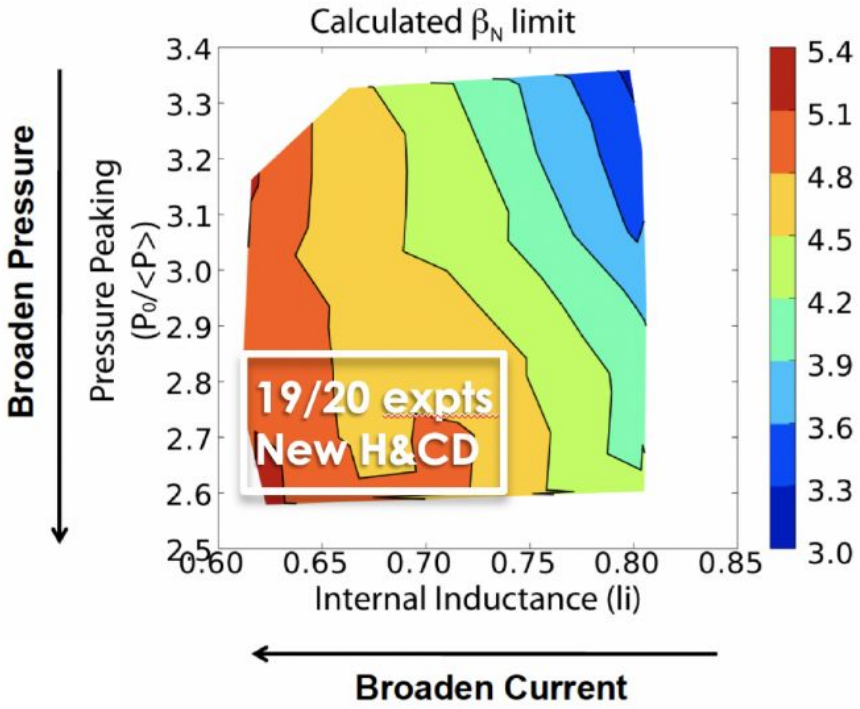
"Many shots had MHD modes at 3 s when we cross the 2/1 surface"



Sudden loss of confinement without any clear change in actuators

Physics calculations

- Hundreds of equations and codes for tackling variety of length and timescales for different tokamak phenomena
- Rarely match experiments exactly, but give scalings and information
  - E.g. DCON used for stability calculations in planning for 2019/20 experiments

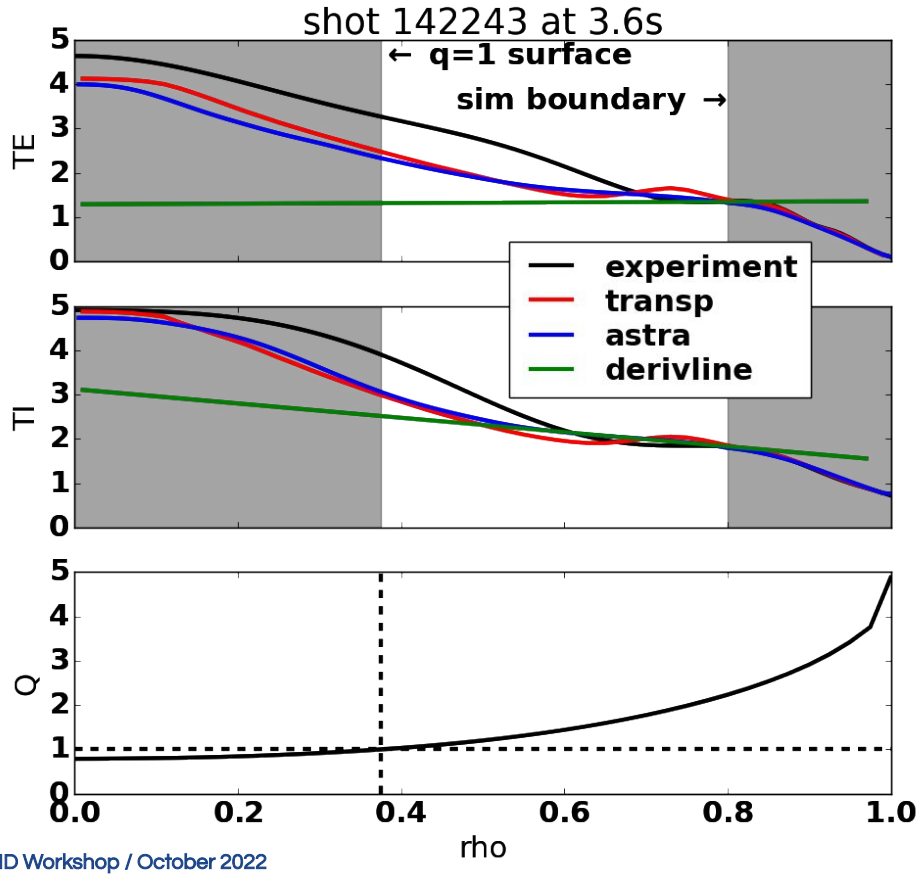


DIII-D Miniproposal, 2019

Plasma predictions: traditional methods
Transport

Reduced-physics state prediction: Transport

- No good options for full-state evolution yet
  - Transport codes like TRANSP, ASTRA, etc. are closest we have at the moment, though developers note that they are good for seeing plasma response to perturbations and not for quantitatively accurate or complete predictions
- In this case: All info from future given except core Te and Ti (which are predicted by the codes)
- Compare to "derivline": extrapolate inward from boundary value and derivative

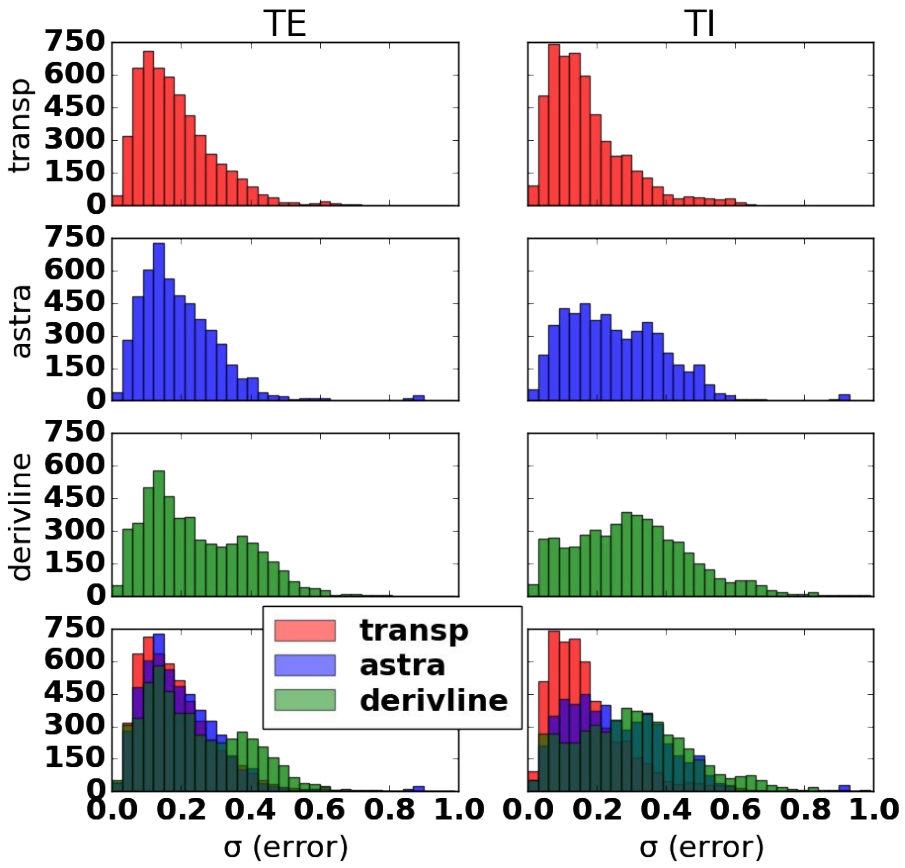


Abbate, *in preparation*

Reduced-physics state prediction: Transport

- Even when given all quantities except core Te and Ti, predictions aren't much better than just extrapolating inward from the boundary ("derivline")
  - However, they are more robust
  - And possibly still useful for control even if inaccurate

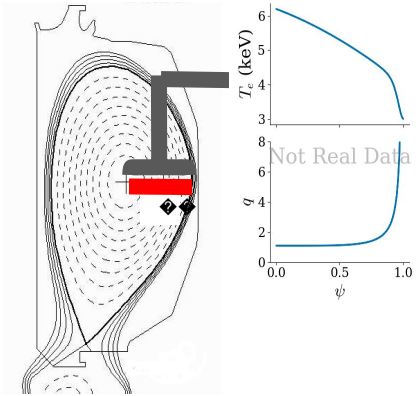
$$\sigma_{T,i} = \frac{\sqrt{\frac{1}{N}\sum_j \epsilon_j^2}}{\sqrt{\frac{1}{N}\sum_j T_{x,j}^2}},$$



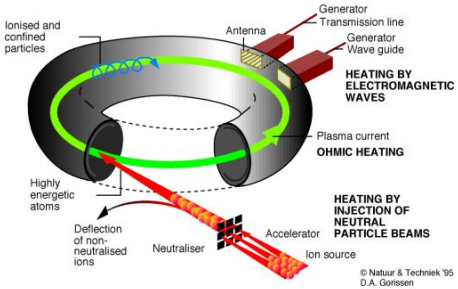
Abbate, *in preparation*

Plasma predictions: data-driven machine learning

Building a “black box” plasma response model



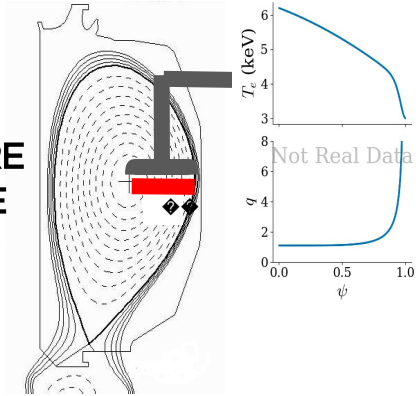
STATE



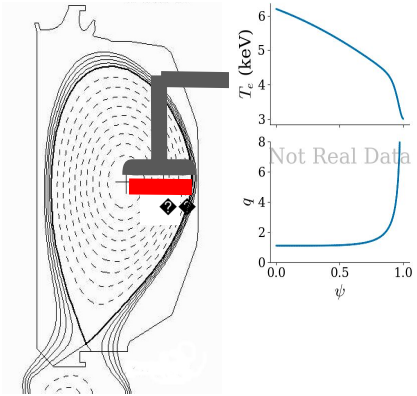
ACTUATOR TRAJECTORIES

Learn f in
y=f(x)

