Coordinate parameterisation and spectral method optimisation for Beltrami field solver in stellarator geometry

Z. S. Qu[1], D. Pfefferlé[2], S. R. Hudson[3], A. Baillod[4], A. Kumar[1], R. L. Dewar[1], M. J. Hole[1,5]

[1]Mathematical Sciences Institute, the Australian National University, Canberra ACT 2601, Australia

[2]The University of Western Australia, 35 Stirling Highway, Crawley WA 6009, Australia

[3]Princeton Plasma Physics Laboratory, PO Box 451, Princeton, New Jersey 08543, USA

[4]École Polytechnique Fédérale de Lausanne, Swiss Plasma Center, CH-1015 Lausanne, Switzerland

[5]Australian Nuclear Science and Technology Organisation, Locked Bag 2001, Kirrawee DC NSW 2232, Australia

E-mail: `zhisong.qu@anu.edu.au`

Abstract.
The numerical solution of the stepped pressure equilibrium [Hudson *et al*, Phys. Plasmas 19 112502 (2012)] requires a fast and robust solver to obtain the Beltrami field in three-dimensional geometry such as stellarators. The spectral method implemented in the Stepped Pressure Equilibrium Code (SPEC) is efficient when the domain is a hollow torus, but ill-conditioning of the discretised linear equations occurs on the (solid) toroid due to the artificially singular coordinate parameterisation near the axis. In this work, we propose an improved choice for the reference axis to prevent coordinates surfaces from overlapping. Then, we examine the parity and asymptotics of the magnetic vector potential near the axis and suggest the use of recombined and rescaled Zernike radial basis functions. The maximum relative error in the magnetic field of the Wendelstein 7-X geometry is shown to reach 10^{-9} at high resolution in a series of convergence tests and benchmarks against the boundary integral solver BIEST. The new method is also reported to significantly improve the accuracy of multi-volume SPEC calculations. A comparison between free-boundary SPEC and the analytical Dommaschk potential is presented with higher-than-usual Fourier resolution. It is illustrated that we are able to resolve low amplitude current sheets when an interface is placed where there is no flux surface in the analytic solution. This was previously concealed because of insufficient numerical resolution.

1. Introduction

The problem of representing an ideal magnetohydrodynamics (MHD) equilibrium in three-dimension (3D) stellarator geometry is subtle in the absence of axisymmetry [1]. The ideal MHD force balance equation has the simplest form $\nabla p = \mathbf{J} \times \mathbf{B}$, where p is the plasma pressure, $\mathbf{B}$ the magnetic field, $\mathbf{J} = \nabla \times \mathbf{B}/\mu_0$ the current density, and μ_0 the magnetic permeability constant. The magnetic field structure in 3D equilibrium has a combination of flux surfaces, islands and chaos [2]. The most widely used 3D equilibrium code, VMEC [3], restricts the class of solutions to have nested toroidal flux surfaces everywhere. However, the existence of the unphysical Pfirsch-Schlüter current on flux surfaces with a rational rotational transform and pressure gradient remains an issue, as the current contains a $1/x$ singularity whose volume integral is not bounded [4, 5]. The presence of islands and chaos are also ignored. To avoid these problems, one may choose to relax the ideal condition by introducing non-ideal effects, such as PIES [6], HINT [7] and SIESTA [8].

Instead of relaxing the ideal condition, one can choose to relax the smoothness condition. The stepped pressure equilibrium, introduced by Bruno and Laurence [9], considers a weak solution to the ideal MHD force balance by allowing discontinuous magnetic field and a piece-wise constant pressure profile. In a stepped pressure equilibrium, the plasma volume is partitioned into a number of sub-volumes. Within each volume, the magnetic field satisfies the Beltrami equation given by

$$\nabla \times \mathbf{B} = \mu\mathbf{B}, \tag{1}$$

where μ is known as the helicity multiplier in each volume. These sub-volumes are separated by ideal interfaces with tangential (ideal) boundary condition given by

$$\mathbf{B} \cdot \mathbf{n} = 0, \tag{2}$$

where $\mathbf{n}$ is the unit vector normal to the surface of the interface, in addition to the toroidal flux (and if the region is annular, the poloidal flux) being specified.

The pressure p is a constant within each volume but can jump at the interfaces, giving a stepped profile. This pressure difference is balanced by a jump in the magnetic pressure $B^2/2$, keeping the total pressure unchanged on either side of the interface, i.e.

$$\left[\!\left[p + \frac{B^2}{2} \right]\!\right] = 0, \tag{3}$$

where $[\![\cdots]\!]$ stands for the difference between either side of the interface. The pressure p here should be understood as $\mu_0 p$ in SI unit. The stepped pressure equilibrium allows the formation of islands and chaos and is free from the problems of diverging singular currents. It was found later by Hole, Hudson and Dewar [10, 11] that the stepped pressure equilibrium can be formulated as a variational problem and described by the stationary points of the Multi-region Relaxed MHD (MRxMHD)

energy functional. To access stepped pressure equilibrium solutions with complicated geometry and parameters, the Stepped-Pressure Equilibrium Code (SPEC) [12, 13] was built and verified in stellarator geometry [14]. Since then, MRxMHD and SPEC have been extended both theoretically [15] and numerically to capture flow [16, 17], pressure anisotropy [18], two fluid effects [19] and time evolution [20, 21, 22]. They are also served as useful tools to investigate equilibrium bifurcation [15], the formation of singular current sheets [4], the penetration of resonant magnetic perturbation [5, 23], the equilibrium beta limit [24] and tearing modes [25, 26]. A free-boundary version of SPEC was also developed recently [27].

The success of SPEC relies on the Beltrami field solver to compute the solution of (1) and (2) using a spectral Galerkin method. Its robustness and efficiency can however be improved, especially in the toroidal region that contains the coordinate axis. In this paper, we focus on the following points:

(i) Once the boundary of the computational domain is given, a set of coordinates to parametrise the interior needs to be constructed. Currently, SPEC uses a toroidal coordinate system (s, θ, ζ), with s being the radial coordinate, θ and ζ being generalised poloidal and toroidal angles, respectively. It selects the centroid of the boundary to be the coordinate axis $s = 0$. For a strongly shaped non-convex domain, the centroid can be outside the plasma boundary. Even when the coordinate axis is inside the plasma boundary, there is nothing that prevents the coordinate surfaces from overlapping, which immediately leads to a failure of the solver. A mathematical explanation and justification of the choice of the coordinate parameterisation is missing.

(ii) An ill-conditioning problem at high Fourier resolutions has been identified [28, 27], mostly due to the artificial singularity in toroidal coordinates, and the choice of its radial basis functions. This leads to the failure of the linear matrix solver.

(iii) There is a pressing requirement to make SPEC faster and more memory efficient, so that it can serve as a core equilibrium solver within stellarator optimisation codes. This will enable the effect of islands and chaos being taken into account in the stellarator optimization process.

In comparison, a new code based on boundary integral methods, the Boundary Integral Equation Solver for Taylor states (BIEST), was developed [29, 28, 30] to alleviate part of these issues. BIEST solves the magnetic field only on the boundary surface, after which the field at an arbitrary interior point can be computed by an integral of the solution on the boundary (layer potential). No coordinate parameterisation is needed and no coordinate singularity is present leading to a well-conditioned problem. The plasma boundary in BIEST can be non-smooth, while SPEC is limited to having only smooth boundaries. The SPEC Beltrami solver, however, has some significant advantages over BIEST. For instance, the magnetic field is easy to compute everywhere within the volume once the solution is obtained. Moreover, it constructs the magnetic vector potential $\mathbf{A}$ directly, so that the magnetic helicity $H = \int \mathbf{A} \cdot \mathbf{B} \, dV$ is easily accessible to study

plasma relaxation.

In this work, we present a number of critical improvements to the SPEC Beltrami solver in toroidal regions, tackling the aforementioned difficulties. An improved coordinate parameterisation is presented for a better specification of the coordinate axis. Part of the ill-conditioning problem is mitigated by the use of Zernike basis functions and basis recombination/rescaling. Finally, the speed and memory efficiency is optimised by the use of iterative linear matrix solver and matrix-free method, making SPEC faster than its previous version and as accurate as BIEST.

This article is organised as follows. Section 2 describes our choice of coordinate parameterisation and geometric axis. Section 3 then deals with the parity and regularity condition at the coordinate axis and introduces the set of basis functions used. Section 4 outlines the discretised Beltrami equation and how the resulting linear equations are solved. Verification, benchmarks and numerical experiments are preformed in Section 5. Section 6 discusses the results and draws useful conclusions.

2. Coordinate parameterisation

The first step of the computation is to prescribe the plasma boundary in cylindrical coordinates as $R = R(\theta, \zeta)$, $Z = Z(\theta, \zeta)$ and $\varphi = \zeta$. A solution to the Beltrami equation (1) is then sought that satisfies the boundary condition (2) and a flux condition in the plasma volume enclosed by the given plasma boundary. For this, the magnetic field is discretised within the plasma volume. Unlike the finite element method that requires a mesh and a collection of low-degree polynomial basis functions with local support, SPEC adopts a *spectral method*, in which the field is approximated by a (linear) combination of *global* basis functions. In particular, the original SPEC uses Fourier-Chebyshev series for the angular/radial direction, respectively. The spectral method has the advantage of displaying a spectral convergence rate [31]. In other words, the difference between the numerical solution and the analytical solution decays exponentially as the resolution increases, e.g. by including more Fourier modes and/or radial basis functions.

The boundary condition is difficult to enforce should the computation be carried out in cylindrical geometry given the complicated boundary. SPEC proposes to construct a set of toroidal coordinates (s, θ, ζ) to parametrise the plasma volume, with $s = 0$ on the geometric axis and $s = 1$ on the boundary. The components of the magnetic field are expressed in terms of the global basis functions in (s, θ, ζ). The geometric axis, similar to the axis in cylindrical geometry, is an artificial reference point whose choice should not affect the resulting magnetic field (represented in Cartesian coordinates). With this system, the boundary conveniently corresponds to the surface $s = 1$ and therefore the boundary condition is easy to enforce.

This is equivalent to mapping the plasma volume into a standard torus (tokamak with circular cross-section), so that the plasma boundary at a fixed toroidal angle corresponds to a circle. The Beltrami equation is then solved over a simpler domain in these coordinates with non-Euclidean metric, and the solution is easily mapped back

to real space. We note that the mapping is unrelated to the magnetic field within the plasma volume, but depends only on the geometric information of the plasma boundary. The magnetic field is a priori unknown.

