Evaluating the Alignment of Astronomical Linear Polarization Data, Intermediate Level Software

Richard Shurtleff *

Abstract

This article is a Mathematica notebook that is meant to serve as a template. User-supplied astronomical observations of transverse vectors on the sky can be evaluated, their alignment judged by the Hub test. The test can be applied to any set of transverse vectors on a spherical surface, but the language here applies to linear polarization directions of electromagnetic radiation from astronomical sources. This article presents a simulation, analyzing artificial data as an illustration of the process. The analysis produces a numerical value quantifying the alignment of the polarization directions and its significance. A visual representation of the alignment is developed, mapping regions of convergence and divergence on the Celestial sphere. This intermediate-level article builds on a previous, basic notebook by carrying uncertainties in the data through the calculations.

Keywords: Polarization ; Alignment ; Computer Program ; Uncertainties

*Department of Sciences, Wentworth Institute of Technology, 550 Huntington Avenue, Boston, MA, USA, 02115, orcid.org/0000-0001-5920-759X, e-mail addresses: shurtleffr@wit.edu, momentummatrix@yahoo.com

UPDATES:

Errata and other changes to the online pdf version may appear here.

```
In[406]:= Print["The date and time that this statement was evaluated: ", Now]
```

```
The date and time that this statement was evaluated: Sun 24 Jan 2021 15:38:55 GMT−5.
```

0. Preface

This notebook is intended to be used as a template. In order to use the notebook, it must be somehow translated into the Mathematica computer language. You can simply copy the text here keystroke-by-keystroke into an active Mathematica notebook. A link[0] to the Mathematica notebook is provided in the references, Ref. 0.

One needs the location of the sources on the sky, a position angle and the uncertainty of its value at each source. Replace the simulated data in *Sec. 3* and run the notebook.

Transverse vectors on the sky can be observed for many situations, linear polarization, major/minor axes, jets and others. These observed asymmetries may be analyzed for their mutual alignment, individually or one with another.

This work is based on an article[1] “Indirect polarization alignment with points on the sky, the Hub Test”. A basic notebook[2] exists that does not deal with experimental uncertainty in polarization directions. Much of the early parts of the present notebook repeat the more basic notebook.

This notebook and the earlier notebooks were created using Wolfram Mathematica[3], Version Number: 12.1 which is running on Microsoft Windows(64-bit).

CONTENTS

1. Introduction

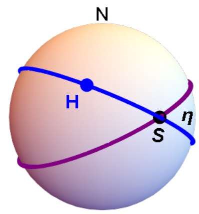
Given a collection of astronomical sources with linearly polarized electromagnetic emissions, one may ask whether the polarization directions align.

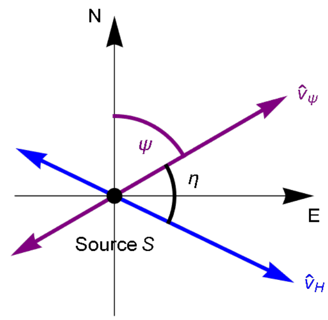
The Hub test answers the question of alignment indirectly. Instead of attempting to find direct correlations of the polarization directions of a number of sources, an alternative process is applied.

The basic idea is illustrated in the figures below. The Celestial sphere is pictured on the left and on the right is the plane tangent to the sphere at the source *S*. The linear polarization direction $\hat{v}_{\psi}$ lies in the tangent plane and determines the purple great circle on the sphere. A point *H* on the sphere and the location *S* of the source determine a second great circle, the blue circle drawn on the sphere at the left. Clearly, *H* and *S* must be distinct points on the sphere. The angle η, with $0° \leq \eta \leq 90°$, measures the "alignment of the polarization direction with the point *H*." Perfect alignment occurs when $\eta = 0°$ and the two great circles form a single circle.

The basic concept includes "avoidance", as well as alignment. Avoidance is high when the two directions $\hat{v}_{\psi}$ and $\hat{v}_{H}$ differ by a large angle, $\eta \to 90°$. Perpendicular great circles at S , $\eta = 90°$, would indicate the maximum avoidance of the polarization direction and the point on the sphere.

Out[ ]=





With many sources S_i, $i = 1, ..., N$, there are N alignment angles η_{iH} for the point H. To quantify the alignment of the N sources with the point H, calculate the arithmetic average alignment angle at H,

$$\overline{\eta}(\mathrm{H}) = \frac{1}{N} \sum_{i=1}^{N} \eta_{\mathrm{iH}} \, . \qquad (1)$$

The alignment angle $\overline{\eta}(\mathrm{H})$ is a function of position H on the sphere. The polarization directions are best aligned with the point $H_{\min}$ where the alignment angle is a minimum $\overline{\eta}_{\min}$. The polarization directions most avoid the point $H_{\max}$where the function $\overline{\eta}(\mathrm{H})$ takes its maximum value $\overline{\eta}_{\max}$. For a visual aid, see the map generated in Sec. 7.

The Hub test is based on the idea that the polarization directions are well-aligned with each other when they are well-aligned with some point $H_{\min}$. Another point, $H_{\max}$, is distinguished by the collection of polarization directions; $H_{\max}$ is the most avoided point. Both $H_{\min}$ and $H_{\max}$ as well as the points $-H_{\min}$ and $-H_{\max}$ diametrically opposite are called "hubs".

The Hub test calculates $\overline{\eta}_{\min}$ and $\overline{\eta}_{\max}$ for a given collection of polarized sources. The smaller the value of $\overline{\eta}_{\min}$, the better aligned the sources are. The larger the value of $\overline{\eta}_{\max}$, the more significant their avoidance of $H_{\max}$.

For more on the Hub test, see the article[1].

Experimental observations return measured values. The values should be accompanied by estimates of their uncertainties. Uncertainties in the measured data produce uncertainties in calculated results. This notebook shows one way to carry uncertainties in the polarization directions through the calculations.

As in a previous more basic notebook[2], the data presented and analysed here are simulated, not measured.

2. Preliminary

We work on a sphere in 3 dimensional Euclidean space. See the figures in the Introduction. The sphere is called the "Celestial sphere" or simply the "sphere". The center of the sphere is the origin of a 3D Cartesian coordinate system with coordinates (x, y,z). The direction of the positive z-axis is associated with "North". Right ascension, RA or α, and declination, dec or δ, are measured as usual with the direction of the positive x-axis along (RA,dec) = (0°, 0°). The declination δ = 90° indicates North pole, the direction from the origin (0,0,0) to (0,0,1) .

From a point-of-view located outside the sphere, as in the left-hand figure in the Introduction, one pictures a source S plotted on the sphere and, in the 2D tangent plane at S, local North is upward and local East is to the right. See the right-hand figure in the Introduction. A "position angle" at the point S on the sphere is measured in the 2D plane tangent to the sphere at S. The position angle ψ is measured clockwise from local North with East to the right.

Definitions:

(α,δ)	Right Ascension RA and declination dec of a point on the sphere. Sometimes we use radians, sometimes degrees.
$er(\alpha,\delta)$	radial unit vector in a Cartesian coordinate system from the Origin to the point on the sphere with (RA,dec) = (α,δ), with α,δ in radians
$eN(\alpha,\delta)$	unit vector along local North at the point (α,δ) on the sphere, with α,δ in radians
$eE(\alpha,\delta)$	unit vector along local East at the point (α,δ) on the sphere, with α,δ in radians
$\alpha FROMr(\hat{r})$	RA for the point on the sphere determined by radial unit vector $\hat{r}$, result in radians
$\delta FROMr(\hat{r})$	dec for the point on the sphere determined by radial unit vector $\hat{r}$, result in radians

```
In[2]:= (* For a Source at (RA,dec) = (α,δ): er, eN,
eE are unit vectors from Origin to Source, local North, local East, resp.  *)
er[α_, δ_] := er[α, δ] = {Cos[α] Cos[δ], Sin[α] Cos[δ], Sin[δ]}
eN[α_, δ_] := eN[α, δ] = {-Cos[α] Sin[δ], -Sin[α] Sin[δ], Cos[δ]}
eE[α_, δ_] := eE[α, δ] = {-Sin[α], Cos[α], 0}
Print["Check er.er = 1, er.eN = 0, er.eE = 0,
   eN.eN = 1, eN.eE = 0,eE.eE = 1, erXeE = eN, eEXeN = er, eNXer = eE: ",
 {0} == Union[Flatten[Simplify[{er[α, δ].er[α, δ] - 1, er[α, δ].eN[α, δ], er[α, δ].eE[α, δ],
      eN[α, δ].eN[α, δ] - 1, eN[α, δ].eE[α, δ], eE[α, δ].eE[α, δ] - 1, Cross[er[α, δ], eE[α, δ]] -
       eN[α, δ], Cross[eE[α, δ], eN[α, δ]] - er[α, δ], Cross[eN[α, δ], er[α, δ]] - eE[α, δ]}]]]]

Check er.er = 1, er.eN = 0, er.eE = 0, eN.eN
   = 1, eN.eE = 0,eE.eE = 1, erXeE = eN, eEXeN = er, eNXer = eE: True
```

Get (α,δ) in radians from radial vector r, with $-\pi < \alpha < +\pi$ and $\frac{-\pi}{2} < \delta < \frac{+\pi}{2}$

```
In[6]:= αFROMr[r_] := N[ArcTan[Abs[r[[2]]/r[[1]]]]] /; (r[[2]] ≥ 0 && r[[1]] > 0)
αFROMr[r_] := N[π - ArcTan[Abs[r[[2]]/r[[1]]]]] /; (r[[2]] ≥ 0 && r[[1]] < 0)
αFROMr[r_] := N[-π + ArcTan[Abs[r[[2]]/r[[1]]]]] /; (r[[2]] < 0 && r[[1]] < 0)
αFROMr[r_] := N[-ArcTan[Abs[r[[2]]/r[[1]]]]] /; (r[[2]] < 0 && r[[1]] > 0)
αFROMr[r_] := π/2. /; (r[[2]] ≥ 0 && r[[1]] == 0)
αFROMr[r_] := -π/2. /; (r[[2]] < 0 && r[[1]] == 0)

In[12]:= δFROMr[r_] := N[ArcTan[r[[3]]/Sqrt[r[[1]]^2 + r[[2]]^2]]] /; (Sqrt[r[[1]]^2 + r[[2]]^2] > 0)
δFROMr[r_] := Sign[r[[3]]] π/2. /; (Sqrt[r[[1]]^2 + r[[2]]^2] == 0)
```

3. Input and Settings

This section is where you would enter your data for analysis. You can input source locations in various ways using the functions in Section 2 above.

Be careful of units. The angles α, δ, ψ are all expected to be in radians.

Definitions:

gridSpacing separation in degrees between grid points on a constant latitude circle and separation of constant latitude circles. There is no bunching at the poles.

ρRegion estimated radius of the region containing the sources, choose from ρRegion = {90° (whole sphere), 48°, 24°, 12°,5°, 0° (point-like)}.

nSrc	number of sources in the region
αSrc	Right Ascension (RA) at the sources, in radians
δSrc	declinations (dec) at the sources, in radians
rSrc	radial unit vectors in Cartesian coordinates from origin to sources S_i
ψn	the polarization position angles for the EM radiation from the sources, in radians
$\sigma\psi$n	uncertainties in the polarization position angles ψn, the half-widths of normal distributions of likelihood of observation
dηContourPlot	separation of successive contour lines on the map in Sec. 7, in degrees
dataDirectory	folder on the computer where the map and data files are to be saved.
nR	number of runs with each run having a different set of polarization directions allowed by uncertainty
ρSrcToCenter	angle between the radial vector to a source and the radial vector to the center of the source region, i.e. angle between rSrc and rCenter

Settings

In[14]:=
```
gridSpacing = 2.(*, in degrees. This is a setting.*);
Print["The grid points are separated by ",
 gridSpacing, "° arcs along latitude and longitude."]
```

```
The grid points are separated by 2.° arcs along latitude and longitude.
```

In[16]:=
```
regionRadiusChoices = {90, 48, 24, 12, 5, 0};(*Do not change this statement*)
regionChoice = 3; (*This is a setting. The choice 24° is 3rd in the list. *)
rgnRadius = regionRadiusChoices[[regionChoice]];
Print["The region radius controls the constants ci and ai for statistics in Sec. 4."]
Print["The region radius ρ is set at ", rgnRadius, "°."]
```

```
The region radius controls the constants ci and ai for statistics in Sec. 4.
```

```
The region radius ρ is set at 24°.
```

In[21]:=
```
dηContourPlot = 4 ;(*, in degrees. This is a setting.*)
```

In[22]:=
```
dataDirectory =
  "C:\\Users\\shurt\\Dropbox\\HOME_DESKTOP-0MRE5OJ\\SendXXX_CJP_CEJPetc\\SendViXra\\
    20200715AlignmentMethod\\20200715AlignmentMMAnotebooks\\StarterKit\\20210110
    MapAndUncertainty"; (*This is a setting.*)
```

In[23]:=
```
nR = 2000;(*number of runs with various ψ allowed by uncertainty. This is a setting. *)
```

Inputs

In[24]:=
```
(*The locations of the sources Sᵢ.  Here (RA,dec) are the inputs and Cartesian
 coordinates are calculated. Alternatively, you can input rSrc and calculate αSrc,
δSrc with the functions αFROMr and δFROMr in Sec. 1.*)
αSrc = {1.0245, 0.2994, 0.8584, 0.4293, 0.7828, 0.7407, 1.1216, 0.5534,
   0.7863, 1.0897, 0.9064, 0.7216, 0.3302, 0.3788, 1.1390, 0.5709};(*Input*)
nSrc = Length[αSrc]; (*The number of sources, calculated from Input.*)
δSrc = {0.8400, 0.6266, 0.2472, 0.2780, 0.3821, 0.3826, 0.5953, 0.9090,
  0.6663, 0.6634, 0.4188, 0.6961, 0.5614, 0.7652, 0.8050, 0.2800}; (*Input*)
rSrc = Table[er[ αSrc[[i]], δSrc[[i]] ],  {i, nSrc}];
(*calculated from Input.*)
```

In[26]:=
```
(*The polarization position angles in radians for the EM radiation from the sources.*)
ψn = {2.2816, 1.3406, 2.6725, 1.9480, 1.7352, 2.2421, 0.1986, 2.1445,
   2.3088, 2.0109, 1.6127, 0.3118, 1.6390, 2.3304, 2.4428, 1.8222};(*Input*)
(*The uncertainties in the polarization position angles
 in radians. This is an input. *)
σψn = { 0.1406, 0.1449, 0.1876, 0.1967, 0.2072, 0.2297, 0.1821, 0.2201,
   0.2235, 0.2143, 0.1512, 0.1532, 0.2182, 0.2323, 0.2424, 0.2131}; (*Input*)
```

4. Significance

When 5% or fewer results with random data are better then a result with observed data, the observed result is called "significant" by definition or by convention.
When 1% or fewer random results are better, then a result is called "very significant" by definition or by convention.

To determine the probability distributions and related formulas, we made many runs with random data and fit the results. There were 2000 runs for each combination of N sources in regions of radii ρ, with $N = \{8,16,32,64,128,181,256,512\}$ and with radii $\rho = \{0°,5°,12°,24°,48°,90°\}$. That makes (2000)(8)(6) = 96000 runs. For more details see Ref. 1.

