TRANSP-based Trajectory Optimization of the Current Profile Evolution to Facilitate Robust Non-inductive Ramp-up in NSTX-U

William Wehner[1], Eugenio Schuster[1], and Francesca M. Poli[2]

[1]Lehigh University, Bethlehem, PA
[2]Princeton Plasma Physics Laboratory, Princeton, NJ

E-mail contact: wehner@lehigh.edu

58th Annual Meeting of the APS Division of Plasma Physics

Supported by SCGSR award.

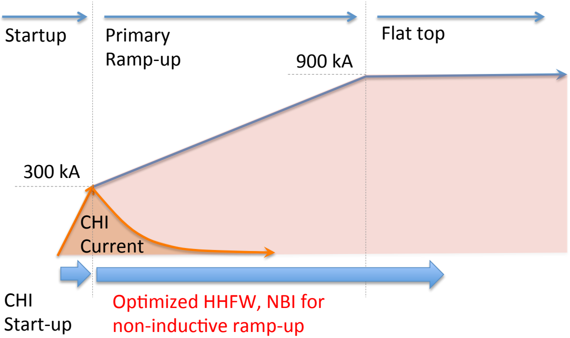
November 2, 2016

Abstract

Initial progress towards the design of non-inductive current ramp-up scenarios in the National Spherical Torus Experiment Upgrade (NSTX-U) has been made through the use of TRANSP predictive simulations [Nucl. Fusion 55 (2015) 123011 (12pp)]. The strategy involves strategic combinations of high harmonic fast waves (HHFW) and neutral beam injection (NBI). However, the early ramping of neutral beams and application of HHFW leads to an undesirably peaked current profile making the plasma unstable to ballooning modes. We present an optimization-based control approach to improve on the non-inductive ramp-up strategy. We combine the TRANSP code with an optimization algorithm based on sequential quadratic programming to search for time evolutions of the NBI powers, the HHFW powers, and the line averaged density that define an open-loop actuator strategy that maximizes the non-inductive current while satisfying constraints associated with the current profile evolution for MHD stable plasmas. This technique has the potential of playing a critical role in achieving robustly stable non-inductive ramp-up, which will ultimately be necessary to demonstrate applicability of the spherical torus concept to larger devices without sufficient room for a central coil.

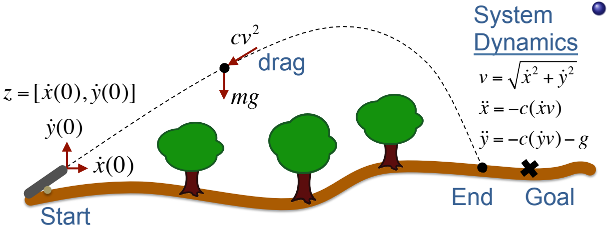
Objectives and Outline

- **Main Objective**
  - Combine Predictive-TRANSP with numerical optimization to find a strategy for non-inductive current ramp-up.



- **In Support of the Main Objective**
  - Control-oriented modeling of NSTX-U and model-based optimization
  - Iterate between model-based and TRANSP-based optimization, improving on control-oriented model

Example: Cannon Targeting via Optimization



- **IDEA:** Transform control problem (cannon aiming) into optimization problem

$$\begin{aligned} \underset{\mathbf{z}}{\text{minimize}} \quad & J(\mathbf{z}) \\ \text{subject to} \quad & h(\mathbf{z}) = 0 \\ & g(\mathbf{z}) \leq 0 \end{aligned}$$

$\mathbf{z} \leftarrow$ optimization variables

$\underset{\mathbf{z}}{\text{minimize}} \quad J(\mathbf{z}) \quad \leftarrow$ optimization objective powder consumed

$$J \propto v = \sqrt{\dot{x}(0)^2 + \dot{y}(0)^2}$$

subject to $\quad h(\mathbf{z}) = 0 \quad \leftarrow$ equality constraints

$$[x, y]_{\text{end}} = [x, y]_{\text{goal}}$$

$$v = \sqrt{\dot{x} + \dot{y}}, \quad \ddot{x} = -c(\dot{x}v), \quad \ddot{y} = -c(\dot{y}v) - g$$

