Ramp-Up-Phase Current-Profile Control of Tokamak Plasmas via Nonlinear Programming

C. Xu, Y. Ou, J. Dalessio, E. Schuster, T. C. Luce, J. R. Ferron, M. L. Walker, and D. A. Humphreys

***Abstract*—The achievement of suitable toroidal-current-density profiles in tokamak plasmas plays an important role in enabling high fusion gain and noninductive sustainment of the plasma current for steady-state operation with improved magnetohydrodynamic stability. The evolution in time of the current profile is related to the evolution of the poloidal magnetic flux, which is modeled in normalized cylindrical coordinates using a partial differential equation (PDE) usually referred to as the magnetic flux diffusion equation. The dynamics of the plasma current density profile can be modified by the total plasma current and the power of the noninductive current drive. These two actuators, which are constrained not only in value and rate but also in their initial and final values, are used to drive the current profile as close as possible to a desired target profile at a specific final time. To solve this constrained finite-time open-loop PDE optimal control problem, model reduction based on proper orthogonal decomposition is combined with sequential quadratic programming in an iterative fashion. The use of a low-dimensional dynamical model dramatically reduces the computational effort and, therefore, the time required to solve the optimization problem, which is critical for a potential implementation of a real-time receding-horizon control strategy.**

***Index Terms*—Distributed parameter systems, nonlinear programming (NLP), proper orthogonal decomposition (POD), tokamak plasma control.**

I. Introduction

THE CONTROL of the current profile in tokamak plasmas has been proved to be the enabler of advanced scenarios characterized by improved confinement, enhanced magnetohydrodynamic stability, and possible steady-state operation [1], [2]. The evolution in time of the current profile is related to the evolution of the spatial derivative of the poloidal flux profile, which is modeled in normalized cylindrical coordinates using a partial differential equation (PDE) usually referred to as the magnetic flux diffusion equation [3]. The total plasma current and the power of the noninductive current drive can be used to modify the dynamics of the poloidal-magnetic-flux profile.

Manuscript received February 18, 2009; revised August 18, 2009. Current version published February 10, 2010. This work was supported in part by a Grant from the Commonwealth of Pennsylvania, Department of Community and Economic Development, through the Pennsylvania Infrastructure Technology Alliance, by the National Science Foundation under CAREER Award Program ECCS-0645086, and by the Department of Energy under Contract DE-FC02-04ER54698.

C. Xu, Y. Ou, J. Dalessio, and E. Schuster are with the Department of Mechanical Engineering and Mechanics, Lehigh University, PA 18015 USA (e-mail: chx205@lehigh.edu).

T. C. Luce, J. R. Ferron, M. L. Walker, and D. A. Humphreys are with General Atomics, San Diego, CA 92186 USA.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TPS.2009.2037626

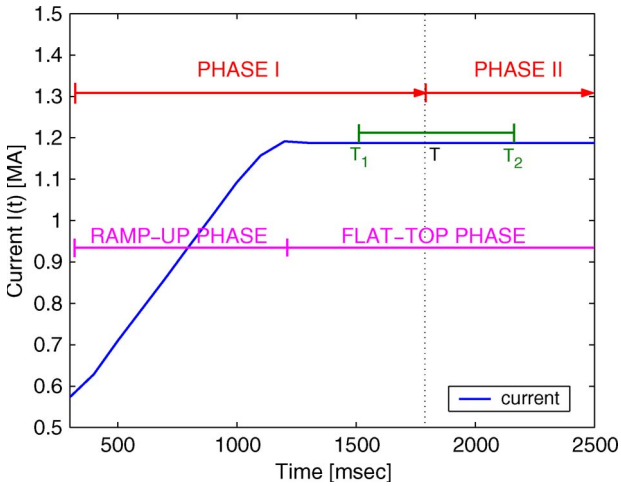

Fig. 1. Total plasma current evolution defines the ramp-up and flattop phases of the discharge. The control problem focuses on Phase I, which includes the ramp-up phase and the first part of the flattop phase. The control goal is to drive the current profile from some initial arbitrary condition to a predefined target profile at some time T within the time window $[T_1, T_2]$ in the flattop phase.

These two physical actuators enter the magnetic flux diffusion equation as interior, boundary, and diffusivity control terms.

Recent experiments in different devices around the world (JET [4], [5], [6], DIII-D [7], JT-60U [8], and Tore Supra [9], [10]) have demonstrated significant progress in achieving profile control. One possible approach to current-profile control focuses on creating the desired current profile during the plasma current ramp-up and early flattop phases with the aim of maintaining this target profile during the subsequent phases of the discharge. Since the actuators that are used to achieve the desired target profile are constrained, experiments have shown that some of the desirable target profiles may not be achieved for all arbitrary initial conditions. Therefore, a perfect matching of the desirable target profile may not be physically possible. In practice, the objective is to achieve the best possible approximate matching at a prespecified time during the early flattop phase of the total plasma current pulse, as shown in Fig. 1. Thus, such matching problem can be formulated as a finite-time optimal control problem for the magnetic flux diffusion PDE [11].

In general, it is very difficult, if not impossible, to provide analytic solutions for optimal control problems defined for PDE systems (see, e.g., [12]–[14] and the references therein). Numerical approaches for the solution of these PDE problems arise as an effective alternative. State-of-the-art numerical algorithms for PDE optimal control synthesis can be categorized

into indirect and direct methods, based on calculus of variations and nonlinear programming (NLP), respectively. The calculus of variations can be used in indirect methods to derive the optimality conditions for a given optimal control problem. These optimality conditions result in a two-boundary-value problem, which usually requires numerical computations for its solution. However, it is often technically challenging to compute the variations for systems described by PDEs. As an alternative approach, system states and controls can be treated as independent variables in direct methods to reformulate the original optimal control problem as a constrained PDE-based optimization problem, which can be solved using NLP techniques. The sequential quadratic programming (QP) (SQP) method [15] is one of the most robust algorithms for the numerical solution of NLP problems. A quadratic approximation of the objective functional and a linear approximation of the constraint equation are carried out around a current estimate of the solution of the optimal control problem to convert the original NLP problem into a QP problem. Newton's method can be used to solve the optimality conditions associated with the obtained QP problem. The solution is then used to update the estimate of the solution of the optimal control problem, which is used for a new approximation of the original NLP problem into a QP problem. The SQP procedure generates then a sequence of QP problems and a series of estimates that converge to the solution of the original optimal control problem.

To overcome the high dimensionality of the problem, reduced order modeling (ROM) techniques play a very important role in dealing with the control of infinite-dimensional dynamical systems. The proper orthogonal decomposition (POD) method is an efficient ROM technique used to obtain low-dimensional dynamical systems (LDDSs) from data ensembles that arise from numerical simulation or experimental observation. The POD method has been widely used and proved successful to discover coherent structures from complex physical processes (e.g., [16]) and to control PDE systems (e.g., [17]).

We combine POD and SQP to solve a "motion planning" problem where the open-loop optimal control sequences must be computed over time to minimize the quadratic error between the actual output profile (spatial derivative of the poloidal-magnetic-flux profile) at a prescribed final time and a desired target profile. This work is aimed at saving long trial-and-error periods of time currently spent by fusion experimentalists trying to manually adjust the time evolutions of the actuators to achieve the desired current profile at some prespecified time. Simulation results show the effectiveness of this approach.

This paper is organized as follows. The optimal control problem for the current-profile system is introduced in Section II. The POD method to obtain a reduced order model is discussed in Section III. In Section IV, the Galerkin projection method based on a test function set composed by dominant POD modes is also discussed. In Section V, the procedure for the POD-based optimization is stated, and a brief introduction to SQP optimization theory is presented. Simulation studies are presented in Section VI. This paper is closed in Section VII by stating conclusions and future research remarks.

II. Formulations of the Optimal Control Problem

A. Control-Oriented Model

To enable the model-based control of the current profile at DIII-D, a control-oriented model for the dynamic evolution of the poloidal flux profile during and just following the ramp-up of the plasma current has been recently proposed [18]. The magnetic diffusion equation is combined with empirical correlations obtained at DIII-D for the temperature and noninductive current to introduce a simplified dynamic model describing the evolution of the poloidal flux during the inductive phase of the discharge. In this paper, we make use of the model introduced in [18] to illustrate the proposed open-loop optimal control technique, but full predictive codes (CORSICA, CRONOS, PTRANSP, etc.) could be used with the potential of a more accurate prediction than that provided by this simplified control-oriented model.

