

**preset**

Allocates and initializes internal arrays.

[called by: [xspech](#).]

[calls: [ra00aa](#).]

**contents**

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>preset</b>  | <b>1</b> |
| 1.1      | definition of internal variables . . . . .   | 1        |
| 1.1.1    | LGdof and NGdof : number of geometrical degrees-of-freedom; . . . . .                    | 1        |
| 1.1.2    | <b>iota</b> and <b>oita</b> : rotational transform on interfaces; . . . . .              | 1        |
| 1.1.3    | dtflux(1:Mvol) and dpflux(1:Mvol) : enclosed fluxes; . . . . .                           | 2        |
| 1.1.4    | sweight(1:Mvol) : star-like angle constraint weight; . . . . .                           | 2        |
| 1.1.5    | TT(0:Mrad,0:1,0:1) : Chebyshev polynomials at inner/outer interface; . . . . .           | 2        |
| 1.1.6    | ImagneticOK(1:Mvol) : Beltrami/vacuum error flag; . . . . .                              | 2        |
| 1.1.7    | Lhessianallocated . . . . .  | 2        |
| 1.1.8    | ki(1:mn,0:1) : Fourier identification; . . . . .   | 2        |
| 1.1.9    | kija(1:mn,1:mn,0:1), kijs(1:mn,1:mn,0:1) : Fourier identification; . . . . .             | 3        |
| 1.1.10   | djkp . . . . .   | 3        |
| 1.1.11   | iotakki . . . . .  | 3        |
| 1.1.12   | cheby(0:Lrad,0:2) : Chebyshev polynomial workspace; . . . . .                            | 3        |
| 1.1.13   | Iquad, gaussianweight, gaussianabscissae : Gaussian quadrature; . . . . .                | 3        |
| 1.1.14   | LBsequad, LBnewton and LBlinear . . . . .  | 3        |
| 1.1.15   | BBweight(1:mn) : weighting of force-imbalance harmonics . . . . .                        | 3        |
| 1.1.16   | mmpp(1:mn) : spectral condensation weight factors . . . . .                              | 3        |
| 1.1.17   | NAdof, Ate, Aze, Ato and Azo : degrees-of-freedom in magnetic vector potential . . . . . | 4        |
| 1.1.18   | workspace . . . . .  | 4        |
| 1.1.19   | goomne, goomno : metric information . . . . .  | 4        |
| 1.1.20   | gssmne, gssmno : metric information . . . . .  | 4        |
| 1.1.21   | gstmne, gstmno : metric information . . . . .  | 4        |
| 1.1.22   | gszmne, gszmno : metric information . . . . .  | 4        |
| 1.1.23   | gttmne, gttmno : metric information . . . . .  | 4        |
| 1.1.24   | gtzmne, gtzmno : metric information . . . . .  | 4        |
| 1.1.25   | gzzmne, gzzmno : metric information . . . . .  | 4        |
| 1.1.26   | cosi(1:Ntz,1:mn) and sini(1:Ntz,1:mn) . . . . .  | 4        |
| 1.1.27   | psifactor(1:mn,1:Mvol) : coordinate “pre-conditioning” factor . . . . .                  | 5        |
| 1.1.28   | Bsupumn and Bsupvmn . . . . .  | 5        |
| 1.1.29   | diotadxup and glambda : transformation to straight fieldline angle . . . . .             | 5        |
| 1.1.30   | vvolume, LBBintegral and LABintegral . . . . .   | 5        |

**1.1 definition of internal variables****1.1.1 LGdof and NGdof : number of geometrical degrees-of-freedom;**

1. LGdof  $\equiv$  the number of degrees-of-freedom in the geometry (i.e. Fourier harmonics) of each interface;
2. NGdof  $\equiv$  total number of degrees-of-freedom in geometry, i.e. of all interfaces;

**1.1.2 **iota** and **oita** : rotational transform on interfaces;**

1. The input variables **iota** and **oita** are the rotational transform on “inner-side” and on the “outer-side” of each interface;
2. These quantities are formally input
3. Note that if  $q_l + \gamma q_r \neq 0$ , then **iota** is given by

$$t \equiv \frac{p_l + \gamma p_r}{q_l + \gamma q_r}, \quad (1)$$

where  $p_l \equiv \text{p1}$ ,  $q_l \equiv \text{q1}$ , etc.; and similarly for **oita**.