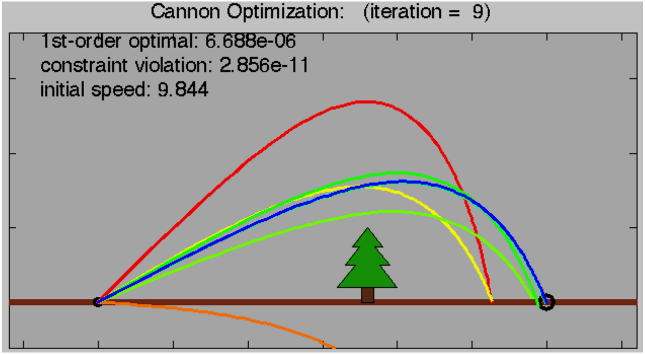
$g(\mathbf{z}) \leq 0 \quad \leftarrow$ inequality constraints

$$y(t) \geq \text{Height Trees}$$

Cannon Targeting: Optimization-based Control

- Start with an approximate solution $\mathbf{z}_0$ (guess)
- Use gradient information of objective and constraints (and approximate hessian of the Lagrangian) to improve on the approximate solution

$$\mathbf{z}_{k+1} = \mathbf{z}_k + \Delta\mathbf{z}$$



Formulation of Non-inductive Ramp-up as Optimization Problem

- A feedforward (open-loop) control policy is obtained via nonlinear optimization - minimize a cost function subject to various constraints

$$\min_{u_{\mathrm{FF}}(t)} \quad J(I_{\mathrm{p}}, I_{\mathrm{p}}^{\mathrm{NI}})$$ } Cost Function, Optimization Objective

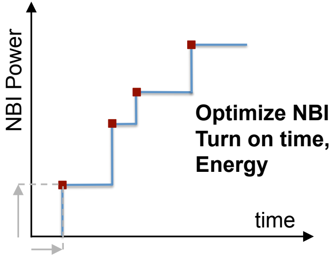
s.t.: $\dot{\psi} = f_\psi(\psi, u_{\mathrm{FF}})$ } Model of poloidal magnetic flux evolution; q profile / current profile are function of ψ; Control-oriented Model or Predictive-TRANSP

$u_{\mathrm{FF}}(t) \in \mathcal{U}$ } Physical limitation on actuators: Bounds / Rate Limit

$\beta_N(t) \leq \beta_{N_{\mathrm{max}}}$ } Nonlinear Constraint: MHD Stability Limit

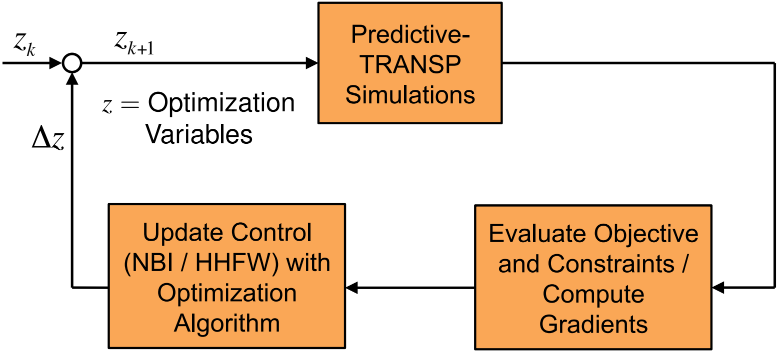
$c(q) \leq 0$ } Constraints on shape of (q): MHD Stability Limit

$c(I_{\mathrm{p}}) \leq 0$ } Constraint on current target



TRANSP-Based Optimization Code

- Use predictive modeling capability of the TRANSP code
- Combine with numerical optimization (OMFIT) to do automated feedforward control optimization.