The current density j, which flows toroidally around the tokamak and whose profile must be controlled, is related to the derivatives of the poloidal magnetic flux ψ with respect to the toroidal magnetic flux Φ. We let ρ be an arbitrary coordinate indexing the magnetic surface. Any quantity constant on each magnetic surface could be chosen as the variable ρ. We choose the mean geometric radius of the magnetic surface as the variable ρ, i.e., $\pi B_{\phi,o}\rho^2 = \Phi$, where $B_{\phi,o}$ is the reference toroidal magnetic field at the geometric plasma center R_o. The variable $\hat{\rho}$ denotes the normalized radius ρ/ρ_b, and ρ_b is the radius of the last closed flux surface. The evolution of the poloidal flux in normalized cylindrical coordinates is given by the magnetic diffusion equation

$$\frac{\partial \psi}{\partial t} = \frac{\eta(T_e)}{\mu_o \rho_b^2 \hat{F}^2}\frac{1}{\hat{\rho}}\frac{\partial}{\partial \hat{\rho}}\left(\hat{\rho}\hat{F}\hat{G}\hat{H}\frac{\partial \psi}{\partial \hat{\rho}}\right) - R_o \hat{H}\eta(T_e)\frac{\langle \bar{j}_{\mathrm{NI}} \cdot \bar{B}\rangle}{B_{\phi,o}} \tag{1}$$

where t is the time, ψ is the poloidal magnetic flux, η is the plasma resistivity, T_e is the plasma electron temperature, μ_o is the vacuum permeability, $\bar{j}_{\mathrm{NI}}$ is the noninductive source of current density (neutral beam, electron cyclotron, etc.), $\bar{B}$ is the toroidal magnetic field, and $<>$ denotes flux-surface average. $\hat{F}, \hat{G}, \hat{H}$ are geometric factors, which are functions of $\hat{\rho}$. The boundary conditions of (1) are given by

$$\left.\frac{\partial \psi}{\partial \hat{\rho}}\right|_{\hat{\rho}=0} = 0 \qquad \left.\frac{\partial \psi}{\partial \hat{\rho}}\right|_{\hat{\rho}=1} = \frac{\mu_o}{2\pi}\frac{R_o}{\hat{G}\Big|_{\hat{\rho}=1}\hat{H}\Big|_{\hat{\rho}=1}} I(t) \tag{2}$$

where $I(t)$ denotes the total plasma current.

During "Phase I" (see Fig. 1), mainly governed by the ramp-up phase, the plasma current is mostly driven by induction. In this case, it is possible to decouple the equation for the evolution of the poloidal flux from the equation for the evolution of the temperature $T_e(\hat{\rho}, t)$. Highly simplified models for the temperature and noninductive toroidal current density are chosen for this phase.

The temperature T_e is assumed to follow

$$T_e(\hat{\rho}, t) = k_{Te} T_e^{\mathrm{profile}}(\hat{\rho})\frac{I(t)\sqrt{P}}{\bar{n}(t)} \tag{3}$$

where the reference profile T_e^{profile} is identified from DIII-D through Thomson scattering and $k_{Te} = 1.7295 \cdot 10^{10}$ $\text{m}^{-3} \cdot \text{A}^{-1} \cdot \text{W}^{-1/2}$. The average density $\bar{n}$ is defined as

$$\bar{n}(t) = \int_0^1 n(\hat{\rho}, t) d\hat{\rho} \tag{4}$$

where n denotes the plasma density.

The noninductive toroidal current density $\langle \bar{j}_{\text{NI}} \cdot \bar{B} \rangle / B_{\phi,o}$ is assumed to follow

$$\frac{\langle \bar{j}_{\text{NI}} \cdot \bar{B} \rangle}{B_{\phi,o}} = k_{\text{NIpar}} j_{\text{NIpar}}^{\text{profile}}(\hat{\rho}) \frac{I(t)^{1/2} P(t)^{5/4}}{\bar{n}(t)^{3/2}} \tag{5}$$

where the reference profile $j_{\text{NIpar}}^{\text{profile}}$ is identified from DIII-D through a combination of MSE diagnostics and the EFIT equilibrium reconstruction code and $k_{\text{NIpar}} = 1.2139 \cdot 10^{18}$ $\text{m}^{-9/2} \cdot \text{A}^{-1/2} \cdot \text{W}^{-5/4}$. The model for T_e and $\langle \bar{j}_{\text{NI}} \cdot \bar{B} \rangle / B_{\phi,o}$ previously presented considers neutral beams as the only source of current and heating. In the case where more heating and current sources are considered, (3) and (5) should include the weighted contributions of the different sources, and reference profiles need to be identified for each heating and current source.

The resistivity η scales with the temperature T_e as

$$\eta(\hat{\rho}, t) = \frac{k_{\text{eff}} Z_{\text{eff}}}{T_e^{3/2}(\hat{\rho}, t)} \tag{6}$$

where $Z_{\text{eff}} = 1.5$ and $k_{\text{eff}} = 4.27.2 \cdot 10^{-8}\ \Omega \cdot \text{m}\ (\text{keV})^{3/2}$.

By introducing

$$\vartheta_1(\hat{\rho}) = \frac{k_{\text{eff}} Z_{\text{eff}}}{k_{Te}^{3/2} \mu_o \rho_b^2} \frac{1}{\hat{F}^2(\hat{\rho}) (T_e^{\text{profile}}(\hat{\rho}))^{3/2}} \tag{7}$$

$$\vartheta_2(\hat{\rho}) = R_o \hat{H} \mu_o \rho_b^2(\hat{\rho}) k_{\text{NIpar}} j_{\text{NIpar}}^{\text{profile}}(\hat{\rho}) \tag{8}$$

$$D(\hat{\rho}) = \hat{F} \hat{G} \hat{H} \quad k = \frac{\mu_o}{2\pi} \frac{R_o}{\hat{G}\Big|_{\hat{\rho}=1} \hat{H}\Big|_{\hat{\rho}=1}} \tag{9}$$

the normalized poloidal magnetic flux can be rewritten as

$$\frac{1}{\vartheta_1(\hat{\rho})} \frac{\partial \psi}{\partial t} = u_1(t) \frac{1}{\hat{\rho}} \frac{\partial}{\partial \hat{\rho}} \left[\hat{\rho} D(\hat{\rho}) \frac{\partial \psi}{\partial \hat{\rho}} \right] + \vartheta_2(\hat{\rho}) u_2(t). \tag{10}$$

The control inputs u_1 and u_2 are functions of physical actuators such as the total power P of the noninductive current drive, the total plasma current I, and the average density $\bar{n}$, i.e.,

$$u_1(t) = \bar{n}^{1.5} I^{-1.5} P^{-0.75} \quad u_2(t) = P^{0.5} I^{-1}. \tag{11}$$

In this paper, we consider the average density $\bar{n}$ as an uncontrolled but measurable input. This decision is motivated by the fact that the tight control of $\bar{n}$ in experiments can be challenging. The poloidal magnetic flux at the spatial boundaries is determined by the Neumann conditions

$$\frac{\partial \psi}{\partial \hat{\rho}}(0, t) = 0 \quad \frac{\partial \psi}{\partial \hat{\rho}}(1, t) = k u_3(t) \tag{12}$$

where k is a constant and $u_3(t) = I$. The initial condition for the magnetic flux profile is given by

$$\psi(\hat{\rho}, 0) = \psi_0(\hat{\rho}). \tag{13}$$

B. Cost Functional and Constraints

In practice, the toroidal current density is usually specified indirectly by the rotational transform $\bar{\iota}$ (or the safety factor $q = \bar{\iota}^{-1}$), which is defined as

$$\bar{\iota}(\rho, t) = \frac{\partial \psi(\rho, t)}{\partial \Phi}. \tag{14}$$

The constant relationship between Φ and ρ, namely, $\rho = \sqrt{\Phi / \pi B_{\phi,o}}$, and the definition of the normalized radius $\hat{\rho}$ allow us to rewrite (14) as $\bar{\iota}(\hat{\rho}, t) = (\partial \psi / \partial \hat{\rho})(1 / B_{\phi,o} \rho_b^2 \hat{\rho})$. Since $\bar{\iota}$ is uniquely defined by the spatial derivative of the magnetic flux ψ, in this paper, we define the system output as

$$\iota(\hat{\rho}, t) = \frac{\partial \psi}{\partial \hat{\rho}}. \tag{15}$$

The three control inputs $u_1(t)$, $u_2(t)$, and $u_3(t)$ are indeed functions of only two independent variables, namely, the total noninductive power $P(t)$ and the total plasma current $I(t)$. The control objective is therefore to find control inputs $P(t)$ and $I(t)$ that minimize the cost functional

$$J = \frac{1}{2} \int_0^1 \left| \iota(\hat{\rho}, t_f) - \iota^d(\hat{\rho}) \right|^2 d\hat{\rho} + \frac{1}{2} \int_{t_0}^{t_f} (\gamma_I I^2 + \gamma_P P^2)\, dt, \tag{16}$$

where $\iota^d(\hat{\rho})$ is the desired target profile at time t_f and the positive constants γ_I and γ_P are control weighting factors. The control actuators must satisfy the following constraints:

$$\text{Magnitude saturation}: \ I_l^{(0)} \le I \le I_u^{(0)} \quad P_l \le P \le P_u \tag{17}$$

$$\text{Rate saturation}: \ \left| \frac{dI(t)}{dt} \right| \le I_u^{(1)} \tag{18}$$

$$\text{Initial and final values}: \ I(t_0) = I_0 \quad I(t_f) = I_f. \tag{19}$$

We introduce two admissible control sets $\mathcal{P}$ and $\mathcal{I}$ for all $t \in [t_0, t_f]$ to denote the constraints (17)–(19):

$$\mathcal{P} = \{ P(t) | P_l \le P(t) \le P_u \} \tag{20}$$

$$\mathcal{I} = \left\{ I(t) \middle| I_l^{(0)} \le I(t) \le I_u^{(0)}, \left| \frac{dI(t)}{dt} \right| \le I_u^{(1)}, I(t_0) = I_0, I(t_f) = I_f \right\}. \tag{21}$$

III. POD METHOD

Given a collection of snapshots $\Psi = \{ \psi(\hat{\rho}, t_j) \} = \{ \psi_j(\hat{\rho}) \}$, $j = 1, 2, \ldots, n$, on the domain $0 \le \hat{\rho} \le 1$, the goal of the POD method (see, e.g., [16]) is to produce a set

of basis functions $V = \{v_1(\hat{\rho}), v_2(\hat{\rho}), \ldots, v_1(\hat{\rho})\}$ $(l \leq n)$ to optimally approximate any snapshot $\psi_j(\hat{\rho})$ in Ψ, i.e.,

$$\psi_j \approx \sum_{i=1}^{l} z_i v_i \tag{22}$$

where $z_i,\ i = 1, 2, \ldots, l$, are constants. We refer to Ψ as the data collection set and to V as the POD basis set.

