

A fast, high-order solver for the Grad–Shafranov equation



Andras Pataki ^{a,1}, Antoine J. Cerfon ^{a,2}, Jeffrey P. Freidberg ^{b,2}, Leslie Greengard ^{a,1}, Michael O’Neil ^{a,*1}

^a Courant Institute of Mathematical Sciences, New York University, New York, NY 10012, United States

^b Plasma Science and Fusion Center, Massachusetts Institute of Technology, Cambridge, MA 02139, United States

ARTICLE INFO

Article history:

Received 8 October 2012

Received in revised form 15 January 2013

Accepted 22 February 2013

Available online 13 March 2013

Keywords:

Grad–Shafranov

Plasma physics

Poisson solver

Spectrally-accurate

Conformal mapping

High-order

Kerzman–Stein

ABSTRACT

We present a new fast solver to calculate fixed-boundary plasma equilibria in toroidally axisymmetric geometries. By combining conformal mapping with Fourier and integral equation methods on the unit disk, we show that high-order accuracy can be achieved for the solution of the equilibrium equation and its first and second derivatives. Smooth arbitrary plasma cross-sections as well as arbitrary pressure and poloidal current profiles are used as initial data for the solver. Equilibria with large Shafranov shifts can be computed without difficulty. Spectral convergence is demonstrated by comparing the numerical solution with a known exact analytic solution. A fusion-relevant example of an equilibrium with a pressure pedestal is also presented.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

High performance numerical equilibrium solvers are crucial to the computational effort in plasma physics for magnetic fusion applications. Numerically computed plasma equilibria are needed as an input to the algorithms responsible for calculating the macroscopic stability and transport properties of plasma configurations. Equilibrium solvers need to be fast, so that computational time can primarily be spent on the stability and transport calculations. High accuracy is also required in order to minimize error propagation.

The equilibrium magnetic configuration in all toroidally axisymmetric magnetic confinement devices, including the tokamak and the spherical tokamak (ST), is determined by solving the Grad–Shafranov (GS) equation [15,29,38] (also known as the Grad–Schlüter–Shafranov equation). The Grad–Shafranov equation is a nonlinear elliptic partial differential equation, which, in general, can only be solved numerically. Because of its importance in computational plasma physics, a wide range of numerical methods have been developed [41]. These schemes generally fall into two categories. Eulerian or “direct” solvers use a prescribed mesh to calculate the unknown function [13,18,20,22,30], while Lagrangian or “inverse” solvers find the mapping of the plasma geometry in terms of magnetic coordinates [7,17,26,28,43]. The advantages and disadvantages of one formulation as compared to the other depend on the application of interest [44], on the plasma geometry, and on the type of

* Corresponding author.

E-mail addresses: apataki@apataki.net (A. Pataki), cerfon@cims.nyu.edu (A.J. Cerfon), jfreid@mit.edu (J.P. Freidberg), greengard@cims.nyu.edu (L. Greengard), oneil@cims.nyu.edu (M. O’Neil).

¹ Research supported in part by the National Science Foundation under Grant DMS06-02235, the US Department of Energy under contract DE-FG02-88ER-25053, and the Air Force Office of Scientific Research under NSSEFF Program Award FA9550-10-1-0180.

² Research supported in part by US Department of Energy under contract DE-FG02-91ER-54109.

inputs that plasma stability and transport codes require. For the Eulerian formulations, finite difference [13,18] and finite element [19,20,22,30,31] methods are systematically favored.

In this article we present a new, high-order, fixed boundary, direct GS solver and demonstrate its effectiveness as part of an Eulerian solver. High-order accuracy is achieved through the combination of three key elements: a re-scaling of the unknown which reformulates the GS equation as a nonlinear Poisson problem, a spectrally-accurate numerical method to compute the conformal map from the plasma cross-section to the unit disk, and finally, a fast, high-order Poisson solver on the unit disk.

Conformal mapping techniques in the context of numerical equilibrium solvers in plasma physics were first considered by Goedbloed [10–12] as a convenient way to decouple the numerical issues associated with the plasma geometry from the rest of the problem. We show that for smooth-boundary plasma cross-sections, spectrally-accurate conformal maps to the unit disk can be efficiently computed. It is important to note that the map needs to be computed only once for a particular geometry, and can be used to iteratively solve the nonlinear equation or to compute equilibria for several plasma profiles. By combining this mapping with a Green's function-based Poisson solver on the unit disk, we obtain high-order accuracy for the solution as well as its derivatives. This is one of the main motivations for this work, since the stability and transport properties of the plasma are very sensitive to quantities such as the local magnetic shear and the magnetic field curvature which depend on the second derivatives of the GS solution [9].

The structure of this article is as follows. In Section 2, we introduce the Grad–Shafranov equation and give the mathematical formulation of the fixed-boundary problem. In Section 3, we present a re-scaling of the unknown which reformulates the GS equation as a nonlinear Poisson problem, and describe the iterative method used for its solution. Section 4 contains the details of the numerical conformal mapping scheme which maps the plasma domain to the unit disk. The conformal map is computed by solving the Kerzman–Stein integral equation, and using its relation to the Szegő kernel. The fast Poisson solver on the unit disk is presented in Section 5. In Section 6, we illustrate the efficiency and accuracy of our new solver with a few specific examples, and in Section 7 we conclude with a discussion of the limitations of the solver, and its potential extensions.

2. The Grad–Shafranov equation as a nonlinear eigenvalue problem

2.1. The Grad–Shafranov equation

In toroidally axisymmetric systems, in the usual cylindrical coordinate system (R, ϕ, Z) , the magnetic field can be expressed as

$$\vec{B} = \frac{g(\Psi)}{R} \vec{e}_\phi + \frac{1}{R} \nabla \Psi \times \vec{e}_\phi, \quad (1)$$

where \vec{e}_ϕ is a unit vector in the toroidal direction. By the assumption of axisymmetry, none of the physical functions of interest depend on the angle ϕ . Here $2\pi\Psi(R, Z)$ represents the poloidal magnetic flux, and $2\pi g(\Psi) = -I_p(\Psi)$ is the net poloidal current flowing in the plasma and the toroidal field coils [9]. The flux function Ψ satisfies the Grad–Shafranov equation

$$\Delta^* \Psi = R \frac{\partial}{\partial R} \left(\frac{1}{R} \frac{\partial \Psi}{\partial R} \right) + \frac{\partial^2 \Psi}{\partial Z^2} = -\mu_0 R^2 \frac{dp}{d\Psi} - \frac{1}{2} \frac{dg^2}{d\Psi}, \quad (2)$$

where p is the plasma pressure. Both p and g are application-specific functions of Ψ , which, along with the boundary conditions, determine the equilibrium. Indeed, once (2) is solved and Ψ is known, p and g can be immediately evaluated, \vec{B} can be computed from (1), and the current density \vec{J} in the plasma is given by (see [9], for example):

$$\vec{J} = \frac{1}{\mu_0 R} \frac{dg}{d\Psi} \nabla \Psi \times \vec{e}_\phi - \frac{1}{\mu_0 R} \Delta^* \Psi \vec{e}_\phi. \quad (3)$$

2.2. Boundary conditions

Eq. (2) is a second-order elliptic nonlinear partial differential equation. Depending on the boundary conditions to be enforced, we distinguish between two general classes of problems: (i) fixed-boundary problems in which the plasma boundary is prescribed, with $\Psi = \text{constant}$ on the boundary and one solves for Ψ inside the plasma, and (ii) free-boundary problems where the current flowing in a set of external coils is given and one has to find Ψ such that the equilibrium is self-consistent with these currents. In this paper, we consider only fixed-boundary problems. That is, the shape of the plasma boundary is given and $\Psi = \text{constant}$ on the boundary. Furthermore, since only derivatives of Ψ represent physical quantities, we can choose $\Psi = 0$ on the boundary without loss of generality.

Our motivation for focusing on the fixed-boundary problem is twofold. First, a large number of plasma stability and transport numerical codes take fixed boundary equilibria as their initial data. In particular, parametric studies to understand and optimize plasma properties by varying a few parameters defining a given generic plasma boundary are very common.

Second, many free-boundary GS solvers use iterative schemes which require a robust fixed-boundary solver in the iterative loop [21,41].

2.3. The eigenvalue problem

As explained in the previous section, the generic form of the fixed-boundary GS equation can be written as

$$\begin{aligned} \Delta^* \Psi &= F(\Psi, R, Z) \quad \text{in } \Omega \\ \Psi &= 0 \quad \text{on } \partial\Omega, \end{aligned} \tag{4}$$

with

$$F(\Psi, R, Z) = -\mu_0 R^2 \frac{dp}{d\Psi} - \frac{1}{2} \frac{dg^2}{d\Psi},$$

where Ω represents the interior of the plasma, and $\partial\Omega$ the plasma boundary. The nature of the previous problem depends on the behavior of the right hand side as a function of Ψ ; the user-specified plasma profiles determine this. For many profiles, the right hand side F is of the form

$$F(\Psi, R, Z) = \tilde{F}(\Psi, R, Z) \Psi. \tag{5}$$

When F is of the above form, the function $\Psi = 0$ is a trivial, but not physically relevant, solution to Eq. (4). It is well-known [11,27,41] that in this case the Grad-Shafranov equation is solved as an eigenvalue problem, which is linear when F is linear in Ψ , and nonlinear otherwise. No matter the dependence of F on Ψ , we wish to normalize the problem. Consider the scalings

$$\Psi = |\Psi|_m \bar{\Psi}, \quad \frac{dp}{d\Psi} = \frac{\sigma}{|\Psi|_m} \frac{d\bar{p}(\bar{\Psi})}{d\bar{\Psi}}, \tag{6}$$

$$\frac{dg^2}{d\Psi} = \frac{\sigma}{|\Psi|_m} \frac{d\bar{g}^2(\bar{\Psi})}{d\bar{\Psi}}, \quad F(\Psi, R, Z) = \frac{\sigma}{|\Psi|_m} \bar{F}(\bar{\Psi}, R, Z), \tag{7}$$

where $|\Psi|_m = \sup |\Psi|$ and σ is a normalization factor for the pressure and poloidal current profiles (the function \bar{F} is *not* the same as \tilde{F}). Note that $\bar{\Psi}$, as defined in (6), takes values in the interval $[-1, 1]$. Inserting the normalizations in Eqs. (6) and (7) into (4), we find

$$\Delta^* \bar{\Psi} = \frac{\sigma}{|\Psi|_m^2} \bar{F}(\bar{\Psi}, R, Z). \tag{8}$$

Eq. (8) is in the desired form, highlighting a well-known property of the GS equation: its scale-invariance under the transformation

$$(\Psi, \sigma, R, Z) \rightarrow (\lambda \Psi, \lambda^2 \sigma, R, Z). \tag{9}$$

This scale invariance implies that by defining the ratio $\bar{\sigma} = \sigma / |\Psi|_m^2$, we can write the fixed-boundary GS equation in the following scale-independent form:

$$\begin{aligned} \Delta^* \bar{\Psi} &= \bar{\sigma} \bar{F}(\bar{\Psi}, R, Z) \quad \text{in } \Omega \\ \bar{\Psi} &= 0 \quad \text{on } \partial\Omega. \end{aligned} \tag{10}$$

This is now an eigenvalue problem to be solved for the eigenfunction $\bar{\Psi}$ with eigenvalue $\bar{\sigma}$. When transforming back to the original variables that represent physical quantities of interest, there are two choices. Either $|\Psi|_m$ is given (or alternatively I_ϕ , the total toroidal current flowing in the plasma), and the magnitude of the pressure and current profiles is computed from the relation $\sigma = |\Psi|_m^2 \bar{\sigma}$, or the normalization σ is given and the corresponding $|\Psi|_m$ or I_ϕ is determined.