The desired properties of the coordinate parameterisation are as follows. The mapping should be

(i) real analytic; a Taylor series of the mapping around any point should converge.

(ii) continuous and bijective; for every point in the (s, θ, ζ) domain, there should correspond a unique point in the plasma domain and vice versa.

The first property is to ensure the spectral convergence of the numerical method, and is automatically satisfied in SPEC by writing the parameterisation of R, Z in terms of polynomials in s and Fourier series in θ and ζ. The second property is essential to the well-posedness of the boundary-value problem and the consistency of the discretisation scheme, but is unfortunately not automatic.

Formally, the toroidal plasma domain $\bar{\Omega} \subset \mathbb{R}^3$ is considered to be the image of a smooth embedding into the Euclidean 3-space of the standard solid torus $\bar{M} = \bar{D}^2 \times S^1$, created by sweeping a unit disk $\bar{D}^2 = \{(x, y) \in \mathbb{R}^2 | x^2 + y^2 \leq 1\}$ around the unit circle $S^1 = \{(x_3, x_4) \in \mathbb{R}^2 : x_3^2 + x_4^2 = 1\}$. We denote the embedding $F : \bar{M} \to \mathbb{R}^3$. Since $\bar{M}$ is a compact manifold (with boundary), it is sufficient to ensure that the derivative of the map F is everywhere injective. Since $\bar{M}$ and $\bar{\Omega}$ are three-dimensional, it is equivalent to verifying that the determinant of the Jacobian matrix nowhere vanishes.

Starting from an obvious local coordinate for S^1, namely the toroidal angle ζ such that $(\cos\zeta, \sin\zeta) \in S^1$, we consider mappings $F : \bar{M} \to \mathbb{R}^3$ locally of the form

$$F(x, y, \zeta) = (R(x, y, \zeta)\cos\zeta, R(x, y, \zeta)\sin\zeta, Z(x, y, \zeta)), \tag{4}$$

where $R, Z \in C^\infty(\bar{M}, \mathbb{R})$ are smooth functions on $\bar{M}$. The toroidal angle ζ coincides with the (geometric) polar angle of cylindrical coordinates. The Jacobian determinant of the map F at (x, y, ζ) is

$$\det J_F(x, y, \zeta) = - R(\partial_x R \partial_y Z - \partial_y R \partial_x Z). \tag{5}$$

It is non-zero if $R(x, y, \zeta) > 0$ and the "2D Jacobian" satisfies

$$J_{D^2}(x, y, \zeta) = \frac{\partial R}{\partial x}\frac{\partial Z}{\partial y} - \frac{\partial Z}{\partial x}\frac{\partial R}{\partial y} \neq 0. \tag{6}$$

This condition is equivalent to asking that the level sets of R and Z on a poloidal plane at a fixed ζ are nowhere tangential. A necessary requirement is that R and Z have no critical points (zero gradient) over the unit disk.

SPEC explicitly uses the coordinate parameterisation R, Z of the form

$$
\begin{aligned}
R(x,y,\zeta) &= s^2 R_0(\zeta) + (1 - s^2) R_x(\zeta) \\
&+ \sum_{n=-N}^{N} \sum_{m=1}^{M} R_{m,n} s^m \cos(m\theta - nN_p\zeta), \\
Z(x,y,\zeta) &= s^2 Z_0(\zeta) + (1 - s^2) Z_x(\zeta) \\
&+ \sum_{n=-N}^{N} \sum_{m=1}^{M} Z_{m,n} s^m \sin(m\theta - nN_p\zeta),
\end{aligned}
\tag{7}
$$

where $x = s\cos\theta$ and $y = s\sin\theta$ are polar coordinates on the unit disk, $M, N \in \mathbb{N}^+$ is the Fourier resolution, $N_P \in \mathbb{N}^+$ the field period, and $R_{m,n}, Z_{m,n} \in \mathbb{R}$ are Fourier coefficients. The pair (R_x, Z_x) locates the image through F of the origin of the unit disk on a poloidal plane at fixed ζ. The curve traced in $\mathbb{R}^3$ is referred to as the *geometric axis*. Again, we emphasis that it is similar to the cylindrical axis and has no particular physical meaning, other than being the reference axis for the (local) toroidal coordinate system (s, θ, ζ). It is not necessarily the magnetic axis. In fact, there can be multiple magnetic axes, or no apparent axis if the region displays chaotic field-lines. The pair (R_0, Z_0) is the θ-*average* of the shape drawn by the plasma boundary on a poloidal plane at fixed ζ. The form of these functions is taken to be

$$
\begin{aligned}
R_{0/x}(\zeta) &= \sum_{n=0}^{N} R_{0/x,n} \cos(-nN_P\zeta), \\
Z_{0/x}(\zeta) &= \sum_{n=1}^{N} Z_{0/x,n} \sin(-nN_P\zeta).
\end{aligned}
\tag{8}
$$

In the expressions above, we are assuming stellarator symmetry [32] such that $R(s, \theta, \zeta) = R(s, -\theta, -\zeta)$ and $Z(s, \theta, \zeta) = -Z(s, -\theta, -\zeta)$. This is an imposed symmetry for stellarator design purpose and reduces to the up-down symmetry in the tokamak case. Our method can be straightforwardly extended to the general case.

2.1. Finding the coordinate axis to ensure a non-zero Jacobian

With arbitrary Fourier coefficients $\{R_{m,n}, Z_{m,n}\}_{m,n}$ and $\{R_{x,n}, Z_{x,n}\}_n$, the map defined via (7) may not be an embedding. Two different points of (s, θ, ζ) can be mapped to the same (X, Y, Z), resulting in overlap. The transformation (7) only "discourages" overlapping but does not prevent it completely [13]. In such cases, there are points where $J_{D^2} = 0$ within the plasma volume. We should examine the properties of (7) and find conditions that help meet the stated requirements. In particular, efforts should be made to ensure $J_{D^2} \neq 0$ everywhere in the plasma volume and on its boundary.

Writing $R(x,y,\zeta) = \rho(x,y,\zeta) + \Delta R s^2$ and $Z(x,y,\zeta) = \Gamma(x,y,\zeta) + \Delta Z s^2$ with $\Delta R(\zeta) = R_0(\zeta) - R_x(\zeta)$, $\Delta Z(\zeta) = Z_0(\zeta) - Z_x(\zeta)$, the Jacobian can be rewritten as

$$
J_{D^2}(x,y,\zeta) = 2\Delta R \frac{\partial \Gamma}{\partial \theta} - 2\Delta Z \frac{\partial \rho}{\partial \theta} + J_0(s,\zeta) + \tilde{J}(x,y,\zeta)
\tag{9}
$$

with

$$J_0(s, \zeta) = \sum_{n=0}^{2N} J_{0,n}(s) \cos(-nN_P\zeta)$$

$$\tilde{J}(x, y, \zeta) = \sum_{m=1}^{M} \sum_{n=-2N}^{2N} J_{m,n}(s)s^m \cos(m\theta - nN_P\zeta), \tag{10}$$

where $J_{m,n}(s)$ are polynomials of s, the exact form of which is given in Appendix B. These functions only have parametric dependencies on the Fourier harmonics of the boundary $R_{m,n}$ and $Z_{m,n}$ and are determined when the boundary is given. On the other hand, the location of the geometric axis, controlled by $\Delta R = R_0 - R_x$ and $\Delta Z = Z_0 - Z_x$, is free.

As long as the plasma boundary is convex, the choice $R_x = R_0$ and $Z_x = Z_0$ is suitable. This is the case in many applications such as tokamaks without an X-point. In the case of stellarators with a highly shaped plasma boundary, doing so will unfortunately lead to $J_{D^2} = 0$ on a finite set of lines across every poloidal plane. Luckily, by adjusting ΔR and ΔZ, these lines of degeneracy can often be pushed outside the scope of the unit disk. This is how an initial guess of the magnetic axis is obtained in the VMEC code. In VMEC, a search is performed on a grid of (R, Z) on each toroidal plane to find the location of the axis that maximises the minimum of $|J_{D^2}|$.

In this work, we propose a reliable choice of R_x and Z_x to ensure a non-zero "2D Jacobian". One advantage of this choice is that R_x and Z_x can be expressed in simple functions of the Fourier harmonics of the boundary. Thanks to this, we are able to differentiate the axis location with respect to the boundary harmonics for optimisation purposes (where minimisation of a target functional requires gradient information). The method is based on the following idea. At a fixed toroidal angle ζ, the 2D Jacobian is a two-variable function on the unit disk, pictured as a "potato chip" in a 3D surface plot. The average value

$$\langle J_{D^2}\rangle(\zeta) = \frac{1}{\pi} \int_0^1 \int_0^{2\pi} J_{D^2} \; sdsd\theta = 2 \int_0^1 J_0(s, \zeta) \; sds \tag{11}$$

depends only on the boundary data. Ensuring that $\langle J_{D^2}\rangle(\zeta) \neq 0$ is a first consistency check.

The *average orientation* of the potato chip is then controlled by $\langle \vec{\partial} J_{D^2}\rangle(\zeta) =$

$(\langle\partial_x J_{D^2}\rangle, \langle\partial_y J_{D^2}\rangle)$ where

$$\langle\partial_x J_{D^2}\rangle(\zeta) = \left\langle \cos\theta \frac{\partial J_{D^2}}{\partial s} - \frac{\sin\theta}{s}\frac{\partial J_{D^2}}{\partial\theta} \right\rangle$$

$$= \frac{1}{\pi}\int_0^{2\pi} J_{D^2}|_{s=1}\cos\theta d\theta \tag{12}$$

$$\langle\partial_y J_{D^2}\rangle(\zeta) = \left\langle \sin\theta \frac{\partial J_{D^2}}{\partial s} + \frac{\cos\theta}{s}\frac{\partial J_{D^2}}{\partial\theta} \right\rangle$$

$$= \frac{1}{\pi}\int_0^{2\pi} J_{D^2}|_{s=1}\sin\theta d\theta \tag{13}$$

Setting the average orientation to zero ensures that the potato chip is fairly flat, thereby limiting its chances to cross zero over the unit disk. This works provided that the average value exceeds the maximum variations. Fourier decomposing (12) and (13) in the toroidal direction yields a linear system of $4N + 1$ equations for the coefficients of ΔR and ΔZ, and is equivalent to ensuring that the following integrals vanish for all $n \in [-2N, 2N]$

$$\frac{1}{2\pi}\int_0^{2\pi} [\langle\partial_x J_{D^2}\rangle\cos(nN_p\zeta) + \langle\partial_y J_{D^2}\rangle\sin(nN_p\zeta)]d\zeta$$

$$= \frac{1}{2\pi^2}\int_0^{2\pi}\int_0^{2\pi} J_{D^2}|_{s=1}\cos(\theta - nN_p\zeta)d\theta d\zeta = 0. \tag{14}$$