Definitions:

probMIN0, probMAX0	probability distributions for alignment (MIN) and avoidance (MAX), functions of η, η_0, σ
probMIN, probMAX	same as above except these are functions of η and N, using η_0(N,c1,a1) and σ(N,c2,a2) to get η_0 and σ
signiMIN0, signiMAX0	significance as a function of (η, η_0, σ)
signiMIN, signiMAX	significance as a function of (η,N) using η_0(N,c1,a1) and σ(N,c2,a2) to get η_0 and σ
norm	a constant used to normalize the distribution (the integral of probability must be 1)
η	alignment angle
η0	"mean", a parameter with a value near the peak of the probability distribution
σ	"half-width", a parameter with a value near the distribution's half-width
c1MIN, a1MIN,...	parameters needed to find η0 and σ from the number of sources N.
c1MINplusMinus, ...	standard error (plus/minus) in parameters found in fitting random data
η0MIN, η0MAX	functions for finding the mean η0
σMIN, σMAX	functions for half-width σ

In[28]:=
```
(* y = ((η - η0)/σ) *)
(* dy = dη/σ *)
(* The normalization factor "norm" is needed for the probability density *)
```
$$norm = \left(NIntegrate\left[\left(1 + e^{4\,(y-1)}\right)^{-1} e^{-\frac{y^2}{2}}, \{y, -\infty, \infty\}\right]\right)^{-1};$$
$$\sqrt{2\pi}\ norm\ (*\text{Constant needed for Eq. } (10) \text{ and } (11) \text{ in the article}^1.*)$$

Out[29]= 1.22029

In[30]:= probMIN0[η_, η0_, σ_] :=
$$\frac{norm}{\sigma}\left(1 + e^{4\,\frac{(\eta-\eta0-\sigma)}{\sigma}}\right)^{-1} e^{-\frac{1}{2}\left(\frac{\eta-\eta0}{\sigma}\right)^2}\ (*\text{A Gaussian modified by an S-function } \left(1+e^{4\,\frac{(\eta-\eta0-\sigma)}{\sigma}}\right)^{-1}.*)$$

In[31]:= signiMIN0[η_, η0_, σ_] := NIntegrate[probMIN0[η1, η0, σ], {η1, -∞, η}]

Next, check that the normalization constant does not change from the alignment (MIN) case to the avoidance (MAX) case:

In[32]:=
$$normMAX = NIntegrate\left[\left(1 + e^{-4\,(y+1)}\right)^{-1} e^{-\frac{y^2}{2}}, \{y, -\infty, \infty\}\right]^{-1};$$
```
Print["The normalization constant for probMIN and probMAX are equal: ",
 normMAX, " and ", norm]
```

The normalization constant for probMIN and probMAX are equal: 0.486826 and 0.486826

In[34]:= probMAX0[η_, η0_, σ_] :=
$$\frac{norm}{\sigma}\left(1 + e^{-4\,\frac{(\eta-\eta0+\sigma)}{\sigma}}\right)^{-1} e^{-\frac{1}{2}\left(\frac{\eta-\eta0}{\sigma}\right)^2}$$

In[35]:= signiMAX0[η_, η0_, σ_] := NIntegrate[probMAX0[η1, η0, σ], {η1, η, ∞}]

The significance signiMIN0[η, η0, σ] is the integral of probMIN0, i.e. signiMIN0 = $\int_{-\infty}^{\eta} P_{MIN}(\eta)\, d\eta$.

The significance signiMAX0[η, η0, σ] is the integral of probMAX0, i.e. signiMAX0 = $\int_{\eta}^{\infty} P_{MAX}(\eta)\, d\eta$.

The formulas for mean $\eta_0 = \frac{\pi}{4} \pm \frac{c1}{N^{a1}}$ and half-width $\sigma = \frac{c2}{4\,N^{a2}}$ estimate η_0 and σ by functions of the number N of sources.

These formulas depend on the size of the region (radius ρ) by the choice of parameters c_i and a_i, i = 1,2. The following values for the parameters c_i and a_i are based on random runs. For each combination of N = {8,16,32,64,128,181,256,512} and ρ = {0°,5°,12°,24°,48°,90°}, there were 2000 random runs completed.

A notation conflict between this notebook and the article[1] should be noted. We doubled the exponent “a” so $N^{a/2}$ appears in the article, whereas in the random runs and here we see N^a. Thus $a \approx 1/2$ here and in the random run fits, but the paper has $a_{\text{Article}} \approx 1$. That explains the “/2” in the following arrays.

In[36]:=
$$\rho\text{ciaiMIN} = \begin{matrix} \text{"}\rho\text{"} & \text{"c1"} & \text{"a1"} & \text{"c2"} & \text{"a2"} \\ 90 & 0.9423 & 1.0046/2 & 1.061 & 0.954/2 \\ 48 & 0.9505 & 1.0156/2 & 1.166 & 0.9956/2 \\ 24 & 0.9235 & 1.0069/2 & 1.127 & 0.964/2 \\ 12 & 0.8912 & 1.0054/2 & 1.238 & 1.021/2 \\ 5 & 0.8363 & 1.0088/2 & 1.076 & 0.940/2 \\ 0 & 0.5031 & 1.0153/2 & 1.522 & 1.053/2 \end{matrix};$$

In[37]:=
$$\rho\text{ciaiMAX} = \begin{matrix} \text{"}\rho\text{"} & \text{"c1"} & \text{"a1"} & \text{"c2"} & \text{"a2"} \\ 90 & 0.9441 & 1.0055/2 & 1.000 & 0.931/2 \\ 48 & 0.9572 & 1.0165/2 & 1.090 & 0.958/2 \\ 24 & 0.927 & 1.0068/2 & 1.101 & 0.964/2 \\ 12 & 0.9049 & 1.0090/2 & 1.228 & 1.018/2 \\ 5 & 0.8424 & 1.0062/2 & 1.168 & 0.992/2 \\ 0 & 0.4982 & 1.0093/2 & 1.543 & 1.060/2 \end{matrix};$$

In[38]:=
$$\rho\Delta\text{ciaiMIN} = \begin{matrix} \text{"}\rho\text{"} & \text{"c1"} & \text{"a1"} & \text{"c2"} & \text{"a2"} \\ 90 & 0.0050 & 0.0036/2 & 0.026 & 0.016/2 \\ 48 & 0.0079 & 0.0057/2 & 0.016 & 0.0095/2 \\ 24 & 0.0024 & 0.0018/2 & 0.022 & 0.013/2 \\ 12 & 0.0034 & 0.0026/2 & 0.039 & 0.021/2 \\ 5 & 0.0035 & 0.0028/2 & 0.030 & 0.019/2 \\ 0 & 0.0059 & 0.0080/2 & 0.052 & 0.024/2 \end{matrix};$$

In[39]:=
$$\rho\Delta\text{ciaiMAX} = \begin{matrix} \text{"}\rho\text{"} & \text{"c1"} & \text{"a1"} & \text{"c2"} & \text{"a2"} \\ 90 & 0.0061 & 0.0044/2 & 0.038 & 0.025/2 \\ 48 & 0.0063 & 0.0045/2 & 0.026 & 0.016/2 \\ 24 & 0.011 & 0.0079/2 & 0.019 & 0.011/2 \\ 12 & 0.0069 & 0.0052/2 & 0.039 & 0.022/2 \\ 5 & 0.0038 & 0.0031/2 & 0.022 & 0.013/2 \\ 0 & 0.0058 & 0.0080/2 & 0.057 & 0.025/2 \end{matrix}$$

If you have trouble translating the arrays from the pdf version into a viable Mathematica notebook, the following cells are equivalent.
To activate a cell, remove the remark brackets (* and *).

In[40]:=
```
(*ρciaiMIN={{"ρ","c1","a1","c2","a2"},
  {90,0.9423`,0.5023`,1.061`,0.477`},{48,0.9505`,0.5078`,1.166`,0.4978`},
  {24,0.9235`,0.50345`,1.127`,0.482`},{12,0.8912`,0.5027`,1.238`,0.5105`},
  {5,0.8363`,0.5044`,1.076`,0.47`},{0,0.5031`,0.50765`,1.522`,0.5265`}}*)
```

In[41]:=
```
(*ρciaiMAX={{"ρ","c1","a1","c2","a2"},
  {90,0.9441`,0.50275`,1.`,0.4655`},{48,0.9572`,0.50825`,1.09`,0.479`},
  {24,0.927`,0.5034`,1.101`,0.482`},{12,0.9049`,0.5045`,1.228`,0.509`},
  {5,0.8424`,0.5031`,1.168`,0.496`},{0,0.4982`,0.50465`,1.543`,0.53`}}*)
```

In[42]:=
```
(*ρΔciaiMIN={{"ρ","c1","a1","c2","a2"},
  {90,0.005`,0.0018`,0.026`,0.008`},{48,0.0079`,0.00285`,0.016`,0.00475`},
  {24,0.0024`,0.0009`,0.022`,0.0065`},{12,0.0034`,0.0013`,0.039`,0.0105`},
  {5,0.0035`,0.0014`,0.03`,0.0095`},{0,0.0059`,0.004`,0.052`,0.012`}}*)
```

```
In[43]:= (*ρΔciaiMAX={{"ρ","c1","a1","c2","a2"},
  {90,0.0061`,0.0022`,0.038`,0.0125`},{48,0.0063`,0.00225`,0.026`,0.008`},
  {24,0.011`,0.00395`,0.019`,0.0055`},{12,0.0069`,0.0026`,0.039`,0.011`},
  {5,0.0038`,0.00155`,0.022`,0.0065`},{0,0.0058`,0.004`,0.057`,0.0125`}}*)

In[44]:= (*Change the region radius, if necessary, in Section 3 Inputs and Settings. *)
iρ = regionChoice + 1; (* Parameters ci, ai, i = 1,2. *)
Print["These constants are for sources confined to regions with radii ρ = ",
 ρciaiMIN[[iρ, 1]], "°."]
{c1MIN, a1MIN, c2MIN, a2MIN} = Table[ρciaiMIN[[iρ, j]], {j, 2, 5}]
{c1MAX, a1MAX, c2MAX, a2MAX} = Table[ρciaiMAX[[iρ, j]], {j, 2, 5}]
Clear[iρ]

These constants are for sources confined to regions with radii ρ = 24°.

Out[46]= {0.9235, 0.50345, 1.127, 0.482}

Out[47]= {0.927, 0.5034, 1.101, 0.482}

In[49]:= (*Change the region radius, if necessary, in Section 3 Inputs and Settings. *)
iρ = regionChoice + 1; (* ± uncertainty for the parameters ci and ai, i = 1,2. *)
Print["These uncertainties are for sources confined to regions with radii ρ = ",
 ρciaiMAX[[iρ, 1]], "°."]
{c1MINplusMinus, a1MINplusMinus, c2MINplusMinus, a2MINplusMinus} =
 Table[ρΔciaiMIN[[iρ, j]], {j, 2, 5}]
{c1MAXplusMinus, a1MAXplusMinus, c2MAXplusMinus, a2MAXplusMinus} =
 Table[ρΔciaiMAX[[iρ, j]], {j, 2, 5}]
Clear[
 iρ]

These uncertainties are for sources confined to regions with radii ρ = 24°.

Out[51]= {0.0024, 0.0009, 0.022, 0.0065}

Out[52]= {0.011, 0.00395, 0.019, 0.0055}
```

In[54]:= $\eta 0\mathrm{MIN}[\mathrm{nSrc\_, c1\_, a1\_}] := \frac{\pi}{4} - \frac{\mathrm{c1}}{\mathrm{nSrc}^{\mathrm{a1}}}$

$\sigma\mathrm{MIN}[\mathrm{nSrc\_, c2\_, a2\_}] := \frac{\mathrm{c2}}{4\,\mathrm{nSrc}^{\mathrm{a2}}}$

In[56]:= $\eta 0\mathrm{MAX}[\mathrm{nSrc\_, c1\_, a1\_}] := \frac{\pi}{4} + \frac{\mathrm{c1}}{\mathrm{nSrc}^{\mathrm{a1}}}$

$\sigma\mathrm{MAX}[\mathrm{nSrc\_, c2\_, a2\_}] := \frac{\mathrm{c2}}{4\,\mathrm{nSrc}^{\mathrm{a2}}}$

The following probability distributions and significances make use of the above formulas for mean η_0 and half-width σ. They are functions of the alignment angle η and the number of sources N.

```
In[58]:= probMIN[η_, nSrc_] := probMIN0[ η, η0MIN[nSrc, c1MIN, a1MIN], σMIN[nSrc, c2MIN, a2MIN] ]

In[59]:= signiMIN[η_, nSrc_] := signiMIN0[η, η0MIN[nSrc, c1MIN, a1MIN], σMIN[nSrc, c2MIN, a2MIN]]

In[60]:= probMAX[η_, nSrc_] := probMAX0[ η, η0MAX[nSrc, c1MAX, a1MAX], σMAX[nSrc, c2MAX, a2MAX] ]
signiMAX[η_, nSrc_] := signiMAX0[η, η0MAX[nSrc, c1MAX, a1MAX], σMAX[nSrc, c2MAX, a2MAX]]
```

5. Grid

We avoid bunching at the poles by taking into account the diminishing radii of constant latitude circles as the latitude approaches the poles. Successive grid points along any latitude or along any longitude make an arc that subtends the same central angle dθ.
We grid one hemisphere at a time, then they are combined.

Definitions:

gridSpacing separation in degrees between grid points on a constant latitude circle and separation of constant latitude circles. Set by the user in Sec. 2.

dθ	grid spacing in radians
αpointH,δpointH	RA and dec of the grid points H_j
grid	see listing below for “grid” table entries
nGrid	number of grid points H_j , $j = 1, 2, ...,$ nGrid
rGrid	radial unit vectors from origin to grid points, in 3D Cartesian coordinates
αGrid	RAs for grid points
δGrid	decs for grid points

Tables:

grid, gridN* and *gridS

1. sequential point # 2. RA index 3. dec index 4. RA (rad) 5. dec (rad) 6. Cartesian coordinates of the grid point

In[62]:=
```
(*When gridSpacing = 2°, we get a 2°x2° grid.*)
Print["The grid spacing is a setting that was chosen in Sec. 3 to be gridSpacing = ",
 gridSpacing, "°."]
dθ = (2. π)/(360.) gridSpacing; (*Convert gridSpacing to radians*)
```

```
The grid spacing is a setting that was chosen in Sec. 3 to be gridSpacing = 2.°.

The grid spacing has been chosen in Sec. 3 to be gridSpacing = 2.°.
```

In[64]:=
```
(*The Northern Grid "gridN". *)
gridN = {}; idN = 1;

For[δj = 0., δj < π/(2. dθ), δj++, δpointH = δj dθ;
 For[ ai = 0., ai < Ceiling[(2. π)/dθ (Cos[δpointH] + 0.01)], ai++, αpointH = ai dθ/(Cos[δpointH] + 0.01);
  AppendTo[gridN, {idN, ai, δj, αpointH, δpointH, er[αpointH, δpointH]}];
  idN = idN + 1
 ]]
```

In[66]:=
```
(*The Southern Grid "gridS". *)
gridS = {}; idS = 1;

For[δj = 1., δj < π/(2. dθ), δj++, δpointH = -δj dθ;
 (*Print["{δj,δpointH} = ",{δj,δpointH}];*)
 For[ ai = 0., ai < Ceiling[(2. π)/dθ (Cos[δpointH] + 0.01)], ai++, αpointH = ai dθ/(Cos[δpointH] + 0.01);
  (*Print["{ai,αpointH} = ",{ai,αpointH}];*)
  AppendTo[gridS, {idS, ai, δj, αpointH, δpointH, er[αpointH, δpointH]}];
  idS = idS + 1
 ]]
```

In[68]:=
```
grid = {}; j = 1;
For[jN = 1, jN ≤ Length[gridN], jN++, AppendTo[grid,
   {j, gridN[[jN, 2]], gridN[[jN, 3]], gridN[[jN, 4]], gridN[[jN, 5]], gridN[[jN, 6]]}];
 j = j + 1]
For[jS = 1, jS ≤ Length[gridS], jS++, AppendTo[grid,
   {j, gridS[[jS, 2]], gridS[[jS, 3]], gridS[[jS, 4]], gridS[[jS, 5]], gridS[[jS, 6]]}];
 j = j + 1]
nGrid = Length[grid];
```

In[72]:=
```
αGrid = Table[αFROMr[grid[[j, 6]] ], {j, Length[grid]}];
δGrid = Table[δFROMr[grid[[j, 6]] ], {j, Length[grid]}];
rGrid = Table[grid[[j, 6]] , {j, Length[grid]}];
```

In[75]:=
```
Print["There are ", nGrid, " points on the grid. "]
```

```
There are 10518 points on the grid.
```

6. Analysis of the best values input

Definitions:

vψSrc	unit vectors along the polarization directions in the tangent planes of the sources
eNSrc	unit vectors along local North in the tangent planes of the sources
eESrc	unit vectors along local East in the tangent planes of the sources

jηBarHj {j,$\overline{\eta}$(H)}, where j is the index for grid point H_j and $\overline{\eta}$(H) is the average alignment angle at H_j. See Eq. (1) in the Introduction.

