Accepted Manuscript

A practical globalization of one-shot optimization for optimal design of tokamak divertors

Maarten Blommaert, Wouter Dekeyser, Martine Baelmans, Nicolas R. Gauger,
Detlev Reiter

PII: S0021-9991(16)30539-3
DOI: <http://dx.doi.org/10.1016/j.jcp.2016.10.041>
Reference: YJCPH 6919

To appear in: *Journal of Computational Physics*

Received date: 4 December 2015
Revised date: 22 August 2016
Accepted date: 14 October 2016

Please cite this article in press as: M. Blommaert et al., A practical globalization of one-shot optimization for optimal design of tokamak divertors, *J. Comput. Phys.* (2016), <http://dx.doi.org/10.1016/j.jcp.2016.10.041>

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

A Practical Globalization of One-shot Optimization for Optimal Design of Tokamak Divertors

Maarten Blommaert¹, Wouter Dekeyser², Martine Baelmans², Nicolas R. Gauger³, Detlev Reiter¹

Abstract

In past studies, nested optimization methods were successfully applied to design of the magnetic divertor configuration in nuclear fusion reactors. In this paper, so-called one-shot optimization methods are pursued. Due to convergence issues, a globalization strategy for the one-shot solver is sought. Whereas Griewank introduced a globalization strategy using a doubly augmented Lagrangian function that includes primal and adjoint residuals, its practical usability is limited by the necessity of second order derivatives and expensive line search iterations. In this paper, a practical alternative is offered that avoids these drawbacks by using a regular augmented Lagrangian merit function that penalizes only state residuals. Additionally, robust rank-two Hessian estimation is achieved by adaptation of Powell's damped BFGS update rule. The application of the novel one-shot approach to magnetic divertor design is considered in detail. For this purpose, the approach is adapted to be complementary with practical in parts adjoint sensitivities. Using the globalization strategy, stable convergence of the one-shot approach is achieved.

Keywords: one-shot optimization, globalization strategy, continuous adjoint method, damped Hessian updates, magnetic divertor design, target heat load

Email address: m.blommaert@fz-juelich.de (Maarten Blommaert)

¹Institute of Energy and Climate Research (IEK-4), FZ Jülich GmbH, D-52425 Jülich, Germany

²KU Leuven, Department of Mechanical Engineering, 3001 Leuven, Belgium

³TU Kaiserslautern, Chair for Scientific Computing, 67663 Kaiserslautern, Germany

1. Introduction

Adjoint based optimal design methods have been applied to solve design problems governed by partial differential equations in different research fields. The design objective is hereby recasted into a cost or objective functional that represents the performance of possible designs and minimized using an optimization algorithm. Especially nested optimization methods are wide spread, a.o. since the state solver can be integrally reused [1]. In each optimization step of such a nested optimization, the segregated state solver of the partial differential equations is iterated up to convergence to evaluate the cost functional. Subsequently, a segregated adjoint simulation is performed to efficiently evaluate the gradient of the cost functional with respect to the design variables. Any gradient-based optimization method can then in principle be used to find the optimal design. An attractive alternative to the nested optimization approach is offered by one-shot optimization. This method aims at solving state, adjoint and design ('optimality') equations, simultaneously. Instead of solving state and adjoint equations sequentially in each optimization step, a single-step one-shot algorithm only performs one iteration of state and adjoint equations before making a design step. As such, the one-shot procedure again leaves the structure of the state and adjoint solver untouched, and uses them to construct an appropriate iterative solution procedure for the Karush-Kuhn-Tucker (KKT) conditions. Often, the one-shot procedure leads to a significantly reduced computational cost in comparison to nested optimization methods [2, 3].

In past studies, adjoint shape sensitivity analysis was deployed in a one-shot optimization procedure to design the divertor target shape in fusion tokamaks [4, 5]. Peak target heat loads were successfully reduced, eventually including radiative heat loads that were evaluated with a Monte Carlo procedure [6]. Nested optimization methods were also successfully applied to design the magnetic divertor configuration [7]. Since a full adjoint procedure, including sensitivities of a specific field-aligned grid generator, appeared cumbersome, an in parts adjoint procedure was developed that focuses on the expensive plasma edge transport simulation [8, 9].

This paper considers the application of one-shot optimization to magnetic divertor design. The main challenge appeared to be the sporadic presence of gradient discontinuities. Their destructive effect on one-shot convergence is resolved by the implementation of a novel and practical one-shot globalization strategy. Only recently, a first globalization approach for one-shot methods

has been introduced by Hamdi and Griewank [10]. The strategy proposed in this paper builds on this method, while circumventing the associated practical difficulties and its significant computational cost. Additionally, the one-shot strategy is complemented with a robust yet cheap methodology for BFGS Hessian estimation. Furthermore, since only in parts adjoint sensitivities are at hand for magnetic divertor design, the methodology is adapted to meet the requirements of this application.

This paper is structured as follows. In section 2 the optimization problem is described in a general setting, after which the KKT (Karush-Kuhn-Tucker) conditions of the discretized optimization problem are listed. Subsequently, in section 3, the one-shot optimization strategy is derived, starting from a brief review of the theory by Hamdi and Griewank [11, 10]. In section 4, we introduce the optimal magnetic divertor design problem and the strategy for in parts adjoint sensitivity calculation. Finally, in section 5, the one-shot strategy is tested and discussed for magnetics design of poloidal field divertors.

2. Optimal design

2.1. Optimization problem

We consider the general divertor design problem as a constrained mathematical optimization problem

$$\begin{aligned} \min_{\boldsymbol{\varphi} \in \Phi_{\text{ad}}, \mathbf{q}} \quad & \mathcal{I}(\boldsymbol{\varphi}, \mathbf{q}) \\ \text{s.t.} \quad & \mathbf{c}(\boldsymbol{\varphi}, \mathbf{q}) = 0, \end{aligned} \tag{1}$$

where we make the distinction between control and state variables. The control or design variables $\boldsymbol{\varphi} \in X$ are the independent variables that need to be optimized. The state variables $\mathbf{q} \in Y$ are dependent variables that can be uniquely determined from the control variables by the state or model equations $\mathbf{c}(\boldsymbol{\varphi}, \mathbf{q}) = 0$, $\mathbf{c} : X \times Y \rightarrow Y$. The model equations under consideration are a set of partial differential equations. The scalar-valued convex objective $\mathcal{I}(\boldsymbol{\varphi}, \mathbf{q}) : X \times Y \rightarrow \mathbb{R}$ represents the design goal. We assume that X and Y are Hilbert spaces. This means that they are complete spaces equipped with an inner product $\langle \cdot, \cdot \rangle$. Additionally, we assume that the feasible set of control variables $\Phi_{\text{ad}} \subset X$ is a convex subset of X . The feasible set can be represented by $\Phi_{\text{ad}} = \{\boldsymbol{\varphi} \in X \mid \mathbf{h}(\boldsymbol{\varphi}) \leq 0, \mathbf{h} : X \rightarrow H\}$, where $\mathbf{h}(\boldsymbol{\varphi}) \leq 0$ are

so-called design constraints and H is a Hilbert space as well. The design constraints are not the focus of this paper and will therefore not be considered in the following sections.

After discretization, X and Y become finite dimensional. We will directly use the discrete framework in the following sections for ease of notation. For the extension of the one-shot convergence theory to a functional setting, we refer to [12]. We use the notation $\partial_{\mathbf{x}}f$ to denote the row vector of partial derivatives of a function $f = f(\mathbf{x}, \mathbf{y}) : \mathbb{R}^m \times \mathbb{R}^n \mapsto \mathbb{R}$ with respect to the components x_i of the subscripted vector $\mathbf{x}$. Similarly, we will denote the column vector of partial derivatives by $\nabla_{\mathbf{x}}f = \partial_{\mathbf{x}}f^\top$.

In discrete form, $\mathbf{c}(\boldsymbol{\varphi}, \mathbf{q})$ are the residuals of the discretized state equations. The state equations $\mathbf{c}(\boldsymbol{\varphi}, \mathbf{q}) = 0$ can then be converted into fixed-point iteration form $\mathbf{q} = \mathbf{G}(\boldsymbol{\varphi}, \mathbf{q})$, where the Jacobian $\partial_{\mathbf{q}}\mathbf{G}$ should have a spectral radius $\rho < 1$, in order to have a contractive iteration procedure $\mathbf{q}_{k+1} = \mathbf{G}(\boldsymbol{\varphi}, \mathbf{q}_k)$ that leads to a feasible solution $\mathbf{q} = \mathbf{q}(\boldsymbol{\varphi})$. The fixed-point solver $\mathbf{G}$ is then linked to $\mathbf{c}$ by

$$\mathbf{c} = A(\mathbf{G}(\boldsymbol{\varphi}, \mathbf{q}) - \mathbf{q}), \quad (2)$$

where A is given by the solution procedure. For a direct Newton solver, $A = -\partial_{\mathbf{q}}\mathbf{c}$. Often, additional relaxation needs to be introduced to ensure convergence of the state solver. In the plasma edge code, a false time stepping scheme is used to iterate the equation $\frac{\partial \mathbf{q}}{\partial t} = \mathbf{c}$ to steady state, giving rise to $A = -1/\omega (\partial_{\mathbf{q}}\mathbf{c} - \mathbb{I}/\Delta t)$, with $\mathbb{I}$ an identity matrix of appropriate dimensionality and ω a constant relaxation factor. In reality, the state equations are solved with a segregated solver that gives rise to an inexact approximation of the latter expression.