On the other hand, the pressure profile or the poloidal current profile may be specified to include terms which are linear in Ψ . Under this assumption, their derivatives can be written in the form

$$-\mu_0 \frac{dp(\Psi)}{d\Psi} = -\mu_0 S(\Psi) \Psi + C, \quad -\frac{dg^2}{d\Psi} = T(\Psi) \Psi + A, \tag{11}$$

with A and C constants such that $C + A \neq 0$. This choice of p and g result in the right hand side F of Eq. (4) having a constant term. This choice corresponds to a discontinuous toroidal current J_ϕ at the plasma edge, and Eq. (4) does *not* correspond to an eigenvalue problem. While this is not a physically relevant choice, it plays an important role in the benchmarking of numerical Grad-Shafranov solvers. When the functions S and T are identically zero (corresponding to the so-called Solov'ev profiles [40]) exact analytic solutions to the GS equation with fixed boundary conditions can be constructed [9,45,5], which can then be used to test the accuracy of a numerical solver. In Section 6 we present convergence results of our solver benchmarked against such analytic solutions.

In what follows, we describe a new numerical scheme which solves Eq. (4) with high accuracy. Both the eigenvalue and non-eigenvalue problems will be considered. For the eigenvalue problem, we only treat the question of finding the smallest eigenvalue $\bar{\sigma}$ and its associated eigenfunction because almost all magnetic fusion confinement problems of interest have a single extremum of Ψ within the plasma region. Our procedure could however be easily generalized to find other eigenvalues, for so-called *doublet* or more ornate plasma configurations.

3. The Grad-Shafranov equation as an iterative Poisson problem

3.1. Transformation to a two-dimensional nonlinear Poisson problem

We consider the generic form of the fixed-boundary Grad-Shafranov equation, given by Eq. (4). In all but the simplest cases, an iterative scheme is necessary since either $F(\Psi, R, Z)$ is a nonlinear function of Ψ , or a nonlinear eigenvalue problem is to be solved (or both). As defined earlier, the differential operator Δ^* is a second-order elliptic operator, similar to the Laplacian:

$$\Delta^*\Psi = \Psi_{RR} - \frac{1}{R}\Psi_R + \Psi_{ZZ}.$$

If iteration is to be used, one might consider rewriting Eq. (4) as

$$\begin{aligned} \Psi_{RR} + \Psi_{ZZ} &= F(\Psi, R, Z) + \frac{1}{R}\Psi_R && \text{in } \Omega \\ \Psi &= 0 && \text{on } \partial\Omega \end{aligned} \quad (12)$$

and solve iteratively using a two-dimensional Cartesian Poisson solver at each step, holding the right hand side fixed at the previous iterate. However, this approach has one caveat. It is empirically observed that fixed-point iteration converges faster when the right-hand side is slowly varying. It would therefore be preferable to have a right-hand side that is only dependent on the unknown function itself, and not on its derivatives. In order to obtain this desired form, we consider the following scaling of the unknown function Ψ :

$$U(R, Z) = \frac{1}{\sqrt{R}}\Psi(R, Z). \quad (13)$$

This transformation is clearly singular at $R = 0$. Fortunately, this singularity does not have any consequence because in all applications of interest, the physical domain excludes the Z -axis. Inserting transformation (13) into (4), we readily find that U satisfies the following equation:

$$\begin{aligned} U_{RR} + U_{ZZ} &= \mathcal{F}(U, R, Z) && \text{in } \Omega \\ U &= 0 && \text{on } \partial\Omega, \end{aligned} \quad (14)$$

where \mathcal{F} is defined as

$$\mathcal{F}(U, R, Z) = \frac{1}{\sqrt{R}}F(\sqrt{R}U, R, Z) + \frac{3}{4R^2}U.$$

Here, \mathcal{F} is a function of U only, and not of its derivatives. Eq. (14) has the form of a two-dimensional nonlinear Poisson problem (treating R and Z as Cartesian coordinates), which we solve iteratively as described in the next section. Once U and all its derivatives have been computed, all quantities of physical interest can be evaluated to the same accuracy using the relations:

$$\begin{aligned} \Psi_R &= \frac{1}{2\sqrt{R}}U + \sqrt{R}U_R \\ \Psi_Z &= \sqrt{R}U_Z \\ \Psi_{RR} &= -\frac{1}{4R^{3/2}}U + \frac{1}{\sqrt{R}}U_R + \sqrt{R}U_{RR} \\ \Psi_{ZZ} &= \sqrt{R}U_{ZZ}. \end{aligned}$$

3.2. Iterative solution for the Grad-Shafranov equation

When the nonlinear Poisson problem defined by (14) does not correspond to an eigenvalue problem (i.e. the right hand side contains a constant term), we use a straight-forward fixed-point iteration scheme. At each iteration, the right-hand side of (14) is evaluated at the previous value of U , and the Laplacian $\Delta = \partial^2/\partial R^2 + \partial^2/\partial Z^2$ is inverted to calculate the updated value of U . In other words, on the l th iteration, we solve

$$\Delta U_l = \mathcal{F}(U_{l-1}, R, Z) \quad (15)$$

for U_l . The iterative process is stopped when $\|U_l - U_{l-1}\|_\infty < \epsilon$ for a few consecutive iterations, with ϵ small and chosen by the user.

The iterative scheme for the related eigenvalue problem is more subtle. Eq. (10) describes a *nonlinear* eigenvalue problem to be solved for the smallest eigenvalue $\bar{\sigma}$ and its associated eigenfunction $\bar{\Psi}$, with $\|\bar{\Psi}\|_\infty = 1$. Here, the norm $\|\cdot\|_\infty$ has its usual meaning, i.e. $\|\bar{\Psi}\|_\infty = \sup(|\bar{\Psi}|)$. When solving a linear eigenvalue problem, the eigenvalue does not depend on the scaling of the eigenfunction. However, this is generally not true for nonlinear problems. Therefore, we use a modified version of the well-known inverse iteration method [42] that evolves the eigenvalue–eigenfunction pair simultaneously instead of evolving the eigenfunction alone. First, the previous iterate eigen-pair $(\bar{\Psi}_{l-1}, \bar{\sigma}_{l-1})$ is used to solve the PDE

$$\begin{aligned} \Delta^* \Psi_l &= \bar{\sigma}_{l-1} F(\bar{\Psi}_{l-1}, R, Z) && \text{in } \Omega \\ \Psi_l &= 0 && \text{on } \partial\Omega. \end{aligned} \quad (16)$$

Then, the next eigen-pair iterate $(\bar{\Psi}_l, \bar{\sigma}_l)$ is computed via correcting by the norm of Ψ_l :

$$\bar{\Psi}_l = \frac{\Psi_l}{\|\Psi_l\|_\infty}, \quad \bar{\sigma}_l = \frac{\bar{\sigma}_{l-1}}{\|\Psi_l\|_\infty}. \quad (17)$$

Once again, the iterative process can be terminated when $\|\bar{\Psi}_l - \bar{\Psi}_{l-1}\|_\infty < \epsilon$ for a few consecutive iterations, with ϵ small and chosen by the user.

Applying the transformation (13) to (16) and (17) gives the iterative scheme in terms of U . First we solve the Poisson problem for U_l :

$$\begin{aligned} \Delta U_l &= \frac{\bar{\sigma}_{l-1}}{\sqrt{R}} F\left(\sqrt{R} U_{l-1}, R, Z\right) + \frac{3}{4R^2} U_{l-1} && \text{in } \Omega \\ U_l &= 0 && \text{on } \partial\Omega. \end{aligned} \quad (18)$$

Then, we compute the updated U_l and $\bar{\sigma}_l$ as:

$$U_l = \frac{U_l}{\|\sqrt{R} U_l\|_\infty}, \quad \bar{\sigma}_l = \frac{\bar{\sigma}_{l-1}}{\|\sqrt{R} U_l\|_\infty} \quad (19)$$

This eigenvalue iteration scheme generally converges at a geometric rate, as shown in [36]. Note that in this scheme, we calculate $\|\sqrt{R} U_l\|_\infty$ by first finding the maximum on the computational grid, and then using Newton's method to refine the maximum. Without this latter step, the problem we solve would depend on the computational grid and would converge only linearly, and not spectrally approximate $\|\sqrt{R} U_l\|_\infty$ as the grid is refined.

In summary, we have shown that the Grad–Shafranov equation can be solved using a sequence of iterations which require a single two-dimensional Poisson boundary value solve in Cartesian coordinates at each step. We now move onto developing a highly accurate Poisson solver in Ω with Dirichlet conditions $U = 0$ on $\partial\Omega$. This will be accomplished in two steps. First, we present a fast spectrally-accurate method for numerically computing the conformal map from our domain Ω to the unit disk. Then, we develop a fast, high-order Poisson solver on the unit disk using separation of variables.

4. Spectrally-accurate conformal mapping for smooth boundaries

4.1. Computation of the forward map of the boundary to the unit circle

In this section, the notation will be as follows. The original (R, Z) domain will be identified with the complex z -plane, with a point in this plane denoted as $z = x + iy$. The image domain will be referred to as the w -plane, with a point denoted as $w = \alpha + i\beta$. The forward map is the complex valued function $w = W(z)$ with real and imaginary parts $\alpha = A(x, y)$ and $\beta = B(x, y)$. The inverse map is the complex valued function $z = Z(w)$ with real and imaginary parts $x = X(\alpha, \beta)$ and $y = Y(\alpha, \beta)$ (see Fig. 1). Note that the function Z is obviously not the same mathematical object as the coordinate Z of the (R, ϕ, Z) coordinate system used in this article; this notation should not lead to any confusion since the two objects are never used simultaneously and can easily be distinguished based on the context.

