

# Numerical evaluation of line, surface and toroidal integrals on level sets of toroidally symmetric functions



Matthias Wiesenberger<sup>a,\*</sup>, Raul Gerrú<sup>a</sup>, Markus Held<sup>b,c</sup>

<sup>a</sup> Department of Physics, Technical University of Denmark, Kgs. Lyngby, 2800, Denmark

<sup>b</sup> Department for Mathematics and Statistics, UiT The Arctic University of Norway, Tromsø, N-9037, Norway

<sup>c</sup> Institute for Ion Physics and Applied Physics, Universität Innsbruck, Innsbruck, A-6020, Austria

## ARTICLE INFO

### Article history:

Received 14 December 2022

Received in revised form 15 June 2023

Accepted 28 July 2023

Available online 5 August 2023

### Keywords:

Flux-surface average

Toroidal average

Poloidal average

Delta-function

Surface integral

Line integral

## ABSTRACT

We investigate strategies to numerically integrate closed lines and surfaces that are implicitly defined by level sets (iso-contours) of continuously differentiable toroidally symmetric functions. The “grid-transform” approach transforms quantities given in non-surface-aligned coordinates onto a numerically constructed surface-aligned grid. Here, line and surface integrals, as well as so-called flux-surface averages, can be easily evaluated using high order integration formulas. We compare this method to ones that base on numerical representations of the delta-function.

For the grid-transform method we observe high order convergence of line, surface and volume integration. Quantitatively, the errors for line and area integration are several orders of magnitude smaller than previously reported errors for delta-function methods. Furthermore, a delta-function method based on a Gaussian representation shows qualitatively wrong results of surface integrals near O- and X-points. Contrarily, the grid transform method suffers no deterioration near O-points. However, close to X-points we observe reduced first order convergence in volume integral and derivative tests due to the diverging volume element.

Finally, we derive a toroidal integration based on toroidal summation and a smoothing kernel that assumes field-alignment of structures between the toroidal planes. The smoothing kernel can be interpreted as a partial flux-surface average. The resulting smoothed toroidal average eliminates unphysical poloidal oscillations that are otherwise present in the simple toroidal average.

Our methods can be applied to toroidal and flux-surface averages in simulations of three-dimensional plasma dynamics on non-aligned grids. Further applications include closed line and surface integrals in level set methods. Efficient implementations can be found in the freely available FELTOR library.

© 2023 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

It is crucial for three-dimensional fusion plasma simulations to predict reliably the confinement of the plasma. To this end, an accurate evaluation of the mass, momentum and energy fluxes through the flux-surfaces in general and the last

\* Corresponding author.

E-mail address: [mattwi@fysik.dtu.dk](mailto:mattwi@fysik.dtu.dk) (M. Wiesenberger).

closed flux-surface in particular is required [1]. Flux-surfaces in this context are surfaces implicitly defined by level sets (iso-surfaces) of the magnetic flux-function  $\psi_p(R, Z)$ , where  $R, Z, \varphi$  are cylindrical coordinates [1,2]. We here discuss toroidally symmetric (flux-)functions (independent of  $\varphi$ ) that model so-called tokamaks [3]. The last closed flux-surface is the one that includes an X-point (a saddle point of  $\psi_p$ ). Typically, so called flux-surface averages are computed. Even though these are defined as differential volume averages their evaluation requires the computation of surface integrals of the form

$$\int_{\psi_p=\text{const}} f(R, Z, \varphi) dA \quad (1)$$

where  $f$  is a function and the integration domain is the toroidally symmetric surface implicitly defined by a level set (iso-surface) of  $\psi_p(R, Z)$  [1,2].

Surface integration of the form in Eq. (1) is numerically straightforward in so-called flux-aligned coordinate systems [2]. These are coordinate systems where one coordinate is given by (a function of)  $\psi_p$  itself. Care must only be taken of the metric elements in the area integral in curvilinear coordinates [2,4]. In fact, many three-dimensional plasma turbulence simulations base on flux- or more specifically *field*-aligned coordinate systems [5–7]. These coordinate systems exploit the fact that plasma turbulence is field-aligned that is structures are elongated along the magnetic field and very narrow perpendicular to it. This reflects in a reduction of grid points along the magnetic field line compared to the perpendicular resolution and an associated saving in computational cost.

The problem with flux-aligned grids is that they can become inconsistent when the flux-surfaces exhibit one or more X-points. In that case, the convergence rate of numerical solutions to elliptic equations is reduced to first order due to the singular metric as was recently pointed out by [8]. For this reason, non-aligned coordinates like cylindrical ones must be used. Straightforward implementations exist [9] as well as more sophisticated approaches that relieve the flux-alignment only in a small region around the X-point [10]. However, these approaches do not exploit the field-aligned character of turbulence and can thus only be used for small to medium sized tokamaks due to both high numerical diffusion and extreme computational cost [11,12]. These problems are solved by the so-called flux-coordinate independent approach [13,11,14]. Here, the grid is not aligned to the flux function  $\psi_p$  and at the same time the toroidal ( $\varphi$ ) direction can be resolved by only 16-32 points in practical simulations (compared to several hundred points in non-aligned simulations) [15–18]. The issue then remains how to compute the surface integral (1) in these non-aligned grids.

In fact, integrals of the form (1) on non-aligned grids appear similarly outside the field of plasma turbulence in the field of level set methods [19]. Here,  $\psi_p$  takes the role of the level-set function that captures an interface implicitly given by  $\psi_p = 0$ . Contrary to the magnetic flux function above, the level set function develops in time and is given as the solution to the level set equation. Level set methods have triggered active research into the discretization of implicitly defined line and surface integrals. Note here that a line integral is obtained by restricting Eq. (1) to two dimensions. In particular, discretizations based on delta-function formulations [20,21] as well as others such as a tree-based approach [22] are discussed.

A second issue in simulations with the flux-coordinate independent (FCI) approach arises in toroidal averages. In general, *toroidal* refers to the  $\varphi$  direction. A toroidal average of a function  $f(R, Z, \varphi)$  is thus defined as

$$\langle f \rangle_\varphi(R, Z) := \frac{1}{2\pi} \oint f(R, Z, \varphi) d\varphi \quad (2)$$

In the FCI approach the number of toroidal planes (the number of grid points in the  $\varphi$  direction) is typically very low as explained above. A problem then occurs when trying to compute a toroidal average since it is the parallel (to the magnetic field) direction that is resolved by the simulation and not the toroidal one. Simply summing up all toroidal planes yields poor results seen in cross-sections as little humps [15].

In this contribution we compare strategies to compute line and surface integrals as well as flux-surface and toroidal averages. First, line and surface integrals as well as flux-surface averages are computed via the numerical construction of and interpolation to flux-aligned grids. Such a strategy was first proposed in an appendix [17] but no systematic analysis was given and the behavior near X-points remained unclear. Flux-aligned grids can be quickly and accurately constructed including the metric tensor with the methods presented in [23,8]. We compare this “grid-transform” method to “delta-function” methods that are based on approximations of the Dirac delta-function. For the toroidal average we derive a smoothing operation based on the assumption that structures are field-aligned in-between toroidal planes. This yields superior results for the toroidal average in such a situation. The algorithm can be easily applied as a post-processing diagnostics on a pre-computed simple toroidal average based on summation of toroidal planes. We provide implementations freely in the C++ library FELTOR [24,25].

In Section 2 we derive the suggested formulas for the smoothed toroidal average. Furthermore, the delta-function approach and the flux-aligned grid approach for line and surface integrals as well as the flux-surface average are derived. In a second part 3 we numerically test our proposed schemes for convergence and accuracy. We conclude in Section 4. All figures and tables in this paper can be reproduced following the data statement in Appendix A.

## 2. Derivation of the algorithms

We will start with the derivation of the smoothed toroidal average in Section 2.1. The second part of this Section 2.2 consists of the derivation of line and surface integration formulas. In the last part 2.3 of this Section we discuss and summarize our analytical findings.

### 2.1. Toroidal averages

We first introduce a suitable representation of a toroidal magnetic field in Section 2.1.1 including the introduction of the magnetic flux function  $\psi_p(R, Z)$ . This can be used to define field-alignment via the flow operator of the magnetic field in Section 2.1.2. The final result is presented in Section 2.1.3.

#### 2.1.1. The magnetic field

In cylindrical coordinates  $(R, Z, \varphi)$  the general toroidally symmetric magnetic field  $\mathbf{B}$  obeying an MHD equilibrium ( $\mu_0 \mathbf{j} = \nabla \times \mathbf{B}$ ,  $\nabla p = \mathbf{j} \times \mathbf{B}$  with current  $\mathbf{j}$ , magnetic constant  $\mu_0$  and pressure  $p(\psi_p)$ ) can be written as [2] (in dimensionless units)

$$\mathbf{B} = \frac{R_0}{R} \left[ I(\psi_p) \hat{\mathbf{e}}_\varphi + \frac{\partial \psi_p}{\partial Z} \hat{\mathbf{e}}_R - \frac{\partial \psi_p}{\partial R} \hat{\mathbf{e}}_Z \right], \quad (3)$$

with  $I$  the current streamfunction,  $R_0$  the major radius,  $\psi_p$  the poloidal magnetic flux and  $\hat{\mathbf{e}}_R$ ,  $\hat{\mathbf{e}}_Z$ ,  $\hat{\mathbf{e}}_\varphi$  the coordinate unit vectors.