sortjηBarHj	{j,$\overline{\eta}$(H)}, rearranged by value of $\overline{\eta}$(H), with smallest angles $\overline{\eta}$(H) first.
jηBarMin	{j,$\overline{\eta}$(H)}, the j and $\overline{\eta}$ for the smallest value of $\overline{\eta}$(H) , best alignment
ηBarMin	the smallest value of $\overline{\eta}$(H) , measures alignment of the polarization directions
jηBarMax	{j,$\overline{\eta}$(H)}, the j and $\overline{\eta}$ for the largest value of $\overline{\eta}$(H) , most avoided
ηBarMax	the largest value of $\overline{\eta}$(H) , measures avoidance
sigηBarMin	significance of the smallest alignment angle
sigRangeηBarMin	using the plus/minus values on the parameters c_i and a_i, the table collects corresponding values of the significance
sigSmallηBarMin	the smallest of the values in sigRangeηBarMin

sigBigηBarMin	the largest of the values in sigRangeηBarMin
sigηBarMax	significance of the largest alignment angle (i.e. avoidance)
sigRangeηBarMax	using the plus/minus values on the parameters c_i and a_i, the table collects corresponding values of the significance
sigSmallηBarMax	the smallest of the values in sigRangeηBarMax
sigBigηBarMax	the largest of the values in sigRangeηBarMax
αHminDegrees	RA of the point $H_{\min}$ where $\overline{\eta}$(H) is the smallest
δHminDegrees	dec of the point $H_{\min}$ where $\overline{\eta}$(H) is the smallest
αHmaxDegrees	RA of the point $H_{\max}$ where $\overline{\eta}$(H) is the largest
δHmaxDegrees	dec of the point $H_{\max}$ where $\overline{\eta}$(H) is the largest

In[76]:=

```
(* vψ, eN, eE unit vectors in the tangent plane of each source Si,
pointing along the polarization direction, local North, and local East, respecively.*)
vψSrc = Table[Cos[ψn[[i]]] eN[ αSrc[[i]], δSrc[[i]] ] +
    Sin[ψn[[i]]] eE[ αSrc[[i]], δSrc[[i]] ], {i, nSrc}];
eNSrc = Table[eN[ αSrc[[i]], δSrc[[i]] ], {i, nSrc}];
eESrc = Table[eE[ αSrc[[i]], δSrc[[i]] ], {i, nSrc}];
```

In[79]:=

```
(* Analysis using Eq (5) in the article1 to get ηiH,
cos(η) = |v̂H.v̂ψ|, then {j,η̄(Hj)}. *)
jηBarHj =
  Table[{j, (1/nSrc) Sum[ArcCos[ Abs[ rGrid[[j]].vψSrc[[i]] / ((rGrid[[j]] - (rGrid[[j]].
              rSrc[[i]]) rSrc[[i]]).(rGrid[[j]] - (rGrid[[j]].rSrc[[i]])
            rSrc[[i]]))^(1/2) ] - 0.000001 ] , {i, nSrc}]}, {j, nGrid}];
 sortjηBarHj = Sort[jηBarHj, #1[[2]] < #2[[2]] &];
jηBarMin = sortjηBarHj[[1]]; (*  {j,η̄(Hj)} for smallest η̄(Hj) *)
ηBarMin = jηBarMin[[2]];
jηBarMax = sortjηBarHj[[-1]]; (*  {j,η̄(Hj)} for largest η̄(Hj) *)
ηBarMax = jηBarMax[[2]] ;
```

In[85]:=
```
(*Alternate analysis using Eq (7) in the article1 to get ηiH, cos(η) = |n̂Sxψ.n̂SxH|.*)
(*nSxψn  =  Table[ Sin[ψn[[n]]]eN[αSrc[[n]],δSrc[[n]]]-
   Cos[ψn[[n]]]eE[αSrc[[n]],δSrc[[n]]],    {n,nSrc}];
nSxHnj[j_]:=nSxHnj[j]=Table[ Cross[  rSrc[[n]],rGrid[[j]]  ]/
    (√((Cross[  rSrc[[n]],rGrid[[j]]  ]).(Cross[  rSrc[[n]],rGrid[[j]]  ]))) ,    {n,
    nSrc}];
ηnHj[j_]:=ηnHj[j]=Table[ ArcCos[  Abs[  nSxψn[[n]].nSxHnj[j][[n]]  ] -
     0.000001  ],      {n,nSrc}];
ηBarHj[j_]:=ηBarHj[j]=Sum[ηnHj[j][[n]],{n,nSrc}]/nSrc
   jηBarHj=Table[{j,ηBarHj[j]},{j,Length[grid]}];
sortjηBarHj=Sort[jηBarHj,#1[[2]]<#2[[2]]&];
jηBarMin=sortjηBarHj[[1]];
ηBarMin=jηBarMin[[2]]
  jηBarMax=sortjηBarHj[[-1]];
ηBarMax=jηBarMax[[2]]*)
```

In[86]:=
```
(*Significance of the alignment of the polarization directions with hub point Hmin.*)
sigηBarMin = signiMIN[ηBarMin, nSrc];
sigRangeηBarMin = Sort[Partition[Flatten[Table[
      {signiMIN0[ηBarMin, η0MIN[nSrc, c1MIN + γ1 c1MINplusMinus, a1MIN + α1 a1MINplusMinus],
        σMIN[nSrc, c2MIN + γ2 c2MINplusMinus, a2MIN + α2 a2MINplusMinus]], γ1, α1, γ2, α2},
      {γ1, -1, 1}, {α1, -1, 1}, {γ2, -1, 1}, {α2, -1, 1}] ], 5 ]  ];
{sigRangeηBarMin[[1]], sigRangeηBarMin[[-1]]};
sigSmallηBarMin = sigRangeηBarMin[[1, 1]];
sigBigηBarMin = sigRangeηBarMin[[-1, 1]];
Print["The best value for the significance of alignment is sig. =  ", sigηBarMin,
 ". Using the uncertainties +/- of the ci,ai, the lowest and highest values are ",
 sigSmallηBarMin, " and ", sigBigηBarMin  , " giving the range from sig. = ",
 sigSmallηBarMin, " to ", sigBigηBarMin, " . "]
```

```
The best value for the significance of alignment is sig. =  0.0111662
 . Using the uncertainties +/- of the ci,ai, the lowest and highest values are
 0.00832443 and 0.0146188 giving the range from sig. = 0.00832443 to 0.0146188 .
```

In[92]:=
```
(*Significance of the polarization directions' avoidance of the hub point Hmax.*)
sigηBarMax = signiMAX[ηBarMax, nSrc];
sigRangeηBarMax = Sort[Partition[Flatten[Table[
      {signiMAX0[ηBarMax, η0MAX[nSrc, c1MAX + γ1 c1MAXplusMinus, a1MAX + α1 a1MAXplusMinus],
        σMAX[nSrc, c2MAX + γ2 c2MAXplusMinus, a2MAX + α2 a2MAXplusMinus]], γ1, α1, γ2, α2},
      {γ1, -1, 1}, {α1, -1, 1}, {γ2, -1, 1}, {α2, -1, 1}] ], 5 ]  ];
{sigRangeηBarMax[[1]], sigRangeηBarMax[[-1]]};
sigSmallηBarMax = sigRangeηBarMax[[1, 1]];
sigBigηBarMax = sigRangeηBarMax[[-1, 1]];
Print["The best value for the significance of avoidance is sig. =  ", sigηBarMax,
 ". Using the uncertainties +/- of the ci,ai, the lowest and highest values are ",
 sigSmallηBarMax, " and ", sigBigηBarMax  , " giving the range from sig. = ",
 sigSmallηBarMax, " to ", sigBigηBarMax, " . "]
```

```
The best value for the significance of avoidance is sig. =  0.00636211
 . Using the uncertainties +/- of the ci,ai, the lowest and highest values are
 0.00397639 and 0.00975809 giving the range from sig. = 0.00397639 to 0.00975809 .
```

In[98]:=
```
{jηBarMin, jηBarMax} ;(* {1. grid#, 2. alignment angle η} at Min and Max η .*)
αHminDegrees0 = grid[[ jηBarMin[[1]] ]][[4]] (360 / (2 π));
δHminDegrees0 = grid[[ jηBarMin[[1]] ]][[5]] (360 / (2 π));
If[(180 < αHminDegrees0 < 361), αHminDegrees = αHminDegrees0 - 180;
  δHminDegrees = -δHminDegrees0 , αHminDegrees = αHminDegrees0;
  δHminDegrees = δHminDegrees0];
αHmaxDegrees0 = grid[[ jηBarMax[[1]] ]][[4]] (360 / (2 π));
δHmaxDegrees0 = grid[[ jηBarMax[[1]] ]][[5]] (360 / (2 π));
If[(180 < αHmaxDegrees0 < 361), αHmaxDegrees = αHmaxDegrees0 - 180;
  δHmaxDegrees = -δHmaxDegrees0 , αHmaxDegrees = αHmaxDegrees0;
  δHmaxDegrees = δHmaxDegrees0];
Print["The alignment hub H_min is located at (RA,dec) = ", {αHminDegrees, δHminDegrees },
 " and at ", {αHminDegrees - 180, -δHminDegrees }, " , in degrees"]
Print["The avoidance hub H_max is located at (RA,dec) = ", {αHmaxDegrees, δHmaxDegrees },
 " and at ", {αHmaxDegrees - 180, -δHmaxDegrees }, " , in degrees"]
```

```
The alignment hub H_min is located at (RA,dec) =
 {106.408, -20.} and at {-73.5915, 20.} , in degrees

The avoidance hub H_max is located at (RA,dec) =
 {9.93072, -22.} and at {-170.069, 22.} , in degrees
```

In[107]:=
```
(*The names are used again below,
so save the current values. "Best" means we used th ψn that
  were input in Sec. 3. Later we allow ψn + δψ.*)
{jηBarMinBest, jηBarMaxBest} = {jηBarMin, jηBarMax} ;
(* {1. grid#, 2. alignment angle η} at Min and Max η .*)
```

7. Plot of the alignment function $\overline{\eta}$(H) using the best values input

Definitions

αjδjηBarHjTable	{RA$_j$, dec$_j$, $\overline{\eta}$(H)} at each grid point $H = H_j$, in degrees
ηBarHjSmooth	interpolation of αjδjηBarHjTable yields $\overline{\eta}$(H) as a smooth function of the (RA,dec) of H
xyηBarAitoffTable	{x, y, $\overline{\eta}$(x,y)} , where x,y are Aitoff coordinates and $\overline{\eta}$(x,y) is the alignment angle
dηContourPlot	separation of successive contour lines, in degrees
listCP	list contour plot of $\overline{\eta}$(H), from xyηBarAitoffTable
xyAitoffSources	{x,y} Aitoff coordinates for the sources' locations on the sphere
mapOfηBar	contour plot listCP of the alignment angle $\overline{\eta}$(H) , with source locations and labels

αH(α,δ) , xH(α,δ) , yH(α,δ) are functions needed when making a 2-D map of the Celestial sphere. The origin xH, yH is centered on $\alpha = \delta = 0$.

Notice the naming conflict: αH(α,δ) is an Aitoff parameter which, in general, differs from the Right Ascension α .

In[108]:=
```
(*The following table αjδjηBarHjTable is interpolated below
 to yield a smooth function of the alignment angle over the sphere.*)
(* Table Entries: 1.  RA at jth grid point (degrees) 2. dec at jth grid
   point (degrees)  3. alignment angle ηBarRgnkj at jth grid point (degrees)*)
αjδjηBarHjTable = (αjδjηBarHjTable0 = {};
   For[j = 1, j ≤ Length[jηBarHj], j++,
    AppendTo[ αjδjηBarHjTable0,  {grid[[j, 4]] * (360. / (2. π)), grid[[j, 5]] * (360. / (2. π)),
      jηBarHj[[j, 2]] * (360. / (2. π))}]   ; If[ 360 ≥ grid[[j, 4]] * (360. / (2. π)) > 354.,
     AppendTo[ αjδjηBarHjTable0,  {grid[[j, 4]] * (360. / (2. π)) - 360.,
       grid[[j, 5]] * (360. / (2. π)), jηBarHj[[j, 2]] * (360. / (2. π))}]   ] ;
    If[ 6. > grid[[j, 4]] * (360. / (2. π)) ≥ 0., AppendTo[ αjδjηBarHjTable0,
      {grid[[j, 4]] * (360. / (2. π)) + 360, grid[[j, 5]] * (360. / (2. π)),
       jηBarHj[[j, 2]] * (360. / (2. π))}]   ]         ];
   αjδjηBarHjTable0);
```

In[109]:=
```
ηBarHjSmooth = Interpolation[αjδjηBarHjTable]
 (*The smooth alignment angle function for the region.*)
```

Interpolation: Interpolation on unstructured grids is currently only supported for InterpolationOrder->1 or InterpolationOrder->All. Order will be reduced to 1.

Out[109]= InterpolatingFunction[Domain: {{-5.92, 366.}, {-88., 88.}} Output: scalar]

The following Aitoff Plot formulas[4] were be found in, for example, Wikipedia contributors. “Aitoff projection.” Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 25 May. 2017. Web. 3 Jan. 2018.

In[110]:=
```
αH[α_, δ_] := αH[α, δ] = ArcCos[Cos[2. π/360. δ] Cos[2. π/360. α / 2.]] (*angles α and δ are in degrees*)

xH[α_, δ_] := xH[α, δ] = (2. Cos[2. π/360. δ] Sin[2. π/360. α / 2.]) / Sinc[αH[α, δ]]

yH[α_, δ_] := yH[α, δ] = Sin[2. π/360. δ] / Sinc[αH[α, δ]]
```

In[113]:=
```
xyηBarAitoffTable = Partition[Flatten[Table[{xH[α - 180, -δ], yH[α - 180, -δ], ηBarHjSmooth[α, δ]},
      {α, 0, 360., 2.}, {δ, -88., 88., 2.}]], 3];
(* The smooth alignment angle function ηBarHjSmooth mapped onto a 2D
 Aitoff projection of the sphere. *)

xyAitoffSources =
  Table[{xH[ αSrc[[n]] 360/(2 π), δSrc[[n]] 360/(2 π) ], yH[ αSrc[[n]] 360/(2 π), δSrc[[n]] 360/(2 π) ]}, {n, nSrc}];
 (*The Aitoff coordinates for the sources' locations.*)
xyAitoffOppositeSources =
  Table[{xH[ If[0 < αSrc[[n]] 360/(2 π) < +180, αSrc[[n]] 360/(2 π) - 180, If[0 > αSrc[[n]] 360/(2 π) > -180,
       αSrc[[n]] 360/(2 π) + 180]], -δSrc[[n]] 360/(2 π) ], yH[ If[0 < αSrc[[n]] 360/(2 π) < +180, αSrc[[n]] 360/(2 π) -
       180, If[0 > αSrc[[n]] 360/(2 π) > -180, αSrc[[n]] 360/(2 π) + 180]], -δSrc[[n]] 360/(2 π) ]}, {n, nSrc}];
```

In[116]:=
```
(* Contour plot of the alignment function ηBarHjSmooth. *)
listCP = ListContourPlot[Union[xyηBarAitoffTable(*,
    {{xH[αHminDegrees,δHminDegrees],yH[αHminDegrees,δHminDegrees],ηBarMin*(360./(2.π))-1.0}},
    {{xH[αHmaxDegrees,δHmaxDegrees],yH[αHmaxDegrees,δHmaxDegrees],
      ηBarMax*(360./(2.π))+1.0}}*)], AspectRatio → 1/2,
   Contours → Table[η, {η, Floor[jηBarMin[[2]] * (360. / (2. π))] + 1,
      Ceiling[jηBarMax[[2]] * (360. / (2. π))] - 1, dηContourPlot}],
   ColorFunction → "TemperatureMap", PlotRange → {{-7, 7}, {-3, 3}}, Axes -> False, Frame → False] ;
```