### 1.1.3 dtflux(1:Mvol) and dpflux(1:Mvol) : enclosed fluxes;

1. dtflux  $\equiv \Delta\psi_{tor}/2\pi$  and dpflux  $\equiv \Delta\psi_{pol}/2\pi$  in each volume.
2. (Note that the total toroidal flux enclosed by the plasma boundary is  $\Phi_{edge} \equiv \text{phiedge.}$ )
3.  $\psi_{tor} \equiv \text{tflux}$  and  $\psi_{pol} \equiv \text{pflux}$  are immediately normalized (in **global**) according to  $\psi_{tor,i} \rightarrow \psi_{tor,i}/\psi_0$  and  $\psi_{pol,i} \rightarrow \psi_{pol,i}/\psi_0$ , where  $\psi_0 \equiv \psi_{tor,N}$  on input.

### 1.1.4 sweight(1:Mvol) : star-like angle constraint weight;

1. the “star-like” poloidal angle constraint weights (only required for toroidal geometry, i.e. **Igeometry=3**) are given by

$$\text{sweight}_v \equiv \text{upsilon} \times (l_v/N_{vol})^w, \quad (2)$$

where  $l_v$  is the volume number, and  $w \equiv \text{wpoloidal.}$

### 1.1.5 TT(0:Mrad,0:1,0:1) : Chebyshev polynomials at inner/outer interface;

1. TT(0:Lrad,0:1,0:1) gives the Chebyshev polynomials, and their first derivative, evaluated at  $s = -1$  and  $s = +1$ .
2. Precisely,  $\text{TT}(1,i,d) \equiv T_l^{(d)}(s_i)$  for  $s_0 = -1$  and  $s_1 = +1$ .
3. Note that  $T_l^{(0)}(s) = s^l$  and  $T_l^{(1)}(s) = s^{l+1}l^2$  for  $s = \pm 1$ .
4. Note that

$$T_l(-1) = \begin{cases} +1, & \text{if } l \text{ is even,} \\ -1, & \text{if } l \text{ is odd;} \end{cases} \quad T_l(+1) = \begin{cases} +1, & \text{if } l \text{ is even,} \\ +1, & \text{if } l \text{ is odd;} \end{cases} \quad (3)$$

$$T'_l(-1) = \begin{cases} -l^2, & \text{if } l \text{ is even,} \\ +l^2, & \text{if } l \text{ is odd;} \end{cases} \quad T'_l(+1) = \begin{cases} +l^2, & \text{if } l \text{ is even,} \\ +l^2, & \text{if } l \text{ is odd.} \end{cases} \quad (4)$$

5. TT(0:Mrad,0:1,0:1) is used in routines that explicitly require interface information, such as

- (a) the interface force-balance routine, **lforce**;
- (b) the virtual casing routine, **casing**;
- (c) computing the rotational-transform on the interfaces, **tr00ab**;
- (d) computing the covariant components of the interface magnetic field, **sc00aa**;
- (e) enforcing the constraints on the Beltrami fields, **matrix**; and
- (f) computing the enclosed currents of the vacuum field, **curent**.

### 1.1.6 ImagneticOK(1:Mvol) : Beltrami/vacuum error flag;

1. error flags that indicate if the magnetic field in each volume has been successfully constructed;
2. ImagneticOK is initialized to **.false.** in **dforce** before the Beltrami solver routines are called. If the construction of the Beltrami field is successful (in either **ma02aa** or **mp00ac**) then ImagneticOK is set to **.true.**

### 1.1.7 Lhessianallocated

1. The internal logical variable, **Lhessianallocated**, indicates whether the “Hessian” matrix of second-partial derivatives (really, the first derivatives of the force-vector) has been allocated, or not!

### 1.1.8 ki(1:mn,0:1) : Fourier identification;

1. Consider the “abbreviated” representation for a double Fourier series,

$$\sum_i f_i \cos(m_i\theta - n_i\zeta) \equiv \sum_{n=0}^{N_0} f_{0,n} \cos(-n\zeta) + \sum_{m=1}^{M_0} \sum_{n=-N_0}^{N_0} f_{m,n} \cos(m\theta - n\zeta), \quad (5)$$

and the same representation but with enhanced resolution,

$$\sum_k \bar{f}_k \cos(\bar{m}_k\theta - \bar{n}_k\zeta) \equiv \sum_{n=0}^{N_1} f_{0,n} \cos(-n\zeta) + \sum_{m=1}^{M_1} \sum_{n=-N_1}^{N_1} f_{m,n} \cos(m\theta - n\zeta), \quad (6)$$

with  $M_1 \geq M_0$  and  $N_1 \geq N_0$ ;

then  $k_i \equiv \text{ki}(i,0)$  is defined such that  $\bar{m}_{k_i} = m_i$  and  $\bar{n}_{k_i} = n_i$ .