We employ cylindrical coordinates with  $\varphi$  anti directed to the geometric toroidal angle (*clockwise* if viewed from above) to obtain a right handed system. The parametric representation in Cartesian  $(x, y, z)$  coordinates is therefore simply:

$$x = R \sin(\varphi), \quad y = R \cos(\varphi), \quad z = Z. \quad (4)$$

With a typically convex function  $\psi_p$  (the second derivatives matrix, the Hessian, is positive semi-definite),  $I(\psi_p) > 0$  and the previously defined coordinate system the field line winding is a *left handed screw* in the positive  $\hat{\mathbf{e}}_\varphi$ -direction. Note the contra- and co-variant components of the magnetic field

$$B^R = B_R = \frac{R_0}{R} \psi_{p,Z}, \quad B^Z = B_Z = -\frac{R_0}{R} \psi_{p,R}, \quad B^\varphi = \frac{1}{R^2} B_\varphi = \frac{R_0 I}{R^2}. \quad (5)$$

#### 2.1.2. Field-alignment

We integrate the streamlines (integral curves) of the magnetic field (3) via

$$\frac{d\bar{R}}{d\bar{\varphi}} = \frac{B^R}{B^\varphi}(\bar{R}, \bar{Z}, \bar{\varphi}) \quad (6a)$$

$$\frac{d\bar{Z}}{d\bar{\varphi}} = \frac{B^Z}{B^\varphi}(\bar{R}, \bar{Z}, \bar{\varphi}) \quad (6b)$$

As initial conditions for  $\bar{\varphi} = 0$  we choose  $\bar{R}(0) = R$ ,  $\bar{Z}(0) = Z$ . Then, we can represent the flow of  $\mathbf{B}/B^\varphi$  as

$$\mathcal{T}_{\pm\Delta\varphi} \mathbf{x} \equiv \mathcal{T}_{\pm\Delta\varphi}[R, Z, \varphi] := (\bar{R}(\pm\Delta\varphi), \bar{Z}(\pm\Delta\varphi), \varphi \pm \Delta\varphi) \quad (7)$$

The flow  $\mathcal{T}_{+\Delta\varphi}$  is a map that defines a pullback (that we also call  $\mathcal{T}_{+\Delta\varphi}$ )

$$[\mathcal{T}_{+\Delta\varphi} f](\mathbf{x}) \equiv f(\mathcal{T}_{+\Delta\varphi} \mathbf{x}) \quad (8)$$

We call a function  $f(\mathbf{x}) = f(R, Z, \varphi)$  *field-aligned* if  $\mathcal{T}_{+\Delta\varphi} f = f$ . Note that  $\mathcal{T}_{\pm\Delta\varphi}$  is the same operator we introduced in [11]. From the group properties of the flow [4] we have that  $\mathcal{T}_{-\Delta\varphi} \circ \mathcal{T}_{+\Delta\varphi} = \mathbf{1}$  and

$$\mathcal{T}_{+k\Delta\varphi} = (\mathcal{T}_{+\Delta\varphi})^k \quad (9)$$

Finally, we define the pullback (7) applied to two-dimensional functions via

$$[\mathcal{T}_{+\Delta\varphi} f](R, Z) \equiv f(\bar{R}(\Delta\varphi), \bar{Z}(\Delta\varphi)) \quad (10)$$

This preserves the group properties of  $\mathcal{T}_{+\Delta\varphi}$  because our vector field  $\mathbf{B}$  is independent of  $\varphi$ . Numerically, the integration of Eq. (6) can be achieved with any standard ODE integrator [26] while the evaluation of  $f$  at the transformed points  $\mathcal{T}_{\pm\Delta\varphi} \mathbf{x}$  necessitates an interpolation. Any standard algorithm like bilinear or bicubic interpolation is viable or, as is used in FELTOR, a polynomial interpolation of arbitrary order given by a discontinuous Galerkin discretization in the perpendicular direction [11]. In passing we note that in flux-aligned coordinates the above discussion works completely analogous replacing  $R$  with the flux-aligned coordinate  $\zeta$ , and  $Z$  with the angle coordinate  $\eta$ . The only difference is that the resulting interpolation is then necessary only in the angle coordinate  $\eta$ .

2.1.3. The smoothed toroidal or partial flux-surface average

We can now discretize the toroidal average given in Eq. (2) assuming that the function  $f$  is field-aligned. We start with a numerical representation of  $f(R, Z, \varphi)$  on a toroidally symmetric grid with  $N_\varphi$  planes in the  $\varphi$  direction. We here assume equidistant planes for simplicity but the following scheme can be easily generalized to non-equidistant planes (and other integration schemes) as well. A simple numerical expression of Eq. (2) is the trapezoidal rule, which since  $f$  is periodic in  $\varphi$ , amounts to

$$\langle f \rangle_\varphi^T(R, Z) := \frac{1}{N_\varphi} \sum_{i=0}^{N_\varphi-1} f_i(R, Z) \tag{11}$$

where  $f_i(R, Z) := f(R, Z, \varphi_i)$  is the  $i$ -th toroidal plane with  $\varphi_i := 2\pi i/N_\varphi$ .

We now seek to improve the accuracy of  $\langle f \rangle_\varphi^T$  in the case that  $f$  is field-aligned and the number of toroidal planes  $N_\varphi$  is low. Our idea is simple: we first interpolate the function along the magnetic field-lines onto a large number of toroidal planes and only then use Eq. (11) to compute the average. In other words we increase the accuracy of the integration by artificially increasing the number of toroidal planes using the field-aligned property of  $f$ .

Let us insert  $K_\varphi - 1$  equidistant planes in the  $\varphi$  direction between each neighboring two original planes. The total number of planes is now  $N_\varphi^{\text{tot}} = N_\varphi K_\varphi$  and the grid distance  $\delta\varphi = \Delta\varphi/K_\varphi$ .

We are now able to extend the numerical function  $f$  given on the coarse  $\varphi$  grid unto the fine  $\varphi$  grid via a linear interpolation along field-lines in between planes

$$f(R, Z, \varphi_i + k\delta\varphi) = \frac{K_\varphi - k}{K_\varphi} \mathcal{T}_{-\delta\varphi}^k f_i + \frac{k}{K_\varphi} \mathcal{T}_{\delta\varphi}^{K_\varphi - k} f_{i+1}$$

In words, we pull the planes  $f_i$  and  $f_{i+1}$  back to the in-between plane  $f_{i+k/K_\varphi}$ . Now, we can apply the simple toroidal average (11) on the fine grid

$$\begin{aligned} \langle f \rangle_\varphi^{ST} &:= \frac{1}{K_\varphi N_\varphi} \sum_{i=0}^{N_\varphi-1} \sum_{k=0}^{K_\varphi-1} \frac{K_\varphi - k}{K_\varphi} \mathcal{T}_{-\delta\varphi}^k f_i + \frac{k}{K_\varphi} \mathcal{T}_{\delta\varphi}^{K_\varphi - k} f_{i+1} \\ &= \frac{1}{K_\varphi} \left[ \langle f \rangle_\varphi^T + \sum_{k=1}^{K_\varphi-1} \frac{K_\varphi - k}{K_\varphi} \left( \mathcal{T}_{-\delta\varphi}^k + \mathcal{T}_{+\delta\varphi}^k \right) \langle f \rangle_\varphi^T \right] \end{aligned} \tag{12}$$

The central step is to use that the pullback  $\mathcal{T}_{-\delta\varphi}$  is linear that is  $\mathcal{T}_{-\delta\varphi} f_0 + \mathcal{T}_{-\delta\varphi} f_1 = \mathcal{T}_{-\delta\varphi} (f_0 + f_1)$  [4]. This recovers the simple toroidal summation  $\langle f \rangle_\varphi^T$  in Eq. (11). In the limit  $K_\varphi \rightarrow \infty$  the discrete sum represents the integral

$$\langle f \rangle_\varphi^{ST}(R, Z) = \int_0^{2\pi} w(\varphi) \langle f \rangle_\varphi^T(\bar{R}(\varphi), \bar{Z}(\varphi)) d\varphi \equiv \mathcal{S}_{\Delta\varphi} \left( \langle f \rangle_\varphi^T(R, Z) \right) \tag{13}$$

with  $w(\varphi)$  a linear smoothing kernel

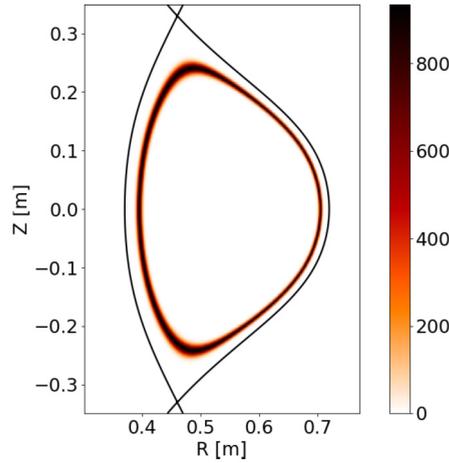
$$w(\varphi) = \frac{1}{\Delta\varphi} \begin{cases} 1 - \frac{\varphi}{\Delta\varphi} & \text{for } 0 < \varphi \leq \Delta\varphi \\ 1 + \frac{\varphi}{\Delta\varphi} & \text{for } 0 \geq \varphi \geq -\Delta\varphi \\ 0 & \text{else} \end{cases} \tag{14}$$

We note in passing that by using nearest neighbor or cubic interpolation schemes to extend  $f$  in-between toroidal planes we obtain different smoothing kernels  $w(\varphi)$ , but  $\int w(\varphi) = 1$  holds in general. The toroidal average on field-aligned functions is obtained by simple toroidal summation Eq. (11) followed by a smoothing operation  $\mathcal{S}_{\Delta\varphi}$  with smoothing kernel  $w(\varphi)$ . We implicitly define the smoothing operation  $\mathcal{S}_{\Delta\varphi}$  in Eq. (13) as an independent two-dimensional operation. In fact we will also refer to  $\mathcal{S}_{\Delta\varphi}(f(R, Z))$  as a ‘‘partial flux-surface average’’, seeing that it integrates a function on a part of the flux-surface, which in 2d is just a line. Eq. (13) is thus suggestively denoted with the index ‘‘ST’’ to stand for ‘‘toroidal average (T) followed by smoothing (S)’’. If we slightly extend the definition of  $\mathcal{S}_{\Delta\varphi}(f)$  to three-dimensional functions

$$\mathcal{S}_{\Delta\varphi}(f)(R, Z, \varphi) := \int_0^{2\pi} d\varphi' w(\varphi') f(\bar{R}(\varphi'), \bar{Z}(\varphi'), \varphi) \tag{15}$$

then we can easily see that the smoothing operation commutes with toroidal averaging

$$\langle f \rangle_\varphi^{TS} = \langle \mathcal{S}_{\Delta\varphi}(f) \rangle_\varphi^T = \mathcal{S}_{\Delta\varphi} \left( \langle f \rangle_\varphi^T \right) = \langle f \rangle_\varphi^{ST} \tag{16}$$



**Fig. 1.** Plot of Eq. (18) for an example magnetic field equilibrium  $h = \psi_p(R, Z)$  and a high value for  $\epsilon$  for better visibility. The black line indicates the separatrix  $\psi_p = 0$ .

