Fusion Engineering and Design 146 (2019) 547–550

Contents lists available at ScienceDirect

Fusion Engineering and Design

journal homepage: www.elsevier.com/locate/fusengdes

TRANSP-based optimization towards tokamak scenario development

W.P. Wehner[a,*], E. Schuster[a], M.D. Boyer[b], F.M. Poli[b]

[a] Department of Mechanical Engineering and Mechanics, Lehigh University, Bethlehem, PA 18015, USA
[b] Princeton Plasma Physics Laboratory, Princeton, NJ, USA

ARTICLE INFO

Keywords:
TRANSP
Scenario development
Optimal control design
Optimal plasma scenario planning
NSTX-U
DIII-D

ABSTRACT

An optimization approach that incorporates the predictive transport code TRANSP is proposed for tokamak scenario development. Optimization methods are often employed to develop open-loop control strategies to aid access to high performance tokamak scenarios. In general, the optimization approaches use control-oriented models, i.e. models that are reduced in complexity and prediction accuracy as compared to physics-oriented transport codes such as TRANSP. In the presented approach, an optimization procedure using the TRANSP code to simulate the tokamak plasma is considered for improved predictive capabilities. As a test case, the neutral beam injection (NBI) power is optimized to develop a control strategy that maximizes the noninductive current fraction during the ramp-up phase for NSTX-U. Simulation studies towards the achievement of noninductive ramp-up in NSTX-U have already been carried out with the TRANSP code. The optimization-based approach proposed in this work is used to maximize the noninductive current fraction during ramp-up in NSTX-U, demonstrating that the scenario development task can be automated. An additional test case considers optimization of the current ramp rate in DIII-D for obtaining a stationary plasma characterized by a flat loop voltage profile in the flattop phase.

1. Introduction

Many tokamak plasma-control problems are well suited to be posed as optimization problems and can be solved with numerical optimization methods. Numerous pieces of work have in fact considered numerical optimization as a means to develop open-loop control solutions for safety factor profile regulation and access to stationary plasmas [1–6]. In general, the optimization approaches use simplified control-oriented models [7–11], i.e. models that are often reduced in complexity and prediction accuracy as compared to physics-oriented transport codes such as TRANSP [12]. The next step towards improving the quality of the optimization results is naturally to replace the control-oriented model with a more sophisticated physics-based transport code like TRANSP at the expense of slower convergence times.

This work presents progress towards embedding the TRANSP code into an optimization procedure for synthesizing control solutions towards the attainment of high performance plasma scenarios in tokamaks. The TRANSP code is combined with an optimization algorithm embedded in OMFIT [13,14], where OMFIT acts primarily as an accessory code to automate the issuing of TRANSP runs necessary to carry out the optimization procedure.[1] Two sample plasma control problems are considered as test cases. First, the problem of achieving noninductive current ramp-up in NSTX-U via optimization of the NBI power is considered. An optimal ramp-up strategy, i.e. one that maximizes the noninductive current fraction, has already been studied through TRANSP simulations [15]. The TRANSP-based optimization approach is used to reproduce the ramp-up strategy proposed in [15], demonstrating the design task can be automated. Second, the problem of achieving a stationary plasma characterized by a flat loop voltage profile during the flattop phase in DIII-D is considered.

It is not uncommon for plasma physics researchers to use transport codes to solve engineering design problems. For example, a study considering the optimal launch location of high frequency fast waves for maximum current drive efficiency at DIII-D was carried out using GENRAY simulations [16]. These types of studies, often involving "optimization" by-hand, could be aided by the tool described in this work. Not only does the tool provide a truly optimized result, but also automates the process, saving many hours of work.

This paper is organized as follows. An overview of the TRANSP-based optimization approach is described in Section 2, the two example test cases are presented in Section 3, and conclusions are given in Section 4.

* Corresponding author.
E-mail address: wehner@lehigh.edu (W.P. Wehner).