### 1.1.9 kija(1:mn,1:mn,0:1), kajs(1:mn,1:mn,0:1) : Fourier identification;

1. Consider the following quantities, which are computed in [ma00aa](#), where  $\bar{g}^{\mu\nu} = \sum_k \bar{g}_k^{\mu\nu} \cos \alpha_k$  for  $\alpha_k \equiv m_k \theta - n_k \zeta$ ,

$$\oint \oint d\theta d\zeta \bar{g}^{\mu\nu} \cos \alpha_i \cos \alpha_j = \frac{1}{2} \oint \oint d\theta d\zeta \bar{g}^{\mu\nu} (+\cos \alpha_{k_{ij+}} + \cos \alpha_{k_{ij-}}), \quad (7)$$

$$\oint \oint d\theta d\zeta \bar{g}^{\mu\nu} \cos \alpha_i \sin \alpha_j = \frac{1}{2} \oint \oint d\theta d\zeta \bar{g}^{\mu\nu} (+\sin \alpha_{k_{ij+}} - \sin \alpha_{k_{ij-}}), \quad (8)$$

$$\oint \oint d\theta d\zeta \bar{g}^{\mu\nu} \sin \alpha_i \cos \alpha_j = \frac{1}{2} \oint \oint d\theta d\zeta \bar{g}^{\mu\nu} (+\sin \alpha_{k_{ij+}} + \sin \alpha_{k_{ij-}}), \quad (9)$$

$$\oint \oint d\theta d\zeta \bar{g}^{\mu\nu} \sin \alpha_i \sin \alpha_j = \frac{1}{2} \oint \oint d\theta d\zeta \bar{g}^{\mu\nu} (-\cos \alpha_{k_{ij+}} + \cos \alpha_{k_{ij-}}), \quad (10)$$

where  $(m_{k_{ij+}}, n_{k_{ij+}}) = (m_i + m_j, n_i + n_j)$  and  $(m_{k_{ij-}}, n_{k_{ij-}}) = (m_i - m_j, n_i - n_j)$ ;  
then  $\text{kija}(i, j, 0) \equiv k_{ij+}$  and  $\text{kajs}(i, j, 0) \equiv k_{ij-}$ .

2. Note that Eqn.(6) does not include  $m < 0$ ; so, if  $m_i - m_j < 0$  then  $k_{ij-}$  is re-defined such that  $(m_{k_{ij-}}, n_{k_{ij-}}) = (m_j - m_i, n_j - n_i)$ ; and similarly for the case  $m = 0$  and  $n < 0$ . Also, take care that the sign of the sine harmonics in the above expressions will change for these cases.

### 1.1.10 djkp

### 1.1.11 iotakki

### 1.1.12 cheby(0:Lrad,0:2) : Chebyshev polynomial workspace;

1. `cheby(0:Lrad,0:2)` is global workspace for computing the Chebyshev polynomials, and their derivatives, using the recurrence relations  $T_0(s) = 1$ ,  $T_1(s) = s$  and  $T_l(s) = 2sT_{l-1}(s) - T_{l-2}(s)$ .
2. These are computed as required, i.e. for arbitrary  $s$ , in [bfield](#), [jo00aa](#) and [ma00aa](#).
3. (Note that the quantities required for [ma00aa](#) are for fixed  $s$ , and so these quantities should be precomputed.)

### 1.1.13 Iquad, gaussianweight, gaussianabscissae : Gaussian quadrature;

1. The volume integrals are computed using a ‘‘Fourier’’ integration over the angles and by Gaussian quadrature over the radial, i.e.  $\int f(s)ds = \sum_k \omega_k f(s_k)$ .
2. The quadrature resolution in each volume is give by `Iquad(1:Mvol)` which is determined as follows:
  - if `Nquad.gt.0`, then `Iquad(vvol) = Nquad`;
  - if `Nquad.le.0` and `.not.Lcoordinatesingularity`, then `Iquad(vvol) = 2*Lrad(vvol)-Nquad`;
  - if `Nquad.le.0` and `Lcoordinatesingularity`, then `Iquad(vvol) = 2*Lrad(vvol)-Nquad+Mpol`;
3. The Gaussian weights and abscissae are given by `gaussianweight(1:maxIquad,1:Mvol)` and `gaussianabscissae(1:maxIquad,1:Mvol)` which are computed using modified Numerical Recipes routine `gauleg`.
4. `Iquad_v` is passed through to [ma00aa](#) to compute the volume integrals of the metric elements; also see [jo00aa](#), where `Iquad_v` is used to compute the volume integrals of  $||\nabla \times \mathbf{B} - \mu \mathbf{B}||$ ;

### 1.1.14 LBsequad, LBnewton and LBlinear

1. `LBsequad`, `LBnewton` and `LBlinear` depend simply on `LBeltrami`, which is described in [global](#).