As per expression (8), the resolution in SPEC is restricted to $N + 1$ and N coefficients in R_x and Z_x respectively, meaning that only the $2N + 1$ equations with $n \in [-N, N]$ are considered. After some algebra left in Appendix B.2, we obtain the following linear system

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}\begin{bmatrix} \Delta\vec{R} \\ \Delta\vec{Z} \end{bmatrix} = -\begin{bmatrix} \vec{J}_+ \\ \vec{J}_- \end{bmatrix} \tag{15}$$

where $\Delta\vec{R} = (\Delta R_0, \ldots, \Delta R_N)$ and $\Delta\vec{Z} = (\Delta Z_1, \ldots \Delta Z_N)$, $\vec{J}_+ = (J_{1,0}, \ldots J_{1,N})$, $\vec{J}_- = (J_{1,-1}, \ldots J_{1,-N})$ and

$$\begin{array}{llll} A_{n,k} &= Z_{1,n-k} + \Theta_{N-k-n}Z_{1,k+n}, & n \in [0,N], k \in [0,N] \\ B_{n,k} &= -R_{1,n-k} + \Theta_{N-k-n}R_{1,k+n}, & n \in [0,N], k \in [1,N] \\ C_{n,k} &= Z_{1,k-n} + \Theta_{N-k-n}Z_{1,-n-k}, & n \in [1,N], k \in [0,N] \\ D_{n,k} &= R_{1,k-n} - \Theta_{N-k-n}R_{1,-k-n}, & n \in [1,N], k \in [1,N] \end{array} \tag{16}$$

with $\Theta_i = \begin{cases} 1, & i \geq 0 \\ 0, & i < 0 \end{cases}$. The derivatives of $\Delta\mathbf{X} = (\Delta\vec{R}, \Delta\vec{Z})$ with respect to the boundary parameterisation are computed using this linear system.
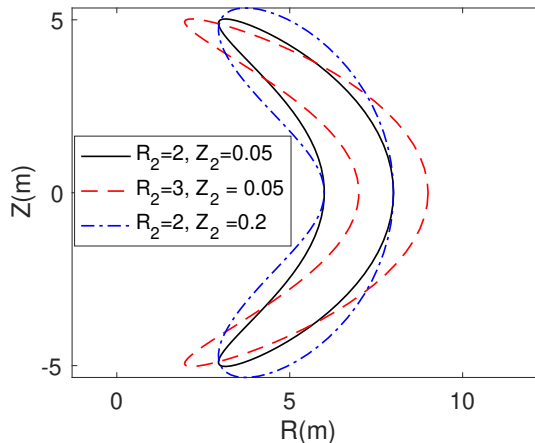
Figure 1. Plasma boundary generated from (17) with $R_0 = 5, R_1 = 1, \kappa = 5$ and different R_2 and Z_2.

To demonstrate the method, we consider a family of plasma boundaries, parameterised by

$$R(\theta) = R_0 + R_1 \cos \theta + R_2 \cos 2\theta,$$
$$Z(\theta) = -\kappa R_1 \sin \theta - \kappa Z_2 \sin 2\theta, \tag{17}$$

in which $\kappa > 0$ is a constant controlling the ellipiticity, and $R_0, R_1, R_2, Z_2 \in \mathbb{R}$ are free parameters. This resembles the $\zeta = 0$ toroidal cut of a stellarator with a bean-shaped plasma boundary, where it is most challenging to determine a good geometric axis and the boundary is dominated by the $m = 1, 2$ Fourier harmonics. figure 1 gives examples of plasma boundaries generated by (17) with $R_0 = 5, R_1 = 1, \kappa = 5$ and different R_2 and Z_2. It is seen that R_2 controls how elongated the "arms" are while Z_2 controls the arms' width.

The Jacobian in this case is given by

$$J_{D_2} = -\kappa R_1^2 - 4\kappa R_2 Z_2 s^2 - 2s\kappa R_1 \cos \theta \left(\Delta R + R_2 + Z_2 \right)$$
$$- 4\kappa s^2 Z_2 \Delta R \cos 2\theta, \quad (18)$$

where $\Delta R = R_0 - R_x$. Note that we have $\Delta Z = 0$ due to assumed stellarator symmetry. The potato chip related to this Jacobian with $R_2 = 2, Z_2 = 0.05$ (black solid curve in figure 1) is plotted in figure 2 (a) with a trivial choice of axis $R_x = R_0$. In this case, the potato chip is so tilted that it crosses the plane $J_{D^2} = 0$ leading to an inconsistent coordinate parameterisation of the toroidal domain. Substituting (18) into (12) yields an average x-orientation (gradient) of $\langle \partial_x J_{D^2} \rangle = -2\kappa R_1 (\Delta R + R_2 + Z_2)$. Setting this orientation to zero, we obtain $\Delta R = -(R_2 + Z_2)$. The corresponding potato chip is shown in figure 2 (b). The chip is now rotated to a horizontal position and no longer has zeroes over the unit disk. If the plasma boundary is more complicated, namely its Fourier content is richer, J_{D^2} is expected to display wilder variations. However, the
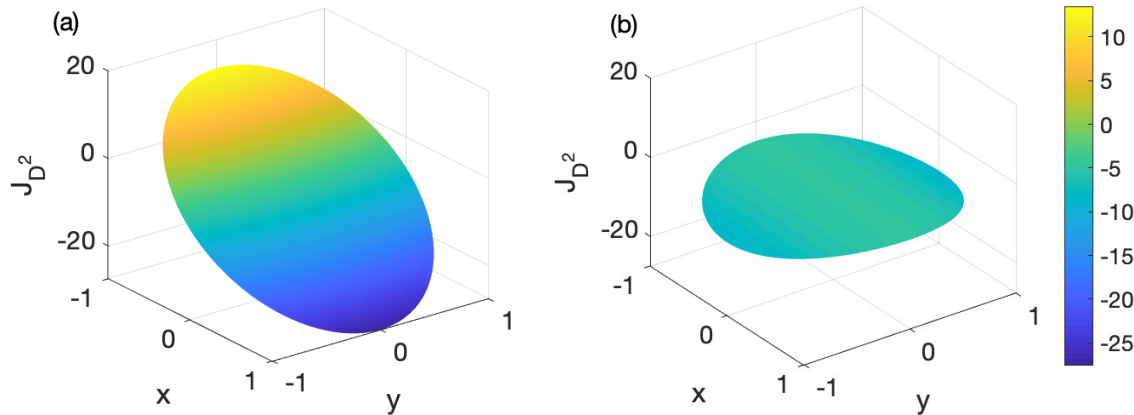
Figure 2. The Jacobian "potato chip" for the bean-shaped plasma boundary, with different choices of the geometric axis R_x. (a) $\Delta R = 0$, i.e. $R_x = R_0$. (b) $\Delta R = -2.05$, i.e. $R_x = R_0 + 2.05$, obtained by eliminating the orientation of the potato chip.

method of "rotating" the potato chip such that it remains flat will work well for most practical applications.

We note that the above method is equivalent to cancelling the $m = 1$ component of the Jacobian on the boundary, which is justified via the following two observations. First, the level surfaces of s=constant determine toroidal shells starting from the boundary at $s = 1$ and shrinking to the axis as s reduces. These toroidal shells become less shaped as s decreases and tend towards an ellipse near the axis. Although not always guaranteed, if $J \neq 0$ everywhere on the boundary, it is less likely to have $J = 0$ somewhere inside the plasma volume because the internal surfaces are less and less shaped. The second observation is that the Fourier harmonics of J have the $m = 1$ harmonics much greater than the $m > 1$ ones and the Jacobian is well approximated by $J_0 + J_1 \cos \theta$. As mentioned in the potato chip argument, the constant term in the Jacobian, J_0, is not affected by the choice of the axis and is therefore fixed, but J_1 is linearly proportional to the free parameters of the axis. If the amplitude of $m = 1$ term, $|J_1|$, is less than the constant term $|J_0|$, then $J_0 + J_1 \cos \theta$ will vanish nowhere. Choosing R_x and Z_x so as to eliminate the $m = 1$ harmonics of J at $s = 1$ is the optimal way to ensure this, since the remaining $m > 1$ harmonics are usually smaller.

We show another application of the method outlined above to choose the geometric axis and parameterise the plasma domain. The design of the National Compact Stellarator Experiment (NCSX) [33] has a highly shaped plasma boundary. The toroidal cut of a NCSX-like plasma volume at $\zeta = 0$ is shown in figure 3. The plasma boundary is non-convex and consequently, the coordinate axis chosen by setting either $R_x = R_0$ or the usual centroid method is outside the plasma boundary as indicated in figure 3. We applied the aforementioned method to choose an axis that eliminates the $m = 1$ harmonics of J on the plasma boundary. This method leads to geometric axis right in the middle of the plasma volume, close to the magnetic axis predicted by a VMEC vacuum calculation, though our method is based entirely on geometric arguments and

knows nothing about MHD equilibrium. We overplot the contour of the Jacobian J in figure 3, showing non-zero value within the volume.
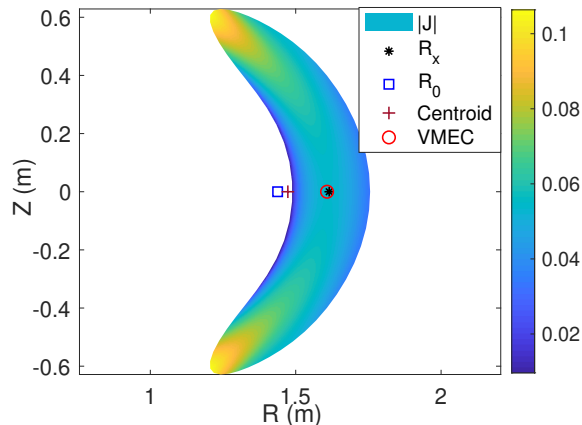

Figure 3. The Jacobian J_{D^2} in a NCSX-like plasma volume at $\zeta = 0$, using the coordinate axis designed to eliminate $m = 1$ Fourier harmonics of J on the plasma boundary. The corresponding coordinate axis R_x is overplotted (black star), along with R_0 of the boundary (blue square), the centroid of the boundary (brown plus) and the magnetic axis computed by VMEC assuming vacuum (red circle).

To demonstrate the robustness of our method, we test it against the family of boundaries in (17). We find that our method guarantees $J \neq 0$ for all the allowed parameter space except some extreme cases. The detail is shown in Appendix A.

