Evaluating the Alignment of the Polarized Radio Waves from 13 QSOs in Ursa Major

Richard Shurtleff *

Abstract

A sample of 13 quasi-stellar objects, QSOs, with polarized radio emissions and located in the Southern part of Ursa Major is shown by the Hub Test to have significantly aligned polarization directions. The QSOs are taken from the JVAS1450 subset of the JVAS/-CLASS 8.4-GHz surveys. The Hub Test evaluates alignment indirectly by extending the sources' polarization directions around the Celestial Sphere and quantifying the degree of convergence of these geodesics, *i.e.* great circles, at points on the Celestial Sphere. The hub of best convergence is found to be close to the sources. About one in 50,000 randomly directed samples would be better aligned than the polarization directions of these 13 QSOs. Some underlying calculations are presented in a Mathematica-coded Appendix. Access to a ready-to-run version is provided.

Keywords: Polarized Radio Sources; Alignment; Quasi-stellar objects

*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

0. Preface

The pdf version of this notebook is available online from the viXra archive.
To find the ready-to-run notebook follow the link in Ref. 1. The notebooks in this series were created using Wolfram Mathematica, Version Number: 12.1, Ref. 2.

Note(s):

(1) Random numbers should be reliable. Thus, numerical quantities in the pdf version should differ from the live ready-to-run version in Ref. 1. Different sets of random runs, for a sufficiently large number of runs, should provide numerical values that differ only slightly.

CONTENTS

0. Preface
Part I: Article
1. Introduction to Part I
2. Sample Selection and the Hub Test
3. The alignment of the polarization directions for the 13 QSOs
4. Experimental uncertainty
5. Significance
6. Conclusions
7. References

Part II: Appendix, Mathematica Computer Program
1. Introduction to Part II

2. Coordinates, utility functions, derivation of basic formula
    2a. Coordinates, utility functions
    2b. Derivation of a formula for the alignment angle η_{iH}
3. Polarization and Position Data
    3a. Source Data
    3b. Section Summary
4. Grid
    4a. Construct the grid
    4b. Section Summary
5. The alignment function $\overline{\eta}$(H) for the sample of sources
    5a. Determine the alignment angle $\overline{\eta}$(H)
    5b. Plot the Alignment Angle Function $\overline{\eta}$(H)
    5c. Section Summary
6. Uncertainty Runs
    6a. Creating and Storing Uncertainty Runs
    6b. The Effects of Uncertainty on the Smallest Alignment Angle $\overline{\eta}_{min}$
    6c. The Effects of Uncertainty on the Largest Avoidance Angle $\overline{\eta}_{max}$
    6d. The Effects of Uncertainty on the Locations (α,δ) of the Alignment Hubs H_{min}
    6e. The Effects of Uncertainty on the Locations (α,δ) of the Avoidance Hubs H_{max}
    6f. The Effects of Uncertainty on the angle between the planes of the Sample center to H_{min} Great Circle and the Sample to H_{max} Great Circle.
    6g. Map of the Hubs for the Uncertainty Runs
    6h. Section Summary
7. Probability and Significance
    7a. Probability and Significance Formulas
    7b. Generating random ψ runs
    7c. Analyzing random ψ runs
    7d. Significance of the alignment and avoidance Hub Test metrics for the sample studied in this work
    7e. Conclusion

In[1]:=

1. Introduction to Part I

Quasi-Stellar Objects, QSOs or quasars, can be polarized, making them candidates for studying correlations of polarization alignment. Large scale alignments are found for both optical and radio quasi stellar objects (QSOs), Refs. 3,4,5. In some studies, the tests that determine significant alignment compare the polarization direction of the electromagnetic radiation from one of the QSOs with one or more of its neighbors. An example of the potential value of such research is the finding of correlations between polarization directions and the local large scale structure, Refs. 6,7.

The Hub Test does not compare polarizations directly with each other, but indirectly, by finding points of convergence of the great circle geodesics obtained by extending polarization directions around the Celestial Sphere. Places where the geodesics are most dense are called “hubs” much as International Travel Hubs are places where the paths of passenger jets converge. Some other studies, Refs. 8,9, employ the Hub Test that is used here.

All tests, direct or indirect, serve to add to the information defining the behavior of QSOs. The tests inform Large Scale Structure, as noted above, as well as possibly intergalactic magnetic fields, Ref. 10, the properties of these objects, and other topics of interest.

2. Sample selection and the Hub Test

The sample of 13 QSOs in this report are taken from the JVAS1450, Ref. 11,12, a catalog of 1450 QSOs that was kindly communicated to me by one of the authors of Ref. 11. Details of the dataset can be found in Ref. 11. As explained in Ref. 11, the JVAS1450 catalog builds on data from the earlier large JVAS/CLASS 8.4-GHz catalog, Ref. 13.

To find candidate samples in the JVAS1450 to study, a survey was conducted. The QSO sources were binned, assigned to 5° radius circular regions centered on the grid points of a 2° mesh. A minimum of seven sources was enforced. The regions were sorted by the significance of their alignments according to the Hub Test. A previous report, Ref. 8, evaluated a clump of 27 QSOs, Clump 1 in Fig. 1, found in the overlap of eight of the 5° regions.

In this report we investigate a second clump, ‘Clump 2’, of QSOs inhabiting the overlap of three significantly aligned regions somewhat North of Clump1. The 13 QSOs are all the sources in the JVAS1450 catalog that have RA and dec in the ranges 161.86° < RA < 179.62° and 44.34° < dec < 53.60° and are located within 6.494° from the sample center at (RA,dec) = (171.445°, 48.678°) . The alignment of these 13 QSOs is evaluated with the Hub Test.

Out[∘]=

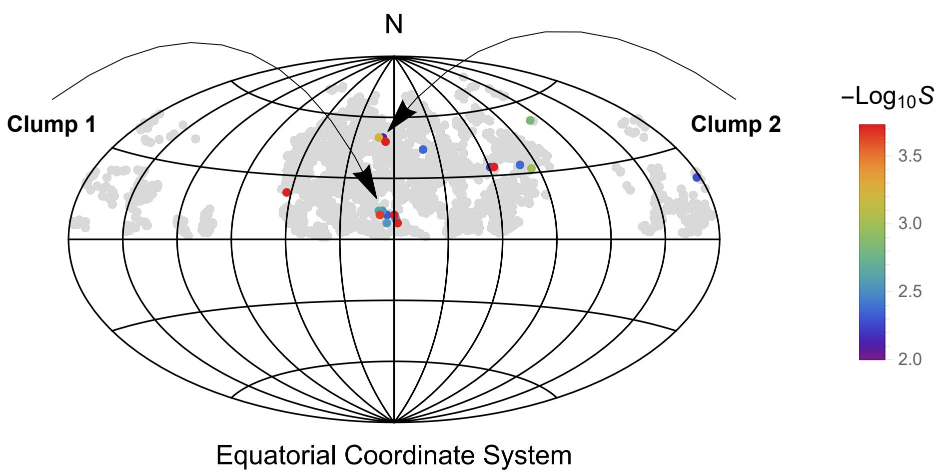


Figure 1. Survey of some polarized radio QSOs. (Equatorial Coordinates, centered at $(\alpha,\delta) = (180°,0°)$, East to the right.) The 1450 QSOs were grouped into 5° radius regions centered on grid points. Those regions having at least 7 QSOs are plotted as gray dots. Just 35 regions showed very significant alignment, *i.e.* $S \leq 0.01 = 10^{-2}$, or, equivalently, $-\text{Log}_{10} S \geq 2.0$, and these are shaded in color. Clump 1 has 14 regions containing 27 QSOs and is analyzed elsewhere, Ref. 8. Clump 2 has 3 regions containing 13 QSOs and is selected for analysis here. Clump 3 remains unidentified.

The Hub Test is discussed more fully in Ref. 14. The basic idea is analogous to a well-known prescription for finding Polaris, the North Star. Assume one can find the stars Merak and Dubhe which are two stars in the constellation Ursa Major. Then the direction from Merak to Dubhe aligns with the direction from Merak to Polaris. In analogy with Fig. 2, let the source S be the star Merak, take the direction from Merak to Dubhe to be the direction of polarization $\hat{v}_\psi$, and let Polaris be the point H. Then the alignment of the Merak-to-Dubhe direction $\hat{v}_\psi$ with the direction toward Polaris, the point H, illustrates the concept of alignment in the Hub Test. The alignment angle η would be about $\eta = 3.47°$ and the blue great circle would almost coincide with the purple great circle .

Out[ ]=

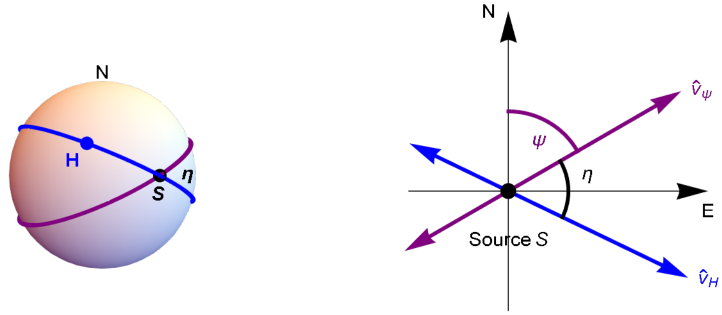


Figure 2: The Celestial sphere is pictured on the left and on the right is the plane tangent to the sphere at the source *S*. The linear polarization direction $\hat{v}_\psi$ lies in the tangent plane and determines the purple great circle on the sphere. A point *H* on the sphere together with the point *S* determine a second great circle, the blue circle drawn on the sphere. Clearly, *H* and *S* must be distinct in order to determine a great circle. The angle η measures the alignment of the polarization direction ψ with the point *H*.

In Fig. 2, the “alignment angle” η is the acute angle η between two great circles at *S*, $0° \leq \eta \leq 90°$. The alignment angle η measures how well the polarization direction $\hat{v}_\psi$ matches the direction $\hat{v}_H$ toward the point *H*. Perfect alignment occurs when $\eta = 0°$ and the two great circles overlap. Perpendicular great circles, $\eta = 90°$, indicates maximum “avoidance” of the polarization direction $\hat{v}_\psi$ with the point *H* on the sphere. The halfway value, $\eta = 45°$, favors neither alignment nor avoidance.

With *N* sources S_i, $i = 1, ..., N$, there are *N* alignment angles η_{iH} at each point *H* . One can calculate an average alignment angle $\overline{\eta}$ at *H*,

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

where

$$\cos(\eta_{\text{iH}}) = | \hat{v}_\psi . \hat{v}_H | . \qquad (2)$$

Each angle η_{iH} is taken to be the acute angle solving (2). Then the average alignment angle $\overline{\eta}(\text{H})$ at the point H must also be acute.

The alignment angle $\overline{\eta}(\text{H})$ is a function of position *H* on the sphere. It is symmetric across diameters, $\overline{\eta}(\text{H}) = \overline{\eta}(\text{-H})$, because great circles are symmetric across diameters. The function $\overline{\eta}(\text{H})$ measures convergence and divergence of the great circles determined by the polarization directions. For random polarization directions, the average $\overline{\eta}(\text{H})$ should be near 45°, since each alignment angle η_{iH} is acute, $0° \leq \eta_{\text{iH}} \leq 90°$, and random polarization directions should not favor any one value. Points *H* where the alignment angle $\overline{\eta}(\text{H})$ is smaller than 45°, the great circles tend to converge, where $\overline{\eta}(\text{H})$ is larger than 45°, the great circles can be said to diverge.

In this article and notebook, we often use “min” to label the smallest alignment angle $\overline{\eta}_{\text{min}}$ and the associated points on the sphere, the “hubs” H_{min} and $-H_{\text{min}}$. Thus “min” is associated with convergence of the polarization directions. For divergence, the hubs H_{max} and $-H_{\text{max}}$ locate places where the polarization directions avoid, as indicated by the largest alignment angle $\overline{\eta}_{\text{max}}$. Thus, we very often label an avoidance related quantity with “max”.

3. The alignment of the polarization directions for the 13 QSOs

For the 13 sources considered in this report, the alignment angle function $\overline{\eta}(\text{H})$ makes the following contour map. The global and local maps are computed in the Mathematica program below in Part II, Secs. 5b,c.

Out[ ]=

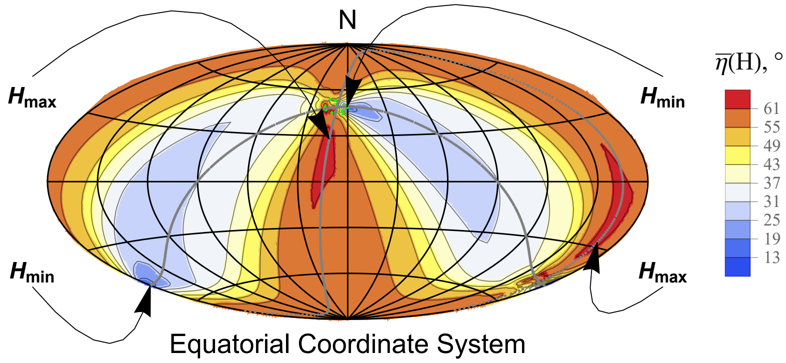


Figure 3: The alignment angle function $\overline{\eta}(H)$ mapped on the Celestial Sphere (Aitoff plot, centered on $(\alpha,\delta) = (180°,0)$, East to the right). The QSOs are shaded green ■. To guide the eye, two Great Circles are plotted in gray, one through the sources' center point and the avoidance hubs H_{max} and $-H_{max}$ while the other Great Circle runs through the sources' enter and the alignment hubs H_{min} and $-H_{min}$. The circles cross at an angle of 105°. The smallest alignment angle, $\overline{\eta}_{min} = 10.86°$, is located at the hubs H_{min} and $-H_{min}$, where the polarization directions converge best. One alignment hub H_{min} is located very close to the QSOs.

Out[ ]=

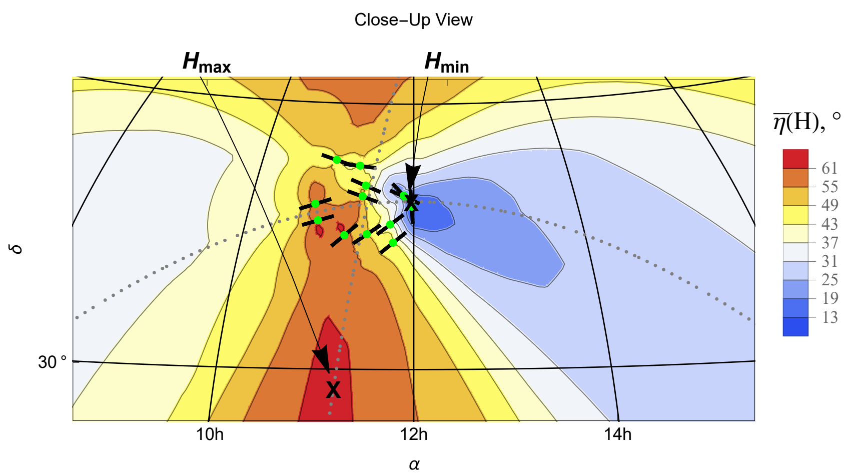


Figure 4: The region near the QSOs. The QSOs are located at the green dots. The short black lines through the QSOs indicate the polarization directions. Two of the QSOs are so close to the hub H_{min} that it is difficult to distinguish the "X" at the hub from the polarization direction markers. Measuring polarization directions ψ clockwise from North, one sees that the angles ψ range from above $\psi = 90°$ for the northern-most QSOs to 45° or so for the more southerly QSOs. The QSOs display parallax: all are in the general direction of the alignment hub H_{min}, but their directions depend on where they are located.

4. Experimental uncertainty

All experimental results include uncertainty. The maps above were drawn based on the values reported in the JVAS1450 catalog. The catalog also reports uncertainties in the polarization directions. In Part II Sec. 6, below, the uncertainties are carried through the calculations yielding the uncertainties in the results.

The uncertainties reported with the observed polarization directions are assumed to make normal distributions, *i.e.* Gaussians that integrate to unity. For example, one of the QSOs, the sixth one, has a measured polarization position angle of $\psi_{obs} \pm \sigma = 115.1° \pm 7.6°$. We take this to mean that the probability that the actual value of ψ was not $\psi_{obs} = 115.1°$, but some other value ψ_1, is given by the Gaussian

$$P(\psi_1) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[\frac{-1}{2}\left(\frac{\psi_1-\psi_{obs}}{\sigma}\right)^2\right]. \qquad (3)$$

The Mathematica software has a special command, “RandomVariate”, that produces random values of ψ_1 with respect to the probability distribution in Eq. (3). Thus, an “uncertainty run” begins by selecting a set of polarization directions for the 13 QSOs conforming to the uncertainty distributions like the one in Eq. (3). The alignment angle function $\overline{\eta}$(H) in Eq. (1) is evaluated to find the smallest alignment angle $\overline{\eta}_{\text{min}}$. As expected, the small changes to the observed polarization directions make small changes to the resulting angle $\overline{\eta}_{\text{min}}$. By repeating the process many times, one obtains a distribution of values for the smallest alignment angle $\overline{\eta}_{\text{min}}$.

The many uncertainty run values for the smallest alignment angle $\overline{\eta}_{\text{min}}$ produce a distribution of the smallest alignment angle $\overline{\eta}_{\text{min}}$, as well as the locations of alignment hubs These distributions have corresponding mean values and distribution widths. See Fig. 5. The distribution of the uncertainty run values for the smallest alignment angle $\overline{\eta}_{\text{min}}$ in Fig. 5 can be summarized by $\overline{\eta}_{\text{min}} = 11.39° \pm 1.07°$. As noted previously, the recorded polarization directions ψ_{obs}, the “best” values of ψ, give the observed value, $\overline{\eta}_{\text{min}} = 10.86°$, and that value is in the range, $\overline{\eta}_{\text{min}} = 11.39° \pm 1.07°$, determined by experimental uncertainty.

Out[∘]=

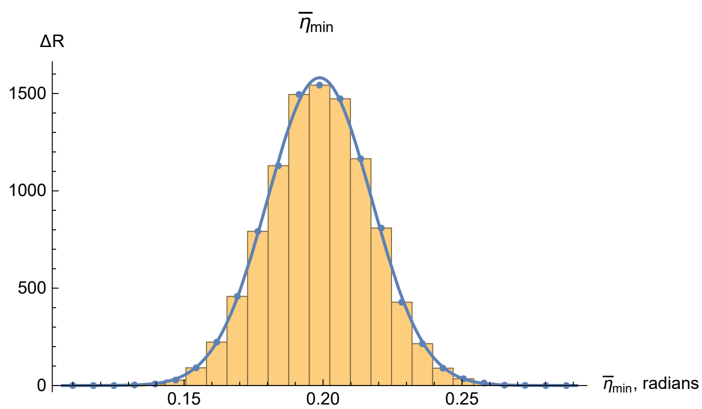


Figure 5: Histogram of the smallest alignment angle $\overline{\eta}_{\text{min}}$ for R = 10,000 uncertainty runs. The height ΔR is the number of uncertainty runs with a value of $\overline{\eta}_{\text{min}}$ in the ‘bin’, the range covered by each bar. This Gaussian distribution peaks at a mean value of $\overline{\eta}_{\text{min}}$ of 0.1988 radians = 11.39° and has a half-width of $\sigma = 0.0187055 = 1.07°$ where the distribution is down from the peak by a fraction $e^{-1/2} = 0.607 = 60.7\%$. One writes the result as $\overline{\eta}_{\text{min}} = 0.1988 \pm 0.0187$ radians $= 11.39° \pm 1.07°$.

Besides the uncertainty in the smallest alignment angle $\overline{\eta}_{\text{min}}$, the uncertainty runs yield uncertainty ranges for other quantities such as the largest avoidance angle $\overline{\eta}_{\text{max}}$. Each uncertainty run has its own set of alignment and avoidance hubs, H_{min} and H_{max}, respectively. A plot of the polarization directions with their uncertainties and the locations of the uncertainty run hubs is displayed in Fig. 6.

Out[ ]=

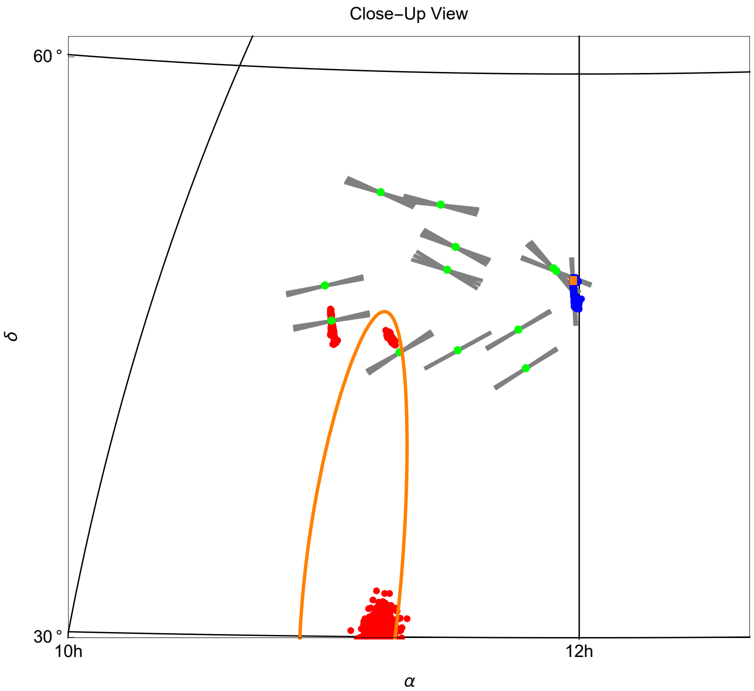


Figure 6: The QSOs as green dots plotted with the experimental uncertainties in polarization directions and hub locations from 10,000 uncertainty runs. Not all avoidance hubs H_{max}, red dots, are displayed since many are outside the region plotted. The uncertainty in the location of avoidance hubs, represented by the partial orange oval, is huge compared to the little orange dot representing the uncertainty in the location of the alignment hubs. All 10,000 alignment hubs H_{min}, blue dots, are displayed. In the following section, we find that the avoidance of the hubs H_{max} is much like what random directions would produce, while the alignment of the polarization directions with the hubs H_{min} is very significant.

5. Significance

Finally, we need to determine the significance of the alignment found for the polarization directions of these 13 QSOs. ‘Significance’ means how likely it is that randomly directed polarization vectors would give the same or better alignments than the observed polarization directions give.

To determine significance, we repeatedly find the smallest alignment angle function $\overline{\eta}(\text{H})$ many times, but with random ψ for the 13 QSOs. The process is similar to the process that determines uncertainties in the previous section. Instead of experimental values of ψ_{obs}, one substitutes random ψ for the 13 QSOs. The only experimental data used in this process is the location of the 13 QSO sources. The goal is to see what fraction of random runs yield a value with a lower $\overline{\eta}_{\text{min}}$ than the value $\overline{\eta}_{\text{min}} = 10.86°$ obtained with the observed data.

Below, we deal with 10,000 random runs. By sorting those 10,000 runs by the value of $\overline{\eta}_{\text{min}}$, smaller $\overline{\eta}_{\text{min}}$ before larger $\overline{\eta}_{\text{min}}$, one can find how many of those 10,000 runs gives a smaller alignment angle $\overline{\eta}_{\text{min}}$ than the observed value of $\overline{\eta}_{\text{min}}$, *i.e.* $\overline{\eta}_{\text{min}} = 10.86°$ using the recorded polarization directions ψ_{obs} from the catalog. One and only one of the 10,000 runs is better. So the significance of $\overline{\eta}_{\text{min}} = 10.86°$ is about one in 10,000 or 0.0001, more or less. Clearly, we would need many more sets of 10,000 random runs for such considerations to produce a value of significance that we could assign a plus/minus, an uncertainty.

Rather than expending a large amount of computer time generating more random runs, we follow conventional practice and make do with the 10,000 random runs. We start by finding a function that fits the distribution of the 10,000 $\overline{\eta}_{\text{min}}$, one smallest alignment angle $\overline{\eta}_{\text{min}}$ per random run. Having found a function that fits the distribution, we make the assumption that the function accurately describes the distribution far down on the “tail” of the function where our well-aligned QSOs have their $\overline{\eta}_{\rm min}$.