For any two functions $f_i(\hat{\rho})$ and $f_j(\hat{\rho})$ in either Ψ or V, we define their inner product as $\langle f_i, f_j \rangle = \int_0^1 f_i f_j \hat{\rho} d\hat{\rho}$ and the induced norm of any function $f_i(\hat{\rho})$ as $\|f_i\| = \langle f_i, f_i \rangle = \int_0^1 f_i^2 \hat{\rho} d\hat{\rho}$. We multiply (22) on both sides by $\hat{\rho} v_k$ and integrate over $\hat{\rho} \in [0, 1]$ to obtain

$$\int_0^1 \psi_j v_k \hat{\rho} d\hat{\rho} \approx \sum_{i=1}^{l} z_i \int_0^1 v_i v_k \hat{\rho} d\hat{\rho}. \tag{23}$$

If the basis functions in V satisfy the orthonormal condition

$$\int_0^1 v_i v_k \hat{\rho} d\hat{\rho} = \begin{cases} 1, & i = k \\ 0, & i \neq k \end{cases} \tag{24}$$

we can obtain $z_k = \int_0^1 \psi_j v_k \hat{\rho} d\hat{\rho} = \langle \psi_j, \psi_k \rangle$ and write the snapshot approximation (22) as

$$\psi_j \approx \sum_{i=1}^{l} \langle \psi_j, v_i \rangle v_i. \tag{25}$$

The objective of the POD method is to find a set V that minimizes the error of the approximation (25), i.e.,

$$\begin{aligned} \min_{v_i \in V} J_b(v_1, \ldots, v_l) &= \sum_{j=1}^{n} \left\| \psi_j - \sum_{i=1}^{l} \langle \psi_j, v_i \rangle v_i \right\|^2 \\ \text{s.t. } \langle v_i, v_j \rangle &= \delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j. \end{cases} \end{aligned} \tag{26}$$

We first rewrite the cost functional (26) as

$$\begin{aligned} &J_b(v_1, \ldots, v_l) \\ &= \sum_{j=1}^{n} \left\langle \psi_j - \sum_{i=1}^{l} \langle \psi_j, v_i \rangle v_i, \psi_j - \sum_{i=1}^{l} \langle \psi_j, v_i \rangle v_i \right\rangle \\ &= \sum_{j=1}^{n} \left[\langle \psi_j, \psi_j \rangle - 2 \sum_{i=1}^{l} \langle \psi_j, v_i \rangle^2 + \sum_{i=1}^{l} \langle \psi_j, v_i \rangle^2 \right] \\ &= \sum_{j=1}^{n} \left[\langle \psi_j, \psi_j \rangle - \sum_{i=1}^{l} \langle \psi_j, v_i \rangle^2 \right]. \end{aligned} \tag{27}$$

Therefore, solving the minimization problem (26) is equivalent to solving the maximization problem

$$\begin{aligned} \max_{v_i \in V} J_B(v_1, \ldots, v_l) &= \sum_{j=1}^{n} \sum_{i=1}^{l} \langle \psi_j, v_i \rangle^2, \\ \text{s.t.} \langle v_i, v_j \rangle &= \delta_{ij}. \end{aligned}$$

By introducing

$$\mathcal{K}(\hat{\rho}, \hat{\rho}') = \sum_{j=1}^{n} \psi_j(\hat{\rho}) \psi_j(\hat{\rho}') \qquad \mathcal{R}v = \int_0^1 \mathcal{K}(\hat{\rho}, \hat{\rho}') v(\hat{\rho}') \hat{\rho}' d\hat{\rho}'$$

we can obtain

$$\begin{aligned} \langle \mathcal{R}v_i, v_i \rangle &= \int_0^1 \mathcal{R}v_i(\hat{\rho}) v_i(\hat{\rho}) \hat{\rho} d\hat{\rho} \\ &= \int_0^1 \int_0^1 \mathcal{K}(\hat{\rho}, \hat{\rho}') v_i(\hat{\rho}') \hat{\rho}' d\hat{\rho}' v_i(\hat{\rho}) \hat{\rho} d\hat{\rho} \\ &= \int_0^1 \int_0^1 \sum_{j=1}^{n} \psi_j(\hat{\rho}) \psi_j(\hat{\rho}') v_i(\hat{\rho}') \hat{\rho}' d\hat{\rho}' v_i(\hat{\rho}) \hat{\rho} d\hat{\rho} \\ &= \sum_{j=1}^{n} \left[\int_0^1 \psi_j(\hat{\rho}) v_i(\hat{\rho}) \hat{\rho} d\hat{\rho} \right]^2 = \sum_{j=1}^{n} \langle \psi_j, v_i \rangle^2 \end{aligned} \tag{28}$$

and rewrite J_B as $J_B(v_1, \ldots, v_l) = \sum_{i=1}^{l} \langle \mathcal{R}v_i, v_i \rangle$.

Therefore, for any POD basis function $v \in V$, we formulate the following optimization problem:

$$\max_{v} J_{\text{POD}} = \langle \mathcal{R}v, v \rangle \qquad \text{s.t. } \langle v, v \rangle = 1. \tag{29}$$

We define the Lagrange functional

$$L_{\text{POD}} = \langle \mathcal{R}v, v \rangle - \lambda(\langle v, v \rangle - 1) \tag{30}$$

where λ is a Lagrange multiplier and assume that $v = v^* + \eta v'$. Then, we can obtain

$$\begin{aligned} L_{\text{POD}}(\eta) &= \langle \mathcal{R}(v^* + \eta v'), (v^* + \eta v') \rangle \\ &\quad - \lambda \langle (v^* + \eta v'), (v^* + \eta v') \rangle + \lambda \\ &= \langle \mathcal{R}v^*, v^* \rangle + \eta \langle \mathcal{R}v^*, v' \rangle \\ &\quad + \eta \langle \mathcal{R}v', v^* \rangle + \eta^2 \langle \mathcal{R}v', v' \rangle \\ &\quad - \lambda \left[\langle v^*, v^* \rangle + \eta \langle v^*, v' \rangle + \eta \langle v', v^* \rangle + \eta^2 \langle v', v' \rangle \right] \\ &\quad + \lambda \end{aligned} \tag{31}$$

where η is an arbitrary real number and v' is an arbitrary variation with respect to the optimal solution $v^* \in V$. The optimality condition then becomes

$$\begin{aligned} \left. \frac{dL_{\text{POD}}(\eta)}{d\eta} \right|_{\eta=0} &= 2\langle \mathcal{R}v^*, v' \rangle - 2\langle \lambda v^*, v' \rangle \\ &= 2\langle \mathcal{R}v^* - \lambda v^*, v' \rangle = 0. \end{aligned} \tag{32}$$

Since v' is arbitrary, the optimality condition is reduced to the following eigenvalue problem:

$$\mathcal{R}v = \lambda v \quad \text{or} \quad \int_0^1 \mathcal{K}(\hat{\rho}, \hat{\rho}')v(\hat{\rho}')\,\hat{\rho}'d\hat{\rho}' = \lambda v(\hat{\rho}). \tag{33}$$

The integral equation (33) can be discretized in space to obtain a finite-dimensional eigenvalue problem. When the number of discrete points (m) is much larger than the number of snapshots (n), i.e., $m \gg n$, the discrete version of (33) generates a large-scale eigenvalue problem of dimension m. We follow instead an approach based on Sirovich's method [19], [16], which allows for the solution of the integral equation (33) based on an eigenvalue problem of dimension n. We assume that each POD basis function v can be expressed as a linear combination of the snapshots $\psi_j,\ j = 1, 2, \ldots, n$, i.e.,

$$v \sum_{k=1}^{n} a_k \psi k \tag{34}$$

which means that it is possible to find a combination of observation data (i.e., to determine the coefficients a_k) from which dominant characteristics can be extracted. We substitute now the snapshot expansion (34) into (33) to obtain

$$\int_0^1 \sum_{j=1}^{n} \psi_j(\hat{\rho})\psi_j(\hat{\rho}') \left[\sum_{k=1}^{n} a_k \psi_k(\hat{\rho}')\right] \hat{\rho}'d\hat{\rho}' = \lambda \sum_{j=1}^{n} a_j \psi_j(\hat{\rho})$$

or

$$\sum_{j=1}^{n} \left[\sum_{k=1}^{n} \int_0^1 \psi_j(\hat{\rho}')\psi_k(\hat{\rho}')\,\hat{\rho}'d\hat{\rho}' a_k\right] \psi_j(\hat{\rho}) = \lambda \sum_{j=1}^{n} a_j \psi_j(\hat{\rho}). \tag{35}$$

By introducing the following matrix notation:

$$C_{jk} = \int_0^1 \psi_j(\hat{\rho}')\psi_k(\hat{\rho}')\hat{\rho}'d\hat{\rho}' \quad \mathbf{a} = [a_1, a_2, \ldots, a_n]^{\mathrm{T}} \tag{36}$$

we can rewrite (35) as

$$\sum_{j=1}^{n} \left[\sum_{k=1}^{n} C_{jk}a_k - \lambda a_j\right] \psi_j(\hat{\rho}) = 0, \text{ i.e., } \mathbf{Ca} = \lambda \mathbf{a} \tag{37}$$

where $\mathbf{C} = [C_{jk}] \in \mathbf{R}^{n\times n}$. Since $\mathbf{C}$ is a nonnegative Hermitian matrix, i.e., $\mathbf{C} = \mathbf{C}^{\mathrm{T}}$, it has a complete set of orthogonal eigenvectors $(\mathbf{a}_1, \ldots, \mathbf{a}_n)$. Each POD basis function can be expressed then as

$$v_i = [\psi_1, \ldots, \psi_n]\mathbf{a}_i, \qquad i = 1, 2, \ldots, l. \tag{38}$$