2.2. One-shot optimization

Consider the Lagrangian functional

$$\mathcal{L}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{q}^*) = \mathcal{I}(\boldsymbol{\varphi}, \mathbf{q}) + \mathbf{c}(\boldsymbol{\varphi}, \mathbf{q})^\top \mathbf{q}^*, \quad (3)$$

where $\mathbf{q}^*$ are Lagrange multipliers. Setting the derivatives with respect to $\mathbf{q}^*$, $\mathbf{q}$ and $\boldsymbol{\varphi}$ equal to zero then leads to the KKT-conditions

$$\begin{cases} \nabla_{\mathbf{q}^*}\mathcal{L}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{q}^*) = \mathbf{c}(\boldsymbol{\varphi}, \mathbf{q}) = 0 & \text{State} \\ \nabla_{\mathbf{q}}\mathcal{L}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{q}^*) = \nabla_{\mathbf{q}}\mathcal{I}(\boldsymbol{\varphi}, \mathbf{q}) + \partial_{\mathbf{q}}\mathbf{c}(\boldsymbol{\varphi}, \mathbf{q})^\top \mathbf{q}^* = 0 & \text{Adjoint} \\ \nabla_{\boldsymbol{\varphi}}\mathcal{L}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{q}^*) = \nabla_{\boldsymbol{\varphi}}\mathcal{I}(\boldsymbol{\varphi}, \mathbf{q}) + \partial_{\boldsymbol{\varphi}}\mathbf{c}(\boldsymbol{\varphi}, \mathbf{q})^\top \mathbf{q}^* = 0 & \text{Design} \end{cases} \quad (4)$$

that represent the first order necessary conditions for optimality. In former papers, it was illustrated how a nested optimization solver based on Sequential Quadratic Programming (SQP) can be used to effectively find state, adjoint and design variables $\bar{\mathbf{q}}, \bar{\mathbf{q}}^*$ and $\bar{\varphi}$ that satisfy the KKT conditions of the magnetic divertor design problem [8, 13]. In this nested approach, state and adjoint equations are fully converged in each optimization iteration and used as a tool for efficient sensitivity evaluations. The latter are then transferred to a black-box optimization routine. In contrast to the nested approach, the one-shot strategy simultaneously converges the state, adjoint, and design equation in the KKT system (4). Moreover, they do so without requiring major modifications to state and adjoint solvers. For example, the following one-shot step iteration can be used to converge to a KKT point:

$$\begin{aligned}\mathbf{q}_{k+1} &= \mathbf{q}_k + A^{-1} \mathbf{c}(\varphi_k, \mathbf{q}_k), \\ \mathbf{q}_{k+1}^* &= \mathbf{q}_k^* + A^{-\top} \nabla_{\mathbf{q}} \mathcal{L}(\varphi_k, \mathbf{q}_{k+1}, \mathbf{q}_k^*), \\ \varphi_{k+1} &= \varphi_k + B^{-1} \nabla_{\varphi} \mathcal{L}(\varphi_k, \mathbf{q}_{k+1}, \mathbf{q}_{k+1}^*).\end{aligned}\tag{5}$$

The first two steps in this iterator are simply a single iteration of the state and adjoint solver, respectively, thus requiring very little coding. One-shot methods are closely related to inexact reduced Sequential Quadratic Programming (rSQP) methods in which an approximation of the gradient $\partial_{\mathbf{q}} c \approx -A$ is used [14]. The above method then correspond to a Gauss-seidel type iterator, while a parallel solution of these equations would correspond to a Jacobi-type iterator.

If algorithmic differentiation (AD) is used to find the adjoint variables, the KKT conditions should be derived from the Lagrangian [11]

$$\mathcal{L}_G(\varphi, \mathbf{q}, \mathbf{y}^*) = \mathcal{I}(\varphi, \mathbf{q}) + (\mathbf{G}(\varphi, \mathbf{q}) - \mathbf{q})^\top \mathbf{y}^* = \mathcal{N}(\varphi, \mathbf{q}, \mathbf{y}^*) - \mathbf{q}^\top \mathbf{y}^*, \tag{6}$$

where the fixed-point form of the state equations is used. The resulting KKT-conditions then read

$$\begin{cases} \mathbf{q} = \mathbf{G}(\varphi, \mathbf{q}) & \text{State} \\ \mathbf{y}^* = \nabla_{\mathbf{q}} \mathcal{N} = \nabla_{\mathbf{q}} \mathcal{I}(\varphi, \mathbf{q}) + \partial_{\mathbf{q}} \mathbf{G}(\varphi, \mathbf{q})^\top \mathbf{y}^* & \text{Adjoint} \\ 0 = \nabla_{\varphi} \mathcal{N} = \nabla_{\varphi} \mathcal{I}(\varphi, \mathbf{q}) + \partial_{\varphi} \mathbf{G}(\varphi, \mathbf{q})^\top \mathbf{y}^* & \text{Design} \end{cases} \tag{7}$$

A one-shot iterator can again be constructed to solve for a feasible point of these KKT conditions. When using AD, the state and derivative are

calculated simultaneously. Therefore, the one-shot algorithm corresponds to the parallel iterates [14]:

$$\begin{cases} \mathbf{q}_{k+1} = \mathbf{G}(\boldsymbol{\varphi}_k, \mathbf{q}_k), \\ \mathbf{y}_{k+1}^* = \nabla_{\mathbf{q}} \mathcal{N}(\boldsymbol{\varphi}_k, \mathbf{q}_k, \mathbf{y}_k^*), \\ \boldsymbol{\varphi}_{k+1} = \boldsymbol{\varphi}_k - B_k^{-1} \nabla_{\boldsymbol{\varphi}} \mathcal{N}(\boldsymbol{\varphi}_k, \mathbf{q}_k, \mathbf{y}_k^*) \end{cases}. \quad (8)$$

The adjoint iterates

$$\mathbf{y}_{k+1}^* = \nabla_{\mathbf{q}} \mathcal{N}(\boldsymbol{\varphi}, \mathbf{q}_k, \mathbf{y}_k^*). \quad (9)$$

converge here to slightly different adjoint variables $\bar{\mathbf{y}}^* = A^\top \bar{\mathbf{q}}^*$, with the overbar notation indicating the values at the optimum.

Notice that the adjoint equations in the KKT systems (4) and (7) are the so-called discrete adjoint equations since a discretize-then-optimize approach is written here. In the application considered in section 4, an alternative optimize-then-discretize approach will be used. The influence on the results of this modification will be considered in section 5.3.

3. Globalized one-shot optimization

In the nested approach to optimal magnetic divertor design, globalization was achieved through a line search strategy that enforces the strong Wolfe conditions [7]. It is clear that in combination with one-shot optimization, these methods need revisiting. Indeed, looking solely at the objective function data in line searches is misleading since state and costate are not converged in each optimization step. Only recently, a first globalization approach for one-shot methods has been derived by Hamdi and Griewank [11, 10]. In section 3.1, we will give a summary of their globalization approach based on an exact penalty function of doubly augmented Lagrangian type [15, 16, 17, 18] and indicate the challenges for its practical application. Next, starting from this sound theoretical basis, we propose a modified strategy in sections 3.2 to 3.5 that aims at reducing implementation efforts and increasing the efficiency.

3.1. One-shot optimization using a doubly augmented Lagrangian function

3.1.1. Review of theory

The one-shot method of Hamdi and Griewank looks for descent on the doubly augmented Lagrangian function

$$L_G^a(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) = \frac{\alpha_G}{2} \|\mathbf{G}(\boldsymbol{\varphi}, \mathbf{q}) - \mathbf{q}\|^2 + \frac{\beta_G}{2} \|\nabla_{\mathbf{q}} \mathcal{N}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) - \mathbf{y}^*\|^2 + \mathcal{N}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) - \mathbf{q}^\top \mathbf{y}^*, \quad (10)$$

with α_G and β_G positive weighting coefficients for the forward and dual feasibility penalties, respectively. It was proven in [11] to be an exact penalty function if the weighting coefficients obey the inequality

$$\alpha_G \beta_G (1 - \rho)^2 > 1 + \beta_G \theta, \quad \text{with} \quad \theta = \|\partial_{\mathbf{q}\mathbf{q}} \mathcal{N}\|. \quad (11)$$

Furthermore, it is shown that the step increment vector

$$\mathbf{s}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) = \begin{bmatrix} \Delta \mathbf{q} = \mathbf{G}(\boldsymbol{\varphi}, \mathbf{q}) - \mathbf{q} \\ \Delta \mathbf{y}^* = \partial_{\mathbf{q}} \mathcal{N}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*)^\top - \mathbf{y}^* \\ \Delta \boldsymbol{\varphi} = -B^{-1} \partial_{\boldsymbol{\varphi}} \mathcal{N}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*)^\top \end{bmatrix}, \quad (12)$$

is a descent direction of the doubly augmented Lagrangian, if the stronger condition

$$\sqrt{\alpha_G \beta_G} (1 - \rho) > 1 + \frac{\beta_G}{2} \theta \quad (13)$$

is met and if a large positive design space preconditioner B is chosen.