3. Parity and boundary conditions on the coordinate axis

3.1. Constraints on the coefficients of the Fourier expansions of physical quantities

If we write a physical scalar f in Fourier harmonics of θ and ζ as

$$f(s, \theta, \zeta) = \sum_{m,n} f_{m,n}(s) \cos(m\theta - nN_p\zeta), \tag{19}$$

then $f_{m,n}(s)$ cannot be arbitrarily chosen if f is analytic in Euclidean space. It must follow the parity constraint and have certain asymptotic behaviour as $s \to 0$. These constraints are given by [31]

$$f_{m,n}(-s) = (-1)^m f_{m,n}(s), \tag{20}$$

$$\lim_{s \to 0} \frac{f_{m,n}(s)}{s^m} < \infty. \tag{21}$$

The parity requirement (20) guarantees that f takes the same value under the transformation $s \to -s$ and $\theta \to \theta + \pi$, as $x = s\cos\theta = -s\cos(\theta + \pi)$ and $y = s\sin\theta = -s\sin(\theta + \pi)$ refers to the same point in Cartesian coordinates. The asymptotic requirement (21) guarantees that f is analytic in Cartesian coordinates

(x, y), i.e. it can be written into converged Taylor series in (x, y) and is thus smooth. These results were derived on a unit disk by Boyd and Yu [31], but they can be trivially extended to a toroid. As a corollary, $f_{m,n}(s)$ can be expanded into polynomials as

$$f_{m,n}(s) = s^m \sum_{l \geq 0} f_{m,n,2l} s^{2l}. \tag{22}$$

Similar constraints apply to the magnetic vector potential $\mathbf{A} = A_s \nabla s + A_\theta \nabla \theta + A_\zeta \nabla \zeta$. Following Lewis and Bellan [34], we first write $\mathbf{A} = A_x \nabla x + A_y \nabla y + A_\zeta \nabla \zeta$, in which A_x, A_y and A_ζ follows the requirements for a scalar (22). Then we expand ∇x and ∇y into ∇s and $\nabla \theta$. After some algebra, we obtain that

$$A_s = \sum_{m \geq 0, n} [\lambda_{m,n} s^{m-1} + s h_{s,m,n}(s)] \sin(m\theta - nN_p\zeta), \tag{23}$$

$$A_\theta = \sum_{m \geq 0, n} [\lambda_{m,n} s^m + s^2 h_{\theta,m,n}(s)] \cos(m\theta - nN_p\zeta), \tag{24}$$

$$A_\zeta = \sum_{m \geq 0, n} h_{\zeta,m,n}(s) \cos(m\theta - nN_p\zeta), \tag{25}$$

with $\lambda_{0,n} = 0$, while $h_{s,m,n}(s)$, $h_{\theta,m,n}(s)$ and $h_{\zeta,m,n}(s)$ follow (22). We introduce a gauge $g(s, \theta, \zeta) = \sum g_{m,n}(s) \sin(m\theta - nN_p\zeta)$, where

$$g_{m,n}(s) = \frac{\lambda_{m,n}}{m} s^m + \int_0^s s' h_{s,m,n}(s') ds'. \tag{26}$$

The vector potential $\hat{\mathbf{A}}$ after the gauge transformation $\hat{\mathbf{A}} = \mathbf{A} - \nabla g$ will have $\hat{A}_s = 0$ and will not contain the $\lambda_{m,n}$ term, i.e.

$$\hat{A}_\theta = \sum_{m \geq 0, n} s^2 \hat{h}_{\theta,m,n}(s) \cos(m\theta - nN_p\zeta), \tag{27}$$

$$\hat{A}_\zeta = \sum_{m \geq 0, n} \hat{h}_{\zeta,m,n}(s) \cos(m\theta - nN_p\zeta), \tag{28}$$

where $\hat{h}_{\theta,m,n}(s)$ and $\hat{h}_{\zeta,m,n}(s)$ follow (22). One last gauge freedom remains. That is, one can choose a new gauge $\hat{g}(\zeta)$ depending only on ζ, with $\mathbf{A}$ after gauge transformation still conforming (27) and (28). We therefore enforce that

$$\hat{h}_{\zeta,0,0}(1) = \hat{h}_{\zeta,0,n \geq 1}(0) = 0, \tag{29}$$

to eliminate this gauge freedom. For simplicity, we will drop the hat on $\hat{\mathbf{A}}$ from now on.

3.2. The Zernike polynomials and their recombination

The conditions (20) and (21) are known as *behavioural boundary conditions* [35]. In other words, they do not enforce the solution or its derivatives to any specific value but instead restrict the class of functions that the solution can take. The behavioural boundary conditions are usually treated in two different ways when the corresponding PDE is solved numerically using a spectral method. The first way is to choose a set

of basis functions that satisfy these conditions from the very beginning. For example, one chooses the Fourier basis for a periodic boundary condition. The second way is to choose a more general set of basis functions that does not satisfy the behavioural boundary condition, but rely on the properties of the PDE so that the numerical solution satisfy these conditions automatically [31]. For example, one can still use the parity-restricted Chebyshev polynomials to solve the Lapalce equation on a unit disk, although the Chebyshev polynomials do not satisfy the asymptotic condition (21) [31].

In this work, we follow the first path and use the Zernike polynomials [36, 37] (one-sided Jacobi polynomials) as our basis functions. The Zernike polynomials are orthogonal basis functions on the unit disk. They have the advantage of displaying spectral convergence (the error decays exponentially in the radial resolution L) when solving the Laplace equation on the unit disk [31], while the usual Fourier-Bessel series converge only algebraically (polynomial order in L). Furthermore, the Zernike basis only needs half the number of degrees of freedom used by the parity-restricted Chebyshev polynomials to achieve comparable accuracy. The Zernike polynomials take the form

$$
\begin{aligned}
Z_l^{-m}(s, \theta) &= R_l^m(s) \sin m\theta, \\
Z_l^m(s, \theta) &= R_l^m(s) \cos m\theta,
\end{aligned}
\tag{30}
$$

where $R_l^m(s)$ is a l-th order polynomial given by

$$
R_l^m(s) = \sum_{k=0}^{\frac{l-m}{2}} \frac{(-1)^k (l-k)!}{k! \left[\frac{1}{2}(l+m) - k\right]! \left[\frac{1}{2}(l-m) - k\right]!} s^{l-2k},
\tag{31}
$$

and is only non-zero for $l \geq m$ and even $l - m$. The radial basis $R_l^m(s)$ is a polynomial of s with lowest order m and therefore the Zernike basis functions satisfy (22).

The unknowns, namely the components of the vector potential, are discretised in terms of the Zernike polynomials. The on-axis boundary condition for A_ζ in (28) is automatically satisfied by the Zernike polynomials. But the condition for A_θ in (27) is not since $A_{\theta,m,n} \sim s^{m+2}$ rather than s^m at the same time as the Zernike polynomials scales as s^m near the axis. Therefore, additional treatments are needed to enforce the on-axis boundary condition for A_θ. For $m = 0$ and $m = 1$, a basis recombination method is used by defining new radial basis functions as

$$
\begin{aligned}
\hat{R}_0^0 &= 1, \quad \hat{R}_l^0 = \frac{1}{l+1} R_l^0 - \frac{(-1)^{l/2}}{l+1}, \\
\hat{R}_1^1 &= s, \quad \hat{R}_l^1 = \frac{1}{l+1} R_l^1 - \frac{(-1)^{(l-1)/2}}{2} s.
\end{aligned}
\tag{32}
$$

so that the basis scales as s^{m+2} except for $\hat{R}_0^0$ and $\hat{R}_1^1$, which are excluded from the representation of $A_{\theta,m,n}$. For $m \geq 2$, the radial basis functions are only rescaled as

$$
\hat{R}_l^m = \frac{1}{l+1} R_l^m.
\tag{33}
$$

Lagrange multipliers are introduced for the $m \geq 2$ components to ensure that the s^m terms vanish.

Now, we can write the components of the vector potential as

$$A_\theta = \sum_{m \geq 0, n} \sum_{l=m}^{L} A_{\theta,m,n,l} \hat{R}_l^m(s) \cos(m\theta - nN_p\zeta), \tag{34}$$

$$A_\zeta = \sum_{m \geq 0, n} \sum_{l=m}^{L} A_{\zeta,m,n,l} \hat{R}_l^m(s) \cos(m\theta - nN_p\zeta), \tag{35}$$

where the unknowns are $A_{\theta,m,n,l}$ and $A_{\zeta,m,n,l}$, and L is the maximum order of polynomial in s. We let $A_{\theta,0,n,0} = A_{\theta,1,n,1} = 0$, while keeping in mind that an additional Lagrange multiplier is introduced later to force the m-th harmonic of A_θ to scale as s^{m+2} for $s \to 0$. The reason for implementing a mixture of basis recombination, rescaling and Lagrange multiplier method is to ensure the condition number of the discretised problem only increases mildly as L and the poloidal Fourier resolution M increases.

4. Implementation in the Stepped Pressure Equilibrium Code (SPEC)

4.1. The SPEC energy functional

Our Beltrami solver improves and extends the original Beltrami solver of the Stepped Pressure Equilibrium Code (SPEC). SPEC defines the energy functional F in the toroidal volume as

$$\begin{aligned}
F = \frac{1}{2} \int_\Omega |\nabla \times \mathbf{A}|^2 dV &- \frac{\mu}{2} \left[\int_\Omega \mathbf{A} \cdot \nabla \times \mathbf{A} dV - K_0 \right] \\
&+ \sum_{m \geq 2, n} a_{m,n} \left[\lim_{s \to 0} A_{\theta,m,n}(s)/s^m \right] \\
&+ \sum_n b_n \left[A_{\zeta,0,n}(s = s_n) \right] \\
&+ d_1 \left[\oint_{C_p} \mathbf{A} \cdot d\mathbf{l} - \Delta\psi_t \right] \\
&+ \sum_{m,n} e_{m,n} \iint_{\partial\Omega} \nabla \times \mathbf{A} \cdot \mathbf{n} \cos(m\theta - nN_p\zeta) dS, \tag{36}
\end{aligned}$$

where K_0 is the target helicity, $s_0 = 1$, $s_{n>0} = 0$, $\Delta\psi_t$ the toroidal flux, and C_p is a *poloidal* loop on $\partial\Omega$'s short way around the torus. We have introduced Lagrange multipliers $a_{m,n}$, b_n, d_1 and $e_{m,n}$ to ensure various boundary conditions. They correspond to the regularity condition (27), the gauge condition (29), the total toroidal flux constraint and the ideal boundary condition $\mathbf{B} \cdot \mathbf{n} = 0$, respectively. The free variables are $\mathbf{a} = \{A_{\theta,m,n,l}, A_{\zeta,m,n,l}, a_{m,n}, b_n, e_{m,n}, d_1\}$. Substituting the discretisation (34) and (35) into (36), F can be rewritten into matrix-vector products as

$$F = \frac{1}{2} \mathbf{a}^{\mathrm{T}} \cdot \mathcal{A} \cdot \mathbf{a} - \frac{\mu}{2} (\mathbf{a}^{\mathrm{T}} \cdot \mathcal{B} \cdot \mathbf{a} - K_o) - \mathbf{a}^{\mathrm{T}} \cdot \mathbf{b}, \tag{37}$$

where $\mathbf{b} = \{\mathbf{0}, \Delta\psi_t\}$ and the matrices $\mathcal{A}$ and $\mathcal{B}$ are constructed by completing the integral in (36) using a mixture of Gaussian quadrature and FFT. This is equivalent

to the Galerkin method. The evaluation of radial quadrature integrand at each radius is independent to each other and therefore can be parallelised. Alternatively, since the matrix elements in $\mathcal{A}$ are independent to each other, we can assign one CPU to compute each element. We remark that these are parallelised using a shared memory scheme, in addition to the original SPEC parallelisation over the number of volumes with distributed memory.

