Model-based electron density profile estimation and control, applied to ITER

Document status and date:
Published: 19/11/2021

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Model-based electron density profile estimation and control, applied to ITER

To cite this article: T O S J Bosman *et al* 2021 *J. Phys. Commun.* **5** 115015

View the article online for updates and enhancements.

Journal of Physics Communications

Model-based electron density profile estimation and control, applied to ITER

T O S J Bosman[1,2,*] ⬥, M van Berkel[1] ⬥ and M R de Baar[1,2] ⬥

[1] DIFFER—Dutch Institute for Fundamental Energy Research, PO Box 6336, 5600 HH Eindhoven, The Netherlands
[2] Eindhoven University of Technology, Dept. of Mechanical Engineering, Control Systems Technology Group, PO Box 513, 5600 MB Eindhoven, The Netherlands
[*] Author to whom any correspondence should be addressed.

E-mail: t.o.s.j.bosman@differ.nl

Abstract

In contemporary magnetic confinement devices, the density distribution is sensed with interferometers and actuated with feedback controlled gas injection and open-loop pellet injection. This is at variance with the density control for ITER and DEMO, that will depend mainly on pellet injection as an actuator in feed-back control. This paper presents recent developments in state estimation and control of the electron density profile for ITER using relevant sensors and actuators. As a first step, Thomson scattering is included in an existing dynamic state observer. Second, model predictive control is developed as a strategy to regulate the density profile while avoiding limits associated with the total density (Greenwald limit) or gradients in the density distribution (e.g. neoclassical impurity transport). Simulations show that high quality density profile estimation can be achieved with Thomson Scattering and that the controller is capable of regulating the distribution as desired.

1. Introduction

Future reactors, such as ITER and DEMO, aim to have a net energy gain [1]. This requires these reactors to operate close to physical limits and to optimize quantities such as temperature, density, and confinement. In practice, shot-to-shot differences in reactor conditions will always be present and the optimal control action will vary subject to the plasma scenario and plasma state. Hence, active plasma feedback control is essential in achieving these requirements [2].

In a tokamak, the produced fusion power directly correlates to the density in the hot core of the reactor [3], particle transport and non-inductive current drive depend on the gradient of the spatial particle distribution [4, 5], a.k.a the (particle) density profile. Furthermore, turbulence and impurity transport depend on the logarithmic gradient of the density profile [6–8] and the density is subject to limits that can lead to detrimental plasma instabilities when violated [9–11]. Consequently, for optimal and reliable high-performance operation of fusion reactors, particle density profile estimation and control are mandatory [12–15].

To synthesize such controllers, model-based control design is required as experimental time for tuning and validation on reactors such as ITER will be scarce and expensive [16, 17]. Model-based design has already successfully been applied to synthesize controllers for plasma shape [18] and profiles in a tokamak [19–22]. Especially, in [23], a controller for the current density profile is presented that can handle plasma limits and actuator constraints simultaneously. The design procedure requires a control-oriented model of the input-output dynamics, i.e., a model that captures the main dynamic relationships between inputs (actuators) and outputs (plasma quantities) to be controlled.

Recently, the control-oriented model RAPDENS was developed for the particle transport dynamics in a tokamak [24]. This model has been used to perform density profile estimation in TCV and AUG with interferometry and spectroscopy [24, 25], to design a controller for the volume-averaged density during the

ITER ramp-up with pellets and gas [26], to estimate the core density in density control experiments with pellets in AUG [27], and to derive a controller for the volume-average density in TCV using gas injection [25].

However, a controller for the particle density distribution,: (a) that is capable of handling shot-to-shot differences; (b) can handle state physical limits; (c) and uses multiple actuators is one key research and development aspect still to be solved for a fusion reactor.

In this work, a two-step model-based approach is proposed to develop such a density profile controller for ITER. The first is the estimation (or reconstruction) of the particle density profile with an observer and relevant sensors. The second is the design and validation of a model-predictive controller for the density profile using two pellet injectors.

For the first step, RAPDENS [24] is applied to ITER and extended with a Thomson scattering (TS) output model. Subsequently, the model and synthetic TS measurements are used together in an extended version of the dynamic state observer (DSO) proposed in [24] to reconstruct the density profile. Simulations of the observer are used to determine the quality of the achieved profile estimation for an H-mode, 15 MA, 5.3 T, DT ITER baseline discharge.

For the second step, model-predictive control (MPC) is applied to synthesize a controller for the density distribution in ITER using pellet injection as main actuator. For the first time, constraints on the density profile due to impurity transport are taken into consideration in the control problem. Simulations using the lightweight control-oriented plasma simulator RAPDENS are used to assess the effectiveness of the controller.

The remainder of this paper is structured as follows. Section 2 briefly summarizes the control-oriented model used for design and simulation. The forward TS model and the DSO are presented in section 3. The density reconstruction results are presented in section 4. The controller design methodology is described in section 5 and the results of control simulations are shown in section 6. This paper is concluded by a discussion of the work in section 7.

2. Control-oriented plasma simulator

In this work, we have used and extended the RAPDENS model (originally proposed in [24]) for the DSO and the synthesis of the MPC controller. A summary of the model is given here.

The RAPDENS model is a lightweight, control-oriented model of the electron particle transport in a tokamak [24]. It is a multi-inventory model in which the particles inside the vessel are attributed to one of three inventories: the plasma, the wall, or the vacuum. Note that the vacuum denotes the region surrounding the plasma that is filled with neutrals. However, the name 'vacuum' is kept for continuity and conherence with previous papers on RAPDENS. The model consists of a 1D-PDE for the evolution of the flux-surface averaged electron density and two OD-ODE's for the evolution of the wall and neutral vacuum inventories. Radial particle transport is modeled with a drift-diffusion model with ad-hoc chosen transport coefficients. The processes that drive the particle exchange between these inventories are modeled in a heuristic fashion. An overview is given in figure 1. Additionally, it contains representations of the fueling with neutral beam, pellet and gas injection. A summary of the model equations is given is appendix A. The full details and derivations of the model equations are outside the scope of this paper and can be found in [24].

For the applications reported in this paper, RAPDENS is used to formulate a system of nonlinear discrete-time ODEs that describes the evolution of the plasma density due to relevant actuator inputs. The internal variables of the model (states), the external parameters, the considered actuators inputs, and the system of ODEs are briefly discussed next.

2.1. States

In RAPDENS, the flux-surface averaged electron density distribution $n_e(\rho, t)$ is discretized in space using finite elements (see e.g. [29]), i.e, the electron density is approximated by

$$n_e(\rho, t) = \sum_{\alpha=1}^{m} \Lambda_\alpha(\rho) b_\alpha(t), \tag{1}$$

where ρ is the normalized toroidal flux. The basis function $\Lambda_\alpha: [0, \rho_e] \rightarrow [0, 1], \alpha = 1, 2, \ldots, m$ are defined as cubic B-splines with finite support [30] ($\rho_e > 1$ being the artificially prolonged flux label to encompass the scrape-off layer. This is discussed in more details in [24]). The variables $b_\alpha(t) \in \mathbb{R}^m$ are time dependent spline coefficients. These coefficients together with the time dependent wall and vacuum inventories, respectively denoted $N_w(t)$ and $N_v(t)$, comprise the state vector $x(t) \in \mathbb{R}^{n_x}$ with $n_x = m + 2$, i.e., the state vector is given by

$$x(t) = \begin{bmatrix} b(t) & N_w(t) & N_v(t) \end{bmatrix}^\top. \tag{2}$$

Figure 1. Overview of the modeled processes in the RAPDENS model. The particles inside the tokamak are attributed either to the plasma, the wall or the vacuum. Green arrows represent the modeled particle fluxes between the inventories. Orange arrows represent particle fluxes crossing the system boundaries. Reproduced from [28] CC BY 4.0.
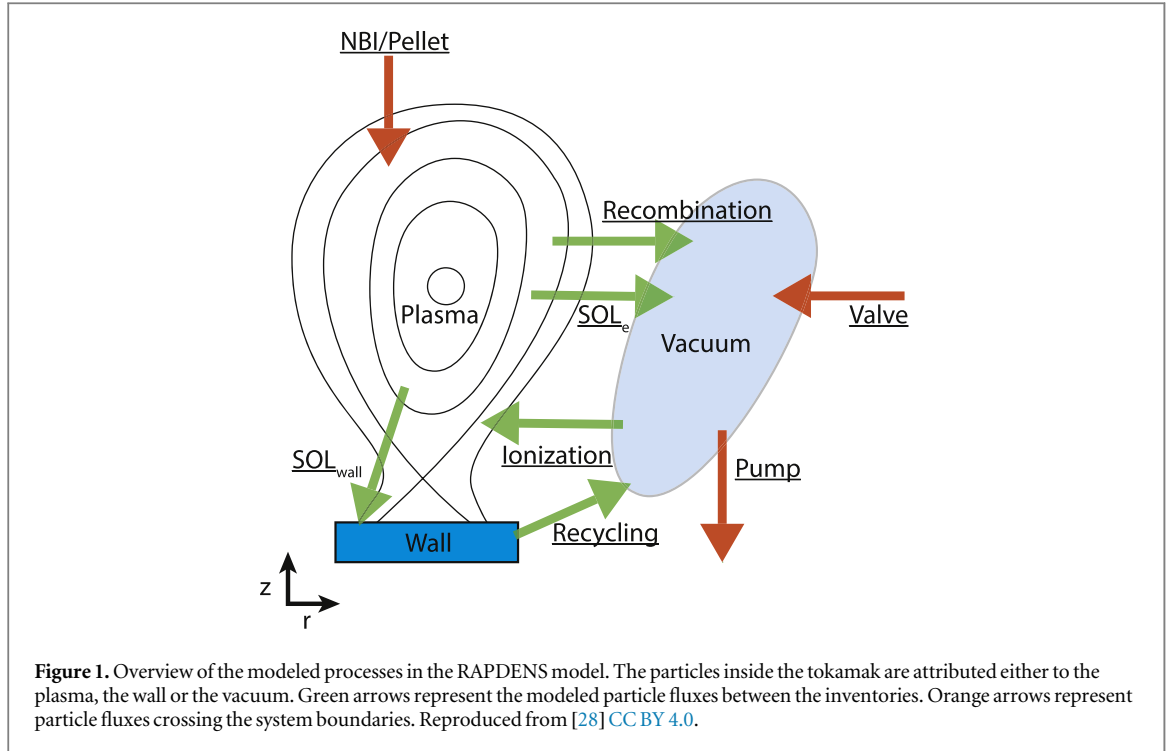
2.2. External parameters:

RAPDENS requires external parameters to be provided in an external parameter vector $p(t) \in \mathcal{P}$ defined as

$$p = [c_D \quad c_H \quad T_{e,b} \quad I_p \quad V \quad V' \quad \kappa \quad \psi_{axis} \quad \psi_{LCFS} \quad G_1 \quad G_0], \tag{3}$$

where c_D indicates limited ($c_D = 0$) or diverted ($c_D = 1$) plasma, c_H implies low confinement ($c_H = 0$) or high confinement ($c_H = 1$) regime, $T_{e,b}$ is the electron temperature at the last-closed flux surface (LCFS), I_p is the plasma current, V is the plasma volume, $V' = dV/dt$, and κ is the plasma elongation, ψ_{axis} and ψ_{LCFS} are the equilibrium $\psi(R, Z)$ evaluated at the magnetic axis and LCFS, $G_1 = \langle (\nabla \rho)^2 \rangle$ and $G_0 = \langle |\nabla \rho| \rangle$ with $\langle |\nabla \rho| \rangle = \langle |\nabla \psi| \rangle \left(\frac{\partial \psi}{\partial \rho} \right)^{-1}$ (where $\langle \cdot \rangle$ denote the flux-surface average).

2.3. Inputs

The particle fueling rate by pellet injection $\Gamma^i_{pellet}(t) \forall i = 1,\ldots,n_u$, where n_u is the number of used pellet injectors, comprise the inputs $u(t) \in \mathbb{R}^{n_u}$ to the model[3]. The input vector is defined as

$$u(t) = \left[\Gamma^1_{pellet}(t),\ldots,\Gamma^{n_u}_{pellet}(t) \right]^\top. \tag{4}$$

2.4. Nonlinear system of ODEs

The spatial discretization of the density profile (1) is inserted in (A.1)–(A.5) and discretized in time using an equidistant temporal discretization $t_k = t_0 + kT_s$ to form the nonlinear set of ODEs

$$x(t_{k+1}) = f(p(t_k), x(t_k), u(t_k)), \tag{5}$$

where $f \colon \mathcal{P} \times \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ is a nonlinear function[4] of the state that represents the state evolution due to physical phenomena and the influence of the actuators.

This formulation is the foundation of the work presented in the coming sections. In sections 3 and 4, it is used to perform state estimation and in sections 5 and 6, it is used to derive and test a density profile controller.

3. Dynamic state observer for the density profile

A requirement to perform profile control is the estimation of the profile from available measurements. For this purpose, a dynamic state observer (DSO) was proposed in [24, 25]. In this section, we summarize the DSO and

---

[3] Note that RAPDENS also contains models for gas fueling and NBI but these are not considered in this work.