The Riemann mapping theorem guarantees the existence and uniqueness of an analytic map between any simply-connected set and the unit disk. It is straightforward to show that under such a mapping, solving the Poisson problem

$$\Delta u(x, y) = f(x, y), \quad u|_{\partial\Omega} = 0,$$

in the original domain Ω is equivalent to solving the scaled Poisson problem

$$\Delta v(\alpha, \beta) = f(X(\alpha, \beta), Y(\alpha, \beta)) \left| \frac{dz}{dw} \right|^2, \quad v|_{\partial D_1} = 0,$$

on the mapped domain D_1 (in our case the unit disk), with

$$u(x, y) = v(A(x, y), B(x, y)).$$

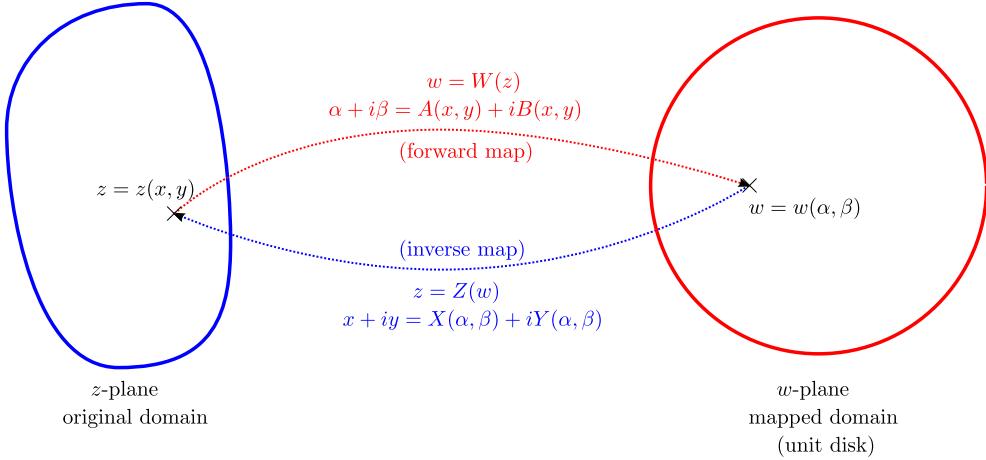


Fig. 1. Conformal map of a domain to the unit disk.

We compute the forward map W on $\partial\Omega$ using the Kerzman–Stein integral equation, see [23,24]. In particular, the derivative of the forward conformal map W on $\partial\Omega$ is given by

$$\begin{aligned} W'(z) &= 2\pi \frac{S^2(z,a)}{S(a,a)}, \\ S(a,a) &= \int_{\partial\Omega} \overline{S(z,a)} S(z,a) ds_z, \end{aligned} \quad (20)$$

where a is the point mapped to the origin, i.e. $W(a) = 0$. In the examples that follow, we always choose a as the geometric center of Ω (see Fig. 6). The function S is the Szegő kernel, which satisfies the integral equation

$$S(z,a) + \int_{\partial\Omega} \mathcal{A}(z,t) S(t,a) ds_t = \overline{H(a,z)}. \quad (21)$$

Here, the overline represents the complex conjugate, H is the Cauchy kernel,

$$H(w,z) = \frac{1}{2\pi i} \frac{dz/dl}{z-w}$$

and the kernel \mathcal{A} is known as the Kerzman–Stein kernel, defined as

$$\mathcal{A}(z,w) = \begin{cases} \overline{H(w,z)} - H(z,w), & z \neq w \\ 0, & z = w. \end{cases}$$

Numerical implementation of (21) is straightforward since the kernel \mathcal{A} is smooth. Since the trapezoidal rule is spectrally-accurate when applied to sufficiently smooth periodic functions, the boundary $\partial\Omega$, and therefore the integral in (21), can be discretized using N_1 points equally spaced in arc-length. The number of discretization points N_1 is chosen so that the boundary parameterization and its tangential derivatives are fully resolved to machine precision. Letting z_k and $(dz/dl)_k$ denote equispaced points and tangential derivatives on $\partial\Omega$ (specified either directly, or obtained from an equation of the plasma surface), the integral equation in (21) is then discretized resulting in a linear system of the form $\mathbf{Ms} = \mathbf{b}$, with

$$M_{jk} = \begin{cases} -\frac{h}{2\pi i} \left[\overline{\left[\frac{(dz/dl)_j}{z_j - z_k} \right]} + \frac{(dz/dl)_k}{z_k - z_j} \right], & j \neq k \\ 1, & j = k, \end{cases}$$

$$b_j = -\frac{1}{2\pi i} \overline{\left[\frac{(dz/dl)_j}{z_j - a} \right]},$$

$$s_k = S(z_k, a),$$

where $h = L/N_1$ is the step size (with L denoting the length of the boundary $\partial\Omega$). Naive inversion of the linear system $\mathbf{Ms} = \mathbf{b}$ to obtain \mathbf{s} requires $O(N_1^3)$ operations. There exist fast methods [35] which can be used to accelerate the solution of this linear system. However, since the conformal map needs to be computed only once in our GS solver, and since the iterative PDE solver is a significantly more expensive step, we have relied in our experiments on the simpler dense matrix formulation.

Once \mathbf{s} is known, the derivative of the forward map for the points on the boundary can be calculated using (20):

$$W'(z_k) = 2\pi \frac{s_k^2}{s_a}, \quad s_a = h \sum_{k=1}^{N_1} \overline{s_k} s_k. \quad (22)$$

Since \mathbf{s} is known at equispaced points in arc length, the FFT can be used to compute $W(z_k)$ by Fourier integration. The integration constant is determined so that the unit disk is centered at the origin. Specifically, three points on the mapped circle can be used to find its center, which is then subtracted from the map. Note that by using Fourier differentiation, we can also compute W'' , which is necessary for the mapping of the second derivatives of functions defined on the original domain and the mapped domain [36].

An example of the boundary values of the forward conformal map is shown in Fig. 2. It is easy to see that for arbitrary boundaries, equally spaced points on the boundary $\partial\Omega$ do not map to equally spaced points on the unit circle, a well known feature of conformal mapping known as *crowding*. In fact, the level of crowding is exponential in the ellipticity of Ω [35]. In many cases, high-order Poisson solvers on the disk which take advantage of Fourier methods require that discretization points be equally spaced in the angular direction. Since these points do not coincide with the ones obtained via the forward conformal map, it is necessary to resample the mapped points.

4.2. Computing the inverse map for equally spaced points on the unit circle

To obtain a uniformly spaced angular grid on the unit disk, we proceed in two steps. First, the boundary values of the forward map $W : \Omega \rightarrow \mathbf{D}_1$ are oversampled via the FFT. This is shown in Fig. 3. The amount of oversampling necessary is at least the crowding factor of the map.

Second, the image points on $\partial\mathbf{D}_1$ (the unit circle) are resampled to equispaced ones, effectively inverting the conformal map. We consider $z \in \partial\Omega$ as a function of $w \in \partial\mathbf{D}_1$ and use Lagrange interpolation to find z values corresponding to equally spaced w values on the circle. In particular, barycentric interpolation is used, which behaves well even for high order [3]. For example, 8th-order barycentric interpolation uses as data 4 points on the left and the right of the target interpolation point. The oversampling rate in the previous step can be tuned so that this interpolation procedure gives the desired accuracy, sometimes requiring a rate larger than the crowding factor. The end result of the interpolation step is shown in Fig. 4.

The equally spaced w boundary points we obtain in this step will be used for the angular grid of a high-order Poisson solver. We know their image $z = Z(w)$ on the boundary, but do not yet know the image of *interior* points of the grid. To compute the inverse map Z (as well as Z' and Z'') for interior points, the Cauchy integral formula can be used. Since the boundary of the domain is the unit circle, the integral formula is particularly easy to evaluate with the FFT. The inverse map of the interior points of a regularly spaced $N_r \times N_\theta$ grid on the unit disk can be computed in $O(N_r N_\theta \log N_\theta)$ time, the same computational complexity as the Poisson solver on the disk (as shown in the next section). This method is spectrally-accurate, and works all the way up to the boundary without any loss of precision or any requirement for adaptive integration (due to the proximity of the singularity of the Cauchy integral to the contour, see [36] for a detailed discussion).

We end this section with a short discussion on the previously mentioned issue of *crowding*, inherent to the conformal mapping technique, and clearly visible in Figs. 2–4 (see also [36]). For certain shapes (such as ones with high ellipticity), conformal maps are clearly not a viable method, at least when used in combination with solvers which require an equally spaced grid on the disk. For example, an ellipse with a major to minor axis ratio of 10 has a crowding factor on the order of 10^{10} ,

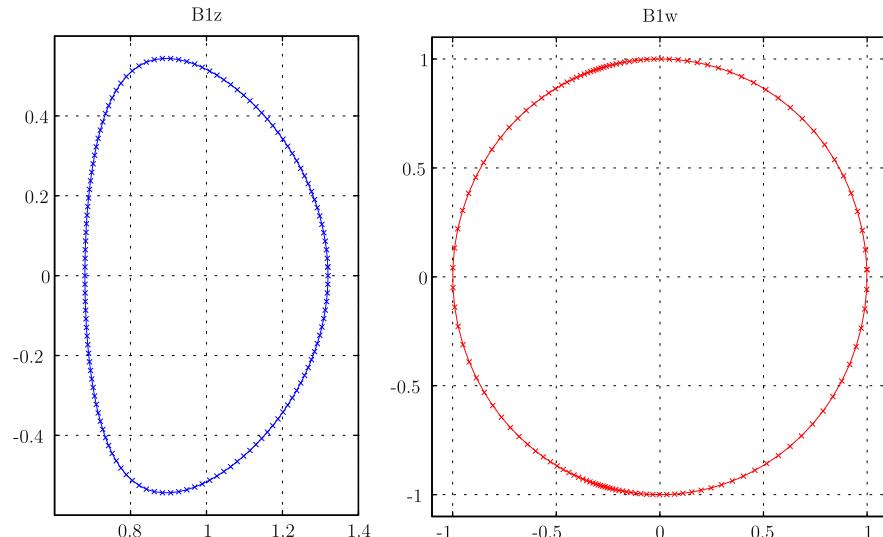


Fig. 2. Conformal map of boundary via the Kerzman–Stein integral equation: original domain (left), map of points to the unit circle (right).

which would obviously lead to prohibitive oversampling. Fortunately, for the cross-sections of tokamaks and spherical tokamaks, the crowding factor is much more manageable, on the order of $\sim 10 - 20$. For numerous geometries relevant to magnetic fusion, conformal mapping is an efficient method, as will be shown in more detail in Section 6.