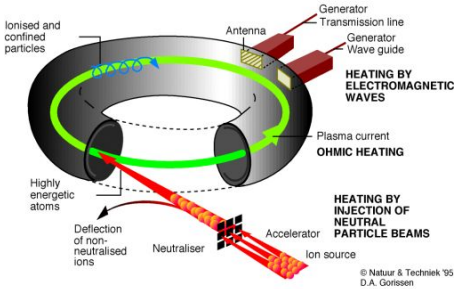
FUTURE STATE



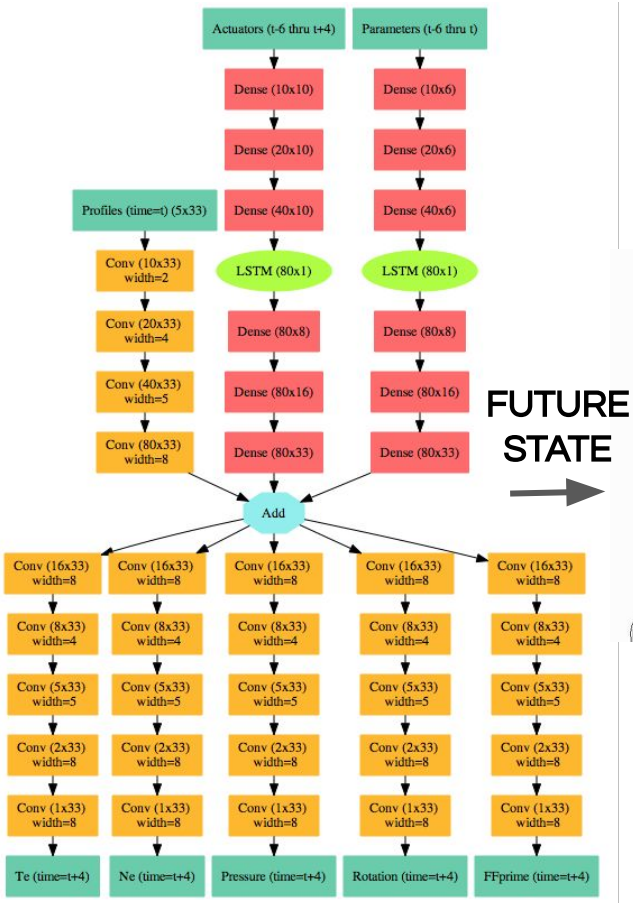
Building a "black box" plasma response model



STATE

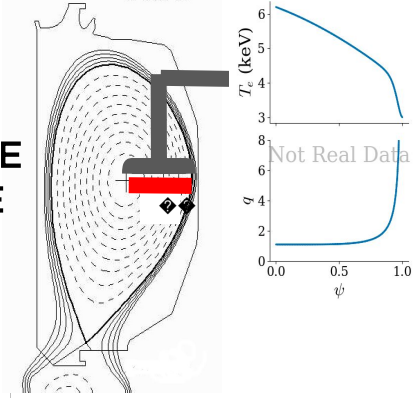


ACTUATOR TRAJECTORIES



J. Abbate /

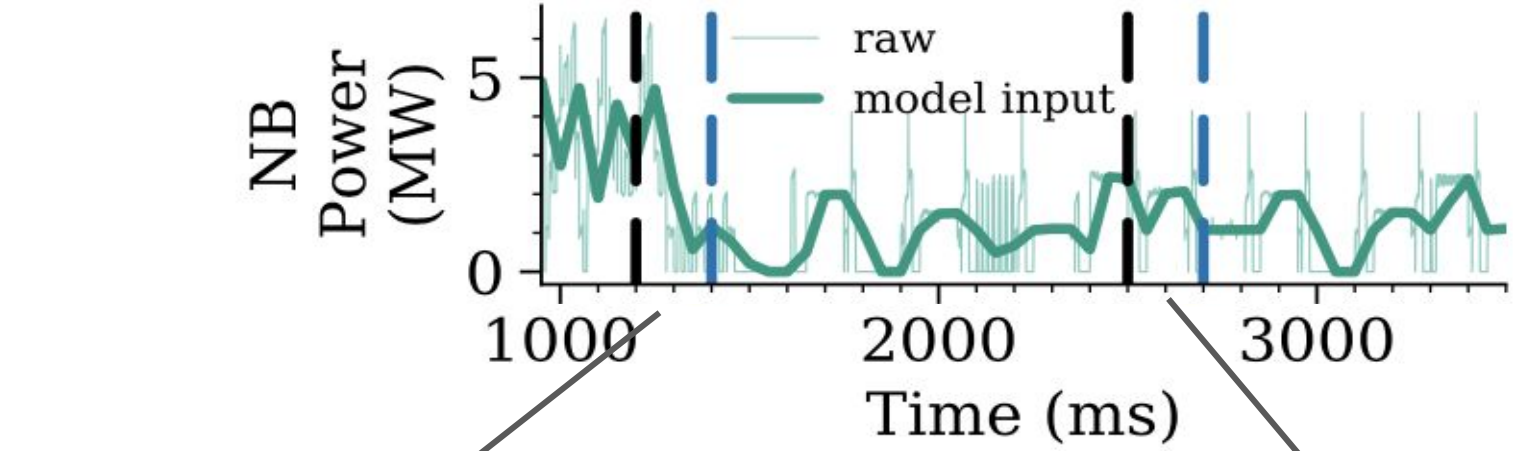
FUTURE STATE



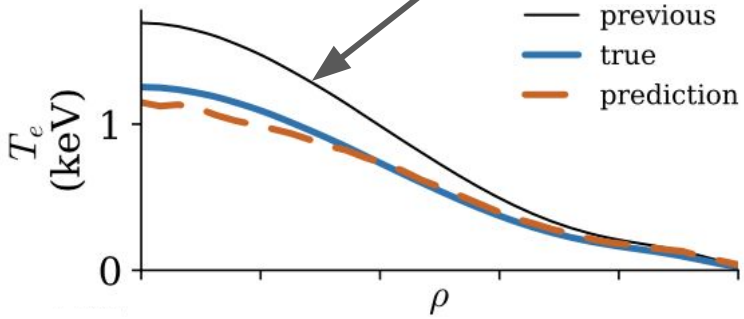
Abbate, Conlin 2021
Nucl. Fusion **61** 046027

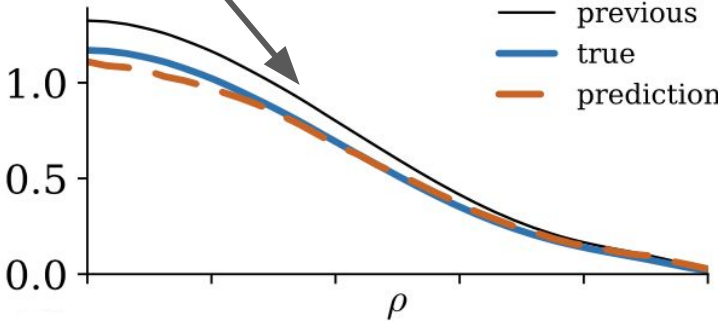
Model demonstration for simple NB-varying shot

Abbate, Conlin 2021
Nucl. Fusion **61** 046027



DIII-D shot 175970

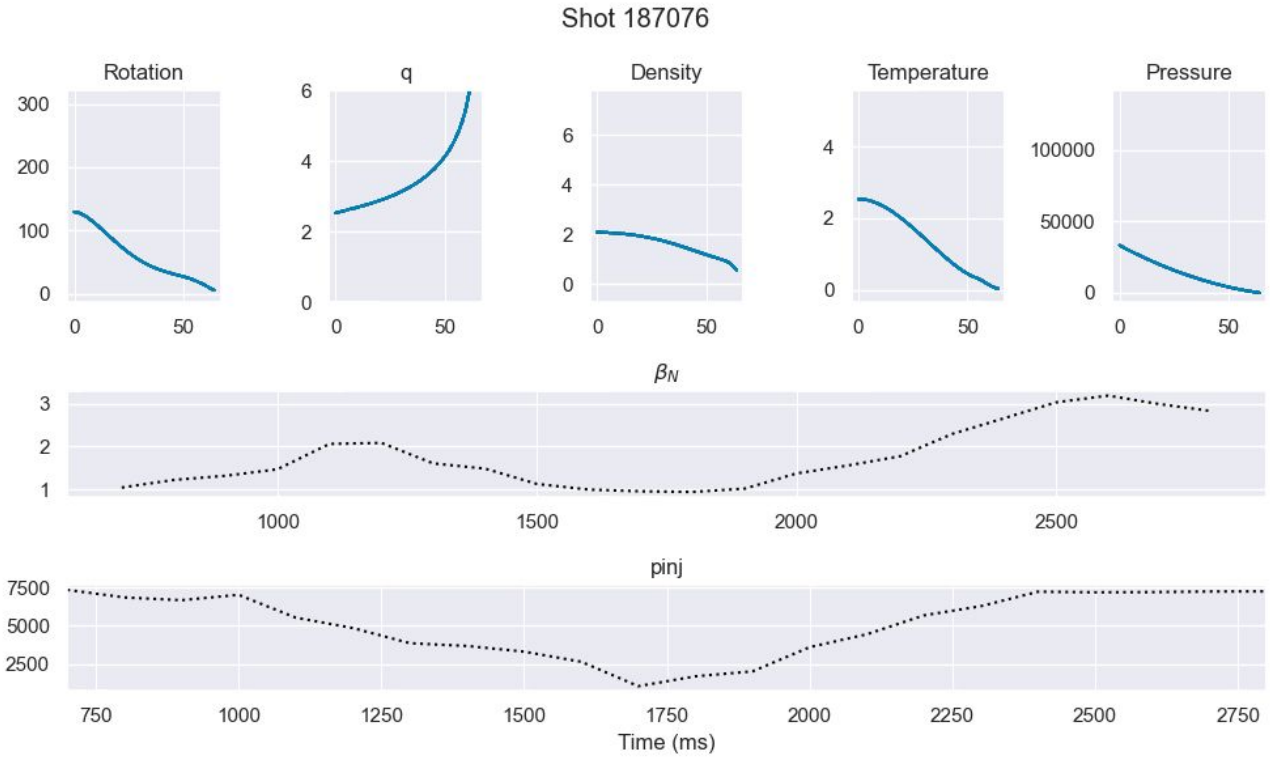




More complicated methods allow uncertainty estimate

CMU collaborators use ensemble of neural networks

- Increases accuracy
- Adds uncertainty



Poster:
Char, Thursday afternoon, UP11.00102

Tokamak control

Better control: human sets desired state, computer actuates

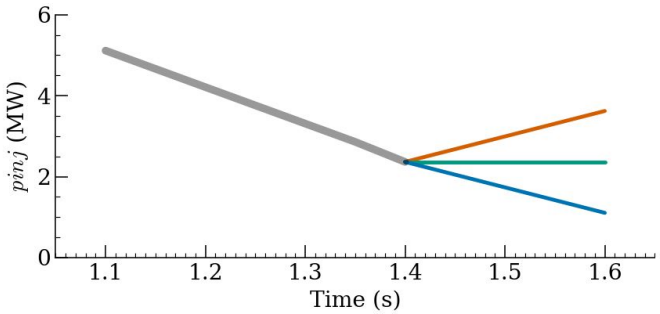
- **Simple operation: Reproducing existing scenarios / performing scans**
  - Surprisingly difficult
- **Scenario development: auto-optimize for reactor goals given constraints**
  - High confinement
  - Closer to steady state
  - Better stability
  - Same performance @ lower torque

Reminder: State-of-the-art is human trial-and-error (@ ~$1M / day)

Simple model-predictive control: Finite Set Controller

Abbate, Conlin, accepted, *JPP*

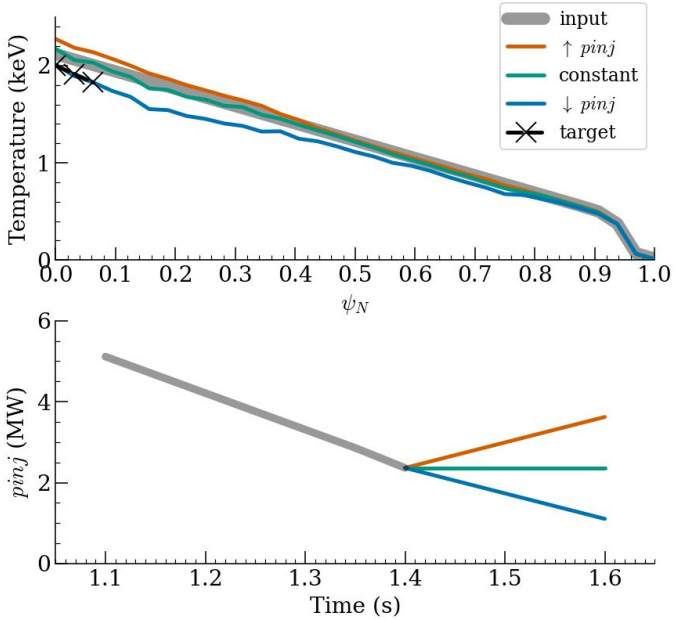
- Model takes in actuator history (gray),
  and a “proposal” for future actuator trajectory (colors)
  - Here we propose 3 Power INJected scenarios