A histogram of the resulting smallest alignment angles $\overline{\eta}_{\rm min}$ from 10,000 runs is displayed in Fig. 7. Look closely at the distribution in Fig. 7. The right side, the side toward $\overline{\eta}_{\rm min} \to \pi/4 \sim 0.79$, has a steeper slope than the left side, the side toward $\overline{\eta}_{\rm min} \to 0$. Thus, the low $\overline{\eta}_{\rm min}$ side is favored; probability is pushed from the right side to the left side. A simple, symmetrical Gaussian would not fit the data well. The fitting curve shown combines a Gaussian with a unit step-function, that is unity to the left, and zero to the right, of the peak. Since the 13 QSOs have an alignment angle $\overline{\eta}_{\rm min}$ that is about 0.2 radians, it occurs far down the tail of the curve on the side where the step-function is unity and the curve is a Gaussian.

It is important for the application here to notice that the step-function is unity along the tail of the distribution on the left, $\overline{\eta}_{\rm min} \to 0$, side. The well-aligned sample of 13 QSOs has a smallest alignment angle around $\overline{\eta}_{\rm min} = 0.2$ radians, which is far down the tail, see the blue arrow in Fig. 7. The net effect of the steep right side of the distribution is to raise the probability of the observed $\overline{\eta}_{\rm min} = 0.2$ radians result by about 20%. Since random runs are thereby more likely in the region of the observed result, that makes the observed result somewhat less significant than if the distribution were symmetric.

Out[ ]=

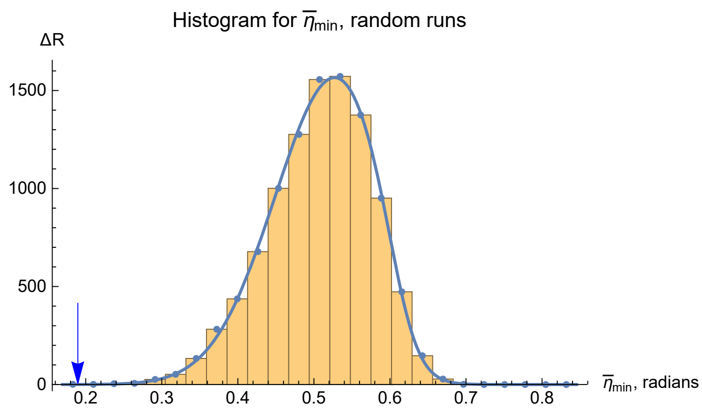


Figure 7. The distribution of the smallest alignment angle $\overline{\eta}_{\rm min}$ for R = 10,000 random runs. Each run assigns a random polarization direction to each of the 13 QSOs. The height ΔR is the number of runs with $\overline{\eta}_{\rm min}$ in the designated range of each bin. The fraction ΔR/R represents the likelihood that a random run result $\overline{\eta}_{\rm min}$ is in the bin. Thus the histogram approximates the shape of the probability distribution, aside from a normalizing scale factor. The observed polarization directions determine a value of $\overline{\eta}_{\rm min}$ at the blue arrow far down the tail.

To find the significance of the observed smallest alignment angle $\overline{\eta}_{\rm min} = 10.86°$, we integrate the probability distribution to find the likelihood that a random run would produce a smaller value. The significance is found to be 1.99 (30) x10^{-5}or about one in fifty thousand random runs would be better aligned than is experimentally observed for these QSOs. The alignment of the polarization directions with the hub $H_{\rm min}$ is, therefore, very significant.

6. Conclusions

The polarization directions of these 13 QSOs are well-aligned with a point on the Celestial Sphere, the hub $H_{\rm min}$, that is very close to the sample. Finding a correlation among polarization directions that display parallax is a property that distinguishes the Hub Test from other tests. Thus, the 13 QSOs offer a satisfying illustration of the Hub Test.

It is unlikely that the alignment is a consequence of selection bias. These 13 QSOs, Clump 2 in Fig. 1, are not alone; a sample of 27 QSOs, Clump 1, has been evaluated by the Hub Test. Clump 1 is better aligned than one in 80,000 random runs, while Clump 2 is better aligned than one in 50,000 random runs. Since the survey of 5°-radius regions, Fig. 1, involves 1863 regions, it seems that the alignments are not due to selection bias. And the survey finds other locations with significant alignment, so there may be a Clump 3 waiting to be investigated.

While the article, Ref. 15, relating alignments to Large Scale Structure constrains the QSOs to have like-redshifts, one might argue that the alignment found in this article is due to a subset of the 13 QSOs with more-or-less equal redshifts. Then the alignment would speak to Large Scale Structures, as in Ref. 15.

Astronomical data is being acquired at fantastic rates, so there may be new catalogs of many more QSOs with linear polarization directions to analyze. Such an investigation would be intriguing.

The main motivation for this study is to illustrate an application of the Hub Test. Interpreting the results is deemed beyond the scope of this study, which is intended to be a simple application of a test of alignment. One hopes the results are of interest and potentially useful.

7. References

1. Shurtleff, R., the ready-to-run Mathematica version of this notebook is available at the following URL:
https://www.dropbox.com/s/6zr8aneysic1zlv/20211009Clump2PaperFirst.nb?dl=0
2. Wolfram Research, Inc., Mathematica, Version 12.1, Champaign, IL (2020).
3. Hutsemékers, D., “Evidence for very large-scale coherent orientations of quasar polarization vectors”, Astronomy and Astrophysics 332, pp. 410-428 (1998)
4. Jain, P., Narain, G., and Sarala, S., “Large-scale alignment of optical polarizations from distant QSOs using coordinate-invariant statistics”, MNRAS 347, pp. 394-402 (2004).
5. Pelgrims, V. & Cudell, J. R., “A new analysis of quasar polarization alignments”, MNRAS 442, pp. 1239-1248 (2014).
6. Hutsemékers, D., Braibant, L., Pelgrims, V., and Sluse, D., Alignment of quasar polarizations with large-scale structures, Astron. Astrophys., 572, A18, doi: 10.1051/0004-6361/201424631, arXiv:astro-ph:1409.6098 (2014).
7. Pelgrims, V. & Hutsemékers, D., “Evidence for the alignment of quasar radio polarizations with large quasar group axes”, Astron. Astrophys. 590, A53, doi: 10.1051/0004-6361/201526979, ARXIV:1604.03937 (2016).
8. Shurtleff, R., “Evaluating the Alignment of the Polarized Radio Waves from 27 QSOs in a Region near the NGP”, https://vixra.org/abs/2105.0091 (2021).
9. Shurtleff, R., “Evaluating the Alignment of the Polarized Starlight from 99 Stars in a Region off the Disk of the Milky Way”, https://vixra.org/abs/2104.0074 (2021).
10. Tiwari, P. & Jain, P., “Extracting the spectral index of the intergalactic magnetic field from radio polarizations”, MNRAS 460, pp. 2698-2705, doi: 10.1093/mnras/stw1108, https://academic.oup.com/mnras/article-pdf/460/3/2698/8129773/stw1108.pdf (2016).
11. See Table 1 QSOs and Sec. 2 in Pelgrims, V. and Hutsemékers, D., Polarization alignments of quasars from the JVAS/CLASS 8.4-GHz surveys, MNRAS, 450, 4161-4173, doi: 10.1093/mnras/stv917, arXiv:astro-ph:1503.03482 (2015).
12. The JVAS1450 catalog was emailed to me with kind thanks from V. Pelgrims .
13. Jackson, N., Battye, R. A., Browne, I. W. A., Joshi, S., Muxlow, T. W. B., and Wilkinson, P. N., A survey of polarization in the JVAS/CLASS flat-spectrum radio source surveys - I. The data and catalogue production, MNRAS, 376, 371-377, doi: 10.1111/j.1365-2966.2007.11442.x , arXiv:astro-ph/0703273 (2007).
14. R. Shurtleff, “Indirect polarization alignment with points on the sky, the Hub Test” , https://vixra.org/abs/2011.0026 (2020).
15. Hutsemékers, D., Braibant, L., Pelgrims, V., and Sluse, D., Alignment of quasar polarizations with large-scale structures, Astron. Astrophys., 572, A18, doi: 10.1051/0004-6361/201424631, arXiv:astro-ph:1409.6098 (2014).
16. Wikipedia contributors. “Aitoff projection.” Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia, 25 May. 2017. Web. (3 Jan. 2018).
17. The NASA/IPAC Extragalactic Database (NED), https://ned.ipac.caltech.edu/

Part II Computer Program

1. Introduction to Part II

In[2]:=
```
Print["The computer time expended so far is ", TimeUsed[], " seconds."]
```

```
The computer time expended so far is 1.078 seconds.
```

The following computer program, a Mathematica notebook, performs the calculations made to evaluate the alignment of the sources in the sample under consideration.

Since Mathematica encodes the instructions, it is inconvenient to try to run the computer program from the pdf version of this work. A viable .nb version that runs on Mathematica is available by following the link in Ref. 1.

2. Coordinates, utility functions, derivation of basic formula

2a. Coordinates, utility functions

Consider the “Celestial Sphere”, a sphere with unit radius in 3 dimensional Euclidean space. See Figs. 1, 2, 3 in the article, Part 1 above. The sphere is also called the “sphere” or sometimes “the sky”. Picture the dome of a planetarium viewed from the outside. 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 due “North”. Equatorial longitude is the Right Ascension α and latitude is the declination δ.

Definitions:

homeDirectory directory containing the notebook and data files

Utilities:

er, eN, eE unit vectors in a 3D Cartesian coordinate system
(α,δ) equatorial coordinates longitude and latitude
er(α,δ) radial unit vectors from Origin
eN(α,δ) local North at a point on the Celestial Sphere
eE(α,δ) local East at a point on the Celestial Sphere
αFROMr(er) α determined by a radial unit vector er
δFROMr(er) δ determined by a radial unit vector er

Aitoff Plot Functions:

αHA(α,δ) , xH(α,δ) , yH(α,δ), where xH is centered on $\alpha = 0$ and α increases from left-to-right, with α = -180° on the left and +180° on the right

xH180(α,δ) , yH180(α,δ), where xH is centered on $\alpha = 180°$ and α increases from left-to-right, with α = 0° on the left and 360° on the right

mean the arithmetic average of a set of numbers, $\frac{1}{N}\sum_{i=1}^{N} n_i$

stanDev the standard deviation. Given a set of N numbers n_i with mean value m, the standard deviation is $\left(\frac{1}{N}\sum_{i=1}^{N}(n_i - m)^2\right)^{1/2}$, the square root of the average of the squares of the differences of the numbers with the mean. Note that we divide by N to get the average of the deviations squared.

Derivation of η_{iH}:

denoSquared1 magnitude of $r_H - (r_H.r_S)\, r_S$ part of the formula for v_H, see Fig. 2

vHperpS the part of vH that contributes to the dot product $\cos\eta = v\psi.vH$, Eq. 2

vψ the unit vector in the 2D tangent plane at S pointing in the direction of the polarization position angle ψ

ηiH0 the alignment angle η_{iH} between v_H and v_ψ for the ith source

ηiHwithIndeterminate - same as ηiH0, but simplified. It includes the indeterminacy where $H = S$,

ηiH same as ηiHwithIndeterminate, but with $\eta iH = \pi/4$ when H and S are closer than 10^{-3} radians.

In[3]:=
```
homeDirectory =
  "C:\\Users\\shurt\\Dropbox\\HOME_DESKTOP-0MRE5OJ\\SendXXX_CJP_CEJPetc\\SendViXra\\
   20200715AlignmentMethod\\20210505AlignmentMethodv4\\20210916
   Clump2QSOsNearRA175Dec50";
 (*The notebook file and data files for this notebook are put in this directory. *)
```

In[4]:=
```
(* For a Source at (α,δ) = (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}
{"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]}]]]}
```

Out[7]=
```
{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 a radial vector r:

In[8]:=
```
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[2. π - ArcTan[Abs[r[[2]] / r[[1]]]]] /; (r[[2]] < 0 && r[[1]] > 0)
αFROMr[r_] := π / 2. /; (r[[2]] ≥ 0 && r[[1]] == 0)
αFROMr[r_] := 3 π / 2. /; (r[[2]] < 0 && r[[1]] == 0)
```

In[14]:=
```
δFROMr[r_] := N[ArcTan[r[[3]] / (√(r[[1]]^2 + r[[2]]^2))]] /; (√(r[[1]]^2 + r[[2]]^2) > 0)
δFROMr[r_] := Sign[r[[3]]] (π / 2.) /; (√(r[[1]]^2 + r[[2]]^2) == 0)
```

The following Aitoff Plot formulas can be found in Wikipedia, Ref. 16.
For these formulas the angles α and δ should be in degrees.
They give an Aitoff Plot that is centered on (0°,0°)
The quantity "αH" is the RA coordinate of a point H on the Celestial Sphere. Thus, we use "αHA" for Aitoff function.

In[16]:=
```
αHA[α_, δ_] := αHA[α, δ] = ArcCos[Cos[((2. π) / 360.) δ] Cos[((2. π) / 360.) α / 2.]]
xH[α_, δ_] := xH[α, δ] = (2. Cos[((2. π) / 360.) δ] Sin[((2. π) / 360.) α / 2.]) / Sinc[αHA[α, δ]]
yH[α_, δ_] := yH[α, δ] = Sin[((2. π) / 360.) δ] / Sinc[αHA[α, δ]]
```

Using the following functions produces an Aitoff Plot that is centered on (180°,0°)

In[19]:=
```
xH180[α_, δ_] :=
 xH180[α, δ] = (2. Cos[((2. π) / 360.) δ] Sin[((2. π) / 360.) (α - 180.) / 2.]) / Sinc[αHA[(α - 180.), δ]]
yH180[α_, δ_] := yH180[α, δ] = Sin[((2. π) / 360.) δ] / Sinc[αHA[(α - 180.), δ]]
```

In[21]:=
```
mean[data_] := (1 / Length[data]) Sum[data[[i4]], {i4, Length[data]}];
(* arithmetic average *)
stanDev[data_] :=
 ((1 / Length[data]) Sum[(data[[i5]] - mean[data])^2, {i5, Length[data]}])^(1/2)
 (*standard deviation*)
```

2b. Derivation of a formula for the alignment angle η_{iH} given the position r_S of the ith source , the location r_H of point H , and the polarization direction ψ for the ith source

From Fig 2b, we see that $\cos\eta = \mathrm{v}\psi.\mathrm{vH}$, Eq. 2.

$\mathrm{vH} = \frac{\mathrm{rH} - (\mathrm{rH.rS})\,\mathrm{rS}}{[(\mathrm{rH} - (\mathrm{rH.rS})\,\mathrm{rS}).(\mathrm{rH} - (\mathrm{rH.rS})\,\mathrm{rS})]^{1/2}}$: unit vector in the 2D tangent plane at S, in the direction of H from S, vH.rS = 0, where er[αH,δH].er[αS,δS] = rH.rS is the inner product of the radial unit vectors rH and rS to point H and source S

Since vψ is also perpendicular to rS, it follows that vψ.rS = 0, and we have $\frac{\mathrm{rH}}{[(\mathrm{rH} - (\mathrm{rH.rS})\,\mathrm{rS}).(\mathrm{rH} - (\mathrm{rH.rS})\,\mathrm{rS})]^{1/2}}$ as the part of vH that contributes to the dot product $\cos\eta = \mathrm{v}\psi.\mathrm{vH}$. Therefore, define

$$\mathrm{vHperpS} = \frac{\mathrm{rH}}{[(\mathrm{rH} - (\mathrm{rH.rS})\,\mathrm{rS}).(\mathrm{rH} - (\mathrm{rH.rS})\,\mathrm{rS})]^{1/2}}$$

Simplify the denominator,

In[23]:=
```
denoSquared1 = FullSimplify[(er[αH, δH] - (er[αH, δH].er[αS, δS]) er[αS, δS]).
    (er[αH, δH] - (er[αH, δH].er[αS, δS]) er[αS, δS])];
(* denoSquared = [rH - (rH.rS) rS].[rH - (rH.rS) rS] =
     rH.rH - 2(rH.rS)^2 + (rH.rS)^2 rS.rS =
      1 - 2(rH.rS)^2 + (rH.rS)^2 = 1 - (rH.rS)^2*)
```

In[24]:=
```
FullSimplify[denoSquared1 - (1 - (er[αH, δH].er[αS, δS])^2)] (*check that*)
```

Out[24]=
```
0
```

Write the formula for the vector vHperpS, with a denominator of $(1 - (\mathrm{rH.rS})^2)^{1/2}$:

In[25]:=
```
vHperpS[αS_, δS_, αH_, δH_] := er[αH, δH] / (1 - (er[αH, δH].er[αS, δS])^2)^(1/2)
```

In[26]:=
```
Simplify[vHperpS[αH, δH, αH, δH] ];(* BANG,
BOOM!! when H = S . See Fig. 2 for why this happens.*)
```

```
••• Simplify: Expression Cos[αH] Cos[δH] / Sqrt[1 - (Power[«2»] Power[«2»] + Power[«2»] Power[«2»] + Sin[«1»]^2)^2] simplified to ComplexInfinity.
••• Simplify: Expression Cos[δH] Sin[αH] / Sqrt[1 - (Power[«2»] Power[«2»] + Power[«2»] Power[«2»] + Sin[«1»]^2)^2] simplified to ComplexInfinity.
••• Simplify: Expression Sin[δH] / Sqrt[1 - (Power[«2»] Power[«2»] + Power[«2»] Power[«2»] + Sin[«1»]^2)^2] simplified to Indeterminate.
••• General: Further output of Simplify::infd will be suppressed during this calculation.
```

The other vector we need is vψ, the unit vector in the 2D tangent plane at S pointing in the direction of the polarization position angle ψ. By Fig. 2b, one sees that

$$\mathrm{v}\psi \;=\; \cos(\psi)\,\mathrm{N} \;+\; \sin(\psi)\,\mathrm{E}\,,$$

where N and E are local north and east unit vectors in the 2D tangent plane at S.

In[27]:=
```
vψ[αS_, δS_, αH_, δH_, ψ_] := Cos[ψ] eN[αS, δS] + Sin[ψ] eE[αS, δS]
(*vψ[αS,δS,αH,δH,ψ]*)
```

The alignment angle η is the acute angle between vψ and vH in the 2D tangent plane at S. By Eq. 2,

In[28]:=
```
ηiH0[αS_, δS_, αH_, δH_, ψ_] :=
 ArcCos[ Abs[vψ[αS, δS, αH, δH, ψ].vHperpS[αS, δS, αH, δH] ] ]
(*ηiH0[αS,δS,αH,δH,ψ]*)
FullSimplify[ηiH0[αS, δS, αH, δH, ψ]]
```

Out[29]=
```
ArcCos[
 Abs[(Cos[δS] Cos[ψ] Sin[δH] + Cos[δH] (-Cos[αH - αS] Cos[ψ] Sin[δS] + Sin[αH - αS] Sin[ψ])) / Sqrt[1 - (Cos[αH - αS] Cos[δH] Cos[δS] + Sin[δH] Sin[δS])^2]]]
```

In[30]:=
```
(*The following function is well-
 behaved everywhere except where ±H coincides with ±S.*)
ηiHwithIndeterminate[αS_, δS_, αH_, δH_, ψ_] := ArcCos[Abs[
   (Cos[δS] Cos[ψ] Sin[δH] + Cos[δH] (-Cos[αH - αS] Cos[ψ] Sin[δS] + Sin[αH - αS] Sin[ψ])) /
    (Sqrt[(1 - (Cos[αH - αS] Cos[δH] Cos[δS] + Sin[δH] Sin[δS])^2)])] ]
```

In[31]:=
```
(*Since η is an acute angle, let us take the halfway value,
η = π/4 in the neighborhood where H ≈ S.*)
ηiH[αS_, δS_, αH_, δH_, ψ_] :=
 ηiHwithIndeterminate[αS, δS, αH, δH, ψ] /; ((1 - (er[αH, δH].er[αS, δS])^2) ≥ 0.000001)
ηiH[αS_, δS_, αH_, δH_, ψ_] := π / 4. /; ((1 - (er[αH, δH].er[αS, δS])^2) < 0.000001)
Print[
 "Thus η_iH = π/4 wherever ±H is 'close' to ±S, with 'close' meaning within an angle of ",
 ArcSin[0.000001^(1/2)], " radians, or ", ArcSin[0.000001^(1/2)] (360./(2. π)), "°."]
```

Thus $\eta_{iH} = \pi/4$ wherever ±H is 'close' to ±S, with 'close' meaning within an angle of 0.001 radians, or 0.0572958°.

3. Polarization and Position Data

3a. Source Data

The JVAS1450 catalog incorporates data from the large JVAS/CLASS 8.4 Ghz catalog Jackson 2007, Refs. 11,12,13. The JVAS1450 catalog sources were filtered from Jackson 2007 sources by identification as QSOs. Filters: for percent polarization, p > 0.6%, for the largest fractional uncertainty in percent polarization, σp/p < 0.6%, and for uncertainty in the polarization position angle $\sigma_\psi < 16°$.

Definitions:

data00	the catalog data, JVAS1450
secondClumpQsosIDinData001450	- record numbers in the catalog of the QSOs in the sample
nSrc	number of sources
αSrc	right ascension of the sources, longitude (radians)
δSrc	declination of the sources, latitude (radians)
ψSrc	PPA, polarization position angle of the sources: clockwise from North with East to the right.
σψSrc	uncertainty in PPA
percentPol	percentage of linear polarization of the sources
redshift	redshift, no uncertainty reported
rSrc	unit vectors from the Origin to Sources on Celestial Sphere
eNSrc	Local North at each Source
eESrc	Local East at each Source
ηBarAtHwithAnyψ	alignment angle function $\overline{\eta}$(H), Eqn. 1, obtained using the location of the sources
sourceCenter	unit radial vector to the arithmetic center of the sources
αSourceCenter	Right Ascension at the sourceCenter
δSourceCenter	Declination at the sourceCenter
angleSourceToCenter	angle from each Source to the sourceCenter
ρRgnRadius	angle to the furthest QSO from the sourceCenter
ρRMS	root-mean-square angular distance to the sources from the sourceCenter

Alternate names:

A position search of the NASA/IPAC Extragalactic Database (NED)*, Ref. 17, returned the following names of 13 QSOs whose position is coincident with those reported in the JVAS1450 catalog:

1. WISEA J104732.27+483531.1
2. B3 1048+470B (Redshift = 1.4194 JVAS1450, 1.8x10^-4 NED Sloan Digital Sky Survey)
3. WISE J105840.84+533543.1
4. B3 1108+454
5. WISEA J111740.33+525936.4
6. WISEA J112152.33+493225.5
7. SDSS J112337.12+504531.8, SDSS J112337.11+504531.8
8. B3 1124+455
9. B3 1140+466
10. B3 1143+446A
11. SBS 1149+499
12. SBS 1150+497
13. WISEA J115826.77+482516.1

Note the disagreement in the redshift values for object 2. B3 1048+470B. The other redshifts were nearly the same in both NED and JVAS1450.

These identifications are FYI, for your information. No data from the NED search is used in this notebook.

*The NASA/IPAC Extragalactic Database (NED) is funded by the National Aeronautics and Space Administration and operated by the California Institute of Technology.

In[35]:=
```
(*right ascension in radians*)
αSrc = 10^-6. {2825418, 2841679, 2874039, 2928868, 2956905, 2975226,
    2982850, 2997432, 3068298, 3078944, 3109077, 3112830, 3134812};
```

In[36]:=
```
nSrc = Length[αSrc]
```

Out[36]= 13

In[37]:=
```
(*declination in radians*)
δSrc = 10^-6. {848089, 815735, 935416, 787604, 924911,
    864641, 885908, 790083, 809670, 773869, 866744, 864271, 845109};
```

In[38]:=
```
(* position angle in radians*)
ψSrc = 10^-6. {1293289, 1328196, 1930334, 925025, 1759292,
    2008874, 2021091, 994838, 975639, 945968, 2426008, 1987930, 3082251};
```

In[39]:=
```
Histogram[ψSrc (360./(2. π)), {20}, PlotLabel → "PPA ψ, number ΔR per bin",
 AxesLabel → {"ψ", "ΔR"}, PlotRange → {{0, 200}, Automatic}]
Print["Figure 8: Distribution of position angles for the 13
   polarization directions in the sample. Note the wide distribution
   over a hundred degrees or so, ψ = 40° to ψ = 150°, in two groupings."]
```

Out[39]=

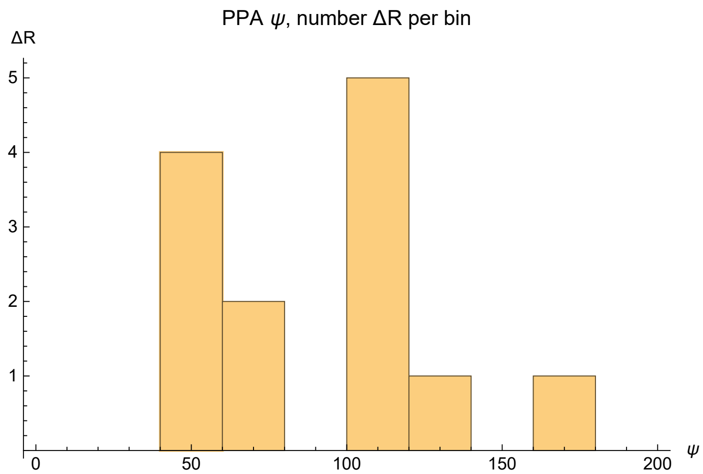


Figure 8: Distribution of position angles for the 13 polarization directions in the sample. Note the wide distribution over a hundred degrees or so, ψ = 40° to ψ = 150°, in two groupings.

In[41]:=
```
(*uncertainty in ψ in radians*)
σψSrc = 10^-6. {39697, 48409, 72563, 55071,
    86756, 131967, 87055, 3977, 21712, 20791, 74085, 24677, 16969};
```

In[42]:=
```
(* % polarization*)
percentPol = 10^-6. {2142363, 575196, 12801608, 4141751, 3722694,
    2159228, 3458875, 1323236, 3206987, 2150994, 471406, 904146, 1224728};
```

In[43]:=
```
(* uncertainty in % polarization*)
σpercentPol = 10^-6. {170077, 55686, 1857703, 456149,
    645884, 569853, 602180, 10524, 139250, 89438, 69845, 44620, 41561};
```

In[44]:=
```
(*Redshift*)
redshift = 10^-6. {867000, 1419400, 1535100, 1492000, 1373300,
    1875000, 2277500, 1819200, 1321800, 299800, 1094100, 333700, 2028000};
```

In[45]:=
```
rSrc = Table[er[ αSrc[[i]], δSrc[[i]] ],  {i, nSrc}];(*calculated from Input.*)
eNSrc = Table[eN[ αSrc[[i]], δSrc[[i]] ], {i, nSrc}];(*calculated from Input.*)
eESrc = Table[eE[ αSrc[[i]], δSrc[[i]] ], {i, nSrc}];(*calculated from Input.*)
```

In[48]:=
```
ηBarAtHwithAnyψ[αH_, δH_, ψ_] :=
 1/nSrc Sum[ηiH[αSrc[[i]], δSrc[[i]], αH, δH, ψ[[i]] ], {i, nSrc}]
(*ηBarAtHwithAnyψ[3.5,0.6,ψSrc]*)(* An example with a selected
 αH and δH and with the observed polarization directions for ψ*)
```

In[49]:=
```
sourceCenter0 = 1/nSrc Sum[rSrc[[i]], {i, nSrc}];
sourceCenter = sourceCenter0/(sourceCenter0.sourceCenter0)^(1/2);
(*unit radial vector to the arithmetic average center of the sources.*)
αSourceCenter = αFROMr[sourceCenter];
δSourceCenter = δFROMr[sourceCenter];
angleSourceToCenter = Table[ArcCos[rSrc[[i]].sourceCenter], {i, nSrc}];
ρRgnRadius = Sort[angleSourceToCenter][[-1]]; (*Furthest source from center*)