Optimization Algorithm Based on Sequential Quadratic Programming (SQP)

1. Consider the equality constrained case

$$\min_{\mathbf{z}} \quad J(\mathbf{z})$$
$$\text{s.t.} \quad \boldsymbol{h}(\mathbf{z}) = 0$$

2. Introduce Lagrangian

$$\mathcal{L}(\mathbf{z}, \lambda) = J(\mathbf{z}) - \boldsymbol{\lambda}^T \boldsymbol{h}(\mathbf{z})$$

3. An optimal solution $(\mathbf{z}^*, \boldsymbol{\lambda}^*)$ must satisfy

$$\begin{aligned} \nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}^*, \boldsymbol{\lambda}^*) &= 0 \\ \boldsymbol{h}(\mathbf{z}^*) &= 0 \end{aligned}$$

4. Linearize this set of equations around approximate solution, $\mathbf{z}_k$, $\boldsymbol{\lambda}_k$

$$\left.\begin{aligned} \nabla_{\mathbf{z}} \mathcal{L}(\mathbf{z}_k, \boldsymbol{\lambda}_k) + \nabla^2_{\mathbf{z}} \mathcal{L}(\mathbf{z}_k, \boldsymbol{\lambda}_k) \Delta \mathbf{z} - \nabla \boldsymbol{h}(\mathbf{z}_k) \Delta \boldsymbol{\lambda} &= 0 \\ \boldsymbol{h}(\mathbf{z}_k) + \nabla \boldsymbol{h}(\mathbf{z}_k)^T \Delta \mathbf{z} &= 0 \end{aligned}\right\} \text{ $\Delta \mathbf{z}_k, \Delta \boldsymbol{\lambda}_k$ improves on approximate sol. $\mathbf{z}_k, \boldsymbol{\lambda}_k$}$$

Optimization Algorithm Based on Sequential Quadratic Programming (SQP)

$$\left.\begin{aligned} \nabla_{\mathbf{z}}\mathcal{L}(\mathbf{z}_k,\boldsymbol{\lambda}_k) + \nabla^2_{\mathbf{z}}\mathcal{L}(\mathbf{z}_k,\boldsymbol{\lambda}_k)\Delta\mathbf{z} - \nabla\boldsymbol{h}(\mathbf{z}_k)\Delta\boldsymbol{\lambda} &= 0 \\ \boldsymbol{h}(\mathbf{z}_k) + \nabla\boldsymbol{h}(\mathbf{z}_k)^T\Delta\mathbf{z} &= 0 \end{aligned}\right\}$$

with $\left[\begin{array}{l} \nabla_{\mathbf{z}}\mathcal{L}(\mathbf{z}_k,\boldsymbol{\lambda}_k) = \nabla J(\mathbf{z}_k) - \nabla\boldsymbol{h}(\mathbf{z}_k)\boldsymbol{\lambda}_k \\ \boldsymbol{\lambda}_{k+1} = \boldsymbol{\lambda}_k + \Delta\boldsymbol{\lambda} \end{array}\right]$

$$\boxed{\begin{aligned} \nabla J(\mathbf{z}_k) + \nabla^2_{\mathbf{z}}\mathcal{L}(\mathbf{z}_k,\boldsymbol{\lambda}_k)\Delta\mathbf{z} - \nabla\boldsymbol{h}(\mathbf{z}_k)\boldsymbol{\lambda}_{k+1} &= 0 \\ \boldsymbol{h}(\mathbf{z}_k) + \nabla\boldsymbol{h}(\mathbf{z}_k)^T\Delta\mathbf{z} &= 0 \end{aligned}}$$

5. The above is equivalent to the optimality conditions of the Quadratic Program (QP)

$$\begin{aligned} \min_{\Delta\mathbf{z}} \quad & \nabla J(\mathbf{z}_k)^T\Delta\mathbf{z} + \frac{1}{2}\Delta\mathbf{z}^T\mathbf{H}_k\Delta\mathbf{z} \\ \text{s.t.} \quad & \boldsymbol{h}(\mathbf{z}_k) + \nabla\boldsymbol{h}(\mathbf{z}_k)^T\Delta\mathbf{z} = 0 \end{aligned} \qquad \text{with} \qquad \mathbf{H}_k = \nabla^2_{\mathbf{z}}\mathcal{L}(\mathbf{z}_k,\boldsymbol{\lambda}_k)$$

