bnorml

Computes $\mathbf{B}_P \cdot \mathbf{e}_\theta \times \mathbf{e}_\zeta$ on computational boundary, $\partial\mathcal{D}$.

[called by: xspech.] [calls: coords and casing.]

contents

1.1 free-boundary constraint

1. The normal field at the computational boundary, $\partial\mathcal{D}$, should be equal to $(\mathbf{B}_P + \mathbf{B}_C)\cdot\mathbf{n}$, where $\mathbf{B}_P$ is the "plasma" field (produced by internal plasma currents) and is computed using virtual casing, and $\mathbf{B}_C$ is the "vacuum" field (produced by the external coils) and is given on input.

1.2 construction of normal field

1. The normal vector to the computational domain is given as follows:

   `Igeometry.eq.1` : Cartesian
   $\mathbf{x} = \theta\ \hat{i} + \zeta\ \hat{j} + R(\theta,\zeta)\ \hat{k}$
   $\mathbf{e}_\theta \times \mathbf{e}_\zeta = -R_\theta\ \hat{i} - R_\zeta\ \hat{j} + \hat{k}$

   `Igeometry.eq.2` : Cylindrical

   `Igeometry.eq.3` : Toroidal
   $\mathbf{x} = R(\theta,\zeta)\cos\zeta\ \hat{i} + R(\theta,\zeta)\sin\zeta\ \hat{j} + Z(\theta,\zeta)\ \hat{k}$
   $\mathbf{e}_\theta \times \mathbf{e}_\zeta = -R\,Z_\theta\,\hat{r} + (Z_\theta\,R_\zeta - R_\theta\,Z_\zeta)\hat{\phi} + R\,R_\theta\,\hat{z}$

1.3 outline

1. The computational boundary is obtained using coords. (Note that the computational boundary does not change, so this needs only to be determined once.)

2. At each point on the computational boundary (i.e., on the discrete grid), casing is used to compute the plasma field using the virtual casing principle.

3. In toroidal geometry, the vector transformation from Cartesian to cylindrical is given by

$$\begin{array}{ccccccc} B^R & = & & +B_x\cos\zeta & + & B_y\sin\zeta & \\ B^\phi & = & (& -B_x\sin\zeta & + & B_y\cos\zeta &)/R \\ B^Z & = & & & & & B_z \end{array} \tag{1}$$

The surface integral is performed using NAG: D01DAF.

1.4 theory and numerics

1. Required inputs to this subroutine are the geometry of the plasma boundary,

$$\mathbf{x}(\theta,\zeta) \equiv x(\theta,\zeta)\mathbf{i} + y(\theta,\zeta)\mathbf{j} + z(\theta,\zeta)\mathbf{k}, \tag{2}$$

   and the tangential field on this boundary,

$$\mathbf{B}_s = B^\theta\mathbf{e}_\theta + B^\zeta\mathbf{e}_\zeta, \tag{3}$$

   where θ and ζ are arbitrary poloidal and toroidal angles, and $\mathbf{e}_\theta \equiv \partial\mathbf{x}/\partial\theta$, $\mathbf{e}_\zeta \equiv \partial\mathbf{x}/\partial\zeta$. This routine assumes that the plasma boundary is a flux surface, i.e. $\mathbf{B}\cdot\mathbf{e}_\theta \times \mathbf{e}_\zeta = 0$.

2. The virtual casing principle [Shafranov & Zakharov (1972)[1], Lazerson (2012)[2], Hanson (2015)[3]] shows that the field outside/inside the plasma arising from plasma currents inside/outside the boundary is equivalent to the field generated by a surface current,

$$\mathbf{j} = \mathbf{B}_s \times \mathbf{n}, \tag{4}$$

where $\mathbf{n}$ is normal to the surface.

3. The field at some arbitrary point, $\bar{\mathbf{x}}$, created by this surface current is given by

$$\mathbf{B}(\bar{\mathbf{x}}) = \int_{\mathcal{S}} \frac{(\mathbf{B}_s \times d\mathbf{s}) \times \hat{\mathbf{r}}}{r^2}, \tag{5}$$

where $d\mathbf{s} \equiv \mathbf{e}_\theta \times \mathbf{e}_\zeta \, d\theta d\zeta$.