[4] Note that for the modeling is this paper, the function f is linear with respect to u. Hence, (5) can be rewritten as $x(t_{k+1}) = f(p(t_k), x(t_k)) + B_d(p(t_k))u(t_k)$, where B_d is a parameter varying matrix.

we present a forward diagnostic model for TS that will allow us to perform profile estimations using TS measurements.

Notation 1: Let a generic variable $e(t)$ be a physical quantity of a real system. The estimate of this quantity made by a DSO at time $t = t_{k_1}$ with diagnostic signals from time $t = t_{k_2}$ is denoted as $\hat{e}_{k_1|k_2}$.

3.1. Forward Thomson scattering model

The DSO is comprised of a prediction model, a diagnostic model, and an observer gain (see figure 4). The original observer, developed in [24, 25], included diagnostic models for interferometry and bremmstrahlung but not for TS. Here, we discuss the inclusion of a TS model that will allow us to perform density estimation with TS measurements.

TS systems are used in tokamaks to measure the radially resolved distributions of electron temperature and density [31]. In combination with knowledge of the 2D equilibrium, the spatial measurements $n_e(R, Z, t)$ can be transformed into equilibrium measurements of the density $n_e(\rho, t)$ at known equilibrium locations $\bar{\rho}_i$ with $i = 1,\ldots,n_{TS}$. n_{TS} denotes the total number of radial measurement locations. The spatial resolution of the ITER core TS system is 67mm [32], which, in a typical ITER plasma with a radius of $r_p \approx 2$ m, provides ≈ 29 measurement locations distributed over the high and low field side. In this work, to be conservative, $n_y = n_{TS} = 11$ is chosen.

The TS output vector $y \in \mathbb{R}^{n_{TS}}$ should contain the plasma density evaluated at the corresponding flux labels $\bar{\rho}_i$, i.e.,

$$y(t) = \begin{bmatrix} n_e(\rho, t)|_{\bar{\rho}_1} & \cdots & n_e(\rho, t)|_{\bar{\rho}_{n_{TS}}} \end{bmatrix}^\top. \tag{6}$$

The relation between (2) and (6) is obtained by evaluating the basis functions used for spatial discretization in (1) at $\rho = \bar{\rho}_i \ \forall \ i = 1,\ldots,n_{TS}$. Inserting the spatial discretization (1) in (6) results in the numerical expression of the output vector as function of the b-spline coefficients:

$$y(t) = \begin{bmatrix} \sum_{\alpha=1}^m \Lambda_\alpha(\rho)|_{\bar{\rho}_1} \ b_\alpha(t) \\ \vdots \\ \sum_{\alpha=1}^m \Lambda_\alpha(\rho)|_{\bar{\rho}_{n_{TS}}} \ b_\alpha(t) \end{bmatrix}, \tag{7}$$

where $\Lambda(\rho)|_{\bar{\rho}^i}$ are the basis function evaluated at the corresponding radial locations. By defining $\Lambda_{TS} \in \mathbb{R}^{n_{TS} \times m}$ the matrix containing the basis function evaluated at all the n_{TS} radial locations, we can formulate an output mapping $C(p(t)) = \begin{bmatrix} \Lambda_{TS} & 0^{n_{TS} \times 2} \end{bmatrix}$ such that the forward outputs are given by

$$y(t) = C(p(t))x(t). \tag{8}$$

The output mapping C changes as function of the equilibrium, hence, it depends on the parameter vector $p(t)$ introduced in a previous paragraph. For the simulations presented in this work however, it is assumed for simplicity that the measured locations are constant.

3.2. Working principles of the observer

A dynamic state observer (DSO) is an algorithm that uses measurements to constrain prediction made by a dynamic model to iteratively estimate the internal state of a system. A brief overview of the working principles of the observer is given here, more details can be found in appendix B.

In the case of the density distribution, we are dealing with a nonlinear system. Hence, we are using the extended Kalman filter (EKF) framework [33] (see appendix B.2) to estimate the density. The algorithm works as follows. At each time step, the latest measurements $\tilde{y}_k$ are compared with the predicted measurements $\hat{y}_{k|k-1}$ to form the residual z_k (B.2). The residual is multiplied by the Kalman gain (B.4) and used to updated the predicted state $\hat{x}_{k|k-1}$ to obtain the state estimate $\hat{x}_{k|k}$. Finally, the predicted state $\hat{x}_{k+1|k}$ and measurement $\hat{y}_{k+1|k}$ at the next time step are computed using RAPDENS (5) and the forward diagnostic model (8).

The performance, stability, and robustness of the observer depends on the choice of the measurement covariance R_k, the process covariance Q_k^x, and the disturbance covariance Q_k^ζ. This is detailed in section 4.2.

In this section, we included a forward TS model in a DSO. With this addition, TS measurements can be used to perform estimations of the density profile. In the next section, the performances of the observer are assessed for ITER.
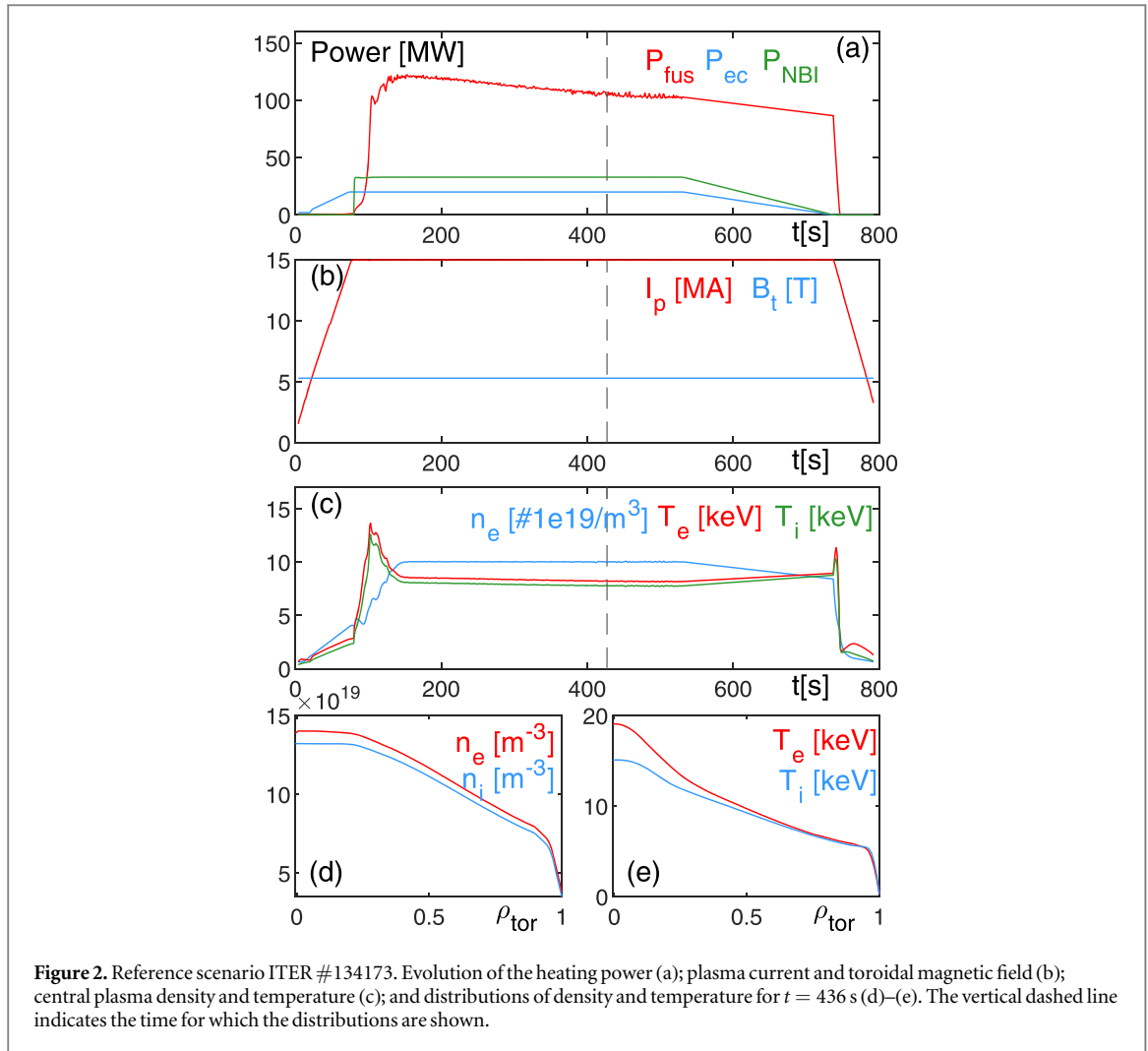
Figure 2. Reference scenario ITER #134173. Evolution of the heating power (a); plasma current and toroidal magnetic field (b); central plasma density and temperature (c); and distributions of density and temperature for $t = 436$ s (d)–(e). The vertical dashed line indicates the time for which the distributions are shown.

4. Density profile estimation in ITER baseline simulations using Thomson scattering measurements

In this section, density estimation simulations are discussed where we demonstrate the effectiveness of using TS measurements to update the state predictions in the DSO. First, we introduce the simulated plasma scenario and argue the choice of the covariance matrices, i.e. the tuning knobs of an EKF. Subsequently, we discuss the simulation set-up and present the observer's performance in a simulation with plant-observer model mismatch and realistic noise levels.

4.1. Plasma scenario
The observer is used to estimate the density profile with TS measurements for an ITER H-mode, 15 MA, -5.3 T, baseline discharge. ITER shot 134 173 is chosen as reference scenario as it is up to date the most complete prediction of an ITER baseline discharge available.

The scenario is obtained with an integrated model (IM) simulation [34]. The simulation was performed as an open-loop coupling within IMAS [35] of the free boundary equilibrium code DINA [36–38] with the JINTRAC [39] suite of codes. The baseline scenario was assessed from the formation of the X-point to the transition from X-point to limiter. The time evolution of relevant plasma parameters are given in figures 2(a)–(c).

In the reference scenario, the ramp-up is modeled to last for 75 s with the L-H transition being triggered by the increase in auxiliary power. The flat-top is modeled to last from $t = 75$ s to $t = 730$ s. During that time the particle source is pellet injection with a pellet composition of 50% deuterium and 50% tritium atoms. It is represented as a continuous source of 2.3×10^{22} atoms/s deposited at $\rho = 0.85$ with Gaussian deposition profile with normalized deposition width of 0.15.
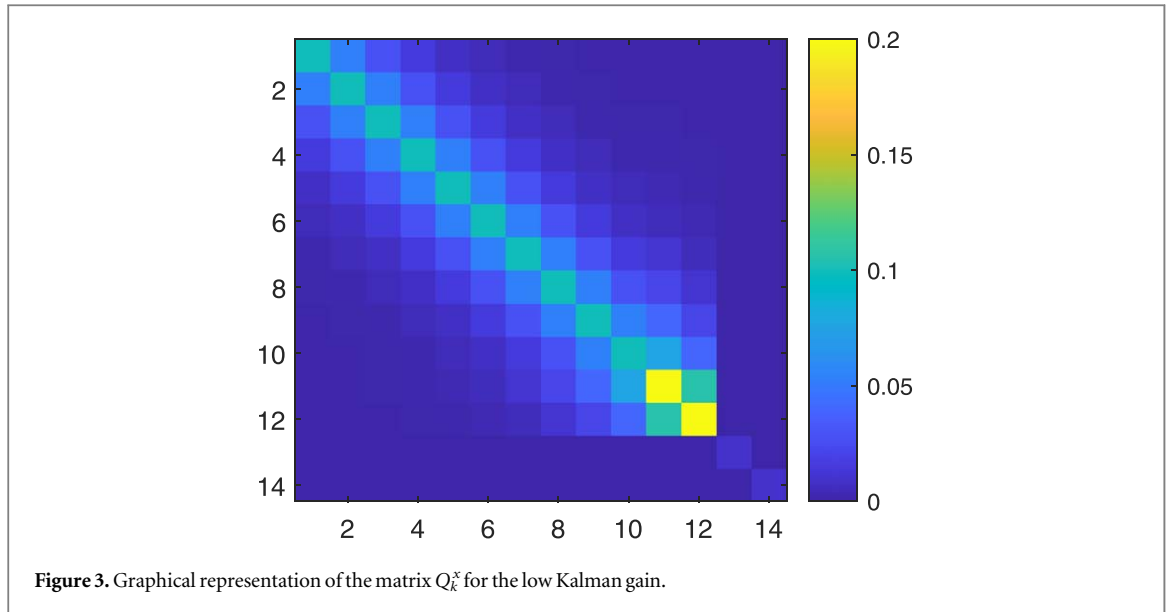
Figure 3. Graphical representation of the matrix Q_k^x for the low Kalman gain.

4.2. Design of covariance matrices

The accuracy, convergence speed, and robustness of the state estimation depends on the choice of the covariance matrices R_k, Q_k^x, and Q_k^ζ [33]. The choice of these matrices is discussed here.

The measurement covariance matrix R_k is chosen as a diagonal matrix with the square root of the modeled noise covariance on the diagonal (the modeled noise is discussed in section 4.3). Note that this assumes that each TS measurement location is an uncorrelated individual output.