The solution $\mathbf{a}$ is a stationary point of the energy functional F. In the case where μ is known, setting the derivatives of F to zero gives

$$(\mathcal{A} - \mu\mathcal{B}) \cdot \mathbf{a} = \mathbf{b}. \tag{38}$$

If μ is not the eigenvalue of the generalised eigenvalue problem $\mathcal{A} \cdot \mathbf{a} = \mu\mathcal{B} \cdot \mathbf{a}$, then $\mathbf{a}$ can be obtained by solving the linear equation (38). In the case μ is unknown but K_0 is known or μ is the eigenvalue, μ is solved together with $\mathbf{a}$ by locating a stationary point of (37) using a Newton's method. We will focus on the first case in which μ is given, since it is more commonly used in practice. For example, if instead the rotational transform ι is prescribed on the boundary, μ will be adjusted iteratively, with (38) solved each time after the adjustment to estimate and match ι on the boundary.

The matrix $\hat{\mathcal{A}} = \mathcal{A} - \mu\mathcal{B}$ is dense and indefinite. The linear equation (38) can be solved either by a direct method such as LU decomposition, or by an iterative method such as the generalised minimal residual method (GMRES) [38]. The direct solver is fast and accurate in low resolution. But as the time complexity of the direct method scales as $O(M^3N^3L^3)$, where L, M and N are the radial, poloidal and toroidal resolution, respectively, the iterative solver is preferred at high resolution ($M, N, L \geq 15$) due to its lower time cost. In the next section, we will focus on solving the linear equation using GMRES.

Once the Beltrami field inside each volume is solved, SPEC also iteratively find a global force balance by adjusting the position of the interface geometry to satisfy (3) on each interface. The procedure is detailed in Hudson *et al* [12, 27] and will not be repeated here.

4.2. The iterative solver and matrix-free method

In this section, we will introduce the iterative matrix solver, GMRES, and its matrix-free form that is newly included in SPEC. GMRES iteratively looks for $\mathbf{a}_n$ that minimises the residual $\epsilon_{\text{GMRES}} = \|\hat{\mathcal{A}} \cdot \mathbf{a}_n - \mathbf{b}\|$, where $\|.\|$ is the Euclidean norm. Instead of solving the original problem which is usually ill-conditioned, a left preconditioner matrix $\mathcal{M}$ is applied on both side of (38) so that the transformed problem is well conditioned. The convergence speed of (the preconditioned) GMRES depends highly on the quality of $\mathcal{M}$. A good preconditioner will require the matrix product $\mathcal{M}^{-1}\hat{\mathcal{A}}$ to be as close as possible to an identity matrix. Also, inverting the preconditioner $\mathcal{M}$ should be considerably cheaper than inverting $\hat{\mathcal{A}}$ itself.

If the i-th and j-th unknowns in $\mathbf{a}$ correspond to A_{θ,m_i,n_i,l_i} and A_{θ,m_j,n_j,l_j}, respectively, then the matrix element $\hat{\mathcal{A}}_{i,j}$ describes the coupling strength between

harmonics (m_i, n_i) and (m_j, n_j). Noting that if the Fourier series of the boundary $R_{m,n}$ and $Z_{m,n}$ have spectral convergence, then the coupling terms between A_{θ,m_i,n_i,l_i} and A_{θ,m_j,n_j,l_j}, formed by the $(|m_i - m_j|, |n_i - n_j|)$ harmonics of the coordinate metrics, should also decay exponentially with $|m_i - m_j|$ and $|n_i - n_j|$ and are thus small compared to the "diagonals" $m_i = m_j$ and $n_i = n_j$. Therefore, we can construct $\mathcal{M}$ from the elements of $\hat{\mathcal{A}}$ by eliminating all the coupling terms with $m_i \neq m_j$ or $n_i \neq n_j$, and keeping the rest ("diagonals" and terms related to Lagrange mulitpliers). Physically, the matrix $\mathcal{M}$ is equivalent to the $\hat{\mathcal{A}}$ matrix of a tokamak with similar major radius and minor radius to the stellarator we are solving for. The preconditioning matrix $\mathcal{M}$ is sparse, with the number of nonzero elements $\sim O(MNL^2)$, while the total number of elements in $\mathcal{M}$ is $O(M^2N^2L^2)$. After the construction of $\mathcal{M}$, the approximate inverse $\mathcal{M}$ is computed by an incomplete LU factorisation [39]. In our numerical experiments, the absolute residual ϵ_{GMRES} usually decreases to 10^{-9} with 50 iterations and 10^{-12} with 80 iterations.

As $\hat{\mathcal{A}}$ is dense, the memory needed to store it scales as $O(M^2N^2L^2)$. The time complexity of constructing it also scales as $O(M^2N^2L^3)$ taking into account the radial Gaussian quadrature, making it costly both in memory and time if the resolution is high. It turns out that at each iteration step of GMRES, we only need the matrix-vector product of $\hat{\mathcal{A}}$ and a trial vector $\mathbf{a}_n$, rather than the matrix itself. It is therefore possible to construct directly $\hat{\mathcal{A}} \cdot \mathbf{a}_n$ without computing each element of $\hat{\mathcal{A}}$. This is known as the *matrix-free method*. We have also implemented the matrix-free GMRES in SPEC. Excluding the preconditioner, the memory and time complexity of the matrix-free method are $O(MNL)$ and $O(N_{\text{GMRES}}L^2MN \log MN)$, respectively, with N_{GMRES} the number of GMRES iterations.

The only caveat with the matrix-free method is that it does not improve the time cost of solving the force balance at this stage. To overcome this drawback, we plan to formulate the force balance problem into a least square form and uses an adjoint-based method (see for example Paul *et al* [40] and references therein). This will be reported in future work.

5. Numerical experiments, verification and examples

5.1. Convergence and benchmark

We conduct our numerical experiments using the boundary of Wendelstein 7-X OP1.1 vacuum field [14, 41] with single-volume SPEC running as a Beltrami solver. A plot of the boundary shape and $|\mathbf{B}|$ on the surface is shown in figure 4, while the Poincaré section of four different toroidal angles are plotted in figure 5. Note that the toroidal flux within the plasma volume is normalized to unity. In SPEC, the averaged Beltrami field error, E_α, is defined as

$$E_\alpha = \frac{1}{V} \int_\Omega |(\nabla \times \mathbf{B} - \mu \mathbf{B}) \cdot \nabla \alpha| dV, \tag{39}$$

where $\alpha = \{s, \theta, \zeta\}$ is the coordinate of interest and V the volume size. Figure 6 shows E_α as a function of the Fourier and radial resolution, for the original SPEC radial basis functions (modified Chebyshev, in dashed red) and the new Zernike basis function (in solid blue). SPEC's previously reported ill-conditioning problem requires the radial resolution to be reduced when it emerges. As a consequence, the error stops converging as the Fourier resolution increases beyond $M = N = 21$, as shown in figure 6. On the other hand, the new Zernike method successfully mitigates the ill-conditioning problem with uninterrupted error convergence, as seen in figure 6. The error decreases exponentially as the resolution increases, matching the expectation of spectral convergence. The ill-conditioning problem is suppressed for Fourier resolutions as high as $M = N = 35$. Such a resolution is sufficient for most applications.
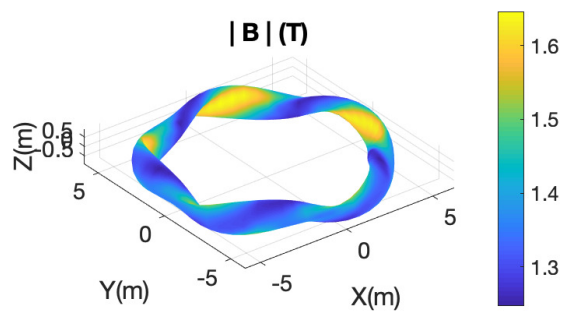
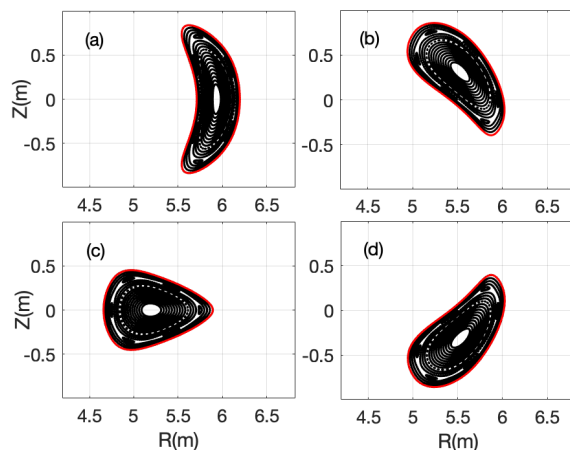**Figure 4.** The W7X OP1.1 boundary we used and the magnitude of the vacuum field on the plasma boundary.

Figure 5. The Poincaré section of the W7X OP1.1 case we used at different toroidal angles: (a) $\zeta = 0$, (b) $\zeta = \pi/10$, (c) $\zeta = 2\pi/10$ and (d) $\zeta = 3\pi/10$.