[1] OMFIT, developed and maintained by General Atomics, San Diego, is a python-based tool designed to standardize and manage the data of many plasma transport codes such as TRANSP.

https://doi.org/10.1016/j.fusengdes.2019.01.019
Received 8 October 2018; Received in revised form 12 December 2018; Accepted 4 January 2019
Available online 05 March 2019

2. Overview of TRANSP-based optimization approach

The optimization problem given by

$$\begin{aligned} \underset{\boldsymbol{\alpha}}{\text{minimize}} \quad & J(\boldsymbol{\alpha}) \\ \text{subjectto} \quad & \boldsymbol{\alpha}_{\min} \le \boldsymbol{\alpha} \le \boldsymbol{\alpha}_{\max}, \\ & \mathbf{g}_{\text{in}}(\boldsymbol{\alpha}) \le 0, \\ & \mathbf{g}_{\text{eq}}(\boldsymbol{\alpha}) = 0, \end{aligned} \tag{P1}$$

is considered in this work. The problem involves the minimization of the scalar objective function, $J(\boldsymbol{\alpha})$, with respect to a set of optimization variables represented by $\boldsymbol{\alpha}$, subject to a set of constraints. The optimization variables could be parameterized in a variety of ways depending on the optimization goal. For example, $\boldsymbol{\alpha}$ may define a control trajectory over time as shown in Fig. 1. The objective function, J, quantifies distance in some sense to the desired design goal; it incorporates the plasma evolution as simulated by predictive TRANSP and assigns a penalty to deviations from the desired goal as a function of the optimization variables. The constraints include simple bounds on the optimization variables, $\boldsymbol{\alpha}_{\min} \le \boldsymbol{\alpha} \le \boldsymbol{\alpha}_{\max}$, and other, possibly nonlinear, constraints described by $\mathbf{g}_{\text{in}}(\boldsymbol{\alpha}) \le 0$ and $\mathbf{g}_{\text{eq}}(\boldsymbol{\alpha}) = 0$, which may define tokamak operational or stability limits. Numerous tokamak design problems can be formulated into an optimization problem of the form of (P1), as will be shown in the following section.

Sequential quadratic programming (SQP), which is the most widely used approach to solving constrained optimal control problems like (P1) [17], is used in this work. Essentially, the algorithm searches for a solution by first starting with an approximate solution, $\boldsymbol{\alpha}_0$, and works to improve on the solution by taking steps, $\Delta\boldsymbol{\alpha}$. Sequential iterates, $\boldsymbol{\alpha}_{k+1} = \boldsymbol{\alpha}_k + \Delta\boldsymbol{\alpha}_k$, are found with the use of gradient information of the objective function and constraints [17,18] (see [19] for implementation details).

The optimization approach treats TRANSP as a black box, which makes it impossible to obtain gradients analytically. Instead, the gradients are calculated by forward finite differences. This is accomplished by perturbing each element of $\boldsymbol{\alpha}$ by a small amount and then computing each element of the gradient according to

$$\nabla J|_j = \frac{J(\boldsymbol{\alpha} + \epsilon \mathbf{e}_j) - J(\boldsymbol{\alpha})}{\epsilon}, \tag{1}$$

where $\mathbf{e}_i$ is the ith coordinate vector, i.e. $\mathbf{e}_1 = [1, 0, \ldots, 0]^T$, $\mathbf{e}_2 = [0, 1, 0, \ldots, 0]^T$, etc. Gradients can be computed similarly for each of the constraint functions. Each iteration of the optimization task, therefore, requires $n + 1$ TRANSP runs in order to obtain gradient information, where n is the dimension of $\boldsymbol{\alpha}$.