Optimization Algorithm Based on Sequential Quadratic Programming (SQP)

- The desired step $\Delta\mathbf{z}$ can be obtained from the Quadratic Program (QP)

$$\begin{aligned} \min_{\Delta\mathbf{z}} \quad & \nabla J(\mathbf{z}_k)^T \Delta\mathbf{z} + \frac{1}{2}\Delta\mathbf{z}^T \mathbf{H}_k \Delta\mathbf{z} \\ \text{s.t.} \quad & \boldsymbol{h}(\mathbf{z}_k) + \nabla \boldsymbol{h}(\mathbf{z}_k)^T \Delta\mathbf{z} = 0 \end{aligned}$$

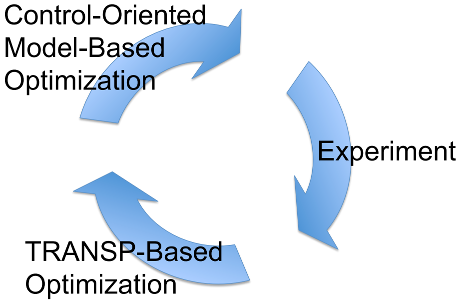
- Then iterate

$$\begin{aligned} \mathbf{z}_{k+1} &= \mathbf{z}_k + \Delta\mathbf{z}_k \\ \boldsymbol{\lambda}_{k+1} &= \boldsymbol{\lambda}_k + \Delta\boldsymbol{\lambda}_k = \boldsymbol{\lambda}_{QP} \text{ (multiplier from QP)} \end{aligned}$$

- Form of Newton's method called sequential quadratic programming
- Inequality constraints can be added to the QP
- Requires gradients, ∇J and $\nabla \boldsymbol{h}$ - obtained by finite differences
  - For n-dimensional $\mathbf{z}$ requires $n+1$ TRANSP simulations to obtain gradients

Current Profile Evolution Simulation Method

- Use simple control oriented model for simulating system in replace of Predictive-TRANSP
- Allows for much faster optimization (minutes)
- Iterate between TRANSP-based optimization and control-oriented model-based optimization
- On each iteration improve control-oriented model and use result of model-based optimization to initialize TRANSP-based optimization



Control-oriented Modeling of the Current Profile Evolution for Model-based Optimization

- Magnetic Diffusion Equation

$$\frac{\partial \psi}{\partial t} = \eta(T_e) c_1 \frac{1}{\hat{\rho}} \frac{\partial}{\partial \hat{\rho}} \left(\hat{\rho} c_2 \frac{\partial \psi}{\partial \hat{\rho}} \right) + \eta(T_e) c_3 \left(j_{\mathsf{aux}} + j_{\mathsf{bs}} \right)$$

- Boundary Conditions

$$\left.\frac{\partial \psi}{\partial \hat{\rho}}\right|_{\hat{\rho}=0} = 0 \qquad \left.\frac{\partial \psi}{\partial \hat{\rho}}\right|_{\hat{\rho}=1} = -c_4 I_{\mathsf{p}}(t)$$

  - c_i are geometric parameters associated with the plasma shape
  - c_i assumed to be fixed from shot to shot, i.e. parameterized by time
  - Plasma Resistivity $\eta(T_e)$ scales inversely with temperature
  - j_{aux}: Auxiliary current drive sources from NBI, ECCD.
  - j_{bs}: Bootstrap current drive.