5. High-order Poisson solver on the unit disk

What remains to be discussed in order to complete our Grad–Shafranov solver is the solution of the two-dimensional Poisson equation on the unit disk. There are, of course, a number of fast solvers available for this problem. For the sake of completeness, we describe one here that is direct, high-order accurate, and straightforward to implement. Other high-order schemes can be found, for example, in [39,6].

5.1. Separation of variables and boundary conditions

Let us consider the fixed-boundary Dirichlet Poisson problem

$$\begin{aligned} \Delta v &= f \quad \text{in } \mathbf{D}_1 \\ v &= 0 \quad \text{on } \partial\mathbf{D}_1, \end{aligned} \tag{23}$$

where \mathbf{D}_1 is the unit disk. Using separation of variables in the usual polar coordinates (r, θ) , we represent both the solution u and the right-hand f side as Fourier series:

$$v(r, \theta) = \sum_{n=-\infty}^{\infty} \hat{v}_n(r) e^{in\theta}, \quad f(r, \theta) = \sum_{n=-\infty}^{\infty} \hat{f}_n(r) e^{in\theta}. \tag{24}$$

Substituting these expressions into (23), we have the following radial ordinary differential equation for each Fourier mode n :

$$\begin{aligned} \hat{v}_n''(r) + \frac{1}{r} \hat{v}_n'(r) - \frac{n^2}{r^2} \hat{v}_n(r) &= \hat{f}_n(r) \\ \hat{v}_n(1) &= 0. \end{aligned} \tag{25}$$

In order for Eq. (25) to be well-posed, a second boundary condition must be enforced. We obtain this condition by requiring regularity of the solution as $r \rightarrow 0$. For the $n = 0$ mode, multiplying (25) by r and taking the limit $r \rightarrow 0$ leads to the condition $\hat{v}'_n(0) = 0$, under the assumption that \hat{f}_n , \hat{v}_n , and \hat{v}'_n are bounded. For $n \neq 0$, multiplying the equation by r^2 and again taking the limit $r \rightarrow 0$ leads to $\hat{v}_n(0) = 0$, under the same boundedness assumptions. In summary, we are to solve the following decoupled system of equations in the radial direction:

$$\begin{aligned} \hat{v}_0''(r) + \frac{1}{r} \hat{v}_0'(r) &= \hat{f}_0(r), \quad \hat{v}_0(0) = 0, \quad \hat{v}_0(1) = 0, \\ \hat{v}_n''(r) + \frac{1}{r} \hat{v}_n'(r) - \frac{n^2}{r^2} \hat{v}_n(r) &= \hat{f}_n(r), \quad \hat{v}_n(0) = 0, \quad \hat{v}_n(1) = 0, \quad \text{for } n \neq 0. \end{aligned} \tag{26}$$

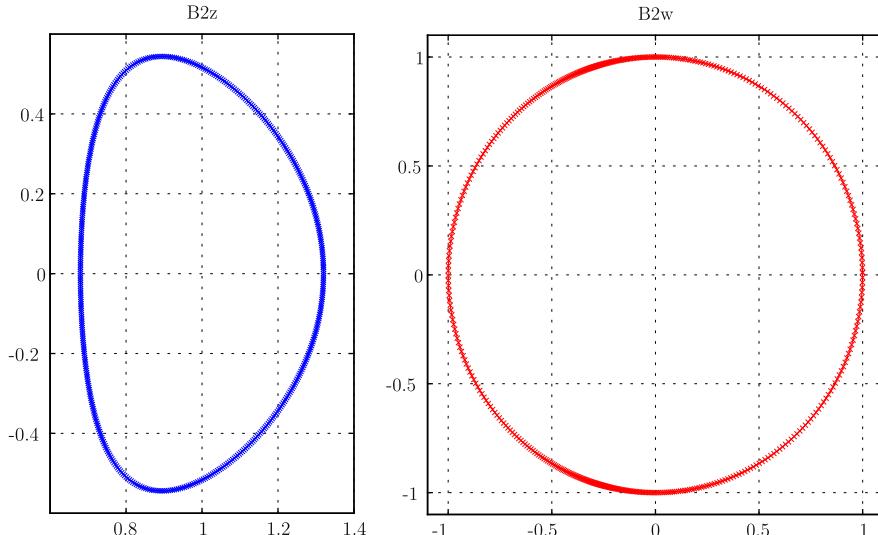


Fig. 3. Conformal map of boundary after oversampling: original domain (left), map of points to the unit circle (right).

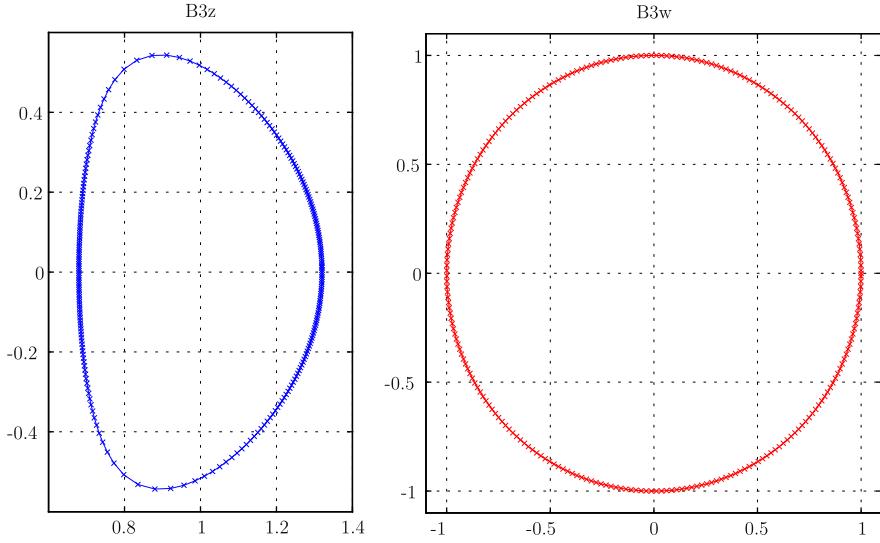


Fig. 4. Inverse of the conformal map of the boundary – equally spaced on the circle: original domain (left), unit circle (right).

5.2. Green's functions solution to the radial equation

Eq. (26) is a well studied ODE in mathematical physics, whose Green's function is known. Using convolution with the Green's function, a particular solution \hat{v}_n^P to the ODE which does *not* satisfy the boundary condition at $r = 1$ can immediately be written down. Then, a correction \hat{v}_n^H can be found which solves the homogeneous equation, and such that the sum $\hat{v}_n = \hat{v}_n^P + \hat{v}_n^H$ satisfies the ODE *and* the boundary conditions, both at $r = 0$ and at $r = 1$. To this end, we proceed as follows.

For the $n = 0$ mode, the Green's function with the proper behavior at 0 and infinity for the ODE in (26) is

$$G_0(r, s) = \begin{cases} s \log s, & r < s, \\ s \log r, & r > s \end{cases}$$

and for the $n \neq 0$ modes,

$$G_n(r, s) = \begin{cases} -\frac{1}{2|n|} r^{|n|} s^{-|n|+1}, & r < s, \\ -\frac{1}{2|n|} r^{-|n|} s^{|n|+1}, & r > s. \end{cases}$$

The solutions are then given by the convolutions

$$\begin{aligned} \hat{v}_0^P(r) &= \log r \int_0^r s \hat{f}_0(s) ds + \int_r^\infty s \log s \hat{f}_0(s) ds, \\ \hat{v}_n^P(r) &= -\frac{1}{2|n|} \left(r^{-|n|} \int_0^r s^{|n|+1} \hat{f}_n(s) ds + r^{|n|} \int_r^\infty s^{-|n|+1} \hat{f}_n(s) ds \right) \quad n \neq 0. \end{aligned}$$

Since we are solving the Poisson problem on the unit disk, we can set $\hat{f}_n(r) = 0$ for $r > 1$, and the convolution solutions to the inhomogeneous equation can be written in the form

$$\hat{v}_0^P(r) = \log r \int_0^r s \hat{f}_0(s) ds + \int_r^1 s \log s \hat{f}_0(s) ds, \tag{27a}$$

$$\hat{v}_n^P(r) = -\frac{1}{2|n|} \left(r^{-|n|} \int_0^r s^{|n|+1} \hat{f}_n(s) ds + r^{|n|} \int_r^1 s^{-|n|+1} \hat{f}_n(s) ds \right) \quad n \neq 0. \tag{27b}$$

The solution to the homogeneous equation

$$\hat{v}_n''(r) + \frac{1}{r} \hat{v}_n'(r) - \frac{n^2}{r^2} \hat{v}_n(r) = 0 \tag{28}$$

satisfying the regularity condition at $r = 0$ is

$$\hat{v}_n^H(r) = c_n r^{|n|},$$

where c_n is a constant to be determined from the boundary condition at $r = 1$:

$$\hat{v}_n(1) = \hat{v}_n^P(1) + \hat{v}_n^H(1) = \hat{v}_n^P(1) + c_n = 0 \quad \Rightarrow \quad c_n = -\hat{v}_n^P(1).$$

Therefore, the general solution to (26) satisfying the boundary conditions at $r = 0$ and $r = 1$ is

$$\hat{v}_n(r) = \hat{v}_n^P(r) - \hat{v}_n^P(1)r^{|n|}. \quad (29)$$

As previously discussed, one of the major advantages of Green's function methods is that the radial and angular derivatives of the solution v can be calculated explicitly by differentiating the separation of variables representation. Numerical differentiation is never required. This allows for accurate computation of the first and second derivatives of Ψ , which are required for stability and transport calculations.