The implementation of the SQP method and necessary code for parameterizing and issuing TRANSP runs automatically is embedded into OMFIT, and the automated TRANSP + OMFIT optimization procedure is outlined in Fig. 2. In summary, the optimization procedure works as follows:

- Choose initial approximate solution $\boldsymbol{\alpha}_0$.
- Repeat $k = 0, 1, \ldots, k_{\max}$:
  - Configure the input files and initiate a TRANSP run parameterized by current iterate $\boldsymbol{\alpha}_k$.
  - Configure and initiate n TRANSP runs parameterized by $\boldsymbol{\alpha}_k + \epsilon \mathbf{e}_i$ for $i = 1, 2, \ldots, n$ to obtain gradients.
  - Evaluate function values and compute gradients.
  - Determine search direction $\Delta\boldsymbol{\alpha}_k$ (descent direction) and size of step via SQP [18].
  - Check convergence.
  - Obtain new iterate $\boldsymbol{\alpha}_{k+1} = \boldsymbol{\alpha}_k + \Delta\boldsymbol{\alpha}_k$.

The optimization time is largely a function of TRANSP simulation run-time, which varies substantially depending on the particular setup.

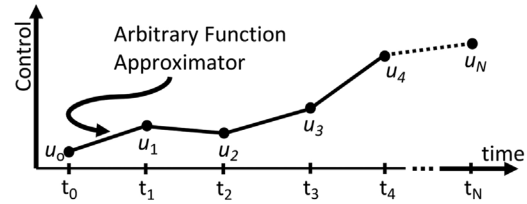


Fig. 1. Control parameterization with arbitrary function approximator in between control updates. In this case, the optimization variables include the entire control sequence, i.e. $\boldsymbol{\alpha}^T = [\mathbf{u}_0^T, \mathbf{u}_1^T, \ldots, \mathbf{u}_N^T]$. An arbitrary function approximator could be employed to define the control value between updates, for example a linear function (as shown).

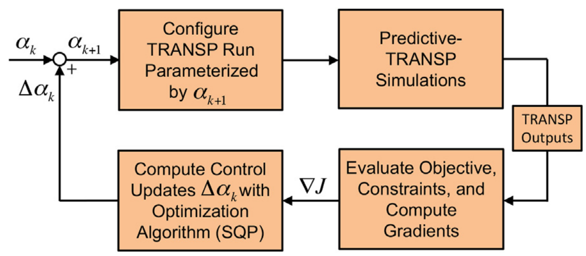


Fig. 2. TRANSP-based optimization loop.

The TRANSP code includes a wide variety of models for neoclassical transport, anomalous transport, heat and current sources, sinks, equilibrium, and plasma boundary shape. A complete list of models is available in the NTCC Model Library (w3.pppl.gov/NTCC). The choice of models and the resolution of the spatial grid will greatly change the length of the TRANSP run-time. For modest runs like the examples presented in this work, we can assume an average run-time of 10–15 h when run on 32 cores on the PPPL cluster. Each iteration of the optimization algorithm requires, on average, 3 TRANSP simulation run times, an initial function evaluation, a set of function evaluations for computing gradients (run in parallel), and an additional run for SQP line-search (backtracking necessary to globalize the algorithm). The number of iterations necessary for convergence to a "good" solution depends greatly on the quality of the initial guess solution, potentially ranging from 1 to 10 iterations. This means a single iteration could take about 30–45 h and a complete solution about a week.

