TC-OXF-22

HIRDLS PROGRAM
FILE COPY

PROJECT

March/April, 1991

Project supervisor : **Chris Palmer**
Project student : **Andrew Longmore**

List of contents

ANGULAR CALIBRATION OF GIMBAL MIRROR

An alternative approach by means of corner cube interferometry investigated

1. Introduction

It is possible that the traditional methods of angular calibration, such as using the angular scales of a theodolite, may be replaced by a more accurate method which uses the properties of corner cubes. A computer program is being developed to test the feasibility of such an approach, determining how sensitive the method would be to basic imperfections in the apparatus.

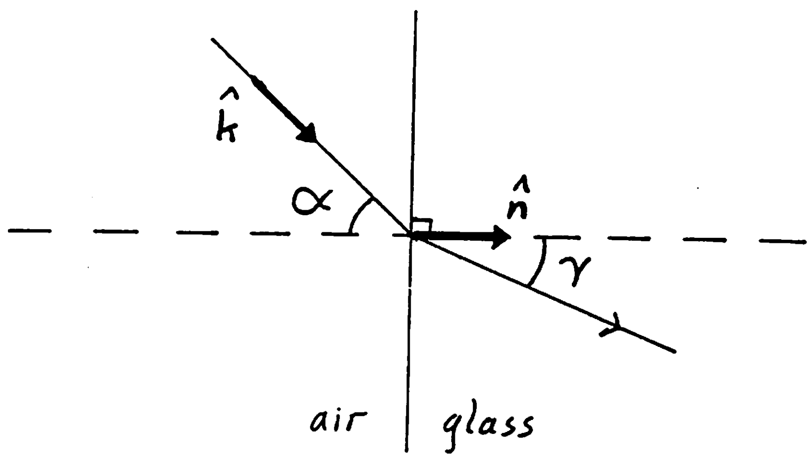
A corner cube prism may be considered as simply a corner cut off from a cubical piece of glass, in which the three mutually perpendicular faces of the cube are the reflcting faces of the prism, and the plane cut glass surface is the face through which the light (in this case, a laser beam) enters and leaves the prism. Such an arrangement has the property of retro-reflection — the emergent ray is anti-parallel to the incoming ray, regardless of the orientation of the incoming ray with respect to the glass surface.

By replacing the two plane mirrors of a Michelson interferometer by corner cubes, and mounting one of these corner cubes in the plane of the gimbal mirror, changes in the path length of the light going to and returning from the mounted corner cube can be measured, despite the fact that, during gimbal mirror rotations, the glass surface of the mounted corner cube will not remain perpendicular to the incoming light — this is by virtue of the retro-reflective property of the corner cube.

If this were done at three points on the plane of the gimbal mirror, then the displacements of each of these points could be measured and the angles of azimuth and elevation deduced. However, if the apparatus is not in perfect arrangement, the deductions of the angles of azimuth and elevation will be incorrect. The purpose of the computer program is to predict the errors in these deduced angles resulting from a variety of basic imperfections in the apparatus.

2. Theory of the corner cube interfermoter

For a fuller analysis on this subject, see EDSON R. PECK, Theory of the Corner Cube Interferometer, *Journal of the Optical Society of America, Volume 38 number 12, pp 1015-1024*, from which the analysis necessary for this project takes its starting point. However, here are some important definitions and properties of corner cubes —