The process covariance matrix Q_k^x is constructed as a symmetric Toeplitz matrix[5] with descending first row. The values on the first row of the Toeplitz matrices determine the spatial correlation between the estimates and is chosen as exponentially decaying. The disturbance covariance matrix Q_k^ζ is constructed as a diagonal matrix.

The magnitude of the entries of Q_k^x and Q_k^ζ reflect the amount of trust in the model predictions and influence the Kalman gain (B.4) [33]. A low observer gain reflects small trust in measurements and high trust in the model predictions. Hence, the EKF will react slowly to the differences between measurements and model predictions. A low gain avoids fitting the noise but relies more heavily on the model predictions, making the reconstruction more sensitive to modeling errors. On the contrary, a high Kalman gain reflects high trust in the measurements and large model uncertainty. The DSO will react fast to discrepancies between model and measurements. The reconstruction will be more robust to modeling error but sensitive to diagnostic noise and errors.

The magnitude of the covariance matrices' entries are chosen based on a preliminary analysis of RAPDENS for ITER. The heuristic parameters of RAPDENS are tuned to approximate the flat-top plasma behavior of the reference discharge scenario described in section 4.1. From the analysis, it is concluded that the model had good predictive capabilities in flat-top but that modeling of the L-mode and L-H transition is not fully accurate. Hence, a high Kalman gain is chosen in the ramp-up and ramp-down phases and a low Kalman gain is chosen during flat-top. Furthermore, it was observed that the density prediction mismatched at the edge of the plasma. Hence, the diagonal entries of Q_k^x corresponding to the edge of the plasma were increased (see figure 3).

4.3. Simulation set-up

Before proceeding to the estimation results, we briefly describe the set-up of the simulation used to evaluate the observer's performances. A visual depiction of the simulation set-up is given in figure 4. The EKF is simulated for the entire discharge presented in section 4.1. The external parameters required to run RAPDENS (see section 2), the particle inputs, the initial conditions, and the 'real' (to be estimated) density profile are taken from the database of the IM simulation.

The heuristic parameters of RAPDENS that were tuned to approximate the flat-top plasma behavior in the preliminary analysis are used in the estimation simulation. However, to represent inaccurate knowledge of the transport and investigate the performance of the observer with model uncertainty, systematic plant-observer model mismatch is manually introduced by changing the tuned transport coefficients and the pellet spatial deposition function.

Furthermore, measurement inaccuracies and noise are included in the simulation with synthetic TS measurements $\tilde{y}_k$. They created by adding artificial noise to the density profile computed by the IM. While the

---

[5] A *Toeplitz matrix* is a matrix of which the values on the diagonals are constant.

Figure 4. Block diagram of the DSO and the set-up used for the density estimation simulations for ITER. The external parameters, the particle fueling rates, and the initial conditions are taken from the ITER database and used to simulate RAPDENS. The 'real' density distribution is used to synthesize synthetic TS measurements.
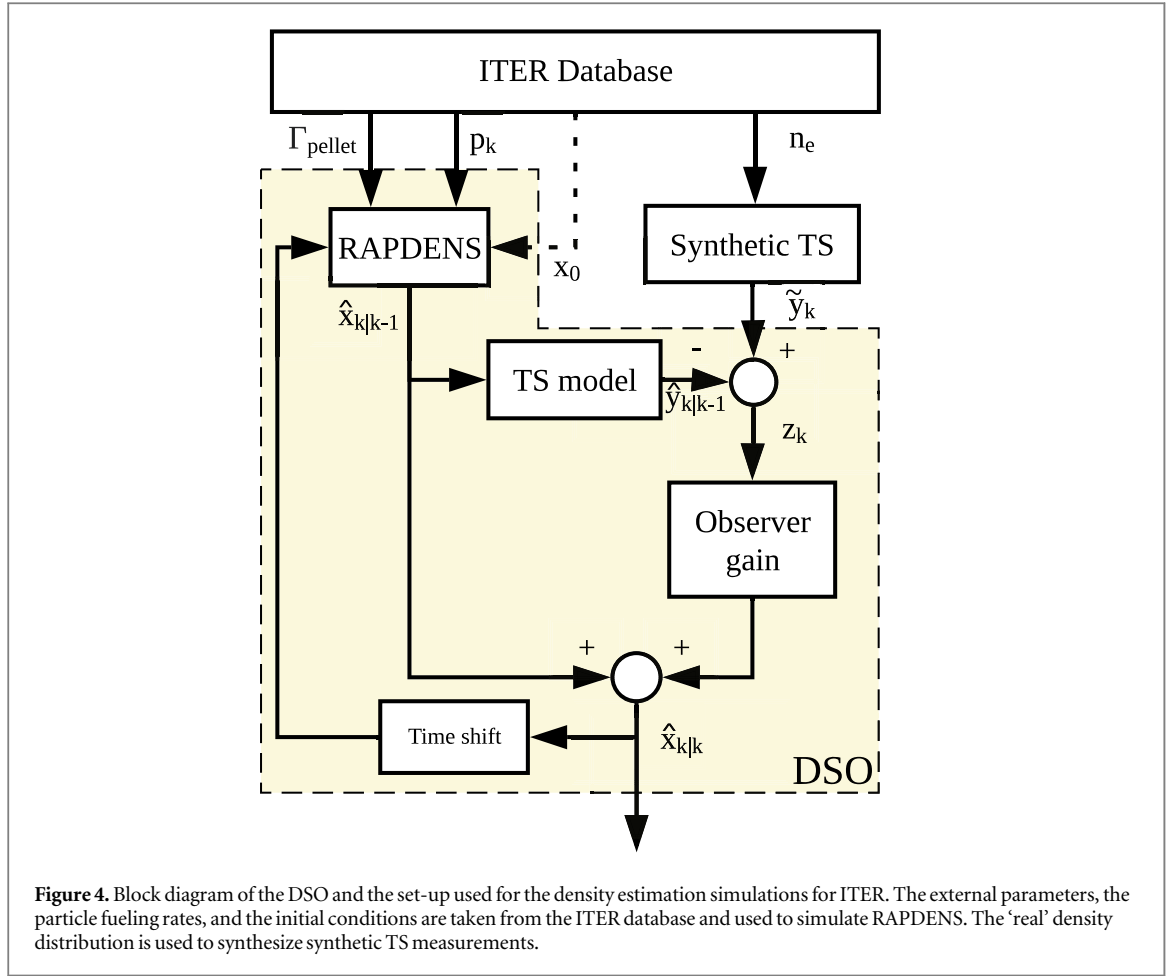
signal-to-noise ratio (SNR) for TS depends on Poisson statistics [40] and thus scales with $1/\sqrt{n_e}$, we choose to model the noise as a Gaussian with expected value $\mathbb{E}[\Delta_{TS}] = 0$ and covariance $\sigma_k^2 = 0.05 * \max(n_e)$. This way, we approximate the required accuracy of the TS system [32] and account for sources of noise such as shot noise, thermal noise in the detection circuit, noise due to plasma light, and noise due to neutron damage on optics installation [41, 42]. It is important to note that Thomson scattering can also be affected by systematic errors, such as misalignment or due to inaccuracies in the equilibrium reconstruction. These are not accounted for in this work. When using 'real' Thomson scattering measurements, detection and correction for these errors can be performed by integrating the Thomson scattering data with other diagnostics in the observer. This is elaborated further in section 7.

4.4. Density profile estimation results

The simulation results are shown in figure 5. The estimation error, depicted in figure 5(b), is expressed as the normalized 2-norm of the error vector: $\|\hat{y}_{k|k} - n_e\|_2 / \|n_e\|_2$, where $\hat{y}_{k|k} = C(p_k)\hat{x}_{k|k}$ with $C(p_k)$ defined in (8) and n_e the density profile of the IM simulation. Only the part of the profile measured by the core TS system [32], i.e., $0 \leqslant \rho \leqslant 0.9$, are used in the error computation.

Overall, good density reconstruction performances can be seen for the entire profile, i.e., the estimated densities are shown to correctly track the true densities (see figure 5(a)). The normalized estimation error averages around $\approx 10\%$ during the ramp-up and $\approx 2\%$ for the flat-top (see figure 5(b)).

During the ramp-up and the L-H transition, from $t = 0$ s to $t = 75$ s the effect of the high Kalman gain can clearly be distinguished. During that time, the artificial measurement noise is visible in the estimation, as can be seen in figure 5(b). Decreasing the Kalman gain during this phase would reduce the influence of noise but increase the overall estimation error as the model is not accurate for the early stages of the discharge.

An increase in the estimation error can be seen between $t = 75$ s and $t = 90$ s. At $t = 75$ s, the plasma enters H-mode. As was discussed in section 4.2, the observer gain is changed from a high to a low Kalman gain at that time. This means that the observer relies more on the model predictions. The modeled density increase in early H-mode is steeper in RAPDENS, hence, the density profile is overestimated (see figure 5(d)). This could be solved by slowly transitioning from a high to a low observer gain once the plasma enters H-mode. After $t = 90$ s,

Figure 5. Results of density estimation simulations for ITER #134173 using synthetic TS data to update the state predictions. Plant-observer mismatch is present. Frame (a) The estimated density (faded) is compared with the IM density (solid) for four flux labels. Dashed vertical lines show the time at which the profiles are shown. Frame (c)–(e) The estimated density profile (red) is compared with the real density profile (black). The TS measurement points are shown (orange). The observer is capable of accurately estimating the density profile. The estimation is of best quality during the flat-top part of the discharge.

the estimation error decreases rapidly. A good agreement can be seen between the estimated and real density profile for the flat-top (see figure 5(e)).

This section shows that using TS measurements to update the predictions made by the RAPDENS model in an EKF results in accurate density profile estimations. Furthermore, the quality of the estimation is deemed sufficient to be used for profile control, especially during the flat-top.

5. Design of an MPC controller

In this section, the design of an MPC controller for the density distribution is presented. We first introduce the concepts of MPC and motivate the use of this strategy for the control of the density distribution. Subsequently, we discuss in detail each component of the controller.

5.1. Model predictive control

Model-predictive control (MPC) is widely adopted as an effective control strategy to deal with multivariate constrained control problems [43, 44]. A block diagram of an MPC controller is given in figure 6(a).

In such a controller, an explicit model of the to be controlled system is used to predict future states/outputs over a prediction horizon (denoted $N \in \mathbb{N}$). These predictions are used to formulate an optimal control problem where the future tracking error, i.e. the difference between the desired and predicted future outputs is minimized. Limits on the output, states, and inputs can be taken into account as constraints in the optimal control problem. This is illustrated for a discrete-time, single-state single-input system in figure 6(b).

The design of an MPC controller consists in: choosing a prediction model, formulating a cost function that encompasses the control objectives, formulating constraint functions that translate physical limits of the plasma

Figure 6. (a) Block diagram of the main components of a MPC controller. (B) Illustration of MPC for a discrete-time single-input single-output system with constant state and input constraints.
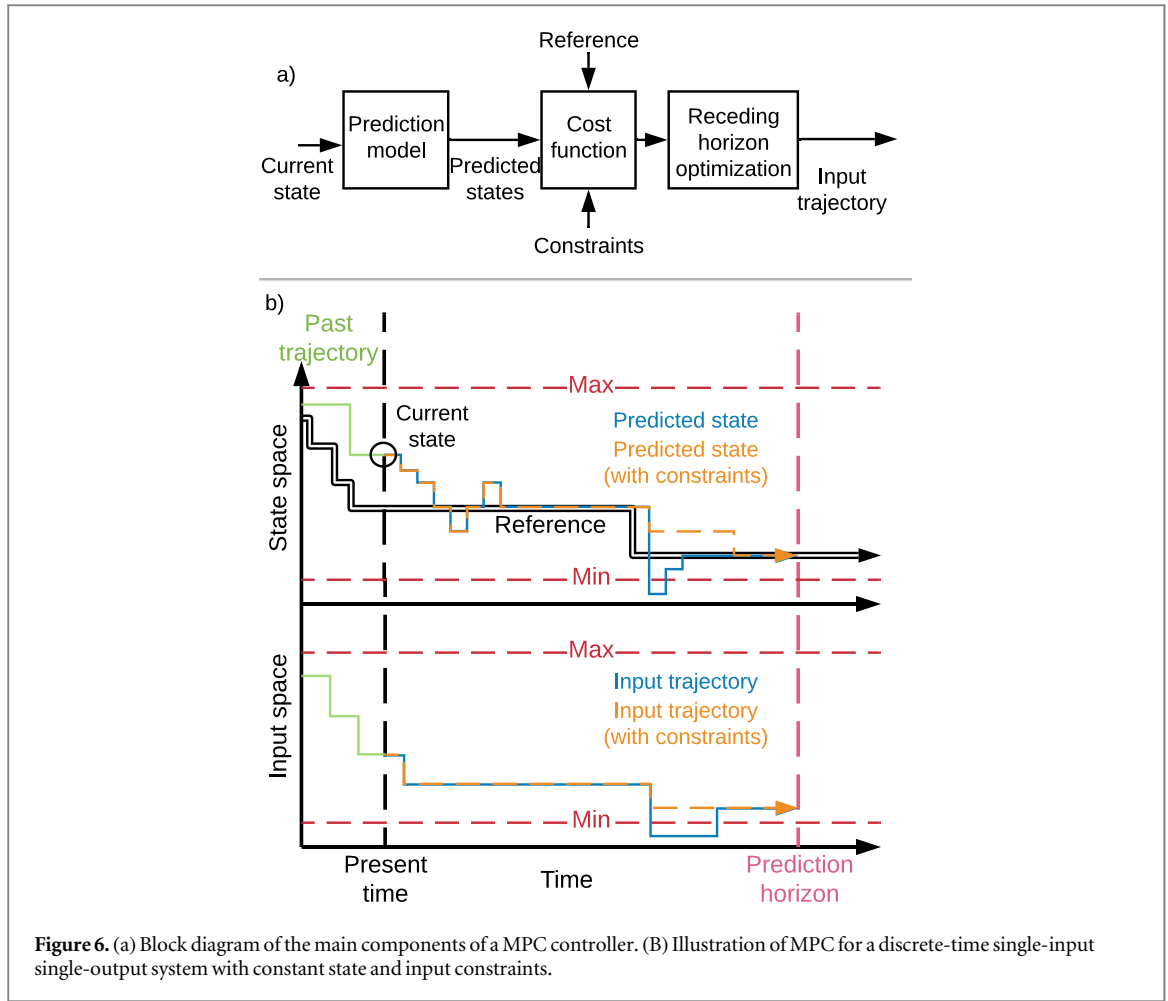
and actuators as state and input constraints, and implementing and solving the optimal control problem in a numerical optimization solver. These steps are discussed in the coming sub-sections.