The radial derivatives are computed by direct differentiation of (27a, 27b) and (29). For the $n = 0$ mode, we find

$$\begin{aligned} \hat{v}'_0(r) &= \frac{1}{r} \int_0^r s \hat{f}_0(s) ds, \\ \hat{v}''_0(r) &= \hat{f}_0(r) - \frac{1}{r^2} \int_0^r s \hat{f}_0(s) ds. \end{aligned} \quad (30)$$

For non-zero n modes, we have

$$\begin{aligned} \hat{v}'_n(r) &= \frac{1}{2} r^{-|n|-1} \int_0^r s^{|n|+1} \hat{f}_n(s) ds - \frac{1}{2} r^{|n|-1} \int_r^1 s^{-|n|+1} \hat{f}_n(s) ds - |n| \hat{v}_n^P(1) r^{|n|-1}, \\ \hat{v}''_n(r) &= \hat{f}_n(r) - \frac{|n|+1}{2} r^{-|n|-2} \int_0^r s^{|n|+1} \hat{f}_n(s) ds - \frac{|n|-1}{2} r^{|n|-2} \int_r^1 s^{-|n|+1} \hat{f}_n(s) ds - |n|(|n|-1) \hat{v}_n^P(1) r^{|n|-2}. \end{aligned} \quad (31)$$

Angular derivatives are obtained via differentiating the Fourier series representation (24), i.e. by a simple multiplication of in . The first and second partial derivatives of the solution v are then computed according to

$$\begin{aligned} v_r(r, \theta) &= \sum_{n=-\infty}^{\infty} \hat{v}'_n(r) / e^{in\theta}, \quad v_\theta(r, \theta) = i \sum_{n=-\infty}^{\infty} n \hat{v}_n(r) e^{in\theta}, \\ v_{rr}(r, \theta) &= \sum_{n=-\infty}^{\infty} \hat{v}''_n(r) e^{in\theta}, \quad v_{r\theta}(r, \theta) = i \sum_{n=-\infty}^{\infty} n \hat{v}'_n(r) e^{in\theta}, \\ v_{\theta\theta}(r, \theta) &= - \sum_{n=-\infty}^{\infty} n^2 \hat{v}_n(r) e^{in\theta}, \end{aligned}$$

where the radial derivatives $\hat{v}'_n(r)$ and $\hat{v}''_n(r)$ are given in Eqs. (30) and (31). As is well known, we observe that taking one derivative introduces a condition number of $\mathcal{O}(n)$, and taking a second derivative introduces a condition number of $\mathcal{O}(n^2)$, due to the multiplication by n and n^2 respectively.

5.3. Numerical considerations

5.3.1. Grid setup

Since the Poisson solver previously outlined relies on separation of variables in polar coordinates, we build a tensor grid in the radial and angular variables r and θ . In the θ variable, we compute and sum Fourier series using the FFT. When sampled at equispaced points, the representation in terms of Fourier series is spectrally-accurate for smooth data. Let N_θ denote the number of equispaced angular grid points in θ . In the radial direction, however, the domain is not periodic. Therefore, an appropriate high-order representation is a piecewise Chebyshev grid: the interval $[0,1]$ is divided into N_L intervals, and a P th order Chebyshev grid is constructed on each interval. The total number of points in the radial direction is therefore $N_r = PN_L$. The order P can be chosen as desired; in all subsequent examples we set $P = 16$, yielding a 16th order scheme. The grid used for a typical tokamak geometry (discussed in more detail in Section 6) is shown in Fig. 5, along with its image under the inverse conformal map in the original domain.

5.3.2. Evaluation of Green's function convolutions

Naive implementations of formulas (27a, 27b) require $O(N_r^2)$ work. However, the first integrals in (27a, 27b) can be computed recursively from $r = 0$ to $r = 1$, while the second integrals in (27a, 27b) can be computed recursively from $r = 1$ to $r = 0$. Thus, all integrals can be calculated in $O(N_r)$ work.

Furthermore, some care must be taken in the computation of the integrals in (27a, 27b) near $r = 0$ because of the rapid growth/decay of the monomials $s^{|n|}$ and $s^{-|n|}$ for large mode numbers n . The difficulties associated with floating-point overflow/underflow are easily avoided by re-scaling. A second problem is that for n large, the integrands $r^{|n|}/s^{(|n|-1)} \hat{f}_n(s)$ and $s^{|n|+1}/r^{|n|} \hat{f}_n(s)$ are poorly resolved by the composite Chebyshev grid. Since $\hat{f}_n(s)$ is well-resolved, however, the change of variables $\sigma = (r/s)^{|n|}$ and $\mu = (r/s)^{1-|n|}$ for the first and second cases, respectively, yield well-resolved functions in the transformed variable. These mapped integrals can be computed accurately with standard quadrature rules [36]. In our experiments below, we used a 16th order Gaussian rule.

5.3.3. Convergence and run time

If the data given is smooth, the Fourier series representation is spectrally-accurate. The overall order of convergence of the algorithm is therefore P , the order of the piecewise Chebyshev polynomials used in the radial ODE solver.

The run time complexity of the algorithm is $\mathcal{O}(N_r N_\theta \log N_\theta)$, nearly optimal with respect to the number of grid points. In detail, we compute $\mathcal{O}(N_r)$ FFTs of size N_θ at a cost of $\mathcal{O}(N_r N_\theta \log N_\theta)$, and N_θ ODE solves of complexity $\mathcal{O}(N_r)$ are performed at a total cost $\mathcal{O}(N_r N_\theta)$. Note that here we treat P as a fixed constant that would not be increased if the grid is refined.

6. Numerical tests – Examples

6.1. Comparison with exact analytic solutions

In order to test the accuracy of our Grad–Shafranov solver, we first consider a case where exact solutions are known. As discussed in Section 2, these are easy to construct for profiles of the form given in Eq. (11) with $S = 0$ and $T = 0$. For simplicity we set $A = 0$ (corresponding to a plasma which is neither paramagnetic nor diamagnetic [5]) and normalize the pressure such that $C = 1$. These choices are equivalent to solving the GS equation

$$\Delta^* \Psi = R^2, \quad (32)$$

which we solve both numerically and analytically. A simple analytic solution to (32) which is relevant to magnetic fusion can be constructed by following the methodology given in [5]. We sum a particular solution to the equation, $R^4/8$, with three solutions to the homogeneous equation:

$$\Psi_{\text{exact}}(R, Z) = \frac{R^4}{8} + d_1 + d_2 R^2 + d_3(R^4 - 4R^2 Z^2). \quad (33)$$

The free coefficients d_1 , d_2 , and d_3 are determined so that the contour $\Psi = 0$ represents a reasonable plasma cross-section. Specifically, we introduce three characteristic quantities describing the cross-section of a magnetic confinement device, as shown in Fig. 6: the inverse aspect-ratio ϵ , the elongation κ , and the triangularity δ .

The boundary conditions to be enforced are:

$$\begin{aligned} \Psi(1 + \epsilon, 0) &= 0 && \text{Condition at the outboard mid-plane,} \\ \Psi(1 - \epsilon, 0) &= 0 && \text{Condition at the inboard mid-plane,} \\ \Psi(1 - \delta\epsilon, \kappa\epsilon) &= 0 && \text{Condition at the top/bottom,} \end{aligned} \quad (34)$$

which gives the following system of three equations for d_1 , d_2 and d_3 :

$$\begin{bmatrix} 1 & (1 + \epsilon)^2 & (1 + \epsilon)^4 \\ 1 & (1 - \epsilon)^2 & (1 - \epsilon)^4 \\ 1 & (1 - \delta\epsilon)^2 & (1 - \delta\epsilon)^4 - 4(1 - \delta\epsilon)^2 \kappa^2 \epsilon^2 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = -\frac{1}{8} \begin{bmatrix} (1 + \epsilon)^4 \\ (1 - \epsilon)^4 \\ (1 - \delta\epsilon)^4 \end{bmatrix}. \quad (35)$$

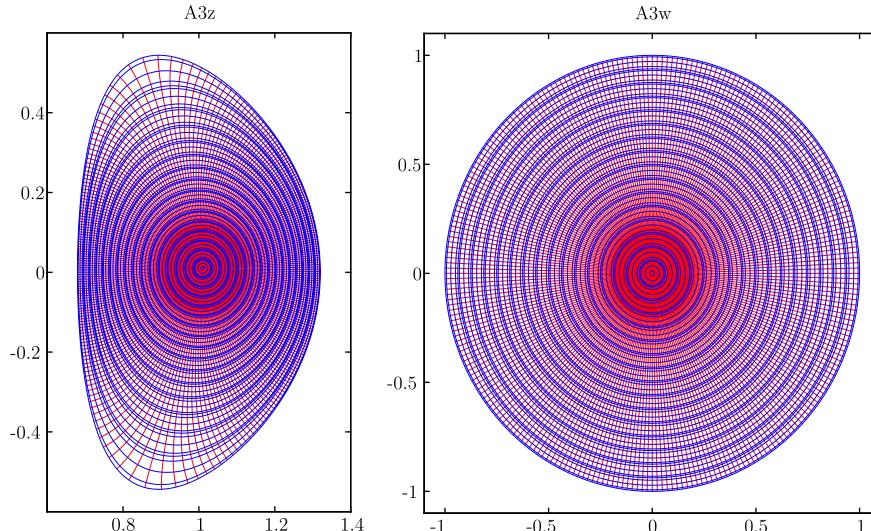
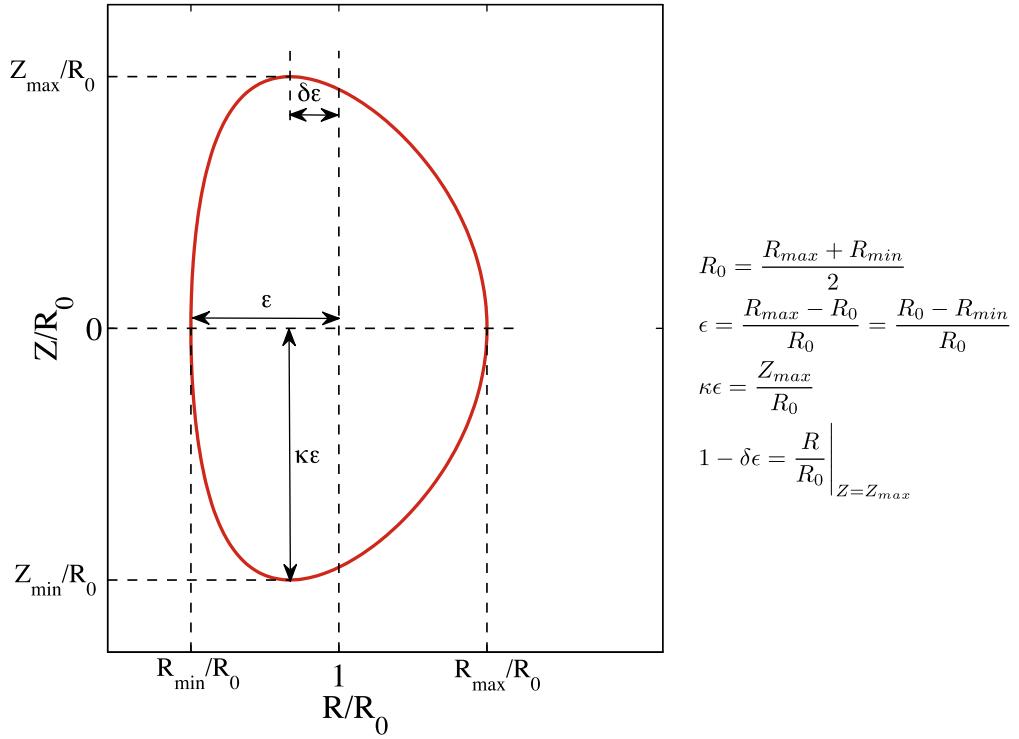


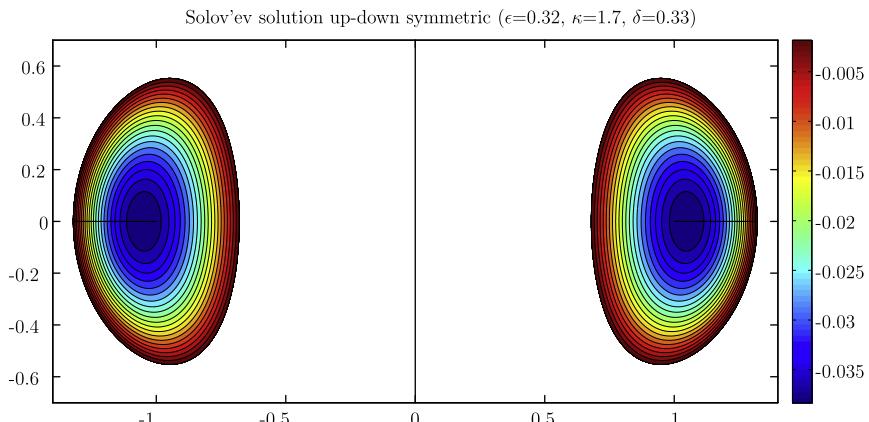
Fig. 5. Tensor grid for the equilibrium calculations in the tokamak presented in Section 6. Grid for the unit disk (right) and its image in the original domain (left).