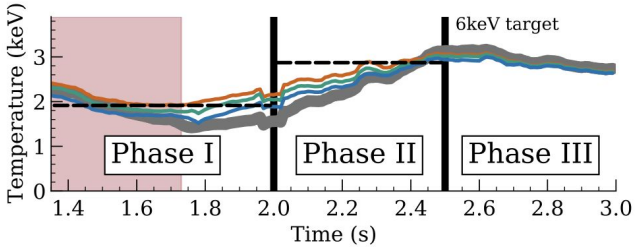
Simple model-predictive control: Finite Set Controller

Abbate, Conlin, accepted, *JPP*

- For each proposal, the model predicts the expected profile response at the end of the proposed trajectory
  - The proposal predicted to yield the closest state to a user-specified target is actuated
- Model takes in actuator history (gray), and a "proposal" for future actuator trajectory (colors)
  - Here we propose 3 Power INJected scenarios



Experiment: reasonable core-Te tracking via pinj control

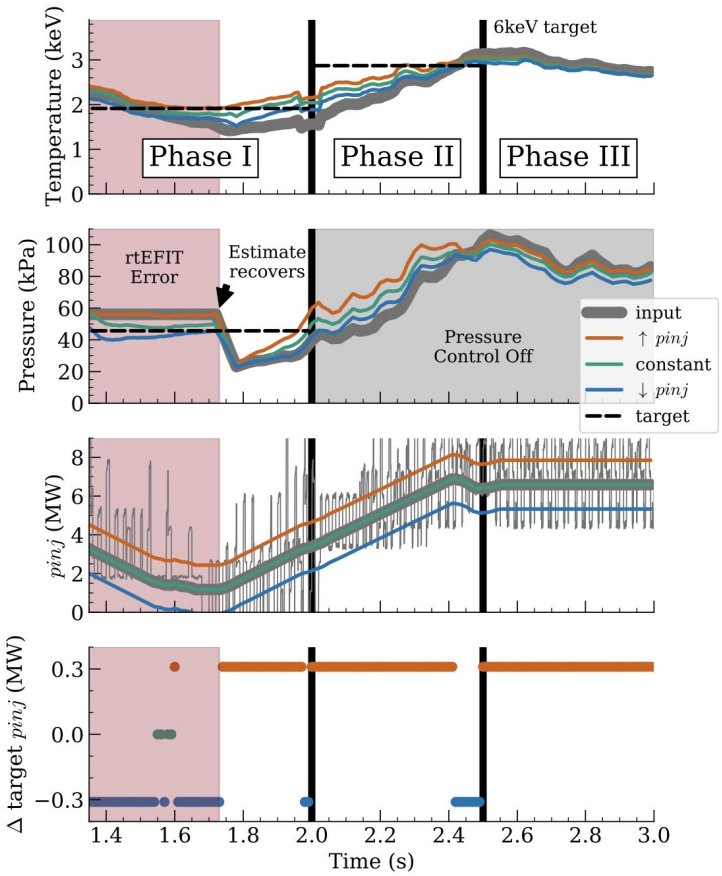


- Finite Set Control
- Preliminary test of core Te/P tracking by varying Power INJected from neutral beams

Abbate, Conlin, accepted, *JPP*

Experiment: reasonable core-Te tracking via pinj control

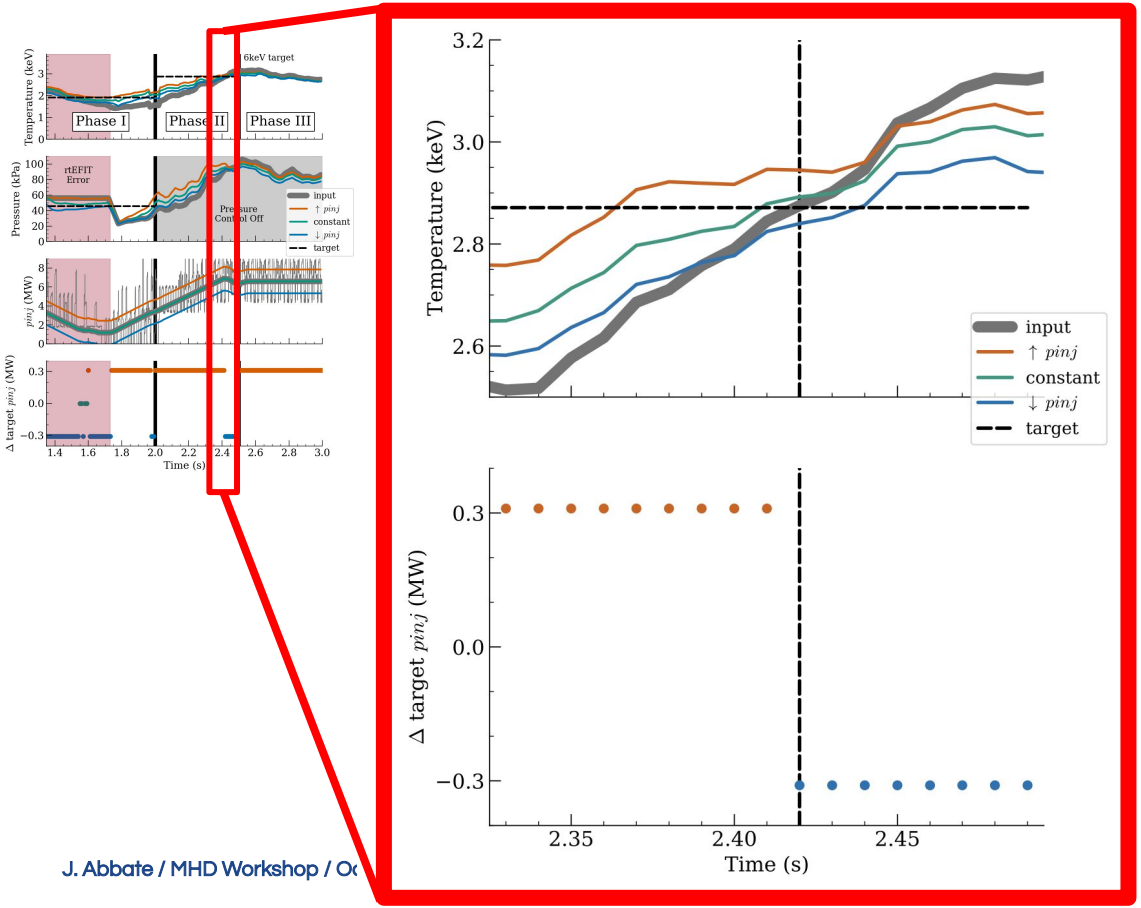
- Finite Set Control
- Preliminary test of core Te/P tracking by varying Power INJected from neutral beams
- Like P controller, when state below target beams increase



Abbate, Conlin, accepted, *JPP*

Experiment: reasonable core-Te tracking via pinj control

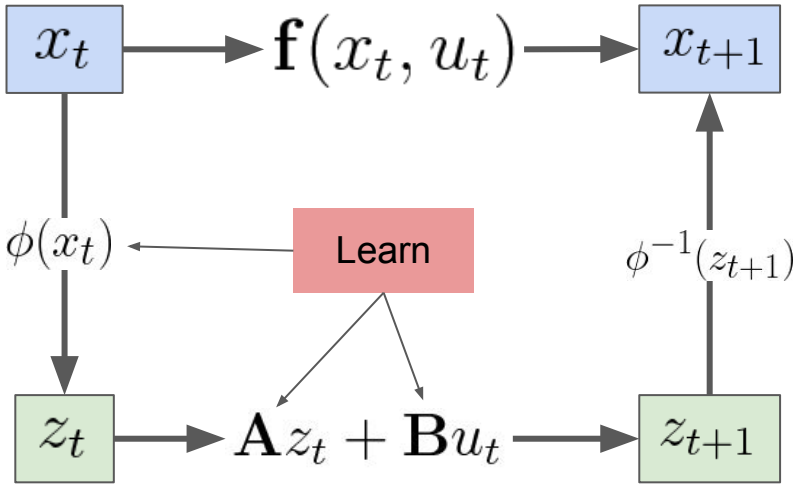
- Finite Set Control
- Preliminary test of core Te/P tracking by varying Power INJected from neutral beams
- Like P controller, when state below target beams increase
- Like PD controller, mitigates overshooting the temperature target by decreasing the injected power right as the target is achieved



Abbate, Conlin, accepted, *JPP*

Latent-space-linear model-predictive control

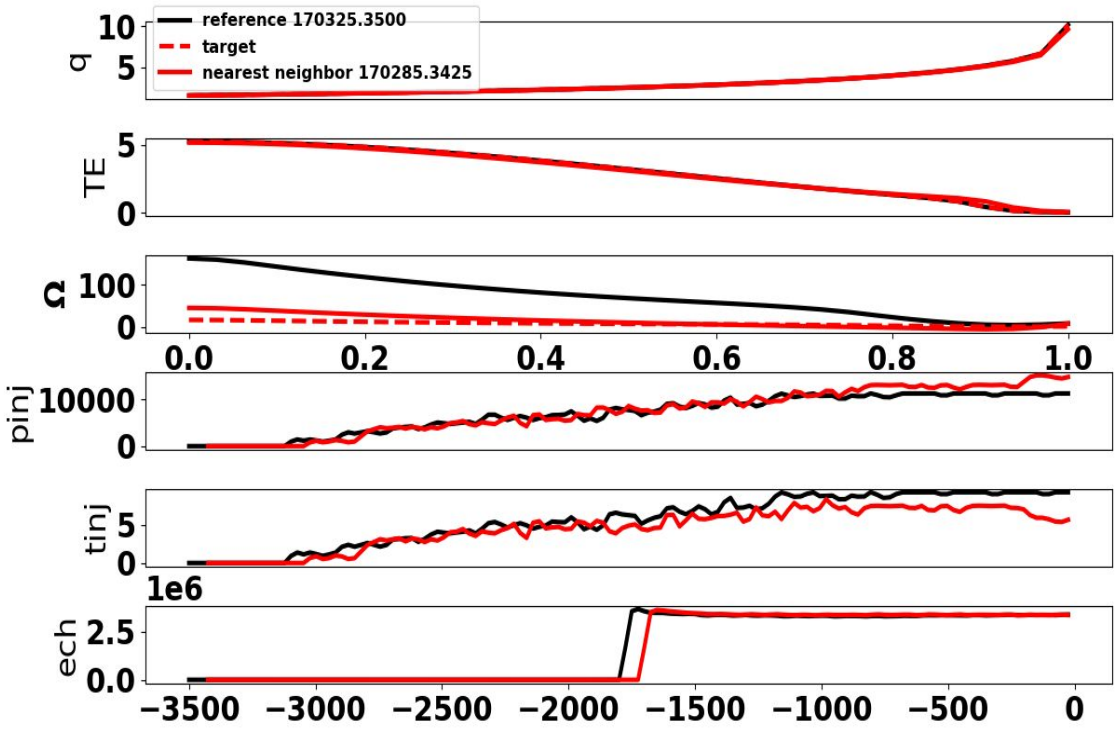
- Finite set control is limited
- One option: locally linearizing model for state-space control
  - However, dynamics aren't linear
- Possible solution: Map to latent space where dynamics *are* linear