5.2. Control objectives and motivation for MPC
We aim to synthesize a controller that satisfies the following control objectives:

  (i) Track a high-performance reference density profile in a region of interest (ROI).

 (ii) Guaranteeing that the Greenwald density limit is not being violated.

(iii) Optimizing impurity transport in the ROI.

Note that these objectives are generic, in section 6.1 the objectives are refined for the specific control simulations.

To control the density profile, the controller will have access to multiple actuators, e.g., multiple pellet injectors [45], and it will have access to the reconstructed density profile. The control of the density profile is thus a multivariate control problem. Additionally, the Greenwald density and the optimization of impurity transport can be formulated as state and inputs constraints. Hence, the control problem at hand can be classified as multivariate and subject to state and actutator constraints, which is exactly the type of problem for which MPC is suited.

5.3. Prediction model
The first component of a MPC controller is an explicit model of the systems dynamics. In this work, we use a linear disturbance-augmented model derived from RAPDENS for this purpose.

The prediction model is derived as follows. First, the nonlinear state-space (5) is linearized around the desired flat-top conditions $\{x_{ft}, p_{ft}\}$ to obtain a linear approximation of the flat-top transport dynamics. Next, as the MPC controller runs in discrete-time, the continuous time model is discretized temporally by exact discretization using an equidistant temporal discretization $t_k = t_0 + kT_s$, with $T_s = 3$ ms. This results in a discrete-time linear state-space given by

$$\begin{cases} x(t_{k+1}) & = & Ax(t_k) + Bu(t_k) \\ y(t_k) & = & Cx(t_k). \end{cases} \tag{9}$$

Next, to incorporate integral action and enable the controller to compensate for steady-state offsets induced by model mismatch and slow changing disturbances, a disturbance model as proposed in [46, 47] is used. To do so, (9) is augmented to

$$\begin{cases} x(t_{k+1}) & = & Ax(t_k) + Bu(t_k) + B_d d(t_k) \\ d(t_{k+1}) & = & d(t_k) \\ y(t_k) & = & Cx(t_k) + C_d d(t_k), \end{cases} \tag{10}$$

such that it includes the disturbance $d(t) \in \mathbb{R}^{n_d = n_y}$ with disturbance model matrices B_d and C_d. These matrices are chosen to guarantee the observability of the augmented system (10) [47].

The first condition for the augmented system to be observable is that the non-augmented system is observable. The local linear model (9) (with state vector (2)) contains two unobservable states. These are the particle inventories of the wall N_w and vacuum N_v. Hence, for the controller we use the minimal realization of input-output dynamics. Next, we choose $B_d = I^{n_x, n_d}$ and $C_d = I^{n_d, n_d}$, such that

$$\mathrm{rank} \begin{bmatrix} A - I & B_d \\ C & C_d \end{bmatrix} = n_x + n_d. \tag{11}$$

guaranteeing that the augmented prediction model is observable [47].

Notation 2: For a system with state variable x and input variable u, we distinguish between the system's state at time $t = t_k$ denoted $x(t_k)$ and the predicted state at time $t = t_k + i$ denoted $x_{i,k}$. Analogously, $u(t_k)$ denotes the input to the system at time $t = t_k$ and $u_{i,k}$ denotes the input that would be applied at time $t = t_k + i$.

Using notation 2, the discrete-time linear disturbance augmented state-space prediction model is then given by

$$\begin{cases} x_{i+1,k} & = & Ax_{i,k} + Bu_{i,k} + B_d d_{i,k} \\ d_{i+1,k} & = & d_{i,k} \\ y_{i,k} & = & Cx_{i,k} + C_d d_{i,k}, \end{cases} \tag{12}$$

where $x_{i,k}$, $d_{i,k}$, $y_{i,k}$, and $u_{i,k}$ denoted the predicted state, disturbance, output and control input at $i \in \mathbb{N}$ time steps ahead of the starting time $t = t_k$.

By introducing, for a given prediction horizon N, the stacked notations

$$\boldsymbol{X}_k = \underbrace{\begin{bmatrix} x_{1,k} \\ x_{2,k} \\ \vdots \\ x_{N-1,k} \\ x_{N,k} \end{bmatrix}}_{\text{Future states}}, \quad \boldsymbol{U}_k = \underbrace{\begin{bmatrix} u_{0,k} \\ u_{1,k} \\ \vdots \\ u_{N-2,k} \\ u_{N-1,k} \end{bmatrix}}_{\text{Future inputs}}, \tag{13}$$

the compact formulation of the entire prediction sequence is given by

$$\boldsymbol{X}_k = \mathbb{A}_x x(t_k) + \mathbb{B}_u \boldsymbol{U}_k + \mathbb{B}_d d(t_k), \tag{14}$$

the stacked prediction matrices $\mathbb{A}_x$, $\mathbb{B}_u$, and $\mathbb{B}_d$ are derived in C.1.

The prediction model allows us to relate an unknown input sequence $\boldsymbol{U}_k$ to the current state $x(t_k)$ and current disturbance $d(t_k)$. In the coming sections, an optimal control problem is formulated using this model that can be used to compute an optimal input sequence.

5.4. Disturbance estimation

The disturbance $d(t_k)$ introduced in (10) is estimated at each time instance using a Kalman filter (KF). The KF equations can be found in appendix B. The performance and stability of the observer (and partially that of the controller) are determined by the choice of the measurements covariance $\mathcal{R}$ and the disturbance covariance $\mathcal{Q}_d$. They have been tuned manually and chosen as diagonal matrices with $\mathcal{R} = 5 \times 10^{-3} I_{n_y}$ and $\mathcal{Q}_d = 1 \times 10^{-6} I_{n_d}$ With this design, the estimate of the disturbances converges in ≈ 2 s. After convergence, the prediction model is capable of dealing with systematic and slow-varying plant-model mismatches.

5.5. State and actuator constraints

Next, the construction of the constraints functions is explained. The constraint functions are used to constraint the future states and inputs in the optimization problem.

Three different type of constraints are imposed on the system in this work: linear input constraints to represent the minimal and maximal fueling rates of the pellet injectors, linear state constraints to represent the Greenwald density limit, and nonlinear state and input constraints to optimize impurity transport.

First, linear constraints are defined to represent actuator limits and plasma density limit. As shown in appendix C.2, these constraints can be formulated as linear inequality constraints on the input sequence U_k as

$$A_{\text{ineq}} U_k \leqslant b_{\text{ineq}}. \tag{15}$$

Second, a nonlinear constraint is imposed to the states and inputs that translate the desire to optimize impurity transport. A condition for outward neo-classical impurity transport can be derived as a favorable ratio between the logarithmic ion temperature gradient L_{T_i} and the logarithmic density gradient L_{n_e}, i.e., $L_{T_i}/L_{n_e} \gtrsim \eta_{ic} \approx 1$ [6]. Assuming typical peaked temperature and density profiles in ITER flat-top, the ratio can be used to formulate an inequality constraint on the logarithmic gradient of the density profile:

$$\frac{|L_{n_e}|}{|L_{T_i}|} \leqslant 0.95 + \epsilon$$

$$\equiv |\frac{\partial n_e(\rho)}{\partial \rho} \frac{1}{n_e(\rho)}| \leqslant |L_{T_i}|(0.95 + \epsilon). \tag{16}$$

Note that a soft constraint formulation is used to avoid infeasibility of the optimization problem [48]. In such a formulation, violation of the constraint is allowed but penalized via the parameter ϵ and the cost matrix W_ϵ in the cost function (see (18)). In appendix C.3, the constraint is formulated as a nonlinear function of the state, control input, disturbance estimate, and soft constraint parameter. The nonlinear constraint function $g_1(x_k, u_k, d_k, \epsilon)$ is defined as (time dependence denoted by subscript k for readability):

$$g_1(x_k, u_k, d_k, \epsilon) \equiv |W_y C'(Ax_k + Bu_k + B_d d_k)| -$$
$$|W_y(C(Ax_k + Bu_k + B_d d_k) + C_d d_k)(0.95 W_y L_{T_i} + \epsilon)| \leqslant 0. \tag{17}$$

The linear and nonlinear constraints are used to constrain the optimization problem. With their inclusion the controller accounts for physical limits of the system (actuator and plasma) and the density profile—inward impurity transport interaction.

5.6. Cost function

The objective of the controller is to track the high-performance reference r_k while avoiding aggressive control and violating the constraints. This is expressed in the cost function J_k as:

$$J_k = \|x_{N,k} - \bar{x}_N\|_P^2 + \sum_{j=0}^{N-1}(\|x_{j,k} - \bar{x}_j\|_{W_x}^2 + \|u_{j,k} - \bar{u}_j\|_{W_u}^2) + W_\epsilon \epsilon^2, \tag{18}$$

where for a given matrix T and vector x of appropriate dimensions, we define $\|x\|_T^2 \triangleq x^\top Tx$. $\bar{x}_i$ and $\bar{u}_i$ are the desired steady-state states and control inputs, i.e., the states and inputs that are to be reached for the reference to be tracked. They are computed using

$$\begin{bmatrix} \bar{x}_i \\ \bar{u}_i \end{bmatrix} = \begin{bmatrix} A - I & B \\ HC & 0 \end{bmatrix}^{-1} \begin{bmatrix} -B_d \hat{d}(t_k) \\ r(t_k) - C_d \hat{d}(t_k) \end{bmatrix}, \tag{19}$$

where (A, B, B_d, C, C_d) are the matrices of (10) and $H \in \{0, 1\}^{n_z, n_x}$ is the controlled variable matrix.

In (18), the *stage state cost* matrix W_x is used to penalize deviations between the desired steady-state state and the future predicted states, up to but excluding the terminal state ($x_{N,k}$). The *stage input cost* matrix W_u is used to weigh the inputs and avoid aggressive control actions. The *terminal state cost* matrix P is used to penalize the deviation of the terminal state. Finally, *soft constraint violation cost* matrix W_ϵ is used to penalize the violation of the nonlinear state constraint (see section 5.5).

An iterative procedure is used to choose the weights. The matrix W_x is chosen as a block-diagonal matrix of two matrices: one being a weighted unity, the other being a zero matrix. The dimensions of these matrices ensure that only the states that parametrize the ROI are penalized. The matrix W_u is chosen as a weighted unity matrix, i.e., $W_u = 5 \cdot 10^{-3} I$, and matrix W_ϵ is chosen as positive definite diagonal matrix with $W_\epsilon \succ Q$. Finally, P is chosen as the solution of the discrete algebraic Riccati equation

$$P = A^\top PA - (A^\top PB)(B^\top PB + W_u)^{-1}(B^\top PA) + W_x. \tag{20}$$

By defining these weights, the cost function J is convex and quadratic in U_k [49].

5.7. Optimization problem

The future inputs U_k are computed by minimizing the cost function (18) subject to the constraints (15) and (17). Using the prediction model, the nonlinear online optimization is defined as:

$$\min_{U_k, \epsilon} \quad U_k^\top \mathbb{W}_{u,u} U_k + U_k^\top (\mathbb{W}_{u,x} \hat{x}_{k,k} + \mathbb{W}_{u,d} \hat{d}_{k|k} + \mathbb{W}_{u,r}) + W_\epsilon \epsilon^2 \text{subj.to} \quad A_{\text{ineq}} U_k \leqslant b_{\text{ineq}},$$

$$g_1(x_k, u_k, d_k, \epsilon) \leqslant 0. \tag{21}$$

The matrices $\mathbb{W}_{u,u}$, $\mathbb{W}_{u,x}$, $\mathbb{W}_{u,d}$, and $\mathbb{W}_{u,r}$ are defined in C.4. The optimization problem is nonlinear due to the presence of the impurity transport constraint (17). Therefore, sequential quadratic programming [50] is used to solve the problem. Due to the nonlinear constraint, no guarantee on convexity (and global optimality) can be given.