ρRMS = (1/nSrc Sum[angleSourceToCenter[[i]]^2, {i, nSrc}])^(1/2);
```

3b. Section Summary

We consider Quasi-Stellar Objects, QSOs. From the data in JVAS1450, 5° radius regions are constructed, one centered at each of the 10518 grid points of a 2°x2° mesh. The 1450 QSOs were assigned to the regions based on location and we calculated the significance of the alignment of the polarization directions for the sources in each region.

The three such QSO regions selected for this notebook satisfied many requirements: (*i*) have 7 or more sources in order to use the significance formulas in Sec. 4 accurately, (*ii*) have longitude RA 160° $\leq \alpha \leq$ 180°, (*iii*) have latitude dec 40° $\leq \delta \leq$ 55°, (*iv*) whose QSOs are very significantly aligned, $S \leq 10^{-2}$. There are 3 regions satisfying (*i*) - (*iv*) containing a total of 13 sources. See Fig. 1 and the discussion there.

In[56]:=
```
Print["There are ", nSrc, " sources in the sample."]
Print["Check that the Sample obeys the data cuts:"]
Print[
 "Check that the smallest % polarization p in the sample is 0.5% or more. Smallest: ",
 Sort[percentPol][[1]], "% ."]
Print["Check that the largest fractional uncertainty in % polarization, σp/p,
   is less than 0.6 . Largest: ", Sort[σpercentPol / percentPol][[-1]], " ."]
Print["Check that the largest PPA ψ uncertainty σψ is less than 16°. Largest: ",
 Sort[σψSrc][[-1]] (360./(2. π)), "° ."]
```

```
There are 13 sources in the sample.

Check that the Sample obeys the data cuts:

Check that the smallest % polarization p in the sample is 0.5% or more. Smallest: 0.471406% .

Check that the largest fractional uncertainty
   in % polarization, σp/p, is less than 0.6 . Largest: 0.263915 .

Check that the largest PPA ψ uncertainty σψ is less than 16°. Largest: 7.56115° .
```

In[61]:=
```
ListPlot[Table[{αSrc[[j]], δSrc[[j]]} (360./(2. π)), {j, nSrc}],
 PlotRange → {{0, 360}, {-90, 90}},
 Ticks → {Table[{i, i}, {i, 0, 360, 60}], Table[{j, j}, {j, -90, 90, 30}]},
 PlotLabel → "Sources", AxesLabel → {"α, degrees", "δ, degrees"}, PlotStyle → Green]
Print["Figure 9: The locations of the ", nSrc,
 " QSOs in the sample. The center of the sample has (RA,Dec) = ",
 {αSourceCenter (24./(2. π)), δSourceCenter (360./(2. π))}, ", in {hours, degrees}.
   The angular separation of the furthest QSO from the sample center is ",
 Sort[angleSourceToCenter][[-1]] (360./(2. π)), "°. The RMS radius is ", ρRMS (360./(2. π)), "°."]
```

Out[61]=

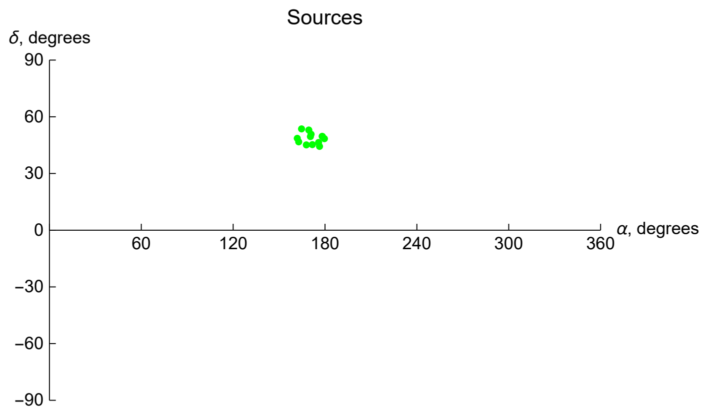


Figure 9: The locations of the 13
 QSOs in the sample. The center of the sample has (RA,Dec) = {11.4297, 48.6782}
, in {hours, degrees}. The angular separation of the furthest QSO from the sample center is
6.49406°. The RMS radius is 4.72813°.

4. Grid

While we have a formula $\overline{\eta}$(H) for the alignment angle at a point H on the Celestial Sphere, there are occasions when it is better not to use it and, instead, construct a discrete table of values. To locate the values $\overline{\eta}$(H) at a finite number of points *H* on the sphere, we create a grid, or mesh, of grid points.

When building the 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. Symmetry across diameters gives the other hemisphere. The grid is conveniently developed centered at the North pole and then rotated to be centered on the sample of sources. For detailed work near the sources a 30° finely spaced grid cap is produced to supplement the more coarsely spaced grid. The fine and coarse grids are offset so that no grid points are common to the two grids.

4a. Construct the grid

Definitions:

gridSpacing, coarseGridSpacing - fine, coarse grid separation in degrees between grid points on and between constant latitude circles

fineCapRadius	radius of the fine grid cap in radians
dθ1, dθ2	fine, coarse grid spacing in radians
idN, ai, ji, δj	dummy indices
αpointH,δpointH	α and δ of the grid points H_j

fineGrid, coarseGrid, gridN, grid - tables of data associated with grid points, record descriptions below

rotzToSample	rotation matrix from North pole to sourceCenter
lpgrid	plot of the radial unit vectors to the grid points
nGrid	number of grid points
αGrid	longitudes at the grid points ($-\pi \le \alpha \le +\pi$)
δGrid	latitudes at the grid points ($-\pi/2 \le \alpha \le \pi/2$)
rGrid	radial unit vectors from origin to grid points, in 3D Cartesian coordinates

In[63]:=
```
gridSpacing = 0.6(*degrees*);
fineCapRadius = 0.5;
```

In[65]:=
```
(*KEEP this cell - DO NOT DELETE*)
(*The Northern Grid "gridN". *)
dθ1 = 2. π/360. gridSpacing (*Convert gridSpacing to radians*); fineGrid = {}; idN = 1;

For[δj = 0., δj < fineCapRadius/dθ1, δj++, δpointH = π/2. - δj dθ1 - dθ1/2.^(1/2);
  (*Print["{δj,δpointH} = ",{δj,δpointH}];*)
  For[ ai = 0., ai < Ceiling[2. π/dθ1 (Cos[δpointH] + 0.01)], ai++, αpointH = ai dθ1/(Cos[δpointH] + 0.01);
   (*Print["{ai,αpointH} = ",{ai,αpointH}];*)
   AppendTo[fineGrid, {idN, ai, δj, αpointH, δpointH, er[αpointH, δpointH]}];
   idN = idN + 1
  ]]
Length[fineGrid];
lpFine = ListPointPlot3D[Table[fineGrid[[i, 6]], {i, 1, Length[fineGrid], 10}], PlotRange →
    {{-1.2, 1.2}, {-1.2, 1.2}, {-1.2, 1.2}}, AxesLabel → {"x", "y", "z"}, BoxRatios → {1, 1, 1}];
```

Coarse Grid band runs from latitude ($\frac{\pi}{2}$ − fineGridMAX) to latitude ($\frac{\pi}{2}$ − southOfEquator)

In[69]:=
```
coarseStart = fineCapRadius; coarseEnd = 1.65 ;(*radians*)
coarseGridSpacing = 2.0(*degrees*);
```

In[71]:=
```
(*KEEP this cell - DO NOT DELETE*)
(*The coarse grid band. *)
dθ2 = (2. π/360.) coarseGridSpacing (*Convert grid spacing to radians*);
coarseGrid = {};
idB = 1 + Length[fineGrid];(* ID for the coarse band grid points*)

For[δj = 0., δj < (coarseEnd - coarseStart)/dθ2, δj++, δpointH = π/2. - coarseStart - δj dθ2 - dθ2/3.^(1/2);
 (*Print["{δj,δpointH} = ",{δj,δpointH}];*)
 For[ ai = 0., ai < Ceiling[(2. π/dθ2) (Cos[δpointH] + 0.01)], ai++, αpointH = ai dθ2/(Cos[δpointH] + 0.01);
  (*Print["{ai,αpointH} = ",{ai,αpointH}];*)
  AppendTo[coarseGrid, {idB, ai, δj, αpointH, δpointH, er[αpointH, δpointH]}];
  idB = idB + 1
 ]]
```

In[73]:=
```
lpCoarse1 = ListPointPlot3D[Table[coarseGrid[[i, 6]], {i, 1, Length[coarseGrid], 10}],
   PlotRange → {{-1.2, 1.2}, {-1.2, 1.2}, {-1.2, 1.2}},
   AxesLabel → {"x", "y", "z"}, BoxRatios → {1, 1, 1}];
Length[coarseGrid];
(*Show[{lpFine,lpCoarse}]*)
```

Now we need to rotate the combined fine/coarse grid ‘gridN’ so that it is centered on the sample, the sourceCenter .

In[75]:=
```
rotzToSample = RotationMatrix[{{0, 0, 1}, sourceCenter }];
%.{0, 0, 1};
sourceCenter ;
```

In[78]:=
```
gridN = Join[fineGrid, coarseGrid];
grid = Table[{gridN[[i, 1]], gridN[[i, 2]], gridN[[i, 3]], gridN[[i, 4]],
    gridN[[i, 5]], rotzToSample.gridN[[i, 6]]}, {i, Length[gridN]}];
lpgrid = ListPointPlot3D[Table[grid[[i, 6]], {i, 1, Length[grid], 10}],
   PlotRange → {{-1.2, 1.2}, {-1.2, 1.2}, {-1.2, 1.2}},
   AxesLabel → {"x", "y", "z"}, BoxRatios → {1, 1, 1}];
```

In[81]:=
```
lpgrid
Print[
 "Figure 10: The grid. The grid is centered on the source sample, with a finely spaced
   cap. The  grid covers one hemisphere, centered on the sample. The fine and coarse
   grids are off-set, so they do not share any grid points. There are ",
 nGrid, " grid points on the hemisphere."]
```

Out[81]=

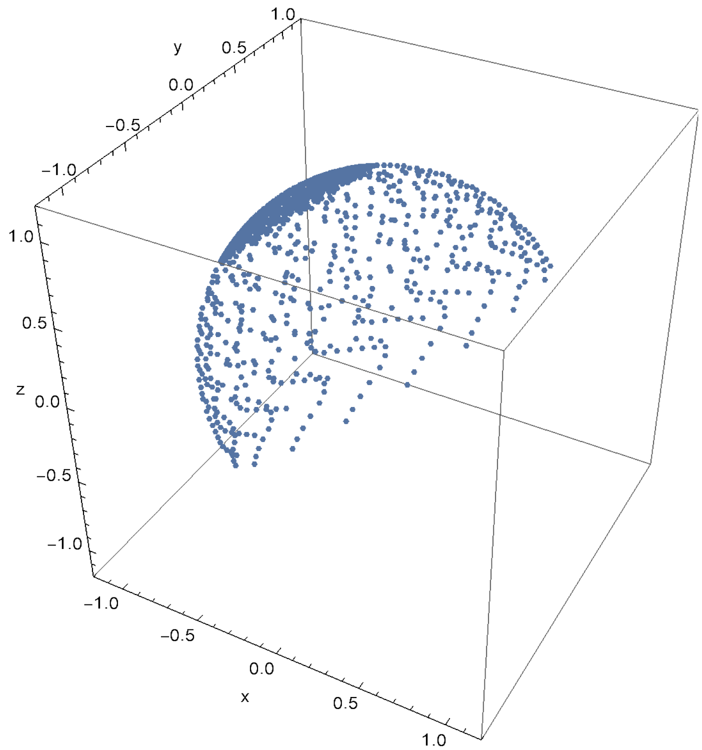


Figure 10: The grid. The grid is centered on the source sample, with a finely spaced cap. The grid covers one hemisphere, centered on the sample. The fine and coarse grids are off-set, so they do not share any grid points. There are nGrid grid points on the hemisphere.

In[83]:=
```
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]}];
nGrid = Length[grid];
```

4b. Section Summary

In[87]:=
```
Print["The fine grid on the 'cap' has ", Length[fineGrid], " grid points."]
Print["The grid points on the cap are separated by gridSpacing = ",
 gridSpacing, "° in latitude and longitude."]
Print["On the entire hemisphere, there is a second set of grid
   points that are separated by gridSpacing = ", coarseGridSpacing,
 "° in latitude and longitude. The two sets do not share any grid points."]
Print["The second set has ", Length[coarseGrid], " grid points."]
Print["The total grid, 'grid', has ", Length[fineGrid],
 " + ", Length[coarseGrid], " = ", Length[grid], " grid points."]
```

```
The fine grid on the 'cap' has 7459 grid points.

The grid points on the cap are separated by gridSpacing = 0.6° in latitude and longitude.

On the entire hemisphere, there is a
   second set of grid points that are separated by gridSpacing =
 2.° in latitude and longitude. The two sets do not share any grid points.

The second set has 5026 grid points.

The total grid, 'grid', has 7459 + 5026 = 12485 grid points.
```

5. The alignment function $\overline{\eta}$(H) for the sample of sources

"Best" means we use the ψSrc that were listed in the catalog. We calculate the alignment function $\overline{\eta}$(H) at the grid points H . Given the alignment function $\overline{\eta}$(H) , one can find the smallest alignment angle $\overline{\eta}_{\min}$ and the largest avoidance angle $\overline{\eta}_{\max}$ and determine the significances for the alignment and avoidance of the polarization directions.

5a. Determine the alignment angle $\overline{\eta}$(H)

First find $\overline{\eta}$(Hj) on the grid and find the smallest and largest values of the alignment function on the grid. Then use the function "ηBarAtHwithAnyψ" derived in Secs. 2 and 3 to go between grid points and locate the smallest and largest angles, $\overline{\eta}_{\min}$ and $\overline{\eta}_{\max}$, and their locations, the hubs $H_{\min}$ and $H_{\max}$. These are the extremes for convergence and divergence of the polarization directions.

Definitions:

vψSrc	unit vectors along the polarization directions ψ in the tangent planes of the sources
eN	local unit vectors along local North
eE	local unit vectors along local East
gridjηBarHj	{j, $\overline{\eta}$(Hj) }, 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).
sortgridjηBarHj	{j, $\overline{\eta}$(Hj) }, with smallest angles $\overline{\eta}$(H) first.
gridjηBarMin	{j,$\overline{\eta}$(H)}, the j and $\overline{\eta}$ for the smallest value of $\overline{\eta}$(H) , best alignment
gridjηBarMin	index j for the grid point H with the smallest value of $\overline{\eta}$(H)
gridηBarMin	smallest $\overline{\eta}$(H) on grid
gridjηBarMax	{j,$\overline{\eta}$(H)}, the j and $\overline{\eta}$ for the largest value of $\overline{\eta}$(H) , best alignment
gridjηBarMax	index j for the grid point H with the largest value of $\overline{\eta}$(H)
gridηBarMax	largest $\overline{\eta}$(H) on grid

ηmin$\alpha\delta$HObs	smallest $\overline{\eta}$(H) and H, local min near gridjηBarMin (use "ηBarAtHwithAnyψ" off-grid)
ηmax$\alpha\delta$HObs	largest $\overline{\eta}$(H) and H, local max near gridjηBarMax
funcDataObs	off-grid data for extreme alignment angles $\overline{\eta}$ and their hubs H

ηBarMinfunDataObs	$\overline{\eta}_{\min}$
ηBarMaxfunDataObs	$\overline{\eta}_{\max}$
HminαfunDataObs	$H_{\min}$ location RA α in radians
HminδfunDataObs	$H_{\min}$ location dec δ in radians
Hmin$\alpha\delta$funDataObs	$H_{\min}$ location (RA,dec) = (α, δ) in radians

HmaxαfunDataObs H_{max} location RA α in radians

HmaxδfunDataObsH_{max} location dec δ in radians

HmaxαδfunDataObs H_{max} location (RA,dec) = (α, δ) in radians

In[92]:=
```
(* vψ, eN, eE unit vectors in the tangent plane of each source Si,
pointing along the polarization direction, local North,
and local East, respectively. See Fig. 2.*)
vψSrc = Table[Cos[ ψSrc[[i]] ] eN[ αSrc[[i]], δSrc[[i]] ] +
     Sin[ ψSrc[[i]] ] eE[ αSrc[[i]], δSrc[[i]] ], {i, nSrc}];
```

In[93]:=
```
(* Analysis using Eq (5) in Ref. 14 to get η̄(Hj). First ηiH, cos(ηiH) = |v̂H.v̂ψi|,
where "v̂H" was called "vHperpS" in a previous discussion. Thus,
we can get η̄(Hj), by Eq. (2): *)
gridjηBarHj =
  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}];
 sortgridjηBarHj = Sort[gridjηBarHj, #1[[2]] < #2[[2]] &];
gridjηBarMin = sortgridjηBarHj[[1]]; (*  {j,η̄(Hj)} for smallest η̄(Hj) *)
gridηBarMin = gridjηBarMin[[2]];
gridjηBarMax = sortgridjηBarHj[[-1]]; (*  {j,η̄(Hj)} for largest η̄(Hj) *)
gridηBarMax = gridjηBarMax[[2]] ;
```

The results just found on the grid should be close to the results. Use FindMinimum and FindMaximum to go off-grid and get closer.

In[99]:=
```
ηminαδHObs = FindMinimum[ηBarAtHwithAnyψ[αH, δH, ψSrc],
    {{αH, αGrid[[ gridjηBarMin[[1]] ]]}, {δH, δGrid[[ gridjηBarMin[[1]] ]]}}];
ηmaxαδHObs =
  FindMaximum[ηBarAtHwithAnyψ[αH, δH, ψSrc],
    {{αH, αGrid[[ gridjηBarMax[[1]] ]]}, {δH, δGrid[[ gridjηBarMax[[1]] ]]}}];
funcDataObs = {1, { ηminαδHObs[[1]], {αH, δH} /. ηminαδHObs[[2]]},
  { ηmaxαδHObs[[1]], {αH, δH} /. ηmaxαδHObs[[2]]}}
```

FindMinimum: The line search decreased the step size to within the tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient decrease in the function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

FindMaximum: The line search decreased the step size to within the tolerance specified by AccuracyGoal and PrecisionGoal but was unable to find a sufficient increase in the function. You may need more than MachinePrecision digits of working precision to meet these tolerances.

Out[101]= {1, {0.189626, {3.13394, 0.854791}}, {1.09371, {2.93195, 0.485158}}}

In[102]:=

```
ηBarMinfunDataObs = funcDataObs[[2, 1]];
ηBarMaxfunDataObs = funcDataObs[[3, 1]];
HminαfunDataObs = funcDataObs[[2, 2, 1]];
HminδfunDataObs = funcDataObs[[2, 2, 2]];
HminαδfunDataObs = funcDataObs[[2, 2, 1]];
HmaxαfunDataObs = funcDataObs[[3, 2, 1]];
HmaxδfunDataObs = funcDataObs[[3, 2, 2]];
HmaxαδfunDataObs = {funcDataObs[[3, 2, 1]], funcDataObs[[3, 2, 2]]};
```

In[110]:=

```
Print["When moving off-grid, check that  the
   hubs Hmin and Hmax did not move more than a grid spacing:"]
Print["When we found a local minimum, the hub Hmin moved off-grid by ",
 ArcCos[er[HminαfunDataObs, HminδfunDataObs].
    er[αGrid[[ gridjηBarMin[[1]] ]], δGrid[[ gridjηBarMin[[1]] ]]]   ] (360./(2. π)), "°."]
Print["When we found a local maximum, the hub Hmax moved off-grid by ",
 ArcCos[er[HmaxαfunDataObs, HmaxδfunDataObs].
    er[αGrid[[ gridjηBarMax[[1]] ]], δGrid[[ gridjηBarMax[[1]] ]]]   ] (360./(2. π)), "°."]
Print["The alignment hub Hmin is ",
 ArcCos[er[HminαfunDataObs, HminδfunDataObs].sourceCenter   ] (360./(2. π)),
 "° from the source center."]
Print["The alignment hub Hmin is ",
 ArcCos[er[HmaxαfunDataObs, HmaxδfunDataObs].sourceCenter   ] (360./(2. π)),
 "° from the source center."]
Print["Now compare that with the grid: The fine grid spacing close to the sources is ",
 gridSpacing, "°. If the hub is more than ", fineCapRadius (360./(2. π)),
 "° from the sample center, then the grid spacing is ", coarseGridSpacing, "°."]
```