In[117]:=
```
(*Construct the map of η̄(H).*)
Print["The map is centered on (RA,dec) = (0°,0°)."]
Print["The map is symmetric across diameters, i.e.
   diametrically opposite points -H and H have the same alignment angle."]
Print["The contour lines are separated by ", dηContourPlot,
  "°. This setting was chosen in Sec. 3."]
Print["Source dots are Purple ", Purple,
  ", the dots opposite the sources are Magenta ", Magenta, "."]
Print["The best alignment angle (min) is η̄min = ", jηBarMin[[2]] (360. / (2. π)), "°.", Blue]
Print["The best avoidance angle (max) is η̄max = ", jηBarMax[[2]] (360. / (2. π)), "°.", Red]
Print["The alignment hubs Hmin and -Hmin are located at (RA,dec) = ",
  {αHminDegrees, δHminDegrees }, " and at ", {αHminDegrees - 180, -δHminDegrees }, " , in degrees."]
Print["The avoidance hubs Hmax and -Hmax are located at (RA,dec) = ",
  {αHmaxDegrees, δHmaxDegrees }, " and at ", {αHmaxDegrees - 180, -δHmaxDegrees }, " , in degrees."]
mapOfηBar =
  Show[{listCP,
    Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {δ, -90, 90}, PlotStyle → {Black, Thickness[0.002]},
       (*Mesh→{11,5,0}(*{23,11,0}*),MeshStyle→Thick,*)PlotPoints → 60], {α, -180, 180, 30}],
    Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {α, -180, 180},
      PlotStyle → {Black, Thickness[0.002]}, (*Mesh→{11,5,0}
      (*{23,11,0}*),MeshStyle→Thick,*)PlotPoints → 60], {δ, -60, 60, 30}],
    Graphics[{PointSize[0.007], Text[StyleForm["N", FontSize -> 10, FontWeight -> "Plain"],
        {0, 1.85}], (*Sources S:*)Purple, Point[ xyAitoffSources ],
      (*Opposite from sources, -S:*)Magenta, Point[xyAitoffOppositeSources],
       Black, Text[StyleForm["Max", FontSize → 8, FontWeight -> "Bold"],
        {xH[ - 180 , 0], yH[0, -60]}], {Arrow[BezierCurve[{{xH[ - 180 , 0], yH[0, -70]}, {-2.3, -2.0},
           {xH[αHmaxDegrees - 180, -δHmaxDegrees], yH[αHmaxDegrees - 180, -δHmaxDegrees]}}]]},
      Text[StyleForm["Min", FontSize → 8, FontWeight -> "Bold"], {xH[ 180 , 0], yH[0, -60]}],
      {Arrow[BezierCurve[{{xH[ 180 , 0], yH[0, -70]}, {2.3, -2.0},
           {xH[αHminDegrees, δHminDegrees], yH[αHminDegrees, δHminDegrees]}}]]},
      Text[StyleForm["Min", FontSize → 8, FontWeight -> "Bold"], {xH[ -180 , 0], yH[0, 60]}],
      {Arrow[BezierCurve[{{xH[ -180 , 0], yH[0, 70]}, {-2.3, 2.0},
           {xH[αHminDegrees - 180, -δHminDegrees], yH[αHminDegrees - 180, -δHminDegrees]}}]]},
      Text[StyleForm["Max", FontSize → 8, FontWeight -> "Bold"], {xH[ 180 , 0], yH[0, 60]}] ,
      {Arrow[BezierCurve[{{xH[ 180 , 0], yH[0, 70]}, {2.3, 2.0}, {xH[αHmaxDegrees, δHmaxDegrees],
            yH[αHmaxDegrees, δHmaxDegrees]}}]]}       }]}, ImageSize → 432]
Print["Caption: A map of the alignment function η̄(H), Eq. (1). "]
```

The map is centered on (RA,dec) = (0°,0°).

The map is symmetric across diameters, i.e.
  diametrically opposite points -H and H have the same alignment angle.

The contour lines are separated by 4°. This setting was chosen in Sec. 3.

Source dots are Purple ■, the dots opposite the sources are Magenta ■.

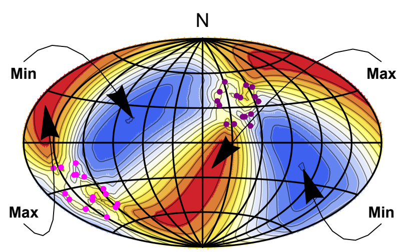
The best alignment angle (min) is $\overline{\eta}_{min}$ = 21.8882°.■

The best avoidance angle (max) is $\overline{\eta}_{max}$ = 68.769°.■

The alignment hubs H_{min} and $-H_{min}$ are located at (RA,dec) =
 {106.408, -20.} and at {-73.5915, 20.} , in degrees.

The avoidance hubs H_{max} and $-H_{max}$ are located at (RA,dec) =
 {9.93072, -22.} and at {-170.069, 22.} , in degrees.

Out[125]=



Caption: A map of the alignment function $\overline{\eta}(H)$, Eq. (1).

In[127]:=
```
(*Export the map "mapOfηBar" as a pdf. The export location can be reset in Sec. 3.*)
(*To activate, remove the remark brackets "(*" and "*)". *)
(*SetDirectory[dataDirectory];
Export["mapOfEtaBarExample.pdf",
 Show[mapOfηBar,ImageSize→432],"PDF",ImageSize→{480,Automatic}]*)
```

In[392]:=
```
Print["The number of sources: N = ", nSrc]
Print["The min alignment angle is ηmin = ", jηBarMin[[2]] * (360. / (2. π)),
 "° , which has a significance of sig. = ", sigηBarMin, ",  plus/minus = + ",
 sigBigηBarMin - sigηBarMin, " and - ", sigηBarMin - sigSmallηBarMin,
 " , giving a range from sig. = ", sigSmallηBarMin, " to ", sigBigηBarMin, " ."]
Print["The max avoidance angle is ηmax = ", jηBarMax[[2]] * (360. / (2. π)),
 "° , which has a significance of sig. = ", sigηBarMax, ",  plus/minus = + ",
 sigBigηBarMax - sigηBarMax, " and - ", sigηBarMax - sigSmallηBarMax,
 " , giving a range from sig. = ", sigSmallηBarMax, " to ", sigBigηBarMax, " ."]
Print["These uncertainties are due to the uncertainties in the
   constants aᵢ and cᵢ used in the significance formulas in Sec. 4."]
```

```
The number of sources: N = 16
The min alignment angle is ηmin = 26.8088
 ° , which has a significance of sig. = 0.0111662,  plus/minus = + 0.0034526
  and - 0.00284176 , giving a range from sig. = 0.00832443 to 0.0146188 .
The max avoidance angle is ηmax = 63.5621
 ° , which has a significance of sig. = 0.00636211,  plus/minus = + 0.00339597
  and - 0.00238572 , giving a range from sig. = 0.00397639 to 0.00975809 .
These uncertainties are due to the uncertainties in
  the constants ai and ci used in the significance formulas in Sec. 4.
```

8. Repeatedly running the process to determine uncertainties

For each run, let the polarization direction $\psi = \psi n + \delta\psi n$ for each source is allowed to differ from the best value ψn by an amount chosen according to a Gaussian distribution with mean (best) value ψn and half-width $\sigma\psi$n, both values ψn and $\sigma\psi$n taken from the input in Sec. 3.

Definitions:

rSrcxrGrid — unit vector cross product of rSrc for S_i and rGrid for H_j

$\mu = \psi_n$ — by convention, the best value ψ_n, input in Sec. 3, is the mean value μ of a Gaussian of half-width $\sigma_{\psi n}$, $\psi \pm \sigma\psi$

$\sigma = \sigma\psi_n$ — uncertainty of the measured polarization position angle ψ, an input in Sec. 3

runData — collection of data from the uncertainty $\sigma\psi$ runs

nRunPrint — dummy index controlling when TimeUsed and MemoryInUse data are printed

ψSrc — a polarization direction ψ for the run. This ψ is moved off the best value ψ_n by an increment determined by the uncertainty $\sigma\psi$

rSrcxψSrc — unit vector cross product of rSrc for S_i and vSrc for v_ψ

jηBarToGrid — {j, $\overline{\eta}(H_j)$}, where j is the index # for the grid point H_j and $\overline{\eta}(H_j)$ is the average of the alignment angles for H_j with the sources.

sortjηBarToGrid — {j, $\overline{\eta}(H_j)$}, reordered by the value of $\overline{\eta}$(H), with smallest angles $\overline{\eta}$(H) first.

jηBarHj — {j,$\overline{\eta}$(H)}, where j is the index for grid point H_j and $\overline{\eta}$(H) is the average alignment angle at H_j. See Eq. (1) in the Introduction.

sortjηBarHj — {j,$\overline{\eta}$(H)}, rearranged by value of $\overline{\eta}$(H), with smallest angles $\overline{\eta}$(H) first.

jηBarMin — {j,$\overline{\eta}$(H)}, the j and $\overline{\eta}$ for the smallest value of $\overline{\eta}$(H) , best alignment

jηBarMax — {j,$\overline{\eta}$(H)}, the j and $\overline{\eta}$ for the largest value of $\overline{\eta}$(H) , most avoided

ηBarMin — the smallest value of $\overline{\eta}$(H) , measures alignment of the polarization directions

ηBarMax — the largest value of $\overline{\eta}$(H) , measures avoidance

Table:

runData — entries: 1. Run # 2. ψSrc, list of polarization position angles ψ 3. {$\overline{\eta}_{\rm min}$, {$\alpha$,$\delta$} at $H_{\rm min}$} 4. {$\overline{\eta}_{\rm max}$, {$\alpha$,$\delta$} at $H_{\rm max}$}

```
In[131]:= rSrcxrGrid1 = Table[ Cross[ rSrc[[i]], rGrid[[j]] ] ,   {i, nSrc}, {j, nGrid}];
(*first step: raw cross product, not unit vectors*)
rSrcxrGrid = Table[ rSrcxrGrid1[[i, j]] /
    (rSrcxrGrid1[[i, j]].rSrcxrGrid1[[i, j]] + 0.000001)^(1/2.) ,   {i, nSrc}, {j, nGrid}];
Clear[rSrcxrGrid1];
(*rSrcxrGrid: table of the unit vectors perpendicular to the plane
  of the great circle containing the source Si and the grid point Hj*)
```

```
In[133]:= μ = ψn;
σ = σψn;
runData = {}; nRunPrint = 0;
For[nRun = 1, nRun ≤ 2000, nRun++,
 If[nRun > nRunPrint, Print["At the start of run ", nRun, ", the time is ",
    TimeUsed[], " seconds and the memory in use is ", MemoryInUse[], " bytes."];
  nRunPrint = nRunPrint + 200];
  ψSrc = Table[RandomVariate[NormalDistribution[μ[[i]], σ[[i]]]], {i, nSrc}];
 (*table of PPA angles ψ for the sources in region j0, in radians*)
 rSrcxψSrc  = Table[ Sin[ψSrc[[i]]] eNSrc[[i]] - Cos[ψSrc[[i]]] eESrc[[i]],   {i, nSrc}];
 (*table of the cross product of rSrc and vector in direction of ψSrc,
 a unit vector*)jηBarToGrid  = Table[{j, (1 / nSrc) Sum[ArcCos[
       Abs[ rSrcxψSrc[[i]].rSrcxrGrid[[i, j]] ] - 0.000001 ], {i, nSrc}]}, {j, nGrid}];
 (*
 {grid point #, value of the alignment angle ηnHj[j] averaged over all sources,
  in radians}*) sortjηBarToGrid = Sort[jηBarToGrid, #1[[2]] < #2[[2]] &];
 (*jηBarToGrid, {j,ηj}, but sorted with the smallest alignment angles first
 *)
 jηBarMin = sortjηBarToGrid[[1]]; (*  {j,ηj}, at the grid point Hj with minimum η̄*)
 jηBarMax = sortjηBarToGrid[[-1]]; (*  {j,ηj},
 at the grid point Hj with maximum η̄*)AppendTo[runData,
  {nRun, ψSrc, { jηBarMin[[2]], {αGrid [ [ jηBarMin[[1]] ]], δGrid [[ jηBarMin[[1]] ]]}},
   { jηBarMax[[2]], {αGrid [[ jηBarMax[[1]] ]], δGrid [[ jηBarMax[[1]] ]]}}} ]
  (*collect data*)  ]
```

```
At the start of run 1, the time is 15.779 seconds and the memory in use is 177464400 bytes.

At the start of run 201, the time is 171.81 seconds and the memory in use is 188962568 bytes.

At the start of run 401, the time is 327.013 seconds and the memory in use is 189114760 bytes.

At the start of run 601, the time is 480.92 seconds and the memory in use is 189266888 bytes.

At the start of run 801, the time is 636.435 seconds and the memory in use is 189419144 bytes.

At the start of run 1001, the time is 792.701 seconds and the memory in use is 189571272 bytes.

At the start of run 1201, the time is 949.31 seconds and the memory in use is 189726184 bytes.

At the start of run 1401, the time is 1105.76 seconds and the memory in use is 189881448 bytes.

At the start of run 1601, the time is 1262.09 seconds and the memory in use is 190037224 bytes.

At the start of run 1801, the time is 1421.89 seconds and the memory in use is 190192552 bytes.
```

```
In[137]:= Print[ "The number of values in the table runData is ", Length[Flatten[runData]] ]
```

```
The number of values in the table runData is 46000
```

In[138]:=
```
(*To save the runData table to a file,
remove the comment marks (* and *) from the following statements.*)
(*   SetDirectory[dataDirectory]
 Put[runData,"20210110runData.dat"  ]  *)
```

9. Uncertainty in the alignment angle $\overline{\eta}_{\mathrm{min}}$

This section fits a Gaussian distribution to the $\overline{\eta}_{\mathrm{min}}$ from the data files

Definitions

sortηBarMin	list of $\overline{\eta}_{\mathrm{min}}$ from the data files, sorted small to large
η0B	estimated mean of the Gaussian fit
σB	estimated half-width of the Gaussian fit
histogramRANGE	{min η, max η, $\Delta\eta$} for the histogram
hist1	histogram collecting the number of runs with $\overline{\eta}_{\mathrm{min}}$ in intervals $\Delta\eta$ from min η to max η
hl0, hl	tables needed to set up the NonlinearModelFit
lphl	list plot of the histogram table hl
nlmB	non-linear model fit of a Gaussian to the $\overline{\eta}_{\mathrm{min}}$ histogram
normalNLMB	convert the fit to an expression
showNLMB	plot of Gaussian and histogram
parametersNLMB	amplitude, half-width, and mean of the Gaussian fit
pTableNLMB	table of parameter attributes, including standard error

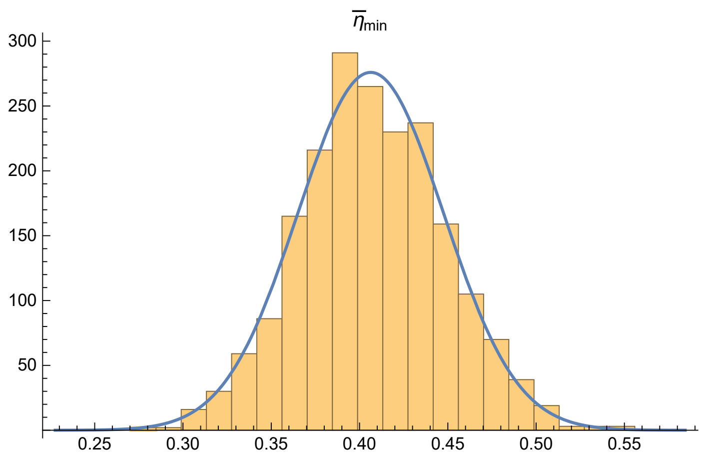
Table:

runData entries: 1. Run # 2. ψSrc, list of polarization position angles ψ 3. {$\overline{\eta}_{\mathrm{min}}$, {$\alpha$,$\delta$} at H_{min}} 4. {$\overline{\eta}_{\mathrm{max}}$, {$\alpha$,$\delta$} at H_{max}}

```
In[139]:= sortηBarMin = Sort[Table[runData[[i1, 3, 1]] , {i1, Length[runData]}]];
ListPlot[%];
η0B = sortηBarMin[[Floor[1/2 Length[sortηBarMin ]]]]; (*Guess the mean. *)
σB = sortηBarMin [[Floor[4/5 Length[sortηBarMin ]]]] - η0B; (*Guess the width.*)
histogramRANGE = {η0B - 5 σB, η0B + 5 σB, 0.4 σB};
hist1 = Histogram[sortηBarMin, histogramRANGE, PlotLabel → "η̄min"];
hl0 = HistogramList[sortηBarMin, histogramRANGE];
{Length[ hl0[[1]] ], Length[ hl0[[2]] ]};
hl =
  Table[{(1/2) (hl0[[1, i1]] + hl0[[1, i1 + 1]]), hl0[[2, i1]]}, {i1, Length[ hl0[[2]] ]}];
lphl = ListPlot[hl, PlotLabel → "η̄min"];
Show[{hist1, lphl}];

In[150]:= nlmB = NonlinearModelFit[hl, a Exp[-1/2. ((x - x0)/b)^2], {{a, nR / 6}, {b, σB}, {x0, η0B}}, x];
(*x is ηBarMin, y is ΔR *)
normalNLMB = Normal[nlmB];
showNLMB = Show[{Histogram[sortηBarMin, histogramRANGE, PlotLabel → "η̄min"],
   Plot[Normal[nlmB], {x, η0B - 5 σB, η0B + 5 σB}, PlotLabel → "η̄min"] }]
parametersNLMB = {a, b, x0} /. nlmB["BestFitParameters"];
pTableNLMB = nlmB["ParameterTable"]
```