To prove the former, the gradient of the augmented Lagrangian is written in [11] as

$$\nabla L_G^a(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) = -M \mathbf{s}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*), \quad \text{where} \quad (14)$$

$$M = \begin{bmatrix} \alpha_G \Delta \mathbf{G}_q^\top & -\mathbb{I} - \beta_G \partial_{\mathbf{q}\mathbf{q}} \mathcal{N} & 0 \\ -\mathbb{I} & \beta_G \Delta \mathbf{G}_q & 0 \\ -\alpha_G \partial_{\boldsymbol{\varphi}} \mathbf{G}^\top & -\beta_G \partial_{\mathbf{q}\boldsymbol{\varphi}} \mathcal{N}^\top & B \end{bmatrix}, \quad \text{with} \quad \Delta \mathbf{G}_q = \mathbb{I} - \partial_{\mathbf{q}} \mathbf{G}.$$

Requiring the step $\mathbf{s}$ to yield descent on L_G^a corresponds to requiring that

$$\mathbf{s}^\top \nabla L_G^a = -\mathbf{s}^\top M \mathbf{s} = -\mathbf{s}^\top \frac{M + M^\top}{2} \mathbf{s} < 0 \quad (15)$$

or that M or $(M + M^\top)/2$ is positive definite. From the latter requirement, it is found that B should be sufficiently large and that

$$\alpha_G \beta_G \Delta \bar{\mathbf{G}}_q \succ \left(\mathbb{I} + \frac{\beta_G}{2} \partial_{\mathbf{q}\mathbf{q}} \mathcal{N} \right) (\Delta \bar{\mathbf{G}}_q)^{-1} \left(\mathbb{I} + \frac{\beta_G}{2} \partial_{\mathbf{q}\mathbf{q}} \mathcal{N} \right), \quad (16)$$

where $\Delta\bar{\mathbf{G}}_q = 1/2 (\Delta\mathbf{G}_q + \Delta\mathbf{G}_q^\top)$. Condition (16) is then shown to be implied by (13).

Moreover, in [10] a sufficient condition for B was derived. Using large parameters α_G and β_G , equation (16) can easily be satisfied. However, increasing the penalties α_G and β_G leads to reduced design steps $\Delta\varphi$ and therefore slower convergence of the optimization procedure. The penalties should therefore be chosen as small as possible to obtain a fast convergence of the one-shot iterations. Using (13) as equality and minimizing α_G leads to the choice of weighting coefficients

$$\alpha_G = \frac{2\theta}{(1-\rho)^2} \quad \text{and} \quad \beta_G = \frac{2}{\theta}. \quad (17)$$

For the design preconditioner B , it is pointed out in [10] that $B \approx \partial_{\varphi\varphi} L_G^a$, which turns to an equality at primal and dual feasibility. A quasi-Newton BFGS update strategy was therefore proposed where

$$B\Delta\varphi \approx \nabla_{\varphi} L_G^a(\varphi + \Delta\varphi, \mathbf{q}, \mathbf{y}^*) - \nabla_{\varphi} L_G^a(\varphi, \mathbf{q}, \mathbf{y}^*) \quad (18)$$

is employed as a secant equation for the Hessian update, with

$$\partial_{\varphi} L_G^a = \alpha_G \Delta\mathbf{q}^\top \partial_{\varphi} G + \beta_G \Delta\mathbf{y}^{*\top} \partial_{\mathbf{q}\varphi} \mathcal{N} + \partial_{\varphi} \mathcal{N}. \quad (19)$$

Furthermore, a set of globalization strategies is listed that look for descent on L_G^a in the direction $\mathbf{s}$. Global convergence could then be proven under reasonable assumptions.

3.1.2. Discussion

The procedure of Hamdi and Griewank is the first procedure that guarantees convergence. However, implementation might be more cumbersome than the classical ad-hoc one-shot methods and the method might not be the fastest with respect to CPU time. In view of implementation efforts, especially the need of a second order adjoint (SOA) for the evaluation of $\Delta\mathbf{y}^{*\top} \partial_{\mathbf{q}\varphi} \mathcal{N}$ in (19) is a drawback. In particular in a continuous adjoint framework, the additional derivation and implementation of this second derivative are impractical. Furthermore, it adds two new equations to be solved to the iterative optimization procedure, namely the perturbed forward and the SOA equation, having also significant repercussions on computational cost. Alternatively, a finite difference approximation for this term is proposed

in [10]. Also with this approach, at least one additional state calculation is needed in each one-shot iteration. We will consider this difficulty further in section 3.2.

Secondly, the parameters derived are sufficient for convergence but not strictly necessary, so that smaller penalties might still be adequate and may lead to accelerated convergence. Also, it should be noted that choice of an appropriate norm for the feasibility penalties in equation (10) is not straightforward. Especially in application of the approach to a problem governed by a system of PDE's, it is clear that using a regular 2-norm might lead to unwanted results. Indeed, depending on the units or scaling of individual equations, the convergence of the one-shot approach might then be significantly altered. Especially in nuclear fusion applications, where densities and temperatures are not in everyday ranges, one equation (e.g. ion continuity equation) in $\|\mathbf{G}(\boldsymbol{\varphi}, \mathbf{q}) - \mathbf{q}\|$ might dominate all other equations so that the feasibility penalty of other equations ends up in the range of numerical error. Additionally, line searches on the augmented Lagrangian function are quite expensive, since they require design, state and costate evaluations. This motivates a reformulation of the feasibility penalty in section 3.3.

Thirdly, the evaluation of $\nabla_{\boldsymbol{\varphi}} L_G^a(\boldsymbol{\varphi} + \Delta\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*)$ needed in the Hessian update requires an additional evaluation of the design gradient. The cost of evaluating this design gradient might not be negligible in certain cases. In section 3.5, a more convenient alternative will be offered.

3.2. Considerations on the adjoint feasibility penalty

As discussed in section 3.1.2, the adjoint feasibility penalty in the doubly augmented Lagrangian (10) complicates implementation and slows down the one-shot optimization. Therefore, it may be worthwhile questioning the necessity of this term. In several applications, the θ occurring in the “optimal” penalty parameters (17) might well be approximated by $\theta \approx 1$ [19]. It is clear that for the slowly converging fixed-point solvers under consideration (ρ close to 1), α_G is several orders of magnitude larger than β_G . Additionally, the dynamics of the adjoint residuals often correlate very well with those of the state residuals. These arguments intuitively suggest that dropping the adjoint feasibility penalty might well be reasonable.

Formally, we indeed observe that the contraction of the adjoint iterator reads $\|\partial_{\mathbf{q}} \mathcal{N}\| = \|\partial_{\mathbf{q}} \mathbf{G}\| \leq \rho < 1$, which means that primal and adjoint have the same contraction rate. The adjoint equation will thus converge at the same rate as the primal, although the convergence may be somewhat delayed [20]. Because of the typically observed bounded retardation, we

assume that the ratio of dual and primal updates is bounded by a coefficient L as $\|\Delta \mathbf{y}^*\| \leq L\|\Delta \mathbf{q}\|$. A further consequence is that one can use so-called piggy-back iterations to simultaneously converge state and adjoint equations. In this procedure, iterations of state and adjoint equations are alternated to achieve the simultaneous convergence.

Using the above arguments, we derive under which conditions the step $\mathbf{s}_1$ is a descent direction of the alternative augmented Lagrangian

$$L_S^a(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) = \frac{\alpha_G}{2} \|\mathbf{G}(\boldsymbol{\varphi}, \mathbf{q}) - \mathbf{q}\|^2 + \mathcal{N}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) - \mathbf{y}^{*\top} \mathbf{q}, \quad (20)$$

which only has a single state penalty and with

$$\mathbf{s}_1(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) = \begin{bmatrix} \Delta \mathbf{q} = \mathbf{G}(\boldsymbol{\varphi}, \mathbf{q}) - \mathbf{q} \\ \Delta \mathbf{y}^* = \partial_{\mathbf{q}} \mathcal{N}(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*)^\top - \mathbf{y}^* \\ \Delta \boldsymbol{\varphi} = -B^{-1} \partial_{\boldsymbol{\varphi}} L_S^a(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*)^\top \end{bmatrix}. \quad (21)$$

In this step vector $\mathbf{s}_1$, we use $\partial_{\boldsymbol{\varphi}} L_S^a(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*)$ in the design step calculation to avoid any possible inconsistencies between descent on cost function and augmented Lagrangian function in the line search algorithm explained later. Similar to equation (14), requiring descent on L_S^a for a one-shot iteration corresponds to

$$\begin{aligned} \mathbf{s}_1^\top \nabla L_S^a &\approx -\mathbf{s}_1^\top M_1 \mathbf{s}_1 < 0, \text{ with} \\ M_1 &= \begin{bmatrix} \alpha_G \Delta \mathbf{G}_q^\top & -\mathbb{I} & 0 \\ -\mathbb{I} & 0 & 0 \\ 0 & 0 & B \end{bmatrix}. \end{aligned} \quad (22)$$

For a general vector $\mathbf{v} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]^\top$, one can observe that

$$\begin{aligned} \mathbf{v}^\top M_1 \mathbf{v} &= \alpha_G \mathbf{v}_1^\top \Delta \mathbf{G}_q^\top \mathbf{v}_1 - 2\mathbf{v}_2^\top \mathbf{v}_1 + \mathbf{v}_3^\top B \mathbf{v}_3, \\ &\geq \alpha_G(1 - \rho) \|\mathbf{v}_1\|^2 - 2\|\mathbf{v}_1\| \|\mathbf{v}_2\| + \lambda_{\min}(B) \|\mathbf{v}_3\|^2, \end{aligned}$$

with $\lambda_{\min}(B)$ the smallest eigenvalue of B . Using the bounded retardation of the adjoint residuals, descent is implied by

$$(\alpha_G(1 - \rho) - 2L) \|\Delta \mathbf{q}\|^2 + \lambda_{\min}(B) \|\Delta \boldsymbol{\varphi}\|^2 > 0.$$

It is thus sufficient to choose the design preconditioner B to be positive definite and

$$\alpha_G > \frac{2L}{1 - \rho}. \quad (23)$$

Since coupled state-costate-design iterations continuously descend on the augmented Lagrangian function L_S^a , the one-shot method will converge to the optimum $(\bar{\mathbf{q}}, \bar{\mathbf{y}}^*, \bar{\boldsymbol{\varphi}})$. We conclude that the adjoint residuals in the doubly augmented Lagrangian can thus be disregarded and as such we may significantly simplify the approach.