IV. POD/Galerkin Method

We let $v(\hat{\rho})$ be a testing function (or trial function) which has at least first-order derivative with respect to the spatial coordinate $\hat{\rho}$. We multiply both sides of (10) by $\hat{\rho}v(\hat{\rho})$ and integrate over $\hat{\rho} \in [0, 1]$ to obtain the weak form of the magnetic flux diffusion equation

$$\begin{aligned}
&\int_0^1 \frac{1}{\vartheta_1(\hat{\rho})}\psi(\hat{\rho})v(\hat{\rho})\hat{\rho}d\hat{\rho} \\
&= u_1(t)\hat{\rho}D(\hat{\rho})\frac{\partial\psi}{\partial\hat{\rho}}v(\hat{\rho})\Big|_0^1 - u_1(t)\int_0^1 D(\hat{\rho})\frac{\partial\psi}{\partial\hat{\rho}}\frac{\partial v}{\partial\hat{\rho}}\hat{\rho}d\hat{\rho} \\
&\quad + u_2(t)\int_0^1 \vartheta_2(\hat{\rho})v(\hat{\rho})\hat{\rho}d\hat{\rho} \\
&= kD(1)v(1)\bar{n}^{1.5}I^{-0.5}P^{-0.75} \\
&\quad - \bar{n}^{1.5}I^{-1.5}P^{-0.75}\int_0^1 D(\hat{\rho})\frac{\partial\psi}{\partial\hat{\rho}}\frac{\partial v}{\partial\hat{\rho}}\hat{\rho}d\hat{\rho} \\
&\quad + P^{0.5}I^{-1}\int_0^1 \vartheta_2(\hat{\rho})v(\hat{\rho})\hat{\rho}d\hat{\rho}.
\end{aligned} \tag{39}$$

We assume that the magnetic flux can be spanned by the POD basis functions $v_j\ (j = 1, 2, \ldots, l)$, i.e.,

$$\psi(\hat{\rho}, t) \approx \sum_{j=1}^{l} z_j(t)v_j(\hat{\rho}). \tag{40}$$

We substitute (40) into the weak form (39) and adopt the POD basis functions as testing functions, i.e., $v = v_i,\ i = 1, 2, \ldots, l$, to write

$$\begin{aligned}
&\sum_{j=1}^{l}\left[\int_0^1 \frac{1}{\vartheta_1(\hat{\rho})}v_j(\hat{\rho})v_i(\hat{\rho})\hat{\rho}d\hat{\rho}\right] z_j(t) \\
&\quad = kD(1)v_i(1)\bar{n}^{1.5}I^{-0.5}P^{-0.75} \\
&\qquad - \bar{n}^{1.5}I^{-1.5}P^{-0.75}\sum_{j=1}^{l}\left[\int_0^1 D(\hat{\rho})\frac{\partial v_j}{\partial\hat{\rho}}\frac{\partial v_i}{\partial\hat{\rho}}\hat{\rho}d\hat{\rho}\right] z_j(t) \\
&\qquad + P^{0.5}I^{-1}\int_0^1 \vartheta_2(\hat{\rho})v_i(\hat{\rho})\hat{\rho}d\hat{\rho}.
\end{aligned} \tag{41}$$

By introducing the matrix notation

$$M_{ij} = \int_0^1 \frac{1}{\vartheta_1(\hat{\rho})}v_i(\hat{\rho})v_j(\hat{\rho})\hat{\rho}d\hat{\rho} \tag{42}$$

$$K_{ij} = -\bar{n}^{1.5}\int_0^1 D(\hat{\rho})\frac{\partial v_j}{\partial\hat{\rho}}\frac{\partial v_i}{\partial\hat{\rho}}\hat{\rho}d\hat{\rho} \tag{43}$$

$$b_i = kD(1)v_i(1)\bar{n}^{1.5} \quad c_i = \int_0^1 \vartheta_2(\hat{\rho})v_i(\hat{\rho})\hat{\rho}d\hat{\rho} \tag{44}$$

we can rewrite (41) as an l-dimensional matrix system

$$\begin{aligned}\mathbf{M}\frac{d\mathbf{z}(t)}{dt} &= \mathbf{F}(\mathbf{z}, P, I)\\ &= I^{-1.5}P^{-0.75}\mathbf{K}(t)\mathbf{z}(t)\\ &\quad + \mathbf{b}(t)I^{-0.5}P^{-0.75} + \mathbf{c}P^{0.5}I^{-1}\end{aligned} \tag{45}$$

where $\mathbf{z} = [z_i] \in \mathbf{R}^{l\times 1}$, $\mathbf{M} = [M_{ij}] \in \mathbf{R}^{l\times l}$, $\mathbf{K} = [K_{ij}] \in \mathbf{R}^{l\times l}$, $\mathbf{b} = [b_i] \in \mathbf{R}^{l\times 1}$, $\mathbf{c} = [c_i] \in \mathbf{R}^{l\times 1}$, and $\mathbf{F} : \mathbf{R}^{l\times 1} \otimes \mathcal{P} \otimes \mathcal{I} \rightarrow \mathbf{R}^{l\times 1}$ ($\otimes$ denotes the Cartesian product). P and I represent the admissible sets of control inputs P and I defined in (20) and (21), respectively.

By noting that the initial condition (13) can be rewritten as

$$\psi_0(\hat{\rho}) = \sum_{j=1}^{l} z_j(0)v_j(\hat{\rho}) \tag{46}$$

we multiply both sides of (46) by $\hat{\rho}v_i(\hat{\rho})$ and integrate over $0 \leq \hat{\rho} \leq 1$ to obtain

$$\int_0^1 \psi_0(\hat{\rho})v_i(\hat{\rho})\hat{\rho}d\hat{\rho} = \sum_{j=1}^{l} z_j(0)\int_0^1 v_j(\hat{\rho})v_i(\hat{\rho})\hat{\rho}d\hat{\rho}. \tag{47}$$

We introduce the matrix notation

$$f_i = \int_0^1 \psi_0(\hat{\rho})v_i(\hat{\rho})\hat{\rho}d\hat{\rho} \tag{48}$$

$$G_{ij} = \int_0^1 v_i(\hat{\rho})v_j(\hat{\rho})\hat{\rho}d\hat{\rho} \quad z_{0,j} = z_j(0) \tag{49}$$

to rewrite (47) as a matrix formulation

$$\mathbf{f} = \mathbf{G}\mathbf{z}_0 \tag{50}$$

where $\mathbf{f} = [f_i] \in \mathbf{R}^{l\times 1}$, $\mathbf{G} = [G_{ij}] \in \mathbf{R}^{l\times 1}$, and $\mathbf{z}_0 = [z_{0,j}] \in \mathbf{R}^{l\times 1}$. Therefore, the Galerkin projection coefficients $z_j(0)$, $j = 1, 2, \ldots, l$, in (46) can be obtained by solving the linear algebraic equation (50)

$$\mathbf{z}_0 = \mathbf{G}^{-1}\mathbf{f}. \tag{51}$$

We rewrite now the output variable ι defined in (15) in terms of the Galerkin expansion (40), i.e., $\iota(\hat{\rho}, t) = \sum_{j=1}^{l} z_j(t)\frac{\partial v_j}{\partial \hat{\rho}}$. Similarly, the desired output variable ι^d can also be rewritten as

$$\iota^d(\hat{\rho}) = \sum_{j=1}^{l} z_j^d \frac{\partial v_j}{\partial \hat{\rho}} \tag{52}$$

where z_j^d, $j = 1, 2, \ldots, l$, are the desired Galerkin projection coefficients in terms of the POD basis functions $v_j(\hat{\rho})$, $j = 1, 2, \ldots, l$. To obtain the specific values of z_j^d, $j = 1, 2, \ldots, l$, we can multiply both sides of (52) by $\hat{\rho}\frac{\partial v_i}{\partial \hat{\rho}}$ and integrate over $0 \leq \hat{\rho} \leq 1$ to obtain

$$\int_0^1 \iota^d(\hat{\rho})\frac{\partial v_i}{\partial \hat{\rho}}\hat{\rho}d\hat{\rho} = \sum_{j=1}^{l} z_j^d \int_0^1 \frac{\partial v_j}{\partial \hat{\rho}}\frac{\partial v_i}{\partial \hat{\rho}}\hat{\rho}d\hat{\rho}. \tag{53}$$