**Fig. 6.** Geometric definition of the parameters ϵ , κ , and δ .

Eq. (35) is easily inverted, and once the coefficients d_1 , d_2 , and d_3 are determined, the analytic solution to (32) given by (33) is straightforwardly computed. Furthermore, the boundary of the plasma is given by the equation

$$\frac{R^4}{8} + d_1 + d_2 R^2 + d_3(R^4 - 4R^2 Z^2) = 0.$$

Using this plasma boundary allows the numerically obtained solution to be compared with the exact analytic one. The grid can then be refined enabling us to check the convergence of the scheme. Two examples are shown here: an ITER-like case [2,1] with parameters $\epsilon = 0.32$, $\kappa = 1.7$, $\delta = 0.33$, and an NSTX-like case [37] with $\epsilon = 0.78$, $\kappa = 2$, $\delta = 0.35$. Convergence is measured by computing the L^∞ norm (i.e. sup norm) of the difference between the numerical solution and the exact solution. For the ITER-like case, contour plots of the numerically obtained solution are shown in Fig. 7, and the convergence behavior is shown in Fig. 8. For the NSTX-like case, the analogous figures are Figs. 9 and 10.

**Fig. 7.** Solution to (32) – ITER parameters.

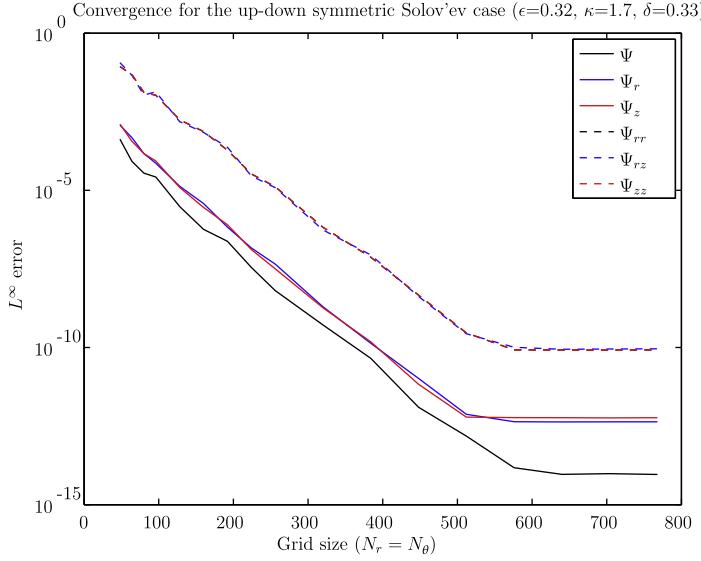


Fig. 8. Convergence of the numerical solution to (32), and of its first and second derivatives – ITER parameters.

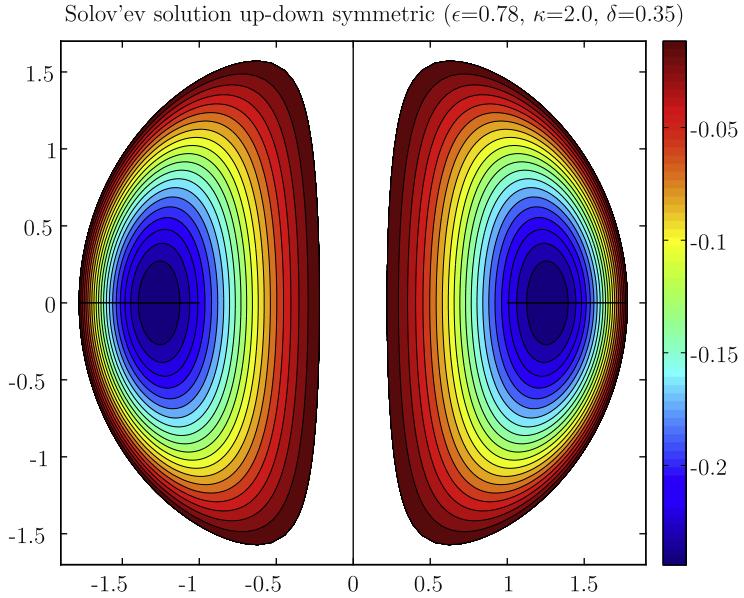


Fig. 9. Numerical solution to (32) – NSTX parameters.

Figs. 8 and 10 demonstrate the spectral convergence of the solution and its first and second derivatives. In the ITER-like case, Ψ is computed with an accuracy close to machine precision for a 600 by 600 grid. One can notice that the errors are a bit larger for the first and second derivatives. This is because we solve the Grad-Shafranov equation with Dirichlet boundary conditions, and in order to satisfy the differential equation near the boundary we effectively compute two derivatives of the boundary condition. This procedure introduces numerical differentiation errors, albeit only of size $\mathcal{O}(n)$ and $\mathcal{O}(n^2)$.

One can also observe that the NSTX-like case has worse convergence than the ITER-like case, and requires a finer grid to obtain comparable precision. This is a direct consequence of the conformal mapping method. Indeed, the NSTX-like cross-section has a crowding factor of about 20, compared to about 7 for the ITER-like case. Therefore, an extra oversampling factor of 3 is required in constructing the conformal map. In addition, the NSTX-like boundary requires slightly more points to be resolved.

The GS solver described is obviously not limited to problems with up-down symmetry (symmetry about the line $Z = 0$). Similar numerical tests were done with exact up-down asymmetric solutions to (32), and showed very similar performance [36].

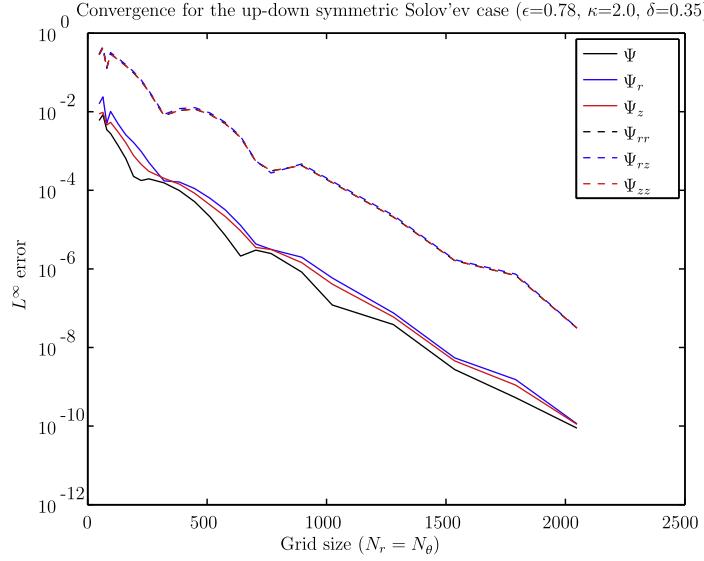


Fig. 10. Convergence of the numerical solution to (32), and of its first and second derivatives – NSTX parameters.

6.2. A linear eigenvalue problem – Equilibria with large Shafranov shifts

In the example in the previous section, the accuracy of our solver was compared with a known exact analytic solution. Unfortunately, in order to test against more realistic plasma profiles, comparison with an analytic solution is not possible. In this case, the L_∞ norm of the difference between successive iterates of the solution and its derivatives is used as a test for convergence. As an example, we will compute several equilibria which exhibit varying *Shafranov shifts* – a shift in the location of the maximum of Ψ , which marks the magnetic axis of the plasma. Using a geometry whose boundary is given in terms of the parameters ϵ , κ , and δ (see [5]),

$$\begin{aligned} x(t) &= 1 + \epsilon \cos(t + \alpha \sin t), \\ y(t) &= \epsilon \kappa \sin t, \end{aligned} \quad (36)$$

for $t \in [0, 2\pi]$, and $\sin \alpha = \delta$, we will solve for Ψ in the presence of quadratic plasma and toroidal field functions:

$$\Delta^* \Psi = \lambda (C_1 + C_2 r^2) \Psi. \quad (37)$$

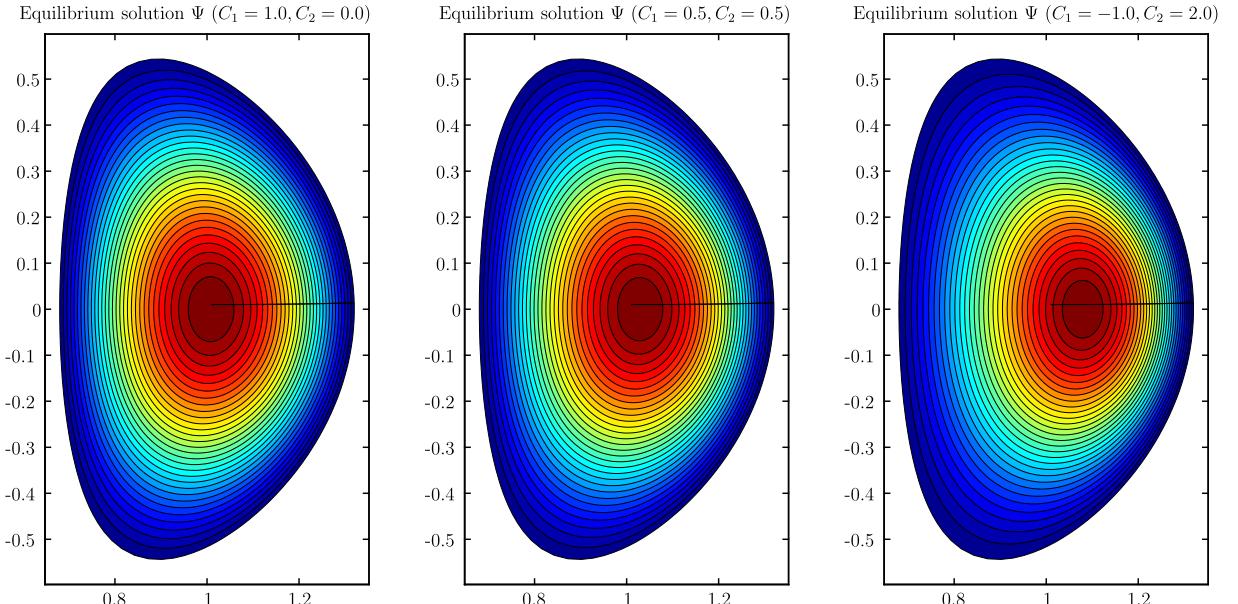


Fig. 11. Equilibria with increasing Shafranov shifts.