3.3. A practical augmented Lagrangian merit function

In section 3.1.2, it was also pointed out that the choice of norm in the penalty contributions is not evident, as the different PDE's have different units and scales. We therefore propose to use the normalized residuals of the discretized equations as a feasibility penalty instead of the state step $\Delta \mathbf{q}$. In many codes they serve as convergence monitor and are therefore at hand. This finally gives an augmented Lagrangian function of the form

$$L^a(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) = \frac{\alpha}{2} \sum_{i=1}^{n_E} \|\mathbf{c}_i(\boldsymbol{\varphi}, \mathbf{q})/s_{c,i}\|^2 + \mathcal{L}_G(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*), \quad (24)$$

with n_E then the number of equations, $s_{c,i}$ a normalization factor and i an index denoting the corresponding equation. When the equations are normalized in advance one finds that

$$L^a(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) = \frac{\alpha}{2} \|\mathbf{c}(\boldsymbol{\varphi}, \mathbf{q})\|^2 + \mathcal{L}_G(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*). \quad (25)$$

It may be interesting to see whether also in this form, an α and B can be found so that a coupled state-design step offers descent in L^a . The only change with respect to the derivation in section 3.2 is of course the feasibility penalty term. Using $\mathbf{c} = A\Delta \mathbf{q}$, we can again write the descent condition similar to equation (22), namely

$$\mathbf{s}_1^\top \nabla L^a \approx -\mathbf{s}_1^\top M_2 \mathbf{s}_1 < 0, \text{ with} \quad (26)$$

$$M_2 = \begin{bmatrix} -\alpha \partial_{\mathbf{q}} \mathbf{c}^\top A & -\mathbb{I} & 0 \\ -\mathbb{I} & 0 & 0 \\ 0 & 0 & B \end{bmatrix},$$

and where the only difference between M_2 and M_1 is clearly the $-\alpha \partial_{\mathbf{q}} \mathbf{c}^\top A$ term in the matrix M_2 . We will now attempt to find a lower bound for this term. For this purpose, one should first notice that

$$\partial_{\mathbf{q}} \mathbf{c} \Delta \mathbf{q} = -A \Delta \mathbf{G}_q \Delta \mathbf{q} + \partial_{\mathbf{q}} A \Delta \mathbf{q} \Delta \mathbf{q} \approx -A \Delta \mathbf{G}_q \Delta \mathbf{q}, \quad (27)$$

where the latter is a good approximation since $\Delta \mathbf{q}$ tends to zero during iterations and $\partial_{\mathbf{q}} A$, typically approximating $-\partial_{\mathbf{q}\mathbf{q}} \mathbf{c}$, is a higher order term. Using this, one finds

$$-\alpha \Delta \mathbf{q}^\top \partial_{\mathbf{q}} \mathbf{c}^\top A \Delta \mathbf{q} \approx \alpha \Delta \mathbf{q}^\top \Delta \mathbf{G}_{\mathbf{q}}^\top A^\top A \Delta \mathbf{q} \quad (28)$$

$$\geq \alpha a^2 (1 - \rho) \|\Delta \mathbf{q}\|^2, \quad (29)$$

where a^2 is the lowest eigenvalue of $A^\top A$. Using this expression and a similar reasoning as the one in section 3.2, the condition

$$(\alpha a^2 (1 - \rho) - 2L) \|\Delta \mathbf{q}\|^2 + \lambda_{\min}(B) \|\Delta \boldsymbol{\varphi}\|^2 > 0.$$

is found to be a sufficient condition for descent on L^a . Again, one concludes that choosing a large positive preconditioner B and a large enough penalty parameter α to meet the condition

$$\alpha > \frac{2L}{a^2(1 - \rho)} \quad (30)$$

will lead to decrease in L^a for one-shot iterations and therefore ultimately to a solution of the KKT-system (7).

3.4. A simple line search procedure for globalization

One can now use the augmented Lagrangian formulation (24) as a merit function to globalize the one-shot optimization procedure. This will then guarantee that a choice of initial design point far away from an optimum will not impede one-shot convergence. For this purpose, we introduce a simple line search algorithm. A design step $\Delta \boldsymbol{\varphi}$ will only be accepted if it obeys the Armijo condition

$$L^a(\boldsymbol{\varphi} + \Delta \boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) < L^a(\boldsymbol{\varphi}, \mathbf{q}, \mathbf{y}^*) + \gamma \Delta \boldsymbol{\varphi}^\top \nabla_{\boldsymbol{\varphi}} L^a, \quad (31)$$

with $\gamma \in]0, 1[$. Otherwise, the step $\Delta \boldsymbol{\varphi}$ is reduced with a constant fraction (a procedure commonly known as backtracking) until the Armijo condition is satisfied. In this perspective, the augmented Lagrangian formulation L^a defined in (24) has clear advantages, since during line search, the adjoint update does not have to be evaluated. In contrast, the original approach of Hamdi and Griewank requires also calculating the adjoint update for the second penalty term. Further in the paper, we will consider the application of a continuous adjoint method based on the Lagrangian $\mathcal{L}$ of equation (40).

If we assume that the Lagrangian $\mathcal{L}_G$ in the augmented Lagrangian L^a can be substituted by this other form of the Lagrangian, the line search cost can be further reduced. Then, only the residuals of the state equations are needed in each line search iteration, when using the state penalty in residual form. Therefore, one avoids the solution of the system $\Delta \mathbf{q} = A^{-1} \mathbf{c}$ as well.

3.5. Hessian estimation

The choice of a sufficiently large design preconditioner B is clearly essential for the convergence of the one-shot iterations. On the other hand, superlinear convergence is desired, while additional efforts to calculate B should be minimal. The BFGS approach is therefore a good candidate. However, as concluded in section 3.1, the choice $B \approx \partial_{\varphi\varphi} L_G^a$ requires an additional design sensitivity evaluation. Therefore, we reconsider the approximation.

To find the best design preconditioner B , the design step

$$\Delta \varphi = -B^{-1} \nabla_{\varphi} L^a(\varphi, \mathbf{q}, \mathbf{y}^*) \quad (32)$$

is sought that maximizes the descent on a quadratic approximation of L^a . This $\Delta \varphi$ solves the minimization problem

$$\min_{\Delta \varphi} \mathbf{s}^{\top} \nabla L^a(\varphi, \mathbf{q}, \mathbf{y}^*) + \frac{1}{2} \mathbf{s}^{\top} \nabla^2 L^a(\varphi, \mathbf{q}, \mathbf{y}^*) \mathbf{s}. \quad (33)$$

Keeping only the terms that depend on $\Delta \varphi$, the minimization problem reduces to

$$\min_{\Delta \varphi} E(\Delta \varphi),$$

with

$$E(\Delta \varphi) = \frac{1}{2} \Delta \varphi^{\top} \nabla_{\varphi\varphi} L^a \Delta \varphi + \Delta \varphi^{\top} (\nabla_{\varphi} L^a + \nabla_{\varphi\mathbf{q}} L^a \Delta \mathbf{q} + \nabla_{\varphi\mathbf{y}^*} L^a \Delta \mathbf{y}^*).$$

To find the minimum, we impose that the derivative of E with respect to $\Delta \varphi$ equals zero:

$$\nabla_{\varphi} L^a(\varphi, \mathbf{q}, \mathbf{y}^*) + \nabla_{\varphi\varphi} L^a \Delta \varphi + \nabla_{\varphi\mathbf{q}} L^a \Delta \mathbf{q} + \nabla_{\varphi\mathbf{y}^*} L^a \Delta \mathbf{y}^* = 0.$$

Eliminating the design gradient $\nabla_{\varphi} L^a$ using the design step relation (32) then leads to

$$B \Delta \varphi = \nabla_{\varphi\varphi} L^a \Delta \varphi + \nabla_{\varphi\mathbf{q}} L^a \Delta \mathbf{q} + \nabla_{\varphi\mathbf{y}^*} L^a \Delta \mathbf{y}^*.$$

From this equality, we may now derive a secant equation for Hessian updates:

$$B\Delta\varphi \approx \nabla_{\varphi} L^a(\varphi + \Delta\varphi, \mathbf{q} + \Delta\mathbf{q}, \mathbf{y}^* + \Delta\mathbf{y}^*) - \nabla_{\varphi} L^a(\varphi, \mathbf{q}, \mathbf{y}^*). \quad (34)$$

During one-shot optimization, we therefore use the BFGS update formula,

$$B_{k+1} = B_k - \frac{B_k \mathbf{s}_k \mathbf{s}_k^{\top} B_k}{\mathbf{s}_k^{\top} B_k \mathbf{s}_k} + \frac{\mathbf{y}_k \mathbf{y}_k^{\top}}{\mathbf{y}_k^{\top} \mathbf{s}_k},$$

with $\mathbf{s}_k = \varphi_{k+1} - \varphi_k$ and

$$\mathbf{y}_k = \nabla_{\varphi} L^a(\varphi_{k+1}, \mathbf{q}_{k+1}, \mathbf{y}_{k+1}^*) - \nabla_{\varphi} L^a(\varphi_k, \mathbf{q}_k, \mathbf{y}_k^*),$$

to update the design preconditioner B . If the convergence of state or adjoint residuals is noisy, one might decide to use other secant equations than equation (34), such as the secant equation (18). As argued by Hamdi and Griewank [10], these approximations becomes valid at primal and dual feasibility. Notice that this issue therefore only arises in one-shot approaches.