Out[152]=



Out[154]=

	Estimate	Standard Error	t-Statistic	P-Value
a	275.897	6.27789	43.9474	6.25265×10^{-23}
b	0.0412574	0.00108402	38.0595	1.4231×10^{-21}
x0	0.406325	0.00108402	374.831	2.32994×10^{-43}

```
In[155]:= {parametersNLMB[[2]] , parametersNLMB[[3]] }(*radians*)
{parametersNLMB[[2]] (360./(2. π)), parametersNLMB[[3]] (360./(2. π)) }(*degrees*)

Out[155]= {0.0412574, 0.406325}

Out[156]= {2.36388, 23.2807}
```

In[157]:=
```
{σηBarMinFit, ηBarMinFit} = {parametersNLMB[[2]] , parametersNLMB[[3]] };(*radians*)
{parametersNLMB[[2]] (360./(2. π)), parametersNLMB[[3]] (360./(2. π)) };(*degrees*)
Print[
 "Therefore, allowing the measured PPA ψ to vary according to their uncertainties in
   many runs, produces a value of  the alignment angle  η̄min  = ", ηBarMinFit (360./(2. π)),
 "° ± ", σηBarMinFit (360./(2. π)), "° , according to the Gaussian fit to the runs." ]
Print["The Gaussian mean η̄min = ", ηBarMinFit (360./(2. π)),
 "° has a significance of ", signiMIN[ηBarMinFit, nSrc], "."]
Print["The value η̄min + ση̄min = ", (ηBarMinFit + σηBarMinFit) (360./(2. π)),
 "° has a significance of ", signiMIN[ηBarMinFit + σηBarMinFit, nSrc], " ." ]
Print["The value η̄min - ση̄min = ", (ηBarMinFit - σηBarMinFit) (360./(2. π)),
 "°  has a significance of ", signiMIN[ηBarMinFit - σηBarMinFit, nSrc], " ." ]
```

Therefore, allowing the measured PPA ψ to vary according to their uncertainties in many runs, produces a value of the alignment angle $\overline{\eta}_{\text{min}}$ = 23.2807° ± 2.36388° , according to the Gaussian fit to the runs.

The Gaussian mean $\overline{\eta}_{\text{min}}$ = 23.2807° has a significance of 0.025768.

The value $\overline{\eta}_{\text{min}} + \sigma\overline{\eta}_{\text{min}}$ = 25.6446° has a significance of 0.0857122 .

The value $\overline{\eta}_{\text{min}} - \sigma\overline{\eta}_{\text{min}}$ = 20.9169° has a significance of 0.00588258 .

In[163]:=
```
bestVersusMeanMin = (Normal[nlmB] /. {x → jηBarMinBest[[2]]}) / parametersNLMB[[1]];
Print["The best ψn give an alignment angle of η̄min =
", jηBarMinBest[[2]] * (360. / (2. π)),
 "°, whose likelihood is a fraction  ", bestVersusMeanMin,
 " of the likelihood of the mean of the Gaussian, η̄min = ", ηBarMinFit (360./(2. π)), "° ."]
Print["The alignment angle  η̄min = ", jηBarMinBest[[2]] * (360. / (2. π)),
 "°,  found with the best ψn, has a significance of ",
 signiMIN[jηBarMinBest[[2]], nSrc], " ." ]
```

The best ψn give an alignment angle of $\overline{\eta}_{\text{min}}$ =
21.8882°, whose likelihood is a fraction 0.840703
of the likelihood of the mean of the Gaussian, $\overline{\eta}_{\text{min}}$ = 23.2807° .

The alignment angle $\overline{\eta}_{\text{min}}$ = 21.8882°, found with the best ψn, has a significance of 0.0111662 .

10. Uncertainty in the avoidance angle $\overline{\eta}_{\text{max}}$

This section fits a Gaussian distribution to the $\overline{\eta}_{\text{max}}$ from the data files.

Definitions

sortηBarMax	list of $\overline{\eta}_{max}$ from the data files, sorted small to large
η0MaxB	estimated mean of the Gaussian fit
σMaxB	estimated half-width of the Gaussian fit
histogramRANGEMAX	{min η, max η, $\Delta\eta$} for the histogram
hist1	histogram collecting the number of runs with $\overline{\eta}_{max}$ in intervals $\Delta\eta$ from min η to max η
hl0, hl	tables needed to set up the NonlinearModelFit
lphlMax	list plot of the histogram table hl
nlmMaxB	Gaussian fit to the $\overline{\eta}_{max}$ histogram
normalNLMMaxB	convert the fit to an expression
showNLMMaxB	plot of the Gaussian and the histogram that it fits
parametersNLMMaxB	amplitude, half-width, and mean of the Gaussian fit
pTableNLMMaxB	table of parameter attributes, including standard error

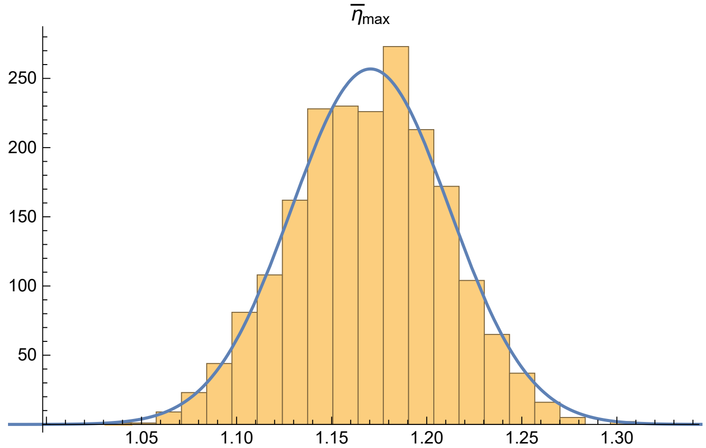
Copied here for reference when defining sortηBarMax:

runData entries: 1. Run # 2. ψSrc, list of polarization position angles ψ 3. {$\overline{\eta}_{min}$, {α,δ} at H_{min}} 4. {$\overline{\eta}_{max}$, {α,δ} at H_{max}}

In[166]:=
```
sortηBarMax = Sort[Table[runData[[i1, 4, 1]] , {i1, Length[runData]}]];
ListPlot[%];
η0MaxB = sortηBarMax [[Floor[1/2 Length[sortηBarMax ]]]];
σMaxB  = sortηBarMax [[Floor[4/5 Length[sortηBarMax ]]]] - η0MaxB;
histogramRANGEMAX = {η0MaxB - 5 σMaxB, η0MaxB + 5 σMaxB, 0.4 σMaxB};
hist1Max = Histogram[sortηBarMax, histogramRANGEMAX, PlotLabel → "η̄max"];
hl0Max = HistogramList[sortηBarMax, histogramRANGEMAX];
{Length[ hl0Max[[1]] ], Length[ hl0Max[[2]] ]};
hlMax = Table[{(1/2) (hl0Max[[1, i1]] + hl0Max[[1, i1 + 1]]), hl0Max[[2, i1]]},
   {i1, Length[ hl0Max[[2]] ]}];
lphlMax  = ListPlot[hl, PlotLabel → "η̄max"];
Show[{hist1Max, lphlMax}];
```

In[177]:=
```
nlmMaxB = NonlinearModelFit[hlMax, a Exp[-1/2. ((x - x0)/b)^2],
   {{a, 300.}, {b, σMaxB}, {x0, η0MaxB}}, x];(*x is ηBarMin, y is ΔR *)
normalNLMMaxB = Normal[nlmMaxB];
showNLMMaxB = Show[{Histogram[sortηBarMax, histogramRANGEMAX, PlotLabel → "η̅max"],
   Plot[Normal[nlmMaxB], {x, η0MaxB - 6 σMaxB, η0MaxB + 6 σMaxB}, PlotLabel → "η̅max"] }]
parametersNLMMaxB = {a, b, x0} /. nlmMaxB["BestFitParameters"];
pTableNLMMaxB = nlmMaxB["ParameterTable"]
```

Out[179]=



Out[181]=

	Estimate	Standard Error	t-Statistic	P-Value
a	256.819	5.85815	43.8395	6.59609×10^{-23}
b	0.0415884	0.00109541	37.966	1.50106×10^{-21}
x0	1.17044	0.00109541	1068.5	2.28724×10^{-53}

In[182]:=
```
{σηBarMaxFit, ηBarMaxFit} = {parametersNLMMaxB[[2]] , parametersNLMMaxB[[3]] };
(*radians*)
{parametersNLMMaxB[[2]] (360./(2. π)), parametersNLMMaxB[[3]] (360./(2. π)) };(*degrees*)
Print[
 "Therefore, allowing the measured PPA ψ to vary according to their uncertainties in
   many runs, produces a value of  the alignment angle  η̅max  = ", ηBarMaxFit (360./(2. π)),
 "° ± ", σηBarMaxFit (360./(2. π)), "° , according to the Gaussian fit to the runs."]
Print["The Gaussian mean η̅max = ", ηBarMaxFit (360./(2. π)),
 "° has a significance of ", signiMAX[ηBarMaxFit, nSrc], "."]
Print["The value η̅max + ση̅max = ", (ηBarMaxFit + σηBarMaxFit) (360./(2. π)),
 "° has a significance of ", signiMAX[ηBarMaxFit + σηBarMaxFit, nSrc], " ." ]
Print["The value η̅max - ση̅max = ", (ηBarMaxFit - σηBarMaxFit) (360./(2. π)),
 "°  has a significance of ", signiMAX[ηBarMaxFit - σηBarMaxFit, nSrc], " ." ]
```

Therefore, allowing the measured PPA ψ to vary according to their uncertainties in many runs, produces a value of the alignment angle $\overline{\eta}_{\max}$ = 67.0614° ± 2.38284° , according to the Gaussian fit to the runs.

The Gaussian mean $\overline{\eta}_{\max}$ = 67.0614° has a significance of 0.0192856.

The value $\overline{\eta}_{\max} + \sigma\overline{\eta}_{\max}$ = 69.4443° has a significance of 0.00393187 .

The value $\overline{\eta}_{\max} - \sigma\overline{\eta}_{\max}$ = 64.6786° has a significance of 0.0704099 .

In[188]:=
```
bestVersusMeanMax = (Normal[nlmMaxB] /. {x → jηBarMaxBest[[2]]}) / parametersNLMMaxB[[1]];
Print["The best ψn give an alignment angle of η̄max = ,
", jηBarMaxBest[[2]] * (360. / (2. π)),
 "°,  whose likelihood is a fraction  ", bestVersusMeanMax,
 " of the likelihood of the mean of the Gaussian,  η̄max = ", ηBarMaxFit (360./(2. π)), "° ."]
Print["The avoidance angle  η̄max = ", jηBarMaxBest[[2]] * (360. / (2. π)),
 "°,  found with the best ψn, has a significance of ",
 signiMAX[jηBarMaxBest[[2]], nSrc], " ." ]
```

The best ψn give an alignment angle of $\overline{\eta}_{\max}$ = ,
68.769°, whose likelihood is a fraction 0.773565
of the likelihood of the mean of the Gaussian, $\overline{\eta}_{\max}$ = 67.0614° .

The avoidance angle $\overline{\eta}_{\max}$ = 68.769°, found with the best ψn, has a significance of 0.00636211 .

11. Uncertainty in the locations of the alignment hubs $H_{\min}$

Find the location (α,δ) of $H_{\min}$ including uncertainty.

Issues:

(a) In any one run, the analysis produces an alignment angle $\overline{\eta}$ at each grid point. There can be just one minimum alignment angle $\overline{\eta}_{\min}$, and, therefore, just one grid point $H_{\min}$ determined. However, by the symmetry across a diameter, the diametrically opposite location $-H_{\min}$ should have the same minimum alignment angle, within the accuracy of the computed values. Note that $-H_{\min}$ may not be a grid point. So we expect the hubs to collect in diametrically opposed collections.

(b) The spread of near-minimum $\overline{\eta}$ may extend over a large portion of the Celestial sphere. The alignment angle function $\overline{\eta}$(H) may have more than one local minimum, so there may be several disparate places where hubs $H_{\min}$ appear. If more than one cluster of hubs $H_{\min}$ appear, then I plan to invent a reasonable response, either focus on just the one hub cluster that is the most popular, or, alternatively, perhaps I could analyse more than one.

(c) Since the hubs are grid points, the cluster of hubs may be so tightly determined that just a handful of grid points are populated. In such cases, the Gaussian fit is not appropriate and estimating the most likely location and the range of likely RAs and decs can be done by inspection, or by inventing a reasonable response.

A. By the symmetry across a diameter, a cluster of hubs $H_{\min}$ at one location implies the existance of a second cluster of hubs $-H_{\min}$

diametrically opposed. Collect the hubs by choosing one cluster and move the opposite hubs across the diameter.

B. Once we have collected the hubs we can find the most likely value for the location and estimate the uncertainty. Depending on the distribution of grid points, we may fit a Gaussian to RA = α and another to dec = δ. Alternatively, we may guess the result by inspection. The smallest uncertainty in α and δ is half the smallest division, so half the grid spacing is as precise as we can be.

C. Finally we plot the (α,δ) for the H_{min} and confirm the results from B.

Definitions

Hminα RA = α in radians for H_{min}, “0” is raw data, “1” has been worked on, ...
Hminδ dec = δ in radians for H_{min}, “0” is raw data, “1” has been worked on, ...
HminαAVE arithmetic average of the RAs, in radians
sortHαMin list of RA = α for H_{min} from the data files, sorted small to large
μ0αMinB estimate of the mean value for the RA = α of H_{min}
$\sigma\alpha$MinB estimate of the half-width of the RA = α pf H_{min}
histogramRangeMin parameter range for several histograms
histαMin histogram of the α for H_{min}
hloMin, hlMin tables of histogram data needed for plot and fit
lphlαMin list plot of histogram data, with dots not bars
nlmαMinB Gaussian fit to the histogram of α for H_{min}
normalNLMαMinB normal expression for Gaussian
showNLMαMinB plot of histogram and the Gaussian that fits it
parametersNLMαMinB values of the Gaussians parameters
pTableNLMαMinB table with values and standard errors of the parameters

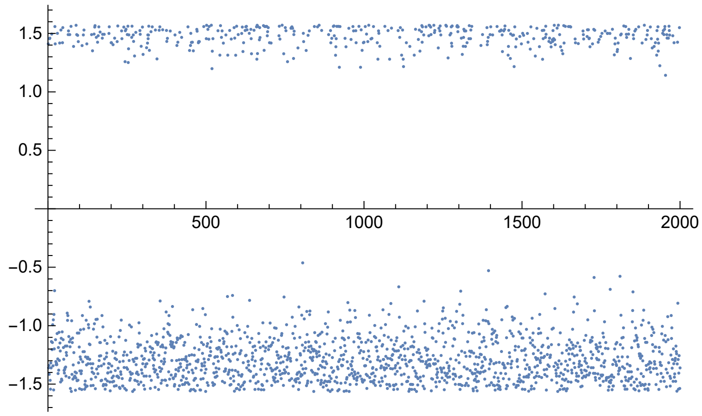
Many of the following sections have similarly named quantities with similar definitions.
(i) Replace “α” by “δ” for the sections dealing with the uncertainty in dec = δ.
(ii) Replace “min” with “max” in the context of the avoidance hubs H_{max}.