```
When moving off-grid, check that  the hubs Hmin and Hmax did not move more than a grid spacing:
When we found a local minimum, the hub Hmin moved off-grid by 0.133122°.
When we found a local maximum, the hub Hmax moved off-grid by 0.378856°.
The alignment hub Hmin is 5.34907° from the source center.
The alignment hub Hmin is 21.0509° from the source center.
Now compare that with the grid: The fine grid spacing close to the sources is 0.6
 °. If the hub is more than 28.6479° from the sample center, then the grid spacing is 2.°.
```

5b. Plot the Alignment Angle Function $\overline{\eta}$(H)

Definitions

αHminDegrees	$H_{\min}$ location RA α in degrees
αHminHours	$H_{\min}$ location RA α in hours

δHminDegrees	$H_{\min}$ location Dec δ in degrees
αHmaxDegrees	$H_{\max}$ location RA α in degrees
αHmaxHours	$H_{\max}$ location RA α in hours
δHmaxDegrees	$H_{\max}$ location Dec δ in degrees
rHmin, rHmax	radial unit vectors to the alignment and avoidance hubs $H_{\min}$ and $H_{\max}$
rPerpHmin (max)	a unit vector in the plane of the great circle combining rCenterSrc and rHmin (max)
rGreatMinCircle(θ) (Max)	radial unit vector to a point on the great circle
αGreatMin (Max)	longitude at the point for θ
δGreatMin (Max)	latitude at the point for θ
xyAitoffGreatMin (Max)	Aitoff plot coordinates for the great circles
crossMin (Max)	unit vector perpendicular, normal to the plane of the great circle
θminMAXgreatcircles	angle between the vectors normal to the planes of the two great circles

αjδjηBarHjTable	$\{\alpha_j, \delta_j, \overline{\eta}(\mathrm{H})\}$ at each grid point $H = H_j$, in degrees
xyηBarAitoffTable	{x, y, $\overline{\eta}$(x,y)} , where x,y are Aitoff coordinates and $\overline{\eta}$(x,y) is the alignment angle on grid
xyAitoffSources	{x,y} Aitoff coordinates for the sources' locations on the sphere
dηContourPlot	separation of successive contour lines, in degrees
listCP	list contour plot of $\overline{\eta}$(H) from xyηBarAitoffTable
rPlusψ	unit vector in the polarization directions ψ
polarLines	lines from each source along its polarization direction ψ
mapOfηBar	contour plot of the alignment angle $\overline{\eta}$(H) , adorned with source locations and labels
mapOfηBarLocal	magnified, local view of the map

In[116]:=
```
(* Equatorial coordinates (α,δ) for the hubs Hmin and Hmax in other units.*)
αHminDegrees = HminαfunDataObs (360 / (2 π));
αHminHours = HminαfunDataObs (24 / (2 π)); (*Hmin*)
δHminDegrees = HminδfunDataObs (360 / (2 π));

αHmaxDegrees = HmaxαfunDataObs (360 / (2 π)); (*Hmax*)
αHmaxHours = HmaxαfunDataObs (24 / (2 π));
δHmaxDegrees = HmaxδfunDataObs (360 / (2 π));
```

In[122]:=
```
rHmin = er[ αHminDegrees (2. π / 360.) + π, -δHminDegrees (2. π / 360.) ];
rPerpHmin0 = rHmin - (rHmin.sourceCenter) sourceCenter;
rPerpHmin = rPerpHmin0 / (rPerpHmin0.rPerpHmin0)^(1/2.);
rGreatMinCircle[θ_] := Cos[θ] sourceCenter + Sin[θ] rPerpHmin
αGreatMin[θ_] := αFROMr[rGreatMinCircle[θ]]
δGreatMin[θ_] := δFROMr[rGreatMinCircle[θ]]
xyAitoffGreatMin = Table[{xH180[ αGreatMin[θ] (360 / (2 π)), δGreatMin[θ] (360 / (2 π)) ],
    yH180[ αGreatMin[θ] (360 / (2 π)), δGreatMin[θ] (360 / (2 π)) ]}, {θ, 1, 360}];
```

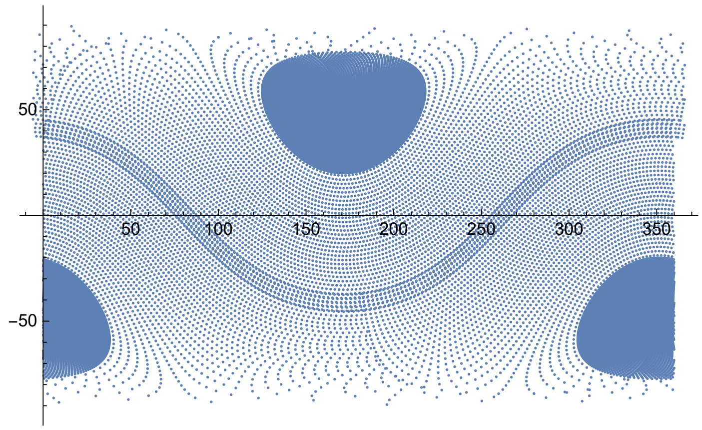
In[129]:=
```
rHmax = er[ αHmaxDegrees (2. π/360.) + π , - δHmaxDegrees (2. π/360.) ];
rPerpHmax0 = rHmax - (rHmax.sourceCenter) sourceCenter;
rPerpHmax = rPerpHmax0/(rPerpHmax0.rPerpHmax0)^(1/2.);
rGreatMaxCircle[θ_] := Cos[θ] sourceCenter + Sin[θ] rPerpHmax
αGreatMax[θ_] := αFROMr[rGreatMaxCircle[θ]]
δGreatMax[θ_] := δFROMr[rGreatMaxCircle[θ]]
xyAitoffGreatMax = Table[{xH180[ αGreatMax[θ] (360 / (2 π)), δGreatMax[θ] (360 / (2 π)) ],
    yH180[ αGreatMax[θ] (360 / (2 π)), δGreatMax[θ] (360 / (2 π)) ]}, {θ, 1, 360}];
```

In[136]:=
```
crossMin0 = Cross[rHmin, sourceCenter];
crossMin = crossMin0/(crossMin0.crossMin0)^(1/2.);
crossMax0 = Cross[rHmax, sourceCenter];
crossMax = crossMax0/(crossMax0.crossMax0)^(1/2.);
θminMAXgreatcircles = ArcCos[crossMax.crossMin] (360./2. π);
```

In[141]:=
```
(*The following table αjδjηBarHjTable is created to
 generate a map of the alignment angle η̅(H) over the sphere.*)
(* Table αjδjηBarHjTable
 entries: 1.  α  2. δ  3. alignment angle ηBarRgnkj at grid point (all in degrees)*)
αjδjηBarHjTable = ( αjδjηBarHjTable0 = {};
   For[j = 1, j ≤ Length[gridjηBarHj], j++,
    AppendTo[ αjδjηBarHjTable0,  {αGrid[[j]] * (360. / (2. π)), δGrid[[j]] * (360. / (2. π)),
      gridjηBarHj[[j, 2]] * (360. / (2. π))}]   ; If[ 360. ≥ αGrid[[j]] * (360. / (2. π)) > 180.,
     AppendTo[ αjδjηBarHjTable0,  {αGrid[[j]] * (360. / (2. π)) - 180.,
       -δGrid[[j]] * (360. / (2. π)), gridjηBarHj[[j, 2]] * (360. / (2. π))}]   ] ;
    If[ 180. > αGrid[[j]] * (360. / (2. π)) > 0., AppendTo[ αjδjηBarHjTable0,
      {αGrid[[j]] * (360. / (2. π)) + 180., -δGrid[[j]] * (360. / (2. π)),
       gridjηBarHj[[j, 2]] * (360. / (2. π))}]   ] ;
    If[ 360. ≥ αGrid[[j]] * (360. / (2. π)) > 354., AppendTo[ αjδjηBarHjTable0,  {αGrid[[j]] * (360. /
          (2. π)) - 360., δGrid[[j]] * (360. / (2. π)), gridjηBarHj[[j, 2]] * (360. / (2. π))}]   ] ;
    If[ +6. > αGrid[[j]] * (360. / (2. π)) ≥ 0., AppendTo[ αjδjηBarHjTable0,
      {αGrid[[j]] * (360. / (2. π)) + 360, δGrid[[j]] * (360. / (2. π)),
       gridjηBarHj[[j, 2]] * (360. / (2. π))}]   ]         ];
   αjδjηBarHjTable0);
```

In[142]:=
```
(*The grid does not cover the sphere. Check that the
 αjδjηBarHjTable table covers the entire Celestial Sphere. *)
ListPlot[Table[{αjδjηBarHjTable[[i, 1]], αjδjηBarHjTable[[i, 2]]},
  {i, Length[αjδjηBarHjTable]}]]
Print["Figure 11: Check.  Since the grid does not cover the sphere, only half, we
   should check that the αjδjηBarHjTable table covers the entire Celestial Sphere. "]
```

Out[142]=




```
Figure 11: Check.  Since the grid does not cover the sphere, only half, we
  should check that the αjδjηBarHjTable table covers the entire Celestial Sphere.
```

In[144]:=
```
(*Transcribe the alignment function  η̄(H), the location of the sources,
and the Celestial Equator onto an Aitoff plot.*)
xyηBarAitoffTable = Table[{xH180[αjδjηBarHjTable[[k, 1]], αjδjηBarHjTable[[k, 2]]],
    yH180[αjδjηBarHjTable[[k, 1]], αjδjηBarHjTable[[k, 2]]], αjδjηBarHjTable[[k, 3]]},
   {k, Length[αjδjηBarHjTable]}]; (* The alignment angle function η̄(H) on the grid,
mapped onto a 2D Aitoff projection of the sphere. *)

xyAitoffSources = Table[{xH180[ αSrc[[n]] (360/(2π)), δSrc[[n]] (360/(2π)) ],
    yH180[ αSrc[[n]] (360/(2π)), δSrc[[n]] (360/(2π)) ]}, {n, nSrc}];
 (*The Aitoff coordinates for the sources' locations.*)
```

In[146]:=
```
(* Contour plot of the alignment angle function η̄(H) on the grid. *)
dηContourPlot = 6;
(*, in degrees. *)listCP = ListContourPlot[Union[xyηBarAitoffTable(*,{{xH180[αHminDegrees,
      δHminDegrees],yH180[αHminDegrees,δHminDegrees],ηBarMin*(360./(2.π))-1.0}},
   {{xH180[αHmaxDegrees,δHmaxDegrees],yH180[αHmaxDegrees,δHmaxDegrees],ηBarMax*(360./(2.π))+
      1.0}}*)], AspectRatio→1/2, Contours→Table[η, {η, Floor[gridjηBarMin[[2]]*
        (360./(2.π))] + 1, Ceiling[gridjηBarMax[[2]]*(360./(2.π))] - 1, dηContourPlot}],
  ColorFunction→"TemperatureMap", PlotRange→{{-4.0, 3.5}, 7.5/11.0 {-3, 3}}, Axes -> False,
  Frame→False, PlotLegends→Placed[BarLegend[Automatic, LegendMargins→{{0, 0}, {10, 5}},
     LegendLabel→"η̄(H), °", LabelStyle→{Plain, FontFamily→"Times"}], Right]];
```

In[147]:=
```
(*Construct the map of η̄(H).*)
mapOfηBar =
  Show[{listCP, Table[ParametricPlot[{xH180[α, δ], yH180[α, δ]},
      {δ, -90, 90}, PlotStyle → {Black, Thickness[0.002]}, (*Mesh→{11,5,0}
      (*{23,11,0}*),MeshStyle→Thick,*)PlotPoints → 60], {α, 0, 360, 30}],
    Table[ParametricPlot[{xH180[α, δ], yH180[α, δ]}, {α, 0, 360},
      PlotStyle → {Black, Thickness[0.002]}, (*Mesh→{11,5,0}(*{23,11,0}*),
      MeshStyle→Thick,*)PlotPoints → 60], {δ, -60, 60, 30}], Graphics[
     {PointSize[0.004], Text[StyleForm["N", FontSize -> 14, FontWeight -> "Plain"], {0, 1.85}],
      Text[StyleForm["Equatorial Coordinate System", FontSize -> 14, FontWeight -> "Plain"],
       {0, -1.85}], (*Sources S:*)Green, Point[ xyAitoffSources ], Gray,
      PointSize[0.002], Point[ xyAitoffGreatMin ], Point[ xyAitoffGreatMax ], Black,
      Text[StyleForm["Hmax", FontSize → 12, FontWeight -> "Bold"], {-3.3, +1.0}],
      {Arrow[BezierCurve[{{-3.3, +1.2}, {-1.3, +3.0},
          {xH180[αHmaxDegrees, δHmaxDegrees], yH180[αHmaxDegrees, δHmaxDegrees]}}]]},
      Text[StyleForm["Hmin", FontSize → 12, FontWeight -> "Bold"], {3.3, 1.0}],
      {Arrow[BezierCurve[{{3.3, 1.2}, {0.3, 3.0},
          {xH180[αHminDegrees, δHminDegrees], yH180[αHminDegrees, δHminDegrees]}}]]},
      Text[StyleForm["Hmin", FontSize → 12, FontWeight -> "Bold"], {-3.3, -1.0}],
      {Arrow[BezierCurve[{{-3.3, -1.2}, {-2.3, -2.5}, {xH180[αHminDegrees - 180, -δHminDegrees],
           yH180[αHminDegrees - 180, -δHminDegrees]}}]]}, (**)
      Text[StyleForm["Hmax", FontSize → 12, FontWeight -> "Bold"], {3.3, -1.0}] ,
      {Arrow[BezierCurve[{{3.3, -1.2}, {2.3, -2.0}, {xH180[αHmaxDegrees + 180, -δHmaxDegrees],
           yH180[αHmaxDegrees + 180, -δHmaxDegrees]}}]]}
     }]}, ImageSize → 0.9 × 432];
```

In[148]:=
```
(*
SetDirectory[homeDirectory]
 Export["20210517QSOnearbyHmin.pdf",mapOfηBar]
*)
```

5c. Section Summary

In[149]:=
```
mapOfηBar
Print[
 "Figure 12: The alignment function η̄(H), Eq. (1). The map is centered on (α,δ)=(180°,0°),"]
Print["with α = 0° on the left and α = 360° on the right, Equatorial Coordinates."]
Print["The sources are located at the dots, shaded ", Green, " ."]
Print["The smallest alignment angle is η̄min = ",
 ηBarMinfunDataObs (360. / (2. π)), "°, located at the"]
Print["alignment hubs Hmin and -Hmin in the areas shaded ", Blue, " . "]
Print["The hubs Hmin and -Hmin are located at (α,δ) = ", Round[{αHminDegrees, δHminDegrees}],
 " and ", Round[{αHminDegrees - 180, -δHminDegrees}], " , in degrees."]
Print["The angle between the sample's center and the closest alignment hub Hmin is ",
 ArcCos[-(rHmin.sourceCenter)] (360./(2. π)), "°."]
Print["The largest avoidance angle is η̄max = ",
 ηBarMaxfunDataObs (360. / (2. π)), "°, located at the"]
Print["avoidance hubs Hmax and -Hmax in the areas shaded ", Red, " . "]
Print["The hubs Hmax and -Hmax are located at (α,δ) = ",
 Round[{αHmaxDegrees + 180, -δHmaxDegrees}], " and at ",
 Round[{αHmaxDegrees, δHmaxDegrees}], " , in degrees."]
Print["The angle between the sample's center and the closest avoidance hub Hmax is ",
 ArcCos[(-rHmax.sourceCenter)] (360./(2. π)), "°."]
Print["To guide the eye, two Great Circles are plotted, one through the sources' center and the
   avoidance hubs Hmax and -Hmax. The other connects the center of the sources' locations
   with the alignment hubs Hmin and -Hmin. The Great Circles are shaded Gray, ", Gray, " ."]
Print["The angle between the normals to the planes of the two great circles is ",
 θminMAXgreatcircles, "°."]
Print["Notes: Although somewhat obscured by the distortion needed to plot a
   sphere on a flat surface, the function η̄(H) is symmetric across diameters:
   Diametrically opposite points -H and H have the same alignment angle η̄(H)."]
```

Out[149]=

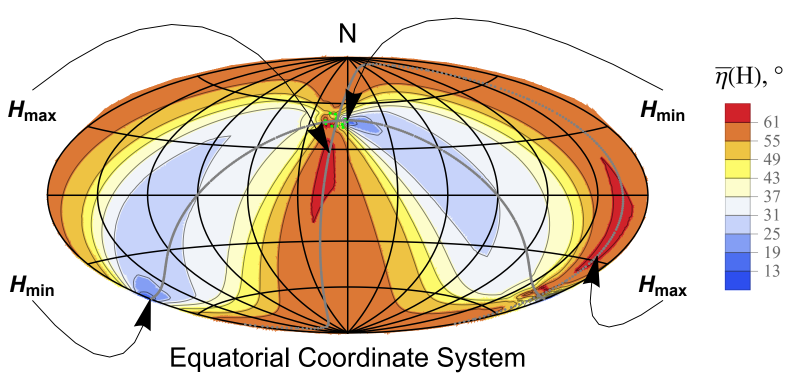


Figure 12: The alignment function $\overline{\eta}(H)$, Eq. (1). The map is centered on $(\alpha,\delta) = (180°,0°)$,

with α = 0° on the left and α = 360° on the right, Equatorial Coordinates.

The sources are located at the dots, shaded .

The smallest alignment angle is $\overline{\eta}_{min}$ = 10.8648°, located at the

alignment hubs H_{min} and $-H_{min}$ in the areas shaded .

The hubs H_{min} and $-H_{min}$ are located at (α,δ) = {180, 49} and {0, -49} , in degrees.

The angle between the sample's center and the closest alignment hub H_{min} is 5.34907°.

The largest avoidance angle is $\overline{\eta}_{max}$ = 62.6651°, located at the

avoidance hubs H_{max} and $-H_{max}$ in the areas shaded .

The hubs H_{max} and $-H_{max}$ are located at (α,δ) = {348, -28} and at {168, 28} , in degrees.

The angle between the sample's center and the closest avoidance hub H_{max} is 21.0509°.

To guide the eye, two Great Circles are plotted, one through the sources' center and the avoidance hubs H_{max} and $-H_{max}$. The other connects the center of the sources' locations with the alignment hubs H_{min} and $-H_{min}$. The Great Circles are shaded Gray, .

The angle between the normals to the planes of the two great circles is 104.78°.

Notes: Although somewhat obscured by the distortion needed to plot a sphere on a flat surface, the function $\overline{\eta}(H)$ is symmetric across diameters: Diametrically opposite points -H and H have the same alignment angle $\overline{\eta}(H)$.

In[164]:=
```
(* Local contour plot of the alignment function ηBar(H). *)
dηContourPlot = 6;(*, in degrees. *)
frameticks = {{{ {yH[135, 24.5], 30 °}, {yH[125, 60], 60 °}}, None},
   {{{xH180[150, 25], "10h"}, {xH180[180, 25], "12h"}, {xH180[210, 25], "14h"}},
    {{xH180[130, 62], StyleForm["Hmax", FontSize → 12, FontWeight -> "Bold"]},
     {xH180[188, 62], StyleForm["Hmin", FontSize → 12, FontWeight -> "Bold"]}}}};
listCPlocal = ListContourPlot[Union[xyηBarAitoffTable(*,{{xH180[αHminDegrees,δHminDegrees],
       yH180[αHminDegrees,δHminDegrees],ηBarMin*(360./(2.π))-1.0}},
     {{xH180[αHmaxDegrees,δHmaxDegrees],yH180[αHmaxDegrees,δHmaxDegrees],
       ηBarMax*(360./(2.π))+1.0}}*)], AspectRatio → 1/2,
   Contours → Table[η, {η, Floor[gridjηBarMin[[2]] * (360. / (2. π))] + 1,
       Ceiling[gridjηBarMax[[2]] * (360. / (2. π))] - 1, dηContourPlot}],
   ColorFunction → "TemperatureMap", PlotRange → {{xH180[135, 0], xH180[225, 0]},
     {yH180[180, 25], yH180[180, 62]}}, Axes -> False, Frame → True,
   FrameLabel → {"α", "δ", "Close-Up View"}, FrameTicks → frameticks,
   PlotLegends → Placed[BarLegend[Automatic, LegendMargins → {{0, 0}, {10, 5}},
       LegendLabel → "η̄(H), °", LabelStyle → {Plain, FontFamily → "Times"}], Right]];
```

In[167]:=
```
(*Plot polarization directions*)
rPlusψ[i_, d_] :=
 (rSrc[[i]] + d vψSrc[[i]]) / ((rSrc[[i]] + d vψSrc[[i]]).(rSrc[[i]] + d vψSrc[[i]]))^(1/2)
polarLines[d_] :=
 Table[Line[{{xH180[αFROMr[ rPlusψ[i, d]] (360./(2. π)), δFROMr[ rPlusψ[i, d]] (360./(2. π))],
     yH180[αFROMr[ rPlusψ[i, d]] (360./(2. π)), δFROMr[ rPlusψ[i, d]] (360./(2. π))]},
    {xH180[αFROMr[ rPlusψ[i, -d]] (360./(2. π)), δFROMr[ rPlusψ[i, -d]] (360./(2. π))],
     yH180[αFROMr[ rPlusψ[i, -d]] (360./(2. π)), δFROMr[ rPlusψ[i, -d]] (360./(2. π))]}}], {i, nSrc}]
```

In[169]:=
```
(*Construct the map of η̅(H).*)
mapOfηBarLocal =
  Show[{listCPlocal, Table[ParametricPlot[{xH180[α, δ], yH180[α, δ]}, {δ, 20, 90},
       PlotStyle → {Black, Thickness[0.002]}, PlotPoints → 60], {α, 120, 240, 30}],
     Table[ParametricPlot[{xH180[α, δ], yH180[α, δ]}, {α, 90, 270},
       PlotStyle → {Black, Thickness[0.002]}, PlotPoints → 60], {δ, 0, 90, 30}],
     Graphics[{PointSize[0.009], Black, {Thick, polarLines[0.03]}, (*Sources S:*)
       Green, PointSize[0.012], Point[ xyAitoffSources ], Gray,
       PointSize[0.005], Point[ xyAitoffGreatMin ], Point[ xyAitoffGreatMax ],
       Black, Text[StyleForm["X", FontSize → 12, FontWeight -> "Bold"],
        {xH180[αHminDegrees, δHminDegrees], yH180[αHminDegrees, δHminDegrees]}],
       Text[StyleForm["X", FontSize → 12, FontWeight -> "Bold"],
        {xH180[αHmaxDegrees, δHmaxDegrees], yH180[αHmaxDegrees, δHmaxDegrees]}],
       {Arrow[BezierCurve[{{-3.3, +1.2}, {-1.3, +3.0}, {xH180[αHmaxDegrees, δHmaxDegrees] - 0.01,
            yH180[αHmaxDegrees, δHmaxDegrees] + 0.03}}]]},
       {Arrow[BezierCurve[{{3.3, 1.2}, {0.3, 3.0}, {xH180[αHminDegrees, δHminDegrees] - 0.005,
            yH180[αHminDegrees, δHminDegrees] + 0.02}}]]}
         }]}, ImageSize → 0.9 × 432];
```

In[170]:=
```
mapOfηBarLocal
Print["Figure 13: Map of the alignment angle function
  η̅(H) in the neighborhood of the sources. The polarization directions display parallax,
  generally pointing toward the alignment hub Hmin.
   Note how close three of the sources are to the hub Hmin. "]
```

Out[170]=

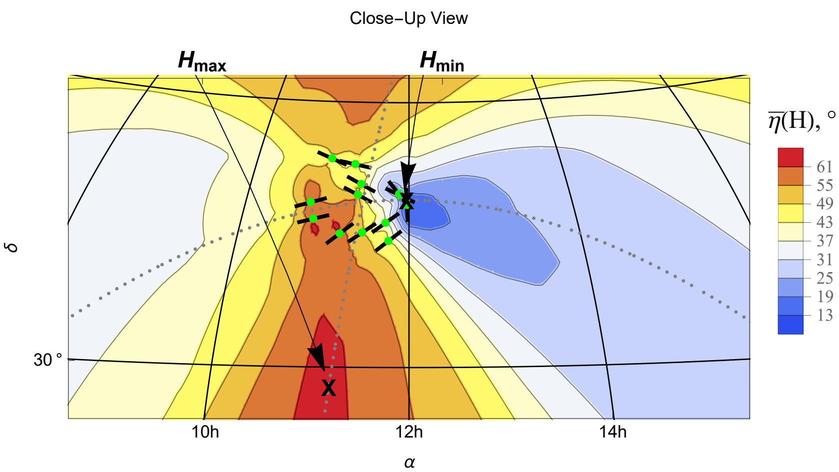


Figure 13: Map of the alignment angle function $\overline{\eta}(H)$ in the neighborhood of the sources. The polarization directions display parallax, generally pointing toward the alignment hub H_{min}.
Note how close three of the sources are to the hub H_{min}.

6. Uncertainty Runs

6a. Creating and Storing Uncertainty Runs

For each “uncertainty run”, the polarization direction ψ for each source is allowed to differ from the best value ψSrc by an amount $\delta\psi$ chosen according to a Gaussian distribution with a mean equal to the best value ψSrc and half-width $\sigma\psi$Src, $\psi = \psi$Src + $\delta\psi$. Both values ψSrc and $\sigma\psi$Src are taken from the JVAS1450 catalog.

The notebook .nb version generates new uncertainty runs. The pdf version uses old uncertainty runs that are uploaded from previously saved files that are not publically available. Thus both versions have some cells commented out: (* comments are not processed by Mathematica*).

Definitions:

rSrcxrGrid	unit vector $S_i \times H_j$, the cross product of the radial unit vector to source S_i with the radial unit vector to grid point H_j
nR	number of uncertainty runs
nRun	sequential index labeling the runs
ψData	table {nRun, ψ} of polarization directions $\psi = \psi$Src + $\delta\psi$ for each run
runData	collection of data to save from the uncertainty runs, see below for content list
nRunPrint	dummy index controlling when current TimeUsed and MemoryInUse are printed
ψSrcU	the polarization direction ψ for the run.
rSrcxψSrc	unit vector, $S_i \times \psi_i$, cross product of the radial vector S_i to the source with the vector $\hat{v}_\psi$ in the direction of the polariza-

tion

jηBarToGridU	{j, $\overline{\eta}(H_j)$}, where j is the index for the grid point H_j and $\overline{\eta}(H_j)$ is the alignment angle function, (1), at H_j
sortjηBarToGridU	sort {j, $\overline{\eta}(H_j)$}, with the smaller angle $\overline{\eta}$(H) first.
jηBarMinU	{j,$\overline{\eta}$(H)} for the smallest value of $\overline{\eta}$(H) , best alignment
jηBarMaxU	{j,$\overline{\eta}$(H)}, for the largest value of $\overline{\eta}$(H) , most avoided
ηmin$\alpha\delta$HU	off-grid local min data {$\overline{\eta}_{\min}$, {α,δ} at $H_{\min}$}
ηmax$\alpha\delta$HU	off-grid local max data {$\overline{\eta}_{\max}$, {α,δ} at $H_{\max}$}
funcDataU	off-grid, superior values of {nRun, ηmin$\alpha\delta$HU, ηmax$\alpha\delta$HU} collected results
HminαfunDataU	values of $\alpha = \alpha$ for hub $H_{\min}$ from uncertainty runs, alignment
HminδfunDataU	values of $\delta = \delta$ for hub $H_{\min}$ from uncertainty runs, alignment
HmaxαfunDataU	values of $\alpha = \alpha$ for hub $H_{\max}$ from uncertainty runs, avoidance
HmaxδfunDataU	values of $\delta = \delta$ for hub $H_{\max}$ from uncertainty runs, avoidance

Tables:

ψData	entries: 1. Run # 2. ψSrcU, list of polarization position angles ψ
gridDataUn	on-grid, entries: 1. Run # 2. {$\overline{\eta}_{\min}$, {α,δ} at $H_{\min}$} 3. {$\overline{\eta}_{\max}$, {α,δ} at $H_{\max}$}
funcDataU	off-grid, (better) entries: 1. Run # 2. {$\overline{\eta}_{\min}$, {α,δ} at $H_{\min}$} 3. {$\overline{\eta}_{\max}$, {α,δ} at $H_{\max}$}

To generate your own Uncertainty Runs:

First calculate “rSrcxrGrid” and then evaluate the “For” statement in the following two cells.

One can save the results with the “Put[]” statements.

Once saved, there is no need to repeat the runs. Comment out the “rSrcxrGrid” and “For” statements by enclosing them in (*comment brackets*).

The data can be retrieved with the “Get” statements.

In[172]:=
```
(*Remove comment marks, "(*" and "*)", below to generate your own tables. *)
```

In[173]:=