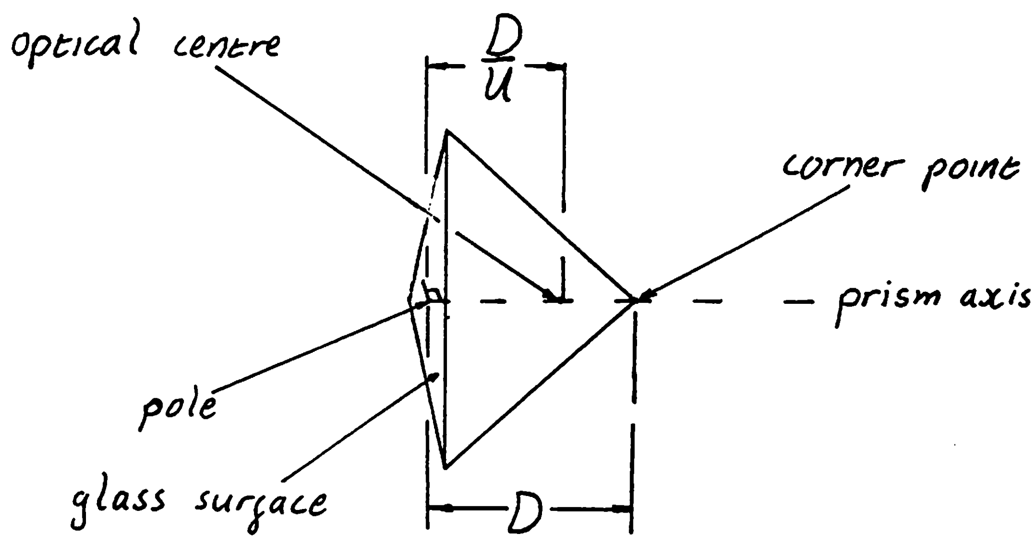


Figure 2.1
Description of light and corner cube

where D is the depth of the corner cube, i.e. the distance between the corner point and the pole

$\hat{k}$ is the unit wave vector

$\hat{n}$ is the unit normal to the glass surface, pointing in to the cube

α is the angle made by an incident or emergent ray in air to the normal of the glass surface

γ is the angle made by an incident or emergent ray in glass to the normal of the glass surface

U is the refractive index of the glass.

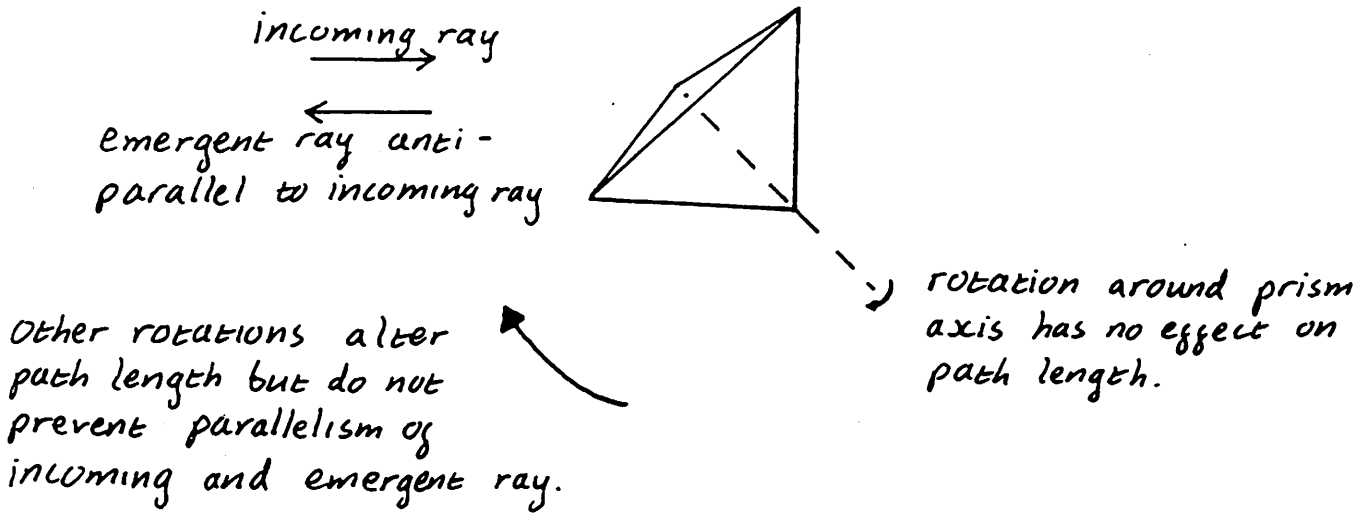


Figure 2.2
Retro-reflection

By Peck's original analysis, the path length difference between the light going to the real corner cube (which will be the one mounted in the plane of the gimbal mirror) and the virtual corner cube (the image formed by reflection at the dividing plate of the interferometer of the second corner cube) is

$$\text{path length difference } = 2S\cos\beta + 2UD(\cos\gamma - \frac{\cos\alpha}{U^2}) - 2U'D'(\cos\gamma' - \frac{\cos\alpha'}{U'^2}).$$

(Primed values referring to the virtual corner cube).