By introducing the matrix notation

$$g_i = \int_0^1 \iota^d(\hat{\rho})\frac{\partial v_i}{\partial \hat{\rho}}\hat{\rho}d\hat{\rho} \quad H_{ij} = \int_0^1 \frac{\partial v_j}{\partial \hat{\rho}}\frac{\partial v_i}{\partial \hat{\rho}}\hat{\rho}d\hat{\rho} \tag{54}$$

we can rewrite (53) as

$$\mathbf{g} = \mathbf{H}\mathbf{z}^d \tag{55}$$

where $\mathbf{g} = [g_i] \in \mathbf{R}^{l\times 1}$, $\mathbf{H} = [H_{ij}] \in \mathbf{R}^{l\times l}$, and $\mathbf{z}^d = [z_j^d] \in \mathbf{R}^{l\times 1}$. Therefore, the Galerkin projection coefficients of the desired profile can be obtained by solving the algebraic equation (55)

$$\mathbf{z}^d = \mathbf{H}^{-1}\mathbf{g}. \tag{56}$$

Finally, we can rewrite the optimization problem (16)–(21) subject to (10) in an l-dimensional POD subspace $V = \text{span}\{v_1, v_2, \ldots v_l\}$

$$\begin{aligned}\min_{P\in\mathcal{P}, I\in\mathcal{I}} J_G &= [\mathbf{z}(t_f) - \mathbf{z}^d]^{\mathrm{T}}[\mathbf{z}(t_f) - \mathbf{z}^d]\\ &\quad + \frac{1}{2}\int_{t_0}^{t_f} (\gamma_I I^2 + \gamma_P P^2)\, dt,\end{aligned} \tag{57}$$

$$\text{s.t.}\frac{d\mathbf{z}(t)}{dt} = \mathbf{M}^{-1}\mathbf{F}(\mathbf{z}, P, I),\ \mathbf{z}(0) = \mathbf{z}_0. \tag{58}$$

V. Numerical Optimization

A. Control Parameterization

We use function values $P(t_k)$ and $I(t_k)$ at each time point t_k over the time grid

$$t_0 = t_1 \leq t_2 \leq \cdots \leq t_k \leq \cdots \leq t_T = t_f \tag{59}$$

to parameterize the control inputs $P(t)$ and $I(t)$ for all $t \in [t_0, t_f]$. For $t \in (t_k, t_{k+1})$, $k = 1, 2, \ldots, T-1$, we use linear interpolations to approximate $P(t)$ and $I(t)$ as

$$P^{\text{lin}}(t) = P(t_k) + \frac{P(t_{k+1}) - P(t_k)}{t_{k+1} - t_k}(t - t_k) \tag{60}$$

$$I^{\text{lin}}(t) = It_k) + \frac{I(t_{k+1}) - I(t_k)}{t_{k+1} - t_k}(t - t_k). \tag{61}$$

We denote $P(t_k) = P_k$ and $I(t_k) = I_k$ for all $k = 1, 2, \ldots, T-1$. Then, the constraints (20)–(21) become

$$\begin{aligned}P_l &\leq P_k + \frac{P_{k+1} - P_k}{t_{k+1} - t_k}(t - t_k)\\ &\leq P_u \Rightarrow P_l \leq P_k \leq P_u\end{aligned} \tag{62}$$

$$\begin{aligned}I_l^{(0)} &\leq I_k + \frac{I_{k+1} - I_k}{t_{k+1} - t_k}(t - t_k)\\ &\leq I_u^{(0)} \Rightarrow I_l^{(0)} \leq I_k \leq I_u^{(0)}\end{aligned} \tag{63}$$

$$\begin{aligned}\left|\frac{I_{k+1} - I_k}{t_{k+1} - t_k}\right| &\leq I_u^{(1)} \Rightarrow -I_u^{(1)}\\ &\leq \frac{I_{k+1} - I_k}{t_{k+1} - t_k} \leq I_u^{(1)}\end{aligned} \tag{64}$$

$$I_1 = I_0 \quad I_T = I_f. \tag{65}$$

To take into account the control parameterization, the admissible sets $\mathcal{P}$ and $\mathcal{I}$ are redefined as

$$\mathcal{P}^{\text{lin}} = \{P_k, (k = 0, 1, \ldots, T) \text{ satisfies } (62)\} \quad (66)$$
$$\mathcal{I}^{\text{lin}} = \{I_k, (k = 0, 1, \ldots, T) \text{ satisfies } (63)-(65)\}. \quad (67)$$

B. Nonlinear Optimization

After the parameterization, the control inputs can be represented as functions of time and the to-be-optimized parameters P_k and I_k, for $k = 1, 2, \ldots, T$, i.e.,

$$P^{\text{lin}}(t) = P^{\text{lin}}(P_1, \ldots, P_T, t) \quad I^{\text{lin}}(t) = I^{\text{lin}}(I_1, \ldots, I_T, t).$$

We define a parameterization vector containing all the to-be-optimized parameters $U = (P_1, \ldots, P_T, I_1, \ldots, I_T)^{\text{T}} \in \mathcal{U} \in R^{2\text{T}\times 1}$, where $\mathcal{U} = \mathcal{P}^{\text{lin}} \cup \mathcal{I}^{\text{lin}}$.

With U denoting the to-be-optimized parameters and $\mathcal{U}$ denoting the admissible parameter set, we can rewrite the optimization problem (57)–(58) into a general nonlinear optimization formulation

$$\min_{U\in U} J_G(\mathbf{z}, U) \quad (68)$$
$$\text{s.t. } \frac{d\mathbf{z}(t)}{dt} = \mathbf{M}^{-1}\mathbf{F}(\mathbf{z}, U), \qquad \mathbf{z}(0) = \mathbf{z}_0. \quad (69)$$

C. SQP

The SQP algorithm is an effective approach to find the local minima of the NLP problem formulated by (68)–(69). SQP is an iterative method which solves, at each iteration, a QP problem and is an effective approach to obtain local minimal solutions with superlinear convergence. The basic idea of SQP is to generate a quadratic subproblem whose solution yields a step toward the solution of the original nonlinear problem. We summarize below the SQP iterative method for a general NLP problem

$$\text{NLP}: \begin{cases} \min\limits_{X} \mathcal{F}(X) \\ \text{s.t.} \quad e(X) = 0 \end{cases} \quad (70)$$

where $\mathcal{F}$ and e have continuous second derivatives with respect to X. We note that inequality constraints (e.g., $e(X) \le 0$) can be converted into equalities by introducing a slack variable (vector) ϵ such that $e(X) + \epsilon = 0$. Then, in order to simplify the explanation of the SQP method, we only consider NLP with equality constraints in this section. If λ^* is the Lagrange multiplier corresponding to a local minimizer X^* of (70), the Lagrangian $\mathcal{L}(X, \lambda) = \mathcal{F}(X) - \lambda^{\text{T}} e(X)$ satisfies $\mathcal{L}(X, \lambda^*) = \mathcal{F}(X)$ for all admissible X's. Then, we can equivalently rewrite the constrained NLP problem (70) as

$$\text{NLP}: \begin{cases} \min\limits_{X} \mathcal{L}(X, \lambda^*) = \mathcal{F}(X) - (\lambda^*)^{\text{T}} e(X) \\ \text{s.t.} \quad e(X) = 0. \end{cases} \quad (71)$$

The first-order optimality condition for (71) is given by the following nonlinear equations:

$$\begin{cases} \nabla_X \mathcal{L}(X^*, \lambda^*) = \nabla\mathcal{F}(X^*) - \nabla e(X^*)\lambda^* = 0 \\ \nabla_\lambda \mathcal{L}(X^*, \lambda^*) = e(X^*) = 0. \end{cases} \quad (72)$$

One possible approach to the solution of this nonlinear problem is to assume that we have an iteration

$$\left(X^{(k+1)}, \lambda^{(k+1)}\right) = \left(X^{(k)}, \lambda^{(k)}\right) + \left(p^{(k)}, \omega^{(k)}\right) \quad (73)$$

converging to (X^*, λ^*) defined by (72), where $(X^{(k)}, \lambda^{(k)})$ is the current estimate of (X^*, λ^*). Assuming that $(X^{(k)}, \lambda^{(k)})$ is close enough to (X^*, λ^*), the optimality condition (72) can be linearized around the current estimate $(X^{(k)}, \lambda^{(k)})$, i.e., (74), as shown at the bottom of the page. By noting from (72) that

$$\begin{cases} \nabla^2_{X\lambda} \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) \omega^{(k)} = -\nabla e\left(X^{(k)}\right) \\ \nabla^2_{\lambda} \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) = -e\left(X^{(k)}\right) \end{cases} \quad (75)$$

it is possible to rewrite (74) into a compact matrix form, from which $(p^{(k)}, \omega^{(k)})$ can be obtained ($\nabla = \nabla_X$)

$$\begin{pmatrix} \nabla^2 \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) & -\nabla e\left(X^{(k)}\right) \\ -\left(\nabla e\left(X^{(k)}\right)\right)^{\text{T}} & 0 \end{pmatrix} \begin{pmatrix} p^{(k)} \\ \omega^{(k)} \end{pmatrix} = -\begin{pmatrix} \nabla \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) \\ -e\left(X^{(k)}\right) \end{pmatrix}. \quad (76)$$

It is interesting to note that (76) is the first-order optimality condition of the QP problem which represents a quadratic approximation of $\mathcal{L}$ and a linear approximation of e in (71) around the current solution $(X^{(k)}, \lambda^{(k)})$. Therefore, from the sequence of QP subproblems (77), shown at the bottom of the page, it is possible to obtain search directions for the general NLP problem (71), where each QP subproblem (77) minimizes a quadratic approximation of the original Lagrangian subject to a linear approximation of the constraints.

$$\begin{cases} \nabla_X \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) + \nabla^2_X \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) p^{(k)} + \nabla^2_{X\lambda} \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) \omega^{(k)} = 0 \\ \nabla_\lambda \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) + \nabla^2_{\lambda X} \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) p^{(k)} + \nabla^2_{\lambda} \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) \omega^{(k)} = 0 \end{cases} \quad (74)$$

$$\text{QP}^{(k)}: \begin{cases} \min\limits_{p^{(k)}} \quad \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) + \nabla \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right)^{\text{T}} p^{(k)} + \frac{1}{2}\left[p^{(k)}\right]^{\text{T}} \nabla^2 \mathcal{L}\left(X^{(k)}, \lambda^{(k)}\right) p^{(k)} \\ \text{s.t.} \quad e\left(X^{(k)}\right) + \nabla e^{\text{T}}\left(X^{(k)}\right) p^{(k)} = 0 \end{cases} \quad (77)$$

D. Iteration of POD and SQP Methods

The POD model reduction technique captures the most dominant modes of the system, which usually depend on the input excitation. Since, during the numerical optimization procedure, the control inputs are changed as the parameters are optimized, it is a natural idea to combine the POD model reduction method and the SQP optimization procedure. We use the control inputs obtained during the last stage of the optimization procedure to generate the data ensemble employed for the extraction of the POD modes. These POD modes are then used to implement a Galerkin projection that provides a low-dimensional model approximating the original high-dimensional PDE system. The low-dimensional model, which can dramatically reduce computational effort, is used for a new stage of the optimization procedure that provides updated control inputs. The overall algorithm blends the POD model reduction method and the SQP optimization procedure in an iterative fashion. Such algorithm can be summarized as follows.