This information is copied here for convenience:

runData entries: 1. Run # 2. ψSrc, list of polarization position angles ψ 3. {$\overline{\eta}_{\text{min}}$, {$\alpha$,$\delta$} at H_{min}} 4. {$\overline{\eta}_{\text{max}}$, {$\alpha$,$\delta$} at H_{max}}

In[396]:=
```
(*A. Move hubs, if necessary, so that -180° ≤ α < 180° *)
Hminα0 = Table[ runData[[i1, 3, 2, 1]] , {i1, Length[runData ]}];
{Min[Hminα0], Max[Hminα0]};
ListPlot[Hminα0];
Hminδ0 = Table[runData[[i1, 3, 2, 2]], {i1, Length[runData ]}];
HminαBy180n = Round[Hminα0/π];
Hminα1 = Table[Hminα0[[i1]] - HminαBy180n[[i1]] π, {i1, Length[runData]}];
Hminδ1 = Table[(-1)^HminαBy180n[[i1]] Hminδ0[[i1]] , {i1, Length[runData]}];
{Min[Hminα1], Max[Hminα1]};
{Min[Hminδ1], Max[Hminδ1]};
ListPlot[Hminα1]
ListPlot[Hminδ1];
```

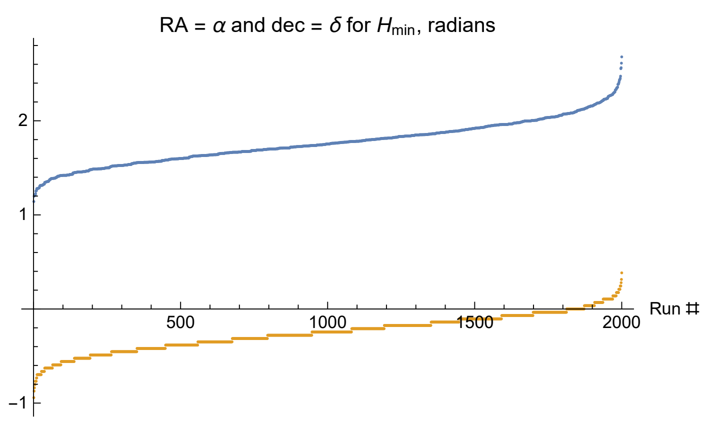
Out[404]=



There are two bands, one at RA = α = $\pi/2$ and one at α = $-\pi/2$, roughly. By the symmetry across a diameter, we can move all the hubs to the α = $+\pi/2$ band. The move across a diameter changes the sign of the dec = δs.

In[201]:=
```
Hminα = Table[
   If[Hminα1[[i1]] < 0, Hminα1[[i1]] + π, Hminα1[[i1]], "huh?"] , {i1, Length[runData ]}];
Hminδ = Table[If[Hminα1[[i1]] < 0, -Hminδ1[[i1]], Hminδ1[[i1]], "huh?"] ,
   {i1, Length[runData ]}];
ListPlot[{Sort[Hminα], Sort[Hminδ]}, PlotLabel → "RA = α and dec = δ for H_min, radians",
 AxesLabel → {"Run #"}]
```

Out[203]=



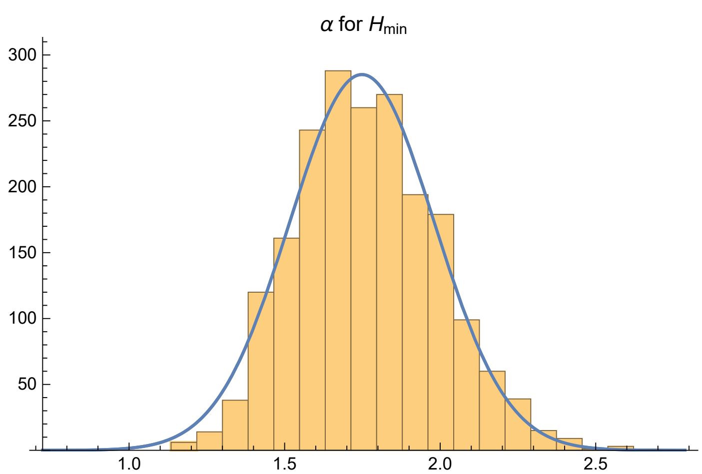
It looks like we can fit Gaussians to both α and δ for H_{min}. (The values are spread out over many times the grid spacing. When the values occupy only a few grid points, it may be better to judge the uncertainty another way.)

Fit a Gaussian to the α for $H_{\min}$.

In[204]:=
```
sortHαMin = Sort[Table[Hminα [[i2]], {i2, nR}]];
μ0αMinB = sortHαMin[[ Floor[nR/2] ]];
σαMinB  = sortHαMin[[Floor[4/5 Length[sortHαMin ]]]] - μ0αMinB;
histogramRangeMin = {μ0αMinB - 5 σαMinB, μ0αMinB + 5 σαMinB, 0.4 σαMinB};
histαMin = Histogram[sortHαMin, histogramRangeMin, PlotLabel → "α for Hmin"];
hl0Min = HistogramList[sortHαMin, histogramRangeMin];
{Length[ hl0Min[[1]] ], Length[ hl0Min[[2]] ]};
hlMin = Table[{(1/2) (hl0Min[[1, i1]] + hl0Min[[1, i1 + 1]]), hl0Min[[2, i1]]},
   {i1, Length[ hl0Min[[2]] ]}];
lphlαMin  = ListPlot[hlMin, PlotLabel → "α for Hmin"];
Show[{histαMin, lphlαMin}];
```

In[214]:=
```
nlmαMinB = NonlinearModelFit[hlMin, a Exp[-1/2. ((x - x0)/b)^2],
   {{a, 300.}, {b, σαMinB}, {x0, μ0αMinB}}, x];(*x is α, y is ΔR *)
normalNLMαMinB = Normal[nlmαMinB];
showNLMαMinB = Show[{Plot[Normal[nlmαMinB], {x, μ0αMinB - 5 σαMinB, μ0αMinB + 5 σαMinB},
    PlotLabel → "α for Hmin", PlotRange → {0, 1.1 a /. nlmαMinB["BestFitParameters"]}],
   Histogram[sortHαMin, histogramRangeMin, PlotLabel → "α for Hmin"],
   Plot[Normal[nlmαMinB], {x, μ0αMinB - 5 σαMinB, μ0αMinB + 5 σαMinB},
    PlotLabel → "α for Hmin", PlotRange → {0, 700}] }]
parametersNLMαMinB  = {a, b, x0} /. nlmαMinB["BestFitParameters"];
pTableNLMαMinB  = nlmαMinB["ParameterTable"]
```

Out[216]=



Out[218]=

	Estimate	Standard Error	t-Statistic	P-Value
a	285.142	7.00863	40.6844	3.34638×10^{-22}
b	0.232561	0.00660051	35.2337	7.56707×10^{-21}
x0	1.74856	0.00660051	264.913	4.81651×10^{-40}

In[219]:=
```
{parametersNLMαMinB[[2]] , parametersNLMαMinB[[3]] }(*radians*)
{parametersNLMαMinB[[2]] (360./(2. π)), parametersNLMαMinB[[3]] (360./(2. π)) }(*degrees*)
```

Out[219]= {0.232561, 1.74856}

Out[220]= {13.3247, 100.185}

In[221]:=
```
{σαMinFit, αMinFit} = {parametersNLMαMinB[[2]] , parametersNLMαMinB[[3]] };(*radians*)
{parametersNLMαMinB[[2]] (360./(2. π)), parametersNLMαMinB[[3]] (360./(2. π)) };(*degrees*)
Print["(B) Therefore, the measured PPA ψ, including
   their uncertainties, produce a value of RA = α for H_min of  α  = ",
 αMinFit (360./(2. π)), "° ± ", σαMinFit (360./(2. π)), "° , according to the Gaussian fit."]
```

(B) Therefore, the measured PPA ψ, including
their uncertainties, produce a value of RA = α for H_{min} of α =
100.185° ± 13.3247° , according to the Gaussian fit.

In[224]:=
```
HminαAVE = (1/nR) Sum[Hminα[[i4]], {i4, nR}];(* average α for H_max in radians *)
Print["Also note that the average α for H_min in degrees is ",
 HminαAVE (360/(2. π)), "°, averaging over all runs."]
```

Also note that the average α for H_{min} in degrees is 101.214°, averaging over all runs.

Fit a Gaussian to the δ for H_{min}.

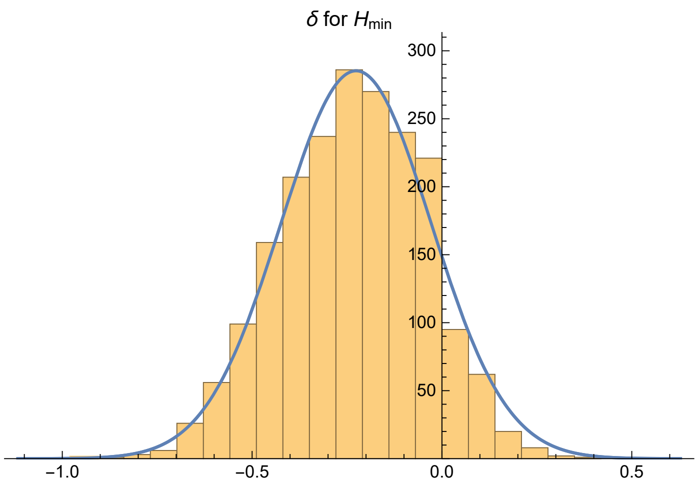
Definitions:

Replace “α” with “δ” in the quantities defined above for RA = α.

In[226]:=
```
sortHδMin = Sort[Table[Hminδ [[i2]], {i2, nR}]];
μ0δMinB = sortHδMin[[ Floor[nR/2] ]];
σδMinB = sortHδMin[[Floor[(4/5) Length[sortHδMin ]]]] - μ0δMinB;
histogramRangeMin = {μ0δMinB - 5 σδMinB, μ0δMinB + 5 σδMinB, 0.4 σδMinB};
histδMin = Histogram[sortHδMin, histogramRangeMin, PlotLabel → "δ for H_min"];
hl0Min = HistogramList[sortHδMin, histogramRangeMin];
{Length[ hl0Min[[1]] ], Length[ hl0Min[[2]] ]};
hlMin = Table[{(1/2) (hl0Min[[1, i1]] + hl0Min[[1, i1 + 1]]), hl0Min[[2, i1]]},
   {i1, Length[ hl0Min[[2]] ]}];
lphlδMin = ListPlot[hlMin, PlotLabel → "δ for H_min"];
Show[{histδMin, lphlδMin}];
```

```
nlmδMinB = NonlinearModelFit[hlMin, a Exp[-1/2. ((x - x0)/b)^2],
   {{a, 300.}, {b, σδMinB}, {x0, μ0δMinB}}, x];(*x is δ*)
normalNLMδMinB = Normal[nlmδMinB];
showNLMδMinB = Show[{Plot[Normal[nlmδMinB], {x, μ0δMinB - 5 σδMinB, μ0δMinB + 5 σδMinB},
    PlotLabel → "δ for Hmin", PlotRange → {0, 1.1 a /. nlmδMinB["BestFitParameters"]}],
   Histogram[sortHδMin, histogramRangeMin, PlotLabel → "δ for Hmin"],
   Plot[Normal[nlmδMinB], {x, μ0δMinB - 5 σδMinB, μ0δMinB + 5 σδMinB},
    PlotLabel → "δ for Hmin", PlotRange → {0, 700}] }]
parametersNLMδMinB = {a, b, x0} /. nlmδMinB["BestFitParameters"];
pTableNLMδMinB = nlmδMinB["ParameterTable"]
```

Out[238]=



Out[240]=

	Estimate	Standard Error	t-Statistic	P-Value
a	285.247	6.97035	40.9229	2.9474×10^{-22}
b	0.19792	0.00558461	35.4402	6.66837×10^{-21}
x0	−0.226146	0.0055846	−40.4945	3.70441×10^{-22}

In[241]:=
```
{σδMinFit, δMinFit} = {parametersNLMδMinB[[2]] , parametersNLMδMinB[[3]] };(*radians*)
{parametersNLMδMinB[[2]] (360./(2. π)), parametersNLMδMinB[[3]] (360./(2. π)) };(*degrees*)
Print["(B) Therefore, the measured PPA ψ, including
   their uncertainties, produce a value of dec = δ for Hmin of  δ  = ",
 δMinFit (360./(2. π)), "° ± ", σδMinFit (360./(2. π)), "° , according to the Gaussian fit."]
```

(B) Therefore, the measured PPA ψ, including
their uncertainties, produce a value of dec = δ for H_{min} of δ =
−12.9572° ± 11.34° , according to the Gaussian fit.

In[244]:=
```
HminδAVE = 1/nR Sum[Hminδ[[i4]], {i4, nR}];(* average δ for Hmax in radians *)
Print["While the average δ for Hmin in degrees is ",
 HminδAVE (360/(2. π)), "°, averaging over all runs."]
```

While the average δ for H_{min} in degrees is −14.431°, averaging over all runs.

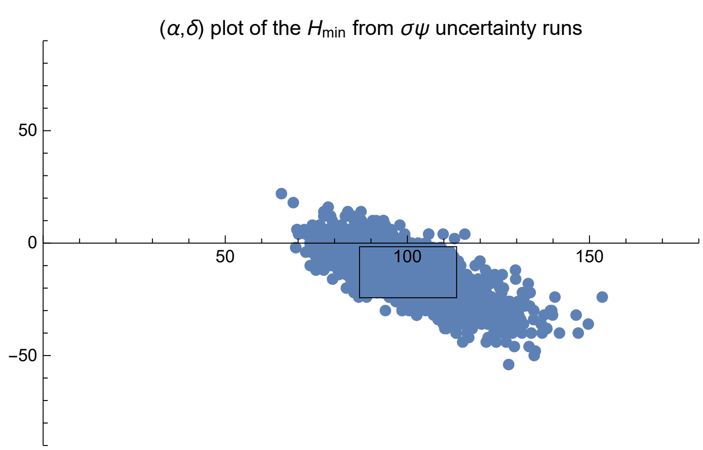
In[246]:=
```
Print["(B) Therefore, the measured PPA ψ, including their
   uncertainties, produce a value of (RA,dec) = (α,δ) for Hmin of  (α, δ)  = ",
 {αMinFit (360./(2. π)), δMinFit (360./(2. π))}, " ± ", {σαMinFit (360./(2. π)), σδMinFit (360./(2. π))},
 ", in degrees , according to the Gaussian fit."]
```

(B) Therefore, the measured PPA ψ, including their
   uncertainties, produce a value of (RA,dec) = (α,δ) for H_{min} of (α, δ) =
{100.185, -12.9572} ± {13.3247, 11.34}, in degrees , according to the Gaussian fit.

In[346]:=
```
Print["The best values ψn produce an alignment hub Hmin with (RA,dec) = ",
 {αHminDegrees, δHminDegrees}]
```

The best values ψn produce an alignment hub H_{min} with (RA,dec) = {106.408, -20.}

In[247]:=
```
(*C. Plot and check the value for Hmin in (B). *)
Hminαδ = Sort[Table[{Hminα[[i5]], Hminδ[[i5]]}, {i5, nR}]];
{Hminαδ[[1]], Hminαδ[[-1]]};(*radians*)
{Hminαδ[[1]], Hminαδ[[-1]]} (360./(2. π));(*degrees*)
lpHmin =
  ListPlot[Hminαδ (360./(2. π)), PlotRange → {{0, 180}, {-90, 90}}, PlotMarkers → Automatic,
   PlotLabel → "(α,δ) plot of the Hmin from σψ uncertainty runs"];
α1Min = (αMinFit - σαMinFit) (360./(2. π));
α2Min = (αMinFit + σαMinFit) (360./(2. π));
δ1Min = (δMinFit - σδMinFit) (360./(2. π));
δ2Min = (δMinFit + σδMinFit) (360./(2. π));
Show[{lpHmin, Graphics[Line[
    {{α1Min, δ1Min}, {α1Min, δ2Min}, {α2Min, δ2Min}, {α2Min, δ1Min}, {α1Min, δ1Min}}]]}]
```

Out[255]=


12. Uncertainty in the locations of the avoidance hubs H_{max}

Find the likelihood of the location (α,δ) of H_{max} in the runs made with ψ allowed to take values based on the uncertainty $\sigma\psi$ in the measurements.

The comments at the start of Sec. 11 apply here with obvious modifications.