- Great in theory, so far not so great in practice

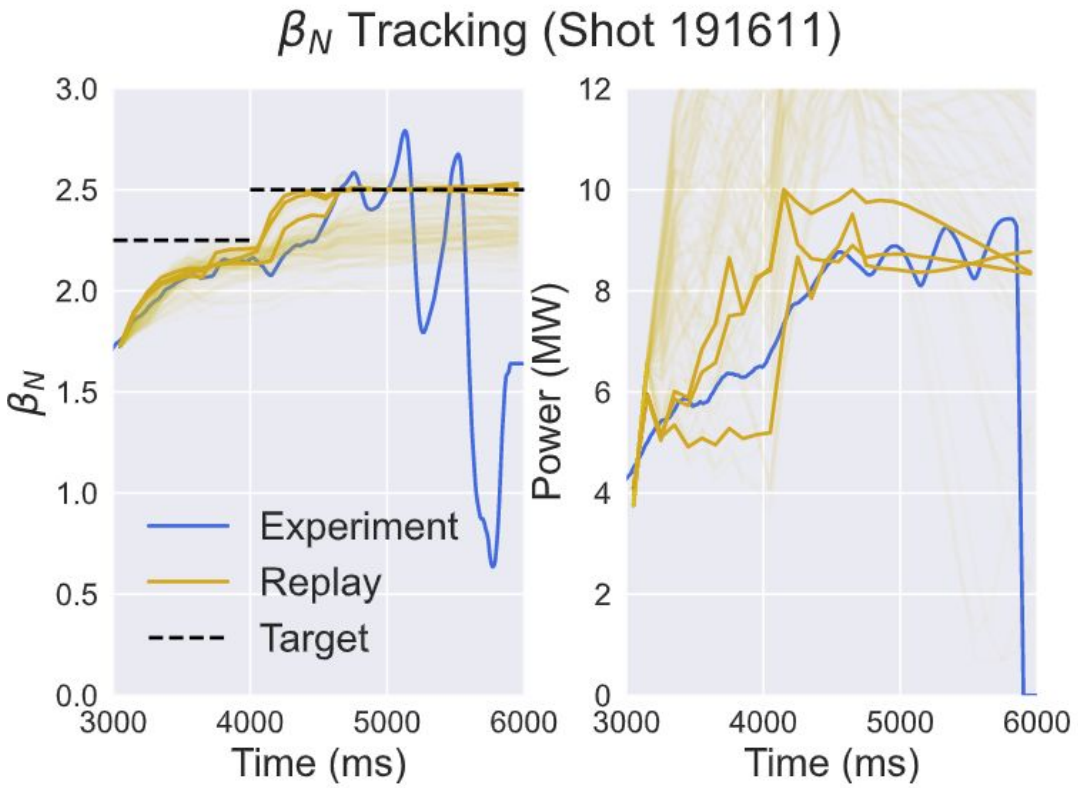
Actuators from target state: Nearest Neighbor

- **Like a physicist, find shot in database with similar parameters to reuse its actuator trajectory**
  - E.g. at right the nearest neighbor to 170325 but with 0 rotation has lower tinj / higher pinj
- **Too slow to run in realtime**



Actuators from target state: Reinforcement learning

- Train RL using predictor model, directly returns actuators given target and present state
- Experimental test: Error in PCS beam setup led to TM



Poster:
Char, Thursday afternoon, UP11.00102

Tokamak prediction and control via machine learning

- **Plasma predictions: traditional methods**
  - Reduced transport models require many inputs from future, only consider transport, not much better than heuristic guesses
- **Plasma predictions: data-driven machine learning**
  - We're attempting to learn dynamics based exclusively on experimental data
- **Tokamak control**
  - We can use data-driven models in controllers via variety of techniques, have tested 2 so far with variety of rundays upcoming to try more

- **Bonus takeaway:**
  - Beam control is error-prone, working on new PCS algo with Dan Boyer

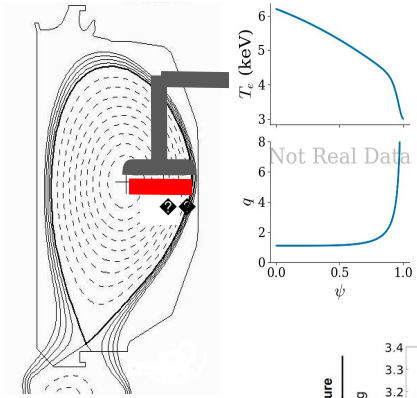
EXTRA SLIDES

Better physics models via physics sims

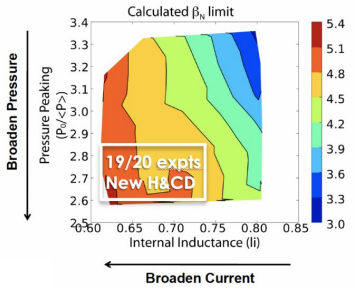
- **Data-driven models primarily lack**
  - Extrapolability
  - Causality
  - Data coverage
- **These are exactly the areas predict-first physics simulations succeed**

- **<u>Train model with both data *and* simulation information</u>**

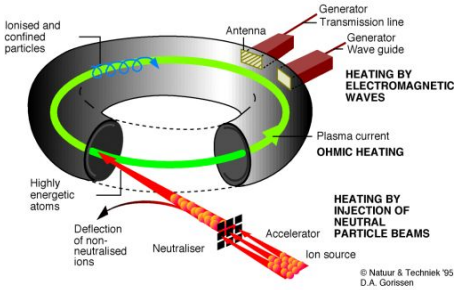
Building a “black box” plasma response model: data+sim



STATE



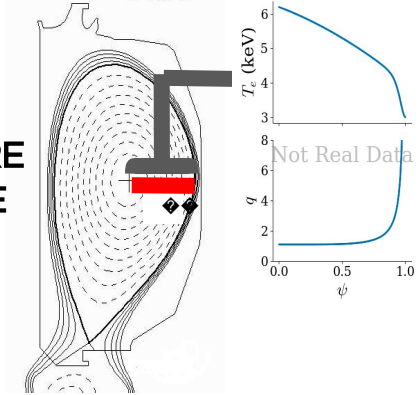
SIMS



ACTUATOR TRAJECTORIES

Learn f in
y=f(x)

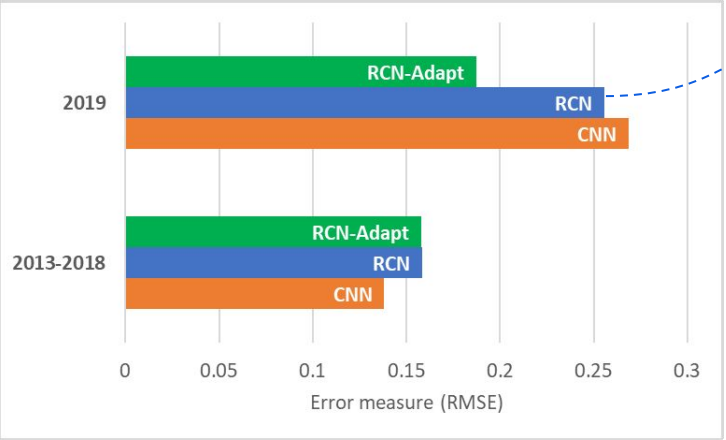
FUTURE STATE

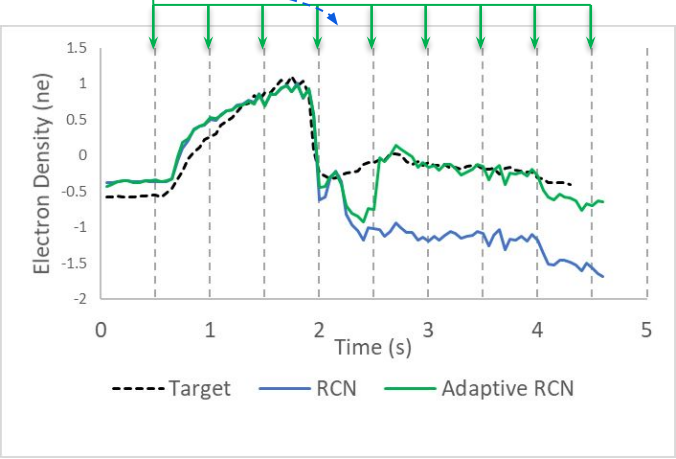


“But unlike sims, data-driven techniques don’t extrapolate”

	CNN/LSTM	RCN
Training	5h on GPU	45s on CPU
Adaptation	Difficult	Easy (100ms)

Real-time Adapting every 500ms





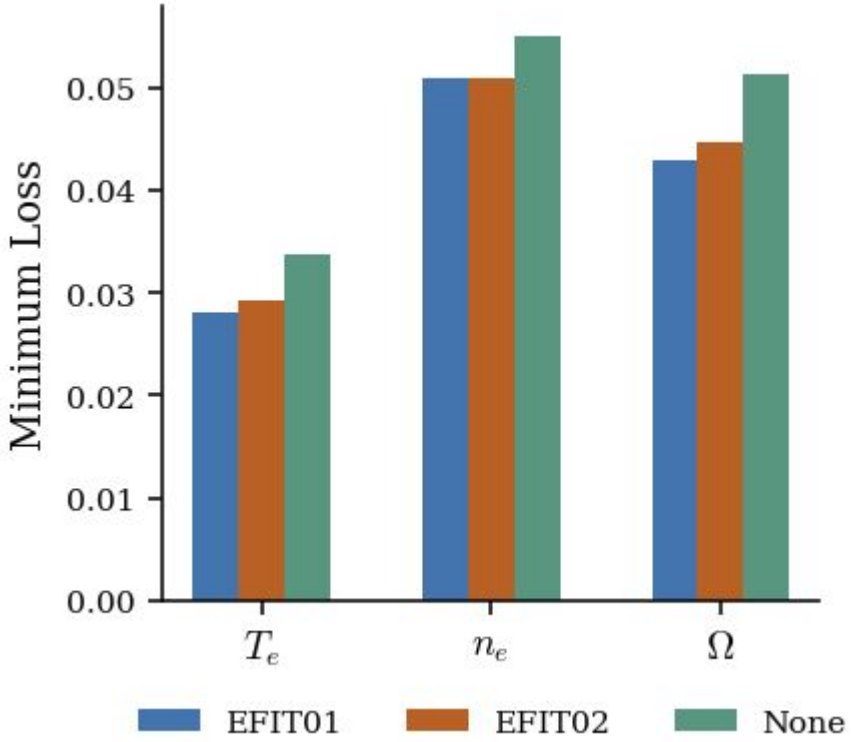
Jalalvand 2021 *IEEE TNNLS*

Signal importance: magnetics

Abbate, Conlin 2021
Nucl. Fusion **61** 046027

- Having EFIT data for pressure & q helps prediction of kinetic quantities
- Magnetics-only EFIT01 found to be better than Magnetics+MSE EFIT02

- Next step: train on CAKE



Seeking more intelligent control algorithm

For a controlled system where $x_{t+1} = Ax_t + Bu_t$

Linear-Quadratic Regulator (LQR), Model-Predictive Control (MPC), etc. find control trajectory (u_1, ..., u_k) to achieve a target state (x_k). **If A and B are constant**…

$$x_{t+1} = A\mathbf{x_t} + B\mathbf{u_t}$$

$$x_{t+2} = Ax_{t+1} + B\mathbf{u_{t+1}}$$

$$= A(A\mathbf{x_t} + B\mathbf{u_t}) + B\mathbf{u_{t+1}}$$

$$= A^2\mathbf{x_t} + AB\mathbf{u_t} + B\mathbf{u_{t+1}}$$

$$x_{t+3} = A^3\mathbf{x_t} + A^2B\mathbf{u_t} + AB\mathbf{u_{t+1}} + B\mathbf{u_{t+2}}$$