Finally, we realize that for  $N_\varphi \rightarrow \infty$  (and thus  $\Delta\varphi \rightarrow 0$ ) Eq. (12) becomes the derivative of the integral over  $\varphi$  and thus  $\lim_{\Delta\varphi \rightarrow 0} \langle f \rangle_\varphi^{ST} = \lim_{\Delta\varphi \rightarrow 0} \langle f \rangle_\varphi^T = \langle f \rangle_\varphi$ .

### 2.2. Line and surface integrals and the flux surface average

We start with the basic identities for a delta-function formulation in Section 2.2.1. This is then followed by area integrals in flux-aligned coordinates 2.2.2. Finally, we combine the results into a definition of the flux-surface average and show flux-surface averaged conservation equations in Section 2.2.3.

#### 2.2.1. Dirac delta and area integrals

Recall that the *Dirac delta-function* has the property (in any dimension) [27]:

$$\int_V f(\mathbf{x}) \delta(h(\mathbf{x}) - h') dV = \int_{h=h'} \frac{f(\mathbf{x})}{|\nabla h|} dA \tag{17}$$

which means that the delta-function can be used to express area integrals of the submanifold given by the level set of the function  $h(\mathbf{x})$ . A numerically tractable approximation to the delta-function reads

$$\delta_\epsilon(h(\mathbf{x}) - h') := \frac{1}{2\pi\epsilon^2} \exp\left(-\frac{(h(\mathbf{x}) - h')^2}{2\epsilon^2}\right) \tag{18}$$

where  $\epsilon$  is a small, free parameter. In Fig. 1 we plot Eq. (18) for an example equilibrium. The chosen representation in Eq. (18) is very easy to implement and fast to evaluate. However, it must be mentioned that there are other more sophisticated numerical approximations to the delta-function, for example the ones presented in [21,20]. In particular, our representation (18) does not have compact support. We will discuss this further in Section 3.2.2.

Inserting Eq. (18) into the left-hand side of Eq. (17) thus yields a method to compute area integrals even if the coordinate system is not aligned to the area [19].

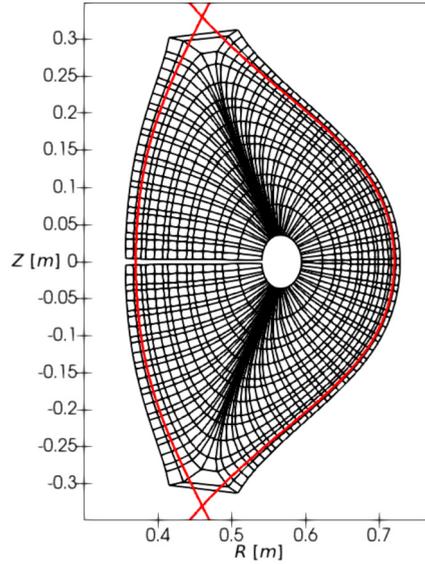
$$\int_{\psi_{p,0}} f(\mathbf{x}) dA = \int_{\psi_{p,0}} f(\mathbf{x}) |\nabla \psi_p| \delta_\epsilon(\psi_p(\mathbf{x}) - \psi_{p,0}) dV \tag{19}$$

In our work we use Gauss-Legendre integration to represent this integral numerically.

Finally, recall the *co-area formula*

$$\int_{\Omega_0} f(\mathbf{x}) dV = \int_0^{h_0} \left( \int_{h=h'} \frac{f(\mathbf{x})}{|\nabla h|} dA \right) dh' \tag{20}$$

where  $\Omega_0$  is the volume enclosed by the level set  $h = h_0$ . The co-area formula can be viewed as a change of variables in the volume integral and we here obtain it by integrating Eq. (17) over  $h'$ .



**Fig. 2.** Plot of a flux-aligned grid (visible are the cell-centers, not the cell boundaries) for an example magnetic field equilibrium. In red we plot the separatrix  $\psi_p = 0$ . (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

### 2.2.2. Area integrals in flux-aligned coordinates

In a flux-aligned coordinate system one coordinate is given as a function of the magnetic flux  $\zeta = \zeta(\psi_p)$  while the other coordinates are the poloidal angle-like coordinate  $\eta$  and the toroidal angle  $\varphi$ . For simplicity, we will here assume  $\zeta(\psi_p) = \zeta_0 \psi_p$  with constant  $\zeta_0 > 0$ . Any function  $f(R, Z, \varphi)$  can be pulled back to flux-aligned coordinates by

$$F(\zeta, \eta, \varphi) := f(R(\zeta, \eta), Z(\zeta, \eta), \varphi) \quad (21)$$

which, numerically, is represented by a simple interpolation operation, for which we already discussed available methods at the end of Section 2.1.2.

The area 2-form is given by the interior product of the surface normal with the volume form [4]

$$dA = i_{\hat{\psi}_p} \cdot \text{vol}^3, \quad \hat{\psi}_p := \frac{\nabla \psi_p}{|\nabla \psi_p|} \quad (22)$$

In a flux-aligned coordinate system  $\nabla \psi_p$  only has one component  $g^{\zeta\zeta} \partial \psi_p / \partial \zeta$  and we thus arrive at

$$dA = \sqrt{g^{\zeta\zeta}} \sqrt{g} d\eta d\varphi = |\zeta_0| |\nabla \psi_p| \sqrt{g} d\eta d\varphi, \quad (23)$$

where we used that  $g^{\zeta\zeta} = (\nabla \zeta)^2 = \zeta_0^2 (\nabla \psi_p)^2$ . In a flux-aligned coordinate system we can thus compute [2]

$$\int_{\psi_{p,0}} f(\mathbf{x}) dA = \oint F(\zeta(\psi_{p,0}), \eta, \varphi) |\zeta_0| |\nabla \psi_p| \sqrt{g} d\eta d\varphi \quad (24)$$

which next to Eq. (19) constitutes the second method to compute area integrals that we study in this work.

We note here that in order to compute line-integrals in two dimensions we use that  $\sqrt{g_{2D}} = \sqrt{g}/R(\zeta, \eta)$  and write

$$\int_{\psi_{p,0}} f(R, Z) ds = \oint F(\zeta, \eta) |\zeta_0| |\nabla \psi_p| \sqrt{g_{2D}} d\eta \quad (25)$$

while the two-dimensional area integral is

$$\int_{\Omega} f(R, Z) dA = \int_0^{\zeta(\psi_p)} \int_0^{2\pi} F(\zeta, \eta) \sqrt{g_{2D}} d\zeta d\eta \quad (26)$$

where here  $\Omega$  is the area enclosed by  $\psi_p$ .

In Fig. 2 we plot a flux-aligned grid for an example magnetic field equilibrium with a reduced resolution to increase visibility. We use the methods described in [23,8] to produce the grid. In particular, the grid can be constructed including

the metric elements up to machine precision. We choose the grid spacing such that the separatrix ( $\psi_p = 0$ ) is reasonably discretized in the sense that grid cells are close to being quadratic. However, this choice leads to a poloidal over-resolution of the region close to the O-point (minimum of the flux-function), where grid cells are very small in the poloidal direction but large in the radial direction. This means that flux-surfaces close to the O-point are better resolved than flux-surfaces close to the separatrix.

Notice the coordinate singularities at the O-point and the X-point, which in principle could lower or prevent the convergence of numerical integrals. As we will show in Section 3, the evaluation of integration formulas like Eq. (24) and the flux-surface averages in Section 2.2.3 converge well near the O-point but have reduced convergence rates near the X-point. Note also that we integrate close to but not including the singularities.

### 2.2.3. Flux surface average

We first define the volume flux label and its derivative by [2]

$$v(\psi_{p,0}) := \int H(\psi_p(R, Z) - \psi_{p,0}) dV = 2\pi \int_{\zeta(\psi_{p,0})}^{\zeta(\psi_{p,0})} \oint \sqrt{g} d\zeta d\eta, \tag{27}$$

$$\frac{dv}{d\psi_p} = \int |\nabla \psi_p|^{-1} dA = 2\pi |\zeta_0| \oint_{\zeta(\psi_p)} \sqrt{g} d\eta \tag{28}$$

where  $\psi_{p,0}$  is the value of  $\psi_p$  at the O-point and  $H$  is a Heaviside function defined such that it is 1 in the interior and 0 in the exterior of  $\psi_{p,0}$ . Notice that Eqs. (27) and (28) can be evaluated numerically in both a non-aligned or an aligned coordinate system.

Also pay attention to the orientation. We assume that the value of  $v$  increases if we go from the O-point towards the separatrix. That means that if  $\psi_{p,0} > \psi_{p,x}$ , then the signs of the definitions in Eq. (27) and (28) need to change, i.e. use  $H(\psi_{p,0} - \psi_p)$ , integrate from  $\int_{\zeta(\psi_{p,0})}^{\zeta(\psi_{p,x})}$  and add a negative sign in Eq. (28). At the same time the following definition of the flux-surface average is independent of orientation.

The flux surface average is defined as a *volume average* over a small volume - a shell centered around the flux-surface - defined by two neighboring flux-surfaces [2,1]. With the help of the volume flux label (27) and the co-area formula (20) we define

$$\langle f \rangle(\psi_p) := \frac{\partial}{\partial v} \int f dV = \frac{1}{\int_{\psi_p} |\nabla \psi_p|^{-1} dA} \int_{\psi_p} \frac{f(\mathbf{x})}{|\nabla \psi_p|} dA \tag{29}$$

In a cylindrical coordinate system we can use Eq. (19) to write

$$\langle f \rangle = \frac{\int_{\Omega} \langle f \rangle_{\varphi}(R, Z) \delta_{\varepsilon}(\psi_p(R, Z) - \psi_{p,0}) R dR dZ}{\int_{\Omega} \delta_{\varepsilon}(\psi_p(R, Z) - \psi_{p,0}) R dR dZ} \tag{30}$$

In flux-aligned coordinates the flux-surface average simplifies to [2]

$$\langle f \rangle = \frac{1}{\oint \sqrt{g} d\eta} \int_0^{2\pi} \langle f \rangle_{\varphi}(\zeta(\psi_{p,0}), \eta) \sqrt{g} d\eta \tag{31}$$

We immediately see that the flux-surface average is particularly easy to compute in a flux-aligned coordinate system. Notice however that the volume element does appear (and depends on both  $\zeta$  and  $\eta$ ) and that the flux-surface average is *not* simply an angle average.