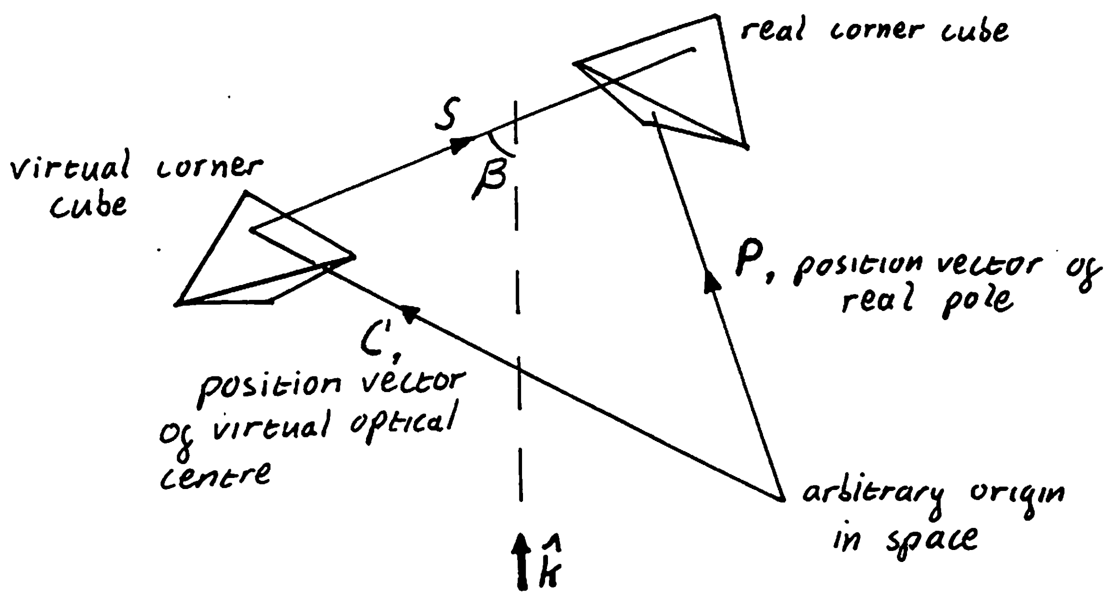


Figure 2.3
Orientation of real and virtual cube

(S being the vector from the optical centre of the virtual corner cube to the optical centre of the real corner cube).

Now the virtual corner cube will be fixed in space, so

$$
\begin{aligned}
\text{path length difference} &= 2S\cos\beta + 2UD\cos\gamma - \frac{2D\cos\alpha}{U} + \text{constant} \\
&= 2\mathbf{S}.\hat{\mathbf{k}} + 2UD\cos\gamma - \frac{2D}{U}\hat{\mathbf{n}}.\hat{\mathbf{k}} + \text{constant} \\
&= 2(\mathbf{S} - \frac{D}{U}\hat{\mathbf{n}}).\hat{\mathbf{k}} + 2UD\cos\gamma + \text{constant}. \\
\text{Now, } \mathbf{C}' + \mathbf{S} &= \mathbf{P} + \frac{D}{U}\hat{\mathbf{n}} \\
\text{therefore } \mathbf{P} - \mathbf{C}' &= \mathbf{S} - \frac{D}{U}\hat{\mathbf{n}} \\
\text{and path length difference} &= 2(\mathbf{P} - \mathbf{C}').\hat{\mathbf{k}} + 2UD\cos\gamma + \text{constant} \\
&= 2(\mathbf{P}.\hat{\mathbf{k}} + UD\cos\gamma) + \text{constant},
\end{aligned}
$$

since $\mathbf{C}'$ is fixed.

Hence, if the pole of the corner cube mounted in the plane of the gimbal mirror moves from $\mathbf{P}_o$ to $\mathbf{P}$, then the change in the path length difference will be given by

$$\Delta = 2(\mathbf{P}.\hat{\mathbf{k}} + UD\cos\gamma) - 2(\mathbf{P}_o.\hat{\mathbf{k}} + UD\cos\gamma_o)$$

Note — since the virtual cube is fixed, the change in path length difference is the same as the change in the path length to the real cube.