In figure 7, we also compare $\mathbf{B}$ solved by SPEC and $\mathbf{B}$ computed by the BIEST code. The maximum point-wise error is defined as

$$e_{\max} = \frac{\|\mathbf{B} - \mathbf{B}_{\text{ref}}\|_\infty}{\|\mathbf{B}_{\text{ref}}\|_\infty}, \tag{40}$$

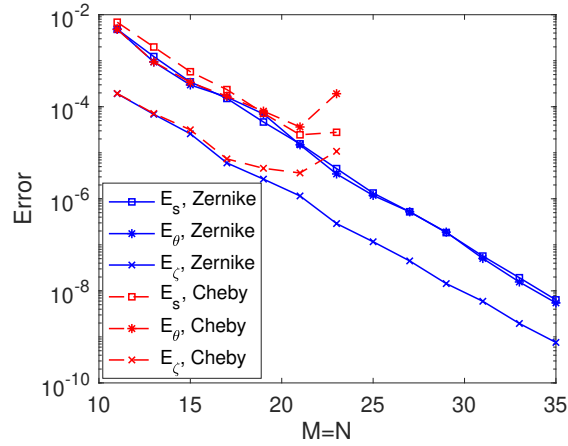**Figure 6.** The volume averaged Beltrami error E_α as a function of SPEC resolution, with the Zernike basis and the original modified Chebyshev basis, for $\mu = 1$ Beltrami field with W7X boundary. We have $L = M + 4$ in the Zernike case and $L = L_{\max}$ in the Chebyshev case, with $L_{\max}$ the maximum allowed L without triggering the ill-conditioning error.

where $\mathbf{B}_{\mathrm{ref}}$ is the reference solution, and the infinity norm $\|\mathbf{x}\|_\infty$ takes the maximum value of $\mathbf{x}$ among its three Euclidean components and the maximum of every point on a grid in Ω. Here, the reference solution is taken to be the BIEST magnetic field on the plasma boundary computed with a relative error of 10^{-9}. Again, we find a spectral convergence rate with the Zernike basis up to the precision of the reference solution. The convergence study and benchmark shows that the ill-conditioning problem no longer appears in high resolution SPEC calculations.
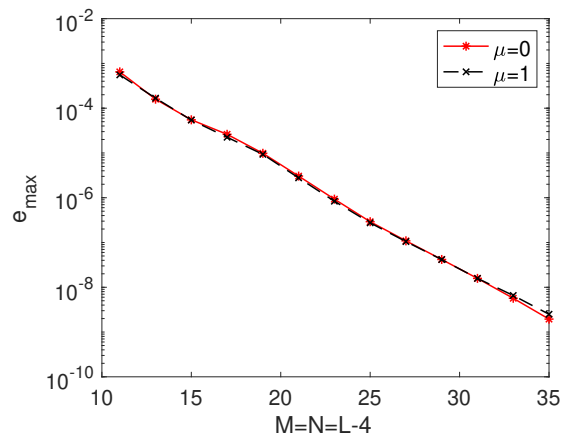

Figure 7. The maximum magnetic field error between SPEC and a reference BIEST run as a function of SPEC resolution for the vacuum solution $\mu = 0$ and the Taylor state $\mu = 1$. For the reference BIEST calculation, we set BIEST parameters $N = 448 \times 2240, 392 \times 1960$, $\epsilon_{\mathrm{GMRES}} = 10^{-12}, 10^{-11}$ for $\mu = 0, 1$, respectively. These parameters are set so that the reference BIEST magnetic field has 10^{-9} relative error. For SPEC, we use matrix-free method.

In Table 1, we present the computation time for the three different matrix solving methods implemented in SPEC: the direct method, GMRES with matrix and GMRES matrix-free. The first two methods are similar in speed, mostly due to the same amount of the time spent constructing the matrix. On the other hand, the matrix-free method is orders of magnitude faster and scales only moderately with increasing resolution. Even when the slower non-matrix-free methods are used, SPEC only spends a fraction of the time used by BIEST for computing the Taylor state, despite the fact that BIEST uses 6 times as many CPUs.

SPEC					BIEST			
$M \times N \times L$	$e_{\max}$	direct(s)	iter-w-mat(s)	matrix-free(s)	N	ϵ_{GMRES}	$e_{\max}$	t_{solve}(s)
$11 \times 11 \times 15$	5.5×10^{-4}	1.36	1.42	0.33	7.9×10^4	1×10^{-4}	2.3×10^{-4}	316.9
$17 \times 17 \times 21$	2.2×10^{-5}	16.79	17.09	1.90	1.4×10^5	3×10^{-6}	2.7×10^{-5}	1322
$23 \times 23 \times 27$	8.4×10^{-7}	107.8	103.2	5.30	2.5×10^5	1×10^{-7}	9.0×10^{-7}	3722
$29 \times 29 \times 33$	4.2×10^{-8}	604.5	544.3	11.96	5.2×10^5	1×10^{-9}	3.0×10^{-8}	17473
$35 \times 35 \times 39$	2.5×10^{-9}	-	-	16.65	7.7×10^5	1×10^{-10}	1.7×10^{-9}	41512

Table 1. The time consumption measured in seconds for solving the Taylor state $\mu = 1$ with W7X boundary, using the SPEC direct solver with matrix (direct), GMRES with matrix (iter-w-mat) and matrix-free GMRES, and comparing to BIEST with a similar accuracy. We set $\epsilon_{\text{GMRES}} = 10^{-12}$ for SPEC. The SPEC computation was performed on 8×Intel Cascade Lake CPU@3.2GHz, while BIEST uses 48 such CPUs. We did not provide results for $M = 35$ cases with matrix because the number of elements exceeds the upper limit of a 32 bit integer.

5.2. Comparison to the Dommaschk potential, high resolution

As mentioned in Section 4, our Beltrami solver is an integrated part of the SPEC code. Limited by the ill-conditioning problem, previous benchmarks of SPEC multi-volume equilibria cannot proceed to very high Fourier resolution. In this section, we will demonstrate that our new Beltrami solver significantly improves the overall convergence of SPEC multi-volume equilibria and enables more detailed physics studies. We will repeat the free-boundary SPEC comparison [27] with the analytical Dommaschk potential [42]. The difference in the magnetic field between SPEC and the analytical solution was on the order of 10^{-5} for $M = N = 12$, the highest resolution without triggering the ill-conditioning problem. This seemed to be a good match at first glance and it was hoped that the difference would continue to decrease as the Fourier resolution increases. But we will show that this is not the case. There are unresolved problems in this comparison and interesting physics that can only be discovered at a much higher resolution enabled by our improved Beltrami solver.

The Dommaschk potential is given analytically by

$$\Phi(R, \varphi, Z) = \eta\varphi + \Phi_{5,2}(R, \varphi, Z), \tag{41}$$

where $\eta = 1$ is a constant and $\Phi_{5,2}$ a harmonic field (see Appendix 1 of Hudson *et al* [27]). The magnetic field $\mathbf{B} = \nabla\Phi$ is a vacuum field with $\nabla \times \mathbf{B} = 0$ and represents a classical

stellarator vacuum field with 5 field periods ($N_p = 5$) and the plasma cross section being a rotating ellipse. The Poincaré section of the magnetic field at $\zeta = 0$ is given in figure 8. In Hudson *et al* , SPEC was set up with a circular computational boundary
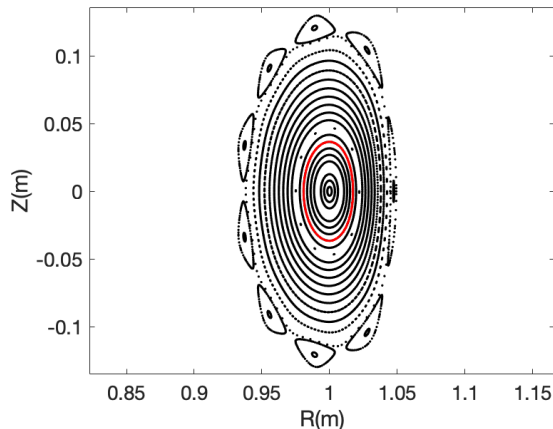

Figure 8. The Poincaré section of the Dommaschk magnetic field at $\zeta = 0$. The red curve shows the plasma boundary.

$R_w = 1 + 0.2 \cos\theta$ and $Z_w = -0.2 \sin\theta$. The normal vacuum field on the computational boundary, $D_V = \mathbf{B} \cdot (\mathbf{e}_\theta \times \mathbf{e}_\zeta)$ and the linking currents on the computational boundary, $I = \int_0^{2\pi} \mathbf{B} \cdot \mathbf{e}_\theta d\theta = 0$ and $G = \int_0^{2\pi} \mathbf{B} \cdot \mathbf{e}_\zeta d\zeta = 2\pi\eta$ are computed from the analytical solution and given to SPEC as inputs, where $\mathbf{e}_\theta$ and $\mathbf{e}_\zeta$ are the covariant basis vectors. The computational domain was separated into a "plasma volume" (with $\mu = 0$, in fact has vacuum field inside) and a vacuum volume, with an ideal interface between them. We shall call this interface the plasma boundary hereafter. The toroidal flux $\Delta\psi_t$ inside the plasma boundary was chosen to be 0.003Wb, with the total toroidal flux inside the computational boundary at $\zeta = 0$ being 0.1257Wb. SPEC then iterates on the position of the plasma boundary to reach a force balance (3). After the force balance is reached, the solution from SPEC should match the analytical Dommaschk magnetic field up to an error controlled by the numerical resolution. It was reported that because of the ill-conditioning problem, the Fourier resolution cannot go beyond $M = N = 12$, and the maximum Euclidean error $e_{\max}$ defined in (40), taking the analytical solution as $\mathbf{B}_{\text{ref}}$ and computed at the grid points inside the computational boundary, can only reach as low as 10^{-5}. In this section, we will show that since the ill-conditioning problem is resolved, $e_{\max}$ continues to reduce spectrally with increased Fourier resolution.

Extra care should be taken to choose the plasma boundary and the choice is not arbitrary. We need to ensure that such a surface exists as a flux surface in the Dommaschk magnetic field to which we compare. Failing to do so will result in a current sheet on the plasma boundary and a jump of ι to shield the magnetic island that was originally there [4], leading to a saturated $e_{\max}$ even with increasing numerical resolution. Given $\mathbf{B}$ from the Dommaschk potential, one will need to trace the field lines to find a flux surface, compute the enclosed toroidal flux, and giving it to SPEC

as an input. Numerical errors can build up and propagate during the process, leading to poor convergence.