```
(* Evaluate this cell for the notebook .nb version *)
(*
nR=500;
t1=TimeUsed[];
rSrcxrGrid1=Table[ Cross[  rSrc[[i]],rGrid[[j]]  ] ,    {i,nSrc},{j,nGrid}];
(*first step: αw 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];

gridDataUn={};ψData={};funcDataU={};nRunPrint=0;
For[nRun=1,nRun≤nR,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+100];
     ψSrcU=Table[RandomVariate[NormalDistribution[ψSrc[[i]],σψSrc[[i]]]],{i,nSrc}];
  (*table of PPA angles ψ for the sources in region j0, in radians*)
  rSrcxψSrc = Table[ Sin[ψSrcU[[i]]]eNSrc[[i]]-
     Cos[ψSrcU[[i]]] eESrc[[i]],    {i,nSrc}];
  (*table of the cross product of rSrc and vector in direction of ψSrcU,
  a unit vector*)jηBarToGridU = 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}*) sortjηBarToGridU=Sort[jηBarToGridU,#1[[2]]<#2[[2]]&];
  (*jηBarToGridU, {j,ηj}, but sorted with the smallest alignment angles first
  *)
  jηBarMinU=sortjηBarToGridU[[1]]; (*  {j,ηj}, at the grid point Hj with minimum η̅*)
  jηBarMaxU=sortjηBarToGridU[[-1]]; (*  {j,ηj},
  at the grid point Hj with maximum η̅*)AppendTo[ψData,{nRun,ψSrcU}];
  AppendTo[gridDataUn,{nRun,{ jηBarMinU[[2]],
     {αGrid [ [ jηBarMinU[[1]]  ]],δGrid [[  jηBarMinU[[1]]  ]]}},
    { jηBarMaxU[[2]],{αGrid [[  jηBarMaxU[[1]]  ]],δGrid [[  jηBarMaxU[[1]]  ]]}}} ];
  (*collect discrete (on-grid) data*)
     ηminαδHU=FindMinimum[ηBarAtHwithAnyψ[αH,δH,ψData[[nRun,2]]],
    {{αH,gridDataUn[[nRun,2,2,1]]},{δH,gridDataUn[[nRun,2,2,2]]}}];
   ηmaxαδHU=
   FindMaximum[ηBarAtHwithAnyψ[αH,δH,ψData[[nRun,2]]],
    {{αH,gridDataUn[[nRun,3,2,1]]},{δH,gridDataUn[[nRun,3,2,2]]}}];
  AppendTo[funcDataU,{nRun,{ ηminαδHU[[1]],{αH,δH}/.ηminαδHU[[2]]},{  ηmaxαδHU[[1]],
     {αH,δH}/.ηmaxαδHU[[2]]}} ](*collect continuous (function-based) data*)   ]
 t2=TimeUsed[];
Print["Time used to compute ψData, gridDataUn, and funcDataU: t2 - t1 = ",t2-t1]
*)
```

Hint: You can save memory if you do not get the “ψData”. The table ψData is needed to reconstruct the exact values of the gridDataUn table, but it is not needed in any following calculation.

In[174]:=
```
SetDirectory[homeDirectory];
(*Save a new data file*)
(*
Put[ψData,"20211005PsiDataUqsoClump2U4000.dat"  ]
*)
(*
Put[gridDataUn,"20211005gridDataUnqsoClump2U4000.dat"  ]
*)
(*
Put[funcDataU,"20211005funcDataQSON13Un10000b.dat"  ]
*)
```

Hint: Saving “gridDataUn” to a file avoids the time it takes to complete the “For” statement. Make the above “For” statement into a remark so that it doesn’t evaluate.

In[175]:=
```
 SetDirectory[homeDirectory];
(*Retrieve an old data file*)
(*
ψData4000=Get["20211004PsiDataUqsoClump2U4000.dat"];
ψData6000=Get["20210928PsiDataUqsoClump2U6000.dat"];
*)
(*
gridDataUn4000=Get["20211004gridDataUnqsoClump2U4000.dat"];
gridDataUn6000=Get["20210928runDataUqsoClump2U6000.dat"];
*)
(*Get the funcDataU file for the pdf version:*)

funcDataU = Get["20211005funcDataQSON13Un10000a.dat"];
```

In[177]:=
```
(*If needed, edit the following to collect data files together.*)
(*
ψData=Join[ψData4000,ψData6000];
Length[ψData]
  ψData[[1]]
  gridDataUn=Join[gridDataUn4000,gridDataUn6000];
nR=Length[gridDataUn]
  gridDataUn[[1]]
*)
```

In[178]:=
```
(*nR may not be previously defined, depending on what cells have been processed.*)
(*Define nR for the pdf version:*)

nR = Length[funcDataU]
```

Out[178]=
```
10 000
```

In[179]:=
```
(*Define quantities based on the function continuous results. The
  continuous results should be better than the on-grid quantities.*)
ηBarMinfunDataU = Table[funcDataU[[i1, 2, 1]] , {i1, Length[funcDataU]}];
ηBarMaxfunDataU = Table[funcDataU[[i1, 3, 1]] , {i1, Length[funcDataU]}];
HminαfunDataU = Table[ funcDataU[[i1, 2, 2, 1]] , {i1, Length[funcDataU]}];
HminδfunDataU = Table[funcDataU[[i1, 2, 2, 2]], {i1, Length[funcDataU]}];
HminαδfunDataU =
  Table[{funcDataU[[i1, 2, 2, 1]], funcDataU[[i1, 2, 2, 2]]}, {i1, Length[funcDataU]}];
HmaxαfunDataU = Table[ funcDataU[[i1, 3, 2, 1]] , {i1, Length[funcDataU]}];
HmaxδfunDataU = Table[funcDataU[[i1, 3, 2, 2]], {i1, Length[funcDataU]}];
HmaxαδfunDataU =
  Table[{funcDataU[[i1, 3, 2, 1]], funcDataU[[i1, 3, 2, 2]]}, {i1, Length[funcDataU]}];
```

In[430]:=
```
ListPlot[{HminαδfunDataU, HmaxαδfunDataU}, PlotRange → All,
 PlotStyle → {{Blue, PointSize[0.01]}, {Red, PointSize[0.01]}},
 PlotLabel → "The hubs from the uncertainty runs", AxesLabel → {"α (rad)", "δ (rad)"}]
Print["Figure 14: Uncertainty run hubs. The alignment hubs H_min are in blue, ",
 Blue, " The avoidance hubs H_max are in ", Red,
 ". Symmetry across a diameter means there are hubs
   diametrically opposed to these. Including any diametrically
   opposed hubs would ruin the statistical calculations for hubs."]
```

Out[430]=

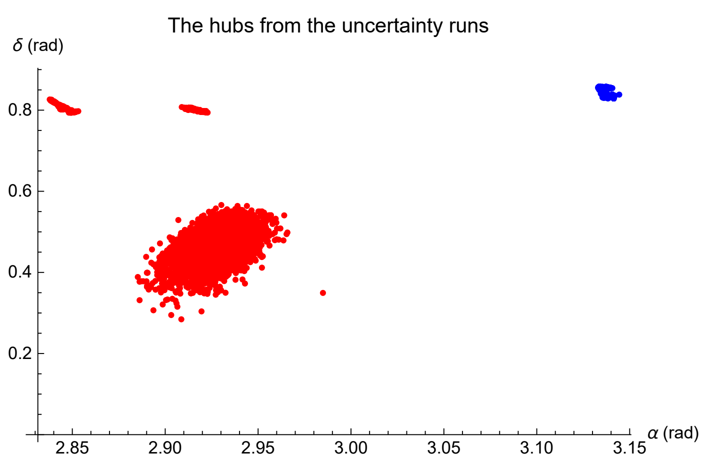


Figure 14: Uncertainty run hubs. The alignment hubs H_{min} are in blue, ■ The avoidance hubs H_{max} are in ■ . Symmetry across a diameter means there are hubs diametrically opposed to these. Including any diametrically opposed hubs would ruin the statistical calculations for hubs.

6b. The Effects of Uncertainty on the Smallest Alignment Angle $\overline{\eta}_{min}$

This section fits a Gaussian distribution to the $\overline{\eta}_{min}$ from the uncertainty runs.

Definitions

sortηBarMin	sort the list of $\overline{\eta}_{min}$ from the uncertainty runs
η0minU	estimated mean of the Gaussian fit
σminU	estimated half-width of the Gaussian fit
hlminU0, hlminU	histogram {η, bin height} tables needed to set up the NonlinearModelFit

nlmminU	non-linear model fit of a Gaussian to the $\overline{\eta}_{\min}$ histogram
showNLMB	plot of Gaussian and histogram
pTableNLMminU	table of parameter attributes, including standard error

$\sigma\eta$BarminUFit, ηBarminUFit - half-width, and mean of the Gaussian fit

In[189]:=
```
Print["The number of uncertainty runs is ", Length[funcDataU], "."]
```

The number of uncertainty runs is 10000.

In[190]:=
```
sortηBarMinU = Sort[ηBarMinfunDataU];
η0minU = mean[ηBarMinfunDataU ]; (*Guess the mean for the Gaussian. *)
σminU = stanDev[ηBarMinfunDataU ];(*Guess the half-width.*)
hlminU0 = HistogramList[sortηBarMinU, {η0minU - 5 σminU, η0minU + 5 σminU, 0.4 σminU}];
hlminU = Table[{(1/2) (hlminU0[[1, i1]] + hlminU0[[1, i1 + 1]]), hlminU0[[2, i1]]},
   {i1, Length[ hlminU0[[2]] ]}];
nlmminU = NonlinearModelFit[hlminU, a Exp[- (1/2.) ((x - x0) / b)^2],
   {{a, Length[sortηBarMinU/6]}, {b, σminU}, {x0, η0minU}}, x];(*x is ηBarMin*)
```

In[195]:=
```
pTableNLMminU = nlmminU["ParameterTable"]
{σηBarminUFit, ηBarminUFit} = {b, x0} /. nlmminU["BestFitParameters"];(*radians*)
```

Out[195]=

	Estimate	Standard Error	t-Statistic	P-Value
a	1580.79	8.78314	179.98	2.36808×10^{-36}
b	0.0187055	0.000120009	155.867	5.59355×10^{-35}
x0	0.198773	0.000120009	1656.32	1.4834×10^{-57}

In[197]:=
```
showNLMB = Show[{Histogram[sortηBarMinU, {η0minU - 5 σminU, η0minU + 5 σminU, 0.4 σminU},
     PlotLabel → "Uncertainty run η̄min ", AxesLabel → {"η̄min, radians", "ΔR"}],
   Plot[Normal[nlmminU], {x, η0minU - 5 σminU, η0minU + 5 σminU}, PlotLabel → "η̄min"],
   ListPlot[hlminU, PlotLabel → "η̄min"] }];
```

In[198]:=
```
showNLMB
Print["Figure 15: The Gaussian fit to the alignment angle η̄min histogram. The height
   is the number of runs ΔR in each bin. Note how nicely symmetric this is."]
Print["The total number of runs is R = Σ(ΔR) = ", Length[funcDataU], "."]
```

Out[198]=

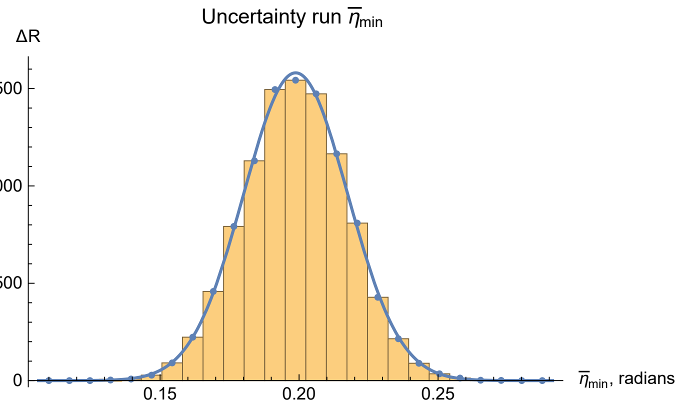


Figure 15: The Gaussian fit to the alignment angle $\bar{\eta}_{min}$ histogram. The height is the number of runs ΔR in each bin. Note how nicely symmetric this is.

The total number of runs is R = Σ(ΔR) = 10 000.

6c. The Effects of Uncertainty on the Largest Avoidance Angle $\bar{\eta}_{max}$

This section fits a Gaussian distribution to the $\bar{\eta}_{max}$ returned by the uncertainty runs.

Definitions: Similar to the definitions in Sec. 6b.

In[201]:=
```
sortηBarMaxU = Sort[ηBarMaxfunDataU];
η0maxU = mean[ηBarMaxfunDataU]; (*Guess the mean for the Gaussian. *)
σmaxU = stanDev[ηBarMaxfunDataU];(*Guess the half-width.*)
histogramrangemaxU = {η0maxU - 5 σmaxU, η0maxU + 5 σmaxU, 0.4 σmaxU};
hl0maxU = HistogramList[sortηBarMaxU, histogramrangemaxU];
hlmaxU = Table[{(1/2) (hl0maxU[[1, i1]] + hl0maxU[[1, i1 + 1]]), hl0maxU[[2, i1]]},
   {i1, Length[ hl0maxU[[2]] ]}];
nlmmaxU = NonlinearModelFit[hlmaxU, a Exp[-(1/2.) ((x - x0)/b)^2],
   {{a, 300.}, {b, σmaxU}, {x0, η0maxU}}, x];(*x is ηBarmaxU *)
```

$$\texttt{nlmBmaxU = NonlinearModelFit}\left[\texttt{hlmaxU, }\left\{a\left(1+e^{-4\frac{(x-x0+b)}{b}}\right)^{-1}\texttt{Exp}\left[-\frac{1}{2.}\left(\frac{x-x0}{b}\right)^{2}\right]\texttt{(*,b>0*)}\right\},\left\{\left\{a,\frac{nR}{12}\right\},\{b,\sigma maxU\},\{x0,\eta 0maxU\}\right\},x\right];$$

In[208]:=
```
pTableNLMmaxU = nlmBmaxU["ParameterTable"]
{σηBarmaxFitU, ηBarmaxFitU} =
  ParametersNLMmaxU = {b, x0} /. nlmBmaxU["BestFitParameters"];(*radians*)
```

Out[208]=

	Estimate	Standard Error	t-Statistic	P-Value
a	1559.75	17.1284	91.0624	7.49644×10^{-30}
b	0.0168288	0.000206563	81.4703	8.61619×10^{-29}
x0	1.10294	0.000172835	6381.43	1.92418×10^{-70}

```
In[210]:= showNLMmaxU = Show[{Histogram[sortηBarMaxU,
      histogramrangemaxU, PlotLabel → "η̅max", AxesLabel → {"η̅max, radians", "ΔR"}],
     Plot[Normal[nlmBmaxU], {x, η0maxU - 5 σmaxU, η0maxU + 5 σmaxU}, PlotLabel → "η̅max"],
     ListPlot[hlmaxU, PlotLabel → "η̅max"]}];

In[211]:= showNLMmaxU
Print[
 "Figure 16: The Non-Gaussian fit to the avoidance angle η̅max histogram. Each bin has a
   height equal to the number of runs ΔR in the bin. This graph slants like a random
   run distribution, i.e. away from η = π/4. See the random
   run discussions in Part I above and below in Sec. 7."]
```

Out[211]=

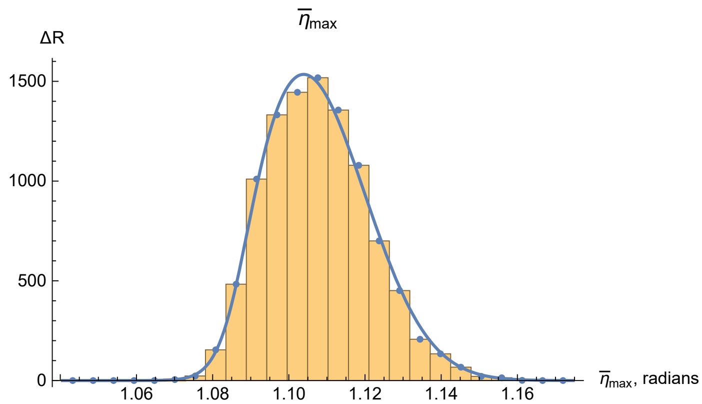


Figure 16: The Non-Gaussian fit to the avoidance angle $\bar{\eta}_{max}$ histogram. Each bin has a height equal to the number of runs ΔR in the bin. This graph slants like a random run distribution, *i.e.* away from $\eta = \pi/4$. See the random run discussions in Part I above and below in Sec. 7.

6d. The Effects of Uncertainty on the Locations (α,δ) of the Alignment Hubs H_{min}

Each uncertainty run returns an alignment hub H_{min}. In this section, we investigate the distribution of the locations the alignment Hubs H_{min}.

There are two hubs, H_{min} and $-H_{min}$ for each uncertainty run, by the symmetry across a diameter. So we collect the data together by moving the $-H_{min}$ hubs across a diameter to join the H_{min} hubs. See Fig. 14.