### 1.1.15 BBweight(1:mn) : weighting of force-imbalance harmonics

1. weight on force-imbalance harmonics;

$$\text{BBweight}_i \equiv \text{epsilon} \times \exp[-\text{escale} \times (m_i^2 + n_i^2)] \quad (11)$$

2. this is only used in [dforce](#) in constructing the force-imbalance vector;

### 1.1.16 mmp(1:mn) : spectral condensation weight factors

1. spectral condensation weight factors;

$$\text{mmp}(i) \equiv m_i^p, \quad (12)$$

where  $p \equiv \text{pcondense}$ .

### 1.1.17 NAdof, Ate, Aze, Ato and Azo : degrees-of-freedom in magnetic vector potential

1. NAdof(1:Mvol)  $\equiv$  total number of degrees-of-freedom in magnetic vector potential, including Lagrange multipliers, in each volume. This can be deduced from [matrix](#).

2. The components of the vector potential,  $\mathbf{A} = A_\theta \nabla + A_\zeta \nabla \zeta$ , are

$$A_\theta(s, \theta, \zeta) = \sum_{i,l} A_{\theta,e,i,l} \bar{T}_{l,i}(s) \cos \alpha_i + \sum_{i,l} A_{\theta,o,i,l} \bar{T}_{l,i}(s) \sin \alpha_i, \quad (13)$$

$$A_\zeta(s, \theta, \zeta) = \sum_{i,l} A_{\zeta,e,i,l} \bar{T}_{l,i}(s) \cos \alpha_i + \sum_{i,l} A_{\zeta,o,i,l} \bar{T}_{l,i}(s) \sin \alpha_i, \quad (14)$$

where  $\bar{T}_{l,i}(s) \equiv \bar{s}^{m_i/2} T_l(s)$ ,  $T_l(s)$  is the Chebyshev polynomial, and  $\alpha_j \equiv m_j \theta - n_j \zeta$ . The regularity factor,  $\bar{s}^{m_i/2}$ , where  $\bar{s} \equiv (1+s)/2$ , is only included if there is a coordinate singularity in the domain (i.e. only in the innermost volume, and only in cylindrical and toroidal geometry.)

3. The Chebyshev-Fourier harmonics of the covariant components of the magnetic vector potential are kept in

$A_{\theta,e,v,j,l} \equiv \text{Ate}(\mathbf{v}, 0, \mathbf{j})\%s(1)$ ,

$A_{\zeta,e,v,j,l} \equiv \text{Aze}(\mathbf{v}, 0, \mathbf{j})\%s(1)$ ,

$A_{\theta,o,v,j,l} \equiv \text{Ato}(\mathbf{v}, 0, \mathbf{j})\%s(1)$ , and

$A_{\zeta,o,v,j,l} \equiv \text{Azo}(\mathbf{v}, 0, \mathbf{j})\%s(1)$ ;

where  $v = 1, \text{Mvol}$  labels volume,  $j = 1, \text{mn}$  labels Fourier harmonic, and  $l = 0, \text{Lrad}(v)$  labels Chebyshev polynomial. (These arrays also contains derivative information.)

4. If **Linitguess=1**, a guess for the initial state for the Beltrami fields is constructed. An initial state is required for iterative solvers of the Beltrami fields, see **LBeltrami**.

5. If **Linitguess=2**, the initial state for the Beltrami fields is read from file (see **ra00aa**). An initial state is required for iterative solvers of the Beltrami fields, see **LBeltrami**.