A. By the symmetry across a diameter, a hub H_{max} at one location implies a second hub $-H_{max}$ exists at the diametrically opposite location. Thus we can choose one hub and move the opposite hubs.
B. Once we have collected the αs near $\alpha = 0$, we can see which values are the most popular. We can find a Gaussian fit to histograms of α and δ or judge the results by eye when the grid spacing is involved.
C. Finally we can plot the (α,δ) for the H_{max} and decide whether the assigned values look okay.

Definitions

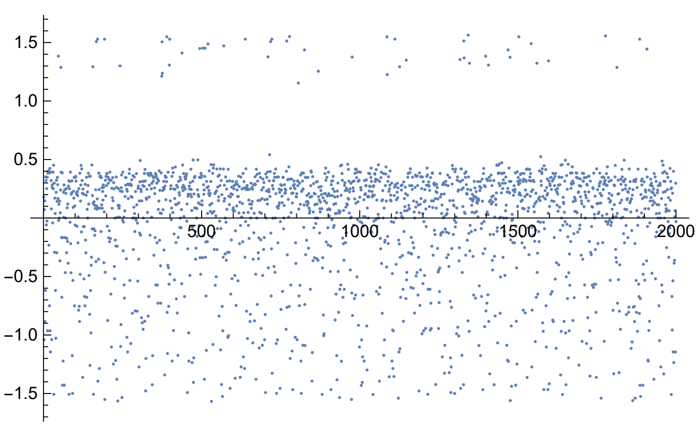
See the definitions for H_{min} at the start of Sec. 11.

The table information is placed here for reference.

runData entries: 1. Run # 2. ψSrc, list of polarization position angles ψ 3. $\{\overline{\eta}_{min}, \{\alpha,\delta\}$ at $H_{min}\}$ 4. $\{\overline{\eta}_{max}, \{\alpha,\delta\}$ at $H_{max}\}$

In[256]:=
```
(*A. Move hubs, if necessary, so that -180° ≤ α < 180° *)
Hmaxα0 = Table[ runData[[i1, 4, 2, 1]] , {i1, Length[runData ]}];
{Min[Hmaxα0], Max[Hmaxα0]};
ListPlot[Hmaxα0];
Hmaxδ0 = Table[runData[[i1, 4, 2, 2]], {i1, Length[runData ]}];
HmaxαBy180n = Round[Hmaxα0 / π];
Hmaxα1 = Table[Hmaxα0[[i1]] - HmaxαBy180n[[i1]] π , {i1, Length[runData]}];
Hmaxδ1 = Table[ (-1)^HmaxαBy180n[[i1]] Hmaxδ0[[i1]] , {i1, Length[runData]}];
{Min[Hmaxα1], Max[Hmaxα1]};
{Min[Hmaxδ1], Max[Hmaxδ1]};
ListPlot[Hmaxα1]
ListPlot[Hmaxδ1];
```

Out[264]=

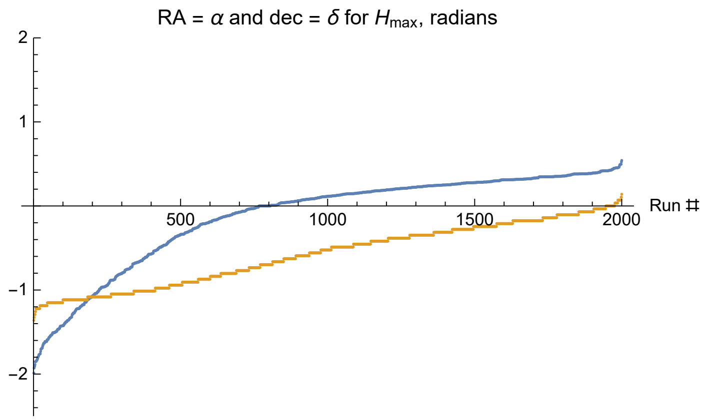


The band at RA $= \alpha = \pi/2$ is not very dense. By the symmetry across a diameter, we can move those hubs below $\alpha = -\pi/2$. The move across a diameter changes the sign of dec $= \delta$.

In[266]:=

```
Hmaxα = Table[If[Hmaxα1[[i1]] > 0.7, Hmaxα1[[i1]] - π, Hmaxα1[[i1]], "huh?"] ,
   {i1, Length[runData ]}];
Hmaxδ = Table[If[Hmaxα1[[i1]] > 0.7, -Hmaxδ1[[i1]], Hmaxδ1[[i1]], "huh?"] ,
   {i1, Length[runData ]}];
ListPlot[{Sort[Hmaxα], Sort[Hmaxδ]}, PlotRange → {-2.5, 2.0},
 PlotLabel → "RA = α and dec = δ for Hmax, radians", AxesLabel → {"Run #"}]
```

Out[268]=



(A.) The hubs found in the runs occupy a wide range in α, $-2 < \alpha < 0.8$. The decs δ occupy the range from -1.5 to 0.2 radians. Both ranges span intervals much larger than the grid spacing. So fitting Gaussians seems appropriate

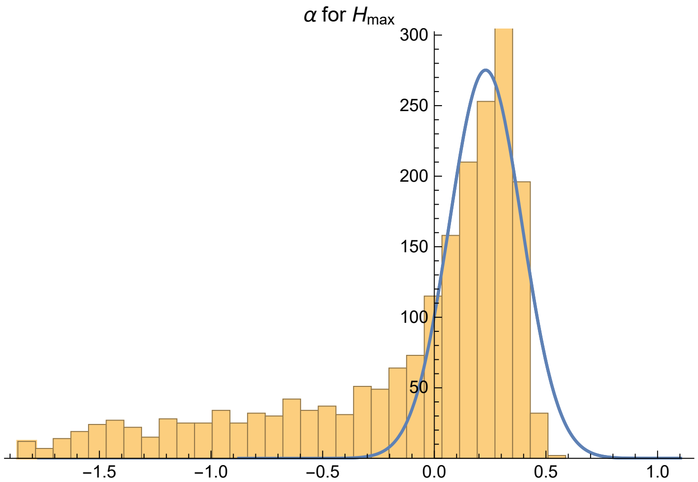
(B.) Fit a Gaussian to the α for H_{max}.

In[269]:=

```
sortHαMax = Sort[Table[Hmaxα [[i2]], {i2, nR}]];
μ0αMaxB = sortHαMax[[ Floor[nR/2] ]];
σαMaxB = sortHαMax[[Floor[4/5 Length[sortHαMax ]]]] - μ0αMaxB;
histogramRangeMax = {μ0αMaxB - 10 σαMaxB, μ0αMaxB + 5 σαMaxB, 0.4 σαMaxB};
histαMax = Histogram[sortHαMax, histogramRangeMax, PlotLabel → "α for Hmax"];
hl0Max = HistogramList[sortHαMax, histogramRangeMax];
{Length[ hl0Max[[1]] ], Length[ hl0Max[[2]] ]};
hlMax = Table[{(1/2) (hl0Max[[1, i1]] + hl0Max[[1, i1 + 1]]), hl0Max[[2, i1]]},
   {i1, Length[ hl0Max[[2]] ]}];
lphlαMax = ListPlot[hlMax, PlotLabel → "α for Hmax"];
Show[{histαMax, lphlαMax}];
```

In[279]:=
```
nlmαMaxB = NonlinearModelFit[hlMax, a Exp[- 1/2. ((x - x0)/b)^2],
    {{a, nR / 6}, {b, σαMaxB}, {x0, μ0αMaxB}}, x];(*x is α*)
normalNLMαMaxB = Normal[nlmαMaxB];
showNLMαMaxB = Show[{Plot[Normal[nlmαMaxB], {x, μ0αMaxB - 10 σαMaxB, μ0αMaxB + 5 σαMaxB},
     PlotLabel → "α for Hmax", PlotRange → {0, 1.1 a /. nlmαMaxB["BestFitParameters"]}],
    Histogram[sortHαMax, histogramRangeMax, PlotLabel → "α for Hmax"],
    Plot[Normal[nlmαMaxB], {x, μ0αMaxB - 5 σαMaxB, μ0αMaxB + 5 σαMaxB},
     PlotLabel → "α for Hmax", PlotRange → {0, 700}] }]
parametersNLMαMaxB = {a, b, x0} /. nlmαMaxB["BestFitParameters"];
pTableNLMαMaxB = nlmαMaxB["ParameterTable"]
```

Out[281]=



Out[283]=

	Estimate	Standard Error	t-Statistic	P-Value
a	275.184	20.4732	13.4412	3.65911×10^{-15}
b	0.162092	0.0139249	11.6404	2.07933×10^{-13}
x0	0.229921	0.0139249	16.5114	8.31351×10^{-18}

In[284]:=
```
{σαMaxFit, αMaxFit} = {parametersNLMαMaxB[[2]] , parametersNLMαMaxB[[3]] };(*radians*)
{parametersNLMαMaxB[[2]] (360./(2. π)), parametersNLMαMaxB[[3]] (360./(2. π)) };(*degrees*)
Print["(B) Therefore, the measured PPA ψ, including
   their uncertainties, produce a value of RA = α for Hmax of  α  = ",
 αMaxFit (360./(2. π)), "° ± ", σαMaxFit (360./(2. π)), "° , according to the Gaussian fit."]
```

(B) Therefore, the measured PPA ψ, including
their uncertainties, produce a value of RA = α for H_{max} of α =
13.1735° ± 9.2872° , according to the Gaussian fit.

In[287]:=
```
HmaxαAVE = 1/nR Sum[Hmaxα[[i4]], {i4, nR}];(* average α for Hmax in radians *)
Print["Also note that the average α for Hmax in degrees is ",
 HmaxαAVE (360/(2. π)), "°, averaging over all runs."]
```

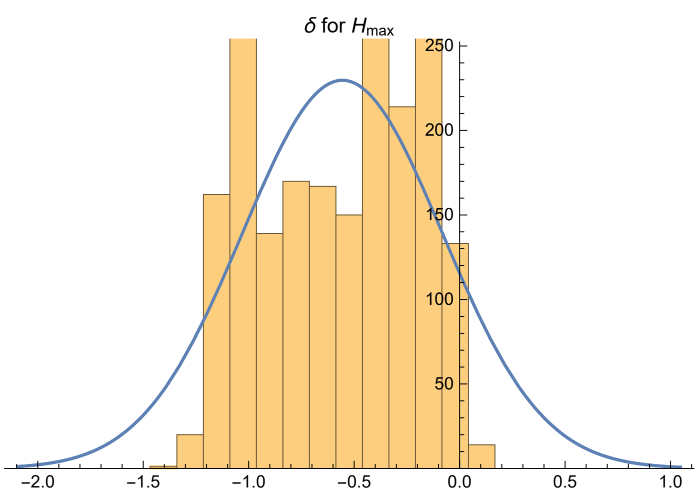
Also note that the average α for H_{max} in degrees is −6.952°, averaging over all runs.

Fit a Gaussian to the δ for $H_{\max}$.

In[289]:=
```
sortHδMax = Sort[Table[Hmaxδ [[i2]], {i2, nR}]];
μ0δMaxB = sortHδMax[[ Floor[nR/2] ]];
σδMaxB = sortHδMax[[Floor[4/5 Length[sortHδMax ]]]] - μ0δMaxB;
histogramRangeMax = {μ0δMaxB - 3 σδMaxB, μ0δMaxB + 3 σδMaxB, 0.4 σδMaxB};
histδMax = Histogram[sortHδMax, histogramRangeMax, PlotLabel → "δ for Hmax"];
hl0Max = HistogramList[sortHδMax, histogramRangeMax];
{Length[ hl0Max[[1]] ], Length[ hl0Max[[2]] ]};
hlMax = Table[{(1/2) (hl0Max[[1, i1]] + hl0Max[[1, i1 + 1]]), hl0Max[[2, i1]]},
   {i1, Length[ hl0Max[[2]] ]}];
lphlδMax = ListPlot[hlMax, PlotLabel → "δ for Hmax"];
Show[{histδMax, lphlδMax}];
```

In[299]:=
```
nlmδMaxB = NonlinearModelFit[hlMax, a Exp[- 1/2. ((x - x0)/b)^2],
   {{a, nR / 6}, {b, σδMaxB}, {x0, μ0δMaxB}}, x];(*x is δ, y is ΔR *)
normalNLMδMaxB = Normal[nlmδMaxB];
showNLMδMaxB = Show[{Plot[Normal[nlmδMaxB], {x, μ0δMaxB - 5 σδMaxB, μ0δMaxB + 5 σδMaxB},
    PlotLabel → "δ for Hmax", PlotRange → {0, 1.1 a /. nlmδMaxB["BestFitParameters"]}],
   Histogram[sortHδMax, histogramRangeMax, PlotLabel → "δ for Hmax"],
   Plot[Normal[nlmδMaxB], {x, μ0δMaxB - 5 σδMaxB, μ0δMaxB + 5 σδMaxB},
    PlotLabel → "δ for Hmax", PlotRange → {0, 700}] }]
parametersNLMδMaxB = {a, b, x0} /. nlmδMaxB["BestFitParameters"];
pTableNLMδMaxB = nlmδMaxB["ParameterTable"]
```

Out[301]=



Out[303]=

	Estimate	Standard Error	t-Statistic	P-Value
a	229.688	36.4783	6.29658	0.0000396791
b	0.473789	0.0946163	5.00748	0.000305462
x0	−0.555688	0.0868996	−6.39459	0.0000342979

In[304]:=
```
{σδMaxFit, δMaxFit} = {parametersNLMδMaxB[[2]] , parametersNLMδMaxB[[3]] };(*radians*)
{parametersNLMδMaxB[[2]] (360./(2. π)), parametersNLMδMaxB[[3]] (360./(2. π)) };(*degrees*)
Print["(B) Therefore, the measured PPA ψ, including
   their uncertainties, produce a value of dec = δ for H_max of  δ  = ",
 δMaxFit (360./(2. π)), "° ± ", σδMaxFit (360./(2. π)), "° , according to the Gaussian fit." ]
```

(B) Therefore, the measured PPA ψ, including
their uncertainties, produce a value of dec = δ for H_{max} of δ =
−31.8386° ± 27.1461° , according to the Gaussian fit.

In[307]:=
```
HmaxδAVE = 1/nR Sum[Hmaxδ[[i4]], {i4, nR}];(* average δ for H_max in radians *)
Print["Also note that the average δ for H_max in degrees is ",
 HmaxδAVE (360/(2. π)), "°, averaging over all runs."]
```

Also note that the average δ for H_{max} in degrees is −33.177°, averaging over all runs.

In[309]:=
```
Print["(B) Therefore, the measured PPA ψ, including their
   uncertainties, produce a value of (RA,dec) = (α,δ) for H_max of  (α, δ)  = ",
 {αMaxFit (360./(2. π)), δMaxFit (360./(2. π))}, " ± ", {σαMaxFit (360./(2. π)), σδMaxFit (360./(2. π))},
 ", in degrees , according to the Gaussian fit." ]
```

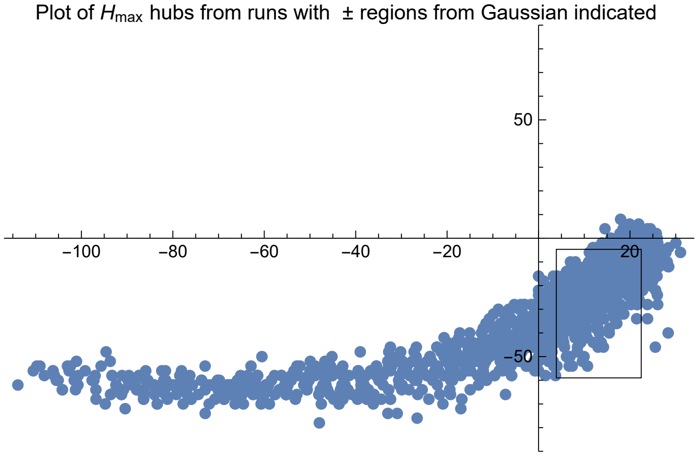
(B) Therefore, the measured PPA ψ, including their
uncertainties, produce a value of (RA,dec) = (α,δ) for H_{max} of (α, δ) =
{13.1735, −31.8386} ± {9.2872, 27.1461}, in degrees , according to the Gaussian fit.