From both Eq. (30) and Eq. (31) we implicitly define a “poloidal average”  $\langle f \rangle_{\eta}$ , which commutes with the toroidal average

$$\langle f \rangle(\psi_p) := \langle \langle f \rangle_{\varphi}(R, Z) \rangle_{\eta}(\psi_p) = \langle \langle f(R, Z, \varphi) \rangle_{\eta}(\psi_p, \varphi) \rangle_{\varphi}(\psi_p) \tag{32}$$

where we slightly overburden the notation in different dimensions. Similar to the smoothing (or partial flux surface) operator  $\mathcal{S}_{\Delta\varphi}$  the poloidal average always acts on the  $R$  and  $Z$  component of a function, while the toroidal average always acts on the  $\varphi$  component. An interesting question is, what happens if we apply the poloidal average to a partial flux surface average. Intuitively we expect  $\langle \mathcal{S}_{\Delta\varphi}(f) \rangle = \langle f \rangle$ , since a partial average on a flux-surface should not change the complete integral over a flux-surface. It turns out however, due to the volume element, that the two expressions are only approximately equal

$$\langle \mathcal{S}_{\Delta\varphi}(f(R, Z)) \rangle_{\eta} \approx \langle f(R, Z) \rangle_{\eta} \tag{33}$$

In section 3, we are going to show that for most practical applications  $\Delta\varphi$  is small enough that the two sides can be considered equal. This is because the volume element varies very slowly in the poloidal direction. From this observation

also follows that in the numerical computation of the flux-surface average both  $\langle f \rangle_\varphi^T$  as well as  $\langle f \rangle_\varphi^{ST}$  should give (approximately) the same result. It is important to notice that this is *irrespective* of whether  $f$  is field-aligned or not. Also notice that in the limit  $\lim_{\Delta\varphi \rightarrow 0} \langle \mathcal{S}_{\Delta\varphi}(f) \rangle = \langle f \rangle$ .

The flux-surface average (29) fulfills the identities [1,2]

$$\langle \mu f + \lambda g \rangle = \mu \langle f \rangle + \lambda \langle g \rangle \quad (34)$$

$$\langle f(\psi_p) \rangle = f(\psi_p) \quad (35)$$

$$\langle \nabla \cdot \mathbf{j} \rangle = \frac{d}{dv} \langle \mathbf{j} \cdot \nabla v \rangle \quad (36)$$

Flux-surface averages are an important diagnostic tool for three-dimensional plasma simulations, typically applied in post-processing of simulation data. Here, they are used to analyse the confinement of the simulated plasma based on mass, energy or angular momentum conservation [28,29]. These conservation laws are of the form  $\partial_t X + \nabla \cdot \mathbf{j}_X = \Lambda_X$  for the quantity  $X$  with associated flux  $\mathbf{j}_X$  and dissipation/source  $\Lambda_X$ . We can write

$$\frac{\partial}{\partial t} \langle X \rangle + \frac{\partial}{\partial v} \langle \mathbf{j}_X \cdot \nabla v \rangle = \langle \Lambda_X \rangle \quad (37)$$

where again  $v = v(\psi_p)$  is the volume flux label. The *total flux* of a given flux density  $\mathbf{j}_X$  through the flux surface  $\psi_p = \psi_{p0}$  is given by

$$\langle \mathbf{j}_X \cdot \nabla v \rangle := J_X = \oint_{\psi_p = \psi_{p0}} \mathbf{j}_X \cdot d\mathbf{A} = \frac{dv}{d\psi_p} \langle \mathbf{j}_X \cdot \nabla \psi_p \rangle \quad (38)$$

Notice that  $d\mathbf{A}$  is oriented with respect to the volume flux label, that is it points *away* from the O-point. Once we have the flux-surface averaged equation (37) we can easily integrate over the volume flux label to get

$$\frac{\partial}{\partial t} \int_{\Omega} X dV + \langle \mathbf{j}_X \cdot \nabla v \rangle (v(\psi_p)) = \int_{\Omega} \Lambda_X dV \quad (39)$$

where  $\Omega$  is the region enclosed by the flux surface  $\psi_p$ . Here, we use the co-area formula (20) to get  $\int_0^{v(\psi_p)} \langle X \rangle dv = \int_{\Omega} X dV$ .

Notice here that in order to evaluate Eq. (37) or Eq. (39) it is necessary to compute the derivatives  $\partial \langle f \rangle / \partial v$  and integrals  $\int \langle f \rangle dv$ . Numerically, this can be done straightforwardly. First, a one-dimensional discretization of  $\psi_p$  must be chosen, for example  $N_\psi$  equidistant values  $\psi_{p,i}$  between  $\psi_{p,0}$  and  $\psi_{p,0}$  (or a discretization on Gaussian abscissas for a higher order result). Then the flux surface averages as well as  $dv/d\psi_p$  can be computed for all  $\psi_{p,i}$ . Finally, a numerical difference and integration formula can be applied to the resulting one-dimensional quantities to obtain an expression for the derivative  $(dv/d\psi_p)^{-1} d \langle \mathbf{j}_X \cdot \nabla v \rangle / d\psi_p$  and integral  $dv/d\psi_p \int \langle X \rangle d\psi_p$ .

### 2.3. Discussion

The main result regarding the toroidal average is Eq. (12). The necessary transformation operators  $\mathcal{T}_{+\delta\varphi}$  and  $\mathcal{T}_{-\delta\varphi}$  have to be constructed by integrating Eq. (6). Numerically, these operators then appear as interpolation matrices on the resulting points. Boundary conditions can for example be implemented by assuming perfect field-alignment of  $f$  outside the domain.

The flux-surface average defined in Eq. (29) can be evaluated on non-aligned grids with the help of Eq. (30) and on flux-aligned coordinates with Eq. (31). The flux-surface average can be interpreted as a poloidal followed by a toroidal average (32) and approximately commutes with the partial flux-surface average  $\mathcal{S}_{\Delta\varphi}$  (33). Finally, we outline how to compute volume derivatives and integrals necessary in flux-surface averaged conservation laws like (37) and (39). Ordinary line and surface integrals in aligned coordinates are given by Eq. (25) and (26).

The main suggestion in this work is to use the formulas for aligned coordinates even if the field to integrate is given in non-aligned coordinates. To this end, first, a structured grid including the metric tensor must be constructed numerically. The pullback to aligned coordinates Eq. (21) is then numerically given by interpolation. We refer to this method as the *grid-transform method*.

We offer several grid generators in our C++ library FELTOR [25]. These use the methods previously presented in [23,8]. In particular, the grid generators are fast, accurate and able to handle one or more X-points in or at the boundary of the domain of interest. The grid generators work with both analytically as well as numerically given functions  $\psi_p$ . In the context of magnetic confinement fusion in tokamaks,  $\psi_p$  can be given as an expansion in analytical basis functions that solve the associated Grad-Shafranov equations as represented in [30]. The first and second derivatives of  $\psi_p$  are then given via the corresponding derivatives of the basis functions, which can be computed analytically. If  $\psi_p$  is given on a discrete grid, the grid generator interpolates  $\psi_p$  and its derivatives at the integration points. This may be of particular interest to the level set method [19].

The scenario in which we envisage the smoothed toroidal and flux-surface averages to be used are as a post-processing tool for three dimensional simulation data. Note that both the smoothed toroidal average as well as the flux surface average can be evaluated given solely the simple toroidal average. Thus, storage of the full three-dimensional fields can be avoided if during the simulation the simple toroidal average Eq. (11) of the quantities of interest is computed and stored in file. This is cheap to compute and can significantly lower the storage costs of simulation outputs since the memory requirement for two-dimensional fields is typically an order of magnitude lower than the full three-dimensional fields.

In this scenario the resolution of the data is a given constant and the task is to compute the smoothed toroidal and flux-surface averages as accurate as possible with the available data. In the next Section 3 we will analyse our methods with this philosophy in mind. The free parameters to tune in our two available methods are the width of the delta-function  $\varepsilon$  in the delta-function method, the resolution in  $\eta$  in the grid-transform method and in both methods the resolution in  $\psi_p$ , which influences the accuracy of volume derivatives and integrals.

Performance-wise, we compute  $N_\psi$  volume integrals over the entire domain. Each of the volume integrals numerically corresponds to a reduction. In the delta-function method each reduction has  $N_R N_Z$  elements while in the grid-transform method only  $N_\eta \ll N_R N_Z$  elements are used. In the grid-transform method the flux-aligned grid has to be constructed which isn't the case for the delta-function approach. The grid, however, has to be computed only once and we here assume that the construction time amortises if many integrals are evaluated. In total, we thus expect a performance gain of the grid-transform method over the delta-function method.

### 3. Numerical tests

We first discuss the toroidal average in Section 3.1. Afterwards, we test the methods to evaluate the line and surface integrals and the flux-surface average in Section 3.2. We briefly discuss the commutation between the smoothed toroidal and flux-surface average in Section 3.3.

#### 3.1. Toroidal averages

First, we investigate the behavior of the smoothed toroidal average Eq. (12). In this section we use a magnetic field defined by Eq. (3) and [31]

$$\psi_p = \cos[\pi(R - R_0)/2] \cos(\pi Z/2) \quad I(\psi_p) = I_0 \quad (40)$$

with  $R_0 = 3$  and  $I_0 = 10$ . We start by creating a hypothetical simulation result; a field-aligned function modulated along the toroidal direction

$$f(R, Z, \varphi) = F(R, Z, \varphi) \exp\left(-\frac{\varphi^2}{2\sigma_\varphi^2}\right) \quad (41)$$

where  $F$  is a field-aligned function, invariant under the field line transformations

$$\mathcal{T}_{+\Delta\varphi} F(\mathbf{z}) = \mathcal{T}_{-\Delta\varphi} F(\mathbf{z}) = F(\mathbf{z}) \quad (42)$$

We can use these relations to numerically construct aligned structures by active transformations of a given field. Our idea is to initialize a two-dimensional field  $F(R, Z, \varphi_0)$  in a given plane  $k=0$  and transform this field to all other planes using the recursive relations [11]

$$\begin{aligned} F(R, Z, \varphi_{k+1}) &= \mathcal{T}_{-\Delta\varphi} F(R, Z, \varphi_{k+1}) = \mathcal{T}_{-\Delta\varphi} F_k(R, Z), \\ F(R, Z, \varphi_{k-1}) &= \mathcal{T}_{-\Delta\varphi} F(R, Z, \varphi_{k-1}) = \mathcal{T}_{+\Delta\varphi} F_k(R, Z), \end{aligned}$$

which is the statement that the two-dimensional function  $F_k$  in a plane  $k$  can be pulled back from the next/previous plane along the streamlines of  $\mathbf{B}/B^\varphi$  to obtain  $F_{k\pm 1}$ . Note here that  $\mathcal{T}_{\pm\Delta\varphi} f(R, Z, \varphi_k) = \mathcal{T}_{\pm\Delta\varphi} f_{k\pm 1}(R, Z) \neq \mathcal{T}_{\pm\Delta\varphi} f_k(R, Z)$ .