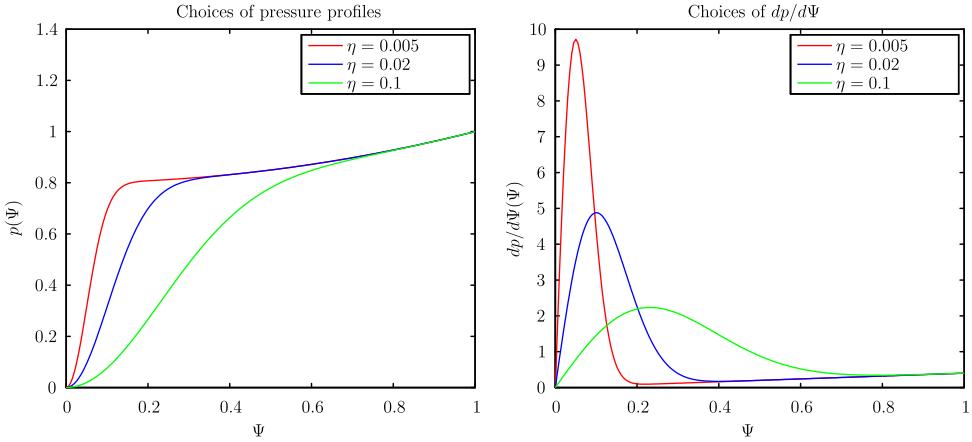


Fig. 12. Pressure profiles for $\eta = 0.005, 0.02, 0.1$ – ITER parameters.

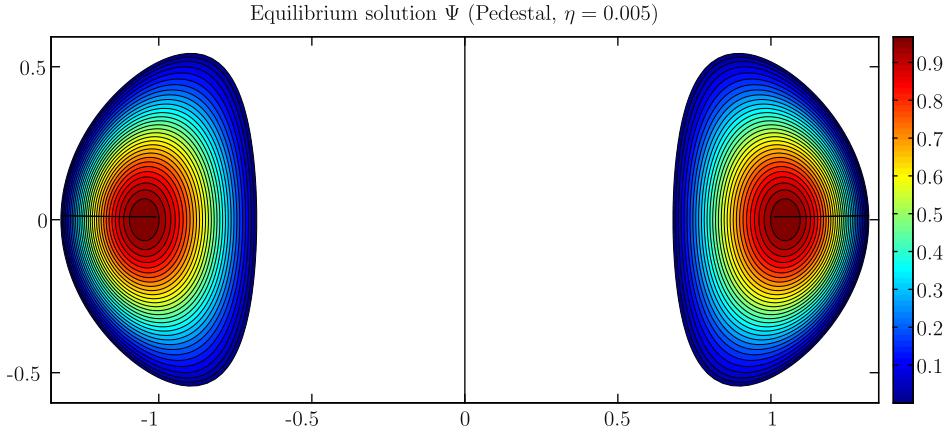


Fig. 13. Numerical solution to (38) for $\eta = 0.005$ – ITER parameters.

In the following numerical examples, we hold the geometry parameters constant at $\epsilon = 0.32$, $\kappa = 1.7$, and $\alpha = 0.336$, and vary the profile parameters C_1 and C_2 . In each test, equilibria with significant Shafranov shifts are computed without any difficulty [36]. See Fig. 11 for a cross-section of the profiles for various values of the parameters C_1 and C_2 .

6.3. A nonlinear eigenvalue problem – Numerical equilibrium with pressure pedestal

As an illustration of the Grad–Shafranov equation as a nonlinear eigenvalue problem (see Section 2), we consider a generic pressure profile corresponding to an equilibrium in an ITER-like (Eq. (36) with $\epsilon = 0.32$, $\kappa = 1.7$, and $\delta = 0.33$) configuration with a pressure pedestal [4,32]:

$$p(\Psi) = (C_1 + C_2\Psi^2)\left(1 - e^{-\Psi^2/\eta}\right),$$

where C_1 and C_2 are normalization constants, set to $C_1 = 0.8$ and $C_2 = 0.2$ in the examples shown below. The parameter η is a constant associated with the width and the steepness of the pressure pedestal. Equilibria are calculated for three different pedestal steepnesses corresponding to $\eta = 0.1$, $\eta = 0.02$ and $\eta = 0.005$ (see Fig. 12). Assuming that the plasma is neither paramagnetic nor diamagnetic ($dg^2/d\Psi = 0$), the GS equation becomes the following nonlinear eigenvalue problem (to within a constant factor that is subsumed into the eigenvalue):

$$\Delta^*\Psi = \bar{\sigma}R^2\left[2C_2\Psi\left(1 - e^{-\Psi^2/\eta}\right) + \frac{2}{\eta}(C_1 + C_2\Psi^2)\Psi e^{-\Psi^2/\eta}\right]. \quad (38)$$

We solve (38) for the eigenfunction Ψ and the eigenvalue $\bar{\sigma}$ using the iterative algorithm presented in Section 3. The contours of the solution for the case $\eta = 0.005$ are shown in Fig. 13.

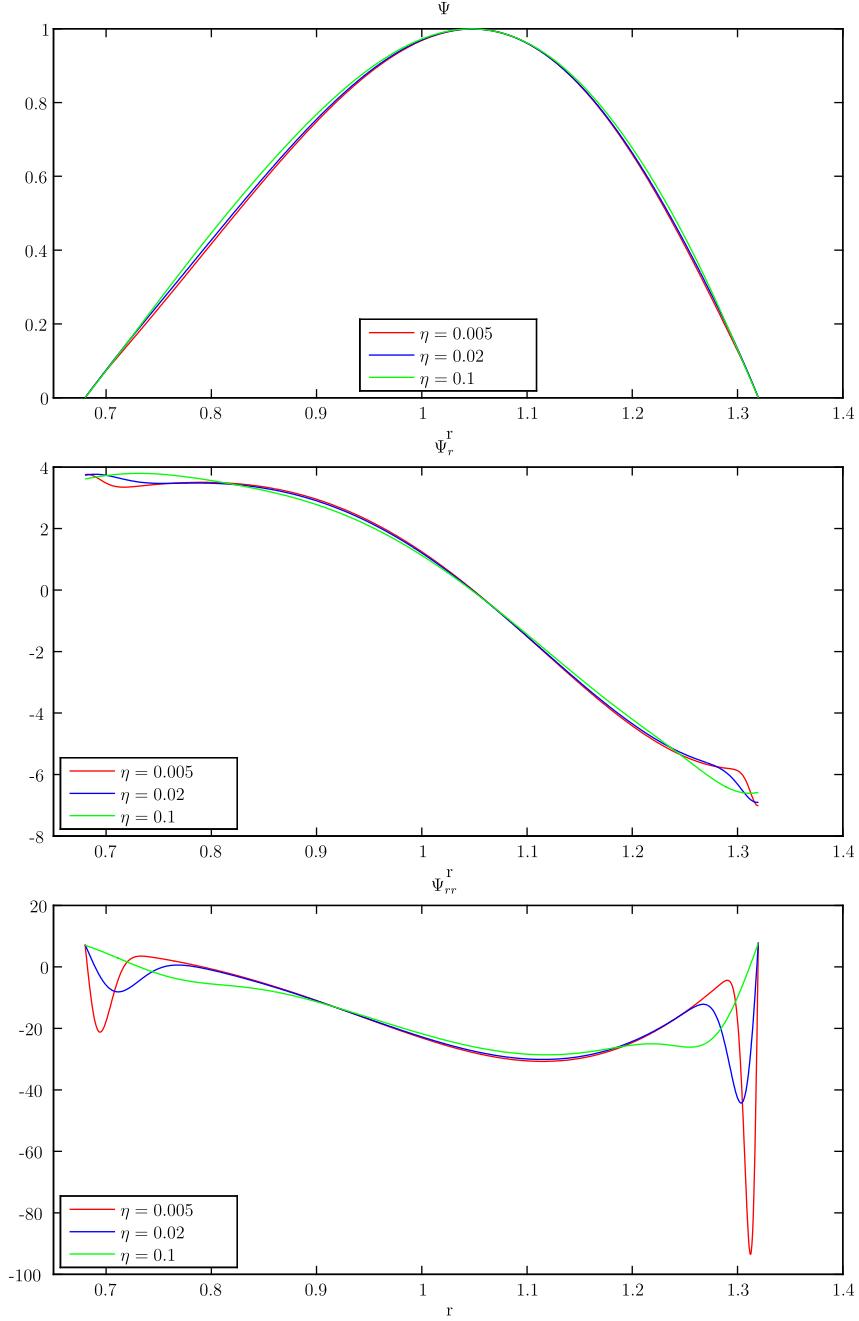


Fig. 14. Ψ, Ψ_r, Ψ_{rr} along $z = 0$.

In order to understand the influence of the steepness of the pressure pedestal on the solution, Ψ and its first and second radial derivatives are plotted along the line $z = 0$. The results are displayed in Fig. 14. We see that the solution Ψ and its first derivative Ψ_r depend only weakly on η . However, in the vicinity of the edge of the plasma, in the pedestal region, the behavior of the second derivative Ψ_{rr} is very sensitive to the value of η .

7. Conclusion

This article describes a new, fixed-boundary, direct Grad-Shafranov solver that relies upon conformal mapping and Green's function methods to compute high-order accurate plasma equilibria. Its attractive features are its proven spectral convergence, its speed, and its versatility in that arbitrary plasma boundaries can be given as input (both up-down symmetric and

asymmetric). The solver is shown to have very good performance for a wide range of pressure profiles, including fusion-relevant profiles with a steep pressure pedestal. Equilibria with large Shafranov shifts are also computed without difficulty.