3. Description of mirror and of corner cubes

We now have to decide how the program will describe the mirror and the corner cubes. The mirror can rotate around two axes, azimuthal and elevation, and in the perfect case it is assumed that the incoming light for each mounted corner cube will be parallel. Hence our fixed coordinate axes in space (x,y,z) will be such that the z-axis is the azimuthal axis, the x-z plane is always perpendicular to the ideal light rays, and the x-y plane always contains the elevation axis. (This defines the origin as being the intersection of the azimuthal and the elevation axis). Furthermore, we will define ψ, the angle of azimuth, as being zero when the elevation axis coincides with the x-axis, and θ, the angle of elevation, as being zero when the mirror normal lies in the x-y plane.

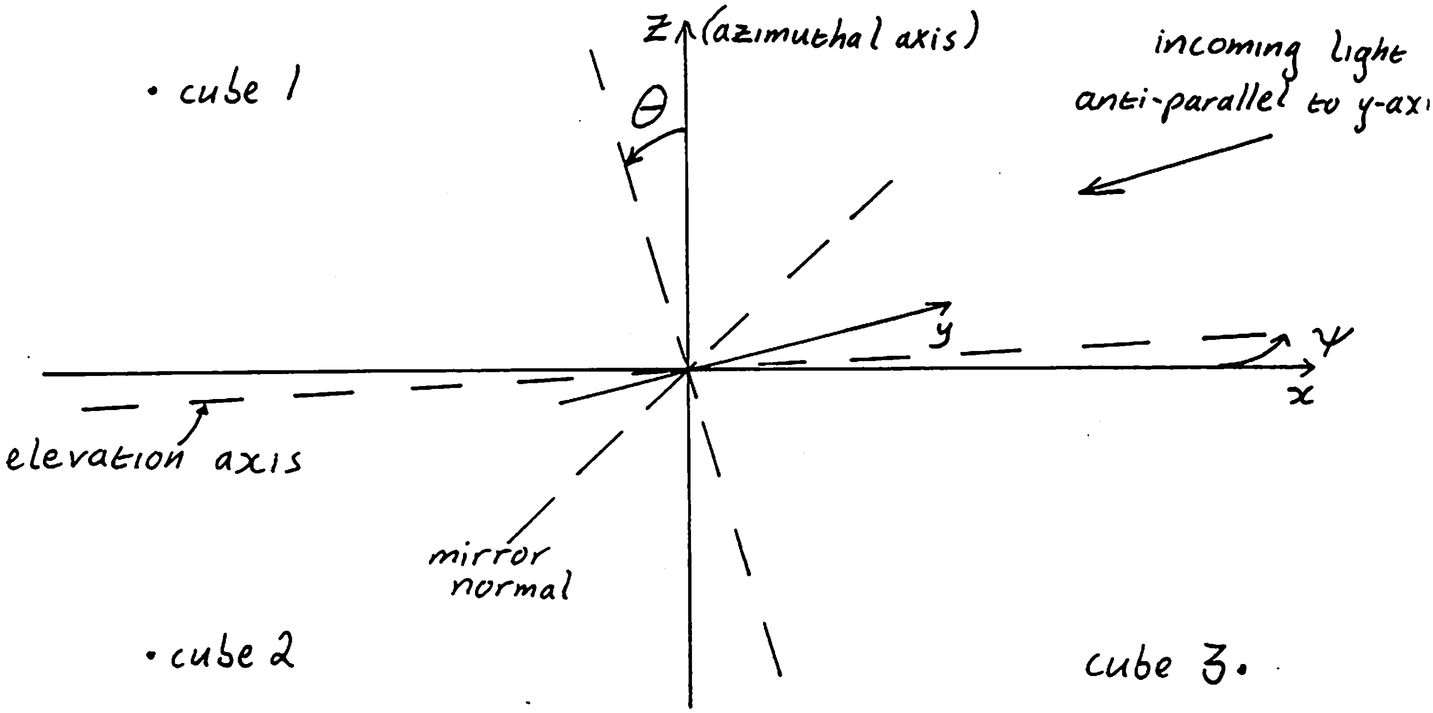


Figure 3.1
Fixed coordinate system

• = position of cubes.

The computer program assumes that each corner cube, in the ideal case, will be the same distance from the azimuthal and elevation axes as the others, and also the same height above or below the plane of the mirror.

Information about corner cube positions is given to the computer in terms of coordinates in the fixed (x,y,z) coordinate system for the case when $\theta = \psi = 0$, referred to as the initial or unrotated case.