Once an 'optimal' solution U_k^* of (21) is obtained, the first n_u entries of the optimal sequence are applied as control inputs to the system. In the next section, the results of control simulations are shown to test the performance of the controller.

6. Simulation results of closed-loop density distribution control

In this section, we present control simulations that were used to test the MPC controller. Note that it is not the objective of these simulations to provide quantitative estimates of the ITER performance or controllability of a particular scenario, but to illustrate the potential of MPC control for the density profile and help identify challenges that remain to be solved.

First, we discuss the simulation set-up and refine the control requirements. Next, we demonstrate tracking of a high-performance density profile while minimizing inward impurity transport. This is done in simulations with plant-controller model mismatch and a continuous pellet representation.

6.1. Simulation set-up and control requirements

The considered actuators to perform control are the pellet injection systems that have their injection location situated at the high field side (HFS). During ITER baseline operation, two HFS PIS are planned with the possibility to expand to four [45]. The control inputs (signals changed by the controller) $u(t_k)$ are chosen as the fueling rates of the PIS and defined as

$$u(t_k) = \begin{bmatrix} \Gamma_{p,1}(t_k) & \Gamma_{p,2}(t_k) \end{bmatrix}^\top. \tag{22}$$

The deposition function are modeled as parabolic functions with normalized radius $\rho_{p,1} = 0.7$ and $\rho_{p,2} = 0.85$ and normalized widths of $w_{p,1} = 0.3$ and $w_{p,2} = 0.15$. These represent obtainable fueling profiles for 3 and 5 mm pellets in ITER [51].
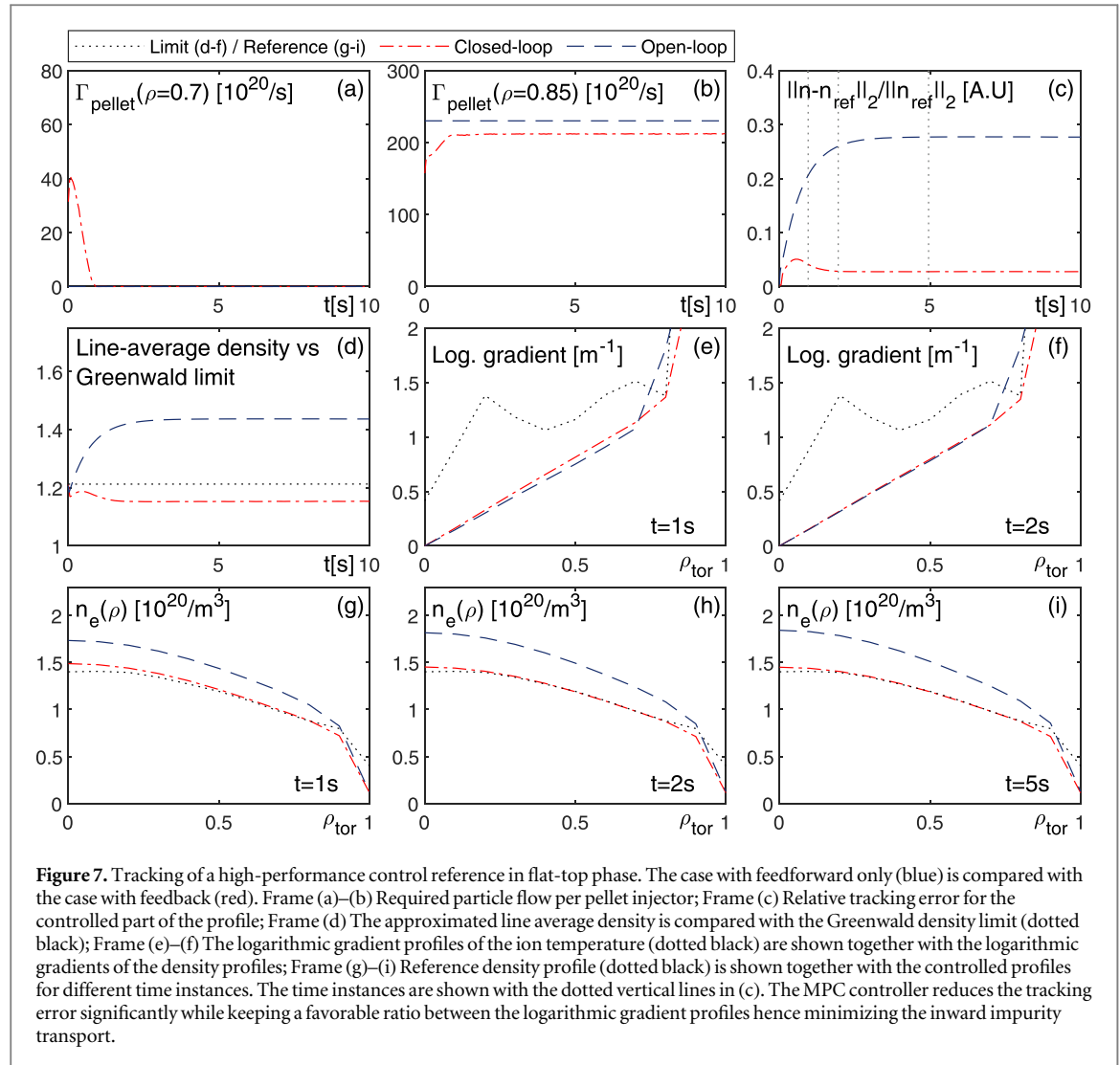
The controlled variables $z(t)$ are the density profile evaluated on an equidistant equilibrium grid for $0 \leqslant \rho \leqslant 0.9$. Similarly, the control reference $r(t)$ is the desired density profile evaluated on the same equilibrium grid. The controller is synthesized using this nominal configuration.

To assess the performance of the controller in simulation, the control objectives discussed in section 5.2 are named and specified as follows:

(i) **Error (Err) requirement:** The controller can maintain the average tracking error below 5%. The tracking error is expressed in the normalized 2-norm of the error vector: $||z(t_k) - r(t_k)||_2 / ||r(t_k)||_2$.

(ii) **Greenwald (Gw) requirement:** The controller can guarantee that the line-average density will not exceed the Greenwald density n_{GW} [9] for any time $t \geqslant t_0$.

(iii) **Transport (Trsp) requirement:** The controller can optimize neoclassical impurity transport by maintaining a favourable ratio between the logarithmic gradients of the ion temperature and density profiles for the domain $\Omega_c = \{(t, \rho) \in \mathbb{R} \mid t \geqslant t_0, \ 0 \leqslant \rho \leqslant 0.9\}$.

6.2. Tracking of a reference in flat-top with plant-controller model mismatch

The goal of the controller is to compensate for mismatches between the real system and the model in the controller. Therefore, mismatch is introduced in the plant with respect to the controller model. The results of two simulations are presented here. In the first (figure 7), transport mismatch is introduced by decreasing the diffusion coefficient and increasing the drift coefficient. Furthermore, a mismatch is introduced by changing the width of the pellet deposition profile. These changes increase inward transport significantly resulting in higher densities at nominal particle inputs. In the second (figure 8), transport mismatch is introduced by increasing the diffusion coefficient and decreasing the drift coefficient in the plant. Furthermore, a mismatch is introduced by

Figure 7. Tracking of a high-performance control reference in flat-top phase. The case with feedforward only (blue) is compared with the case with feedback (red). Frame (a)–(b) Required particle flow per pellet injector; Frame (c) Relative tracking error for the controlled part of the profile; Frame (d) The approximated line average density is compared with the Greenwald density limit (dotted black); Frame (e)–(f) The logarithmic gradient profiles of the ion temperature (dotted black) are shown together with the logarithmic gradients of the density profiles; Frame (g)–(i) Reference density profile (dotted black) is shown together with the controlled profiles for different time instances. The time instances are shown with the dotted vertical lines in (c). The MPC controller reduces the tracking error significantly while keeping a favorable ratio between the logarithmic gradient profiles hence minimizing the inward impurity transport.

changing the deposition radii of the pellet deposition profile, respectively to [0.8,0.9]. These changes decrease inward transport significantly resulting in lower densities at nominal particle inputs.
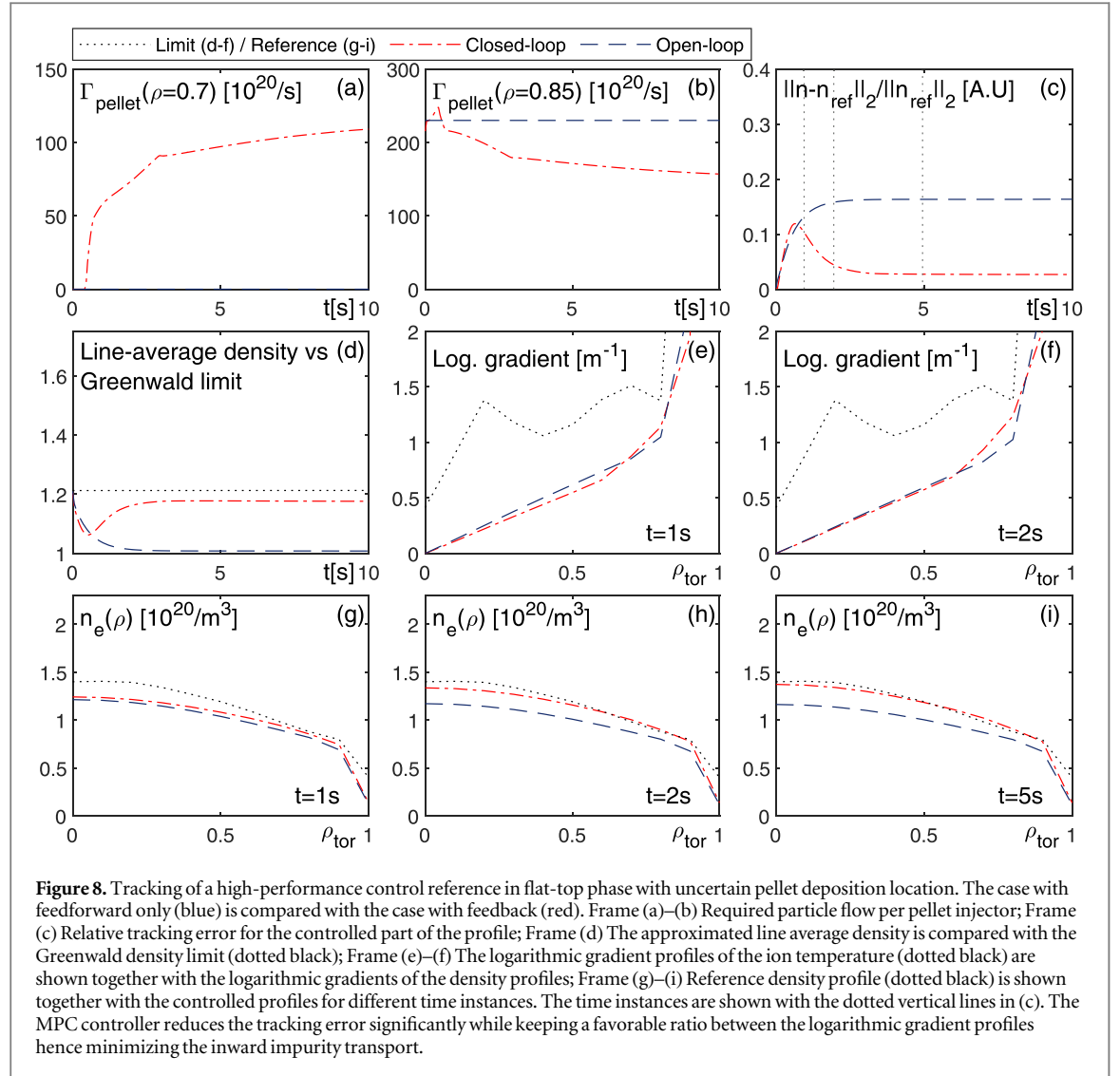
The open-loop control inputs are computed based on the nominal configuration of the system. For both simulations, the uncertainty in transport results in a large tracking error (figures 7(c) and 8(c)). In the first simulation, the uncertainty causes a violation of the Greenwald density limit (figure 7(d)) and of the logarithmic gradient constraint (figures 7(e)–(f). It is shown in figures 7(g)–(i) and 8(g)–(i) that the open-loop density profiles (blue) do not match the reference profile (black).

The MPC controller is used to control the density profile in closed-loop. The control inputs differ from the open-loop (figures 7(a)–(b) and 8(a)–(b). It can be seen that in the presented simulations, the controller achieves the control objectives: the tracking error is below 5% (figures 7(c) and 8(c)); the Greenwald density limit is respected (figure 7(d)); and the favorable ratio between ion-temperature and density logarithmic gradients is maintained (figures 7(e)–(f). Furthermore, a good agreement between the controlled (red) and reference (black) profiles can be seen (figures 7(g)–(i) and 8(g)–(i).