In this work, we follow a slightly different pathway. Instead of setting η in (41) to unity, we consider it as an adjustable parameter that only affects the toroidal field B_ζ. The change of η will not modify D_V, since the computational boundary is axisymmetic and the projection of B_ζ on it is zero. For a vacuum field, the linking current I will stay at zero, but G will be affected through $G = 2\pi\eta$. Therefore, we still give D_V and I to SPEC as inputs, but instead of constraining G, we constrain ι on the plasma boundary on the vacuum side to $\iota_b \approx 0.4966719044$, a noble irrational number with continuous fraction expansion $\langle 0, 2, 74, 1, 1, \cdots \rangle$, in which

$$\langle a_0, a_1, a_2, a_3, \cdots \rangle = a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \ldots}}}. \tag{42}$$

A noble irrational, whose continued fraction expansion has an infinite sequence of 1's on its tail, is chosen because the flux surfaces with a noble rotational transform are more resilient to chaos inducing perturbations and more likely to persist [43]. The existence of such a flux surface as an interface is later verified by our convergence study. We set $\psi_t = 0.0021\text{Wb}$, a fixed value. SPEC then iterates on the position of the plasma boundary to find force balance. The procedure is equivalent to searching for a value of η such that the plasma boundary is a flux surface with $\iota = \iota_b$. Once the force balance is achieved, G is computed from the solution, which is then used in (41) to construct the reference Dommaschk field given $\eta = G/(2\pi)$. Figure 9 shows that $e_{\max}$ continue to reduce beyond $M = N = 12$ as the ill-conditioning problem is solved, and reaches as low as 10^{-8} for $M = N = 24$.
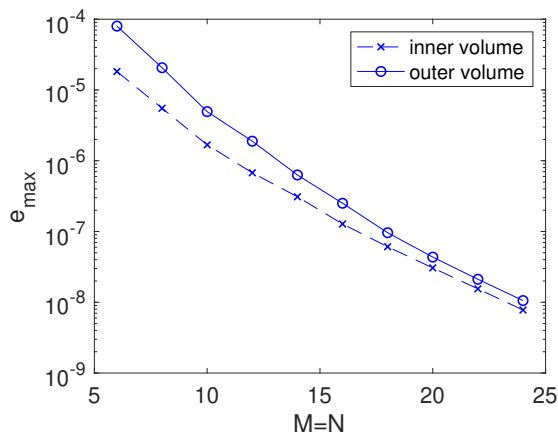

Figure 9. The max Euclidean error $e_{\max}$ between the analytical Dommaschk field and SPEC solution, as a function of Fourier resolution. The radial resolution L is set to sufficiently high until it does not affect the error.

The improved resolution allows us to resolve current sheets on interfaces even with a small amplitude. We take the Dommaschk potential case as an example. As mentioned above, if the plasma boundary is not a flux surface in the Dommaschk vacuum field, a current sheet will develop on it to shield out islands that were originally there. The amplitude of this current sheet depends on the size of n and m if the rotational transform on the surface is written into the form of $\iota = n/m$, where $m, n \in \mathbb{N}$ and coprime. If we set $\eta = 1$ and keep $\psi_t = 0.0021$Wb, we can observe such a current sheet on the plasma boundary. In figure 10, we plotted the discontinuity in B_Z on the plasma boundary as a function of two angles. The magnitude of $[[B_Z]]$ is on the order of 10^{-7}, and as a consequence, the Fourier resolution to resolve it must be high enough such that $e_{\max} < 10^{-7}$. The rotational transform ι jumped from 0.4966703277953 to 0.4966701219709 on the plasma boundary, with their continuous fraction expansion being $\langle 0, 2, 74, 1, 1, 2, 1, 1, 7, 2, 2, 1, 2, 1, 53 \rangle$ and $\langle 0, 2, 74, 1, 1, 2, 1, 2, 2, 10, 31, 1, 14, 1, 1 \rangle$, respectively. It is noteworthy that although the rotation transformation here agrees with the irrational ι_b mentioned above up to the first five digits, it still results in a non-vanishing current sheet. This demonstrates the nature of the mathematical problem and has strong implications for the class of solutions with assumed nested flux surfaces everywhere. In fact, it can be conjectured that since on a computer all numbers have a finite number of digits, they end up being rational leading to a corresponding current sheet. However, the size of the current sheet reduces as we are approaching irrationals with more digits. The current sheet will eventually be buried in the numerical noise and be impossible to resolve due to insufficient numerical resolution. This motivates future studies of the current sheet amplitude as we approach an irrational rotational transform with higher and higher number of digits.
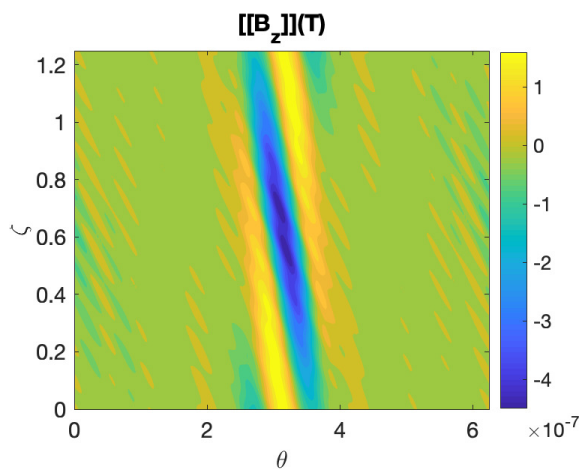

Figure 10. The jump of B_Z on the plasma boundary for $\eta = 1$ and $\psi_t = 0.0021$Wb. The jump indicates the existence of a current sheet on the plasma boundary and such a boundary is not a flux surface in the vacuum field.

6. Conclusion

In this work, we presented a numerical method to solve the Beltrami equation in a shaped torus using a spectral method, based on the Stepped Pressure Equilibrium Code (SPEC). With the coordinate interpolation and ill-conditioning problem properly addressed, the Beltrami field can be solved to high accuracy in shaped stellarator geometry such as the W7X. The speed was improved by the use of iterative matrix solvers and the matrix-free method. We showed that the new Beltrami solver also improved the overall accuracy of SPEC and it is now able to resolve very fine current sheets. The coordinate parameterisation and on-axis boundary condition introduced in this paper can serve as a reference point for future stellarator codes that utilizes toroidal coordinates. This is also the first time the Zernike polynomials are used in a plasma equilibrium code.

To fully explore the benefit of the matrix-free method, we propose to implement an adjoint-based optimisation algorithm to solve the force balance. This, along with other performance improvements, will enable free-boundary SPEC to operate as a core equilibrium solver to be frequently called within the iterations of reconstruction and/or stellarator optimisation tools. Moreover, the current sheets formed due to rational rotational transform on interfaces can now be resolved at very high resolution, enabling future studies into the mathematical and physical nature of 3D MHD equilibria. The improvement in numerical convergence and speed also enables more dedicated physics studies such as multi-volume W7X equilibrium or tokamak/reversed-field pinches sawtooth crashes [17]. A properly designed comparison with other equilibrium codes is also planned for future works.

Appendix A. Robustness of the choice of the geometry axis

Without loss of generality, in (17) we set $R_1 = 1$ and the ellipticity parameter $\kappa = 1$ since it only scales J by a constant factor. The parameters R_2 and Z_2 should stay within a range so the the boundary will not intersect with itself. This range is derived by setting $\partial_\theta R = \partial_\theta Z = 0$, giving

$$1 + R_2 \frac{-1 + \sqrt{1 + 32Z_2^2}}{2Z_2} > 0 \quad \text{and} \quad Z_2^2 < \frac{1}{4}. \tag{A.1}$$

We have plotted this region in $R_2 - Z_2$ parameter space as shown in figure A1.

We choose $R_x = R_0 - R_2 - Z_2$, so that

$$J_{D^2} = -1 - 4s^2 R_2 Z_2 + 4s^2 Z_2 (R_2 + Z_2) \cos 2\theta, \tag{A.2}$$

with no $m = 1$ harmonics. To satisfy the requirement that $|J_{D_2}| \neq 0$, we will need $|1 + 4s^2 R_2 Z_2| > |4s^2 Z_2 (R_2 + Z_2)|$ for $s \in [0,1]$, giving another region in figure A1. Inspection of figure A1 shows that the method works for nearly all the allowed choices of R_2 and Z_2.
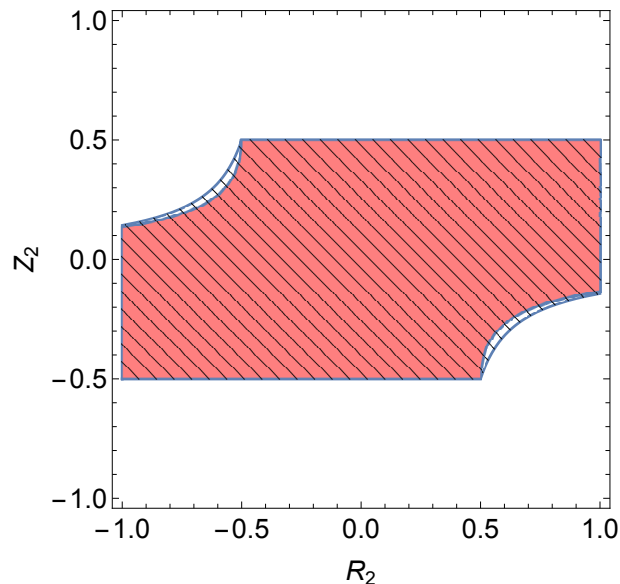

Figure A1. Parameter space in $R_2 - Z_2$ for which the Jacobian $m = 1$ harmonic elimination method is successful. Meshed area: allowed parameter space without boundary self-intersecting. Red solid area: parameter space for which the Jacobian is non-zero with our choice of geometric axis.