4. Mathematical principles

The calculational side of the program falls into two main categories:

i. Determining for each cube the change, Δ, in the path length of light to and from the real cube (under actual conditions) due to a rotation θ, ψ.

ii. Determining the errors incurred when you use the Δs to deduce θ', ψ' as if the rotation had been done under ideal conditions.

i. Δ

$$\Delta = 2(\mathbf{P}.\hat{\mathbf{k}} - UD\cos\gamma) - 2(\mathbf{P}_o.\hat{\mathbf{k}} - UD\cos\gamma_o).$$

Now $U, \mathbf{P}_o, D, \hat{\mathbf{k}}$,and γ_o are all determined directly from the inputted information, and are independent of the gimbal mirror position θ, ψ.

The ideal values for these variables are calculated trivially from the inputted information and the arrangement of the fixed coordinate system (x,y,z). The actual values, i.e. the values which take into account the physical inaccuracies of the system, are calculated as follows:

Refractive index,

$$U = \text{ideal value of } U + \text{ error in } U.$$

Initial position vector of corner point,

$\mathbf{C}_o-$ the ideal magnitudes of the components of $\mathbf{C}_o$ are given in the input, and the signs of these components are determined by which quadrant of the fixed coordinate system the corner cube is initially in. The relevant errors, which are inputted as displacements from the ideal position, are then added.

Initial position of pole,

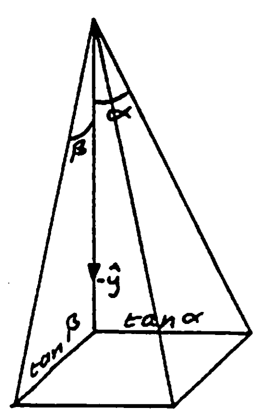
$\mathbf{P}_o-$ determined as for $\mathbf{C}_o$.

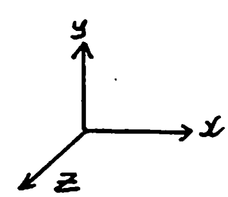
Depth of corner cube,

$D = |\mathbf{C}_o - \mathbf{P}_o|$.

Unit wave vector,

$\hat{\mathbf{k}}-$ if the projection of the actual $\hat{\mathbf{k}}-$vector deviates from the y − axis in the x − y plane by an angle α, and in the y − z plane by an angle β, then the actual value of $\hat{\mathbf{k}}$ will be given by





ideally, $\hat{k} = -\hat{y}$

$$\hat{\mathbf{k}} \propto -\hat{\mathbf{y}} + \hat{\mathbf{x}}\tan\alpha + \hat{\mathbf{z}}\tan\beta$$

$$= \frac{1}{\sqrt{1 + \tan^2\alpha + \tan^2\beta}} \begin{pmatrix} \tan\alpha \\ -1 \\ \tan\beta \end{pmatrix}.$$

Initial unit inwards normal,

$$\hat{\mathbf{n}}_o = \frac{\mathbf{C}_o - \mathbf{P}_o}{D}.$$

Cosine of the initial angle to the normal of the glass surface made by the ray in air,

$$\cos\alpha_o = \hat{\mathbf{n}}_o.\hat{\mathbf{k}}.$$

Cosine of the initial angle to the normal of the glass surface made by the ray in glass,

$\cos\gamma_o -$

$$\sin\gamma_o = \frac{1}{U}\sin\alpha_o$$

$$1 - \cos^2\gamma_o = \frac{1}{U^2}(1 - \cos^2\alpha_o)$$

$$\cos\gamma_o = \frac{1}{U}\sqrt{U^2 - 1 + \cos^2\alpha_o}.$$

The only things which vary under rotation are $\mathbf{P}$ and $\cos\gamma$. If $[\theta, \psi]$ is the rotation matrix of the gimbal mirror through an angle of elevation θ, and an angle of azimuth ψ, then we have:
position vector of the pole,

$$\mathbf{P} = [\theta, \psi]\mathbf{P}_o$$

and cosine of the angle to the normal of the glaas surface made by the ray in glass,

$$\cos\gamma = \frac{1}{U}\sqrt{U^2 - 1 + \cos^2\alpha}$$

where $\cos\alpha = \hat{\mathbf{n}}.\hat{\mathbf{k}}$

and $\hat{\mathbf{n}} = [\theta, \psi]\hat{\mathbf{n}}_o$

Now all we need is the form of the rotation matrix $[\theta, \psi]$. Let (x,y,z) be our fixed axes.