```
In[213]:= sortHminαδfunDataU = Sort[Union[HminαδfunDataU]];
lpHminU =
  ListPlot[Union[HminαδfunDataU], PlotRange → All, PlotStyle → {Blue, PointSize[0.01]},
   PlotLabel → "The alignment hubs from the uncertainty runs",
   AxesLabel → {"α (rad)", "δ (rad)"}];
```

In[215]:=
```
sortHminα = Sort[HminαfunDataU];
x0Hmin = mean[HminαfunDataU ];(*Guess the mean for the Gaussian. *)
dx0Hmin = stanDev[HminαfunDataU ];(*Guess the half-width.*)
histogramrangeRAHminU = {x0Hmin - 5 dx0Hmin, x0Hmin + 5 dx0Hmin, 0.4 dx0Hmin};
hl0xHmin = HistogramList[sortHminα, histogramrangeRAHminU];
hlxHmin = Table[{(1/2) (hl0xHmin[[1, i1]] + hl0xHmin[[1, i1 + 1]]), hl0xHmin[[2, i1]]},
   {i1, Length[ hl0xHmin[[2]] ]}];
nlmxHmin = NonlinearModelFit[hlxHmin, a Exp[- (1/2.) ((x - x0) / b)^2],
   {{a, Length[sortHminα / 6]}, {b, dx0Hmin}, {x0, x0Hmin}}, x];(*x is Hminα*)
```

In[221]:=
```
pTablenlmxHmin = nlmxHmin["ParameterTable"]
{σHminαFit, HminαFit} = ParametersnlmxHmin = {b, x0} /. nlmxHmin["BestFitParameters"];
(*radians*)
Normal[nlmxHmin]
expOfnlmxHmin[x_] := - (1/2.) ((x - x0) / b)^2 /. nlmxHmin["BestFitParameters"]
expOfnlmxHmin[x]
```

Out[221]=

	Estimate	Standard Error	t-Statistic	P-Value
a	4684.95	588.845	7.95618	6.46607×10^{-8}
b	0.000479938	0.0000791483	6.06379	4.20082×10^{-6}
x0	3.13413	0.000130467	24022.4	4.15789×10^{-83}

Out[223]= $4684.95\, e^{-2.1707\times10^{6}\,(-3.13413+x)^{2}}$

Out[225]= $-2.1707\times10^{6}\,(-3.13413+x)^{2}$

In[226]:=
```
shownlmxHmin = Show[{Histogram[sortHminα, histogramrangeRAHminU,
      PlotLabel → "αHmin ", AxesLabel → {"αHmin, radians", "ΔR"}, PlotRange → All],
    Plot[Normal[nlmxHmin], {x, 3.12, 3.145}, PlotRange → All, PlotLabel → "αHmin"],
    ListPlot[hlxHmin, PlotLabel → "αHmin"] }];
```

In[227]:=
```
sortHminδ = Sort[HminδfunDataU];
y0Hmin = mean[HminδfunDataU ];(*Guess the mean for the Gaussian. *)
dy0Hmin = stanDev[HminδfunDataU ];(*Guess the half-width.*)
histogramrangeDecHminU = {y0Hmin - 5 dy0Hmin, y0Hmin + 5 dy0Hmin, 0.4 dy0Hmin};
hl0yHmin = HistogramList[sortHminδ, histogramrangeDecHminU];
hlyHmin = Table[{(1/2) (hl0yHmin[[1, i1]] + hl0yHmin[[1, i1 + 1]]), hl0yHmin[[2, i1]]},
   {i1, Length[ hl0yHmin[[2]] ]}];
nlmyHmin = NonlinearModelFit[hlyHmin, a Exp[- (1/2.) ((y - y0) / b)^2],
   {{a, Length[sortHminδ / 6]}, {b, dy0Hmin}, {y0, y0Hmin}}, y];(*y is Hminδ*)
```

In[233]:=
```
pTablenlmyHmin = nlmyHmin["ParameterTable"]
{σHminδFit, HminδFit} = ParametersnlmyHmin = {b, y0} /. nlmyHmin["BestFitParameters"];
(*radians*)
Normal[nlmyHmin]
expOfnlmyHmin[y_] := - (1/2.) ((y - y0) / b)^2 /. nlmyHmin["BestFitParameters"]
expOfnlmyHmin[y]
```

Out[233]=

	Estimate	Standard Error	t-Statistic	P-Value
a	7835.08	122.439	63.9917	1.71076×10^{-26}
b	0.000909269	0.0000386478	23.5271	4.36022×10^{-17}
y0	0.855277	0.0000977768	8747.24	1.86759×10^{-73}

Out[235]= $7835.08\, e^{-604763. (-0.855277+y)^2}$

Out[237]= $-604763. (-0.855277 + y)^2$

In[238]:=
```
shownlmyHmin = Show[{Histogram[sortHminδ, histogramrangeDecHminU,
     PlotLabel → "δHmin ", AxesLabel → {"δHmin, radians", "ΔR"}, PlotRange → All],
    Plot[Normal[nlmyHmin], {y, 0.82, 0.88}, PlotRange → All, PlotLabel → "δHmin"],
    ListPlot[hlyHmin, PlotLabel → "δHmin"] }];
```

General: Exp[−752.569] is too small to represent as a normalized machine number; precision may be lost.

In[239]:=
```
GraphicsRow[{shownlmxHmin, shownlmyHmin}]
Print["Figure 17: The Gaussian fits to the Hmin RA and DEC
   histograms, where the height is the number of runs ΔR in each bin. "]
Print["In both graphs, the total number of runs is R = Σ(ΔR) = ", Length[funcDataU], "."]
```

Out[239]=

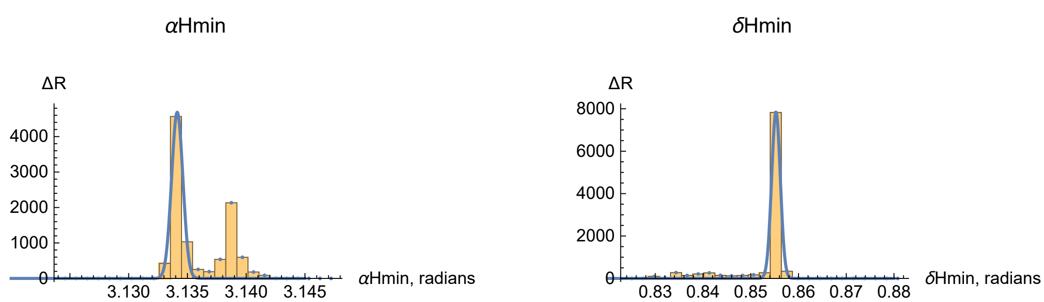


Figure 17: The Gaussian fits to the Hmin RA and DEC
  histograms, where the height is the number of runs ΔR in each bin.

In both graphs, the total number of runs is R = Σ(ΔR) = 10 000.

In[242]:=
```
expoHminU[x_, y_] := - (expOfnlmxHmin[x] + expOfnlmyHmin[y])
Print["The exponent of the probability distribution for
   Hmin, i.e. the negative log of the distribution: ", expoHminU[α, δ]]
```

The exponent of the probability distribution for H_{min}, *i.e.* the negative log of the distribution:
$2.1707 \times 10^6 (-3.13413 + \alpha)^2 + 604763. (-0.855277 + \delta)^2$

In[244]:=
```
Plot3D[{expoHminU[x, y], 0.5}, {x, x0 - 0.0010, x0 + 0.0010} /. nlmxHmin["BestFitParameters"],
 {y, y0 - 0.0015, y0 + 0.0015} /. nlmyHmin["BestFitParameters"],
 PlotLabel → "Negative log of the probability of (α,δ) for H_min",
 AxesLabel → {"α (rad)", "δ (rad)"}]
Print["Figure 18: The negative log of the likelihood of (RA,dec) for H_min, as a
   function of RA and dec. Where the likelihood is down by a factor e^-1/2, the
   negative log is 0.5 and that defines the half-width σ of the distribution."]
```

Out[244]=

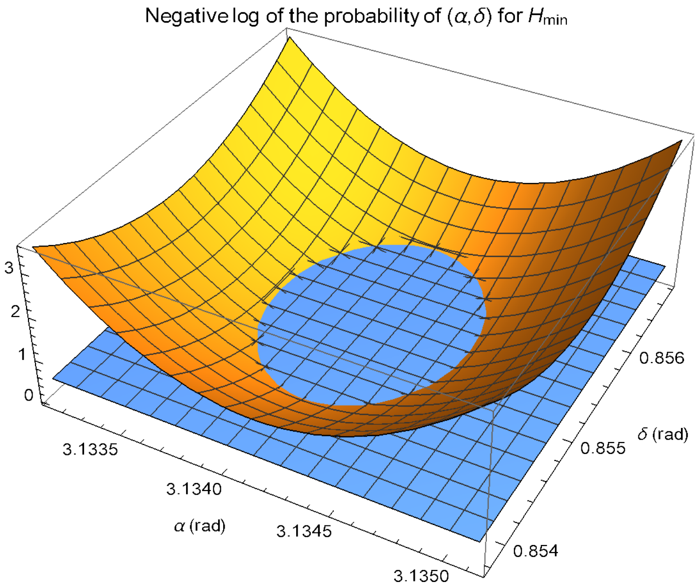


Figure 18: The negative log of the likelihood of (RA,dec) for H_{min}, as a function of RA and dec. Where the likelihood is down by a factor $e^{-1/2}$, the negative log is 0.5 and that defines the half-width σ of the distribution.

In[246]:=
```
(*Find the curve for the intersection in Fig. 18*)
frθHmin[r_, θ_] :=
 Simplify[(expoHminU[x, y]) - 0.5 /. {x → HminαFit + r Cos[θ], y → HminδFit + r Sin[θ]}]
frθHmin[r, θ];
solverHminθ[θ_] := Solve[frθHmin[r, θ] == 0, r];
solverHminθ[θ];
rHminθ[θ_] := Abs[r /. solverHminθ[θ][[2]]]
rHminθ[θ];
rHminθ[0.8];
Plot[rHminθ[θ], {θ, 0, 2. π}];
```

Solve: Solve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result.

Solve: Solve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result.

In[254]:=
```
Show[{lpHminU, ParametricPlot[{HminαFit + rHminθ[θ] Cos[θ], HminδFit + rHminθ[θ] Sin[θ]},
    {θ, 0, 2. π}, PlotStyle → Orange, PlotRange → All(*{{3.12,3.14},{0.84,0.90}}*)]}]
Print["Figure 19: All of the alignment hubs Hmin from uncertainty runs. The ellipse
   encloses the most likely locations of the hubs. Symmetry across diameters
   means there is another set diametrically opposite those displayed here."]
```

Out[254]=

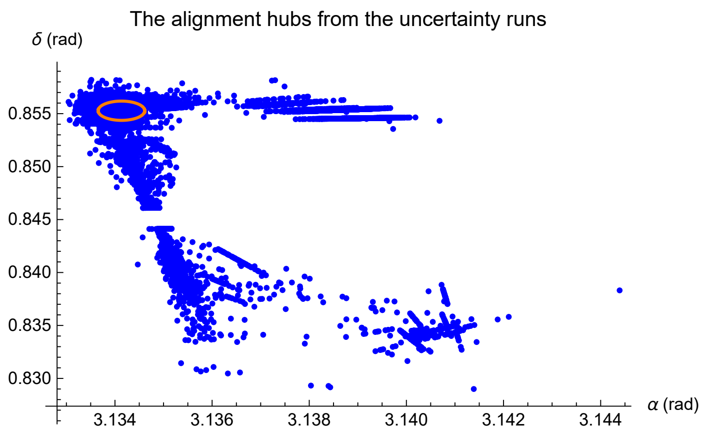


Figure 19: All of the alignment hubs H_{min} from uncertainty runs.
The ellipse encloses the most likely locations of the hubs. Symmetry across
diameters means there is another set diametrically opposite those displayed here.

6e. The Effects of Uncertainty on the Locations (α,δ) of the Avoidance Hubs H_{max}

Each uncertainty run returns an avoidance hub H_{max}. In this section, we investigate the distribution of the locations the avoidance hubs H_{max}.

There are two hubs, H_{max} and $-H_{max}$ for each uncertainty run, by the symmetry across a diameter. So we collect all the hubs together by moving the $-H_{max}$ hubs across a diameter to join the H_{max} hubs. See Fig. 14.

In[256]:=
```
(*Check that 0° ≤ α < 180° and -90° ≤ δ < 90° *)
sortHmaxαδfunDataU = Sort[Union[Hmaxαδfun DataU] (360./2. π)];
lpHmaxU =
  ListPlot[Union[HmaxαδfunDataU], PlotRange → All, PlotStyle → {Red, PointSize[0.01]},
   PlotLabel → "The avoidance hubs from the uncertainty runs",
   AxesLabel → {"α (rad)", "δ (rad)"}];
```

In[258]:=
```
sortHmaxα = Sort[HmaxαfunDataU];
x0Hmax = mean[HmaxαfunDataU];(*Guess the mean for the Gaussian. *)
dx0Hmax = stanDev[HmaxαfunDataU];(*Guess the half-width.*)
histogramrange = {x0Hmax - 5 dx0Hmax, x0Hmax + 5 dx0Hmax, dx0Hmax};
hl0xHmax = HistogramList[sortHmaxα, histogramrange];
hlxHmax = Table[{(1/2) (hl0xHmax[[1, i1]] + hl0xHmax[[1, i1 + 1]]), hl0xHmax[[2, i1]]},
  {i1, Length[ hl0xHmax[[2]] ]}];
nlmxHmax = NonlinearModelFit[hlxHmax, a Exp[- (1/2.) ((x - x0) / b)^2],
  {{a, Length[sortHmaxα / 6]}, {b, dx0Hmax}, {x0, x0Hmax}}, x];(*x is Hmaxα*)
```

In[264]:=
```
pTablenlmxHmax = nlmxHmax["ParameterTable"]
{σHmaxαFit, HmaxαFit} = ParametersnlmxHmax = {b, x0} /. nlmxHmax["BestFitParameters"];
(*radians*)
Normal[nlmxHmax]
expOfnlmxHmax[x_] := -(1/2.) ((x - x0)/b)^2 /. nlmxHmax["BestFitParameters"]
expOfnlmxHmax[x]
```

Out[264]=

	Estimate	Standard Error	t-Statistic	P-Value
a	3279.85	1349.65	2.43016	0.0454102
b	0.0508147	0.0241455	2.10452	0.0733793
x0	2.90428	0.0241444	120.288	7.23729×10^{-13}

Out[266]= $3279.85\, e^{-193.638\,(-2.90428+x)^2}$

Out[268]= $-193.638\,(-2.90428+x)^2$

In[269]:=
```
shownlmxHmax = Show[{Histogram[sortHmaxα, histogramrange,
     PlotLabel → "αHmax ", AxesLabel → {"αHmax, radians", "ΔR"}, PlotRange → All],
    Plot[Normal[nlmxHmax], {x, 2.7, 3.1}, PlotRange → All, PlotLabel → "αHmax"],
    ListPlot[hlxHmax, PlotLabel → "αHmax"] }];
```

In[270]:=
```
sortHmaxδ = Sort[HmaxδfunDataU];
y0Hmax = mean[HmaxδfunDataU ];(*Guess the mean for the Gaussian. *)
dy0Hmax = stanDev[HmaxδfunDataU ];(*Guess the half-width.*)
histogramrange = {y0Hmax - 5 dy0Hmax, y0Hmax + 5 dy0Hmax, 0.4 dy0Hmax};
hl0yHmax = HistogramList[sortHmaxδ, histogramrange];
hlyHmax = Table[{(1/2) (hl0yHmax[[1, i1]] + hl0yHmax[[1, i1 + 1]]), hl0yHmax[[2, i1]]},
   {i1, Length[ hl0yHmax[[2]] ]}];
nlmyHmax = NonlinearModelFit[hlyHmax, a Exp[-(1/2.) ((y - y0)/b)^2],
   {{a, Length[sortHmaxδ/6]}, {b, dy0Hmax}, {y0, y0Hmax}}, y];(*x is Hmaxδ*)
```

In[276]:=
```
pTablenlmyHmax = nlmyHmax["ParameterTable"]
{σHmaxδFit, HmaxδFit} = ParametersnlmyHmax = {b, y0} /. nlmyHmax["BestFitParameters"];
(*radians*)
Normal[nlmyHmax]
expOfnlmyHmax[y_] := -(1/2.) ((y - y0)/b)^2 /. nlmyHmax["BestFitParameters"]
expOfnlmyHmax[y]
```

Out[276]=

	Estimate	Standard Error	t-Statistic	P-Value
a	1252.9	520.149	2.40874	0.0248248
b	0.235369	0.112836	2.08594	0.0487919
y0	0.589584	0.11283	5.22543	0.0000305743

Out[278]= $1252.9\, e^{-9.02549\,(-0.589584+y)^2}$

Out[280]= $-9.02549\,(-0.589584+y)^2$

In[281]:=
```
shownlmyHmax = Show[{Histogram[sortHmaxδ, histogramrange,
     PlotLabel → "δHmax ", AxesLabel → {"δHmax, radians", "ΔR"}, PlotRange → All],
    Plot[Normal[nlmyHmax], {y, 0., 1.5}, PlotRange → All, PlotLabel → "δHmax"],
    ListPlot[hlyHmax, PlotLabel → "δHmax"] }];
GraphicsRow[{shownlmxHmax, shownlmyHmax}]
Print["Figure 20: The Gaussian fits to the Hmax RA and DEC
   histograms, where the height is the number of runs ΔR in each bin. "]
Print["In both graphs, the total number of runs is R = Σ(ΔR) = ", Length[funcDataU], "."]
```

Out[282]=

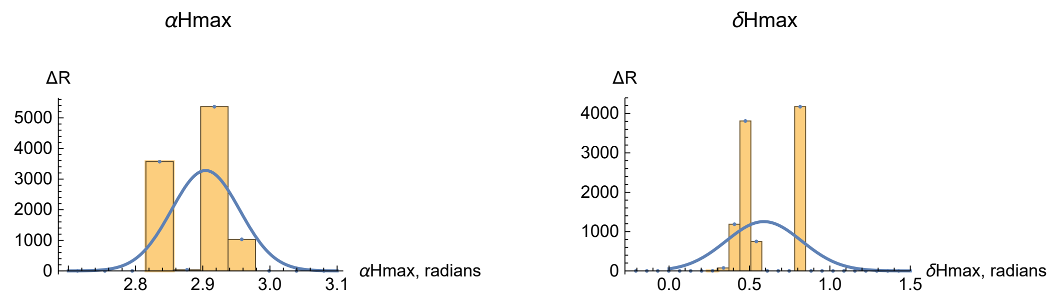


Figure 20: The Gaussian fits to the Hmax RA and DEC
  histograms, where the height is the number of runs ΔR in each bin.

In both graphs, the total number of runs is R = Σ(ΔR) = 10000.

In[285]:=
```
expoHmaxU[x_, y_] := - (expOfnlmxHmax[x] + expOfnlmyHmax[y])
Print["The exponent of the probability distribution for
   H_max, i.e. the negative log of the distribution: ", expoHmaxU[α, δ]]
```

The exponent of the probability distribution for H_{max}, *i.e.* the negative log of the distribution:
$193.638\ (-2.90428 + \alpha)^2 + 9.02549\ (-0.589584 + \delta)^2$

In[287]:=
```
Plot3D[{expoHmaxU[x, y], 0.5}, {x, x0 - 0.08, x0 + 0.08} /. nlmxHmax["BestFitParameters"],
 {y, y0 - 0.5, y0 + 0.5} /. nlmyHmax["BestFitParameters"],
 PlotLabel → "Negative log of the probability of (α,δ) for Hmax",
 AxesLabel → {"α (rad)", "δ (rad)"}]
Print["Figure 21: The negative log of the likelihood of (RA,dec) for Hmax, as a
   function of RA and dec. Where the likelihood is down by a factor e^-1/2, the
   negative log is 0.5 and that defines the half-width σ of the distribution."]
```

Out[287]=

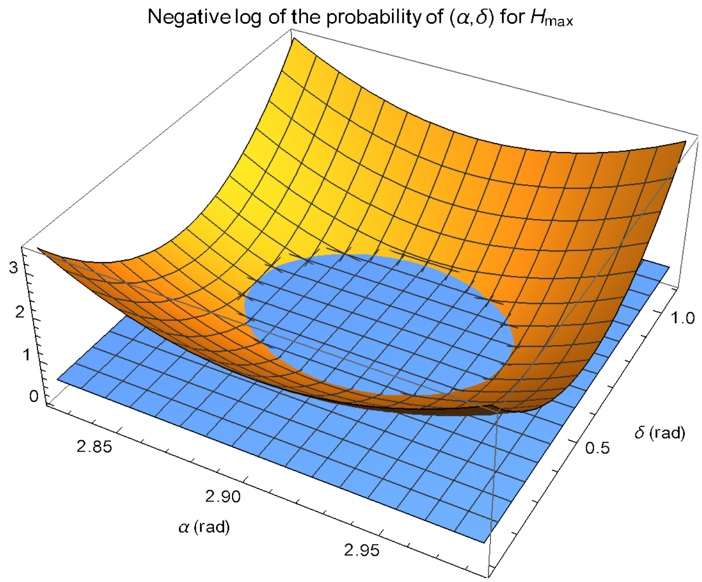


Figure 21: The negative log of the likelihood of (RA,dec) for H_{max}, as a function of RA and dec. Where the likelihood is down by a factor $e^{-1/2}$, the negative log is 0.5 and that defines the half-width σ of the distribution.

In[289]:=
```
(*Find the curve for the intersection in Fig. 21*)
frθHmax[r_, θ_] :=
 Simplify[(expoHmaxU[x, y]) - 0.5 /. {x → HmaxαFit + r Cos[θ], y → HmaxδFit + r Sin[θ]}]
frθHmax[r, θ];
solverHmaxθ[θ_] := Solve[frθHmax[r, θ] == 0, r];
solverHmaxθ[θ];
rHmaxθ[θ_] := Abs[r /. solverHmaxθ[θ][[2]]]
rHmaxθ[θ];
rHmaxθ[0.8];
Plot[rHmaxθ[θ], {θ, 0, 2. π}];
```

••• Solve: Solve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result.

••• Solve: Solve was unable to solve the system with inexact coefficients. The answer was obtained by solving a corresponding exact system and numericizing the result.

In[297]:=
```
Show[{lpHmaxU, ParametricPlot[{HmaxαFit + rHmaxθ[θ] Cos[θ], HmaxδFit + rHmaxθ[θ] Sin[θ]},
    {θ, 0, 2. π}, PlotStyle → Orange, PlotRange → All]}]
Print["Figure 22: Avoidance hubs Hmax from uncertainty runs. The ellipse
   encloses the most likely locations of the hubs. Symmetry across diameters
   means there is another set diametrically opposite those displayed here."]
```

Out[297]=

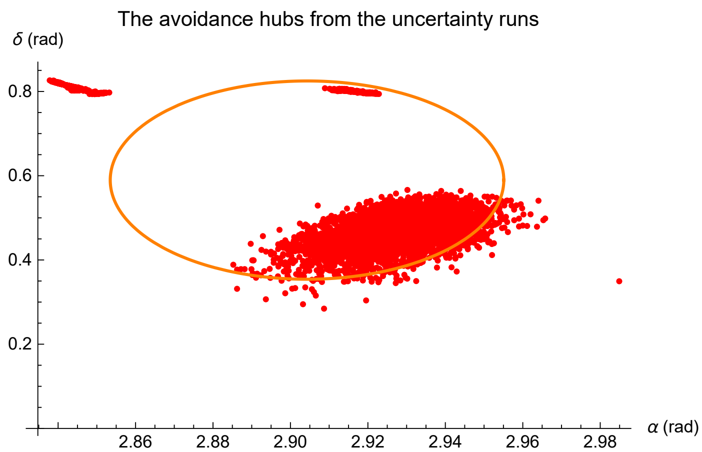


Figure 22: Avoidance hubs H_{max} from uncertainty runs. The ellipse encloses the most likely locations of the hubs. Symmetry across diameters means there is another set diametrically opposite those displayed here.

6f. The Effects of Uncertainty on the angle θ between the planes of the Sample to $H_{\min}$ Great Circle and the Sample to $H_{\max}$ Great Circle.

These are the Gray lines in Figs. 3, 4, 12, 13. However, in Sec. 7 below, we see that the avoidance angle $\overline{\eta}_{\mathbf{max}}$ is not significant, random ψ would be likely to yield a $\overline{\eta}_{\mathbf{max}}$ that is as large or larger. Also, we see a lot of scatter in Fig. 22 for the avoidance hubs $H_{\max}$. Conversely, the alignment angle $\overline{\eta}_{\mathbf{min}}$ is very significant and the alignment hubs collect in a tight formation. Compare the axes scales in Figs. 19 and 22. The Great Circle from the Sample to $H_{\max}$ is not well-defined. So the angle θ varies over a wide range.

Definitions:

"uRuns" prefix	results from the uncertainty runs
uRunsCrossMin	unit vector normal to the Great Circle connecting the center of the source region with the alignment hub $H_{\min}$
uRunsCrossMax	unit vector normal to the Great Circle connecting the center of the source region with the alignment hub $H_{\max}$
uRunsθminmaxUgreatcircles	angle between the two normals in degrees
sortθminmaxU	sort "uRunsθminmaxUgreatcircles", smallest θ first

See Definitions above in Secs. 6a,6b for other quantities below. There you should find similarly named quantities.

In[299]:=
```
uRunsCrossMin0 = Table[Cross[er[HminαfunDataU[[i]], HminδfunDataU[[i]]], sourceCenter],
    {i, Length[HminαfunDataU]}];
uRunsCrossMin = Table[uRunsCrossMin0[[i]] / (uRunsCrossMin0[[i]].uRunsCrossMin0[[i]])^(1/2.),
    {i, Length[HminαfunDataU]}];
uRunsCrossmaxU0 = Table[Cross[er[HmaxαfunDataU[[i]], HmaxδfunDataU[[i]]], sourceCenter],
    {i, Length[HmaxαfunDataU]}];
uRunsCrossmaxU = Table[ uRunsCrossmaxU0[[i]] /
     (uRunsCrossmaxU0[[i]].uRunsCrossmaxU0[[i]])^(1/2.), {i, Length[HmaxαfunDataU]}];
uRunsθminmaxUgreatcircles = Table[ArcCos[uRunsCrossmaxU[[i]].uRunsCrossMin[[i]]] (360./(2. π)),
    {i, Length[HmaxαfunDataU]}];
```

In[304]:=
```
(*Fit two peaks for θ:*)
sortθminmaxU = Sort[uRunsθminmaxUgreatcircles];
x0θ = mean[uRunsθminmaxUgreatcircles]; (*Guess the mean for the Gaussian. *)
dx0θ = 0.3 stanDev[uRunsθminmaxUgreatcircles ];(*Guess the half-width.*)
histogramrange = {x0θ - 5 dx0θ, x0θ + 5 dx0θ, 0.4 dx0θ};
hl0 = HistogramList[sortθminmaxU, histogramrange];
hl =
 Table[{(1/2) (hl0[[1, i1]] + hl0[[1, i1 + 1]]), hl0[[2, i1]]}, {i1, Length[ hl0[[2]] ]}];
nlmθ = NonlinearModelFit[hl, a3 Exp[- (1/2.) ((x - x03) / b3)^2] +
    a4 Exp[- (1/2.) ((x - x04) / b4)^2], {{a3, 2000.}, {b3, 5.}, {x03, 100.},
    {a4, 1000.}, {b4, 15.}, {x04, 155.}}, {x}];(*x is θminmaxU*)
```

In[310]:=
```
pTableNLMθ = nlmθ["ParameterTable"]
{dx0θminmaxUFit3, θminmaxUFit3, dx0θminmaxUFit4, θminmaxUFit4} =
  {b3, x03, b4, x04} /. nlmθ["BestFitParameters"];(*degrees*)
```

Out[310]=

	Estimate	Standard Error	t-Statistic	P-Value
a3	3858.83	306.185	12.6029	1.12711×10^{-10}
b3	−2.01426	0.196727	−10.2389	3.59334×10^{-9}
x03	105.092	0.157148	668.742	5.30798×10^{-43}
a4	1015.42	125.289	8.10463	1.38104×10^{-7}
b4	7.6131	2.48322	3.06581	0.00636138
x04	170.689	3.26776	52.2344	5.45207×10^{-22}

In[312]:=
```
showNLMθ = Show[{Histogram[sortθminmaxU, histogramrange,
      PlotLabel → "Angle θ between the Two Gray Great Circles in Figs. 3, 4, 12, 13.",
      AxesLabel → {"θ, degrees", "ΔR"}],
    Plot[Normal[nlmθ], {x, 0, 250}, PlotRange → All], ListPlot[hl] }];
```

··· General: Exp[−1360.92] is too small to represent as a normalized machine number; precision may be lost.

In[313]:=
```
showNLMΘ
Print["Figure 23: The Gaussian fit to the angle Θ histogram. We fit two angles Θ,
   corresponding to the two likely locations of the avoidance hubs Hmax. "]
```

Out[313]=

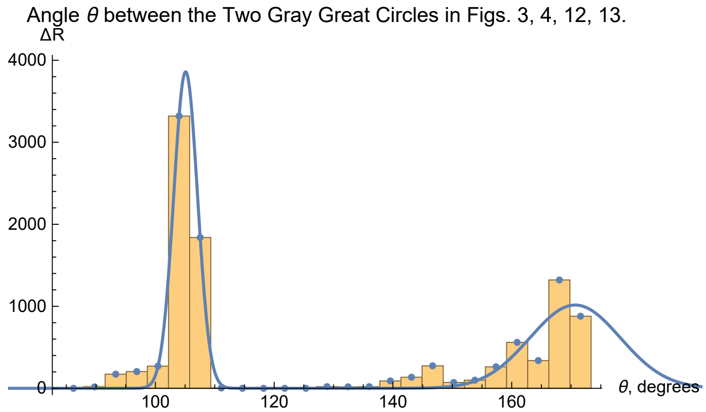


Figure 23: The Gaussian fit to the angle Θ histogram. We fit two angles Θ, corresponding to the two likely locations of the avoidance hubs H_{max}.

6g. Map of the Hubs for the Uncertainty Runs

In this subsection, we map the locations of the many alignment hubs H_{min} and the avoidance hubs H_{max} that are found in the uncertainty runs.

Definitions:

vψSrcBig, Small unit vectors, v($\psi \pm \sigma\psi$), large & small, the one-sigma range of polarization directions ψ

In[315]:=
```
(*The Aitoff coordinates for the hubs Hmin locations.*)
xyAitoffHminU = Table[{xH180[ HminαfunDataU[[n]] (360/(2π)),
     HminδfunDataU[[n]] (360/(2π)) ], yH180[  HminαfunDataU[[n]] (360/(2π)),
     HminδfunDataU[[n]] (360/(2π)) ]}, {n, Length[HminδfunDataU ]}];
```

In[316]:=
```
(*The Aitoff coordinates for the hubs Hmax locations.*)
xyAitoffHmaxU = Table[{xH180[  HmaxαfunDataU[[n]] (360/(2π)),
     HmaxδfunDataU[[n]] (360/(2π)) ], yH180[  HmaxαfunDataU[[n]] (360/(2π)),
     HmaxδfunDataU[[n]] (360/(2π)) ]}, {n, Length[HmaxδfunDataU ]}];
```

In[317]:=
```
(*The Aitoff coordinates for the hubs -Hmin locations.*)
xyAitoffOppositeHminU = Table[{xH180[If[0 ≤ HminαfunDataU [[n]] (360/(2π)) < +180,
      HminαfunDataU[[n]] (360/(2π)) + 180, If[360 > HminαfunDataU[[n]] (360/(2π)) > 180,
       HminαfunDataU[[n]] (360/(2π)) - 180]], -HminδfunDataU[[n]] (360/(2π)) ],
    yH180[ If[0 ≤ HminαfunDataU[[n]] (360/(2π)) < +180,
      HminαfunDataU[[n]] (360/(2π)) + 180, If[
       360 > HminαfunDataU[[n]] (360/(2π)) > 180, HminαfunDataU[[n]] (360/(2π)) - 180]],
     -HminδfunDataU[[n]] (360/(2π)) ]}, {n, Length[HminδfunDataU ]}];
```

In[318]:=
```
(*The Aitoff coordinates for the hubs -Hmax locations.*)
xyAitoffOppositeHmaxU = Table[{xH180[ If[0 ≤ HmaxαfunDataU[[n]] (360 / (2 π)) < +180,
      HmaxαfunDataU[[n]] (360 / (2 π)) + 180, If[360 > HmaxαfunDataU[[n]] (360 / (2 π)) > 180,
       HmaxαfunDataU[[n]] (360 / (2 π)) - 180]], -HmaxδfunDataU[[n]] (360 / (2 π)) ],
    yH180[ If[0 ≤ HmaxαfunDataU[[n]] (360 / (2 π)) < +180,
      HmaxαfunDataU[[n]] (360 / (2 π)) + 180, If[
       360 > HmaxαfunDataU[[n]] (360 / (2 π)) > 180, HmaxαfunDataU[[n]] (360 / (2 π)) - 180]],
     -HmaxδfunDataU[[n]] (360 / (2 π)) ]}, {n, Length[HmaxδfunDataU ]}];
```

In[319]:=
```
(* vψ unit vectors pointing along the polarization direction,
have an experimental uncertainty. These are their plus/minus values. *)
vψSrcBig = Table[Cos[ (ψSrc[[i]] + σψSrc[[i]]) ] eN[ αSrc[[i]], δSrc[[i]] ] +
    Sin[ (ψSrc[[i]] + σψSrc[[i]]) ] eE[ αSrc[[i]], δSrc[[i]] ], {i, nSrc}];
vψSrcSmall = Table[Cos[ (ψSrc[[i]] - σψSrc[[i]]) ] eN[ αSrc[[i]], δSrc[[i]] ] +
    Sin[ (ψSrc[[i]] - σψSrc[[i]]) ] eE[ αSrc[[i]], δSrc[[i]] ], {i, nSrc}];
```

In[321]:=
```
(*Plot polarization direction Uncertainty in Sec. 6*)
rPlusψBig[i_, d_] := (rSrc[[i]] + d vψSrcBig[[i]]) /
  ((rSrc[[i]] + d vψSrcBig[[i]]).(rSrc[[i]] + d vψSrcBig[[i]]))^(1/2)
polarLinesBig[d_] := Table[
  Line[{{xH180[αFROMr[ rPlusψBig[i, d]] (360./(2. π)), δFROMr[ rPlusψBig[i, d]] (360./(2. π))],
     yH180[αFROMr[ rPlusψBig[i, d]] (360./(2. π)), δFROMr[ rPlusψBig[i, d]] (360./(2. π))]},
    {xH180[αFROMr[ rPlusψBig[i, -d]] (360./(2. π)), δFROMr[ rPlusψBig[i, -d]] (360./(2. π))], yH180[
      αFROMr[ rPlusψBig[i, -d]] (360./(2. π)), δFROMr[ rPlusψBig[i, -d]] (360./(2. π))]}}], {i, nSrc}]
```

In[323]:=
```
(*Plot polarization direction Uncertainty in Sec. 6*)
rPlusψSmall[i_, d_] := (rSrc[[i]] + d vψSrcSmall[[i]]) /
  ((rSrc[[i]] + d vψSrcSmall[[i]]).(rSrc[[i]] + d vψSrcSmall[[i]]))^(1/2)
polarLinesSmall[d_] := Table[
  Line[{{xH180[αFROMr[ rPlusψSmall[i, d]] (360./(2. π)), δFROMr[ rPlusψSmall[i, d]] (360./(2. π))],
     yH180[αFROMr[ rPlusψSmall[i, d]] (360./(2. π)), δFROMr[ rPlusψSmall[i, d]] (360./(2. π))]},
    {xH180[αFROMr[ rPlusψSmall[i, -d]] (360./(2. π)), δFROMr[ rPlusψSmall[i, -d]] (360./(2. π))],
     yH180[αFROMr[ rPlusψSmall[i, -d]] (360./(2. π)),
      δFROMr[ rPlusψSmall[i, -d]] (360./(2. π))]}}], {i, nSrc}]
```

In[325]:=
```
(* Local contour plot of the alignment angle function η̄(H) on the grid. *)
(*dηContourPlot = 6 ;*)(*, in degrees. *)
frameticks = {{{ {yH[150, 22.5], 30 °}, {yH[150, 48.5], 60 °}}, None},
   {{{xH180[150, (*15*)30], "10h"},
     {xH180[180, 15], "12h"}, {xH180[210, 15], "14h"}}, {None}}};
```

In[326]:=
```
listCPlocalU = Show[ {Table[
    ParametricPlot[{xH180[α, δ], yH180[α, δ]}, {δ, 10, 90}, PlotStyle → {Black, Thickness[0.002]},
     PlotPoints → 60, PlotRange → {{xH180[(*135*)150, 30], xH180[(*225*)190, 30]},
       {yH180[180, (*15*)30], yH180[180, 62]}}, Axes -> False, Frame → True,
     FrameLabel → {"α", "δ", "Close-Up View"}, FrameTicks → frameticks], {α, 120, 240, 30}],
   Table[ParametricPlot[{xH180[α, δ], yH180[α, δ]}, {α, 90, 270},
     PlotStyle → {Black, Thickness[0.002]}, PlotPoints → 60], {δ, 0, 90, 30}],
   Graphics[{PointSize[0.01],  Red, (*Hmax:*)Point[ xyAitoffHmaxU ], PointSize[0.009], Gray,
     {Thick, polarLines[0.03]}, {Thick, polarLinesBig[0.03]}, {Thick, polarLinesSmall[0.03]},
     (*Sources S:*)Green, PointSize[0.012], Point[ xyAitoffSources ],
     PointSize[0.01], Blue, (*Hmin:*)Point[ xyAitoffHminU ], Gray, PointSize[0.005]
       }], ParametricPlot[{xH180[(HminαFit + rHminθ[θ] Cos[θ]) (360./(2. π)),
      (HminδFit + rHminθ[θ] Sin[θ]) (360./(2. π))],
     yH180[(HminαFit + rHminθ[θ] Cos[θ]) (360./(2. π)), (HminδFit + rHminθ[θ] Sin[θ]) (360./(2. π))]},
    {θ, 0., 2. π}, PlotStyle → {Orange, Thickness[0.01]}], ParametricPlot[
    {xH180[(HmaxαFit + rHmaxθ[θ] Cos[θ]) (360./(2. π)), (HmaxδFit + rHmaxθ[θ] Sin[θ]) (360./(2. π))],
     yH180[(HmaxαFit + rHmaxθ[θ] Cos[θ]) (360./(2. π)), (HmaxδFit + rHmaxθ[θ] Sin[θ]) (360./(2. π))]},
    {θ, 0., 2. π}, PlotStyle → {Orange, Thickness[0.005]}]}, ImageSize → 0.9 × 432  ];
```

In[327]:=
```
listCPlocalU
Print["Figure 24: Uncertainty plot. The sources are shaded green, ",
 Green, ". Polarization directions for the reported value
   ψ, and the one-sigma values ψ ± σψ are plotted as gray, ", Gray,
 ", line segments through the sources. All of the alignment hubs H_min from the uncertainty
   runs are plotted as overlapping blue, ", Blue, ", dots, with the orange, ", Orange,
 ", spot denoting the tiny ellipse of highest hub density. Many of the avoidance red dots, ",
 Red, ", for the H_max are off-graph. The big orange
   ellipse encloses the likely locations for avoidance hubs. "]
```

Out[327]=

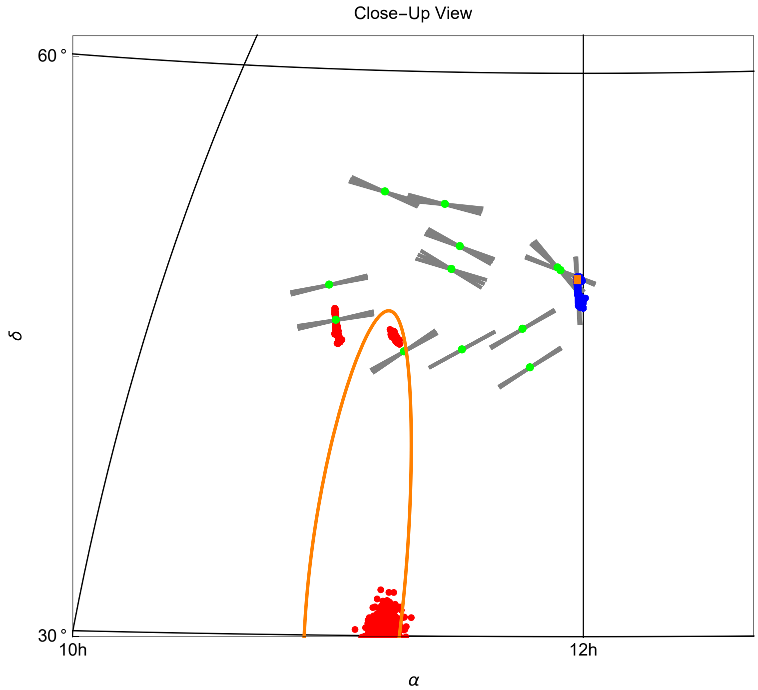


Figure 24: Uncertainty plot. The sources are shaded green, ■. Polarization directions for the reported value ψ, and the one-sigma values $\psi \pm \sigma\psi$ are plotted as gray, ■, line segments through the sources. All of the alignment hubs H_{min} from the uncertainty runs are plotted as overlapping blue, ■, dots, with the orange, ■, spot denoting the tiny ellipse of highest hub density. Many of the avoidance red dots, ■, for the H_{max} are off-graph. The big orange ellipse encloses the likely locations for avoidance hubs.

6h. Section Summary

In[329]:=
```
Print["To estimate the effects of experimental uncertainty, there were ",
 Length[funcDataU], " uncertainty runs."]
Print["Uncertainty runs have polarization directions ψ = ψSrc + δψ, ",
 "where δψ is chosen with a normal
   distribution of half-width σψ about the best value ψSrc."]
Print["The uncertainty runs determine the smallest alignment angle to be  η̄min = ",
 ηBarminUFit (360. / (2. π)), "° ± ", σηBarminUFit (360. / (2. π)), "°."]
Print["The uncertainty runs determine the largest avoidance angle to be  η̄max = ",
 ηBarmaxFitU (360. / (2. π)), "° ± ", σηBarmaxFitU (360. / (2. π)), "°."]
Print["Note, from Fig. 24, the avoidance hubs Hmax from uncertainty
   runs separate into two distinct blobs. Thus, the uncertainty
   runs determine the angle Θ between the two grey Great Circles in
   Figs. 3, 4, 12, 13, to be centered around two different values."]
Print["For the more likely Hmaxs, we have   Θ = ", ΘminmaxUFit3,
 "° ± ", Abs[dx0ΘminmaxUFit3], "°."]
Print["The less likely group of Hmax hubs give the angle
   Θ between the two grey Great Circles Θ = ", ΘminmaxUFit4,
 "° ± ", dx0ΘminmaxUFit4, "°. The more likely Great Circle from the
   sample to the avoidance hubs Hmax is drawn in the figures."]
```