In[347]:=
```
Print["The best values ψn produce an avoidance hub H_max with (RA,dec) = ",
 {αHmaxDegrees, δHmaxDegrees}]
```

The best values ψn produce an avoidance hub H_{max} with (RA,dec) = {9.93072, −22.}

In[310]:=
```
(*C. Plot and decide on a value for Hmax. *)
Hmaxαδ = Table[{Hmaxα[[i8]], Hmaxδ[[i8]]}, {i8, nR}];
{Hmaxαδ[[1]], Hmaxαδ[[-1]]};(*radians*)
{Hmaxαδ[[1]], Hmaxαδ[[-1]]} (360./(2. π));(*degrees*)
lpHmax1 = ListPlot[Hmaxαδ (360./(2. π)), PlotRange → {-90, 90}, PlotMarkers → Automatic,
   PlotLabel → "Plot of Hmax hubs from runs with  ± regions from Gaussian indicated "];
α1Max = (αMaxFit - σαMaxFit) (360./(2. π));
α2Max = (αMaxFit + σαMaxFit) (360./(2. π));
δ1Max = (δMaxFit - σδMaxFit) (360./(2. π));
δ2Max = (δMaxFit + σδMaxFit) (360./(2. π));
Show[{lpHmax1, Graphics[Line[
    {{α1Max, δ1Max}, {α1Max, δ2Max}, {α2Max, δ2Max}, {α2Max, δ1Max}, {α1Max, δ1Max}}]]}]
```

Out[318]=



That looks like it would benefit from putting a tail on the Gaussian.

In[319]:=
```
(*The Aitoff coordinates for the hubs Hmin locations.*)
xyAitoffHmin = Table[{xH[ Hminα [[n]] 360/(2 π), Hminδ[[n]] 360/(2 π) ],
    yH[ Hminα [[n]] 360/(2 π), Hminδ[[n]] 360/(2 π) ]}, {n, nR}];
(*The Aitoff coordinates for the hubs Hmax locations.*)
xyAitoffHmax = Table[{xH[ Hmaxα [[n]] 360/(2 π), Hmaxδ[[n]] 360/(2 π) ],
    yH[ Hmaxα [[n]] 360/(2 π), Hmaxδ[[n]] 360/(2 π) ]}, {n, nR}];
(*The Aitoff coordinates for the hubs -Hmin locations.*)
xyAitoffOppositeHmin = Table[{xH[ If[0 ≤ Hminα [[n]] 360/(2 π) < +180, Hminα [[n]] 360/(2 π) - 180,
      If[0 > Hminα [[n]] 360/(2 π) > -180, Hminα [[n]] 360/(2 π) + 180]], -Hminδ[[n]] 360/(2 π) ],
    yH[ If[0 ≤ Hminα [[n]] 360/(2 π) < +180, Hminα [[n]] 360/(2 π) - 180,
      If[0 > Hminα [[n]] 360/(2 π) > -180, Hminα [[n]] 360/(2 π) + 180]], -Hminδ[[n]] 360/(2 π) ]}, {n, nR}];
(*The Aitoff coordinates for the hubs -Hmax locations.*)
xyAitoffOppositeHmax = Table[{xH[ If[0 ≤ Hmaxα [[n]] 360/(2 π) < +180, Hmaxα [[n]] 360/(2 π) - 180,
      If[0 > Hmaxα [[n]] 360/(2 π) > -180, Hmaxα [[n]] 360/(2 π) + 180]], -Hmaxδ[[n]] 360/(2 π) ],
    yH[ If[0 ≤ Hmaxα [[n]] 360/(2 π) < +180, Hmaxα [[n]] 360/(2 π) - 180,
      If[0 > Hmaxα [[n]] 360/(2 π) > -180, Hmaxα [[n]] 360/(2 π) + 180]], -Hmaxδ[[n]] 360/(2 π) ]}, {n, nR}];
```

In[323]:=
```
probabilitiesForHmin = Table[nlmαMinB[ Hminα[[n]] ]/parametersNLMαMinB [[1]] nlmδMinB[ Hminδ[[n]] ]/parametersNLMδMinB [[1]], {n, nR}];
tableHc1 = Table[{xH[ Hminα [[i9]] 360/(2 π), Hminδ[[i9]] 360/(2 π) ],
    yH[ Hminα [[i9]] 360/(2 π), Hminδ[[i9]] 360/(2 π) ], probabilitiesForHmin[[i9]]}, {i9, 1, 2000}];
ListPlot3D[tableHc1, ColorFunction → "TemperatureMap"];
dpContourPlot = 0.3;
lcpTableHc1 = ListContourPlot[tableHc1,
   Contours → Table[p, {p, 0, 1, dpContourPlot}], ColorFunction → "TemperatureMap"];
```

```
In[328]:= probabilitiesForHmax = Table[nlmαMaxB[Hmaxα[[n]]]/parametersNLMαMaxB[[1]] nlmδMaxB[Hmaxδ[[n]]]/parametersNLMδMaxB[[1]], {n, nR}];
tableHc2 = Table[{xH[ Hmaxα [[i9]] 360/(2 π), Hmaxδ[[i9]] 360/(2 π) ],
    yH[ Hmaxα [[i9]] 360/(2 π), Hmaxδ[[i9]] 360/(2 π) ], probabilitiesForHmax[[i9]]}, {i9, 1, 2000}];
ListPlot3D[tableHc2, ColorFunction → "TemperatureMap"];
dpContourPlot = 0.3;
lcpTableHc2 = ListContourPlot[tableHc2,
   Contours → Table[p, {p, 0, 1, dpContourPlot}], ColorFunction → "TemperatureMap"];

In[372]:= (*Construct the map of Hmin and Hmax hubs with ± regions indicated.*)
Print["The map is centered on (RA,dec) = (0°,0°)."]
Print["The map is symmetric across diameters, i.e.
   diametrically opposite points -H and H have the same alignment angle."]
Print["The alignment hubs Hmin and -Hmin are plotted as light blue dots. ", LightBlue]
Print["The regions (α,δ) ± (σα,σδ) where the alignment hubs Hmin
   and -Hmin are most likely found are enclosed in purple lines. ", Purple]
Print["The avoidance hubs Hmax and -Hmax are plotted as pink dots. ", LightRed]
Print["The regions (α,δ) ± (σα,σδ) where the avoidance hubs Hmax
   and -Hmax are most likely found are enclosed in magenta lines. ", Magenta]
mapOfσψHminHmax =
 Show[
  {Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {δ, -90, 90}, PlotStyle → {Black, Thickness[0.002]},
     PlotPoints → 60, PlotRange → {{-7, 7}, {-3, 3}}, Axes → False], {α, -180, 180, 30}],
   Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {α, -180, 180},
     PlotStyle → {Black, Thickness[0.002]}, PlotPoints → 60], {δ, -60, 60, 30}], Graphics[
    {PointSize[0.007], Text[StyleForm["N", FontSize -> 10, FontWeight -> "Plain"], {0, 1.85}],
     LightBlue, (*Hmin:*)Point[ xyAitoffHmin ], (*-Hmin:*)Point[ xyAitoffOppositeHmin ],
     LightRed, (*Hmax:*)Point[ xyAitoffHmax ], (*-Hmax:*)Point[ xyAitoffOppositeHmax ],
      Black, Text[StyleForm["Max", FontSize → 8, FontWeight -> "Bold"],
      {xH[ - 180 , 0], yH[0, -60]}], {Arrow[BezierCurve[{{xH[ - 180 , 0], yH[0, -70]}, {-2.3, -2.0},
         {xH[αHmaxDegrees - 180, -δHmaxDegrees], yH[αHmaxDegrees - 180, -δHmaxDegrees]}}]]},
     Text[StyleForm["Min", FontSize → 8, FontWeight -> "Bold"], {xH[ 180 , 0], yH[0, -60]}],
     {Arrow[BezierCurve[{{xH[ 180 , 0], yH[0, -70]}, {2.3, -2.0},
         {xH[αHminDegrees, δHminDegrees], yH[αHminDegrees, δHminDegrees]}}]]},
     Text[StyleForm["Min", FontSize → 8, FontWeight -> "Bold"], {xH[ -180 , 0], yH[0, 60]}],
     {Arrow[BezierCurve[{{xH[ -180 , 0], yH[0, 70]}, {-2.3, 2.0},
         {xH[αHminDegrees - 180, -δHminDegrees], yH[αHminDegrees - 180, -δHminDegrees]}}]]},
     Text[StyleForm["Max", FontSize → 8, FontWeight -> "Bold"], {xH[ 180 , 0], yH[0, 60]}] ,
     {Arrow[BezierCurve[{{xH[ 180 , 0], yH[0, 70]}, {2.3, 2.0},
         {xH[αHmaxDegrees, δHmaxDegrees], yH[αHmaxDegrees, δHmaxDegrees]}}]]}      }],
   Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {δ, δ1Max, δ2Max},
     PlotStyle → {Magenta, Thickness[0.002]}, PlotPoints → 60], {α, α1Max, α2Max, α2Max - α1Max}],
   Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {α, α1Max, α2Max},
     PlotStyle → {Magenta, Thickness[0.002]}, PlotPoints → 60], {δ, δ1Max, δ2Max, δ2Max - δ1Max}],
   Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {δ, -δ2Max, -δ1Max},
```

```
      PlotStyle → {Magenta, Thickness[0.002]}, PlotPoints → 60],
     {α, α1Max - 180, α2Max - 180, α2Max - α1Max}],
    Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {α, α1Max - 180, α2Max - 180},
      PlotStyle → {Magenta, Thickness[0.002]}, PlotPoints → 60], {δ, -δ2Max, -δ1Max, δ2Max - δ1Max}],
    Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {δ, -δ2Min, -δ1Min},
      PlotStyle → {Purple, Thickness[0.002]}, PlotPoints → 60],
     {α, α1Min - 180, α2Min - 180, α2Min - α1Min}],
    Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {α, α1Min - 180, α2Min - 180},
      PlotStyle → {Purple, Thickness[0.002]}, PlotPoints → 60], {δ, -δ2Min, -δ1Min, δ2Min - δ1Min}],
    Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {δ, δ1Min, δ2Min},
      PlotStyle → {Purple, Thickness[0.002]}, PlotPoints → 60], {α, α1Min, α2Min, α2Min - α1Min}],
    Table[ParametricPlot[{xH[α, δ], yH[α, δ]}, {α, α1Min, α2Min},
      PlotStyle → {Purple, Thickness[0.002]}, PlotPoints → 60],
     {δ, δ1Min, δ2Min, δ2Min - δ1Min}]]}, ImageSize → 432]
Print["Caption: The hubs found when the polarization direction ψ  =  ψn
   + σψn  for each source is allowed to differ from the best value
   ψn by an amount chosen according to a Gaussian distribution with mean (best) value ψn and
   half-width σψn, both values ψn and σψn taken from the input in Sec. 3. There were ",
 nR, " runs with ψ = ψn + δψn. The resulting hubs are represented as lightly
   shaded dots, light blue for alignment and pink for avoidance. The arrows point
   to the hubs found with the best values of the polarization directions. "]
```

The map is centered on (RA,dec) = (0°,0°).

The map is symmetric across diameters, i.e.
diametrically opposite points -H and H have the same alignment angle.

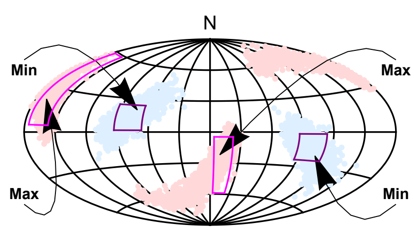
The alignment hubs H_{min} and $-H_{min}$ are plotted as light blue dots.

The regions (α,δ) ± (σα,σδ) where the alignment hubs
H_{min} and $-H_{min}$ are most likely found are enclosed in purple lines.

The avoidance hubs H_{max} and $-H_{max}$ are plotted as pink dots.

The regions (α,δ) ± (σα,σδ) where the avoidance hubs
H_{max} and $-H_{max}$ are most likely found are enclosed in magenta lines.

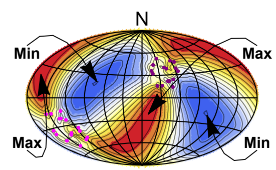
Out[378]=

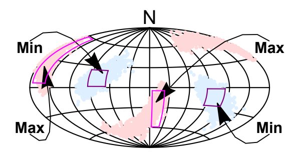


```
Caption: The hubs found when the polarization direction ψ = ψn + σψn for
   each source is allowed to differ from the best value ψn by an amount chosen
   according to a Gaussian distribution with mean (best) value ψn and half-width
   σψn, both values ψn and σψn taken from the input in Sec. 3. There were 2000
  runs with ψ = ψn + δψn. The resulting hubs are represented as lightly shaded
   dots, light blue for alignment and pink for avoidance. The arrows point
   to the hubs found with the best values of the polarization directions.
```

In[341]:= **GraphicsRow[{mapOfηBar, mapOfσψHminHmax}]**

Out[341]=





Maps in equatorial coordinates centered on (RA, dec) = (0°, 0°).

Left: The map from Sec. 7. The best values ψn of polarization directions produce this plot of $\overline{\eta}$(H), the alignment angle $\overline{\eta}$ as a function of location on the Celestial sphere.

Right: Plot of the hubs $H_{\min}$ (pale blue dots) and $H_{\max}$ (pink dots) found in the runs with polarization directions $\psi + \sigma\psi$ allowed to vary according to the uncertainties $\sigma\psi$.

References

0. R. Shurtleff, the ready-to-run Mathematica notebook is available, for a limited time, at the following URL: https://www.dropbox.com/s/ykb0ps6mybkprk3/20210110IntermediateKitForHubTest4b.nb?dl=0 (2021).
1. R. Shurtleff, "Indirect polarization alignment with points on the sky, the Hub Test" , https://vixra.org/abs/2011.0026 (2020).
2. R. Shurtleff, "Evaluating the Alignment of Astronomical Linear Polarization Data, introductory software" , https://vixra.org/ab-s/2101.0073 (2021).
3. Wolfram Research, Inc., Mathematica, Version 12.1, Champaign, IL (2020).
4. Wikipedia contributors. "Aitoff projection." Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 25 May. 2017. Web. 3 Jan. 2018.

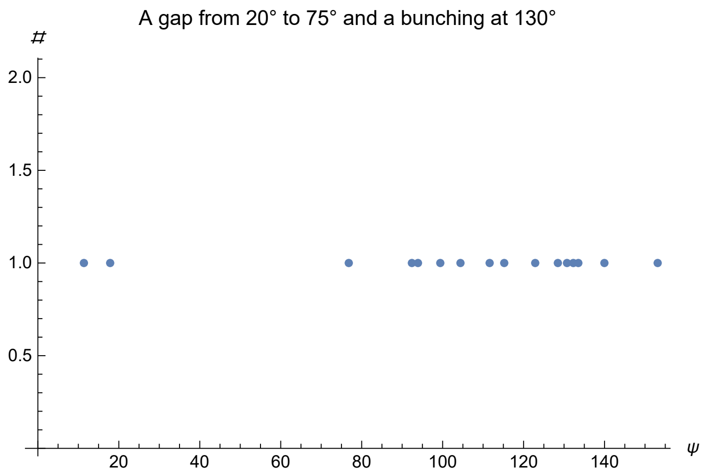
Appendix

The map of the alignment angle $\overline{\eta}$(H) above sections the sphere into four regions, two of alignment (blue) and two of avoidance (red). The hubs $H_{\min}$ and $H_{\max}$ are far from the sources. It follows that the polarization directions should align with the direction toward $H_{\min}$ and there should be few sources with polarization directions toward $H_{\max}$.

The following plot confirms those expectations. The plot shows the occupancy of polarization directions ψ . A dot is placed at # = 1 where the source has a PPA ψ. Clearly there is a bunching of sources near $\psi = 130°$, which is the angle from local North, i.e. South and East. A glance at the $\overline{\eta}$(H) map shows that $H_{\min}$ is southeast of the sources. Likewise the gap between $\psi = 20°$ to 75° corresponds to an avoidance of the northeast direction, i.e. the direction of $H_{\max}$ from the sources.

In[342]:=
```
ListPlot[Table[{ψn[[i]] (360./(2. π)), 1}, {i, nSrc}],
  PlotLabel → "A gap from 20° to 75° and a bunching at 130°", AxesLabel → {"ψ", "#"}]
Print["Caption: The gap and bunching of the polarization directions
   corresponds to the directions from the sources to areas of
   divergence and conversion, respectively, on the Celestial sphere."]
```

Out[342]=



Caption: The gap and bunching of the polarization directions corresponds to the directions from the sources to areas of divergence and conversion, respectively, on the Celestial sphere.