…

$$\boxed{x_{t+N} = A^N\mathbf{x_t} + A^{N-1}B\mathbf{u_t} + \ldots + B\mathbf{u_{t+N-1}}}$$

The state trajectory is a **linear** combination of

(1) the initial state

(2) the actuator trajectory

Given the initial state, optimize the future state over all possible actuator trajectories

Sigma metric for model accuracy

$$\sigma_{T,i} = \frac{\sqrt{\frac{1}{N}\sum_j \epsilon_j^2}}{\sqrt{\frac{1}{N}\sum_j T_{x,j}^2}},$$

XPTOR + TGLF/GLF23 for 25 DIII-D L-, 33 DIII-D H-discharges

Error	TGLF	GLF23
$\bar{\sigma}_{Ti}$	14%	16%
$\bar{\sigma}_{Te}$	15%	13%

JETTO + QuaLiKiz for JET shot 75225

Zone	σ_{T_i}	σ_{T_e}	σ_{n_e}	$\sigma_{V\,\mathrm{tor}}$
$\rho_{\mathrm{norm}} = [0.4–0.8]$	7.9%	8.3%	7.7%	7.3%
$\rho_{\mathrm{norm}} = [0.1–0.8]$	24.1%	5.0%	13.2%	18.8%

RAPTOR + QLKNN4D-kin for JET shot 73342

Case	σ_{T_i}	σ_{T_e}	σ_{n_e}
Nominal	10.0%	6.9%	2.7%
BC reduced by 10%	13.1%	10.6%	11.0%
BC increased by 10%	4.0%	6.5%	7.1%

Kinsey, Staebler, Waltz Phys. Plasmas 15, 055908 (2008)
J Citrin et al Plasma Phys. Control. Fusion 59 124005 (2017)

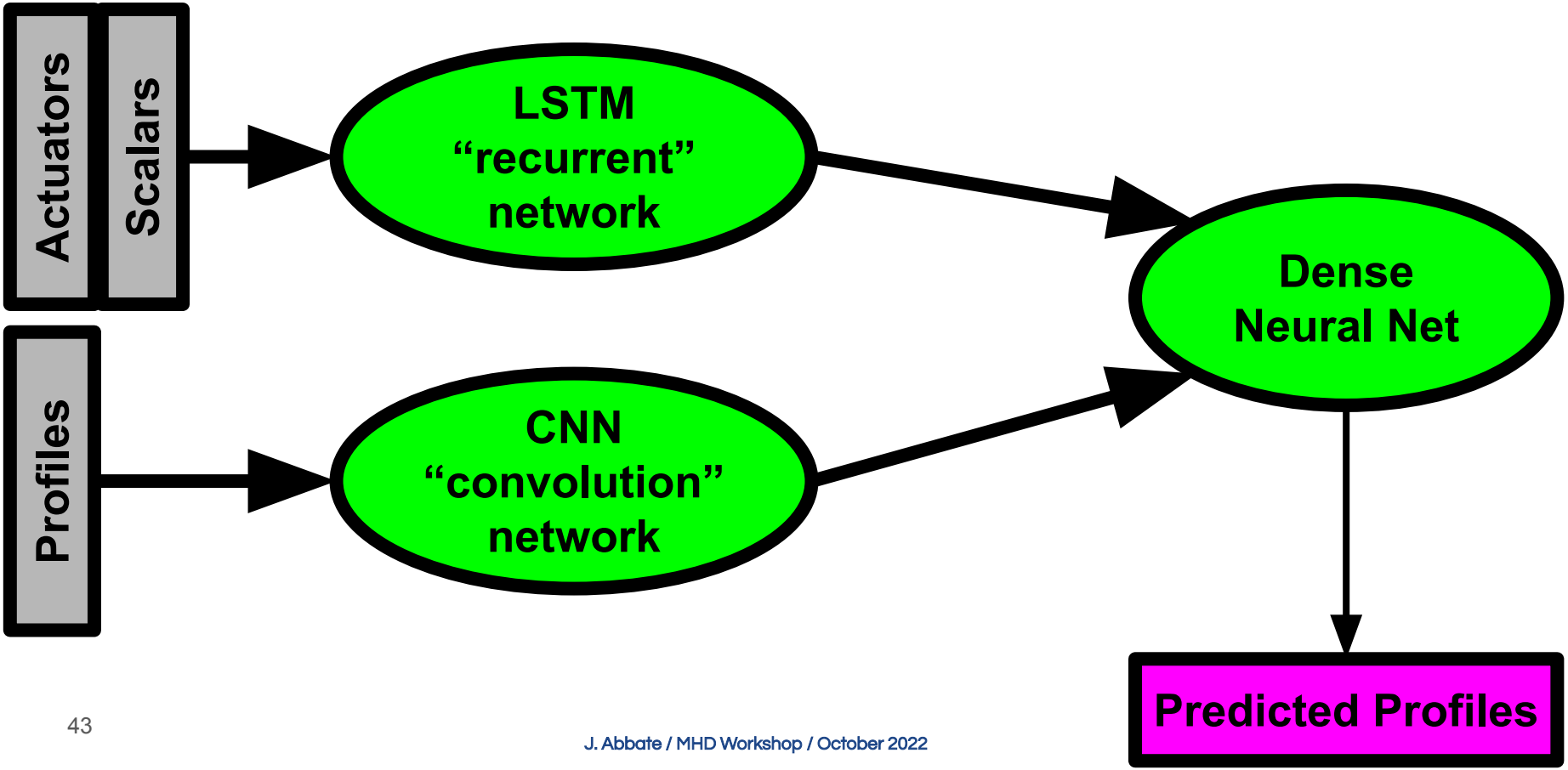
Model training details

- **DIII-D data only**
  - ~200,000 samples from ~5,000 shots from 2010-2018
- **Excludes**
  - disruptions
  - low-beta shots
  - non-standard coil configuration (thanks to C. Paz-Soldan)

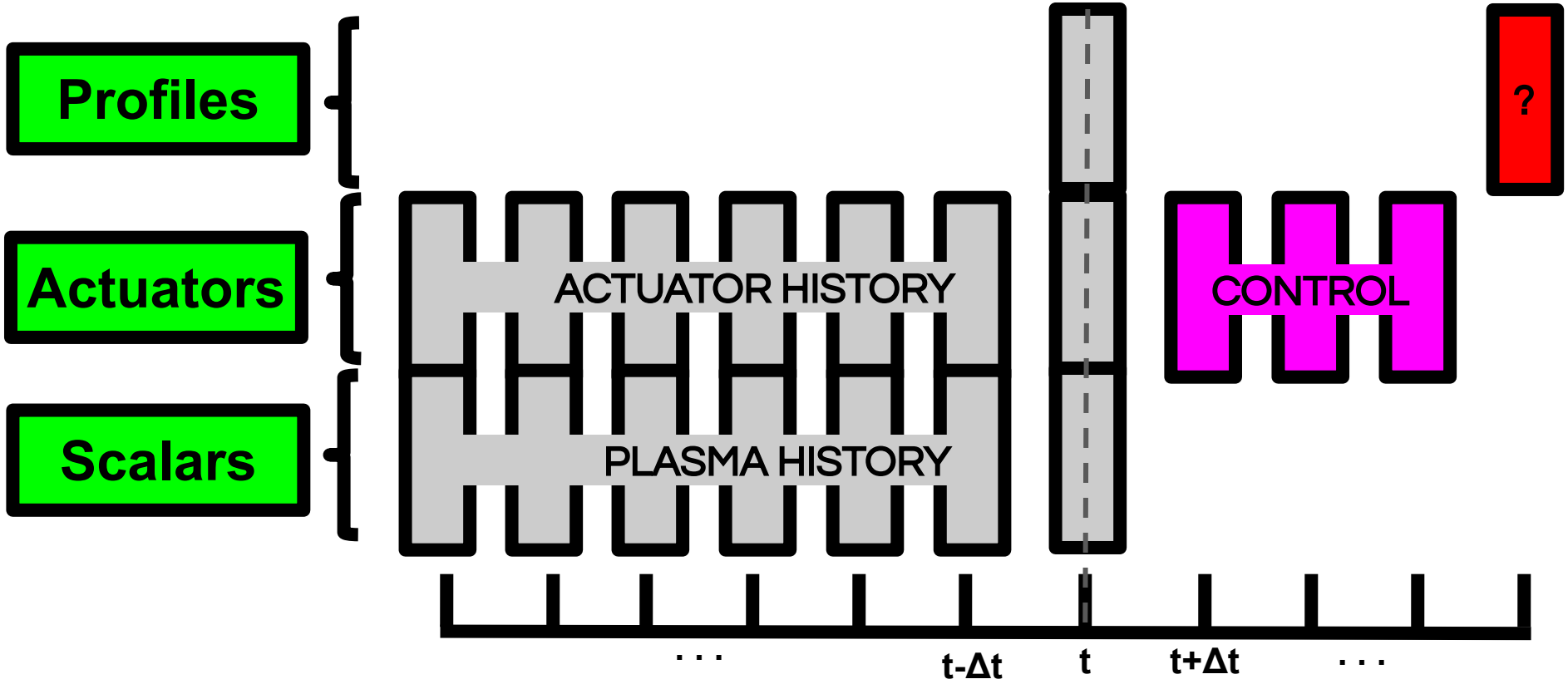
Inputs:

- **Profiles**
  - Te
  - ne
  - Ti
  - **Ω**
  - q (iota)
- **Scalars**
  - triangularity (top+bottom)
  - elongation
  - volume
  - internal inductance
  - line-averaged density
- **Actuators**
  - pinj
  - tinj
  - target current
  - target density