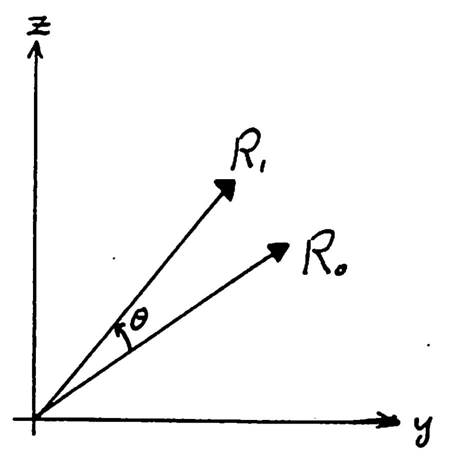


Figure 4i.1
θ rotation

Consider a point

$$\mathbf{R_0} = \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}.$$

This is rotated about the x-axis by an angle θ to

$$\mathbf{R_1} = \begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}.$$

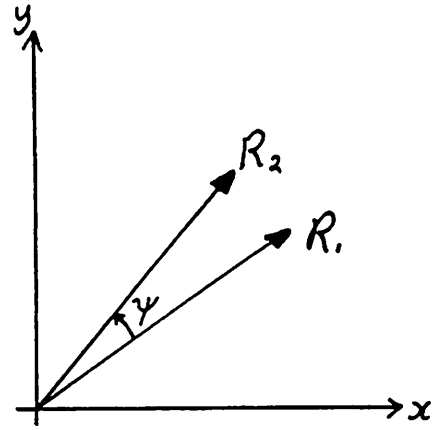


Figure 4i.2
ψ rotation

Now consider rotating this point about the z-axis by an angle ψ to

$$
\begin{aligned}
\mathbf{R_2} &= \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} \\
&= \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \\
&= \begin{pmatrix} \cos\psi & -\sin\psi\cos\theta & \sin\psi\sin\theta \\ \sin\psi & \cos\psi\cos\theta & -\cos\psi\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \\
&= [\theta, \psi]\mathbf{R_0}.
\end{aligned}
$$

Hence we can determine Δ for any rotation θ, ψ.

ii. ϵ

In the ideal case,

$$\Delta^i = 2(\mathbf{P}^i.\hat{\mathbf{k}} - UD\cos\gamma) - 2(\mathbf{P}^i_o.\hat{\mathbf{k}} - UD\cos\gamma_o).$$

(^{i} referring to a particular cube)

If the Δs of different cubes are subtracted, for the same mirror angle θ, ψ, then

$$\frac{1}{2}(\Delta^A - \Delta^B) = \mathbf{P}^A.\hat{\mathbf{k}} - \mathbf{P}^B.\hat{\mathbf{k}},$$

since all variables except $\mathbf{P}^i$ are independent of which cube we are talking about. Also, in the ideal case, $\hat{\mathbf{k}} = -\hat{\mathbf{y}}$, so

$$\begin{aligned}
\frac{1}{2}(\Delta^A - \Delta^B) &= \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} . \begin{pmatrix} \cos\psi & -\sin\psi\cos\theta & \sin\psi\sin\theta \\ \sin\psi & \cos\psi\cos\theta & -\cos\psi\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X^A_o \\ Y^A_o \\ Z^A_o \end{pmatrix} \\
&- \begin{pmatrix} 0 \\ -1 \\ 0 \end{pmatrix} . \begin{pmatrix} \cos\psi & -\sin\psi\cos\theta & \sin\psi\sin\theta \\ \sin\psi & \cos\psi\cos\theta & -\cos\psi\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X^B_o \\ Y^B_o \\ Z^B_o \end{pmatrix} \\
&= -(\sin\psi\ X^A_o + \cos\psi\cos\theta\ Y^A_o - \cos\psi\sin\theta\ Z^A_o) \\
&\quad + (\sin\psi\ X^B_o + \cos\psi\cos\theta\ Y^B_o - \cos\psi\sin\theta\ Z^B_o)
\end{aligned}$$