In Fig. 3 we plot the result of such a process with

$$F(R, Z, 0) := B + A \exp\left(-\frac{(R - R_0)^2 + (Z - Z_0)^2}{2\sigma^2}\right) \quad (43)$$

with  $B = 0.2$ ,  $A = 4$ ,  $R_0 = 3.5$ ,  $Z_0 = 0$ ,  $\sigma = 0.05$  and  $\sigma_\varphi = 0.4\pi$ . Such a function is representative of an almost field-aligned structure with a parallel gradient that is much smaller than its perpendicular gradient.

We now discretize Eq. (41) with  $N_\varphi = 10$  toroidal planes and compute the simple toroidal average Eq. (11). The result is plotted in Fig. 4. We see that the toroidal average in such a situation is clearly under-resolved and shows corrugations in the two-dimensional plot. This behavior was previously reported as spurious oscillatory fields in Reference [11]. Starting from this result we can now apply our post-processing formula Eq. (11). We chose  $K_\varphi = 10$  and show the result in Fig. 5. Clearly, the corrugations vanish and the result is closer to what we expect the toroidal average of Fig. 3 to look like. In order to also

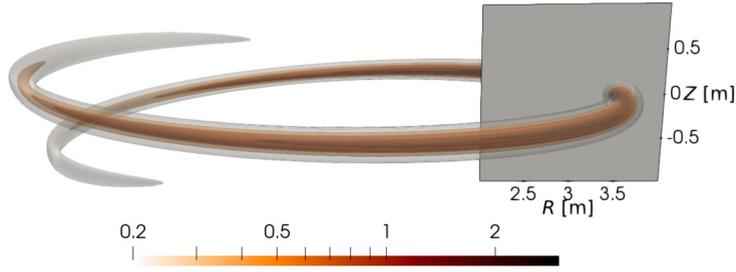


Fig. 3. Contour plot of the initial function Eq. (41) in three dimensions. The grey box shows the plotting area of the following two-dimensional plots.

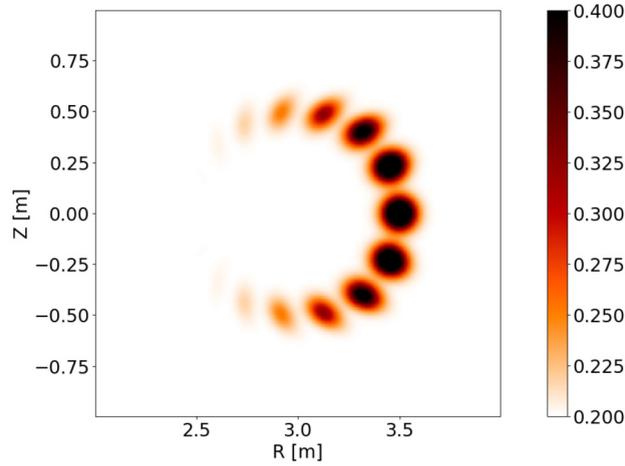


Fig. 4. Plot of the simple toroidal average  $\langle f \rangle_\varphi^T$  with  $N_\varphi = 10$  planes. The toroidal direction is under-resolved.

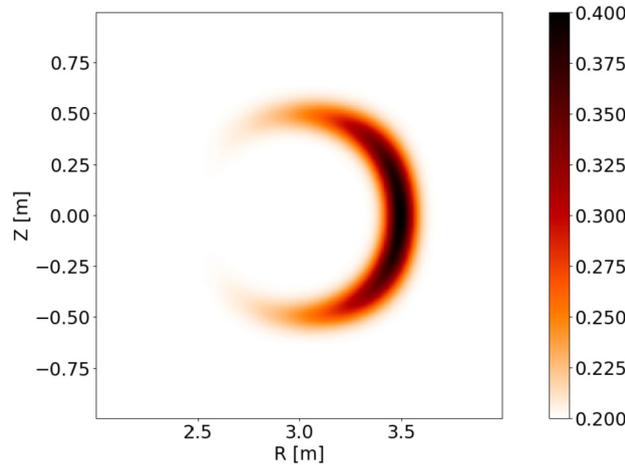


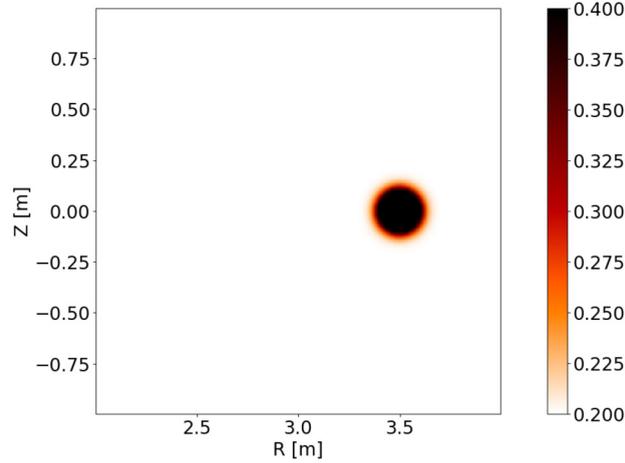
Fig. 5. Plot of the smooth toroidal average  $\langle f \rangle_\varphi^{ST}$  with  $N_\varphi = K_\varphi = 10$ . With the field-aligned assumption we can fill the missing values and obtain a smooth picture.

formally show that Eq. (12) yields a more accurate result than the simple average Eq. (11) we now compare the two results to a highly resolved case with  $N_\varphi = 200$ . We compute the relative error in the  $L_2$  norm and show the results in Table 1. We find that indeed for low resolution in  $\varphi$ , that is for  $N_\varphi \leq 20$ , the smoothed average  $\langle f \rangle_\varphi^{ST}$  yields a superior result than  $\langle f \rangle_\varphi^T$ . For higher resolutions however, the simple average  $\langle f \rangle_\varphi^T$  converges much faster than  $\langle f \rangle_\varphi^{ST}$ , which even slows down in convergence. We reason the difference in behavior in the perpendicular resolution. The convergence of  $\langle f \rangle_\varphi^T$  is entirely independent of  $N_R$  and  $N_Z$  and only depends on  $N_\varphi$ . On the other side the evaluation of  $\langle f \rangle_\varphi^{ST}$  requires interpolations in the  $R$  and  $Z$  directions. If the perpendicular resolution in  $R$  and  $Z$  is kept constant this error will eventually dominate.

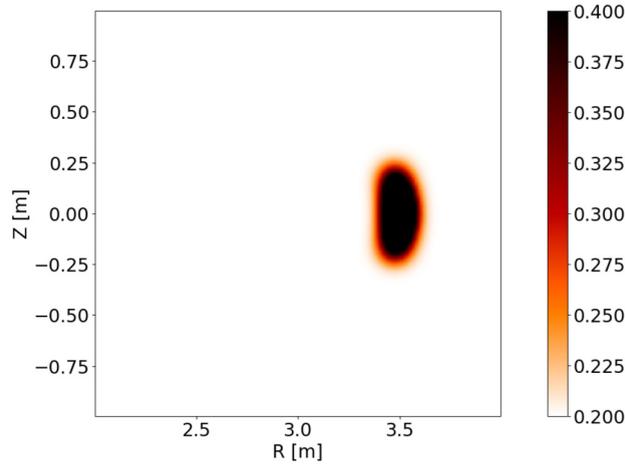
**Table 1**

Convergence table of the simple toroidal average and the smoothed toroidal average applied to the field-aligned  $f$  Eq. (41) with  $K_\varphi = 10$  for various values of  $N_\varphi$  and constant values of  $N_R = N_Z = 150$ .

$N_\varphi$	$\langle f \rangle_\varphi^T$ Eq. (11)		$\langle f \rangle_\varphi^{ST}$ Eq. (12)	
	error	order	error	order
<b>5</b>	1.32e+00	n/a	7.06e-02	n/a
<b>10</b>	6.20e-01	1.09	1.76e-02	2.01
<b>20</b>	5.60e-02	3.47	4.40e-03	2.00
<b>40</b>	1.06e-05	12.37	1.30e-03	1.76
<b>80</b>	1.52e-07	6.12	6.88e-04	0.92



**Fig. 6.** Plot of the simple toroidal average of the toroidally aligned function Eq. (44) with  $N_\varphi = 10$ .



**Fig. 7.** Plot of the smoothed toroidal average of the toroidally aligned function Eq. (44) with  $N_\varphi = K_\varphi = 10$ . The smoothing operation smears out the blob on the flux-surface.

It is interesting to also apply the smoothed toroidal average  $\langle f \rangle_\varphi^{ST}$  to a function that is not field-aligned but toroidally aligned. We change the definition of  $f$  to

$$f(R, Z, \varphi) = F(R, Z, 0) \exp\left(-\frac{\varphi^2}{2\sigma_\varphi^2}\right) \tag{44}$$

We plot the result of  $\langle f \rangle_\varphi^T$  in Fig. 6 and the result of  $\langle f \rangle_\varphi^{ST}$  in Fig. 7. We find that in this case the simple average  $\langle f \rangle_\varphi^T$  gives the better result than  $\langle f \rangle_\varphi^{ST}$ . By the partial flux surface average inherent in the smoothing operator the Gaussian shape is smeared out. Formally, we compare the two algorithms in Table 2, with a reference solution using  $N_\varphi = 400$  and a relative

**Table 2**

Convergence table of the simple toroidal average and the smoothed toroidal average applied to the toroidally aligned  $f$  Eq. (44) with  $K_\varphi = 10$  for various values of  $N_\varphi$  and constant values of  $N_R = N_Z = 150$ .