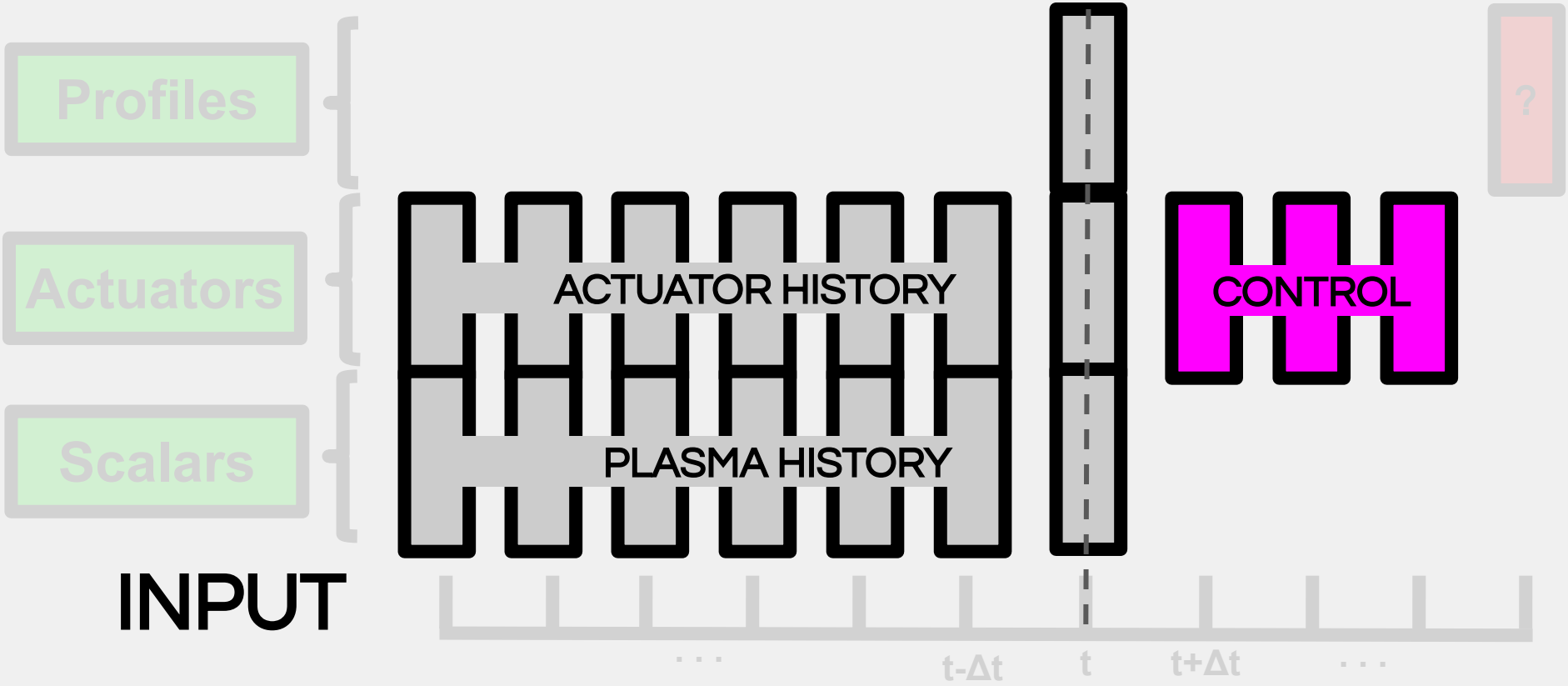
Model Architecture



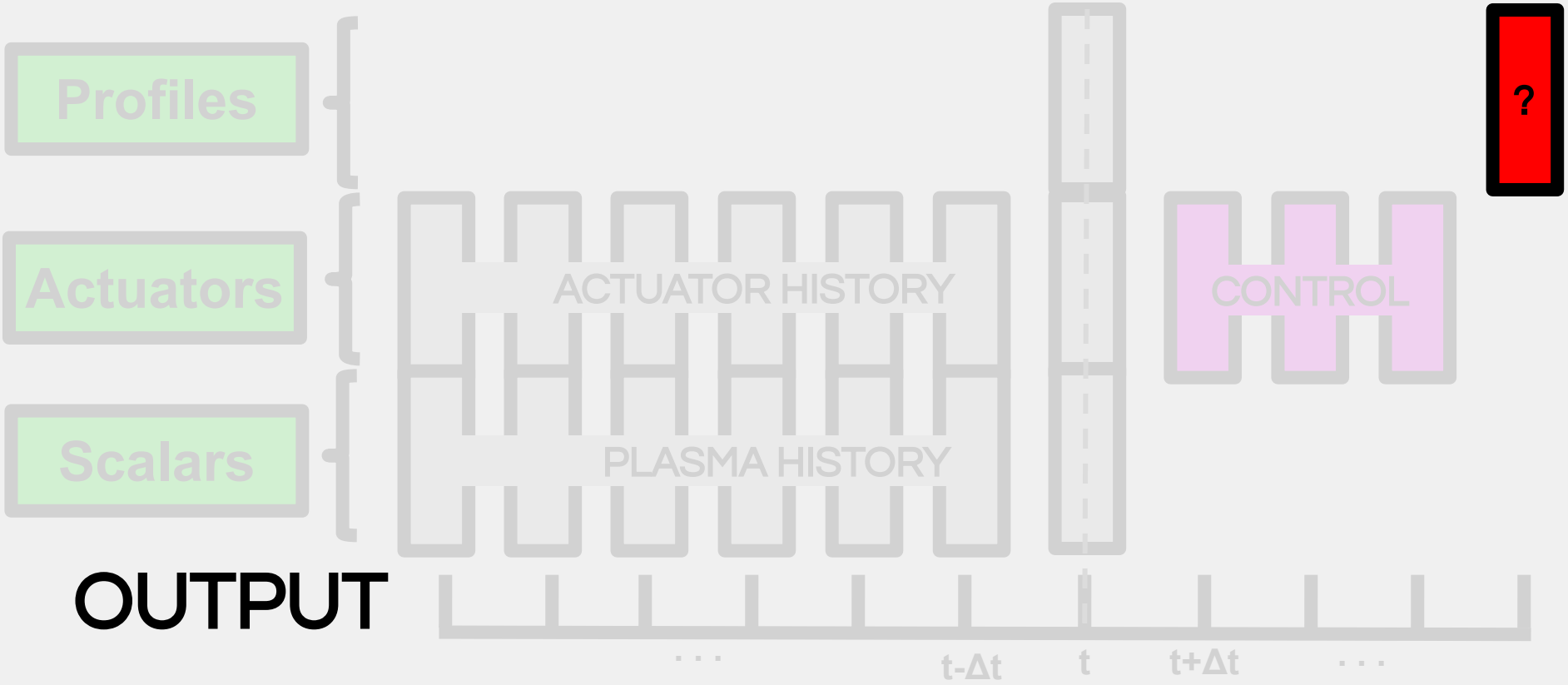
Model Input / Output



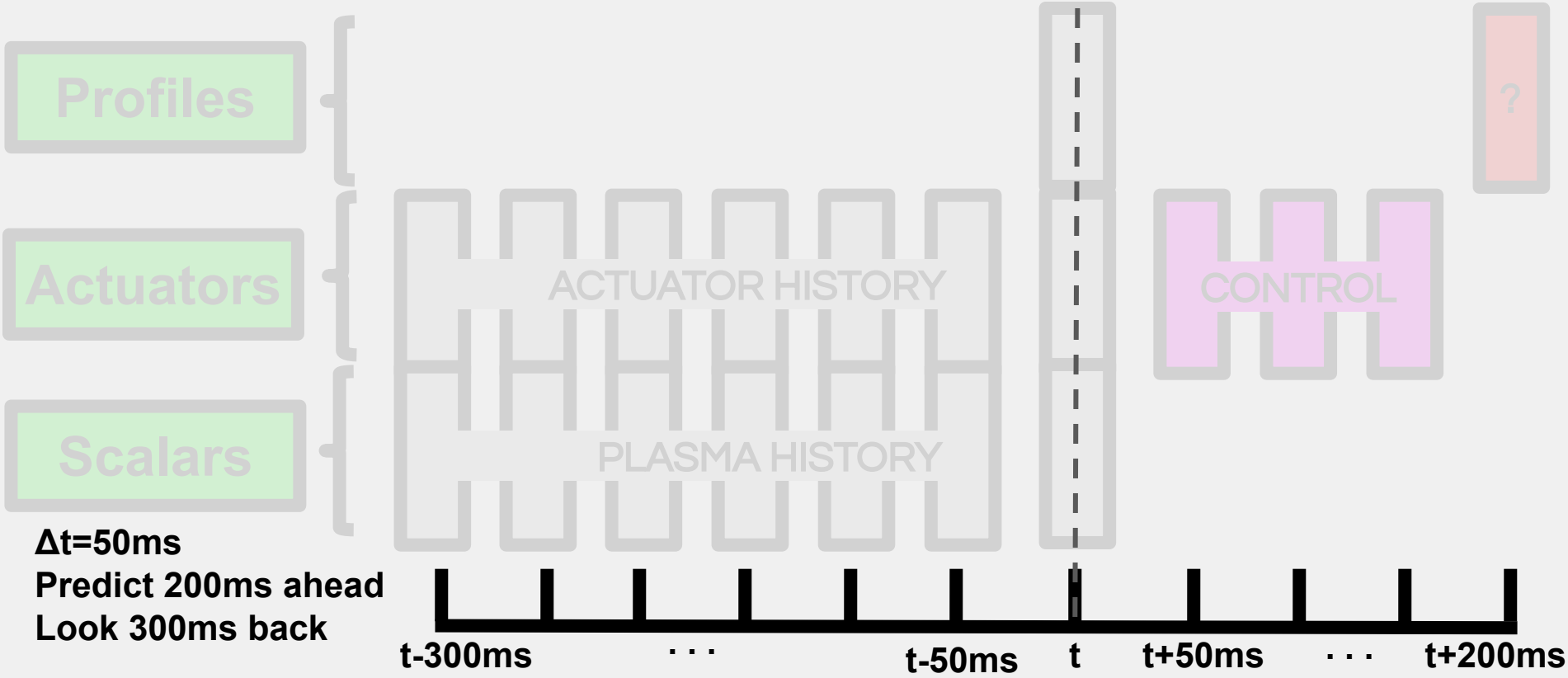
Model Input / Output



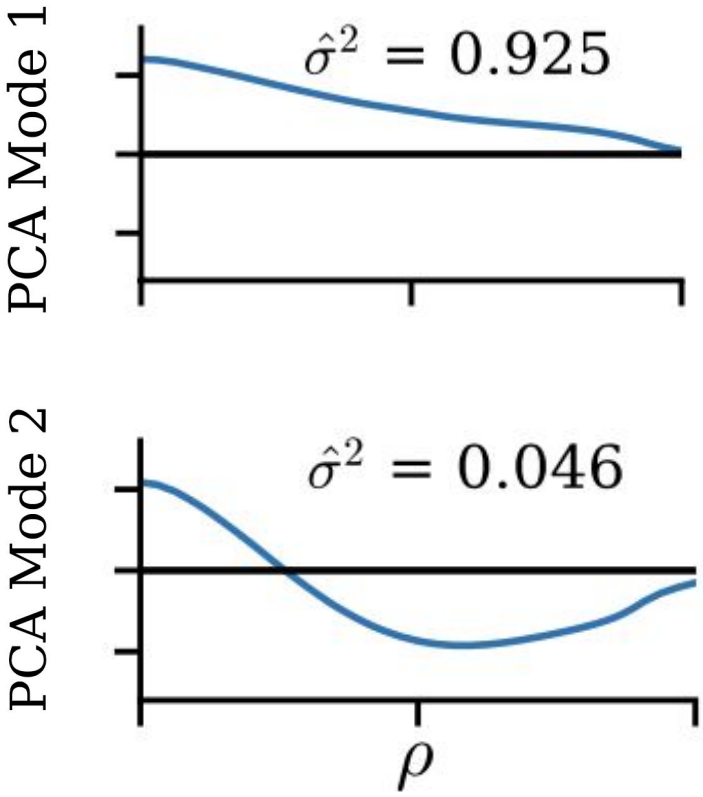
