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 bnorml **1**

1.1 free-boundary constraint 1

1.2 compute the normal field on a regular grid on the computational boundary 1

1.1 free-boundary constraint

1. The normal field at the computational boundary, $\partial\mathcal{D}$, should be equal to $(\mathbf{B}_P + \mathbf{B}_C) \cdot \mathbf{e}_\theta \times \mathbf{e}_\zeta$, 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.

2. The plasma field, $\mathbf{B}_P$, can only be computed after the equilibrium is determined, but this information is required to compute the equilibrium to begin with; and so there is an iteration involved.

3. Suggested values of the vacuum field can be self generated; see xspech for more documentation on this.

1.2 compute the normal field on a regular grid on the computational boundary

1. For each point on the compuational boundary, casing is called to compute the normal field produced by the plasma currents.

2. (There is a very clumsy attempt to parallelize this which could be greatly improved.)

3. An FFT gives the required Fourier harmonics.

bnorml.h last modified on 18-07-26 10:31:20.5; SPEC subroutines;