$N_\varphi$	$\langle f \rangle_\varphi^T$ Eq. (11)		$\langle f \rangle_\varphi^{ST}$ Eq. (12)	
	error	order	error	order
<b>5</b>	3.33e-03	n/a	7.66e-01	n/a
<b>10</b>	9.02e-04	1.88	5.62e-01	0.45
<b>20</b>	2.29e-04	1.98	2.83e-01	0.99
<b>40</b>	5.71e-05	2.00	9.49e-02	1.58
<b>80</b>	1.39e-05	2.04	2.59e-02	1.87

error in the  $L_2$  norm. We find that both methods do converge but that the error constant of  $\langle f \rangle_\varphi^T$  is much smaller than  $\langle f \rangle_\varphi^{ST}$  in this case.

In conclusion we can say that the smoothed toroidal average  $\langle f \rangle_\varphi^{ST}$  is useful as a post-processing tool for cases where the toroidal direction is under-resolved and the function it is applied to is (close to) field-aligned. In all other cases the simple toroidal average  $\langle f \rangle_\varphi^T$  is superior, in particular when the function is not field-aligned. As a rule-of-thumb we find that if corrugations are visible in two-dimensional plots of  $\langle f \rangle_\varphi^T$  then the smoothed toroidal average is applicable and yields superior results.

### 3.2. Line and surface integrals and the flux surface average

We discuss two examples of which we know the analytical solution of a flux-surface average. In Section 3.2.1 we discuss the q-profile of the magnetic field and the divergence of the curvature operator  $\mathbf{K}$ , which should vanish under the flux surface average. Thereafter, we construct a simple arc-length and area test without X-points in Section 3.2.2. Finally, we discuss integrals and derivatives with respect to the volume flux-label in Section 3.2.3.

#### 3.2.1. Convergence of the flux-surface average

Our first test of accuracy for the numerical flux-surface average is the so-called *safety factor*. Assume that we pick a random field line in Eq. (6) and follow it (integrate it) for exactly one poloidal turn. The safety factor is defined as the ratio between the resulting toroidal angle ( $\Delta\varphi$ ) to the poloidal angle ( $2\pi$ ) [2]

$$q := \frac{\Delta\varphi}{2\pi} \tag{45}$$

Since our magnetic field is symmetric in  $\varphi$  and we used one full poloidal turn this definition is independent of which fieldline we pick on a given flux surface. Also notice that with this definition  $q$  can be negative.

We define the geometric poloidal angle  $\Theta$  as the field-line following parameter around the O-point at  $(R_0, Z_0)$

$$\Theta = \begin{cases} +\arccos[(R - R_0)/r] & \text{for } R \geq R_0 \\ -\arccos[(R - R_0)/r] & \text{for } R < R_0 \end{cases}$$

with  $r^2 := (R - R_0)^2 + (Z - Z_0)^2$ . We then have with  $\mathbf{B}$  given by Eq. (3)  $B^\Theta = \mathbf{B} \cdot \nabla\Theta = -R_0(\psi_R(R - R_0) + \psi_Z(Z - Z_0))/(r^2R)$ . We can then directly integrate any field-line as

$$\frac{dR}{d\Theta} = \frac{B^R}{B^\Theta} \quad \frac{dZ}{d\Theta} = \frac{B^Z}{B^\Theta} \quad \frac{d\varphi}{d\Theta} = \frac{B^\varphi}{B^\Theta}$$

from  $\Theta = 0$  to  $\Theta = 2\pi$ . The safety factor results via

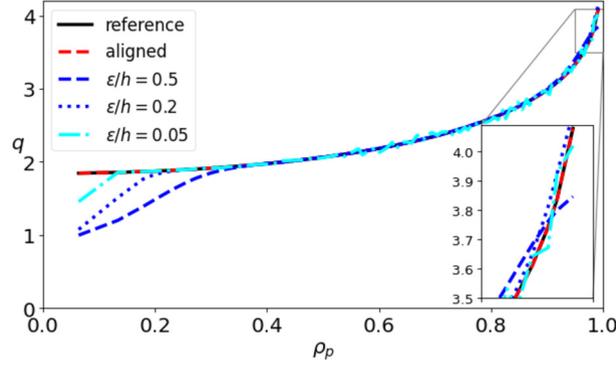
$$q \equiv \frac{1}{2\pi} \oint \frac{B^\varphi}{B^\Theta} d\Theta \tag{46}$$

Alternatively, the safety factor (46) can be formulated in terms of a flux surface average [2]

$$q(\psi_p) = \frac{1}{(2\pi)^2 R_0} \frac{dv}{d\psi_p} \langle B^\varphi \rangle \tag{47}$$

This yields a test for the flux-surface average. Eq. (46) can be computed to machine precision with the help of one of the high order ODE integrators in the FELTOR library [24] and thus can count as a reference solution with which we can compare the implementations of Eq. (47) in terms of delta-functions and the grid-transform method.

In Fig. 8 we plot the result of such a test. We choose a resolution of  $N_R = N_Z = 240$  points in the  $R$ - $Z$  plane on which we discretize  $B^\varphi$  and use the same  $\psi_p$  as in Fig. 2. We plot the highly accurate direct integration of Eq. (46) together with



**Fig. 8.** Plot of the safety factor for various methods as a function of the normalized poloidal flux  $\rho_p = \sqrt{(\psi_p - \psi_{p,0})/\psi_{p,0}}$  with  $\psi_p$  from Fig. 2. The black solid line is the (high accuracy) reference solution, the red dashed line is the result of Eq. (47) using Eq. (31) with  $N_\eta = 1920$ . The remaining lines represent Eq. (47) based on Eq. (30) using various values of  $\varepsilon$  based on  $h = \frac{1}{2}(h_R \max|\partial\psi_p/\partial R| + h_Z \max|\partial\psi_p/\partial Z|)$ , where  $h_R$  and  $h_Z$  are the grid distances in  $R$  and  $Z$ . We choose  $N_R = N_Z = 240$ . The enlarged box shows the region  $0.95 \leq \rho_p \leq 1$ .

**Table 3**

Safety factor convergence table with constant values of  $N_R = N_Z = 240$  and  $N_\psi = 96$ .

$N_\eta$	q-profile	
	error	order
<b>480</b>	5.25e-03	n/a
<b>960</b>	7.15e-04	2.88
<b>1920</b>	3.77e-05	4.24
<b>3840</b>	5.02e-06	2.91
<b>7680</b>	3.99e-06	0.33
<b>15360</b>	3.99e-06	-0.00

Eq. (47) for three different values of  $\varepsilon$  and one value of  $N_\eta = 1920$ , for a fixed number of points in  $\psi_p$ . Note that the safety factor diverges on the last closed flux surface (LCFS) at  $\rho_p = 1$  and remains undefined outside of it. We immediately see that the delta-function method cannot accurately represent the region around the O-point ( $\rho_p = 0$ ) and to a lesser degree also around the X-point (visible in the zoomed-in region). This is because of the finite extension of the approximate delta-function Eq. (18), which causes averaging effects close to the domain boundaries. The effect becomes smaller if the radius of the delta-function is decreased. However, we see that the delta-function method suffers from the fact that the  $\varepsilon$  value cannot be chosen arbitrarily small. This is because the delta-function needs to be numerically resolved on the given grid. If  $\varepsilon$  is chosen smaller than the resolution allows, oscillations appear (seen for  $\varepsilon/h = 0.05$ , cyan dash-dotted line).

An idea to remedy this problem is to interpolate the given function onto a finer grid with better resolution on which the fine delta-function can be resolved. However, recall from the end of Section 2.3 that the computational requirement for the delta-function method increases with  $N_R N_Z$  and the resolution needs to be increased in both  $R$  and  $Z$ . Furthermore, the cost for the interpolation is added to the already high computational cost of the delta-function approach. As pointed out at the end of Section 2.3 it is more efficient to use the grid-transform method, where the computational requirement only increases linearly in the poloidal resolution  $N_\eta$ .

In fact, in Fig. 8 we do not see visible differences between the grid-transform solution and the high-accuracy solution of Eq. (46). We therefore present the quantitative error defined as the relative error in the  $L_2$  norm over  $\rho_p$  in Table 3. There, we see that even for small values of  $N_\eta$  the error is below a range that we can possibly discern in a visual plot. Furthermore, we explicitly show that above a critical resolution in  $\eta$  the distance to the true solution does not further increase. This is the point where we reach the limit of the underlying given data. Further improvement can only be gained if the underlying resolution is increased.

Another possible test of the flux-surface average is the vanishing divergence of the curvature operator  $\mathbf{K} := \nabla \times (\hat{\mathbf{e}}_\varphi/B)$  that is  $\nabla \cdot \mathbf{K} = 0$ , from which follows

$$\langle \mathbf{K} \cdot \nabla \psi_p \rangle = 0 \quad (48)$$

In Fig. 9 we plot the result for the same parameters as in the previous Fig. 8 (except for the smallest  $\varepsilon$  value, which triggered too large oscillations and a  $\psi_p$  with only 1 X-point instead of 2). We see that the delta-function approach deteriorates close to the separatrix and that again oscillations appear for the small value  $\varepsilon/h = 0.1$ . The finite extension of the numerical delta-function introduces a smoothing effect over neighboring contour lines. The grid-transform method has a significantly lower error.

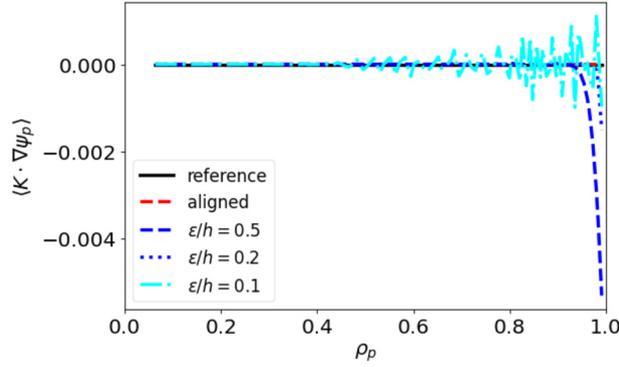


Fig. 9. Plot of Eq. (48) as a function of normalized poloidal flux  $\rho_p = \sqrt{(\psi_p - \psi_{p,0})/\psi_{p,0}}$  with a  $\psi_p$  with 1 X-point. The analytical solution in black is zero. Labels and methods analogous to Fig. 8.