Model Input / Output

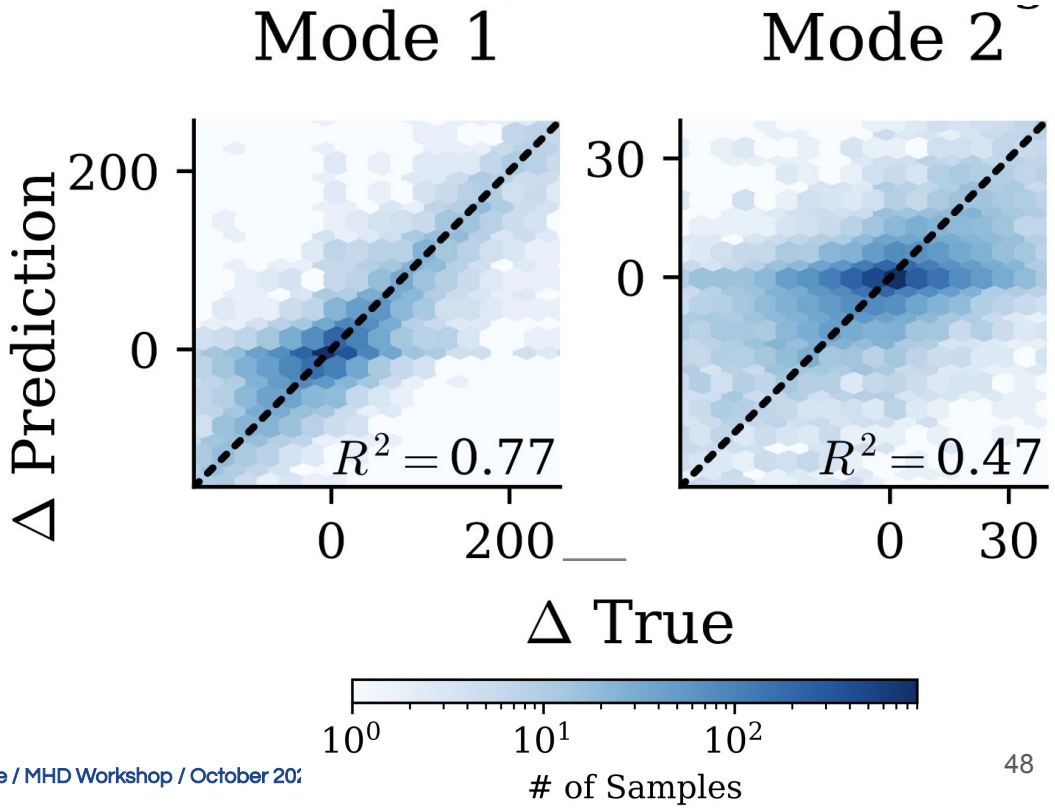


Model Input / Output

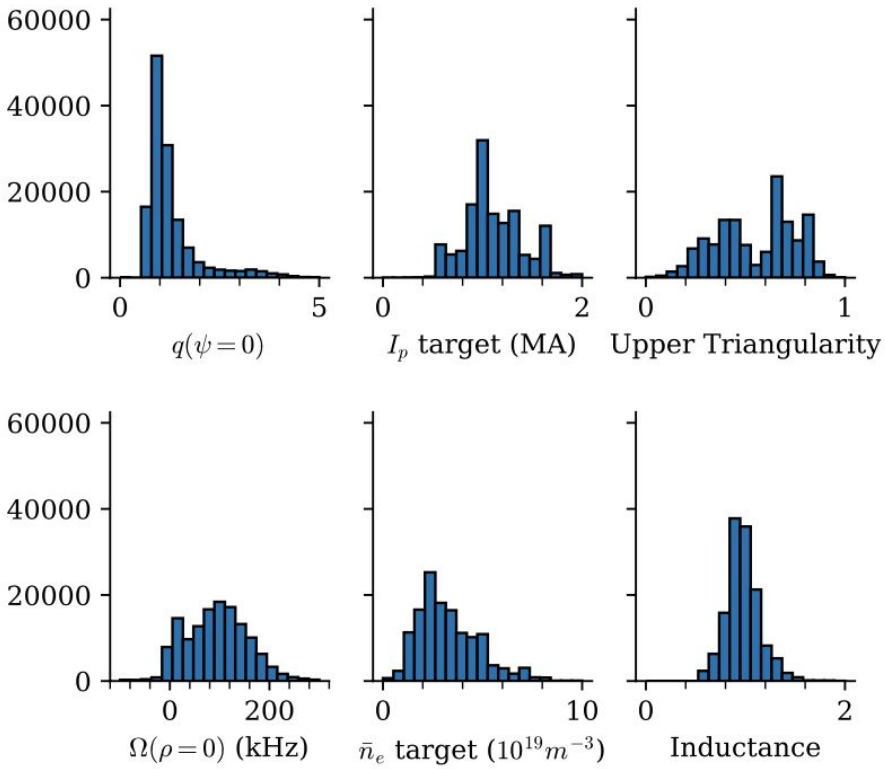


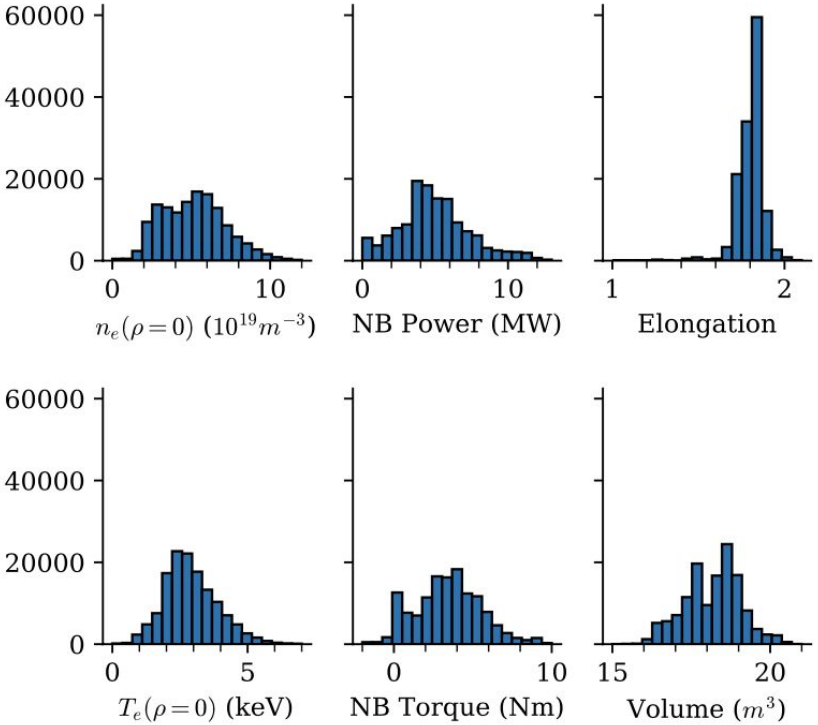
Globally too: Model captures general shape, though noisy