Control-oriented Modeling: Scaling Laws for Temperature, Resistivity

- Scaling law approximations for temperature, density, and current drive efficiencies allow for simplified control-oriented modeling
- Temperature taken as fixed profile that scales with plasma current, line averaged density, and total auxiliary power

$$T_e(\hat{\rho},t) = k_{T_e}(\hat{\rho}) \left[\frac{T_e^{prof}(\hat{\rho})}{n_e^{prof}(\hat{\rho})} \right] \frac{I_p(t)\sqrt{P_{tot}(t)}}{u_n(t)} \qquad n_e(\hat{\rho},t) = n_e^{prof}(\hat{\rho}) u_n(t)$$

- Resistivity scales with temperature

$$\eta(\hat{\rho},t) = k_{sp}(\hat{\rho}) \frac{Z_{eff}}{T_e(\hat{\rho},t)^{3/2}}$$

Control-oriented Modeling: Current Drive Sources

- Auxiliary current drive from NBI and ICRH

$$j_{\text{aux}}\big|_{\text{source}} = \underbrace{j^{dep}_{\text{source}}(\hat{\rho})}_{\text{NUBEAM / TORIC}} \; \underbrace{f\left(T_e(\hat{\rho},t), n_e(\hat{\rho},t), I_{\text{p}}\right)}_{\text{scaling law}} P_{\text{source}}(t)(1 - f_{\text{shine through}})$$

  - Current drive deposition profiles, $j^{dep}_{\text{source}}(\hat{\rho})$, taken as fixed profiles based on TRANSP analysis runs
  - The efficiencies scale with temperature, density, and total plasma current.

- Bootstrap current modeled with Sauter law

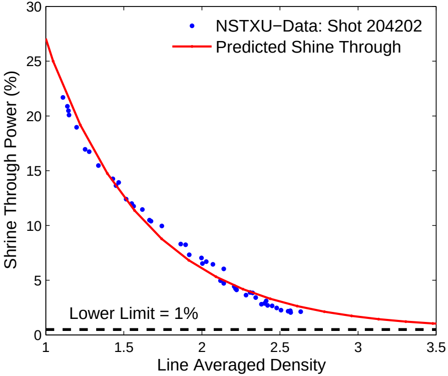
$$\frac{<\bar{j}_{bs} \cdot \bar{B}>}{B_{\phi,0}} = \frac{R_0}{\hat{F}} \left(\frac{\partial \psi}{\partial \hat{\rho}}\right)^{-1} \left[2\mathcal{L}_{31} T_e \frac{\partial n_e}{\partial \hat{\rho}} + (2\mathcal{L}_{31} + \mathcal{L}_{32} + \alpha \mathcal{L}_{34})\, n_e \frac{\partial T_e}{\partial \hat{\rho}}\right]$$

Control-oriented Modeling: NBI Shine Through Loss

- NBI shine through loss approximated by a function of the form

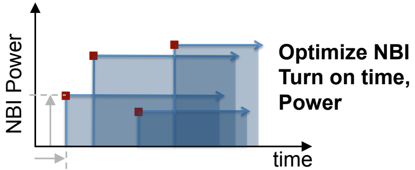
$$f_{\text{shine through}} = \alpha + \beta \exp(-\gamma \bar{n}_e),$$

- Parameters $\alpha \geq 0$, β, γ are obtained by a regression fit to NSTX-U data



Test Case: TRANSP-Based Optimization For Non-inductive Ramp-up Optimization

- Control Parameterization:
  - Assume individual NBI cannot be modulated in ramp-up phase
  - Control parameters include the turn-on time and power of individual NBI



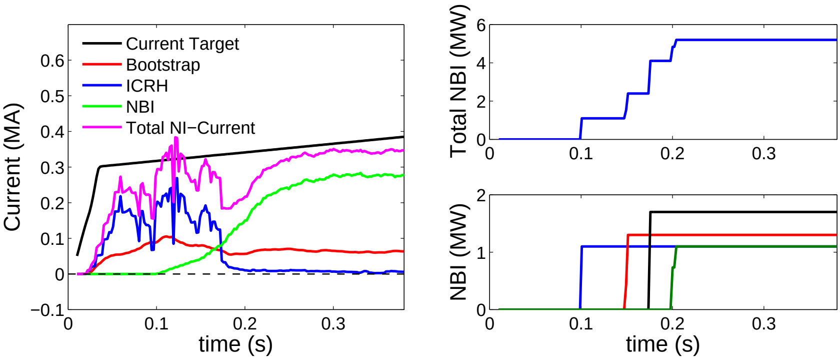
- Cost function

$$J = \int_0^t \left(I_{\mathrm{p}}^{\mathrm{target}}(\tau) - I_{\mathrm{NI}}(\tau) \right)^2 d\tau = \int_0^t \left(I_{\mathrm{OHM}}(\tau) \right)^2 d\tau$$