To safeguard the positive definiteness of the Hessian estimate B , Powell's damped update rule can be used. This update rule relaxes the Hessian update to guarantee that the directional curvature can maximally be decreased by a fraction γ_{min} [21]. The damped BFGS approach thus consists of applying the following rule: If $\mathbf{s}_k^{\top} \mathbf{y}_k < \gamma_{min} \mathbf{s}_k^{\top} B_k \mathbf{s}_k$, substitute $\mathbf{y}_k$ by $\tilde{\mathbf{y}}_k = \theta \mathbf{y}_k + (1 - \theta) B_k \mathbf{s}_k$, such that $\mathbf{s}_k^{\top} \tilde{\mathbf{y}}_k = \gamma_{min} \mathbf{s}_k^{\top} B_k \mathbf{s}_k$. This is achieved when

$$\theta = \frac{(1 - \gamma_{min}) \mathbf{s}_k^{\top} B_k \mathbf{s}_k}{\mathbf{s}_k^{\top} B_k \mathbf{s}_k - \mathbf{s}_k^{\top} \mathbf{y}_k}. \quad (35)$$

To make the Hessian estimation more robust, absolute bounds can be added on the curvature as in the simplified approach of Hazra et al. [2]. Indeed, one needs to avoid that the Hessian definiteness is decreased up to machine accuracy in successive iterations, resulting in unrealistically large design steps. Additionally, a safety margin should be kept, since Powell's trick is a rank-two update and only the definiteness of the Hessian in the step direction is monitored. Similarly, an unbounded increase of the Hessian causes numerical problems as well. Too small design steps will cause the gradient differences $\mathbf{y}_k$ to be in the error margin of the gradients, which might lead to a further misleading increase of the Hessian estimation. Especially when using (in parts) finite difference sensitivity calculations, like the approach introduced in section 4.3, one can easily see that the design

steps should be an order of magnitude bigger than the finite difference steps. For these reasons, (35) is used to bound the relative decrease as well as the relative increase. The latter is then done when $\mathbf{s}_k^\top \mathbf{y}_k > \gamma_{max} \mathbf{s}_k^\top B_k \mathbf{s}_k$.

Absolute bounds β_{min} and β_{max} on the directional curvature can be imposed as well by modifying Powell's trick as follows. The directional curvature of each step should be adjusted so that

$$\beta_{min} < \frac{\mathbf{s}_k^\top \mathbf{y}_k}{\mathbf{s}_k^\top \mathbf{s}_k} < \beta_{max}. \quad (36)$$

If any of the two above constraints is violated $\tilde{\mathbf{y}}_k$ can be calculated from

$$\mathbf{s}_k^\top \tilde{\mathbf{y}}_k = \beta \mathbf{s}_k^\top \mathbf{s}_k, \quad (37)$$

where β represents either of the two bounds β_{min} or β_{max} . Solving (37) for θ leads to

$$\theta = \frac{\mathbf{s}_k^\top (B_k - \beta \mathbb{I}) \mathbf{s}_k}{\mathbf{s}_k^\top B_k \mathbf{s}_k - \mathbf{s}_k^\top \mathbf{y}_k}. \quad (38)$$

Finally, it should be noticed that in one-shot optimization a lot more updates are done to the Hessian than in regular optimization. It is therefore logical to relax the Hessian updates as well. This is achieved when θ is substituted by $\tilde{\theta} = \mu\theta$, with $\mu < 1$ the relaxation factor.

4. Application to optimal magnetic divertor design

With the novel one-shot methodology developed in section 3 at hand, application to magnetic divertor design can now be considered. In section 4.1, we reformulate this design problem in the numerical optimization framework (1). Next, in section 4.2, the model equations are given. Finally, in section 4.3, the design sensitivity calculation is outlined.

4.1. Design controls and objective

The design problem under consideration is the optimization of the plasma-wall interaction at the vacuum vessel of a nuclear fusion tokamak. More specifically, it will be attempted to mitigate excessive heat loads on the so-called divertor target components that cause target erosion. As in past design studies [7, 8, 13], the control variables are the divertor coil currents that directly alter the local magnetic configuration and divert the plasma flow towards for the purpose designed divertor targets (red vessel part in

figure 1). At these target surfaces, specifically designed tiles are foreseen to withstand heat loads up to a maximum of 10 MWm^{-2} . Nevertheless, for economically competitive fusion reactors, it will be necessary to increase the heat exhausted on the target plates, while respecting the material limits of these target tiles [22]. Therefore, it will be attempted to equally distribute the heat load $Q_{\perp}$ on the reactor's target components. The chosen objective function is [23, 7]

$$\mathcal{I}(\boldsymbol{\varphi}, \mathbf{q}) = \frac{1}{2} \lambda_Q \int_{S_t} (Q_{\perp} - Q_{d,t})^2 d\boldsymbol{\sigma} + \frac{1}{2} \lambda_{\varphi} \boldsymbol{\varphi}^{\top} \boldsymbol{\varphi}, \quad (39)$$

with $\boldsymbol{\varphi}$ the dimensionless coil currents, and where the surface integral with elementary surface vector $d\boldsymbol{\sigma}$ integrates over the target surface area S_t . In figure 1, the calculation of the objective functional is illustrated. As can be seen in (39), the objective functional is small when the heat flux density perpendicular to the target surface $Q_{\perp}$ closely approximates a desirable spatially constant heat flux profile $Q_{d,t}$. The second term is a Tikhonov regularization term proportional to the Joule losses in the divertor coils.

4.2. Model

For a more detailed description of the model the interested reader is referred to [7]. Here, we only give a brief overview of the essential characteristics. In short, the model aims at including the relevant physical processes, while keeping simulation times low enough for quick optimization trials. It consists of a reduced magnetic field and plasma edge transport model, which are solved sequentially for the magnetic flux ψ and the plasma edge state variables $\mathbf{q}_{\text{pe}}$, respectively.

Since determining the magnetic field in tokamaks is a strongly nonlinear process that depends on the plasma currents itself, different strategies are possible. For small changes in $\boldsymbol{\varphi}$, one might argue that the change of plasma current can be neglected. This gives rise to a perturbation model that evaluates the magnetic flux change as $\psi = \psi_0 + \delta\psi$, where $\delta\psi$ is calculated as if the coil were in vacuum (the so-called vacuum approximation). Although recent research revealed that this method only gives an accurate approximation of the magnetic field directly around the coils and a free boundary magnetic equilibrium solver might be preferable [13], we will opt for the simplicity of the perturbation model, since the main objective of this paper is the verification of the one-shot optimization approach.

Figure 1: An illustration of the divertor region in a tokamak and the heat exhaust on the divertor target components (red part of vessel). The desired heat flux profile $Q_{d,t}$ is shown in brown. Most of the heat is diverted from the core along the so-called separatrix, the first magnetic flux surface that is not closed within the vessel, and deposited on the target area (the plasma flow approximately follows the pink arrows). This leads to the peaked heat load profile near this magnetic surface.

The plasma edge model features Navier-Stokes like equations that solve for the ion density n and the ion velocity parallel to the magnetic field $u_{\parallel}$ from continuity and momentum conservation equations. A pressure diffusion equation governs the neutral pressure p_n . An internal energy equation is solved for a combined ion-electron-neutral temperature T . Together, they constitute the plasma state vector $\mathbf{q}_{pe} = [n, u_{\parallel}, T, p_n]^{\top}$.

Because of the strong anisotropy in plasma flows along and orthogonal to the magnetic field, these equations are typically solved in a coordinate system aligned to the magnetic field. The plasma edge transport equations $\mathbf{c}_{pe}(\mathbf{q}_{ct}, \mathbf{q}_{pe}) = 0$ therefore depend on the metric coefficients $\mathbf{q}_{ct}$ that govern the coordinate transformation from Cartesian to field-aligned coordinates. $\mathbf{q}_{ct}$ can be evaluated on a discrete level as $\mathbf{q}_{ct}(\psi(\varphi))$ by using sequentially the magnetic perturbation calculation $\psi(\varphi)$ and an evaluation of the discrete metric coefficients $\mathbf{q}_{ct}(\psi)$ by a field-aligned grid generator. The plasma edge transport equations $\mathbf{c}_{pe}(\mathbf{q}_{ct}, \mathbf{q}_{pe}) = 0$ can then be solved for $\mathbf{q}_{pe}(\mathbf{q}_{ct})$ using a slowly converging false time stepping procedure.

4.3. An in parts adjoint strategy to sensitivity calculation

To efficiently calculate the sensitivities of the objective functional (39) with respect to the control variables φ , an in parts adjoint approach is employed [9]. In this section we briefly explain this approach. In short, an iterative procedure that solves the continuous adjoint (optimize-then-discretize approach) of the plasma edge equations, as presented in [5], forms the backbone of the sensitivity calculation. The main advantage is that large parts of the numerical routines of the state solver can be reused. Especially in nested optimization algorithms, the relatively low computational effort of magnetic field calculation and grid generation is not worthwhile the effort of deriving and implementing their adjoint equations. This part is therefore treated with a finite difference sensitivity calculation.