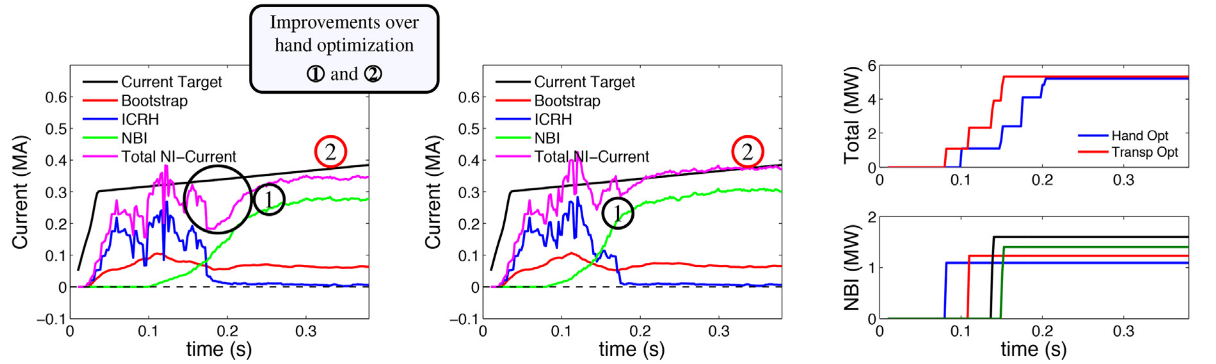
3. Sample problems

3.1. Sample problem 1: noninductive ramp-up in NSTX-U

One of the primary research goals of NSTX-U is to advance the spherical torus concept for a fusion nuclear science facility (FNSF), which requires developing noninductive start-up, ramp-up, and sustainment techniques since a large-scale spherical torus will have little to no room for a central solenoid.[2] Under certain plasma conditions, NSTX has been shown to sustain about 70% of the current noninductively [20]. With recent upgrades, including an additional high tangency neutral beam set and high frequency fast wave (HHFW) antenna, exhaustive simulations anticipate that NSTX-U [21] will be able to sustain fully noninductive current in the flattop phase of the discharge. However, much research is still required to develop a successful approach for noninductive start-up and ramp-up in NSTX-U.

A provisional strategy for fully noninductive ramp-up, which uses a particular timing of the NBI to meet the target plasma current, has been explored by carrying out predictive TRANSP simulations [15]. However, this study involves the arduous task of fine tuning by hand the optimal NBI timing. As a test case, the TRANSP-based optimization

[2] The central solenoid coil serves as the inductive current drive.



(a) By-hand optimization.

(b) TRANSP-based optimization.

(c) Comparison of total NBI powers (by-hand tuning vs TRANSP-based tuning) (top). NBI powers tuned by TRANSP-based optimization (bottom).

Fig. 3. TRANSP-based optimization test case. In (a) the NBI powers are selected by hand to best achieve the noninductive ramp-up in NSTX-U. In (b) the TRANSP-based optimization routine is used to obtain the NBI powers. In (a) significant improvements made over the hand optimization are annotated: **(1)** large deficit in the plasma current at around 0.18 s, **(2)** mismatch with respect to target current after 0.2 s. Optimized NBI powers are shown in (c).

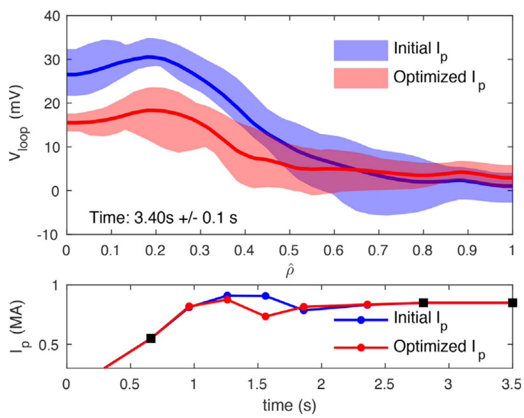


Fig. 4. TRANSP-based optimization of the current ramp-up to obtain a flat loop voltage profile in the flattop. At top, the initial (blue) and optimized (red) loop voltage profile in the flattop phase is plotted, and at bottom, the initial (blue) and optimized (red) current ramp-up is plotted. The circles represent free parameters and squares are fixed. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

approach is employed to automate and improve on the selection of optimal NBI timing.

The goal is to obtain an open-loop control strategy, i.e. sequence of control requests parameterized by time, that could sustain the target plasma current noninductively through the ramp-up phase. This problem is well suited to be tackled by following an optimization approach. To formulate the optimization problem, the objective function

$$J^{\alpha}(I_{\rm p}) = \int_{t_i}^{t_f} (I_{\rm p}^{\rm target}(\tau) - I_{\rm NI}(\tau))^2 d\tau = \int_{t_i}^{t_f} (I_{\rm OHM}(\tau))^2 d\tau \tag{2}$$

is introduced, which penalizes the difference between the total noninductive current drive and the current target. The control parameterization, $\boldsymbol{\alpha}$, consists of the turn-ON time and injected power of each NBI,[3]

$$\boldsymbol{\alpha} = [t_{\rm NB-1}^{\rm on}, P_{\rm NB-1}, t_{\rm NB-2}^{\rm on}, P_{\rm NB-2}, ..., t_{\rm NB-4}^{\rm on}, P_{\rm NB-4}]^T, \tag{3}$$

and the constraints include power limits associated with the NBI sources. Additional constraints (not considered in this sample problem) such as bounds on the allowable q profile shape, which could be represented as nonlinear inequality constraints on the optimization variables, could be introduced to ensure stability against deleterious MHD effects.