1) Set $k = 0$ to give the initial guess for the control sequences $I^{(k)}$ and $P^{(k)}$.
2) Simulate the PDE system (10) to obtain data ensemble Ψ^k.
3) Generate the POD modes $V_{\mathrm{POD}}^{(k)}$ and the LDDS (45) from the data ensemble.
4) Solve the optimization problem (68)–(69) using the SQP approach.
5) Go back to Step 2) and stop the iteration if the data ensemble satisfies

$$\left\| \Psi^{(k+1)} - \Psi^{(k)} \right\| \leq \varepsilon. \tag{78}$$

Otherwise, continue the iteration until the error criterion is satisfied.

Although a formal proof of convergence is out of the scope of this paper, this algorithm has proven, in practice, to be always convergent for the considered number of parameterization points.

VI. Numerical Illustrations

The geometrical parameters $D(\hat{\rho})$, $\vartheta_1(\hat{\rho})$, and $\vartheta_2(\hat{\rho})$ in model (10) are shown in Figs. 2–4, respectively. The time evolutions of the average density $\bar{n}(t)$, total power $P(t)$, and total current $I(t)$ for DIII-D shot #129412 are shown in Fig. 5. These experimental time evolutions for the total power $P(t)$ and the total current $I(t)$ are used as the initial guesses for the optimization procedure. Since the evolution for the average density $\bar{n}(t)$ extracted from DIII-D shot #129412 has been proved to be feasible for the considered scenario, we assume in this paper that such time evolution can be reproduced by a dedicated controller in any shot under the considered scenario, and therefore, we consider it as the measurable input required in our model. The integration of a more sophisticated predictive code within the optimization procedure proposed in this paper could provide a prediction of the (average) density and relax the requirement of regulating it around a predefined trajectory. In addition, it is also possible to include the average density as one of the to-be-optimized actuator trajectories in the optimization procedure proposed in this paper (with the risk of not being able to accurately reproduce the optimized trajectory in experiments). The initial magnetic flux profile in shot #129412 shown in Fig. 6 is used as the initial condition for the numerical simulations carried out as part of the optimization procedure.

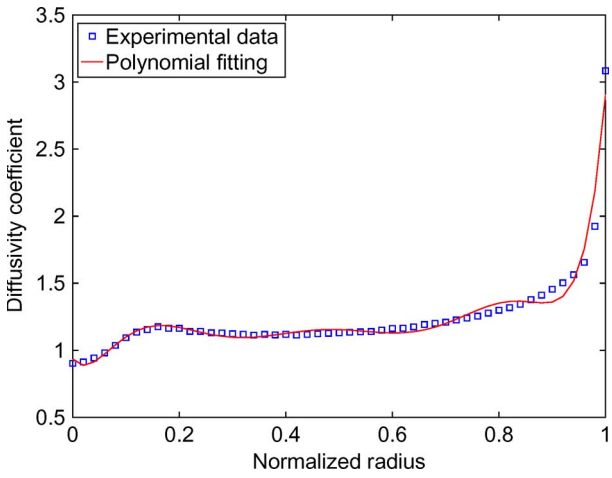

Fig. 2. Diffusivity coefficient D versus the normalized radius $\hat{\rho}$.

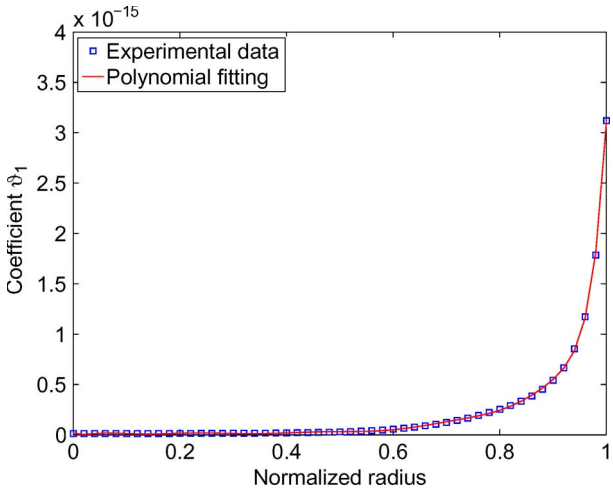

Fig. 3. Shape of the coefficient ϑ_1 versus the normalized radius $\hat{\rho}$.

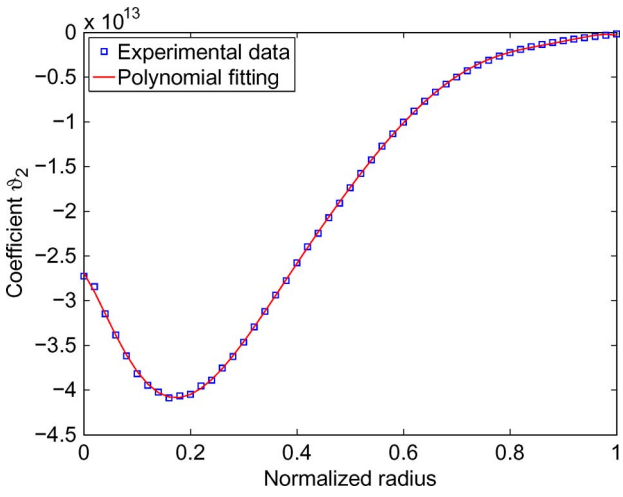

Fig. 4. Shape of the coefficient ϑ_1 versus the normalized radius $\hat{\rho}$.

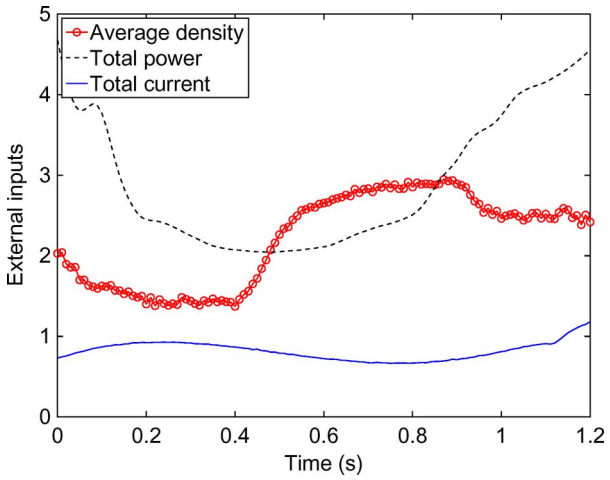

Fig. 5. Evolution of variables $\bar{n}$ (10^{19} m^{-3}) and I (in megaamperes) versus time t (shot #129412).

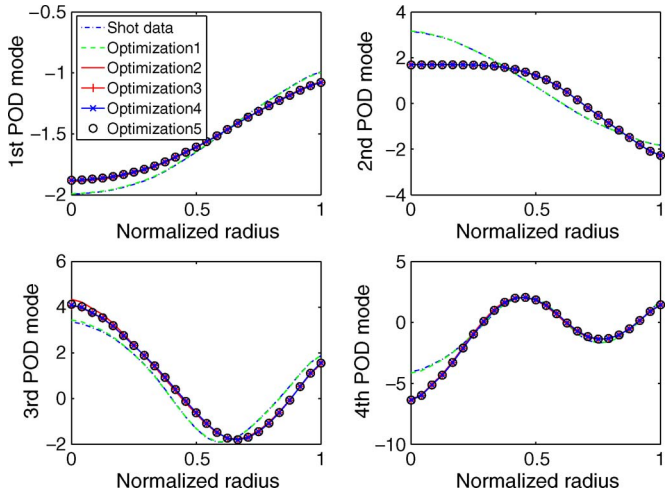

Fig. 7. First four POD modes as a function of the iteration number.

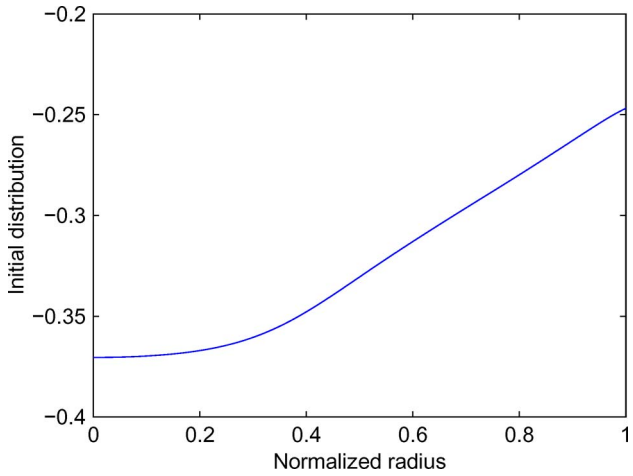

Fig. 6. Initial distribution ψ_0 (in webers) versus the normalized radius $\hat{\rho}$.