- Constraints
  - Consider only limits on NBI power

Test Case: TRANSP-Based Optimization For Non-inductive Ramp-up Optimization

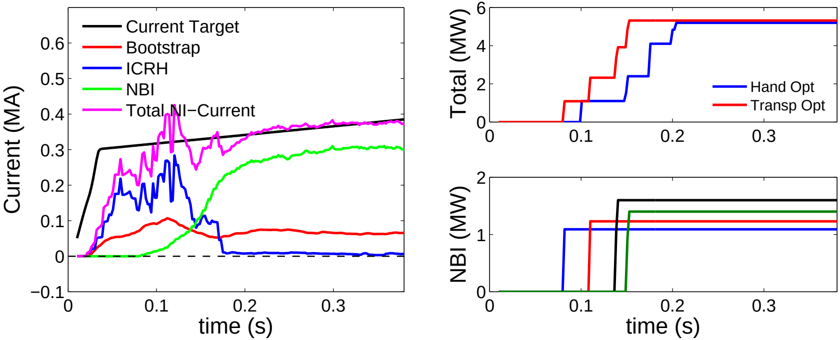
- Case 1: Optimization by Hand



- Figure 9 - F. M. Poli et al. Simulations of non-inductive current ramp-up and sustainment in the National Spherical Torus Experiment Upgrade. Nucl. Fusion 55 (2015) 123011 (12pp)

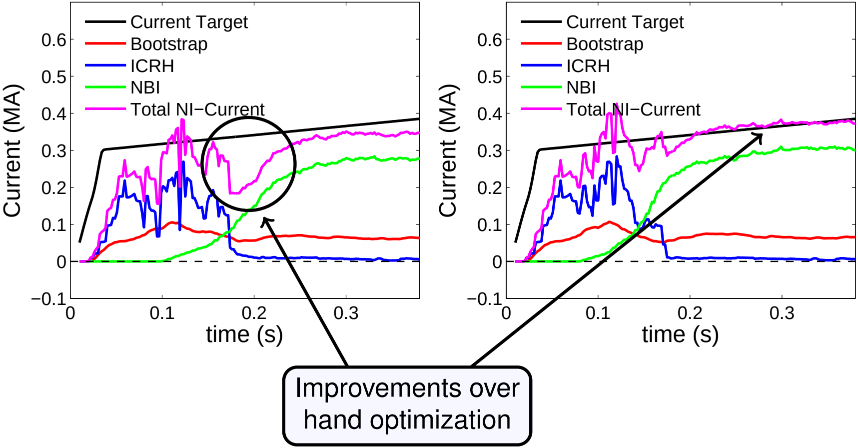
Test Case: TRANSP-Based Optimization For Non-inductive Ramp-up Optimization

- Case 2: Optimization by Automated TRANSP-based optimizer



- Result of an optimization of the NBI powers to meet the current target with non-inductive sources

Test Case: TRANSP-Based Optimization For Non-inductive Ramp-up Optimization



Conclusions and Future Work

- **TRANSP-based Optimization**
  - Optimization code introduced to OMFIT that uses TRANSP simulations to optimize the non-inductive current fraction during ramp-up
  - Code can easily be modified to optimize other objectives
  - Optimization approach allows for inclusion of constraints such as actuator bounds and limits associated with MHD stable plasmas
- **Future Work**
  - Combine the TRANSP-based optimization with constraints for q profile evolution
  - Include current ramp rate as optimization variable
    - Current ramp rate can be optimized simultaneously with the auxiliary current drive powers