Formally, we can represent the entire model by the partially reduced model equations $\tilde{\mathbf{c}}(\varphi, \mathbf{q}_{\text{pe}}) := \mathbf{c}_{\text{pe}}(\mathbf{q}_{\text{ct}}(\psi(\varphi)), \mathbf{q}_{\text{pe}}) = 0$. In a similar way we define the partially reduced cost functional $\tilde{\mathcal{I}}(\varphi, \mathbf{q}_{\text{pe}}) := \mathcal{I}(\varphi, \mathbf{q}_{\text{ct}}(\psi(\varphi)), \mathbf{q}_{\text{pe}})$ ⁴. We will assume both $\tilde{\mathcal{I}}(\varphi, \mathbf{q}_{\text{pe}})$ and $\tilde{\mathbf{c}}(\varphi, \mathbf{q}_{\text{pe}})$ are continuously Fréchet-differentiable. The KKT-conditions can then be derived by taking the derivatives of the partially reduced Lagrangian

$$\tilde{\mathcal{L}}(\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*) = \tilde{\mathcal{I}}(\varphi, \mathbf{q}_{\text{pe}}) + \langle \mathbf{q}_{\text{pe}}^*, \tilde{\mathbf{c}}(\varphi, \mathbf{q}_{\text{pe}}) \rangle \quad (40)$$

and read

$$\begin{cases} \nabla_{\mathbf{q}_{\text{pe}}^*} \tilde{\mathcal{L}}(\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*) = \tilde{\mathbf{c}}(\varphi, \mathbf{q}_{\text{pe}}) = 0 \\ \nabla_{\mathbf{q}_{\text{pe}}} \tilde{\mathcal{L}}(\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*) = \nabla_{\mathbf{q}_{\text{pe}}} \tilde{\mathcal{I}}(\varphi, \mathbf{q}_{\text{pe}}) + (\partial_{\mathbf{q}_{\text{pe}}} \tilde{\mathbf{c}}(\varphi, \mathbf{q}_{\text{pe}}))^* \mathbf{q}_{\text{pe}}^* = 0 \\ \langle \nabla_{\varphi} \tilde{\mathcal{L}}(\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*), \delta \varphi \rangle = \langle \nabla_{\varphi} \tilde{\mathcal{I}}(\varphi, \mathbf{q}_{\text{pe}}), \delta \varphi \rangle + \langle \mathbf{q}_{\text{pe}}^*, \partial_{\varphi} \tilde{\mathbf{c}}(\varphi, \mathbf{q}_{\text{pe}}) \delta \varphi \rangle = 0 \quad \forall \delta \varphi \end{cases}, \quad (41)$$

where we denote the adjoint operator with respect to the scalar product $\langle \cdot, \cdot \rangle$ as $*$. Remark that the design equation is deliberately written in variational form, since we are not willing to derive the adjoint of magnetic field calculation and grid generation $\partial_{\varphi} \tilde{\mathbf{c}}(\varphi, \mathbf{q}_{\text{pe}})^*$, as mentioned before. In practice, we

⁴Notice that the state vector in (39) is defined as $\mathbf{q} = \{\mathbf{q}_{\text{ct}}, \mathbf{q}_{\text{pe}}\}$.

therefore evaluate the design sensitivity in a direction $\delta\varphi$ as

$$\begin{aligned} \langle \Delta_\varphi^\varepsilon \tilde{\mathcal{L}}(\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*), \delta\varphi \rangle &= \frac{\tilde{\mathcal{L}}(\varphi + \varepsilon\delta\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*) - \tilde{\mathcal{L}}(\varphi - \varepsilon\delta\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*)}{2\varepsilon} \\ &= \frac{\tilde{\mathcal{I}}(\varphi + \varepsilon\delta\varphi, \mathbf{q}_{\text{pe}}) - \tilde{\mathcal{I}}(\varphi - \varepsilon\delta\varphi, \mathbf{q}_{\text{pe}})}{2\varepsilon} + \langle \mathbf{q}_{\text{pe}}^*, \frac{\tilde{\mathbf{c}}(\varphi + \varepsilon\delta\varphi, \mathbf{q}_{\text{pe}}) - \tilde{\mathbf{c}}(\varphi - \varepsilon\delta\varphi, \mathbf{q}_{\text{pe}})}{2\varepsilon} \rangle, \end{aligned} \quad (42)$$

where $\Delta_\varphi^\varepsilon$ denotes a finite difference gradient evaluation at constant $\mathbf{q}$ and $\mathbf{q}^*$ with step size ε . The entire gradient $\Delta_\varphi^\varepsilon \tilde{\mathcal{L}}$ then needs $2 \times n_\varphi$ evaluations of $\mathbf{q}_{\text{ct}}(\psi(\varphi))$, with n_φ the number of control variables.

Although after discretizing the continuous adjoint equations significant inconsistency can be introduced between sensitivity and discrete model, it is found in [9] that the accuracy for the heat load sensitivities is sufficiently good for practical optimization purposes at standard grid sizes. We therefore assume that the discrete one-shot theory of section 3 remains valid.

For this one-shot strategy, we will search in each design step for sufficient descent on the partially reduced augmented Lagrangian

$$\tilde{L}^a(\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*) = \frac{\alpha}{2} \sum_{i=1}^{n_E} \left\| \frac{\tilde{\mathbf{c}}_i(\varphi, \mathbf{q}_{\text{pe}})}{s_{c,i}(\varphi, \mathbf{q}_{\text{pe}})} \right\|^2 + \tilde{\mathcal{L}}(\varphi, \mathbf{q}_{\text{pe}}, \mathbf{q}_{\text{pe}}^*). \quad (43)$$

We thus need to accurately evaluate the design gradient $\nabla_\varphi \tilde{L}^a$. The finite difference approach can be used to calculate the sensitivity of this partially reduced augmented Lagrangian with respect to the design variables as $\nabla_\varphi \tilde{L}^a \approx \Delta_\varphi^\varepsilon \tilde{L}^a$. It is obvious that especially the sensitivity of the squared terms in $\tilde{L}^a$ will suffer from poor accuracy. Indeed, since only a finite amount of significant digits are used, the squaring operation will cause a loss of accuracy and will restrict the user to use rather large values for the finite difference perturbation ε . Therefore, we opt to evaluate these squared terms using a semi-analytical approach. E.g., for the feasibility penalty this gives

$$\nabla_\varphi \left(\frac{1}{2} \sum_{i=1}^{n_E} \left\| \frac{\tilde{\mathbf{c}}_i}{s_{c,i}} \right\|^2 \right) \approx \sum_{i=1}^{n_E} \frac{\tilde{\mathbf{c}}_i^\top \Delta_\varphi^\varepsilon \tilde{\mathbf{c}}_i}{s_{c,i}^2} - \sum_{i=1}^{n_E} \left\| \frac{\tilde{\mathbf{c}}_i}{s_{c,i}} \right\|^2 \frac{\Delta_\varphi^\varepsilon s_{c,i}}{s_{c,i}}. \quad (44)$$

The finite difference evaluations $\Delta_\varphi^\varepsilon \tilde{\mathbf{c}}_i$ and $\Delta_\varphi^\varepsilon s_{c,i}$ are then simply substituted in this analytical formula to come to an approximation of each such term in the derivative $\nabla_\varphi \tilde{\mathcal{L}}$. In addition to the feasibility penalty term, this trick

is used for the two squaring operations in the objective function (39). It may be noticed that the scaling variables $s_{c,i}(\boldsymbol{\varphi}, \mathbf{q}_{\text{pe}})$ that make the residuals dimensionless depend indirectly on the control variables through the coordinate transformation $\mathbf{q}_{\text{ct}}(\psi(\boldsymbol{\varphi}))$. This leads to an additional term in this expression. A similar method is applied to the squared objective function terms in (39). Given that the sensitivity calculation is based on a finite difference procedure, we will limit the minimal design step to 10 times the finite difference step size ε . Smaller steps could cause inconsistencies in line search or very inaccurate Hessian estimations.

It is clear that in one-shot optimization, the increased number of sensitivity calculations will create a need for faster sensitivity calculations. The above argument that the influence of magnetic field calculation and mesh generation on the total computational cost is relatively low, might therefore no longer hold. Although we will keep the in parts adjoint approach for now, we aim at reducing the CPU-time associated to computing grid derivatives and design updates by evaluating $\mathbf{q}_{\text{ct}}(\psi)$ using a grid deformation algorithm instead of a full mesh generation step. For this purpose, a grid deformation algorithm based on the spring analogy [24, 25] has been adapted to match the specific grid requirements for plasma edge simulations.

This grid deformation procedure gives significant speed-up with respect to full mesh generation, though larger deformations might lead to qualitatively inferior grids. In the optimization procedure, qualitatively inferior grids are easily avoided by creating a new grid after a fixed number of (one-shot) optimization iterations or when the design change since the last mesh generation event exceeds a limit, that is $\|\boldsymbol{\varphi} - \boldsymbol{\varphi}_{\text{G}}\| > \zeta$, with $\boldsymbol{\varphi}_{\text{G}}$ the control variables associated to the last constructed grid with node locations $\mathbf{x}_{\text{G}}$ and ζ a fixed code parameter.

5. Test case

The objective of this test case is to attempt the novel one-shot optimization method on a typical magnetic field design case without additional design constraints. In this section, we first specify the set-up that was used to test this in section 5.1. Subsequently, we summarize the steps in the applied one-shot algorithm in section 5.2. Afterwards, the results are discussed in section 5.3.