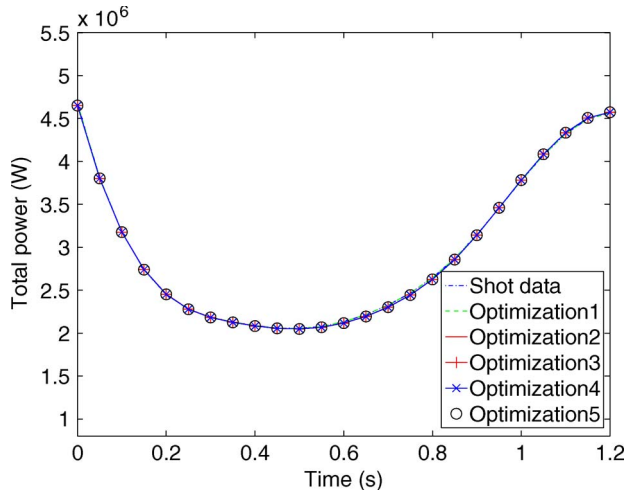

Fig. 8. Evolution of P [in watts] as a function of time t [in seconds].

We parameterize the control input functions by their values at eight equidistant time points $(t_1, t_2, \ldots, t_8)$, i.e.,

$$P(t_{(k)}) = P_k \quad I(t_{(k)}) = I_k, \qquad k = 1, 2, \ldots, 8. \tag{79}$$

The computational grid used for solving the magnetic flux diffusion PDE has $M = 25$ spatial and $N = 25$ temporal points for the domains defined by $0 \leq \hat{\rho} \leq 1$ and $0 \leq t \leq 1.2$ s, respectively. The following constraints are considered: $P_l = 0.2$ MW, $P_u = 10$ MW, $I_l^{(0)} = 0.15$ MA, $I_{u^{(0)}} = 1.19$ MA, $I_u^{(1)} = 5$ MA/s, $I_0 = 0.709$ MA, and $I_f = 1.187$ MA. The error tolerance in the termination condition (78) is chosen as $\varepsilon = 10^{-4}$.

We first generate through simulations the data ensemble required to obtain the POD modes. Let $\lambda_1 > \cdots > \lambda_l > \cdots > \lambda_d > 0$ denote the positive eigenvalues of the correlation matrix $\mathbf{C}$, where $d = \text{rank}(\mathbf{C})$. The error (energy ratio) associated with the approximation with the first l POD modes is

$$\varepsilon_l = \sum_{j=1}^{n} \left\| \psi_j - \sum_{i=1}^{l} \langle \psi_j, v_i \rangle v_i \right\|^2 = \sum_{k=l+1}^{d} \lambda_k. \tag{80}$$

In this case, the eigenvalues of the first four POD modes obtained from the simulated data are $\lambda_1 = 9.96790964 \times 10^{-1}$, $\lambda_2 = 3.121034 \times 10^{-3}$, $\lambda_3 = 7.89143 \times 10^{-5}$, and $\lambda_4 = 8.63946 \times 10^{-6}$. This implies that just four POD modes capture more than 99% of the system energy, or equivalently, that by using just four POD modes, we can approximate the original PDE system with less than 1% error. Fig. 7 shows how the first four POD modes evolve as the iterations of the proposed optimization procedure take place.

The four POD modes are employed to obtain an $l = 4$ dimensional model which is, in turn, used to carry out the numerical optimization. The proposed algorithm satisfies the termination condition after five iterations. The optimized actuator trajectories are shown in Figs. 8 and 9 as a function of the iteration number. As the iterations take place, we note that the sequence of actuator trajectories converge (compare optimizations 4 and 5). From Fig. 8, we can conclude that the initial guess for the total power (labeled "shot data" in the figure) seems to be very close to a converged optimal solution. On the contrary, Fig. 9 shows that the initial guess for the total current is far from optimal and that the iterative optimization procedure converges to such optimum. From Fig. 10, the matching-error reduction

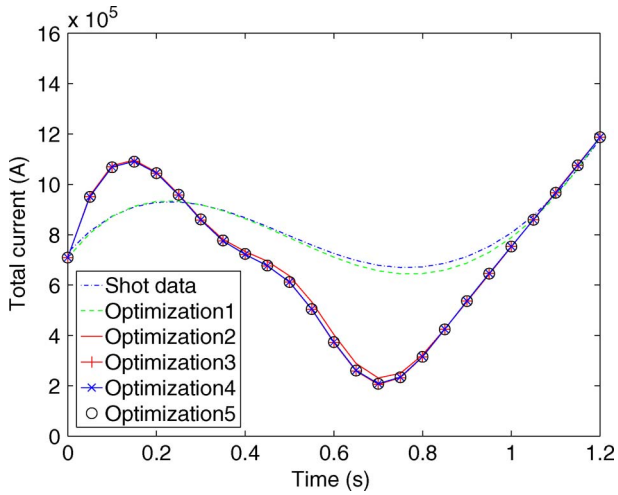

Fig. 9. Evolution of I [in amperes] as a function of time t [in seconds].

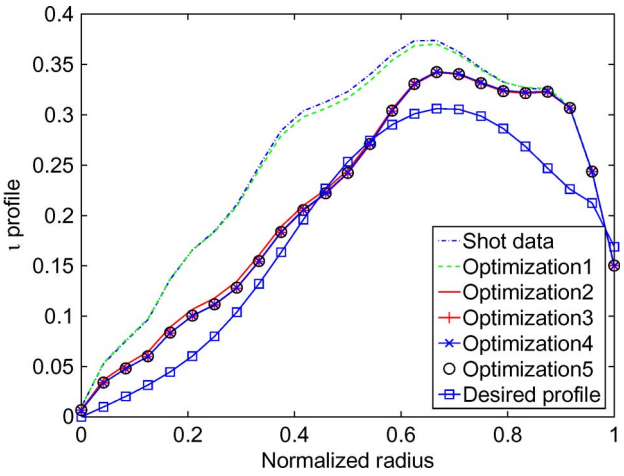

Fig. 10. Spatial ι-profile versus the normalized radius $\hat{\rho}$ at $t_f = 1.2$ s.

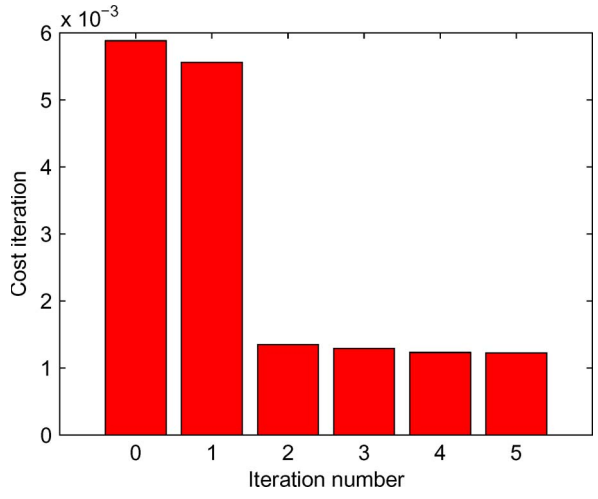

Fig. 11. Cost functional values in terms of iterations.

can be appreciated as the optimization procedure progresses. The monotonic decrease of the cost function value is shown in Fig. 11 as a function of the iteration number. As new optimized trajectories are obtained, iteration after iteration, new POD models are generated to carry out a new SQP optimization

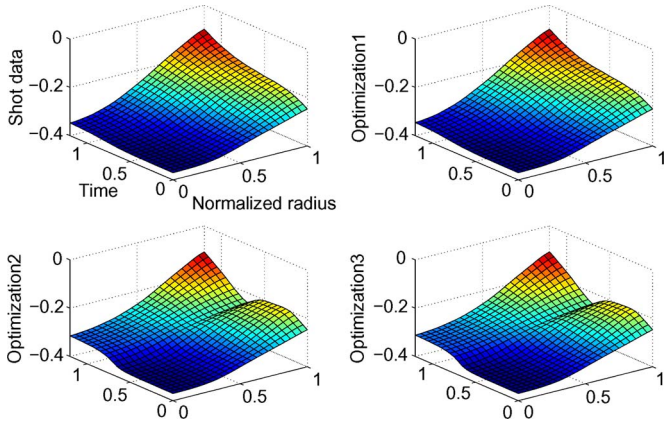

Fig. 12. Magnetic flux evolution ψ_0 [in webers] with the optimized inputs.

until acceptable convergence is achieved. The combination of POD model reduction and SQP-based NLP is what we call an optimization iteration in the figures. Fig. 12 shows the spatial–temporal evolutions of the PDE system as a function of the optimization iteration.