These results show that with continuous actuators, the MPC controller is capable of tracking a high-performance reference density profile in the presence of plant-controller model mismatch while simultaneously accounting for inward impurity transport.

Note that the relation between the density and the logarithmic gradient profile depend on a lot of factors, e.g. ratio between transport coefficients, pellet deposition location and deposition profiles, and the fueling rates (control inputs). For the controller to account for inward impurity transport, the control inputs must give enough freedom to find a feasible input sequence that minimizes the control error while respecting the constraint. It is however possible that the control reference and logarithmic gradient constraint become unfeasible (e.g. if the transport coefficients profiles changes drastically), i.e., there does not exist an input sequence for which the logarithmic gradient constraint is respected (this depends greatly on the degrees of freedom of the controller). In this case, the controller cannot optimize the impurity transport. However, it can

Figure 8. Tracking of a high-performance control reference in flat-top phase with uncertain pellet deposition location. The case with feedforward only (blue) is compared with the case with feedback (red). Frame (a)–(b) Required particle flow per pellet injector; Frame (c) Relative tracking error for the controlled part of the profile; Frame (d) The approximated line average density is compared with the Greenwald density limit (dotted black); Frame (e)–(f) The logarithmic gradient profiles of the ion temperature (dotted black) are shown together with the logarithmic gradients of the density profiles; Frame (g)–(i) Reference density profile (dotted black) is shown together with the controlled profiles for different time instances. The time instances are shown with the dotted vertical lines in (c). The MPC controller reduces the tracking error significantly while keeping a favorable ratio between the logarithmic gradient profiles hence minimizing the inward impurity transport.

give a warning to the overarching supervisory controller that the constraint is violated. Subsequently, fast update methods of the transport coefficients [52] can be used to update the transport model in the controller and determine a better suited control reference.

7. Discussion and outlook

In this work, we extend a dynamic state observer to estimate the density distribution in ITER using Thomson scattering (TS) measurements. For this, we combine the control-oriented particle transport model RAPDENS with a forward TS model and synthetic TS measurements in an extended Kalman filter (EKF) framework. We have shown that reliable, high-quality density profile estimates can be obtained in simulation with realistic measurement noise levels. However, we use a simple model for the synthetic TS measurements (Gaussian distribution) and thus do not account for systematic errors (e.g., due to equilibrium reconstruction errors or misalignment) or density dependent errors. These error can be present when using real TS measurements. Their presence will reduce the performance of the observer and might require a specific error handling procedure to ensure reliable reconstruction of the density profile. As a next step, we propose to test the extended observer on an experimental reactor with real diagnostic data, preferably AUG or TCV as it was already implemented there [24]. On these reactors, the TS measurements will be used together with other diagnostics (interferometers on TCV; interferometers and radiation measurements on AUG) to reconstruct the density profile. It can then be investigated if the expected density estimation performances are achieved and the increase in reconstruction performance by including TS can be quantified. Testing the observer with experimental data will help identify if additional handling is required to deal with systematic errors and design it if need be.

Furthermore in this work, we have synthesized a model-predictive controller for the density distribution in a tokamak. The controller uses the pellet fueling rate as control input to track a high-performance reference

profile, avoid density limits, and optimize for impurity transport. We show that for continuous actuators, the controller is capable of achieving these control requirements in simulations with plant-controller model mismatch.

Even so, we use the heuristic control-oriented model RAPDENS to synthesize and validate the controller. For further validation, it will be relevant as a next step to couple the controller to more complex first-principles transport codes, e.g., to JETTO as part of JINTRAC [39] to perform closed-loop control simulations, and investigate the achieved control performances. Additionally, pellets are inherently discrete events. However since this works presents a for the first time a controller for the density profile, this discrete nature is not included. Knowing that in a real reactor, the pellet size cannot be changed for each control action and is subject to constraints linked to for example plasma penetration, it will be relevant to investigate the performance of the controller when the discrete pellet nature is introduced in the control loop and when only pellets of a fixed size can be used.

Furthermore, we would like to address the possible extensions of the choice of control inputs. In this work and in literature, e.g. [26, 27], the pellet fueling rate is considered as control input. This is a straightforward choice as it enables the use of linear controllers. In practice, however, the actuation with pellet injection is a complex function of pellet size, pellet velocity, injection frequency, and plasma temperature. We think it is relevant to study the possibility of using a more complete input space, e.g., by also taking into account pellet velocity, and analyze the effect on controllability and control performances.

Acknowledgments

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

Appendix A. Summary of RAPDENS equations

In this appendix, a summary of the RAPDENS equations is given. Mass conservation and radial transport [53] are used to model the evolution of the radial plasma density $n_e(\rho, t)$ as

$$\left(\frac{\partial V}{\partial \rho}\right)^{-1} \frac{\partial}{\partial t}\left(n_e \frac{\partial V}{\partial \rho}\right) + \left(\frac{\partial V}{\partial \rho}\right)^{-1} \frac{\partial \Gamma_e}{\partial \rho} = S_e \tag{A.1}$$

where ρ is the normalized toroidal magnetic flux, V is the plasma volume, $\Gamma_e(\rho, t)$ is the radial transport flux modeled as

$$\Gamma_e = -\frac{\partial V}{\partial \rho}\left(\langle(\nabla\rho)^2\rangle \chi \frac{\partial n_e}{\partial \rho} + \langle|\nabla\rho|\rangle \nu n_e\right) \tag{A.2}$$

with $\chi(\rho, c_{HL})$ the diffusivity and $\nu(\rho, c_{LH}, I_p)$ the pinch (or drift) velocity where c_{LH} indicates the regime of the plasma (L or H-mode). In A.2, $S_e(\rho, t)$ is the net electron source modeled as

$$S_e = \langle\sigma v\rangle_{\text{iz}} n_n n_e - \langle\sigma v\rangle_{\text{rec}} n_e n_i - \frac{n_e|_{\rho} > 1}{\tau_{\text{SOL}}} + S_{\text{pellet}} + S_{\text{NBI}} \tag{A.3}$$

with $\langle\sigma v\rangle_{\text{iz}}(T_e)$ and $\langle\sigma v\rangle_{\text{rec}}$ the temperature dependent cross-sections for ionization and recombination [3], τ_{SOL} the time constant for particle loss in the scrape-off layer, S_{pellet} the net particle source due to pellet injection, and S_{NBI} the net particle source due to neutral beam injection.

The evolution of the particle inventories $N_w(t)$ in the wall and neutral vacuum density $n_n(t)$ are modeled by the 0D ODE's

$$\frac{dN_w}{dt} = -\frac{N_w - c_w V_{v,0} n_n}{\tau_{\text{release}}} + \left(1 - \frac{N_w}{N_{\text{sat}}}\right) \int_{V_{\text{SOL}}} \frac{n_e}{\tau_{\text{SOL}}} dV, \tag{A.4}$$

$$V_v \frac{dn_n}{dt} = \frac{N_w - c_w V_{v,0} n_n}{\tau_{release}} + \frac{N_w}{N_{sat}} \int_{V_{SOL}} \frac{n_e}{\tau_{SOL}} dV$$

$$+ \int_{V_p} (\langle \sigma v \rangle_{rec} n_e n_i - \langle \sigma v \rangle_{iz} n_n n_e) dV$$

$$- \frac{n_n V_{v,0}}{\tau_{pump}} + \Gamma|_{\rho_e} + \Gamma_{valve}, \tag{A.5}$$

with $\tau_{release}$ the particle release rate, N_{sat} the wall saturation inventory, τ_{pump} the pumping time scale, c_w a dimensionless constant that determines the steady-state balance between the wall and vacuum inventory, and $V_{v,0}$ the nominal vacuum volume.

Appendix B. Dynamic state observer: theory and application to density profile estimation

B.1. Kalman filter equations

Given a system G with state vector x, output vector y, and input vector u, the Kalman filter (KF) assumes that the dynamics of G evolve in discrete-time following a linear state-space as

$$\begin{cases} x(t_{k+1}) &= F(t_k)x(t_k) + B(t_k)u(t_k) + w(t_k) \\ y(t_k) &= H(t_k)x(t_k) + D(t_k)u(t_k) + v(t_k), \end{cases} \tag{B.1}$$

where the state transition dynamics are captured in matrix F and the input dynamics in B. The output model and influence of the inputs on the outputs are captured in matrices H and D. The stochastic behavior of measurement and process noises are modeled in v and w as zero-mean Gaussian white noises with respective covariance matrices R and Q such that $v(t_k) \sim \mathcal{N}(0, R(t_k))$ and $w(t_k) \sim \mathcal{N}(0, Q(t_k))$. The tuning knobs of the KF are the covariance matrices **R** and **Q**.

The algorithm is initialized with an initial state estimate $\hat{x}_{1|0}$ (with corresponding output estimate $\hat{y}_{1|0}$) and an initial *a priori* estimate covariance $\Sigma_{1|0}$. For each discrete time instance $t = t_k$, the estimate of the states are obtained as follows (for readability the time dependence is denoted by the subscript k).

Update step:
Using the latest available measurements $\tilde{y}_k$, the residual z_k and innovation covariance Ω_k are computed by

$$z_k = \hat{y}_{k|k-1} - \tilde{y}_k \tag{B.2}$$

$$\Omega_k = H_k \Sigma_{k|k-1} H_k^\top + R_k, \tag{B.3}$$

where $H(t_k)$ is the output matrix and $\Sigma_{k|k-1}$ the *a priori* process covariance. Next, the *optimal* Kalman gain L_k is computed with

$$L_k = \Sigma_{k|k-1} H_k^\top \Omega_k^{-1}, \tag{B.4}$$

and is used to obtain the updated state estimate $\hat{x}_{k|k}$ and the *a posteriori* state estimate covariance $\Sigma_{k|k}$ as

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + L_k z_k, \tag{B.5}$$

$$\Sigma_{k|k} = (I - \Sigma_{k|k-1} H_k^\top \Omega_k^{-1} H_k) \Sigma_{k|k-1}. \tag{B.6}$$

Prediction step:
Based on the most likely state $\hat{x}_{k|k}$, the actuator inputs at the present time u_k, and the process covariance Q, the one step ahead predicted state $\hat{x}_{k+1|k}$ and outputs $\hat{y}_{k+1|k}$ and the *a priori* covariance at time step $t = t_{k+1}$ are given by

$$\hat{x}_{k+1|k} = F_k \hat{x}_{k|k} + B_k u_k \tag{B.7}$$

$$\hat{y}_{k+1|k} = H_k \hat{x}_{k|k} + D_k u_k \tag{B.8}$$

$$\Sigma_{k+1|k} = F_k \Sigma_{k|k} F_k^\top + Q_k. \tag{B.9}$$

B.2. Extended Kalman filter equations

In case a system cannot be represented accurately by a linear model, the EKF equations are used. The observer assumes that the system's dynamics evolve following

$$\begin{cases} x(t_{k+1}) &= f(x(t_k), u(t_k)) + w(t_k) \\ y(t_k) &= h(x(t_k), u(t_k)) + v(t_k), \end{cases} \tag{B.10}$$

where $f: \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ and $h: \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_y}$ are non-linear functions of the state and input. The stochastic variables v and w are similar as defined in appendix B.1.

There are two main differences between the KF and EKF equations:

(i) The matrices F and H (used in the computation the measurement covariance (B.3), the Kalman gain (B.4), and the state covariances (B.9) and (B.6)) are the jacobians of f and h, i.e.,

$$F(t_k) = \frac{\partial f}{\partial x}\bigg|_{x_{k-1|k-1}} \qquad\qquad H(t_k) = \frac{\partial h}{\partial x}\bigg|_{x_{k-1|k-1}}. \tag{B.11}$$

(ii) The prediction step is performed using the nonlinear model (B.10), i.e., (B.7) and (B.8) become

$$\hat{x}_{k+1|k} = f(\hat{x}_{k|k}, u_k) \tag{B.12}$$
$$\hat{y}_{k+1|k} = h(\hat{x}_{k|k}, u_k) \tag{B.13}$$

It is important to note that the EKF is a linearized version of the Kalman filter for nonlinear dynamical systems, hence, no guarantee about stability or estimation accuracy can be given [33]. These properties are to be checked experimentally for the specific implementations.

B.3. Application of the in the DSO for the density profile estimation

For the density reconstruction, the RAPDENS plasma simulator is used as the one step ahead prediction model. The physical state vector $x(t_k)$ (2) is augmented with additive unknown disturbances $\zeta(t_k) \in \mathbb{R}^{n_x}$. These disturbances are co-estimated by the observer and give a measure of the modeling errors, unmodeled processes, unaccounted particles sources, and errors in the diagnostics models. The stochastic behavior of the state and disturbances is modeled by additive zero-mean white noises w_k^x and w_k^ζ with respective covariance matrices Q_k^x and Q_k^ζ. The augmented nonlinear state-space model is given by

$$\begin{bmatrix} x_{k+1} \\ \zeta_{k+1} \end{bmatrix} = \begin{bmatrix} f_d(x_k, p_k) & B_\zeta \zeta_k \\ 0 & \zeta_k \end{bmatrix} + \begin{bmatrix} B_d(p_k) \\ 0 \end{bmatrix} u_k + \begin{bmatrix} w_k^x \\ w_k^\zeta \end{bmatrix} \tag{B.14}$$

where $B_\zeta = [I^{m \times m} \quad 0^{m \times 2}]$.