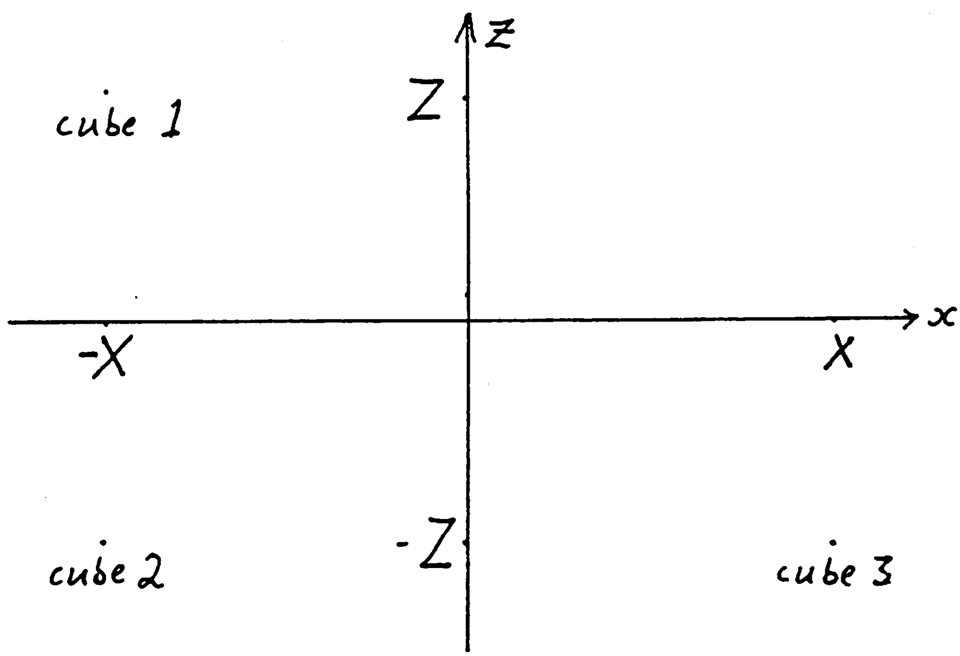


Figure 4ii.1
Orientation of corner cubes

Now the cubes are arranged such that

$$\begin{aligned}
-X^1_o &= -X^2_o = +X^3_o = X \\
+Y^1_o &= +Y^2_o = +Y^3_o \\
+Z^1_o &= -Z^2_o = -Z^3_o = Z.
\end{aligned}$$

Therefore,

$$\begin{aligned}
\frac{1}{2}(\Delta^2 - \Delta^3) &= 2\sin\psi'\ X, \qquad \Rightarrow \psi' \\
\frac{1}{2}(\Delta^1 - \Delta^2) &= 2\cos\psi'\sin\theta'\ Z, \Rightarrow \theta'.
\end{aligned}$$

These values θ' and ψ', the values for the angles of elevation and azimuth which you would deduce from the actual Δs assuming ideal conditions, can be subtracted from the values θ and ψ which did the rotation, producing errors in elevation and azimuth $\delta\theta$ and $\delta\psi$.

Since both are calculated from inverse sines rather than inverse cosines, they should be relatively free from rounding errors and may be either positive or negative.

However, with two errors in angle, $\delta\theta$ and $\delta\psi$, it is difficult to judge the overall effect of the inputted physical errors. The program also calculates ϵ, which is the angle the unit normal of the gimbal mirror turns through from going from the actual arrangement θ, ψ to the incorrectly deduced arrangement θ', ψ'. This is an estimate of the optical effect of the errors.

Now,

$$\begin{aligned}\delta\theta &= \theta - \theta' \\ \delta\psi &= \psi - \psi'\end{aligned}$$

