MODEL PREDICTIVE CONTROL OF PLASMA PROFILES AT DIII-D

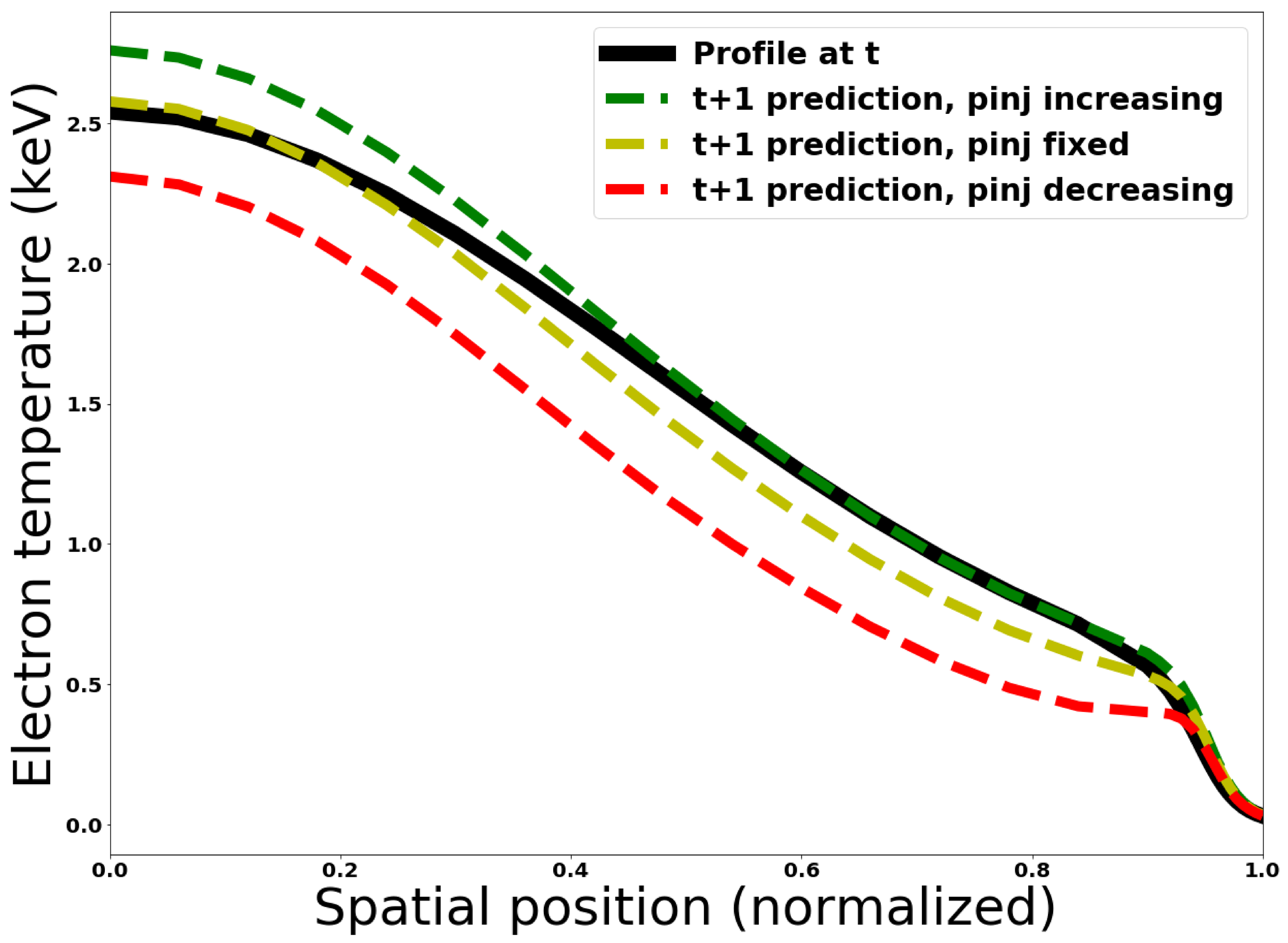
R. CONLIN, J. ABBATE, K. ERICKSON, and E. KOLEMEN

DIII-D NATIONAL FUSION FACILITY

PPPL PRINCETON PLASMA PHYSICS LABORATORY

Introduction

- Want to control the full state of the plasma in a tokamak to maximize performance
- Full state of plasma determined by profiles:
  - pressure (P)
  - current (J)
  - electron temperature and density (T_e, n_e)
  - ion temperature and density (T_i, n_i)
  - rotation
  - safety factor (q)
- Available actuators:
  - Neutral beam power injected
  - Neutral beam torque injected
  - Total plasma current
  - Gas puffing
  - ECH heating
- A neural network model has been developed to predict profile evolution in real time[1]
- Can simulate the effects of multiple different inputs to select actions to reach desired state
- Can also simulate multiple actuators acting at the same time to capture interplay between them



Method

- Operator chooses desired profiles before a shot, along with a set of weights to to rank relative importance
- Actuator signals to simulate can be either specified by the operator or determined in real time
- Select actuators to minimize cost function:

$$u(t) = \operatorname{argmin}_u \sum_{i=1}^{N_{prof}} \sum_{j=1}^{N_\psi} \left\{ w_{ij} \left(y_{ij}^{req}(t+1) - y_{ij}^{pred}(t+1,u,y) \right)^2 + \lambda_i \left| \tilde{y}_i(t+1,u,y) \right| \right\}$$

u: actuator action
y: profile (requested/predicted)
$\tilde{y}$: estimated uncertainty of prediction
w, λ: user selected weights