Table 4

Convergence of the arc-length and area of the ellipse given by  $\psi_p^{\text{ellipse}} = 0$  (49) using Gauss-Legendre integration with 3 polynomial coefficients in Eq. (25) and (26). We use  $N_\eta = 10N_\psi$ .

$N_\psi$	$\int_{\text{ellipse}} ds$		$\int_{\text{ellipse}} dA$	
	error	order	error	order
6	2.78e-04	n/a	1.01e-06	n/a
12	2.56e-07	10.08	6.32e-08	3.99
24	1.38e-08	4.22	1.42e-09	5.48
48	4.20e-10	5.04	5.23e-13	11.40
96	2.67e-11	3.97	2.51e-15	7.70

As a final test we measure the time it takes to compute the curves in Fig. 9 on a single core of a Intel(R) Xeon(R) W-2133 CPU. The aligned curve took  $1.29 \cdot 10^{-3}$  s to compute while the three  $\epsilon/h$  curves took on average  $1.21 \cdot 10^{-1}$  s to compute, which is slower by a factor 90. From our performance estimate at the end of Section 2.3 we expect a speedup of only  $N_R N_Z / N_\eta = 240^2 / 1920 = 30$  of the grid-transform method compared to the delta-function approach. However, this assumes that the implementation is perfectly memory bandwidth bound, which may not be the case for our implementation. Furthermore, these timings should be contrasted with the computation of the flux-aligned grid itself, which took 25 s. Here, we note however that we compute the flux-aligned grid to machine precision, which overly inflates the computation time and may not be necessary in practice. The latter time can be amortized over  $25/0.121 = 200$  flux surface average computations.

### 3.2.2. Line and area integrals

The argument could be made that the poor results of the delta-function approach compared to the grid-transform method in Section 3.2.1 are due to our too simplistic representation in Eq. (18). In this section we thus want to compare the grid-transform method to more sophisticated representations of the delta-function. We here apply the grid-transform method to a test presented in References [20,22] who use it to show the performance of their delta-function approaches. To this end we choose  $\psi_p$  of the form

$$\psi_p^{\text{ellipse}}(R, Z) := 1 - \left(\frac{R - R_0}{3/2}\right)^2 - \left(\frac{Z}{1/2}\right)^2 \tag{49}$$

The iso-contour  $\psi_p^{\text{ellipse}} = 0$  in  $R, Z$  forms an ellipse whose arc-length is given by the numeric value 7.26633616541076 and whose enclosed area is given by  $(9/8)\pi$  [20,22]. We show the result of the grid-transform method to compute the arc-length (25) and area (26) using  $f = 1$  in Table 4. Relative errors are shown. We see irregular convergence behavior of order at least 4 in both quantities. The error of the area is smaller than that of the arc length and for  $N_\psi = 96$  is close to machine precision.

With  $N_\eta = 10N_\psi$  we have that  $N_\psi = 6$  corresponds to a grid that resolves the arc-length at about  $h = 0.1$  (the mesh size used in [20,22]). Note here that with second order polynomials from the Gauss-Legendre integration we have three coefficients per cell. Higher order methods can easily be constructed using higher order polynomials. We thus conclude that the grid-transform method yields errors several orders of magnitude smaller than the delta-function approaches in [20,22] and converges with more than twice the convergence rate.

In the remaining part of this section we only use the grid-transform method to compute integrals.

**Table 5**  
 Comparison of volume integration with flux surface average volume integration for  $N_R = N_Z = 150$  and toroidally symmetric  $f$  Eq. (44) with  $\sigma_\varphi = \infty$  (left). Volume error without X-point (right). We use  $N_\eta = 10N_\psi$  and  $n = 3$  in the Gauss-Legendre integration for  $\psi_p$  given in Eq. (40).

$N_\psi$	$\int \langle f \rangle dv$ (incl. X-point)		$\int dv$ (excl. X-point)	
	error	order	error	order
<b>48</b>	1.22e-03	n/a	1.16e-08	n/a
<b>96</b>	6.53e-04	0.90	2.12e-10	5.78
<b>192</b>	3.24e-04	1.01	3.44e-12	5.94
<b>384</b>	1.63e-04	0.99	4.33e-14	6.31

### 3.2.3. Volume integrals and derivatives

The flux surface average depends on the flux function  $\psi_p$  and so a volume derivative as mandated by the conservation equation (37) reads

$$\frac{d}{dv} \langle f \rangle = \left( \frac{dv}{d\psi_p} \right)^{-1} \frac{d}{d\psi_p} \langle f \rangle \tag{50}$$

and the volume integral

$$\int \langle f \rangle dv = \int \langle f \rangle \frac{dv}{d\psi_p} d\psi_p \tag{51}$$

Numerically, the derivative of the volume flux label  $dv/d\psi_p$  is easily obtained via Eq. (28).

Once the flux-surface average is discretized on a given grid in  $\psi_p$  we can derive and/or integrate with respect to the volume flux-label. We here stay within a discontinuous Galerkin framework for high order derivative and integration formulas [32] but note that any other consistent discretization of derivatives is equally valid.

We test this by first testing the volume integration of the toroidally symmetric  $f$  Eq. (44) ( $\sigma_\varphi = \infty$ ) with  $\psi_p$  given by Eq. (40). The total volume integral in the domain  $[2, 4] \times [-1, 1]$  can be computed via Gauss-Legendre integration on the original grid without involving the flux surface average, i.e.  $\int f dV$  can be computed to high accuracy and yields a test for  $\int \langle f \rangle dv = \int_0^1 \langle f \rangle (dv/d\psi_p) d\psi_p$ . The resulting relative errors are shown in Table 5 in the left columns. We observe that the convergence rate tends to first order convergence even though we expected a 6th order convergence due to the use of Gauss-Legendre integration in the  $(\zeta, \eta)$  space. This reduction of convergence was observed in [8] and is due to the presence of the X-points in the four corners of the domain. The diverging volume element only allows reduced convergence rates.

The 6th order convergence is recovered if we stop the volume integration before we reach the X-points, i.e. we choose  $\int_c^1 \langle f \rangle (dv/d\psi_p) d\psi_p$  with  $c > 0$  to avoid the separatrix at  $\psi_p = 0$  in Eq. (40). Unfortunately, we cannot compute an accurate solution for a non-trivial  $f$  so that for this test we choose  $f = 1$ . Then  $\int_c^1 dv = 4\pi \int_{R_0-R_c}^{R_0+R_c} Z(R) R dR$  with  $Z(R) = (2/\pi) \arccos(c/\cos(\pi(R-R_0)/2))$  and  $R_c = (2/\pi) \arccos(c)$ ,  $c = 1/7$ . The resulting relative errors and orders are shown in Table 5 in the right columns and shows approximately 6-th order convergence as expected.

In order to test the volume derivative we use the identity

$$\langle \Delta \psi_p \rangle = \frac{\partial}{\partial v} \left( \frac{dv}{d\psi_p} \langle |\nabla \psi_p|^2 \rangle \right) \tag{52}$$

which directly follows from Eq. (36). A convergence test with increasing number of points in  $\psi_p$  is shown in Table 6 for a  $\psi_p$  with 2 X-points.  $N_\eta = 10N_\psi$ . We show relative errors in the  $L_2$  norm. Again, we find approximately first order convergence even though we could expect second order from the used discretization of the derivative. We attribute this again to the diverging volume close to the X-points.

In the same Table 6 we show results if the X-points are avoided similarly to the previous example by restricting the grid to a maximum  $\psi_p < 0$  value (i.e. avoiding the separatrix in Fig. 2). Then the expected second order convergence is recovered.

### 3.3. Smoothed toroidal and flux-surface average

Here, we test if the flux surface average applied to  $f$  gives a different result to  $S_{\Delta\varphi}(f)$  that is we want to numerically test Eq. (33). In Table 7 we show the relative error for different numbers of toroidal planes. We see that the error is fairly small even for the very low resolution of only 5 toroidal planes and converges with 2nd order. It is in fact not needed to apply the smoothing procedure to the toroidal average  $\langle f \rangle_\varphi^T$  if one is only interested in the flux-surface average in Eq. (31).

**Table 6**

Comparison of volume derivative with flux surface average for  $\psi_p$  with 2 X-points as in Fig. 2. Approximately first order convergence is shown if values up to the X-points are computed (left). Second order is retained if the X-point is avoided (right).

$N_\psi$	$\langle \Delta\psi_p \rangle$ (incl. X-point)		$\langle \Delta\psi_p \rangle$ (excl. X-point)	
	error	order	error	order
<b>12</b>	9.09e-03	n/a	4.31e-04	n/a
<b>24</b>	1.46e-03	2.64	1.05e-04	2.03
<b>48</b>	6.63e-04	1.14	2.83e-05	1.89
<b>96</b>	4.96e-04	0.42	7.91e-06	1.84
<b>192</b>	2.98e-04	0.74	2.11e-06	1.91
<b>384</b>	9.43e-05	1.66	5.43e-07	1.96

**Table 7**

Comparison of the smoothed vs the normal flux surface average on the field (40) for constant values  $N_R = N_Z = 150$ ,  $N_\eta = 1920$ ,  $N_\psi = 192$ .  $f$  is given by Eq. (44).

$N_\phi$	$\langle f - S(f) \rangle / \langle f \rangle$	
	error	order
<b>5</b>	2.06e-03	n/a
<b>10</b>	5.36e-04	1.94
<b>20</b>	1.51e-04	1.83
<b>40</b>	3.83e-05	1.97

## 4. Conclusions

In this contribution we suggest a smoothed toroidal average as a way to improve toroidal averages in simulations based on the flux-coordinate independent approach. The resulting formula (12) is easily applied in post-processing if the simple toroidal sum is stored during the simulation. We show that the algorithm can be interpreted as a partial surface average. Furthermore, we show the connection to the full flux-surface average in Eq. (33). In the presented numerical tests it is shown that the smoothed toroidal average leads to superior results to the simple toroidal average of field-aligned quantities.

Furthermore, we investigate flux-surface averages as well as general line and surface integrals on non-aligned grids by transforming quantities onto a flux-aligned grid. Numerical grid generators previously constructed in [23,8] can be used. The quantity to integrate can be transformed to the grid via simple interpolation. Once interpolated, the flux-surface average (31), the line integral (25) and the surface integral (24) can readily be computed in the flux-aligned coordinates. Our method is relevant for level set methods as well as three-dimensional simulations of plasma dynamics on non-aligned grids, in particular those using the FCI approach.