The high accuracy achieved for the first and second derivatives of the solution suggests that this solver could successfully be combined with macroscopic stability codes, or implemented in transport studies on long time scales in which the equilibrium profiles are evolved self-consistently according to the GS equation.

This is particularly true for high aspect-ratio [and low ellipticity] devices such as the tokamak, for which crowding in the conformal mapping is not very pronounced, and relatively coarse grids yield very accurate solutions. In low aspect-ratio devices such as the ST, the high crowding factor observed in the conformal mapping suggests that alternative methods may be more efficient (because of the high ellipticity of the cross-section). In this regime, a Poisson solver relying on fast multipole methods [34,8,16,25] or compression [33], both of which are highly adaptive, may represent an attractive alternative. This would also permit calculations on non-smooth plasma boundaries with fusion-relevant X-points to be performed.

We should note that transport calculations often require equilibrium solvers to take the entropy profile and the safety factor (rotational transform) profile as inputs instead of the pressure p and poloidal current g profiles [14,27]. Some modifications to the algorithm presented in this article are needed to adapt our equilibrium solver to these situations. This is the subject of ongoing and future research.

References

- [1] R. Aymar, V. Chuyanov, M. Huguet, R. Parker, Y. Shimomura, The ITER project: a physics and technology experiment, in: Proceedings of the 16th International Conference on Fusion Energy, vol. 1, p. 3, Montreal, Canada, 1996.
- [2] R. Aymar, P. Barabaschi, Y. Shinomura, The ITER design, *Plasma Phys. Controlled Fusion* 44 (5) (2002) 519–565.
- [3] J.-P. Berrut, L.N. Trefethen, Barycentric Lagrange interpolation, *SIAM Rev.* 46 (3) (2004) 501–517.
- [4] M.N.A. Beurskens, T.H. Osborne, P.A. Schneider, E. Wolfrum, L. Frassinetti, R. Groebner, P. Lomas, I. Nunes, S. Saarela, R. Scannell, P.B. Snyder, D. Zarzoso, I. Balboa, B. Bray, M. Brix, J. Flanagan, C. Giroud, E. Giovannozzi, M. Kempenaars, A. Loarte, E. de la Luna, G. Maddison, C.F. Maggi, D. McDonald, R. Pasqualotto, G. Saibene, R. Sartori, E. Solano, M. Walsh, L. Zabeo, *H-mode pedestal scaling in DIII-D ASDEX upgrade and JET*, *Phys. Plasmas* 18 (5) (2011) 056120.
- [5] A.J. Cerfon, J.P. Freidberg, One size fits all analytic solutions to the Grad-Shafranov equation, *Phys. Plasmas* 17 (3) (2010) 032502.
- [6] H. Chen, Y. Su, B. Shizgal, A direct spectral collocation poisson solver in polar and cylindrical coordinates, *J. Comput. Phys.* 160 (2) (2000) 453–469.
- [7] J. DeLucia, S.C. Jardin, A.M.M. Todd, An iterative metric method for solving the inverse tokamak equilibrium problem, *J. Comput. Phys.* 37 (2) (1980) 183–204.
- [8] F. Ethridge, L. Greengard, A new fast-multipole accelerated poisson solver in two dimensions, *SIAM J. Sci. Comput.* 23 (3) (2001) 741–760.
- [9] J.P. Freidberg, 1 ed., *Ideal Magnetohydrodynamics*, Springer, New York, 1985.
- [10] J.P. Goedbloed, Conformal mapping methods in two-dimensional magnetohydrodynamics, *Comput. Phys. Commun.* 24 (3–4) (1981) 311–321.
- [11] J.P. Goedbloed, Some remarks on computing axisymmetric equilibria, *Comput. Phys. Commun.* 31 (2–3) (1984) 123–135.
- [12] J.P. Goedbloed, R. Keppens, S. Poedts, *Advanced Magnetohydrodynamics: With Applications to Laboratory and Astrophysical Plasmas*, Cambridge University Press, Cambridge, 2010.
- [13] P.-A. Gourdin, J.-N. Leboeuf, R.Y. Neches, High-resolution magnetohydrodynamic equilibrium code for unity beta plasmas, *J. Comput. Phys.* 216 (1) (2006) 275–299.
- [14] H. Grad, J. Hogan, Classical diffusion in a Tokomak, *Phys. Rev. Lett.* 24 (24) (1970) 1337–1340.
- [15] H. Grad, H. Rubin, Hydromagnetic equilibria and force-free fields, in: United Nations Conference on the Peaceful Uses of Atomic Energy, vol. 31, 1958, pp. 190–197.
- [16] L. Greengard, J.-Y. Lee, A direct adaptive poisson solver of arbitrary order accuracy, *J. Comput. Phys.* 125 (2) (1996) 415–424.
- [17] R. Gruber, R. Iacono, F. Troyon, Computation of MHD equilibria by a quasi-inverse finite hybrid element approach, *J. Comput. Phys.* 73 (1) (1987) 168–182.
- [18] J.A. Holmes, Y.-K.M. Peng, S.J. Lynch, Evolution of flux-conserving Tokamak equilibria with preprogrammed cross sections, *J. Comput. Phys.* 36 (1) (1980) 35–54.
- [19] E.C. Howell, C.R. Sovinec, NIMEQ: MHD equilibrium solver for NIMROD, in: APS Meeting Abstracts, Nov. 2008, p. 6041P.
- [20] G.T.A. Huysmans, J.P. Goedbloed, W. Kerner, Isoparametric bicubic Hermite elements for solution of the Grad-Shafranov equation, in: Proc. CP90 Conference on Comp. Phys. Proc., 1991, pp. 371–376.
- [21] S. Jardin, 1 ed., *Computational Methods in Plasma Physics*, Chapman & Hall/ CRC Press, New York, 2010.
- [22] S.C. Jardin, A triangular finite element with first-derivative continuity applied to fusion MHD applications, *J. Comput. Phys.* 200 (1) (2004) 133–152.
- [23] N. Kerzman, E.M. Stein, The Cauchy kernel, the Szegő kernel, and the Riemann mapping function, *Math. Ann.* 236 (1) (1978) 85–93.
- [24] N. Kerzman, M.R. Trummer, Numerical conformal mapping via the Szegő kernel, *J. Comput. Appl. Math.* 14 (1–2) (1986) 111–123.
- [25] H. Langston, L. Greengard, D. Zorin, A free-space adaptive FMM-based PDE solver in three dimensions, *Commun. Appl. Math. Comput. Sci.* 6 (1) (2011) 79–122.
- [26] K.M. Ling, S.C. Jardin, The princeton spectral equilibrium code: PSEC, *J. Comput. Phys.* 58 (3) (1985) 300–335.
- [27] L.L. LoDestro, L.D. Pearlstein, On the Grad-Shafranov equation as an eigenvalue problem with implications for q solvers, *Phys. Plasmas* 1 (1) (1994) 90–95.
- [28] G.O. Ludwig, Direct variational solutions of the tokamak equilibrium problem, *Plasma Phys. Controlled Fusion* 39 (1997) 2021–2037.
- [29] R. Lüst, A. Schlüter, Axial symmetrische magnetohydrodynamische Gleichgewichtskonfigurationen, *Z. Naturforsch.* 12a (1957) 850–854.
- [30] H. Lütjens, A. Bondeson, A. Roy, Axisymmetric MHD equilibrium solver with bicubic Hermite elements, *Comput. Phys. Commun.* 69 (2–3) (1992) 287–298.
- [31] H. Lütjens, A. Bondeson, O. Sauter, The CHEASE code for toroidal MHD equilibria, *Comput. Phys. Commun.* 97 (3) (1996) 219–260.
- [32] G. Maddison, A. Hubbard, J. Hughes, J. Snipes, B. LaBombard, I. Nunes, M. Beurskens, S. Erents, M. Kempenaars, B. Alper, S. Pinches, M. Valovič, R. Pasqualotto, A. Alfier, E. Giovannozzi, Dimensionless pedestal identity plasmas on Alcator C-Mod and Jet, *Nucl. Fusion* 49 (12) (2009) 125004.
- [33] P.-G. Martinsson, A fast direct solver for a class of elliptic partial differential equations, *J. Sci. Comput.* 38 (3) (2009) 316–330.
- [34] A. McKenney, L. Greengard, A. Mayo, A fast Poisson solver for complex geometries, *J. Comput. Phys.* 118 (2) (1995) 348–355.
- [35] S.T. Odonnell, V. Rokhlin, A fast algorithm for the numerical evaluation of conformal mappings, *SIAM J. Sci. Stat. Comput.* 10 (3) (1989) 475–487.
- [36] A. Pataki, High-order methods for elliptic problems in plasma physics, Ph.D. thesis, Courant Institute of Mathematical Sciences, New York University, 2011.
- [37] S.A. Sabbagh, S.M. Kaye, J. Menard, F. Paoletti, M. Bell, R.E. Bell, J. Bialek, M. Bitter, E.D. Fredrickson, D.A. Gates, A.H. Glasser, H. Kugel, L.L. Lao, B.P. LeBlanc, R. Maingi, R.J. Maqueda, E. Mazzucato, D. Mueller, M. Ono, S.F. Paul, M. Peng, C.H. Skinner, D. Stutman, G.A. Wurden, W. Zhu, NSTX Research team, Equilibrium properties of spherical torus plasmas in NSTX, *Nucl. Fusion* 41 (11) (2001) 1601–1611.
- [38] V.D. Shafranov, On magnetohydrodynamical equilibrium configurations, *Sov. Phys. JETP* 6 (3) (1958) 545–554.

- [39] J. Shen, Efficient spectral-Galerkin methods III: polar and cylindrical geometries, *SIAM J. Sci. Comput.* 18 (6) (1997) 1583–1604.
- [40] L.S. Solov'ev, The theory of hydromagnetic stability of Toroidal plasma configurations, *Sov. Phys. JETP* 26 (2) (1968) 400–407.
- [41] T. Takeda, S. Tokuda, Computation of MHD equilibrium of Tokamak plasma, *J. Comput. Phys.* 93 (1) (1991) 1–107.
- [42] L. Trefethen, D. Bau, *Numerical Linear Algebra*, SIAM, 1997.
- [43] B. Turkington, A. Lifschitz, A. Eydeland, J. Spruck, Multiconstrained variational problems in magnetohydrodynamics: equilibrium and slow evolution, *J. Comput. Phys.* 106 (2) (1993) 269–285.
- [44] L. Zakharov, A. Pletzer, Theory of perturbed equilibria for solving the Grad–Shafranov equation, *Phys. Plasmas* 6 (12) (1999) 4693–4704.
- [45] S.B. Zheng, A.J. Wootton, E.R. Solano, Analytical Tokamak equilibrium for shaped plasmas, *Phys. Plasmas* 3 (3) (1996) 1176–1178.