For the simulation study presented in this section, the proposed algorithm was implemented in MATLAB and run in a personal computer with the following configuration: Pentium T4200 Dual-Core 2.00-GHz CPU, 3.00-GB RAM, and 32-b Windows XP Operating System. The computation time depends on the number of POD modes used for the model reduction, the spatiotemporal discretization used for the numerical data generation, the control parameterization, the settings of the SQP algorithm, and the error tolerance of the proposed iterative algorithm. Based on an energy analysis, we have found that using four POD modes is accurate enough to approximate the original PDE system. Our numerical studies show that the use of a higher dimensional model for the SQP optimization increases the computational complexity without a comparable increase of accuracy. The choice of the grid size for a numerical simulation is always a tradeoff between accuracy and computational burden. In this case, the numerical simulation over the $M \times N$ discretization grid is carried out with the only purpose of generating the data from which the l POD modes are extracted. Once the l number of POD modes is defined based on an energy analysis, we increase the size of the $M \times N$ discretization grid until the first l number of POD modes no longer change. In our simulation study, increasing the grid beyond 25×25 will not produce any significant change in the first four POD modes. The only effect will indeed be the increase of the overall computational time. The number of control parameters is a design choice. As we increase this number, the algorithm has more degrees of freedom to minimize the cost function. However, a higher price in terms of computational burden must be paid. For instance, it takes around 150 s to optimize sixteen parameters (eight for each control input), while it takes less than 60 s to complete the optimization based on 12 parameters (six for each control input) without a significant change in the achieved value of the cost function.

The POD–SQP iterative optimization procedure can dramatically accelerate the numerical computations. A relatively large-scale discretization on an $M \times N$ grid is only used to generate

the data from which the POD modes are obtained. The SQP optimization is carried out instead based on a low-dimensional model of order l. The achieved computation time based on the configuration and settings described previously is already small enough to enable the solution of the numerical optimization problem between two plasma discharges. However, there is extensive room for improvement. For instance, the computation time could be drastically reduced by implementing the algorithm in C++, by running the algorithm in a multiprocessor machine, and by choosing the appropriate optimization settings. A study on the feasibility of implementing a receding horizon control strategy based on the proposed optimization algorithm is underway. The term receding-horizon control, or model predictive control, describes a class of algorithms that, at each control interval, computes an open-loop sequence of manipulated input variables in order to optimize the future behavior of the plant over a specific time horizon [20]–[22]. The feedback feature of the receding horizon controller may reduce the accuracy requirements on the optimization, relaxing, in this way, the computation-time needs. Even a control interval of several seconds may enable the implementation of a receding-horizon controller in long-discharge tokamaks like ITER.

VII. Conclusion and Future Work

The open-loop finite-time optimal current-profile-control problem arising in tokamak plasmas during the ramp-up phase of the discharge is solved by using POD model reduction and NLP. The proposed numerical optimization procedure alternates POD model reduction and SQP in an iterative scheme. The SQP technique is an effective approach for the solution of the NLP problem derived from the original optimal current-profile-control problem. Numerical studies demonstrate that the iterative POD-SQP optimization procedure is efficient and reduces computational effort. Simulation results show that the numerical optimization procedure can generate control trajectories driving the final ι-profile within the proximity of a predefined desired profile. The POD–SQP optimization procedure is characterized by high-speed computation and shows potential for real-time implementation in a closed-loop receding-horizon scheme, particularly for long-discharge tokamaks such as ITER.

References

[1] T. Taylor, "Physics of advanced tokamaks," *Plasma Phys. Control. Fusion*, vol. 39, no. suppl. 12B, pp. B47–B73, Dec. 1997.

[2] M. Murakami, M. R. Wade, C. M. Greenfield, T. C. Luce, J. R. Ferron, H. E. St. John, J. C. DeBoo, W. W. Heidbrink, Y. Luo, M. A. Makowski, T. H. Osborne, C. C. Petty, P. A. Politzer, S. L. Allen, M. E. Austin, K. H. Burrell, T. A. Casper, E. J. Doyle, A. M. Garofalo, P. Gohil, I. A. Gorelov, R. J. Groebner, A. W. Hyatt, R. J. Jayakumar, K. Kajiwara, C. E. Kessel, J. E. Kinsey, R. J. La Haye, L. L. Lao, A. W. Leonard, J. Lohr, T. W. Petrie, R. I. Pinsker, R. Prater, T. L. Rhodes, A. C. C. Sips, G. M. Staebler, T. S. Taylor, M. A. Vanzeeland, G. Wang, W. P. West, and L. Zeng, "Progress toward fully noninductive, high beta conditions in DIII-D," *Phys. Plasmas*, vol. 13, no. 5, pp. 056 106:1–056 106:9, May 2006.

[3] F. Hinton and R. Hazeltine, "Theory of plasma transport in toroidal confinement systems," *Rev. Mod. Phys.*, vol. 48, no. 2, pp. 239–308, Apr. 1976.

[4] D. Moreau, F. Crisanti, X. Litaudon, D. Mazon, P. De Vries, R. Felton, E. Joffrin, L. Laborde, M. Lennholm, A. Murari, V. Pericoli-Ridolfini, M. Riva, T. Tala, G. Tresset, L. Zabeo, and K. D. Zastrow, "Real-time control of the q-profile in JET for steady state advanced tokamak operation," *Nucl. Fusion*, vol. 43, no. 9, pp. 870–882, Sep. 2003.

[5] L. Laborde, D. Mazon, D. Moreau, A. Murari, R. Felton, L. Zabeo, R. Albanese, M. Ariola, J. Bucalossi, F. Crisanti, M. de Baar, G. de Tommasi, P. de Vries, E. Joffrin, M. Lennholm, X. Litaudon, A. Pironti, T. Tala, and A. Tuccillo, "A model-based technique for integrated real-time profile control in the JET tokamak," *Plasma Phys. Control. Fusion*, vol. 47, no. 1, pp. 155–183, Jan. 2005.

[6] D. Moreau, D. Mazon, M. Ariola, G. De Tommasi, L. Laborde, F. Piccolo, F. Sartori, T. Tala, L. Zabeo, A. Boboc, E. Bouvier, M. Brix, J. Brzozowski, C. D. Challis, V. Cocilovo, V. Cordoliani, F. Crisanti, E. De La Luna, R. Felton, N. Hawkes, R. King, X. Litaudon, T. Loarer, J. Mailloux, M. Mayoral, I. Nunes, E. Surrey, and O. Zimmerman, "A two-time-scale dynamic-model approach for magnetic and kinetic profile control in advanced tokamak scenarios on JET," *Nucl. Fusion*, vol. 48, no. 10, p. 106 001, Oct. 2008.

[7] J. R. Ferron, P. Gohil, C. M. Greenfield, J. Lohr, T. C. Luce, M. A. Makowski, D. Mazon, M. Murakami, C. C. Petty, P. A. Politzer, and M. R. Wade, "Feedback control of the safety factor profile evolution during formation of an advanced tokamak discharge," *Nucl. Fusion*, vol. 46, no. 10, pp. L13–L17, Oct. 2006.

[8] T. Suzuki, A. Isayama, S. Ide, T. Fujita, T. Oikawa, S. Sakata, M. Sueoka, H. Hosoyama, and M. Seki, "Recent RF experiments and application of RF waves to real-time control of safety factor profile in JT-60U," in *Proc. AIP Conf.*, 2005, vol. 787, pp. 279–286.

[9] T. Wijnands, D. Van Houtte, G. Martin, X. Litaudon, and P. Froissard, "Feedback control of the current profile on Tore Supra," *Nucl. Fusion*, vol. 37, no. 6, pp. 777–791, Jun. 1997.

[10] O. Barana, D. Mazon, L. Laborde, and F. Turco, "Feedback control of the lower hybrid power deposition profile on Tore Supra," *Plasma Phys. Control. Fusion*, vol. 49, no. 7, pp. 947–967, Jul. 2007.

[11] J. Blum, *Numerical Simulation and Optimal Control in Plasma Physics*. Hoboken, NJ: Wiley, 1988.

[12] P. Neittaanmaki and D. Tiba, *Optimal Control of Nonlinear Parabolic Systems: Theory, Algorithms, and Applications*. New York: Marcel Dekker, 1994.

[13] V. Arnautu and P. Neittaanmaki, *Optimal Control From Theory to Computer Programs*. Norwell, MA: Kluwer, 1993.

[14] R. Pytlak, *Numerical Methods for Optimal Control Problems With State Constraints*. New York: Springer-Verlag, 1999.

[15] J. Nocedal and S. J. Wright, *Numerical Optimization*, 2nd ed. New York: Springer-Verlag, 2006.

[16] P. Holmes, J. Lumley, and G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. New York: Cambridge Univ. Press, 1996.

[17] K. Kunisch and S. Volkwein, "Control of the Burgers equation by a reduced-order approach using proper orthogonal decomposition," *J. Optim. Theory Appl.*, vol. 102, no. 2, pp. 345–371, Aug. 1999.

[18] Y. Ou, T. C. Luce, E. Schuster, J. R. Ferron, M. L. Walker, C. Xu, and D. A. Humphreys, "Towards model-based current profile control at DIII-D," *Fusion Eng. Des.*, vol. 82, no. 5-14, pp. 1153–1160, Oct. 2007.

[19] M. Bergmann, L. Cordier, and J. Brancher, "Optimal rotary control of the cylinder wake using proper orthogonal decomposition reduced-order model," *Phys. Fluids*, vol. 17, no. 9, p. 097 101(1–21), Aug. 2005.

[20] D. Mayne and H. Michalska, "Receding horizon control of nonlinear systems," *IEEE Trans. Autom. Control*, vol. 35, no. 7, pp. 814–823, Jul. 1990.

[21] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, Jun. 2000.

[22] R. Findeisen and F. Allgower, "An introduction to nonlinear model predictive control," in *Proc. 21st Benelux Meeting Syst. Control*, 2002, pp. 1–23.

Photographs and biographies of all authors not available at the time of publication.