An augmented state is defined as

$$\xi_k = \begin{bmatrix} x_k & \zeta_k \end{bmatrix}^\top. \tag{B.15}$$

The stochastic measurement noise is represented by an additive zero-mean white noise v_k with associated covariance matrix R_k. The outputs are assumed to evolve as:

$$y_k = C(p_k)x_k + v_k. \tag{B.16}$$

The final step consists in defining the matrices

$$F_k = \begin{bmatrix} \dfrac{\partial f}{\partial x_k}\bigg|_{\hat{x}_{k|k}, p_k} & B_\zeta \\ 0 & I^{m \times m} \end{bmatrix} \qquad\qquad B_k = \begin{bmatrix} B_d \\ 0 \end{bmatrix}$$
$$H_k = \begin{bmatrix} C(p_k) & 0 \end{bmatrix}. \tag{B.17}$$

The augmented state $\xi(t_k)$ is then estimated by using (B.14) and (B.17) in (B.2)–(B.9).

Appendix C. Controller: definitions and derivations

In this appendix we provide the definitions and derivations of the controller matrices described in section 5.

C.1. Prediciton model matrices

For a system described by a linear time-invariant disturbance augmented model such as (10), the predicted future states $x_{i,k}$ for $i = 0 : N$ can be related to the current state x_k, current disturbance estimate $\hat{d}_k$, and input sequence $u_k, \ldots, u_{k+N-1}$ by:

$$x_{i,k} = A^i x_{0,k} + \sum_{j=0}^{i-1} A^j B u_{i-1-j,k} + \sum_{j=0}^{i-1} A^j B_d \hat{d}_k. \tag{C.1}$$

We define matrices $\mathbb{A}_x$, $\mathbb{B}_u$, and $\mathbb{B}_d$ as

$$\mathbb{A}_x = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{N-1} \\ A^N \end{bmatrix}, \tag{C.2}$$

$$\mathbb{B}_u = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix}, \tag{C.3}$$

$$\mathbb{B}_d = \begin{bmatrix} B_d & 0 & \cdots & 0 \\ AB_d & B_d & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B_d & A^{N-2}B_d & \cdots & B_d \end{bmatrix} \begin{bmatrix} I \\ \vdots \\ \vdots \\ I \end{bmatrix}. \tag{C.4}$$

Using (C.2), (C.3), and (C.4), we rewrite (C.1) for the stacked predicted states and input sequence introduced in notation 2:

$$\boldsymbol{X}_k = \mathbb{A}_x \hat{x}_k + \mathbb{B}_u \boldsymbol{U}_k + \mathbb{B}_d \hat{\boldsymbol{d}}_k. \tag{C.5}$$

C.2. Linear state and actuator constraints

Here the linear and nonlinear constraints are derived. The pellet particle input is constrained as $0 \leqslant \Gamma_{pellet} \leqslant \Gamma_{pellet}^{\max}$. The pellet injectors are designed to provide a mass flow rate $\dot{m} = 0.23 + 30\% = 0.33 \mathrm{gs}^{-1}$ of solid DT to the plasma [51]. The maximum electron fueling rate is than given by

$$\dot{\Gamma}_{pellet}^{\max} = \dot{m} * (f_D M_D + f_T M_T) * N_{avg}, \tag{C.6}$$

where f_D and f_T are the fractions of deuterium and tritium atoms in the pellet, M_D and M_T the molecular masses of the two isotopes and N_{avg} representing the Avogadro constant. Assuming a pellet composed at 50% of deuterium and 50% of tritium, the maximal fueling rate is equal to $\dot{\Gamma}_{pellet}^{\max} = 4.115e22 \ [\#/s]$. Hence, the number of injected particles is constraint by $\Gamma_{pellet}^{\max} = 2.57e21 \ [\#]$.

No lower constraint exist for the states hence $x_{\min} = -\infty$. The Greenwald density limit [9] sets an upper bound on the line-average electron density in a specific device configuration. This upper limit is given by

$$n_{gw} = \frac{I_p}{\pi a^2} \tag{C.7}$$

where I_p is the plasma current and a the minor radius of the tokamak. By constructing a matrix $Z \in \mathbb{R}^{1 \times n_x}$ such that $Zx \approx \bar{n}$ where $\bar{n}$ is the line-average density, this limit can be formulated as a linear state constraint.

The linear constraints are given by:

$$
\begin{aligned}
& u_{\min} \leqslant u_{i,k} \leqslant u_{\max}, \ \ \forall i = 0, 1, \cdots, N-1, \\
& x_{\min} \leqslant x_{i,k} \ \forall i = 0, 1, \cdots, N, \\
& Zx_{i,k} \leqslant n_{gw} \ \forall i = 1, 2, \cdots, N,
\end{aligned}
\tag{C.8}
$$

By defining

$$
M_i = \begin{pmatrix} 0_{n_u \times n_x} \\ 0_{n_u \times n_x} \\ -I_{n_x} \\ Z \end{pmatrix}, \ E_i = \begin{pmatrix} -I_{n_u} \\ I_{n_u} \\ 0_{n_x \times n_u} \\ 0_{1 \times n_u} \end{pmatrix}, \ b_i = \begin{pmatrix} -u_{\min} \\ u_{\max} \\ -x_{\min} \\ n_{gw} \end{pmatrix}
$$

$$
M_N = \begin{pmatrix} -I_{n_x} \\ Z \end{pmatrix}, \ \text{and} \ b_N = \begin{pmatrix} -x_{\min} \\ n_{gw} \end{pmatrix}, \tag{C.9}
$$

we rewrite (C.8) as

$$
\begin{aligned}
& M_i x_{i,k} + E_i u_{i,k} \leqslant b_i \ \ \forall i = 0, 1, \ldots, N-1 \\
& M_N x_{N,k} \leqslant b_N.
\end{aligned}
\tag{C.10}
$$

Using the stacked notation (13) and defining $\mathcal{M}_0$, $\mathcal{M}_i$, $\mathcal{E}_i$ and $\mathbf{b}$ as

$$\mathcal{M}_0 = \begin{pmatrix} M_0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{C.11}$$

$$\mathcal{M}_i = \begin{pmatrix} 0 & \cdots & 0 \\ M_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & M_N \end{pmatrix}, \tag{C.12}$$

$$\mathcal{E}_i = \begin{pmatrix} E_0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & E_{N-1} \\ 0 & \cdots & 0 \end{pmatrix}, \tag{C.13}$$

$$\mathbf{b} = \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{pmatrix}, \tag{C.14}$$

the constraints in (C.10) are grouped as

$$\mathcal{M}_0 x(k) + \mathcal{M}_i \boldsymbol{X}_k + \mathcal{E}_i \boldsymbol{U}_k \leqslant \mathbf{b}. \tag{C.15}$$

Finally, the matrices of (15) are obtained by inserting (C.5) in (C.15) and are defined as:

$$A_{ineq} = \mathcal{M}_i \Gamma_u + \mathcal{E}_i, \tag{C.16}$$

$$b_{ineq} = \mathbf{b} - (\mathcal{M}_0 + \mathcal{M}_i \Gamma_x) x(k) - \mathcal{M}_i \Gamma_d d(k). \tag{C.17}$$

C.3. Nonlinear state constraints

Here we formulate the nonlinear constraint function g_1 introduced in section 5.5. Recalling that $y(t_{k+1}) = n_e(\bar{\rho}, t_{k+1})$, with $\bar{\rho}$ defined in section 3.1, the augmented model (10) can be used to relate the density profile at time $t = t_{k+1}$ can be related to the state and inputs at time $t = t_k$. The relation is given by:

$$\begin{aligned} n_e(\bar{\rho}, t_{k+1}) &= y(t_{k+1}) \\ &= Cx(t_{k+1}) + C_d d(t_{k+1}) \\ &= C(Ax(t_k) + Bu(t_k) + B_d d(t_k)) + \\ & \quad C_d d(t_{k+1}). \end{aligned} \tag{C.18}$$

Under the assumption of slow varying disturbances, such that $d(t_{k+1}) \approx d(t_k)$, we can write

$$n_e(\bar{\rho}, t_{k+1}) = C(Ax(t_k) + Bu(t_k) + B_d d(t_k)) + C_d d(t_k). \tag{C.19}$$

Finally, we define C' as

$$C' = \frac{\partial \left(\sum_{\alpha=1}^m \Lambda_\alpha(\bar{\rho}) \right)}{\partial \rho}, \tag{C.20}$$

such that $C'x(t_k) = \frac{\partial n_e(\rho, t_k)}{\partial \rho}$ and we construct an output weight matrix W_y to weight the importance of the outputs in the nonlinear constraint. In this work we chose this matrix such that the weight of the outputs for which $\rho < 0.9$ are unity and the outputs for which $\rho > 0.9$ are weighted 0.
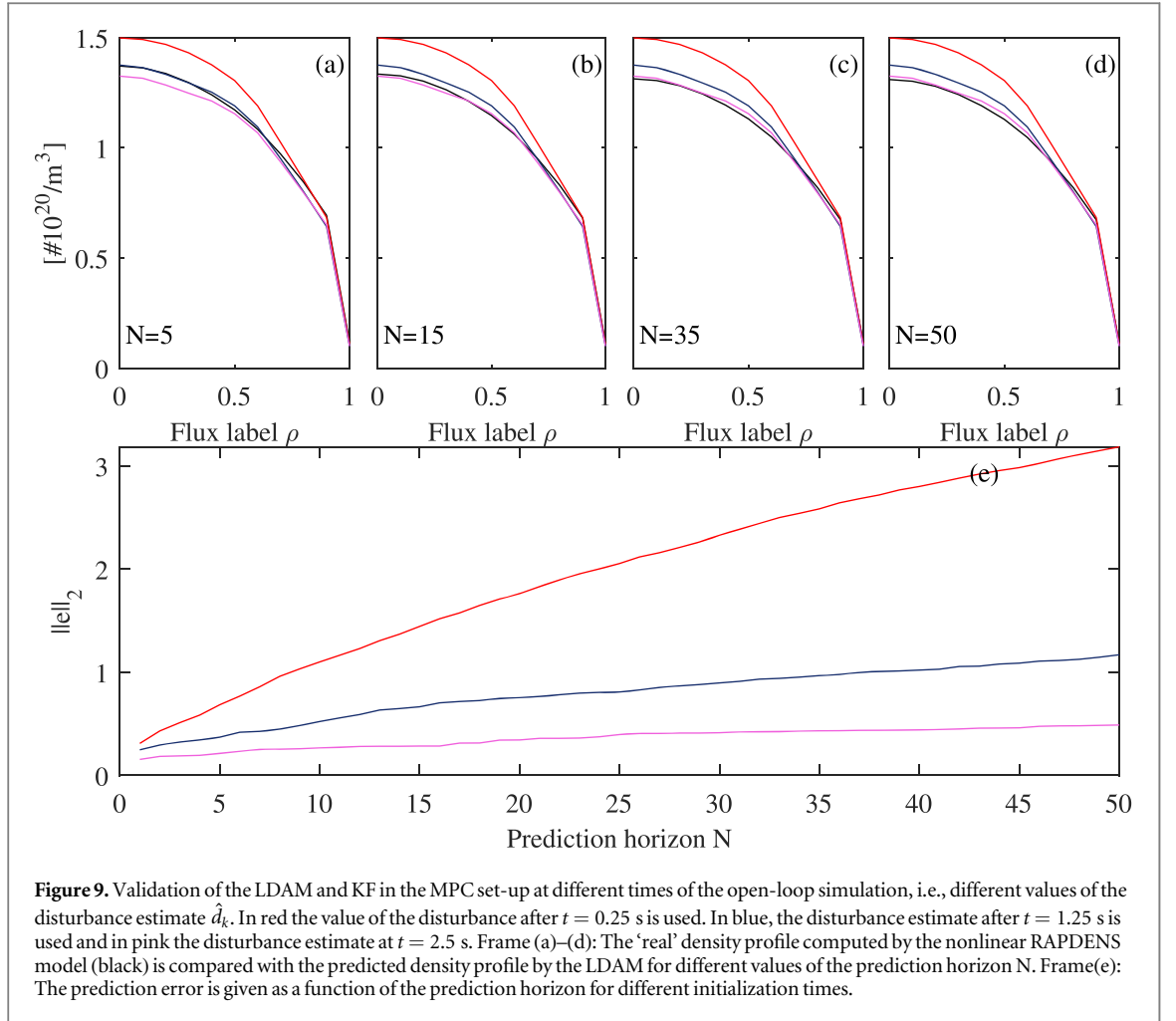
Inserting (C.19), (C.20), and W_y in (16) and rewriting the equation in negative null form, we can formulate the nonlinear constraint function g_1 as (time dependence denoted by subscript k for readability):

$$\begin{aligned} g_1(x_k, u_k, d_k, \epsilon) &\equiv |W_y C'(Ax_k + Bu_k + B_d d_k)| - \\ & |W_y(C(Ax_k + Bu_k + B_d d_k) + C_d d_k)(0.95 W_y L_{T_i} + \epsilon)| \leqslant 0. \end{aligned} \tag{C.21}$$

C.4. Cost function matrices

Here we derive the matrices of the numerical implementation of the cost function (18) as presented in (21).