4. For ease of notation introduce

$$\mathbf{J} \;\; \equiv \;\; \mathbf{B}_s \times d\mathbf{s} = \alpha \, \mathbf{e}_\theta - \beta \, \mathbf{e}_\zeta, \tag{6}$$

where $\alpha \equiv B_\zeta = B^\theta g_{\theta\zeta} + B^\zeta g_{\zeta\zeta}$ and $\beta \equiv B_\theta = B^\theta g_{\theta\theta} + B^\zeta g_{\theta\zeta}$,

5. We may write in Cartesian coordinates $\mathbf{J} = j_x \, \mathbf{i} + j_y \, \mathbf{j} + j_z \, \mathbf{k}$, where

$$j_x = \alpha \, x_\theta - \beta \, x_\zeta \tag{7}$$
$$j_y = \alpha \, y_\theta - \beta \, y_\zeta \tag{8}$$
$$j_z = \alpha \, z_\theta - \beta \, z_\zeta. \tag{9}$$

6. Requiring that the current,

$$\mathbf{j} \;\; \equiv \;\; \nabla \times \mathbf{B} = \sqrt{g}^{-1} (\partial_\theta B_\zeta - \partial_\zeta B_\theta) \, \mathbf{e}_s + \sqrt{g}^{-1} (\partial_\zeta B_s - \partial_s B_\zeta) \, \mathbf{e}_\theta + \sqrt{g}^{-1} (\partial_s B_\theta - \partial_\theta B_s) \, \mathbf{e}_\zeta, \tag{10}$$

has no normal component to the surface, i.e. $\mathbf{j} \cdot \nabla s = 0$, we obtain the condition $\partial_\theta B_\zeta = \partial_\zeta B_\theta$, or $\partial_\theta \alpha = \partial_\zeta \beta$. In axisymmetric configurations, where $\partial_\zeta \beta = 0$, we must have $\partial_\theta \alpha = 0$.

7. The displacement from an arbitrary point, (X, Y, Z), to a point, (x, y, z), that lies on the surface is given

$$\mathbf{r} \equiv r_x \, \mathbf{i} + r_y \, \mathbf{j} + r_z \, \mathbf{k} = (X - x) \, \mathbf{i} + (Y - y) \, \mathbf{j} + (Z - z) \, \mathbf{k}. \tag{11}$$

8. The components of the magnetic field produced by the surface current are then

$$B^x = \oint\!\!\oint d\theta d\zeta \;\; (j_y r_z - j_z r_y)/r^3, \tag{12}$$
$$B^y = \oint\!\!\oint d\theta d\zeta \;\; (j_z r_x - j_x r_z)/r^3, \tag{13}$$
$$B^z = \oint\!\!\oint d\theta d\zeta \;\; (j_x r_y - j_y r_x)/r^3 \tag{14}$$

9. The surface integral is performed using NAG: D01EAF, which uses an adaptive subdivision strategy and also computes absolute error estimates. The absolute and relative accuracy required are provided by the input `vcasingtol`. The minimum number of function evaluations is provided by the input `vcasingits`.

10. It may be convenient to have the derivatives:

$$\frac{\partial B^x}{\partial x} = \oint\!\!\oint d\theta d\zeta \;\; \left[-3 (j_y r_z - j_z r_y)(X - x)/r^5 \right], \tag{15}$$
$$\frac{\partial B^x}{\partial y} = \oint\!\!\oint d\theta d\zeta \;\; \left[-3 (j_y r_z - j_z r_y)(Y - y)/r^5 - j_z/r^3 \right], \tag{16}$$
$$\frac{\partial B^x}{\partial z} = \oint\!\!\oint d\theta d\zeta \;\; \left[-3 (j_y r_z - j_z r_y)(Z - z)/r^5 + j_y/r^3 \right], \tag{17}$$

[1] V.D. Shafranov & L.E. Zakharov, Nucl. Fusion **12**, 599 (1972)
[2] S.A. Lazerson, Plasma Phys. Control. Fusion **54**, 122002 (2012)
[3] J.D. Hanson, Plasma Phys. Control. Fusion **57**, 115006 (2015)

$$
\begin{aligned}
\frac{\partial B^y}{\partial x} &= \oint\!\!\oint d\theta d\zeta \;\; \left[-3(j_z r_x - j_x r_z)(X-x)/r^5 + j_z/r^3 \right], & (18)\\
\frac{\partial B^y}{\partial y} &= \oint\!\!\oint d\theta d\zeta \;\; \left[-3(j_z r_x - j_x r_z)(Y-y)/r^5 \qquad\qquad \right], & (19)\\
\frac{\partial B^y}{\partial z} &= \oint\!\!\oint d\theta d\zeta \;\; \left[-3(j_z r_x - j_x r_z)(Z-z)/r^5 - j_x/r^3 \right], & (20)\\
\frac{\partial B^z}{\partial x} &= \oint\!\!\oint d\theta d\zeta \;\; \left[-3(j_x r_y - j_y r_x)(X-x)/r^5 - j_y/r^3 \right], & (21)\\
\frac{\partial B^z}{\partial y} &= \oint\!\!\oint d\theta d\zeta \;\; \left[-3(j_x r_y - j_y r_x)(Y-y)/r^5 + j_x/r^3 \right], & (22)\\
\frac{\partial B^z}{\partial z} &= \oint\!\!\oint d\theta d\zeta \;\; \left[-3(j_x r_y - j_y r_x)(Z-z)/r^5 \qquad\qquad \right]. & (23)
\end{aligned}
$$

bnorml.h last modified on 7-07-20 15:11:44.00; SPEC subroutines;