
$B(\ell)$


$$B(\theta)$$


$C(\ell)$

• • •

$3.46\ 10^{18}$

$\{\ell, m\}$

$$i = \ell^2 + \ell + m + 1$$

$0 \leq \ell \leq \ell_{\max}$

$$0 \leq m \leq m_{\max}$$

$(\ell, m) \leq 100$

$m > \ell$

ℓ, m


(ℓ, m)

2π

$$x(0), \ldots, x(n-1),\ y(0), \ldots, y(n-1),\ z(0), \ldots, z(n-1)$$

$$x = \sin\theta\cos\phi$$

$$y = \sin\theta \sin\phi$$

$z = \cos\theta$

$\cos^{-1}(\mathbf{V}.\mathbf{W})$

$$2\sin^{-1}\left(||\mathbf{V}-\mathbf{W}||\right)/2$$

W

$$d_i = \mathrm{dist}(\mathbf{V}, \mathbf{W}_i)$$

$$d_i = \text{dist}(\mathbf{V}_i, \mathbf{W})$$

$$d_i = \mathrm{dist}(\mathbf{V}_i, \mathbf{W}_i)$$

$(x, y, z) = (1, 1, 1)/\sqrt{3}$


$b(\theta)$

θ

$$\int d\mathbf{r}\ b(\mathbf{r})\ Y^*_{\ell m}(\mathbf{r})$$

$$b_{\ell 0}\sqrt{\frac{4\pi}{2\ell+1}}$$

$$\int b(\theta) P_\ell(\theta) \sin(\theta)\ d\theta\ 2\pi$$

$\{0, \ldots, 4000\}$

$$C(b)\sqrt{2/((2l_b+1)\Delta l_b)}$$

$l(l+1)/2\pi$

$\propto 2l + 1$

$$b(\mathbf{r}) = \sum_{\ell m} b_{\ell m} Y_{\ell m}(\mathbf{r}),$$

$$b(\theta) = \sum_{\ell} b(\ell) P_{\ell}(\theta) \frac{2\ell + 1}{4\pi},$$

$$b(\ell) = b_{\ell 0}\sqrt{\frac{4\pi}{2\ell+1}}$$

$1 \leq n \leq 3$

euler_matrix_new allows the generation of a rotation Euler matrix. The user can choose the three Euler angles, and the three axes of rotation.
If `vec` is an N×3 array containing N 3D vectors,
`vecr = vec # euler_matrix_new(a1,a2,a3,/Y)`
will be the rotated vectors. Alternatively, rotate_coord can also be used to rotate `vec` into `vecr`.

This routine supersedes euler_matrix, which had inconsistent angle definitions. The relation between the two routines is as follows :

euler_matrix_new(a,b,c,/X) = euler_matrix(−a,−b,−c,/X)
= Transpose(euler_matrix(c, b, a,/X))

euler_matrix_new(a,b,c,/Y) = euler_matrix(−a, b,−c,/Y)
= Transpose(euler_matrix(c,−b, a,/Y))

euler_matrix_new(a,b,c,/Z) = euler_matrix(−a, b,−c,/Z)

$$i = \ell^2 + \ell + m + 1.$$

$\ell^2 + \ell + m + 1$

$C(\ell) = \sum_m a_{\ell m} a^*_{\ell m} / (2\ell + 1)$

$\ell(\ell+1)/2\pi$

$\ell(\ell+1)C(\ell)/2\pi$

$$C(\ell)_{\rm meas} = C(\ell)B(\ell)^2$$

$\leq 12N_{\text{side}}^2$

should produce something like

196608 128 256 2

meaning that the map contained in that file has 196608 pixels, the resolution parameter is nside=128, the maximum multipole was 256, and this a full sky map (type 2).

μ

$\sqrt{3/\pi}\ 3600/N_{\rm side}$

$$C(\ell)_{\rm pix} = C(\ell) W(\ell)^2$$

will print out

```
<Expression>        INT        =        1

A+1        INT        =        1
```

$$a_{\ell m}^{\mathrm{OUT}} = a_{\ell m}^{\mathrm{IN}} \frac{B^{\mathrm{OUT}}(\ell) P^{\mathrm{OUT}}(\ell)}{B^{\mathrm{IN}}(\ell) P^{\mathrm{IN}}(\ell)},$$


$P(\ell)$

$$P(\ell) = 1$$

$$B(\ell) = 1$$

$= \ell^2 + \ell + m + 1$

$$y = \sinh^{-1}(x)$$

$x \ll 1$

$$y \approx \ln(2x)$$

$x \gg 1$

$$y = \sinh^{-1}(x/2)/\ln(10)$$

$y \approx 0.21x$

$= -1.6375\,10^{30}$

$9 \leq |\mathrm{psym}| \leq 46$

$\{2, \ldots, 20\}$

$$P = \sqrt{U^2 + Q^2}$$

$$\phi = \tan^{-1}(U/Q)/2$$

$\{1, \ldots, 2^{29}\}$

$\leq 2^{29}$

$$N_w \simeq N_{\rm pix}/16$$

$$\mathrm{N}_w = \frac{(N_{\mathrm{side}}+1)(3N_{\mathrm{side}}+1)}{4}.$$

$$\mathrm{N_{template}} = \frac{1+N_{\mathrm{side}}(N_{\mathrm{side}}+6)}{4}.$$

$x_N(0), \ldots, x_N(n-1),\ y_N(0), \ldots, y_N(n-1),\ z_N(0), \ldots, z_N(n-1)$

$x_W(0), \ldots, x_W(n-1),\ y_W(0), \ldots, y_W(n-1),\ z_W(0), \ldots, z_W(n-1)$

$3\pi/4$

$$\propto 1/\sqrt{\mathrm{n\_obs}}$$

$$\equiv -1.6375\ 10^{30}$$

$$Q' = Q\cos(s\Delta\psi) - U\sin(s\Delta\psi),$$

$$U' = U\cos(s\Delta\psi) + Q\sin(s\Delta\psi),$$

$$z = \cos(\theta) \geq 2/3,$$

$$0 < \phi \leq \pi/2,$$

$$2/3 > z \geq 0,$$

$$\phi = 0, \quad \text{or} \quad \phi = \frac{\pi}{4N_{\text{side}}}.$$

$$2/3 > z \geq 0,$$

$$\phi = 0, \quad \text{or} \quad \phi = \frac{\pi}{4N_{\text{side}}}.$$

if set, during **degradation** each big pixel containing one bad or missing small pixel is also considered as bad,
if not set, each big pixel containing at least one good pixel is considered as good (optimistic) default = 0 (:not set)

$$N_w \simeq 0.75 N_{\rm side}^2 \simeq N_{\rm pix}/16$$

≥ 4

write_fits_sb writes out the information contained in `Prim_stc` and `Exten_stc` in the primary unit and extension of the FITS file `File` respectively . Coordinate systems can also be specified by `Coordsys`. Specifying the ordering scheme is compulsary for **HEALPix** data sets and can be done either in `Header` or by setting `Ordering` or `Nested` or `Ring` to the correct value. If `Ordering` or `Nested` or `Ring` is set, its value overrides what is given in `Header`.

The data is assumed to represent a full sky data set with the number of data points npix = 12*Nside*Nside unless
Partial is set *or* the input FITS header contains OBJECT = 'PARTIAL'
AND
the Nside qualifier is given a valid value *or* the FITS header contains a NSIDE.

In the **HEALPix** scheme, invalid or missing pixels should be given the value `!healpix.bad_value`$= -1.63750\,10^{30}$.

If `Nothealpix` is set, the restrictions on Nside are void.