One step lookahead

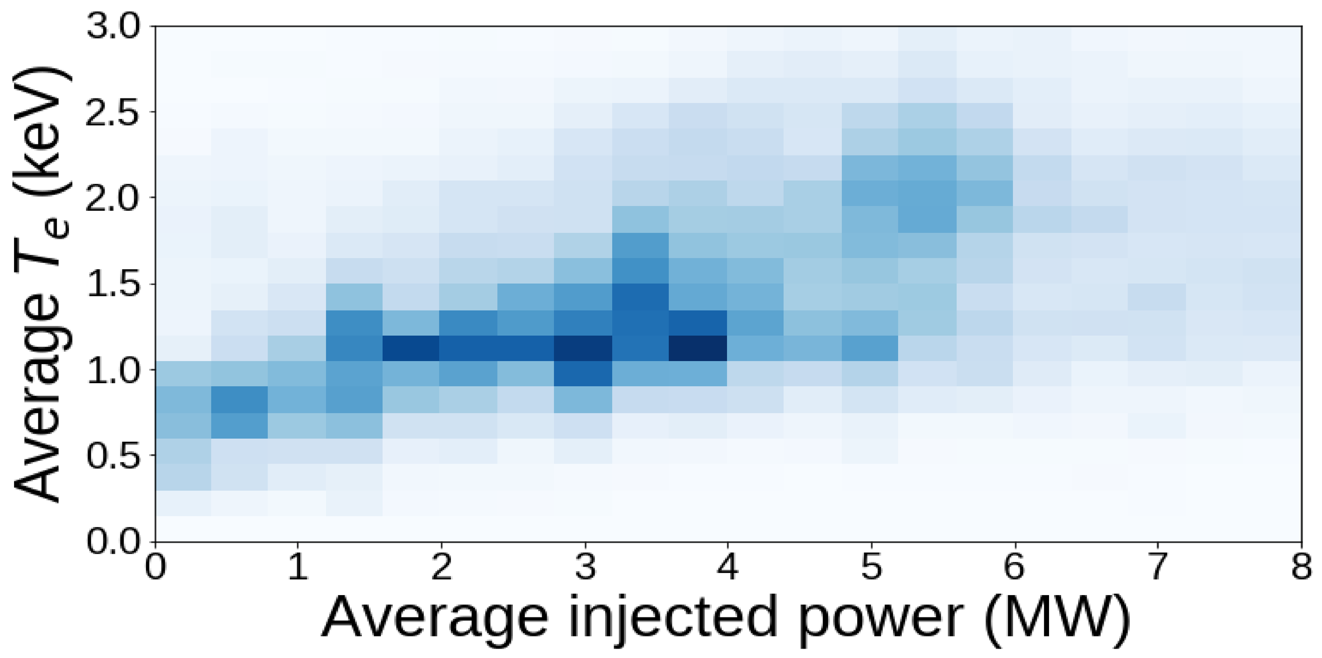
- Only optimizes over the next timestep
- Easy to implement, very fast
- Best for small, incremental changes to profiles

Multi-step lookahead/Tree search

- Can look farther into the future by feeding back on predictions
- Allows actions that are locally suboptimal, but lead to better performance over long term
- Uncertainty increases with longer lookahead
- Number of possible actions grows exponentially, “curse of dimensionality”
- Monte Carlo tree search (MCTS)[2] and heuristic search (A*)[3] to explore efficiently
  - MCTS samples randomly from possible paths
  - A* searches in directions that have the lowest expected cost
  - Together perform similar to simulated annealing for global non-convex optimization – search in locally best direction, with noise to avoid local minima

Sampling possible actions

- Proposed actions to evaluate at each step can be pre-selected
  - Ex: try +10%, -10%, constant each time
- Can also be chosen dynamically using linear correlations
  - Ex: to increase T_e by ~0.5 keV, increase injected power by ~1 MW



Estimating error/uncertainty

- Train simpler linear model to estimate expected prediction error in different regimes[4]
  - Ex: uncertainty should be higher in regimes where there is less training data

Timing / PCS integration

- Each prediction takes ~100 us
- Can sample ~100 possible actions per cycle before deciding on control action
- Also very easy to parallelize for multiple threads

Future work

- Initial test scheduled for Oct 31 using one step lookahead and preselected actuator proposals
- Still need uncertainty estimates
- If initial test successful, additional testing to try multi-step lookahead and dynamic actuator selection

References

1) Joe Abbate et. al “Machine learning plasma profile prediction for model-predictive control at DIII-D” TP10.00136
2) Cameron B Browne et al. “A survey of monte carlo tree search methods”. In: IEEE Transactions on Computational Intelligence and AI in games 4.1 (2012)
3) Richard E Korf. “Real-time heuristic search”. In: Artificial intelligence 42.2-3 (1990)
4) Robert Tibshirani. “A comparison of some error estimates for neural network models”. In: Neural Computation 8.1 (1996),