Appendix B. Algebraic manipulations of the 2D Jacobian

Appendix B.1. Jacobian from the boundary data

$$\frac{1}{s}[\partial_s\rho\partial_\theta\Gamma - \partial_\theta\partial_s\Gamma]$$

$$= \sum_{a,b=1}^{M}\sum_{k,l=-N}^{N} R_{a,k}Z_{b,l}abs^{a-1}s^{b-1}[\cos(a\theta - k\zeta)\cos(b\theta - l\zeta) + \sin(a\theta - k\zeta)\sin(b\theta - l\zeta)]$$

$$= \sum_{a,b=1}^{M}\sum_{k,l=-N}^{N} R_{a,k}Z_{b,l}abs^{a+b-2}\cos[(a-b)\theta - (k-l)\zeta] \tag{B.1}$$

$$= \sum_{m=0}^{M}\sum_{m'=1}^{M-m}\sum_{k,l=-N}^{N} (m'+m)m's^{2m'+m-2}$$

$$\times c_m\{R_{m+m',k}Z_{m',l}\cos[m\theta - (k-l)\zeta] + R_{m',k}Z_{m+m',l}\cos[-m\theta - (k-l)\zeta]\}$$

$$= \sum_{m=0}^{M}\sum_{m'=1}^{M-m}\sum_{k,l=-N}^{N} (m'+m)m's^{2m'+m-2}\cos[m\theta - (k-l)\zeta]$$

$$\times c_m c_{k-l}(R_{m'+m,k}Z_{m',l} + R_{m',-k}Z_{m'+m,-l}) \tag{B.2}$$

$$= \sum_{m=0}^{M}\sum_{n=-2N}^{2N} s^m \cos(m\theta - n\zeta)\sum_{m'=1}^{M-m}\sum_{n'=0}^{2N-|n|} (m'+m)m's^{2m'-2}$$

$$\times c_m c_n(R_{m'+m,n'+|n|-N}Z_{m',n'-N} + R_{m',N-n'-|n|}Z_{m'+m,N-n'}) \tag{B.3}$$

$$= J_0(s,\zeta) + \tilde{J}(s,\theta,\zeta)$$

where $c_i = \begin{cases} \frac{1}{2}, & i = 0 \\ 1, & i \neq 0 \end{cases}$.

Appendix B.2. Average of the 2D gradient over unit disk

In computing the $(m = 1, n)$-Fourier component of $J_{D^2}|_{s=1}$, we need

$$
\sum_{k=0}^{N} \sum_{l=-N}^{N} \Delta R_k Z_{1,l} 2 \cos(k\zeta) \cos(\theta - l\zeta)
$$

$$
= \sum_{k=0}^{N} \sum_{l=-N}^{N} \Delta R_k Z_{1,l} \{ \cos[\theta + (k - l)\zeta] + \cos[\theta - (k + l)\zeta] \}
$$

$$
= \sum_{k=0}^{N} \sum_{l=-N}^{N} \Delta R_k \{ Z_{1,l} \cos[\theta + (k - l)\zeta] + Z_{1,-l} \cos[\theta - (k - l)\zeta] \}
$$

$$
= \sum_{n=-N}^{2N} \sum_{k=\max(0,n-N)}^{\min(N,N+n)} \Delta R_k \{ Z_{1,k-n} \cos(\theta + n\zeta) + Z_{1,-k+n} \cos(\theta - n\zeta) \}
$$

$$
= \sum_{n=0}^{N} \sum_{k=0}^{N} \Delta R_k \{ (\underbrace{Z_{1,n-k} + \Theta_{N-n-k} Z_{1,k+n}}_{A_{n,k}}) \cos(\theta - n\zeta)
$$

$$
+ \sum_{n=1}^{N} \sum_{k=0}^{N} \Delta R_k (\underbrace{Z_{1,k-n} + \Theta_{N-n-k} Z_{1,-k-n}}_{C_{n,k}}) \cos(\theta + n\zeta) \}
$$

$$
+ \sum_{n=N+1}^{2N} \sum_{k=n-N}^{N} \Delta R_k \{ Z_{1,n-k} \cos(\theta - n\zeta) + Z_{1,k-n} \cos(\theta + n\zeta) \} \qquad \text{(B.4)}
$$

where $\Theta_i = \begin{cases} 1, & i \geq 0 \\ 0, & i < 0 \end{cases}$ is a cut-off coefficient.

We also need

$$\sum_{k=1}^{N} \sum_{l=-N}^{N} \Delta Z_k R_{1,l} 2 \sin(-k\zeta) \sin(\theta - l\zeta)$$

$$= \sum_{k=1}^{N} \sum_{l=-N}^{N} \Delta Z_k R_{1,l} \{\cos[\theta + (k - l)\zeta] - \cos[\theta - (k + l)\zeta]\}$$

$$= \sum_{k=1}^{N} \sum_{l=-N}^{N} \Delta Z_k \{R_{1,l} \cos[\theta + (k - l)\zeta] - R_{1,-l} \cos[\theta - (k - l)\zeta]\}$$

$$= \sum_{n=-N}^{2N} \sum_{k=\max(1,n-N)}^{\min(N,N+n)} \Delta Z_k \{R_{1,k-n} \cos(\theta + n\zeta) - R_{1,-k+n} \cos(\theta - n\zeta)\}$$

$$= \sum_{n=0}^{N} \sum_{k=1}^{N} \Delta Z_k \underbrace{(-R_{1,n-k} + \Theta_{N-k-n} R_{1,k+n})}_{B_{n,k}} \cos(\theta - n\zeta)$$

$$+ \sum_{n=1}^{N} \sum_{k=1}^{N} \Delta Z_k \underbrace{(R_{1,k-n} - \Theta_{N-k-n} R_{1,-k-n})}_{D_{n,k}} \cos(\theta + n\zeta)$$

$$+ \sum_{n=N+1}^{2N} \sum_{k=n-N}^{N} \Delta Z_k \{-R_{1,n-k} \cos(\theta - n\zeta) + R_{1,k-n} \cos(\theta + n\zeta)\} \qquad \text{(B.5)}$$

[1] H. Grad, *Physics of Fluids* **10**, 137 (1967).

[2] J. D. Meiss, *Reviews of Modern Physics* **64**, 795 (1992).

[3] S. P. Hirshman, W. I. van RIJ, P. Merkel, *Computer Physics Communications* **43**, 143 (1986).

[4] J. Loizu, S. Hudson, A. Bhattacharjee, P. Helander, *Physics of Plasmas* **22**, 022501 (2015).

[5] J. Loizu, S. R. Hudson, A. Bhattacharjee, S. Lazerson, P. Helander, *Physics of Plasmas* **22**, 090704 (2015).

[6] A. Reiman, H. Greenside, *Computer Physics Communications* **43**, 157 (1986).

[7] Y. Suzuki, N. Nakajima, K. Watanabe, Y. Nakamura, T. Hayashi, *Nuclear Fusion* **46**, L19 (2006).

[8] S. P. Hirshman, R. Sanchez, C. R. Cook, *Physics of Plasmas* **18**, 062504 (2011).

[9] O. P. Bruno, P. Laurence, *Communications on Pure and Applied Mathematics* **49**, 717 (1996).

[10] M. J. Hole, S. R. Hudson, R. L. Dewar, *Nuclear Fusion* **47**, 746 (2007).

[11] M. Hole, R. Mills, S. R. Hudson, R. Dewar, *Nuclear Fusion* **49**, 065019 (2009).

[12] S. R. Hudson, R. L. Dewar, M. J. Hole, M. McGann, *Plasma Physics and Controlled Fusion* **54** (2012).

[13] S. R. Hudson, *et al.*, *Physics of Plasmas* **19**, 112502 (2012).

[14] J. Loizu, S. R. Hudson, C. Nührenberg, *Physics of Plasmas* **23**, 112505 (2016).

[15] G. R. Dennis, S. R. Hudson, R. L. Dewar, M. J. Hole, *Physics of Plasmas* **20**, 032509 (2013).

[16] G. R. Dennis, S. R. Hudson, R. L. Dewar, M. J. Hole, *Physics of Plasmas* **21**, 042501 (2014).

[17] Z. S. Qu, *et al.*, *Plasma Physics and Controlled Fusion* **62**, 054002 (2020).

[18] G. R. Dennis, S. R. Hudson, R. L. Dewar, M. J. Hole, *Physics of Plasmas* **21** (2014).

[19] M. Lingam, H. M. Abdelhamid, S. R. Hudson, *Physics of Plasmas* **23**, 082103 (2016).

[20] R. L. Dewar, Z. Yoshida, A. Bhattacharjee, S. R. Hudson, *Journal of Plasma Physics* **81**, 515810604 (2015).

[21] R. L. Dewar, L. H. Tuen, M. J. Hole, *Plasma Physics and Controlled Fusion* **59**, 044009 (2017).

[22] R. L. Dewar, J. W. Burby, Z. S. Qu, N. Sato, M. J. Hole, *Physics of Plasmas* **27**, 062504 (2020).

[23] J. Loizu, S. R. Hudson, P. Helander, S. A. Lazerson, A. Bhattacharjee, *Physics of Plasmas* **23**, 055703 (2016).

[24] J. Loizu, S. R. Hudson, C. Nührenberg, J. Geiger, P. Helander, *Journal of Plasma Physics* **83**, 715830601 (2017).

[25] J. Loizu, S. R. Hudson, *Physics of Plasmas* **26**, 030702 (2019).

[26] J. Loizu, *et al.*, *Physics of Plasmas* **27**, 070701 (2020).

[27] S. R. Hudson, *et al.*, *Plasma Physics and Controlled Fusion* **62**, 084002 (2020).

[28] D. Malhotra, A. Cerfon, L. M. Imbert-Gérard, M. O'Neil, *Journal of Computational Physics* **397**, 108791 (2019).

[29] M. O'Neil, A. J. Cerfon, *Journal of Computational Physics* **359**, 263 (2018).

[30] D. Malhotra, A. J. Cerfon, M. O'Neil, E. Toler, *Plasma Physics and Controlled Fusion* **62**, 024004 (2020).

[31] J. P. Boyd, F. Yu, *Journal of Computational Physics* **230**, 1408 (2011).

[32] R. Dewar, S. Hudson, *Physica D: Nonlinear Phenomena* **112**, 275 (1998).

[33] B. Nelson, *et al.*, *Fusion Engineering and Design* **66-68**, 169 (2003). 22nd Symposium on Fusion Technology.

[34] H. R. Lewis, P. M. Bellan, *Journal of Mathematical Physics* **31**, 2592 (1990).

[35] J. P. Boyd, *Journal of Computational Physics* **69**, 112 (1987).

[36] v. F. Zernike, *Physica* **1**, 689 (1934).

[37] A. B. Bhatia, E. Wolf, *Mathematical Proceedings of the Cambridge Philosophical Society* **50**, 40 (1954).

[38] Y. Saad, M. H. Schultz, *SIAM Journal on Scientific and Statistical Computing* **7**, 856 (1986).

[39] Y. Saad, *Iterative methods for sparse linear systems*, vol. 82 (siam, 2003).

[40] E. J. Paul, I. G. Abel, M. Landreman, W. Dorland, *Journal of Plasma Physics* **85**, 795850501 (2019).

[41] C. Beidler, *et al.*, *Fusion Technology* **17**, 148 (1990).

[42] W. Dommaschk, *Computer Physics Communications* **40**, 203 (1986).

[43] J. M. Greene, *Journal of Mathematical Physics* **20**, 1183 (1979).