5.1. Set-up

For the set-up of the test case, we base ourselves on a test case presented in [13]. This paper describes a preliminary optimization study for application of optimal magnetic divertor design to the WEST tokamak [26, 27], of which a poloidal cut is shown in figure 2. We aim to minimize the cost functional (39) by designing the currents in the lower divertor coils (coils 14-17 in figure 2), starting from the coil currents summarized in table 1. For the parameters

Conductor	4	5	6	7	10	11	12	13	14	15	16	17
I (kA)	22.3	-186.4	-186.4	-22.3	41.9	41.9	25.3	-11.0	51.2	51.2	51.2	51.2

Table 1: Values of the coil currents before optimization (non-powered coils have been left out). Conductor numbers corresponding to the numbers in figure 2.

Figure 2: A poloidal cut of the WEST tokamak. Coils are indicated in orange and numbered. The controlled coils are the lower divertor coils (numbers 14-17).

of magnetic field and plasma edge model, the interested reader is referred

to [9]. The reference equilibrium ψ_0 of the magnetic perturbation model is the one presented in that paper and plasma model parameters and boundary conditions are identical to those given there. The plasma edge transport equations are discretized using a finite volume method on a structured grid of size 210×80 . This grid is finer than typical grids for single plasma edge simulations, i.e. without optimization. This is necessary as the isotropic neutral diffusion equation is discretized on the same grid as the strongly anisotropic plasma flow. The huge computational cost of these plasma edge simulations prevents the use of much finer grids.

In previous nested optimization studies of the WEST reactor [13], inequality constraints were included to account for current source limitations and restrictions on the magnetic field layout. Since we desire an unconstrained test case for this first one-shot trial, these design constraints $\mathbf{h}(\boldsymbol{\varphi}) \leq 0$ cannot be imposed directly. However, it is crucial that they are imposed since some unphysical results may lead to code crashes. Therefore, the cost functional is modified with a penalty of augmented Lagrangian type. That is, we replace the cost functional (39) by

$$\tilde{\mathcal{I}}^A(\boldsymbol{\varphi}, \mathbf{q}_{\text{pe}}) = \tilde{\mathcal{I}}(\boldsymbol{\varphi}, \mathbf{q}_{\text{pe}}) + \mu \mathbf{h}_{\mathbb{A}} + \eta \|\mathbf{h}_{\mathbb{A}}\|^2, \quad (45)$$

with $\mathbf{h}_{\mathbb{A}}$ the active constraints and η and μ weighting parameters of the penalty terms. If μ approaches the Lagrangian multipliers of the constrained optimization problem and if η is chosen sufficiently big, this is an exact penalty function, where the optimum of the original constrained optimization problem is found to be a stable optimum of the unconstrained penalty function (45) [28]. From a previous nested optimization study, we find that the maximal current limit on coil 14 is the only active constraint. Based on the same study we find $\mu = 0.9$ and we take $\eta = 10$.

After using piggybacking to partially converge state and adjoint equations, the one-shot optimization procedure is used to find a solution to the KKT system (41). As outlined in section 4.3, we search for descent on the augmented Lagrangian (43). Finding a suitable approach to estimate the parameters L and a^2 in the sufficient condition for the penalty parameter α (equation (30)) remains a challenge for future work. Furthermore, the condition itself might be too restrictive. For these reasons, the feasibility penalty parameter α is not chosen based on this sufficient condition. Rather, α was chosen experimentally by trying several values for a limited amount of iterations. The resulting $\alpha = 10^{-8}$ appeared to give fast, yet steady convergence.

It should also be noted that the Hessian updates for the selected results follow the methodology as discussed in section 3.5 but are based on the secant rule

$$B\Delta\varphi \approx \nabla_{\varphi}\tilde{L}^a(\varphi + \Delta\varphi, \mathbf{q} + \Delta\mathbf{q}, \mathbf{q}^*) - \nabla_{\varphi}\tilde{L}^a(\varphi, \mathbf{q}, \mathbf{q}^*).$$

That is, in comparison to the secant rule that followed from theory, the variable $\mathbf{q}^*$ is kept constant. Both approximations have been tested but the choice was not observed to be of great importance for these cases. It should be mentioned finally that although the derivations of section 3 are made for a Jacobi type iteration of state, adjoint, and design equation, we perform Gauss-Seidel type iterations since they most often converge much faster [14, 3].

5.2. Overview of the applied one-shot algorithm

To summarize, the applied one-shot algorithm then completes the following actions in each iteration of the optimization algorithm.

1. Perform one iteration on the state solver of the plasma edge equations $\tilde{\mathbf{c}}(\varphi, \mathbf{q}_{\text{pe}}) = 0$. For the current model, this requires calculating sequentially the poloidal magnetic flux ψ , the metric coefficients of the deformed grid $\mathbf{q}_{\text{ct}}$, and performing a false time step of the plasma edge solver for the updated state variables $\mathbf{q}_{\text{pe},k+1}$.
2. Perform one false time step in the segregated solver of the continuous adjoint plasma edge equations to obtain the updated adjoint variables $\mathbf{q}_{\text{pe},k+1}^*$.
3. Evaluate the gradient of the augmented Lagrangian. In the current model, this is done by evaluating the gradient $\nabla_{\varphi}\tilde{L}^a \approx \Delta_{\varphi}^{\varepsilon}\tilde{L}^a$ of the augmented Lagrangian $\tilde{L}^a$ (equation (43)) with finite differences. To increase the accuracy of the finite difference evaluations, the semi-analytical approach illustrated in equation (44) is used.
4. Calculate the design preconditioner B from the robust BFGS strategy described in section 3.5 and calculate a search direction from

$$\Delta\varphi_k = -B^{-1}\nabla_{\varphi}\tilde{L}^a.$$

5. Reduce the design step $\Delta\varphi_k$ in a backtracking line search until the Armijo condition

$$\tilde{L}^a(\varphi_k + \Delta\varphi_k, \mathbf{q}_{\text{pe},k}, \mathbf{q}_{\text{pe},k}^*) < \tilde{L}^a(\varphi_k, \mathbf{q}_{\text{pe},k}, \mathbf{q}_{\text{pe},k}^*) + \gamma\Delta\varphi_k^{\top}\nabla_{\varphi}\tilde{L}^a,$$

is met and evaluate the updated design variables $\varphi_{k+1} = \varphi_k + \Delta\varphi_k$.

These steps are then repeated until design, state and adjoint variables are converged to the solution of the KKT conditions (equation (41)). Remark that this algorithm has been tailored to meet the requirements of the current application and the existing continuous adjoint framework. Different variants of the proposed one-shot algorithm are possible that may use both continuous or discrete adjoint (AD) sensitivity calculations to suit a specific application.

5.3. Results

We first observe the residuals in figure 3. It can clearly be seen in the initial piggy-backing stage that the state and adjoint equations converge at the same rate, indicating that the continuous adjoint equations are discretized somewhat consistently with respect to the state equations. After starting the optimization procedure, apparently noisy residuals changes are found. Here, we should mention that many of the often regularly spaced peaks in state and adjoint residuals are associated to the strategy of re-meshing after a certain amount of grid deformation steps. As can be seen, this is an instantaneous effect that disappears rather quickly. However, this does not explain the noisy design residuals, given by the norm of the gradient. The latter is caused by gradient discontinuities, arising from discrete grid generator switches and piecewise linear interpolations. It is exactly this irregularity that made standard one-shot strategies insufficient and urged the need for the globalized algorithm.

Indeed, looking at the augmented Lagrangian function value throughout optimization in figure 4, one may see that it decreases monotonically, despite the gradient discontinuities. That is, disregarding the small peaks caused by the re-meshing mentioned earlier and entering through the state feasibility penalty. One might observe here that the increases in state and adjoint residuals in figure 3 (e.g. around iteration 0 and 40000) are compensated by decreases in objective functional. Furthermore, one might observe that cost functional and augmented Lagrangian functional differ significantly when the state and adjoint residuals are high. This is because the second term of the Lagrangian (40) is quite large here. One may interpret this term as a linearized estimation of the effect that converging the state equation will have on the objective functional. Finally, it should be noted that further converging state and adjoint equations after optimization using piggy-backing does not greatly change the cost functional, indicating that the one-shot procedure was sufficiently converged.

Figure 3: Convergence of the one-shot optimization procedure. State residuals are indicated with a solid blue line, adjoint residuals with a dashed red line, and design residuals with green dots. Piggy-backing iterations are used for feasibility convergence before (I) and after (III) one-shot optimization.

Remark that full convergence of the one-shot optimization is not possible, since a continuous adjoint approach is used. It should be noted that finite difference step size studies and systematical grid refinement studies for several cases show that the discretization error is typically much larger than the truncation errors from finite difference calculations in plasma edge calculations [9]. Therefore the main cause for lack of convergence is expected to stem from the discretization error. This is inherently associated with the continuous adjoint approach. Also convergence of the state residuals in the one-shot iterations is not possible because of the presence of the minimally allowed design step, introduced in section 4.3.

6. Conclusions and perspectives

We propose a practical one-shot optimization method that features a line

Figure 4: The augmented Lagrangian function $\tilde{L}^a$ (solid blue line) and the cost functional (dashed red line) during one-shot and feasibility convergence. The roman numbers indicate the different regimes and correspond to the numbers in figure 3. The numbers I and III indicate piggy-backing before and after one-shot optimization (II), respectively.

search procedure with an augmented Lagrangian merit function. Under the assumption that the ratio of dual and primal updates is bounded, the method is proven to converge for an appropriate parameter choice. In comparison to past globalized one-shot methods, it avoids the need of second order adjoint derivatives and reduces the line search costs. Furthermore, we have adapted damped BFGS updates to robustly estimate the Hessian during one-shot optimization, while keeping the Hessian sufficiently large for convergence.