In Fig. 3(a), where Figure 9 of [15] has been reproduced, NBI powers are optimized (by hand) to maximize the noninductive current fraction. Fig. 3(b) shows instead the results when the NBI powers are optimized by the TRANSP + OMFIT optimization routine proposed in this work. The individual four NBI powers tuned by the TRANSP-based optimization are shown at the bottom in Fig. 3(c), while the total NBI powers from both the TRANSP-based optimization and the by-hand optimization are compared at the top. Notable improvements in meeting the current target by TRANSP-based optimization over by-hand optimization are highlighted in Fig. 3(a). A large deficit in the plasma current at around 0.18 s and a mismatch with respect to the target current after 0.2 s are both diminished. Additionally, the optimization approach saves time by automating the entire procedure.

3.2. Sample problem 2: scenario development in DIII-D

A primary goal for the DIII-D research program over the next five years is to develop the physics basis for a high q ($q_{\min} > 2$), high β_N steady-state scenario[4] that can serve as the basis for a steady-state ITER scenario at (fusion gain) $Q = 5$. Various approaches are being considered to maximize the bootstrap current contribution, so that fully noninductive ($f_{NI} = 1$) discharges can be obtained for several resistive current diffusion times. It is anticipated that the upgrades to DIII-D including an additional off-axis neutral beam injection (NBI) system in 2019 will provide sufficient auxiliary current drive to maintain fully noninductive plasmas at high β_N. However, much work is necessary to investigate MHD stability, adequate confinement, and early achievement and sustainment of the steady-state condition.

Recent work, in which the time trajectory of the plasma current has been optimized to guide the plasma to a stationary state characterized by a flat loop voltage profile, has been accomplished with control-oriented (reduced complexity[5]) model-based optimization methods [7,10,8,9,11]. However, the promising results obtained in simulations may not hold up to experimental testing due to the approximate nature of the control-oriented models. For improved predictive quality, the TRANSP-based optimization approach can be employed to optimize the time trajectory of the plasma current to reach a flat loop voltage profile.

An objective function of the form

[3] NSTX-U has 4 NBI sources.

[4] Steady state scenario is characterized by a plasma state that is fully relaxed and the plasma current is composed entirely of intrinsic (bootstrap) and noninductive auxiliary current drives.

[5] Models reduced in complexity and prediction accuracy but much faster to evolve as compared to physics-oriented prediction codes such as TRANSP.

$$J^{\alpha}(V_{\text{loop}}) = \sum_t \int_0^1 (V_{\text{loop}}(t, \hat{\rho}) - V_{\text{loop}}(t, 1))^2 d\hat{\rho} \quad (4)$$

is considered for $t = 3.1$, 3.2, and 3.3 s. The objective function penalizes deviations of the loop voltage profile with respect to its edge value at several times in the flattop phase of the discharge. Assigning penalties at multiple times aids in obtaining a profile that remains flat, or as flat as possible, through the flattop phase instead of obtaining a solution that simply passes through a stationary state momentarily.

The optimization variable considered in this sample case is represented by a piece-wise linear function of the reference plasma current. Five values over time of the plasma current are considered as the optimization variables, $\boldsymbol{\alpha}$ (see red dots in the bottom plot of Fig. 4). The plasma current is treated as an actuator and it is assumed that the inductive-coil current is regulated via a dedicated controller to meet the desired reference current. The constraints include bounds on the reference current and a rate limit on the reference current ramp, i.e.

$$0.3\text{MA} \leq I_p \leq 1.5\text{MA}, \quad |\text{dI}_\text{p}/\text{dt}| \leq 2\,\text{MA/s}. \quad (5)$$

The results of the optimization are shown in Fig. 4. As can be seen, the optimized control policy is parameterized as a piece-wise linear function with updates every 250 ms. The initial (blue) and optimized (red) loop voltage profiles are plotted at the end of the flattop phase. Significant improvements are made in flattening the loop voltage profile by introducing a wiggled current ramp-up.