### 1.1.18 workspace

#### 1.1.19 goomne, goomno : metric information

#### 1.1.20 gssmne, gssmno : metric information

#### 1.1.21 gstmne, gstmno : metric information

#### 1.1.22 gszmne, gszmno : metric information

#### 1.1.23 gttmne, gttmno : metric information

#### 1.1.24 gtzmne, gtzmno : metric information

#### 1.1.25 gzzmne, gzzmno : metric information

1. The metric information are:

goomne(0:mne), goomno(0:mne)

gssmne(0:mne), gssmno(0:mne)

gstmne(0:mne), gstmno(0:mne)

gszmne(0:mne), gszmno(0:mne)

gttmne(0:mne), gttmno(0:mne)

gtzmne(0:mne), gtzmno(0:mne)

gzzmne(0:mne), gzzmno(0:mne)

2. These are defined in [metrix](#), and used in [ma00aa](#).

### 1.1.26 cosi(1:Ntz,1:mn) and sini(1:Ntz,1:mn)

1. Trigonometric factors used in various Fast Fourier transforms, where

$$\text{cosi}_{j,i} = \cos(m_i \theta_j - n_i \zeta_j), \quad (15)$$

$$\text{sini}_{j,i} = \sin(m_i \theta_j - n_i \zeta_j). \quad (16)$$

### 1.1.27 psifactor(1:mn,1:Mvol) : coordinate “pre-conditioning” factor

1. In toroidal geometry, the coordinate “pre-conditioning” factor is

$$f_{j,v} \equiv \begin{cases} \psi_{t,v}^0 & , \text{ for } m_j = 0, \\ \psi_{t,v}^{m_j/2} & , \text{ otherwise.} \end{cases} \quad (17)$$

where  $\psi_{t,v} \equiv \text{tflux}$  is the (normalized?) toroidal flux enclosed by the  $v$ -th interface.

2. psifactor is used in [packxi](#), [dforce](#) and [hesian](#).

### 1.1.28 Bsupumn and Bsupvmn

### 1.1.29 diotadxup and glambda : transformation to straight fieldline angle

1. Given the Beltrami fields in any volume, the rotational-transform on the adjacent interfaces may be determined (in [tr00ab](#)) by constructing the straight fieldline angle on the interfaces.
2. The rotational transform on the inner or outer interface of a given volume depends on the magnetic field in that volume, i.e.  $t_{\pm} = t(\mathbf{B}_{\pm})$ , so that

$$\delta t_{\pm} = \frac{\partial t_{\pm}}{\partial \mathbf{B}_{\pm}} \cdot \delta \mathbf{B}_{\pm}. \quad (18)$$

3. The magnetic field depends on the Fourier harmonics of both the inner and outer interface geometry (represented here as  $x_j$ ), the helicity multiplier, and the enclosed poloidal flux, i.e.  $\mathbf{B}_{\pm} = \mathbf{B}_{\pm}(x_j, \mu, \Delta\psi_p)$ , so that

$$\delta \mathbf{B}_{\pm} = \frac{\partial \mathbf{B}_{\pm}}{\partial x_j} \delta x_j + \frac{\partial \mathbf{B}_{\pm}}{\partial \mu} \delta \mu + \frac{\partial \mathbf{B}_{\pm}}{\partial \Delta\psi_p} \delta \Delta\psi_p. \quad (19)$$

4. The rotational-transforms, thus, can be considered to be functions of the geometry, the helicity-multiplier and the enclosed poloidal flux,  $t_{\pm} = t_{\pm}(x_j, \mu, \Delta\psi_p)$ .
5. The rotational-transform, and its derivatives, on the inner and outer interfaces of each volume is stored in `diotadxup(0:1,-1:2,1:Mvol)`. The arguments label:

- i. the first argument labels the inner or outer interface,
- ii. the the second labels derivative, with

-1 : indicating the derivative with respect to the interface geometry, i.e.  $\frac{\partial t_{\pm}}{\partial x_j}$ ,

0 : the rotational-transform itself,

1,2 : the derivatives with respect to  $\mu$  and  $\Delta\psi_p$ , i.e.  $\frac{\partial t_{\pm}}{\partial \mu}$  and  $\frac{\partial t_{\pm}}{\partial \Delta\psi_p}$ ;

- iii. the third argument labels volume.

6. The values of `diotadxup` are assigned in [mp00aa](#) after calling [tr00ab](#).

### 1.1.30 vvolum, lBBintegral and lABintegral

1. volume integrals

$$\text{vvolum}(i) = \int_{V_i} dv \quad (20)$$

$$\text{lBBintegral}(i) = \int_{V_i} \mathbf{B} \cdot \mathbf{B} dv \quad (21)$$

$$\text{lABintegral}(i) = \int_{V_i} \mathbf{A} \cdot \mathbf{B} dv \quad (22)$$