Furthermore, we have shown how this one-shot optimization procedure can be applied to optimal magnetic configuration design in nuclear fusion tokamaks. For this purpose, a strategy was elaborated that calculates in parts continuous adjoint sensitivities of a specific augmented Lagrangian function. Hereby, a grid deformation method was used to reduce the computational

cost of design updates and sensitivity calculations, a requirement for efficient one-shot optimization. Finally, a test case has been set-up to test the novel one-shot approach on magnetic divertor design. The one-shot approach successfully converges this test case, notwithstanding the sporadic occurrence of gradient discontinuities.

In future research, we aim at solving the constrained magnetic design problem. Therefore, an elegant method will be pursued to embed additional design constraints in the one-shot optimization procedure. Furthermore, an assessment of the computational cost in comparison to nested optimization methods is desirable. To fully reveal the potential of the one-shot method, this would require full adjoint sensitivity calculations, including the adjoint of the novel grid deformation method and the magnetic field calculation. Additionally, the simple backtracking strategy may be replaced by a state-of-the-art method.

- [1] E. Papoutsis-Kiachagias and K. Giannakoglou, “Continuous Adjoint Methods for Turbulent Flows, Applied to Shape and Topology Optimization: Industrial Applications,” *Archives of Computational Methods in Engineering*, pp. 1–45, 2014.
- [2] S. Hazra, V. Schulz, J. Brezillon, and N. Gauger, “Aerodynamic shape optimization using simultaneous pseudo-timestepping,” *Journal of Computational Physics*, vol. 204, pp. 46–64, Mar. 2005.
- [3] T. Bosse, L. Lehmann, and A. Griewank, “Adaptive sequencing of primal, dual, and design steps in simulation based optimization,” *Computational Optimization and Applications*, vol. 57, no. 3, pp. 731–760, 2014.
- [4] W. Dekeyser, D. Reiter, and M. Baelmans, “Automated divertor target design by adjoint shape sensitivity analysis and a one-shot method,” *Journal of Computational Physics*, vol. 278, pp. 117–132, Dec. 2014.
- [5] W. Dekeyser, D. Reiter, and M. Baelmans, “A one shot method for divertor target shape optimization,” *PAMM*, vol. 14, no. 1, pp. 1017–1022, 2014.
- [6] W. Dekeyser, D. Reiter, and M. Baelmans, “Designing divertor targets for uniform power load,” *Journal of Nuclear Materials*, vol. 463, pp. 1243–1247, Aug. 2015.

- [7] M. Blommaert, W. Dekeyser, M. Baelmans, N. Gauger, and D. Reiter, “An Automated Approach to Magnetic Divertor Configuration Design,” *Nuclear Fusion*, vol. 55, no. 013001, 2015.
- [8] M. Blommaert, M. Baelmans, W. Dekeyser, N. Gauger, and D. Reiter, “A novel approach to magnetic divertor configuration design,” *Journal of Nuclear Materials*, vol. 463, pp. 1220–1224, Aug. 2015.
- [9] M. Blommaert, H. Heumann, M. Baelmans, N. R. Gauger, and D. Reiter, “Towards Automated Magnetic Divertor Design for Optimal Heat Exhaust,” *ESAIM: Proceedings and Surveys*, 2015. in press.
- [10] A. Hamdi and A. Griewank, “Reduced quasi-Newton method for simultaneous design and optimization,” *Computational Optimization and Applications*, vol. 49, no. 3, pp. 521–548, 2011.
- [11] A. Hamdi and A. Griewank, “Properties of an Augmented Lagrangian for Design Optimization,” *Optimization Methods Software*, vol. 25, pp. 645–664, Aug. 2010.
- [12] L. Kaland, J. C. De Los Reyes, and N. R. Gauger, “One-shot methods in function space for PDE-constrained optimal control problems,” *Optimization Methods and Software*, vol. 29, pp. 376–405, Feb. 2013.
- [13] M. Blommaert, M. Baelmans, H. Heumann, Y. Marandet, H. Bufferand, N. R. Gauger, and D. Reiter, “Magnetic Field Models and their Application in Optimal Magnetic Divertor Design,” *Contributions to Plasma Physics*, 2016. in press.
- [14] A. Griewank, “Projected Hessians for Preconditioning in One-Step One-Shot Design Optimization,” in *Nonconvex Optimization and Its Applications* (G. Di Pillo and M. Roma, eds.), vol. 83, pp. 151–171, Springer US, 2006.
- [15] G. Di Pillo and L. Grippo, “A continuously differentiable exact penalty function for nonlinear programming problems with inequality constraints,” *SIAM Journal on Control and Optimization*, vol. 23, no. 1, pp. 72–84, 1985.
- [16] G. Di Pillo, “Exact penalty methods,” in *Algorithms for Continuous Optimization*, pp. 209–253, Springer, 1994.

- [17] L. Dixon, *Exact penalty function methods in nonlinear programming*. Numerical Optimisation Centre, Hatfield Polytechnic, 1979.
- [18] R. Fontecilla, T. Steihaug, and R. A. Tapia, “A convergence theory for a class of quasi-newton methods for constrained optimization,” *SIAM Journal on Numerical Analysis*, vol. 24, no. 5, pp. 1133–1151, 1987.
- [19] E. Özkaya and N. R. Gauger, “Single-step one-shot aerodynamic shape optimization,” in *International Series of Numerical Mathematics* (K. Kunisch, J. Sprekels, G. Leugering, and F. Tröltzsch, eds.), vol. 158, pp. 191–204, Birkhäuser Basel, 2009.
- [20] A. Griewank and C. Faure, “Reduced Functions, Gradients and Hessians from Fixed-Point Iterations for State Equations,” *Numerical Algorithms*, vol. 30, no. 2, pp. 113–139, 2002.
- [21] M. Powell, “A fast algorithm for nonlinearly constrained optimization calculations,” in *Lecture Notes in Mathematics* (G. Watson, ed.), vol. 630, pp. 144–157, Springer Berlin Heidelberg, 1978.
- [22] F. Romanelli, P. Barabaschi, D. Borba, G. Federici, L. Horton, R. Neu, D. Stork, and H. Zohm, “A roadmap to the realization of Fusion Energy,” tech. rep., European Fusion Development Agreement (EFDA), 2012.
- [23] W. Dekeyser, D. Reiter, and M. Baelmans, “Divertor Design through Shape Optimization,” *Contrib. Plasma Phys.*, vol. 52, pp. 544–549, June 2012.
- [24] J. T. Batina, “Unsteady Euler airfoil solutions using unstructured dynamic meshes,” *AIAA Journal*, vol. 28, no. 8, 1990.
- [25] F. J. Blom, “Considerations on the spring analogy,” *Int. J. Numer. Meth. Fluids*, vol. 32, pp. 647–668, Mar. 2000.
- [26] C. Bourdelle, J. Artaud, V. Basiuk, M. Bécoulet, S. Brémond, J. Bucalossi, H. Bufferand, G. Ciraolo, L. Colas, Y. Corre, X. Courtois, J. Decker, L. Delpech, P. Devynck, G. Dif-Pradalier, R. Dörner, D. Douai, R. Dumont, A. Ekedahl, N. Fedorczak, C. Fenzi, M. Firdaouss, J. Garcia, P. Ghendrih, C. Gil, G. Giruzzi, M. Goniche,

- C. Grisolia, A. Grosman, D. Guilhem, R. Guirlet, J. Gunn, P. Hennequin, J. Hillairet, T. Hoang, F. Imbeaux, I. Ivanova-Stanik, E. Joffrin, A. Kallenbach, J. Linke, T. Loarer, P. Lotte, P. Maget, Y. Marandet, M. Mayoral, O. Meyer, M. Missirlian, P. Mollard, P. Monier-Garbet, P. Moreau, E. Nardon, B. Pégourié, Y. Peysson, R. Sabot, F. Saint-Laurent, M. Schneider, J. Travère, E. Tsitrone, S. Vartanian, L. Vermare, M. Yoshida, R. Zagorski, and JET EFDA Contributors, “WEST Physics Basis,” *Nuclear Fusion*, vol. 55, no. 6, p. 063017, 2015.
- [27] J. Bucalossi, M. Missirlian, P. Moreau, F. Samaille, E. Tsitrone, D. van Houtte, T. Batal, C. Bourdelle, M. Chantant, Y. Corre, X. Courtois, L. Delpech, L. Doceul, D. Douai, H. Dougnac, F. Faisse, C. Fenzi, F. Ferlay, M. Firdaouss, L. Gargiulo, P. Garin, C. Gil, A. Grosman, D. Guilhem, J. Gunn, C. Hernandez, D. Keller, S. Larroque, F. Leroux, M. Lipa, P. Lotte, A. Martinez, O. Meyer, F. Micolon, P. Mollard, E. Nardon, R. Nouailletas, A. Pilia, M. Richou, S. Salasca, and J.-M. Travère, “The WEST project: Testing ITER divertor high heat flux component technology in a steady state tokamak environment,” *Fusion Engineering and Design*, vol. 89, pp. 907–912, Oct. 2014.
- [28] J. Nocedal and S. Wright, *Numerical Optimization*. Springer, 2006.