4. Conclusions

The TRANSP + OMFIT optimization routine presented in this work represents a valuable tool for developing control strategies of varying objectives at the scenario-development stage. The optimization algorithm written in OMFIT can accept arbitrary objective and constraint functions, and the optimization variables can be parameterized in various ways. It is a straightforward process to introduce other optimization variables such as the NBI tangency radius, line averaged density, electron cyclotron current drive, or the plasma current ramp rate, to name a few.

The toolset described in this work represents the first steps in a natural progression from the well-established control-oriented model-based optimization to a physics-oriented model-based optimization characterized by a higher accuracy. The primary shortcoming of the TRANSP-based optimization is much slower convergence times due to the lengthy simulation times associated with running predictive TRANSP simulations. On the other hand, the control-oriented optimization approaches of prior work have the advantage of fast simulation times at the expense of prediction accuracy. However, the two approaches could be combined in an iterative fashion, in which the fast control-oriented optimization accelerates the TRANSP-based optimization by providing an initial guess solution, and the TRANSP-based optimization improves the accuracy of the control-oriented optimization by providing updated model parameters on each iteration.

Acknowledgments

Work supported by the U.S. Department of Energy (DE-SC0010661 and DE-FC02-04ER54698) and the DOE Office of Science Graduate Student Research (SCGSR) program award.

References

[1] Y. Ou, et al., Plasma Phys. Control. Fusion 50 (2008) 115001.
[2] C. Xu, et al., IEEE Trans. Plasma Sci. 38 (2010) 163–173.
[3] F. Felici, et al., Plasma Phys. Control. Fusion 54 (2012) 025002.
[4] J.V. Dongen, et al., Plasma Phys. Control. Fusion 56 (2014) 125008.
[5] J. Barton, et al., Nucl. Fusion 55 (2015) 093005.
[6] W. Wehner, et al., Fusion Eng. Des. 123 (2017) 513–517 0920-3796 Proceedings of the 29th Symposium on Fusion Technology (SOFT-29), Prague Czech Republic, September 5–9, 2016.
[7] Y. Ou, et al., Fusion Eng. Des. 82 (2007) 1153–1160.
[8] E. Witrant, et al., Plasma Phys. Control. Fusion 49 (2007) 1075.
[9] F. Felici, et al., Nucl. Fusion 51 (2011) 083052.
[10] J.E. Barton, et al., 52nd IEEE Conference on Decision and Control, (2013), pp. 4182–4187.
[11] J.E. Barton, et al., Plasma Phys. Control. Fusion 57 (2015) 115003.
[12] R. Hawryluk, Course on Physics of Plasma Close to Thermonuclear Conditions, (1979) Varenna, Italy.
[13] B.A. Grierson, et al., Fusion Sci. Technol. 74 (2018) 101–115.
[14] O. Meneghini, et al., Nucl. Fusion 55 (2015) 083008.
[15] F. Poli, et al., Nucl. Fusion 55 (2015) 123011.
[16] R. Prater, et al., Nucl. Fusion 54 (2014) 083024.
[17] J.T. Betts, J. Guid. Control Dyn. 21 (1998) 193–207.
[18] J. Betts, Practical Methods for Optimal Control and Estimation Using Nonlinear Programming, 2nd ed., Society for Industrial and Applied Mathematics, 2010.
[19] W.P. Wehner, Ph.D. thesis, Lehigh University, USA, 2017.
[20] S. Gerhardt, et al., Nucl. Fusion 51 (2011) 033004.
[21] S. Gerhardt, et al., Nucl. Fusion 52 (2012) 083020.