$$\begin{aligned}
\cos\epsilon &= ([\theta', \psi']\hat{\mathbf{n}}_o).([\theta, \psi]\hat{\mathbf{n}}_o) \\
&= \begin{pmatrix} \cos\psi' & -\sin\psi'\cos\theta' & \sin\psi'\sin\theta' \\ \sin\psi' & \cos\psi'\cos\theta' & -\cos\psi'\sin\theta' \\ 0 & \sin\theta' & \cos\theta' \end{pmatrix}\begin{pmatrix}0\\1\\0\end{pmatrix} . \begin{pmatrix} \cos\psi & -\sin\psi\cos\theta & \sin\psi\sin\theta \\ \sin\psi & \cos\psi\cos\theta & -\cos\psi\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix}\begin{pmatrix}0\\1\\0\end{pmatrix} \\
&= \begin{pmatrix} -\sin\psi'\cos\theta' \\ \cos\psi'\cos\theta' \\ \sin\theta' \end{pmatrix} . \begin{pmatrix} -\sin\psi\cos\theta \\ \cos\psi\cos\theta \\ \sin\theta \end{pmatrix} \\
&= \sin\psi'\cos\theta'\sin\psi\cos\theta + \cos\psi'\cos\theta'\cos\psi\cos\theta + \sin\theta'\sin\theta \\
&= \cos\theta'\cos\theta(\sin\psi'\sin\psi + \cos\psi'\cos\psi) + \sin\theta'\sin\theta \\
&= \cos\theta'\cos\theta\cos\delta\theta + \sin\theta'\sin\theta \\
1 - 2\sin^2\frac{\epsilon}{2} &= \cos\theta'\cos\theta(1 - 2\sin^2\frac{\delta\psi}{2}) + \sin\theta'\sin\theta \\
&= (\cos\theta'\cos\theta + \sin\theta'\sin\theta) - 2\cos\theta'\cos\theta\sin^2\frac{\delta\psi}{2} \\
&= \cos\delta\theta - 2\cos\theta'\cos\theta\sin^2\frac{\delta\psi}{2} \\
&= 1 - 2\sin^2\frac{\delta\theta}{2} - 2\cos\theta'\cos\theta\sin^2\frac{\delta\psi}{2} \\
\sin^2\frac{\epsilon}{2} &= \sin^2\frac{\delta\theta}{2} + \cos\theta'\cos\theta\sin^2\frac{\delta\psi}{2}. \\
\epsilon &= 2\sin^{-1}\sqrt{\sin^2\frac{\delta\theta}{2} + \cos\theta'\cos\theta\sin^2\frac{\delta\psi}{2}}
\end{aligned}$$

5. Program structure

1. Define global variables.
   1.1 Give type (integer, real or double precision) and dimensions of arrays (if applicable).
   Double precision variables were used for square roots, trigonometric functions and if numbers of much different magnitudes were added together.
   1.2 Define common data blocks.
   Groups of global variables, shared by the same subroutines, were placed in the same global data blocks.

2. Input ideal unrotated coordinates, etc... and the errors in them.

3. Calculate the initial values for the variables in the ideal case.
   These are calculated trivially from the inputted information and the arrangement of the cubes in the fixed coordinate system.

4. Calculate the initial values of the variables in the actual case.
   These are calculated as described previously in **4. Mathematical principles, part i. Δ.**

5. Print out the unrotated coordinates and the values of the initial variables for both the ideal and actual cases.
   This was useful during the development of the program as an intermediate check of the data between input and the rotation calculations. If the program were used interactively, it would also help to pick up typing errors before the main bulk of the processing began. I generally used the program via command files due to the large amount of information the program requires as input.

6. Input the scanning ranges.
   The information is entered simply as a minimum of, maximum of, and interval in elevation and azimuthal angle. The program anticipates angles to be measured in line-of-sight radians and, unlike measurements of distance, the units are not arbitrary provided that the input is consistent. Scanning does not consist of loops of the double precision form

   DO <label>, PSI = PSIMIN, PSIMAX, PSIINT

   since, for large trip-counts, these are subject to relatively high rounding errors. Instead, the trip-count of the loop is calculated, and loops take the integer form

   DO <label>, NPSI = 1, PSITRIP

   The values of PSI are then calculated inside the loop, so that rounding errors are not accumulative.

7. Do the scan for the actual case.
   The values of Δ are calculated, again as described in **4. Mathematical principles, part i. Δ,** and stored in a 3 x 26 x 26 array. The first element refers to which cube the value of Δ refers to, and the second two elements define the scan position.

8. Calculate ϵ.
   This is done as described in **4. Mathematical principles, part ii. ϵ.**
   8.1 Calculate θ', ψ'.
   8.2 Calculate $\delta\theta, \delta\psi$.
   8.3 Calculate ϵ
   8.4 Store $\theta, \psi, \epsilon, \delta\theta$, and $\delta\psi$ in a file called GRAF.DAT.

9. Produce a list file for the presentation of the data. This file also contains the mean value of ϵ.

10. Produce program files for the use of IDL graphics to give 3-D plots of $\epsilon, \delta\theta$ and $\delta\psi$ against θ and ψ.