First, similar as (13), we define the stacked steady-state states and inputs as

Figure 9. Validation of the LDAM and KF in the MPC set-up at different times of the open-loop simulation, i.e., different values of the disturbance estimate $\hat{d}_k$. In red the value of the disturbance after $t = 0.25$ s is used. In blue, the disturbance estimate after $t = 1.25$ s is used and in pink the disturbance estimate at $t = 2.5$ s. Frame (a)–(d): The 'real' density profile computed by the nonlinear RAPDENS model (black) is compared with the predicted density profile by the LDAM for different values of the prediction horizon N. Frame(e): The prediction error is given as a function of the prediction horizon for different initialization times.

$$
\bar{X}_k = \begin{bmatrix} \bar{x}_{1|k} \\ \bar{x}_{2|k} \\ \vdots \\ \bar{x}_{N-1|k} \\ \bar{x}_{N|k} \end{bmatrix} \quad \bar{U}_k = \begin{bmatrix} \bar{u}_{0|k} \\ \bar{u}_{1|k} \\ \vdots \\ \bar{u}_{N-2|k} \\ \bar{u}_{N-1|k} \end{bmatrix}. \tag{C.22}
$$

Subsequently, the cost matrices W_x, W_u, and W_N first introduced in (18) are used to define the block-diagonal matrices $\mathbb{W}_x$ and $\mathbb{W}_u$ as

$$
\mathbb{W}_x = \begin{bmatrix} W_x & 0 & \cdots & 0 \\ 0 & W_x & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & W_N \end{bmatrix}, \tag{C.23}
$$

$$
\mathbb{W}_u = \begin{bmatrix} W_u & 0 & \cdots & 0 \\ 0 & W_u & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & W_u \end{bmatrix}. \tag{C.24}
$$

Using (13), (C.22), (C.23), and (C.24) the cost function (18) can be rewritten as

$$
\begin{aligned}
J_k = \; & x(k)^\top Q x(k) + X_k^\top \mathbb{W}_x X_k - 2 X_k^\top \mathbb{W}_x \bar{X}_k \\
& + \bar{X}_k^\top \mathbb{W}_x \bar{X}_k + U_k^\top \mathbb{W}_u U_k - 2 U_k^\top \mathbb{W}_u \bar{U}_k \\
& + \bar{U}_k^\top \mathbb{W}_u \bar{U}_k.
\end{aligned} \tag{C.25}
$$

Inserting (C.5) in (C.25) we obtain:

$$
\begin{aligned}
J_k ={}& x(k)^\top Q x(k) + x(k)^\top \Gamma_x^\top \mathbb{W}_x \Gamma_x x(k) \\
&+ 2U_k^\top \Gamma^\top \mathbb{W}_x \Gamma_x x(k) + U_k^\top \Gamma_u^\top \mathbb{W}_x \Gamma_u U_k \\
&+ 2x_k^\top \Gamma_x^\top \mathbb{W}_x \Gamma_d d(k) + 2U_k^\top \Gamma_u^\top \mathbb{W}_x \Gamma_d d(k) \\
&+ d(k)^\top \Gamma_d^\top \mathbb{W}_x \Gamma_d d(k) - 2x(k)^\top \Gamma_x^\top \mathbb{W}_x \bar{X}_k \\
&- 2U_k^\top \Gamma_u^\top \mathbb{W}_x \bar{X}_k - 2d(k)^\top \Gamma_d^\top \mathbb{W}_x \bar{X}_k \\
&+ \bar{X}_k^\top \mathbb{W}_x \bar{X}_k + U_k^\top \mathbb{W}_u U_k - 2U_k^\top \mathbb{W}_u \bar{U}_k \\
&+ \bar{U}_k^\top \mathbb{W}_u \bar{U}_k.
\end{aligned}
\tag{C.26}
$$

Finally, by regrouping the quadratic terms in U_k, the terms that depend on $\{U_k, x(k)\}$, $\{U_k, d(k)\}$, and $\{U_k, \bar{X}_k, \bar{U}_k\}$ we can define the matrices $\mathbb{W}_{u,u}$, $\mathbb{W}_{u,x}$, $\mathbb{W}_{u,d}$, and $\mathbb{W}_{u,r}$ from (21) as

$$
\begin{aligned}
\mathbb{W}_{u,u} &= \mathbb{W}_u + \Gamma_u^\top \mathbb{W}_x \Gamma_u, & \mathbb{W}_{u,x} &= 2\Gamma_u^\top \mathbb{W}_x \Gamma_x, \\
\mathbb{W}_{u,d} &= 2\Gamma_u^\top \mathbb{W}_x \Gamma_d, & \mathbb{W}_{u,r} &= -2\Gamma_u \mathbb{W}_x \bar{X}_k - 2\mathbb{W}_u \bar{U}_k,
\end{aligned}
$$

C.5. Choice of prediction horizon

The predictive capacities of the LDAM depend greatly on the estimated disturbance $\hat{d}_k$. The choice of the prediction horizon N is made to trade off control performances and prediction accuracy. In figures 9(a)–(d) the predicted density profile is shown for different prediction horizons and initialization times. The red lines are obtained with the disturbance estimate $\hat{d}_k$ at time $k = 0.25$ s. The blue lines are obtained with the disturbance estimate $\hat{d}_k$ at time $k = 1.25$ s. Finally, the pink lines are obtained with the disturbance estimate $\hat{d}_k$ at time $k = 2.5$ s. In (e), the two-norm of the prediction error is shown as a function of the prediction horizon for the three initialization times.

It can be seen in figure 9(e) that the prediction error increases with the prediction horizon but that is increase is very small when the disturbance estimate has reached the steady-state values (see pink line in 9(e)) and in that case large prediction horizons can be used accurately. In this work, we have chosen to work with N = 20.

ORCID iDs

T O S J Bosman ⬤ https://orcid.org/0000-0002-7922-7974
M van Berkel ⬤ https://orcid.org/0000-0001-6574-3823
M R de Baar ⬤ https://orcid.org/0000-0003-4515-3468

References

[1] Shimomura Y, Aymar R, Chuyanov V, Huguet M and Parker R 1999 *Nucl. Fusion* **39** 1295–308
[2] Pironti A and Walker M 2005 *IEEE Control Systems* **25** 24–9
[3] Wesson J and Campbell D 2011 *Tokamaks* (Oxford: OUP)
[4] Angioni C, Fable E, Greenwald M, Maslov M, Peeters A G, Takenaga H and Weisen H 2009 *Plasma Phys. Controlled Fusion* **51** 124017
[5] Kessel C E 1994 *Nucl. Fusion* **34** 1221–38
[6] Angioni C and Helander P 2014 *Plasma Phys. Controlled Fusion* **56** 124001
[7] Horton W 1999 *Reviews of Modern Physics* **71** 735–38
[8] Garbet X 2006 *C.R. Phys.* **7** 573–83
[9] Greenwald M 2002 *Plasma Phys. Controlled Fusion* **44** R27
[10] Boozer A H 2012 *Phys. Plasmas* **19** 058101
[11] Zohm H 2015 *Magnetohydrodynamic Stability of Tokamaks* (New York, NY: Wiley)
[12] Gribov Y, Humphreys D, Kajiwara K, Lazarus E A, Lister J B, Ozeki T, Portone A, Shimada M, Sips A C and Wesley J C 2007 *Nucl. Fusion* **47** S385
[13] Loarte A *et al* 2007 *Nucl. Fusion* **47** S203
[14] Kurihara K, Lister J B, Humphreys D A, Ferron J R, Treutterer W, Sartori F, Felton R, Brémond S and Moreau P 2008 *Fusion Eng. Des.* **83** 959–70
[15] Biel W, De Baar M, Dinklage A, Felici F, König R, Meister H, Treutterer W and Wenninger R 2015 *Fusion Engineering and Design* **96-97** 8–15
[16] Humphreys D *et al* 2015 *Phys. Plasmas* **22** 021806
[17] de Vries P C *et al* 2018 *Fusion Eng. Des.* **129** 334–40
[18] Albanese R *et al* 2005 *Fusion Engineering and Design* **74** 627–32
[19] Boyer M D, Barton J, Schuster E, Walker M L, Luce T C, Ferron J R, Penaflor B G, Johnson R D and Humphreys D A 2014 *IEEE Trans. Control Syst. Technol.* **22** 1725–39
[20] Barton J E *et al* 2015 *Nucl. Fusion* **55** 093005
[21] Moreau D, Artaud J F, Ferron J R, Holcomb C T, Humphreys D A, Liu F, Luce T C, Park J M, Prater R, Turco F and Walker M L 2015 *Nucl. Fusion* **55** 063011
[22] Maljaars E and Felici F 2017 *Fusion Eng. Des.* **122** 94–112

[23] Maljaars E, Felici F, De Baar M R, Van Dongen J, Hogeweij G M, Geelen P J and Steinbuch M 2015 *Nucl. Fusion* **55** 023001
[24] Blanken T C, Felici F, Rapson C J, de Baar M R and Heemels W P 2018 *Fusion Eng. Des.* **126** 87–103
[25] Blanken T C *et al* 2019 *Fusion Eng. Des.* **147** 111211
[26] Ravensbergen T, De Vries P C, Felici F, Blanken T C, Nouailletas R and Zabeo L 2018 *Nucl. Fusion* **58** 016048
[27] Lang P T *et al* 2018 *Nucl. Fusion* **58** 036001
[28] Bosman T, Kudláček O, Fable E, van Berkel M, Felici F, Bock A, Luda T and de Baar M 2021 *Fusion Eng. Des.* **170** 112510
[29] Hughes T J R 2000 *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis* (New York, NY: Dover Publications)
[30] de Boor C 1978 *A Practical Guide to Splines* (New York, NY: Springer)
[31] Gowers C, Nielsen P and Salzmann H 2016 *J. Instrum.* **11** P02003
[32] Bassan M, Andrew P, Kurskiev G, Mukhin E, Hatae T, Vayakis G, Yatsuka E and Walsh M 2016 *J. Instrum.* **11** C01052
[33] Anderson B and Moore J 1979 *Optimal Filtering* (Englewood Cliffs, NJ: Prentice-Hall)
[34] Koechl F *et al* 2018 Optimising the ITER 15MA DT Baseline Scenario By Exploiting A Self-Consistent Free-Boundary CORE-EDGE-SOL Workflow in IMAS *27th IAEA Fusion Energy Conference* P7–25
[35] Imbeaux F *et al* 2015 *Nucl. Fusion* **55** 123006
[36] Lukash V E, Raju D, Dokouka V N, Favez J Y, Khayrutdinov R R and Lister J B 2003 *Fusion Eng. Des.* **66-68** 767–70
[37] Lukash V, Gribov Y, Kavin A, Khayrutdinov R and Cavinato M 2005 *Plasma Devices Oper.* **13** 143–56
[38] Lukash V E, Kavin A A, Gribov Y V, Khayrutdinov R R and Mineev A B 2011 Simulation of ITER plasma scenarios starting from initial discharge of central solenoid *38th EPS Conference on Plasma Physics*
[39] Romanelli M *et al* 2014 *Plasma and Fusion Research* **9** 3403023
[40] Stoneking M R and Den Hartog D J 1997 *Rev. Sci. Instrum.* **68** 914–7
[41] Togashi H *et al* 2014 *Rev. Sci. Instrum.* **85** 2–5
[42] Tanimura Y and Iida T 1998 *J. Nucl. Mater.* **258-263** 1812–6
[43] Qin J and Badgwell T 2003 *Control Engineering Practice* **11** 733–64
[44] Lee J H 2011 *International Journal of Control, Automation and Systems* **9** 415–24
[45] Maruyama S, Yang Y, Sugihara M, Pitts R A, Baylor L, Combs S, Meitner S, Li B and Li W 2009 *2009 23rd IEEE/NPSS Symposium on Fusion Engineering* 1-4
[46] Pannocchia G 2003 *J. Process Control* **13** 693–701
[47] Maeder U, Borrelli F and Morari M 2009 *Automatica* **45** 2214–22
[48] Vada J, Slupphaug O and Johansen T 1999 *1999 European Control Conference (ECC)* 1166–71
[49] Rawlings J, Mayne D and Diehl M 2017 *Model Predictive Control: Theory, Computation, and Design* (Madison: Nob Hill Publishing)
[50] Nocedal J and Wright S 2006 *Numerical Optimization* (New York, NY: Springer)
[51] Baylor L R, Parks P B, Jernigan T C, Caughman J B, Combs S K, Foust C R, Houlberg W A, Maruyama S and Rasmussen D A 2007 *Nucl. Fusion* **47** 443–8
[52] van de Plassche K L, Citrin J, Bourdelle C, Camenen Y, Casson F J, Dagnelie V I, Felici F, Ho A and Van Mulders S 2020 *Phys. Plasmas* **27** 022310
[53] Hinton L and Hazeltine R D 1976 *Rev. Mod. Phys.* **48** 239