```
To estimate the effects of experimental uncertainty, there were 10000 uncertainty runs.

Uncertainty runs have polarization directions ψ = ψSrc + δψ,
 where δψ is chosen with a normal distribution of half-width σψ about the best value ψSrc.

The uncertainty runs determine the smallest alignment angle to be  η̄min = 11.3889° ± 1.07175°.

The uncertainty runs determine the largest avoidance angle to be  η̄max = 63.1935° ± 0.964218°.

Note, from Fig. 24, the avoidance hubs Hmax from uncertainty runs separate into two
  distinct blobs. Thus, the uncertainty runs determine the angle Θ between the two
  grey Great Circles in Figs. 3, 4, 12, 13, to be centered around two different values.

For the more likely Hmaxs, we have   Θ = 105.092° ± 2.01426°.

The less likely group of Hmax hubs give the angle Θ between the two grey Great Circles Θ =
 170.689° ± 7.6131°. The more likely Great Circle
   from the sample to the avoidance hubs Hmax is drawn in the figures.
```

7. Probability and Significance

The problem of “significance” is to determine the likelihood that random polarizations directions would produce better alignment or avoidance than the observed polarization directions.

To determine the probability distributions and related formulas, we made many runs with random data and fit the results. In this effort, as has occurred previously elsewhere, one finds that the probability distributions for the smallest alignment angle $\overline{\eta}_{\min}$ and the largest avoidance angle $\overline{\eta}_{\max}$ are not well-described by Gaussian functions. Better fits have the Gaussian multiplied by a step-function. The fitting functions are based on the following distribution,

$$f(y) = \frac{1}{(2\pi)^{1/2}} \left(1 + e^{4(y-1)}\right)^{-1} e^{-\frac{y^2}{2}} \tag{4}$$

More discussion appears below when the function (4) is needed.

For example, random polarization directions are well-fit by a probability distribution for the smallest alignment angle $\overline{\eta}_{\min}$ that takes the form

$$P_{\min}(\eta) = \left(\frac{\text{norm}}{\sigma\,(2\,\pi)^{1/2}}\right)\left(1 + e^{4\,\frac{(\eta-\eta 0-\sigma)}{\sigma}}\right)^{-1} e^{-\frac{1}{2}\left(\frac{\eta-\eta 0}{\sigma}\right)^{2}}, \qquad (5)$$

where norm makes the integral of distribution equal to unity, $\eta 0$ and σ are parameters that are adjusted to fit the random run results.

7a. Probability and Significance Formulas

Definitions:

norm	a constant used to normalize the distribution so the integral of probability is 1.
probMIN0, probMAX0	probability distributions for alignment (MIN) and avoidance (MAX), functions of η, η_0, σ
signiMIN0, signiMAX0	significance as a function of (η, η_0, σ)

In[336]:=
```
(* y = ((η - η0)/σ); dy = dη/σ *)
(* The normalization factor "norm" is needed for the probability density *)
norm = (1/(2 π)^(1/2) NIntegrate[(1 + e^(4 (y-1)))^-1 e^(-y^2/2), {y, -∞, ∞}])^-1 ;
norm ;(*Constant needed to make the integral
 of the probability distribution equal to unity.*)
```

In[338]:=
```
probMIN0[η_, η0_, σ_] := (norm/(σ (2 π)^(1/2))) (1 + e^(4 (η-η0-σ)/σ))^-1 e^(-1/2 ((η - η0)/σ)^2)
signiMIN0[η_, η0_, σ_] := NIntegrate[probMIN0[η1, η0, σ], {η1, -∞, η}]
probMAX0[η_, η0_, σ_] := (norm/(σ (2 π)^(1/2))) (1 + e^(-4 (η-η0+σ)/σ))^-1 e^(-1/2 ((η - η0)/σ)^2)
signiMAX0[η_, η0_, σ_] := NIntegrate[probMAX0[η1, η0, σ], {η1, η, ∞}]
```

The significance signiMIN0[η, $\eta 0$, σ] is the Integral of probMIN0, i.e. signiMIN0 $= \int_{-\infty}^{\eta} P_{MIN}(\eta)\, d\eta$.

The significance signiMAX0[η, $\eta 0$, σ] is the Integral of probMAX0, i.e. signiMAX0 $= \int_{\eta}^{\infty} P_{MAX}(\eta)\, d\eta$.

7b. Generating random ψ runs

The notebook .nb version generates new random runs. The pdf version uses old random runs that are uploaded from previously saved files that are not publically available. Thus both versions have some cells commented out: (* comments are not processed by Mathematica*).

Definitions:

nRunMax number of random runs to be generated
ρRgnRadius distance to furthest source from sourceCenter, radians
minGridCenterToHmin, max - minimum number of grid spaces between Hmin, Hmax and sources' center
gridjηBarMinRand
iSminmas parameters for center to hub distance
nRunPrint dummy index to control printing frequency
rSrcxrGrid unit vector perpendicular to the plane of rSrc for S_i and rGrid to point H_j
ψSrcRand random polarization directions for the sources
rSrcxψSrc cross product of rSrc and the vector in direction of ψSrcR, both are unit vectors
jηBarToGrid {j, $\overline{\eta}_j$} = {grid point #, value of the alignment angle Eq. (1) averaged over all sources S_i, in radians}
sortjηBarToGrid - sort jηBarToGrid, smallest alignment angles $\overline{\eta}_j$ first
gridjηBarMinRand - {j,η_j} for the grid point H_j with the smallest alignment angle $\overline{\eta}_j$, not counting H_j that are too close to the sample
gridjηBarMaxRand - {j,η_j} for the grid point H_j with the largest avoidance angle $\overline{\eta}_j$, not counting H_j that are too close to the sample

niSnrData 1. run # 2. iSmin 3. iSmax 4. nSrc 6. ρRgnRadius
ψDataRand 1. run # 2. ψSrcRand table
runData 1. run # 2. sourceCenter 3. {j, $\overline{\eta}$} at point H_j where smallest $\overline{\eta}$ 4. {j, $\overline{\eta}$} at point H_j where largest $\overline{\eta}$ 5. nSrc 6. ρRgnRadius

In[342]:=

```
(*Remove comment marks, "(*" and "*)", below to generate your own table "runData". *)
(* Evaluate this cell for the notebook .nb version *)
(*
nRunMax=500;
niSnrData={};
ψDataRand={};
runData={};
times={};
(*Set up the For statement.*)
nRunPrint=0;
minGridCenterToHmin = 2;
(*minimum number of grid spaces between Hmin and sources' center*)
minGridCenterToHmax = 2;
(*minimum number of grid spaces between Hmax and sources' center*)
*)
```

In[343]:=

```
(* Evaluate this cell for the notebook .nb version *)
(*You may have found rSrcxrGrid already with uncertainty. Here it is again:*)
(*
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}];
*)
(*rSrcxrGrid: table of the unit vectors perpendicular to the plane
  of the great circle containing the source Si and the grid point Hj*)
```

In[344]:=
```
(* Evaluate this cell for the notebook .nb version *)
(*
t[1]=TimeUsed[];
For[nRun=1,nRun≤nRunMax,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+100];