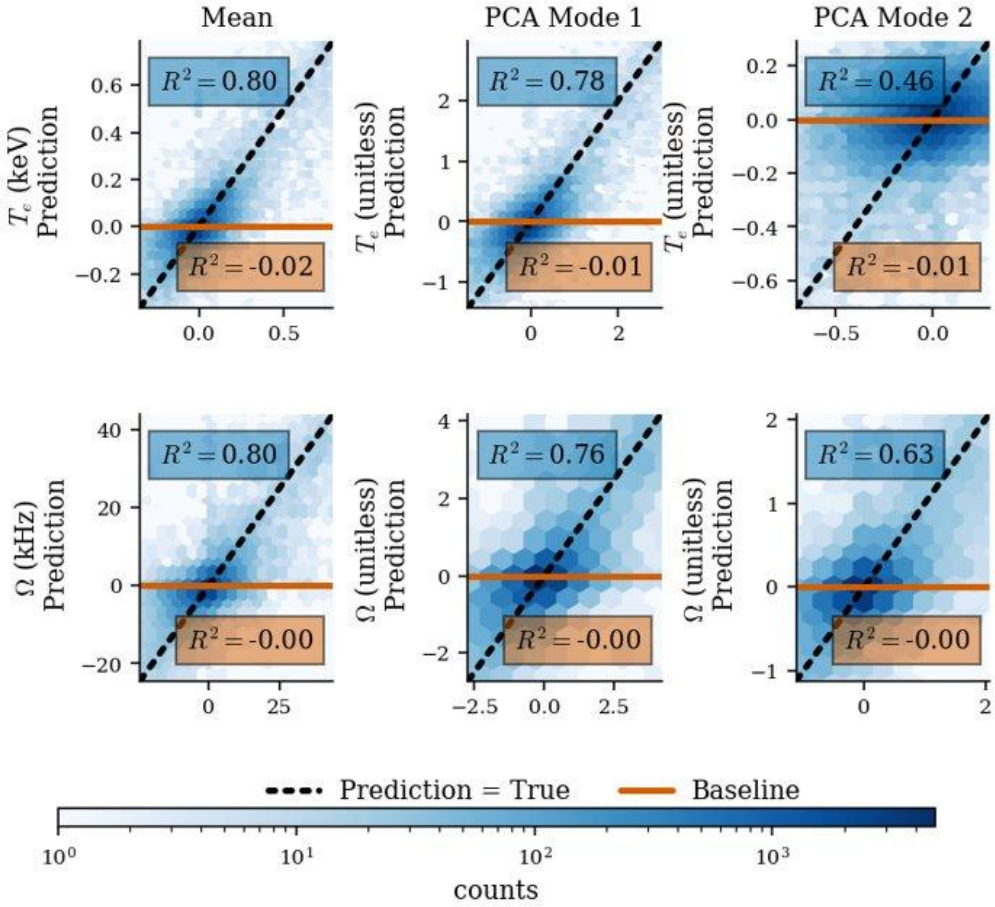


Train set distribution

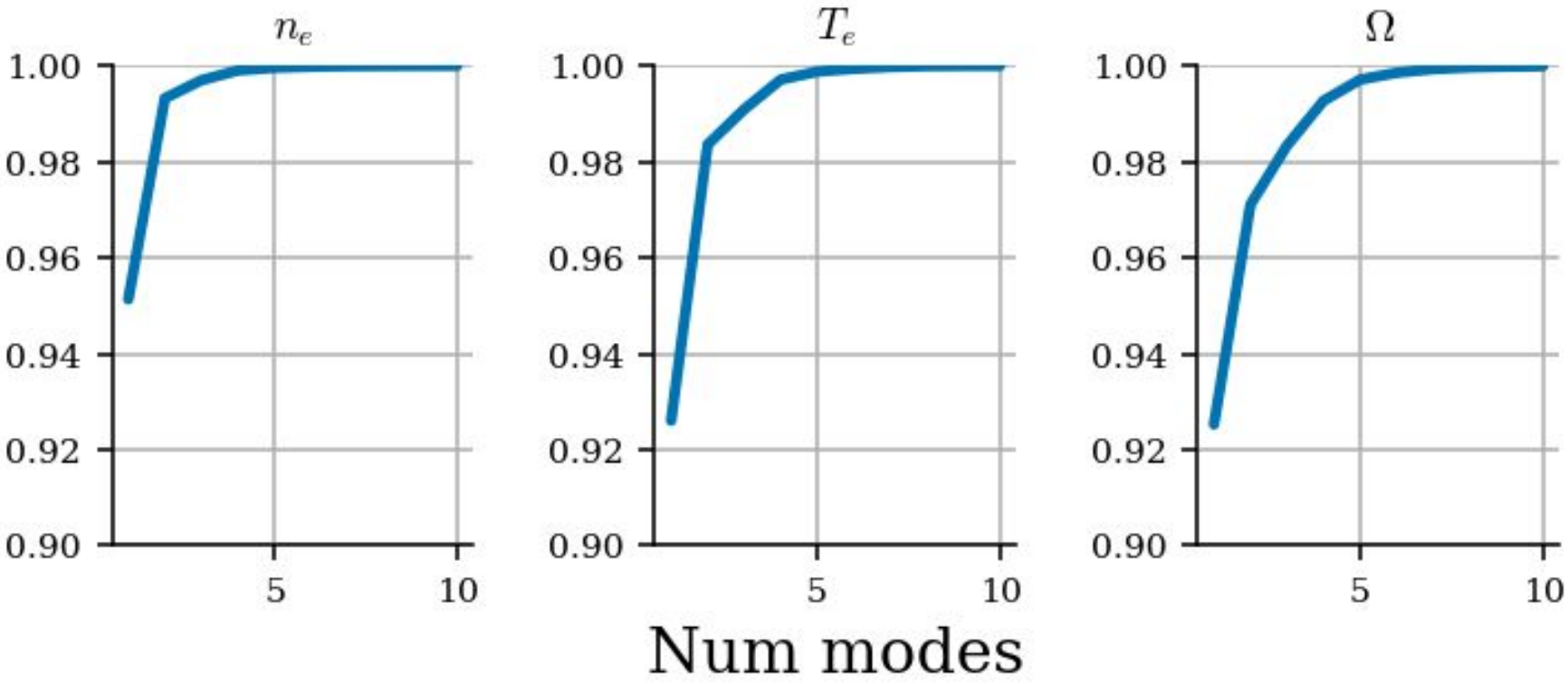




Average Error

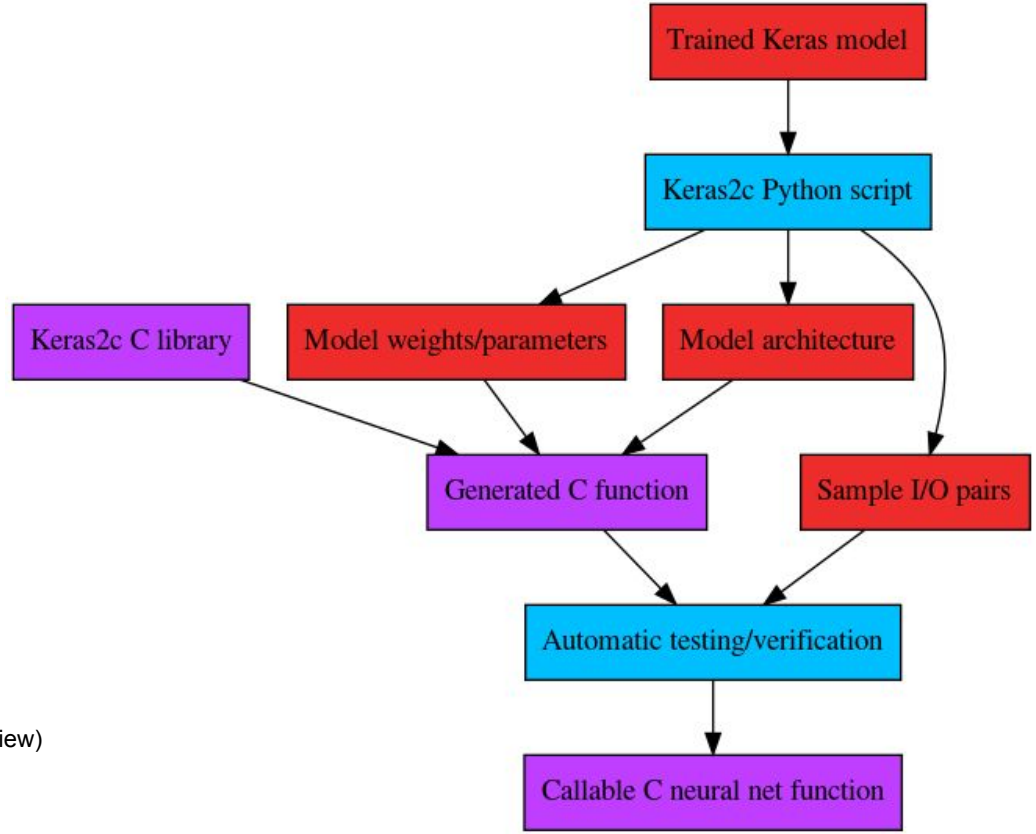


PCA Profiles: Explained Variance



Keras2C (R. Conlin)

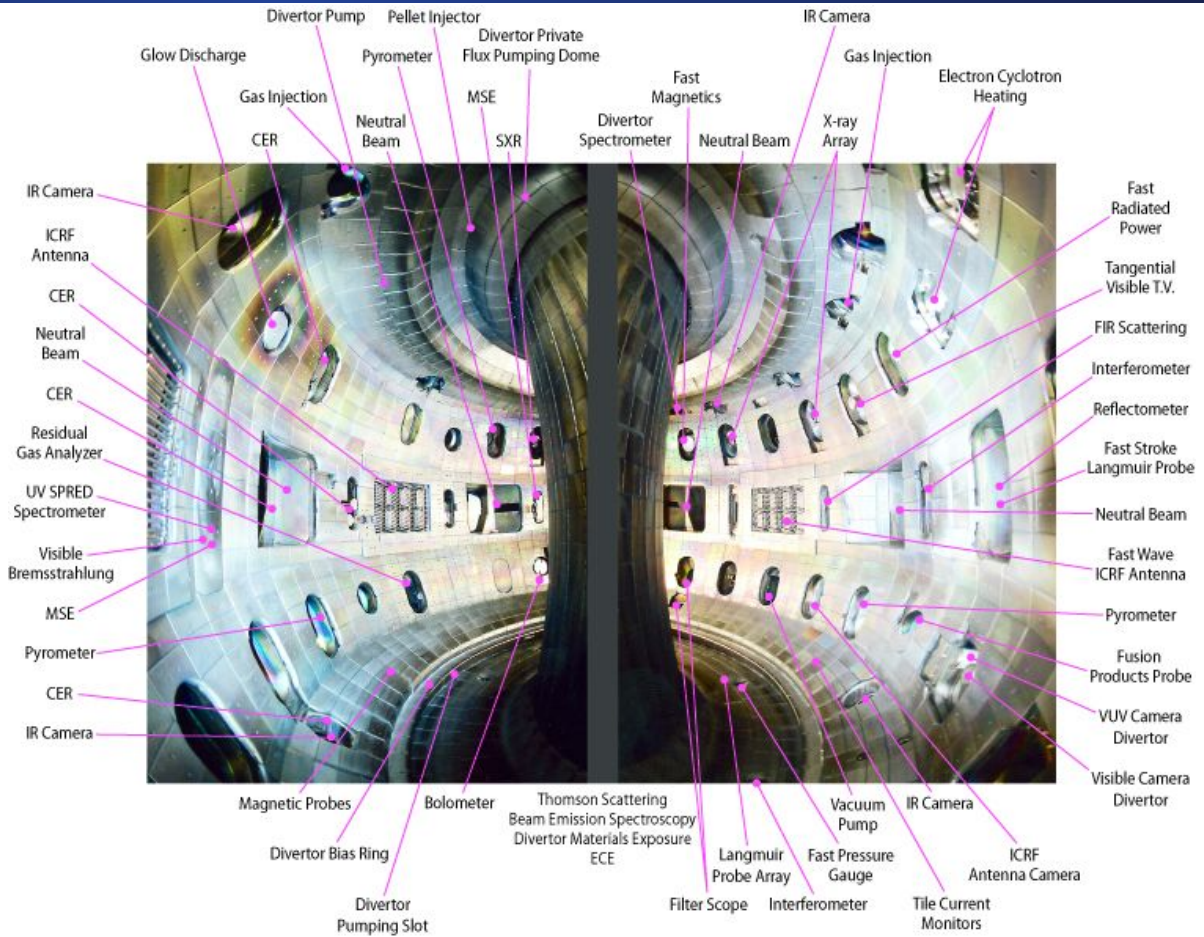
- Script/Library for converting Keras neural nets to realtime-ready C functions
- Also used for FRNN realtime disruption predictor
- Supports full range of operations and architectures



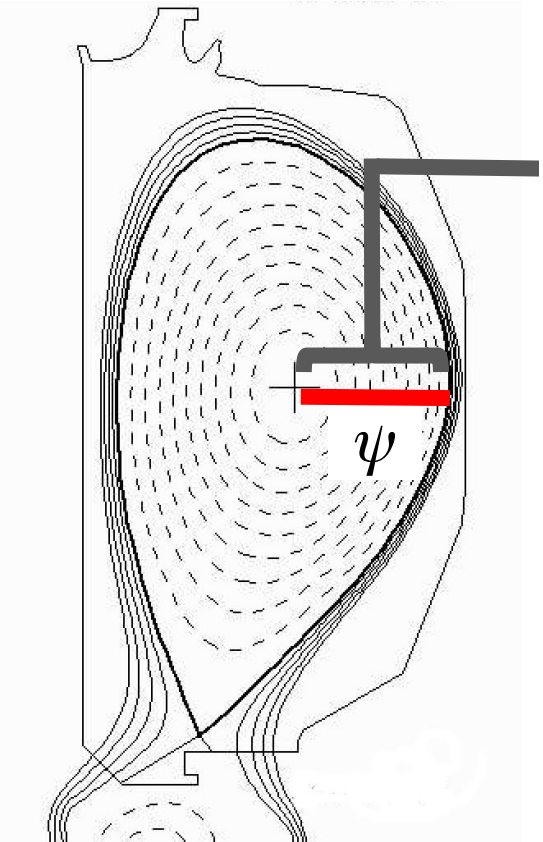
R. Conlin et al 2020 Engineering Applications of Artificial Intelligence (in review)
See R. Conlin's poster: PP12.00039

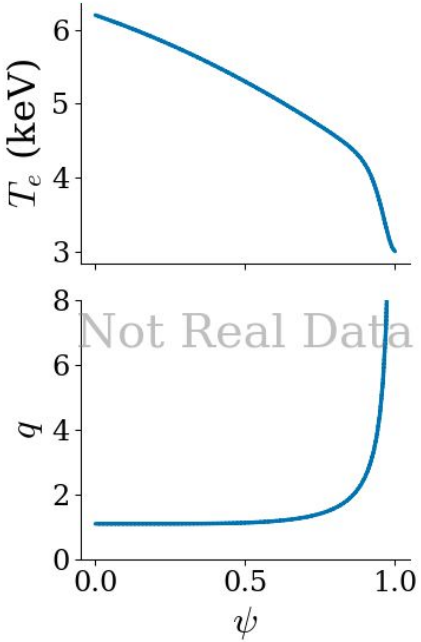
How do we describe the tokamak plasma state?

Massive amounts of data on variety of timebases from myriad diagnostics



How do we describe the tokamak plasma state?

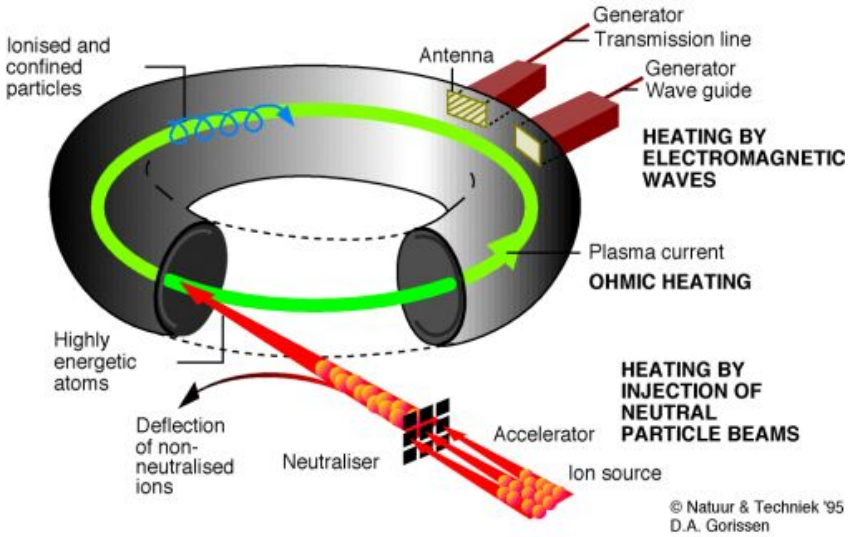




In transport, full state of plasma approximated by ~1D profiles:

- Current (q)
- Electron temperature (T_e)
- Electron density (n_e)
- Ion temperature (T_i)
- Rotation (Ω)
- Flux surface shapes (κ, Δ, δ)
- Impurity content (Z_{eff})
- Toroidal field (B_t)

How do we change the plasma state?



- Neutral beams
- RF waves
- Central solenoid ramp-rate
- Gas valves and gas pellets
- Toroidal field coils
- Poloidal field coils to set shape
- 3D coils to perturb toroidal symmetry (e.g. can induce ELMs)