We compare the grid-transform method to one based on an approximation of the delta-function in an expression of the surface integral as a volume integral over the entire domain (30). It is shown in this paper that the grid-transform method is highly accurate and superior to the delta-function method in every quality measure. Due to the necessarily finite extension of the numerical delta-function the accuracy of the associated flux-surface average is lost close to the O-point and the separatrix. This problem is absent in the grid-transform method. A test of line and area integrals of an ellipse is shown. The grid-transform method shows far better error constants and order of convergence than methods proposed previously in the literature. Volume derivatives and integrals necessary in flux-surface averaged conservation laws can be accurately computed. However, the diverging volume element close to X-points reduces the order of convergence to 1 in those cases.

The methods shown in this work are applied to level-sets of toroidally symmetric functions. An extension to general non-toroidally symmetric surfaces in three-dimensions depends on a suitable and fast method to compute aligned grid coordinates but is otherwise immediate.

### CRedit authorship contribution statement

**Matthias Wiesenberger:** Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Raul Gerrú:** Investigation, Methodology, Software, Validation, Writing – review & editing. **Markus Held:** Conceptualization, Formal analysis, Investigation, Methodology, Software, Validation, Writing – review & editing.

### Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Matthias Wiesenberger reports financial support was provided by European Consortium for the Development of Fusion Energy. Markus Held reports financial support was provided by Austrian Science Fund.

## Data availability

See Appendix A in the main text.

## Acknowledgements

This work has been carried out within the framework of the EUROfusion Consortium, funded by the European Union via the Euratom Research and Training Programme (Grant Agreement No 101052200 – EUROfusion). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the European Commission can be held responsible for them. This work was supported by the UiT Aurora Centre Program, UiT The Arctic University of Norway (2020). This research was funded in whole or in part by the Austrian Science Fund (FWF) [P 34241-N]. For the purpose of Open Access, the author has applied a CC BY public copyright license to any Author Accepted Manuscript (AAM) version arising from this submission.

## Appendix A. Data access

All figures and tables in this paper can be reproduced using the Jupyter Notebooks within the dataset <https://github.com/mwiesenberger/averages>. The necessary binary data was generated with the fully parallelized, bitwise reproducible and accurate FELTOR library [24,25]. The library is freely available and interoperates readily with Python and the netCDF data format through our `simplesimdb` Python package available at <https://pypi.org/project/simplesimdb/>. Moreover, there is ample documentation available on our website <https://feltor-dev.github.io>

## References

- [1] F.L. Hinton, R.D. Hazeltine, Theory of plasma transport in toroidal confinement systems, *Rev. Mod. Phys.* 48 (1976) 239–308, <https://doi.org/10.1103/RevModPhys.48.239>.
- [2] W. D'haeseleer, W. Hitchon, J. Callen, J. Shohet, *Flux Coordinates and Magnetic Field Structure*, Springer Series in Computational Physics, Springer-Verlag, 1991.
- [3] J. Wesson, D.J. Campbell, *Tokamaks*, vol. 149, Oxford University Press, 2011.
- [4] T. Frankel, *The Geometry of Physics: An Introduction*, 2nd edition, Cambridge University Press, 2004.
- [5] R.L. Dewar, A.H. Glasser, Ballooning mode spectrum in general toroidal systems, *Phys. Fluids* 26 (10) (1983) 3038–3052, <https://doi.org/10.1063/1.864028>.
- [6] M.A. Beer, S.C. Cowley, G.W. Hammett, Field-aligned coordinates for nonlinear simulations of tokamak turbulence, *Phys. Plasmas* 2 (7) (1995) 2687–2700, <https://doi.org/10.1063/1.871232>.
- [7] B. Scott, Shifted metric procedure for flux tube treatments of toroidal geometry: avoiding grid deformation, *Phys. Plasmas* 8 (2001) 447, <https://doi.org/10.1063/1.1335832>.
- [8] M. Wiesenberger, M. Held, L. Einkemmer, A. Kendl, Streamline integration as a method for structured grid generation in x-point geometry, *J. Comput. Phys.* 373 (2018) 370–384, <https://doi.org/10.1016/j.jcp.2018.07.007>.
- [9] P. Paruta, P. Ricci, F. Riva, C. Wersal, C. Beadle, B. Frei, Simulation of plasma turbulence in the periphery of diverted tokamak by using the gbs code, *Phys. Plasmas* 25 (11) (2018) 112301.
- [10] M.R. Dorr, P. Colella, M.A. Dorf, D. Ghosh, J.A. Hittinger, P.O. Schwartz, High-order discretization of a gyrokinetic Vlasov model in edge plasma geometry, *J. Comput. Phys.* 373 (2018) 605–630, <https://doi.org/10.1016/j.jcp.2018.07.008>.
- [11] M. Held, M. Wiesenberger, A. Stegmeir, Three discontinuous Galerkin schemes for the anisotropic heat conduction equation on non-aligned grids, *Comput. Phys. Commun.* 199 (2016) 29–39, <https://doi.org/10.1016/j.cpc.2015.10.009>.
- [12] A. Stegmeir, T. Body, W. Zholobenko, Analysis of locally-aligned and non-aligned discretisation schemes for reactor-scale tokamak edge turbulence simulations, *Comput. Phys. Commun.* (2023) 108801, <https://doi.org/10.1016/j.cpc.2023.108801>.
- [13] F. Hariri, M. Ottaviani, A flux-coordinate independent field-aligned approach to plasma turbulence simulations, *Comput. Phys. Commun.* 184 (11) (2013) 2419–2429, <https://doi.org/10.1016/j.cpc.2013.06.005>.
- [14] A. Stegmeir, O. Maj, D. Coster, K. Lackner, M. Held, M. Wiesenberger, Advances in the flux-coordinate independent approach, *Comput. Phys. Commun.* 213 (2017) 111–121, <https://doi.org/10.1016/j.cpc.2016.12.014>.
- [15] M. Held, Full-f gyro-fluid modelling of the tokamak edge and scrape-off layer, Ph.D. thesis, University of Innsbruck, 2016, <http://resolver.obvsg.at/urn:nbn:at:at-ubi:1-6853>.
- [16] M. Wiesenberger, Gyrofluid computations of filament dynamics in tokamak scrape-off layers, Ph.D. thesis, University of Innsbruck, 2014, <http://resolver.obvsg.at/urn:nbn:at:at-ubi:1-1799>.
- [17] A. Stegmeir, D. Coster, A. Ross, O. Maj, K. Lackner, E. Poli, Grillix: a 3d turbulence code based on the flux-coordinate independent approach, *Plasma Phys. Control. Fusion* 60 (3) (2018) 035005.
- [18] W. Zholobenko, A. Stegmeir, T. Body, A. Ross, P. Manz, O. Maj, D. Coster, F. Jenko, M. Francisquez, B. Zhu, B. Rogers, Thermal dynamics in the flux-coordinate independent turbulence code grillix, *Contrib. Plasma Phys.* 60 (5–6) (2020), <https://doi.org/10.1002/ctpp.201900131> e201900131, <https://onlinelibrary.wiley.com/doi/pdf/10.1002/ctpp.201900131>.
- [19] S. Osher, R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer, 2003.
- [20] P. Smerka, The numerical approximation of a delta function with application to level set methods, *J. Comput. Phys.* 211 (1) (2006) 77–90, <https://doi.org/10.1016/j.jcp.2005.05.005>.
- [21] B. Engquist, A.-K. Tornberg, R. Tsai, Discretization of Dirac delta functions in level set methods, *J. Comput. Phys.* 207 (1) (2005) 28–51, <https://doi.org/10.1016/j.jcp.2004.09.018>.
- [22] B. Müller, F. Kummer, M. Oberlack, Y. Wang, Simple multidimensional integration of discontinuous functions with application to level set methods, *Int. J. Numer. Methods Eng.* 92 (7) (2012) 637–651, <https://doi.org/10.1002/nme.4353>, <https://onlinelibrary.wiley.com/doi/pdf/10.1002/nme.4353>.
- [23] M. Wiesenberger, M. Held, L. Einkemmer, Streamline integration as a method for two-dimensional elliptic grid generation, *J. Comput. Phys.* 340 (2017) 435–450, <https://doi.org/10.1016/j.jcp.2017.03.056>.
- [24] M. Wiesenberger, M. Held, Feltor v7.0, <https://doi.org/10.5281/zenodo.596442>, 2023.

- [25] M. Wiesenberger, L. Einkemmer, M. Held, A. Gutierrez-Milla, X. Saez, R. Iakymchuk, Reproducibility, accuracy and performance of the feltor code and library on parallel computer architectures, *Comput. Phys. Commun.* 238 (2019) 145–156, <https://doi.org/10.1016/j.cpc.2018.12.006>.
- [26] E. Hairer, S.P. Nørsett, G. Wanner, *Solving Ordinary Differential Equations I*, 1st edition, Springer Series in Computational Mathematics, Springer Berlin, Heidelberg, 1987.
- [27] A.I. Saichev, W.A. Woyczyński, *Distributions in the Physical and Engineering Sciences*, Birkhäuser, Boston, 1997.
- [28] B. Scott, J. Smirnov, Energetic consistency and momentum conservation in the gyrokinetic description of tokamak plasmas, *Phys. Plasmas* 17 (2010) 112302, <https://doi.org/10.1063/1.3507920>.
- [29] M. Wiesenberger, M. Held, Angular momentum and rotational energy of mean flows in toroidal magnetic fields, *Nucl. Fusion* 60 (9) (2020) 096018.
- [30] A.J. Cerfon, J.P. Freidberg, “One size fits all” analytic solutions to the grad-Shafranov equation, *Phys. Plasmas* 17 (3) (2010) 032502, <https://doi.org/10.1063/1.3328818>.
- [31] S. Günter, Q. Yu, J. Krüger, K. Lackner, Modelling of heat transport in magnetised plasmas using non-aligned coordinates, *J. Comput. Phys.* 209 (1) (2005) 354–370, <https://doi.org/10.1016/j.jcp.2005.03.021>.
- [32] J.S. Hesthaven, T. Warburton, *Nodal Discontinuous Galerkin Methods*, 1st edition, Springer, New York, NY, 2008.