 ψSrcRand=Table[RandomReal[{0.001,π-0.001}],{i,nSrc}];
 (*table of PPA angles ψ for the sources, in radians*)
 rSrcxψSrc =
  Table[ Sin[ψSrcRand[[i]]]eNSrc[[i]]-Cos[ψSrcRand[[i]]]eESrc[[i]],    {i,nSrc}];
 (*table of the cross product of rSrc and vector in direction of ψSrcRand,
 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
 *)iSmin=
  Catch[Do[If[ArcCos[sourceCenter.rGrid[[sortjηBarToGrid[[i,1]]  ]] -0.000001 ]/dθ1≥
      minGridCenterToHmin,Throw[i]],{i,100}]];
 gridjηBarMinRand=sortjηBarToGrid[[iSmin]]; (*  {j,ηj},
 at the grid point Hj with minimum η̅,not counting the center j0*)iSmax=
  Catch[Do[If[ArcCos[sourceCenter.rGrid[[sortjηBarToGrid[[-i,1]]  ]] -0.000001 ]/dθ1≥
      minGridCenterToHmax,Throw[i]],{i,100}]];
 gridjηBarMaxRand=sortjηBarToGrid[[-iSmax]]; (*  {j,ηj},
 at the grid point Hj with maximum η̅, not counting the center j0*)
 AppendTo[niSnrData,{nRun,iSmin,iSmax,nSrc,ρRgnRadius}];
 AppendTo[ψDataRand,{nRun,ψSrcRand}];
 AppendTo[runData,
  {nRun,sourceCenter,{grid[[  gridjηBarMinRand[[1]]  ]], gridjηBarMinRand[[2]]},
   {grid[[  gridjηBarMaxRand[[1]]  ]], gridjηBarMaxRand[[2]]},nSrc,ρRgnRadius   } ]
  (*collect data for saving in a file.*)  ]  ;
*)
```

In[345]:=
```
(* Evaluate this cell for the notebook .nb version *)
(*t[2]=TimeUsed[];
Print["Computer time needed to generate random runs: ",t[2]-t[1]," seconds."]*)
```

In[346]:=
```
(*Save a new table*)
SetDirectory[homeDirectory];
(*Put[niSnrData,"20211012niSnrDataQSON13Random2000a.dat"  ]*)
(*Put[ψDataRand,"20211012ψDataRandQSON13Random2000a.dat"  ]*)
(*Put[runData,"20211012runDataQSON13Random2000a.dat"]*)
```

In[347]:=
```
(*Get and old ψDataRand table*)
SetDirectory[homeDirectory];
(*niSnrData=Get["20210917niSnrDataQSON13Random2000a.dat"]*)
(*ψDataRand=Get["20210917niSnrDataQSON13Random2000a.dat"]*)
(*Get the runData files for the pdf version:*)

runData2000a = Get["20210917runDataQSON13Random2000a.dat"];
runData8000a = Get["20210917runDataQSON13Random8000a.dat"];
```

In[350]:=
```
(*Edit the following statements to Join separate data files, if needed*)
(*Join the runData files for the pdf version:*)

runData = Join[runData2000a, runData8000a];
nRunMax = Length[runData];
```

7c. Analyzing random ψ runs

Definitions:

ηBarminData	$\overline{\eta}_{\rm min}$ in order of random runs
sortηBarmin	sorted
η0Bmin, σBmin	mean and standard deviation of ηBarminData
hlmin, hlmin0	histogram data
nlmBmin	fit to $\overline{\eta}_{\rm min}$ histogram
{a,b,x0}	best fit parameters
showNlmBmin	figure displaying the fit to the $\overline{\eta}_{\rm min}$ from random runs
nlmBminPtable	Parameter table for the fit

ηBarmaxData	$\overline{\eta}_{\rm max}$
sortηBarmax	sorted
η0Bmax, σBmax	mean and standard deviation of ηBarmaxData
hlmax, hlmax0	histogram data
nlmBmax	fit to $\overline{\eta}_{\rm max}$ histogram
{a,b,x0}	best fit parameters
showNlmBmax	figure displaying the fit to the $\overline{\eta}_{\rm max}$ from random runs
nlmBmaxPtable	Parameter table for the fit

rHminR rGrid at $H_{\rm min}$

anglerHminToCenter θ from $H_{\rm min}$ to sourceCenter

θrHminToCenter, $\sigma\theta$rHminToCenter - mean and standard deviation of θ

rHmaxR rGrid at H_{max}
anglerHmaxToCenter θ from H_{max} to sourceCenter
θrHmaxToCenter, $\sigma\theta$rHmaxToCenter - mean and standard deviation of θ

In[352]:=
```
Print["There are ", Length[runData], " random runs to analyze."]
```

```
There are 10000 random runs to analyze.
```

runData
1. nRun 2. $\hat{r}$ at Region Center 3a. grid data for Hmin 3b. $\overline{\eta}_{min}$ 4a. grid data for Hmax 4b. $\overline{\eta}_{max}$ 5. nSrc 6. radius ρRgnRadius

In[353]:=
```
ηBarminData = Table[runData[[i1, 3, 2]] , {i1, Length[runData]}];
ηBarmaxData = Table[runData[[i1, 4, 2]] , {i1, Length[runData]}];
rHminR = Table[runData[[i1, 3, 1, 6]] , {i1, Length[runData]}] ;
rHmaxR =  Table[runData[[i1, 4, 1, 6]] , {i1, Length[runData]}];
sortηBarmin = Sort[ηBarminData];
η0Bmin = mean[ηBarminData ] ;(*Guess the mean for the Gaussian. *)
σBmin = stanDev[ηBarminData ];(*Guess the half-width.*)
hlmin0 = HistogramList[sortηBarmin, {η0Bmin - 5 σBmin, η0Bmin + 5 σBmin, 0.4 σBmin}];
hlmin = Table[{(1/2) (hlmin0[[1, i1]] + hlmin0[[1, i1 + 1]]), hlmin0[[2, i1]]},
   {i1, Length[ hlmin0[[2]] ]}];
nlmBmin = NonlinearModelFit[hlmin, {a (1 + E^(4 (x-x0-b)/b))^-1 Exp[- 1/2. ((x - x0)/b)^2] (*,b>0*)},
   {{a, Length[runData]/12}, {b, σBmin}, {x0, η0Bmin}}, x];
```

In[363]:=
```
{amin, bmin, x0min} = {a, b, x0} /. nlmBmin["BestFitParameters"];
{damin, dbmin, dx0min} = nlmBmin["ParameterErrors"]; (*x is ηBarmin*)
```

In[365]:=
```
sortηBarmax = Sort[ηBarmaxData];
η0Bmax = mean[ηBarmaxData ]; (*Guess the mean for the Gaussian. *)
σBmax = stanDev[ηBarmaxData ];(*Guess the half-width.*)
hlmax0 = HistogramList[sortηBarmax, {η0Bmax - 5 σBmax, η0Bmax + 5 σBmax, 0.4 σBmax}];
hlmax = Table[{(1/2) (hlmax0[[1, i1]] + hlmax0[[1, i1 + 1]]), hlmax0[[2, i1]]},
  {i1, Length[ hlmax0[[2]] ]}];
nlmBmax = NonlinearModelFit[hlmax, {a (1 + E^(-4 (x-x0+b)/b))^-1 Exp[- 1/2. ((x - x0)/b)^2] (*,b>0*)},
  {{a, nRunMax/12}, {b, σBmax}, {x0, η0Bmax}}, x];
```

In[370]:=
```
{amax, bmax, x0max} = {a, b, x0} /. nlmBmax["BestFitParameters"];
{damax, dbmax, dx0max} = nlmBmax["ParameterErrors"];(*x is ηBarmax*)
```

In[372]:=
```
anglerHminToCenter =
  Table[ArcCos[Abs[rHminR[[i]].sourceCenter] - 0.00001], {i, Length[rHminR]}];
θrHminToCenter = mean[anglerHminToCenter];
σθrHminToCenter = stanDev[anglerHminToCenter];
anglerHmaxToCenter =
  Table[ArcCos[Abs[rHmaxR[[i]].sourceCenter] - 0.00001], {i, Length[rHmaxR]}];
θrHmaxToCenter = mean[anglerHmaxToCenter];
σθrHmaxToCenter = stanDev[anglerHmaxToCenter]; t[6] = TimeUsed[];
fitData = {{nSrc,
    ρRgnRadius, ρRMS}, {x0min, dx0min}, {bmin, dbmin}, {amin, damin},
   {x0max, dx0max}, {bmax, dbmax}, {amax, damax}, {σθrHminToCenter,
    θrHminToCenter}, {σθrHmaxToCenter,
    θrHmaxToCenter}} (*collect data for saving in a file.*);
```

In[379]:=
```
ListPlot[{sortηBarmin, sortηBarmax}];
ListPlot[hlmin];
Normal[nlmBmin];
Print["The parameter table for the fit to η̅min: "]
nlmBminPtable = nlmBmin["ParameterTable"]
Normal[nlmBmax];
Print["The parameter table for the fit to η̅max: "]
nlmBmaxPtable = nlmBmax["ParameterTable"]
```

The parameter table for the fit to $\overline{\eta}_{\mathrm{min}}$:

Out[383]=

	Estimate	Standard Error	t-Statistic	P-Value
a	1591.22	9.66492	164.639	1.67881×10^{-35}
b	0.0823933	0.000559361	147.299	1.93825×10^{-34}
x0	0.531878	0.000468054	1136.36	5.90205×10^{-54}

The parameter table for the fit to $\overline{\eta}_{\mathrm{max}}$:

Out[386]=

	Estimate	Standard Error	t-Statistic	P-Value
a	1592.54	14.7099	108.263	1.67991×10^{-31}
b	0.0807127	0.000833291	96.8601	1.93433×10^{-30}
x0	1.0418	0.000697268	1494.12	1.432×10^{-56}

“fitData” table

1a. nSrc, number of sources 1b. rgnRadius, nominal radius of region 1c. RMS radius

2a. x0min: x0 = η0 align (min) 2b. dx0min error: dx0 - σ for x0 = η0 align (min)

3a. bmin: b = σ align (min) 3b. dbmin: err: db - σ for b = σ align (min)

4a. amin: a = Amplitude align (min) 4b. damin: err: da - σ for a = Amplitude align (min)

5a. x0max: x0 = η0 avoid (max) 5b. dx0maxx0max: err: dx0 - σ for x0 = η0 avoid (max)

6a. bmax: b = σ avoid (max) 6b. dbmax: err: db - σ for b = σ avoid (max)

7a. amax: a = Amplitude avoid (max)7b. damax: err: da - σ for a = Amplitude avoid (max)

8a. $\sigma\theta$rHminToCenter: stanDev[anglerHminToCenter] - σ for θ to H 8b. θrHminToCenter: mean[anglerHminToCenter] - θ to H

9a. $\sigma\theta$rHmaxToCenter: stanDev[anglerHmaxToCenter] - σ for θ to H 9b. θrHmaxToCenter: mean[anglerHmaxToCenter] - θ to H

In[387]:=
```
showNlmBmin =
  Show[{Histogram[Sort[ηBarminData], {η0Bmin - 5 σBmin, η0Bmin + 5 σBmin, 0.4 σBmin},
     PlotLabel → "Histogram for η̅min, random runs", AxesLabel → {"η̅min, radians", "ΔR"}],
    Plot[Normal[nlmBmin], {x, η0Bmin - 5 σBmin, η0Bmin + 5 σBmin}], ListPlot[hlmin],
    Graphics[{Blue, Arrow[{{ηBarMinfunDataObs, nRunMax/24}, {ηBarMinfunDataObs, 5.}}]}] }];
```

In[388]:=
```
showNlmBmax =
  Show[{Histogram[Sort[ηBarmaxData], {η0Bmax - 5 σBmax, η0Bmax + 5 σBmax, 0.4 σBmax},
     PlotLabel → "Histogram for η̅max, random runs", AxesLabel → {"η̅max, radians", "ΔR"}],
    Plot[Normal[nlmBmax], {x, η0Bmax - 5 σBmax, η0Bmax + 5 σBmax}], ListPlot[hlmax],
    Graphics[{Red, Arrow[{{ηBarMaxfunDataObs, nRunMax/24}, {ηBarMaxfunDataObs, 5.}}]}] }];
```

In[389]:=
```
GraphicsRow[{showNlmBmin, showNlmBmax}]
Print[
 "Figure 25: Random run results for smallest alignment angle η̅min and largest avoidance
   angle η̅max. Note that both curves have steeper sides toward the middle,
   η = π/4 = 45°. That requires non-Gaussian fitting functions
   in the 'NonlinearModelFit' statements above. The observed
   polarization directions give the results indicated by the arrows."]
```

Out[389]=

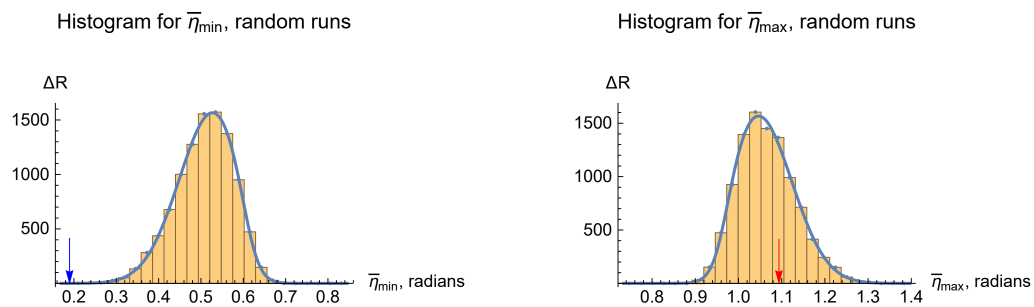


Figure 25: Random run results for smallest alignment angle $\overline{\eta}_{min}$ and largest avoidance angle $\overline{\eta}_{max}$. Note that both curves have steeper sides toward the middle, $\eta = \pi/4 = 45°$. That requires non-Gaussian fitting functions in the 'NonlinearModelFit' statements above. The observed polarization directions give the results indicated by the arrows.

7d. Significance of the alignment and avoidance Hub Test metrics for the sample studied in this work

Definitions

fitting function parameters from random runs:

η0min	mean of probability distribution for smallest alignment angle $\overline{\eta}_{min}$
dη0min	standard error in the mean as reported by Mathematica
σmin	half-width of probability distribution for smallest alignment angle $\overline{\eta}_{min}$
dσmin	standard error in the half-width as reported by Mathematica

η0max mean of probability distribution for largest avoidance angle $\overline{\eta}_{\mathtt{max}}$
dη0max standard error in the mean as reported by Mathematica
σmax half-width of probability distribution for largest avoidance angle $\overline{\eta}_{\mathtt{max}}$
dσmax standard error in the half-width as reported by Mathematica

probmin probability distribution for smallest alignment angle $\overline{\eta}_{\mathtt{min}}$. This depends on the random runs.
signimin significance, integral of probmin over smaller values of $\overline{\eta}_{\mathtt{min}}$

probmax probability distribution for largest avoidance angle $\overline{\eta}_{\mathtt{max}}$
signimax significance, integral of probmax over larger values of $\overline{\eta}_{\mathtt{max}}$

sigηBarMinfunDataObs Significance of the smallest alignment angle $\overline{\eta}_{\mathtt{min}}$
sigrangeηBarMinfunDataObs standard errors in η0min and σmin, *i.e.* dη0min and dσmin, give the significances plus/minus values
sigSmallηBarMinfunDataObs, Big extremes of significance assuming one standard error

sigηBarMaxfunDataObs Significance of the largest avoidance angle $\overline{\eta}_{\mathtt{max}}$
sigrangeηBarMaxfunDataObs standard errors in η0max and σmax, *i.e.* dη0max and dσmax, give the significances plus/minus values
sigSmallηBarMaxfunDataObs, Big extremes of significance assuming one standard error

In[391]:=
```
(*Parameters η0 and σ from random runs, together with their standard errors.*)
η0min = x0min; dη0min = dx0min;
η0max = x0max; dη0max = dx0max;
σmin = bmin; dσmin = dbmin;
σmax = bmax; dσmax = dbmax;
```

In[395]:=
```
probmin[η_] := probMIN0[ η, η0min, σmin ]
signimin[η_] := signiMIN0[η, η0min, σmin]
probmax[η_] := probMAX0[ η, η0max, σmax]
signimax[η_] := signiMAX0[η, η0max, σmax]
```

In[399]:=

```
Print["For this sample, but with random polarization directions ψ, the
   random runs give the mean value η0min and the half-width σmin of the
   fitting function of random runs for the smallest alignment angle η̄min:"]
Print[" η0min =  ", η0min (360./2. π), "° ± ", dη0min (360./2. π), "° and σmin  = ",
 σmin (360./2. π), "° ± ", dσmin (360./2. π), "°. (Random ψ distribution)"]
Print[" "]
Print[
 "For this sample, but with random polarization directions ψ, the random runs give the
   mean η0max  and the half-width σmax for the distributions for avoidance :"]
Print["   η0max =  ", η0max (360./2. π), "° ± ", dη0max (360./2. π), "° and σmax  = ",
 σmax (360./2. π), "° ± ", dσmax (360./2. π), "°. (Random ψ distribution)"]
```

For this sample, but with random polarization directions
ψ, the random runs give the mean value η0min and the half-width σmin of
the fitting function of random runs for the smallest alignment angle $\overline{\eta}_{\min}$:

η0min = 30.4744° ± 0.0268175° and σmin = 4.72079° ± 0.032049°. (Random ψ distribution)

For this sample, but with random polarization directions ψ, the random runs give
the mean η0max and the half-width σmax for the distributions for avoidance :

η0max = 59.6907° ± 0.0399505° and σmax = 4.6245° ± 0.0477441°. (Random ψ distribution)

In[404]:=

```
(*Significance of the smallest alignment angle η̄min .*)
sigηBarMinfunDataObs = signimin[ηBarMinfunDataObs];
sigrangeηBarMinfunDataObs =
  Sort[Partition[Flatten[Table[{signiMIN0[ηBarMinfunDataObs, η0min + γ1 dη0min,
        σmin + γ2 dσmin], γ1, γ2}, {γ1, -1, 1}, {γ2, -1, 1}] ], 3 ]  ];
{sigrangeηBarMinfunDataObs[[1]], sigrangeηBarMinfunDataObs[[-1]]};
sigSmallηBarMinfunDataObs = sigrangeηBarMinfunDataObs[[1, 1]];
sigBigηBarMinfunDataObs = sigrangeηBarMinfunDataObs[[-1, 1]];
```

In[409]:=

```
(*Significance of the largest avoidance angle η̄max .*)
sigηBarMaxfunDataObs = signimax[ηBarMaxfunDataObs];
sigrangeηBarMaxfunDataObs =
  Sort[Partition[Flatten[Table[{signiMAX0[ηBarMaxfunDataObs, η0max + γ1 dη0max,
        σmax + γ2 dσmax], γ1, γ2}, {γ1, -1, 1}, {γ2, -1, 1}] ], 3 ]  ];
{sigrangeηBarMaxfunDataObs[[1]], sigrangeηBarMaxfunDataObs[[-1]]};
sigSmallηBarMaxfunDataObs = sigrangeηBarMaxfunDataObs[[1, 1]];
sigBigηBarMaxfunDataObs = sigrangeηBarMaxfunDataObs[[-1, 1]];
```

In[414]:=
```
(*The names "gridjηBarMinRan", "jηBarMax" are, or perhaps were,
similar to quantities below, so save the current values labeled by "Best".*)
(* jηBar entries: 1. grid point # , 2. alignment angle .*)
{jηBarMinBest, jηBarMaxBest} = {ηBarMinfunDataObs, ηBarMaxfunDataObs} ;
```

In[415]:=
```
Print["The smallest alignment angle is ηmin = ", ηBarMinfunDataObs * (360. / (2. π)),
 "° , which has a significance of sig. = ", sigηBarMinfunDataObs,
 ",  plus/minus = + ", sigBigηBarMinfunDataObs - sigηBarMinfunDataObs, " and - ",
 sigηBarMinfunDataObs - sigSmallηBarMinfunDataObs, " , giving a range from sig. = ",
 sigSmallηBarMinfunDataObs, " to ", sigBigηBarMinfunDataObs, " ."]
Print["The largest avoidance angle is ηmax = ", ηBarMaxfunDataObs * (360. / (2. π)),
 "° , which has a significance of sig. = ", sigηBarMaxfunDataObs,
 ",  plus/minus = + ", sigBigηBarMaxfunDataObs - sigηBarMaxfunDataObs, " and - ",
 sigηBarMaxfunDataObs - sigSmallηBarMaxfunDataObs, " , giving a range from sig. = ",
 sigSmallηBarMaxfunDataObs, " to ", sigBigηBarMaxfunDataObs, " ."]
Print["These uncertainties are due to the standard
   errors for the parameters in the fit to the random runs."]
```

The smallest alignment angle is ηmin = 10.8648
° , which has a significance of sig. = 0.0000199444, plus/minus = + 3.14916×10^{-6}
and − 2.77319×10^{-6} , giving a range from sig. = 0.0000171712 to 0.0000230936 .

The largest avoidance angle is ηmax = 62.6651° , which has a significance of sig. = 0.317233
, plus/minus = + 0.00600697 and − 0.00607283 , giving a range from sig. = 0.31116 to 0.32324 .

These uncertainties are due to the standard
errors for the parameters in the fit to the random runs.

In[432]:=
```
Print["More Statistics of the Alignment Function η̄(H) :"]
Print[" "]
Print["The min alignment angle, ηmin = ", ηBarMinfunDataObs * (360. / (2. π)),
 "°, is Δη = ", (η0min - ηBarMinfunDataObs) * (360. / (2. π)),
 "° below the most likely value, ", η0min * (360. / (2. π)), "°, for random runs."]
Print["Since the half-width σ is ", σmin * (360. / (2. π)),
 "°, the difference, Δη = ", (η0min - ηBarMinfunDataObs) * (360. / (2. π)),
 "° makes ηmin separated from the most likely random run value by ",
 (η0min - ηBarMinfunDataObs) / σmin, "σs."]
Print["Thus, the smallest alignment angle η̄min is " , (η0min - ηBarMinfunDataObs) / σmin,
 "σs below the most likely random run value. (Very Significant)"]
Print[""]
Print["The max avoidance angle, ηmax = ", ηBarMaxfunDataObs * (360. / (2. π)),
 "°, is Δη = ", - (η0max - ηBarMaxfunDataObs) * (360. / (2. π)),
 "° above the most likely value, ", η0max * (360. / (2. π)), "°, for random runs."]
Print["Since the half-width σ is ", σmax * (360. / (2. π)),
 "°, the difference Δη = ", - (η0max - ηBarMaxfunDataObs) * (360. / (2. π)),
 "° makes ηmax separated from the most likely random run value by  ",
 - (η0max - ηBarMaxfunDataObs) / σmax, "σs."]
Print["Thus, the smallest avoidance angle η̄max is " , - (η0max - ηBarMaxfunDataObs) / σmax,
 "σs above the most likely random run value. (Not significant)"]
```

More Statistics of the Alignment Function $\overline{\eta}(H)$:

The min alignment angle, ηmin = 10.8648°, is $\Delta\eta$ = 19.6096° below the most likely value, 30.4744°, for random runs.

Since the half-width σ is 4.72079°, the difference, $\Delta\eta$ = 19.6096 ° makes ηmin separated from the most likely random run value by 4.15388σs.

Thus, the smallest alignment angle $\overline{\eta}_{\min}$ is 4.15388 σs below the most likely random run value. (Very Significant)

The max avoidance angle, ηmax = 62.6651°, is $\Delta\eta$ = 2.97431° above the most likely value, 59.6907°, for random runs.

Since the half-width σ is 4.6245°, the difference $\Delta\eta$ = 2.97431 ° makes ηmax separated from the most likely random run value by 0.643165σs.

Thus, the smallest avoidance angle $\overline{\eta}_{\max}$ is 0.643165 σs above the most likely random run value. (Not significant)

7e. Conclusion

The avoidance of the polarization directions for points on the Celestial Sphere is not significant, with $S = 0.32$. That means about one in three random runs would avoid some place on the Celestial Sphere better than the sample avoids $H_{\max}$. That is not significant.
The polarization directions are very significantly aligned, with $S = 2.\,\mathrm{x}10^{-5}$. That means about one random run in 50,000 would produce better alignment, a “4σ” result.
The polarization directions converge on the hub $H_{\min}$ with a smallest alignment angle $\overline{\eta}_{\min}$ that is very significant. They are therefore correlated.

In[427]:=
```
Print["The computer time on my computer is about one minute because I have
    uploaded the uncertainty runs and random runs from saved data files."]
Print["The same computer takes about 10 minutes to complete the .nb version with the
    bulk of the time needed to generate 500 uncertainty runs and 500 random runs."]
```

The computer time on my computer is about one minute because I have uploaded the uncertainty runs and random runs from saved data files.

The same computer takes about 10 minutes to complete the .nb version with the bulk of the time needed to generate 500 uncertainty runs and 500 random runs.

In[429]:=
```
Print["The computer time expended so far is ", TimeUsed[], " seconds."]
```

The computer time expended so far is 74.067 seconds.