

# Étalonnage d'une monture équatoriale

Michel Llibre - Club d'astronomie de Quint-Fontsegrives

Avril 2023

## Introduction

Une monture équatoriale possède deux axes mobiles en rotation pour permettre de diriger la visée du télescope. Le premier, appelé axe horaire (ou axe d'ascension droite) doit être parallèle à l'axe de rotation de la Terre. Le deuxième axe doit être orthogonal au premier. Ces règles sont généralement respectées, mais avec quelques petites erreurs qui viennent perturber la précision des visées. Ce document présente une méthode d'identification des paramètres responsables des erreurs de visée des montures équatoriales.

Pour modéliser ces erreurs, on peut faire appel aux formules de trigonométrie sphérique. Elles sont utilisées par les astronomes professionnels, mais peu connues des amateurs. De plus elles sont rapidement complexes, et peu appropriées pour enchaîner plus de trois rotations alors que la modélisation des erreurs que nous présentons ici en chaîne jusqu'à onze. Pour faire cette modélisation nous utilisons les matrices 3x3 de rotation qui sont très utilisées dans les programmes de jeux informatiques 3D. Pour ces applications, le nombre de rotations utilisées est faible (3 au plus pour passer d'un point de vue à un autre), et il existe de nombreuses bibliothèques toutes faites. En fait, il n'y a pas besoin de bibliothèque spécialisée au niveau de la programmation : le travail est simple : le calcul d'une matrice de rotation élémentaire prend quelques lignes de code (un sinus, un cosinus et  $3 \times 3 = 9$  affectations) et un enchaînement de deux rotations se traduit par le produit de deux matrices 3x3 soit une boucle de 9 tours contenant 3 multiplications et 2 additions. Mais comme le produit des matrices n'est pas commutatif l'ordre est *significatif*, et comme les matrices de rotation ne sont pas symétriques leur *sens est significatif* (sens du point de vue de la transposition). Ces deux points sont source de confusion. Le premier chapitre est uniquement consacré à présenter la convention que nous utilisons pour ce sens et pour cet ordre. Certains auteurs utilisent un sens et un ordre inverse au notre, ce qui est strictement équivalent. Il suffit que sens et ordre choisis soient cohérents entre eux.

Le deuxième chapitre présente la modélisation de l'erreur de pointage en fonction des écarts sur les paramètres responsables de cette erreur.

Le troisième chapitre présente une méthode d'identification de ces écarts à partir d'un lot de mesures d'erreurs de pointage.

Le dernier chapitre présente un programme Octave (simili Matlab) d'identification de ces erreurs.

## 1 Changements de bases et opérateurs rotations

Cette chapitre peut être sauté, car il ne concerne pas le problème étudié. Il est destiné aux lecteurs qui veulent vérifier les développements mathématiques ou qui ne sont habitués à nos notations.

### 1.1 Changement de bases d'un vecteur

Considérons deux bases orthonormées  $\mathcal{B}_A$  de vecteurs unitaires  $\vec{i}_A$ ,  $\vec{j}_A$  et  $\vec{k}_A$ , et  $\mathcal{B}_B$  de vecteurs unitaires  $\vec{i}_B$ ,  $\vec{j}_B$  et  $\vec{k}_B$ .

Considérons également un vecteur  $\vec{u}$  défini par :

$$\vec{u} = x_A \vec{i}_A + y_A \vec{j}_A + z_A \vec{k}_A = x_B \vec{i}_B + y_B \vec{j}_B + z_B \vec{k}_B$$

Nous noterons  $(u)_A$  les composantes du vecteur  $\vec{u}$  dans la base  $\mathcal{B}_A$  et  $(u)_B$  ses composantes dans la base  $\mathcal{B}_B$  comme ceci :

$$(u)_A = \begin{pmatrix} x_A \\ y_A \\ z_A \end{pmatrix} \quad \text{et} \quad (u)_B = \begin{pmatrix} x_B \\ y_B \\ z_B \end{pmatrix}$$

Nous noterons  $R_{AB} = [a_{ij}]$  la matrice  $3 \times 3$  de changement de base telle que :

$$(u)_A = R_{AB} (u)_B$$

Autrement dit c'est celle qui permet de calculer les composantes dans la base  $\mathcal{B}_A$  à partir de celles dans la base  $\mathcal{B}_B$ .

## 1.2 Enchaînement de changements de bases

Si nous considérons une troisième base  $\mathcal{B}_C$  de vecteurs unitaires  $\vec{i}_C, \vec{j}_C$  et  $\vec{k}_C$ , nous pouvons écrire :

$$\begin{aligned} (u)_A &= R_{AB} (u)_B \\ (u)_B &= R_{BC} (u)_C \\ (u)_A &= R_{AB} R_{BC} (u)_C = R_{AC} (u)_C \end{aligned}$$

d'où, avec ces notations :

$$R_{AC} = R_{AB} R_{BC}$$

relation facile à mémoriser car elle rappelle la relation de Chasles  $AC = AB + BC$ .

Les matrices  $R_{CA}, R_{BA}$  et  $R_{CB}$  qui font les changements de base inverses sont notées :

$$\begin{aligned} R_{CA} &= R_{AC}^{-1} = R_{AC}^T \\ R_{BA} &= R_{AB}^{-1} = R_{AB}^T \\ R_{CB} &= R_{BC}^{-1} = R_{BC}^T \end{aligned}$$

Ce sont les matrices transposées.

On peut constater que :

$$\begin{aligned} R_{AC}^T &= R_{CA} = R_{CB} R_{BA} = R_{BC}^T R_{AB}^T \\ R_{AC}^T &= R_{BC}^T R_{AB}^T \end{aligned}$$

Toutes ces relations sont bien connues : la transposée du produit de matrices est le produit des matrices transposées effectué dans l'ordre inverse.

## 1.3 Opérateurs rotations

Considérons maintenant l'opérateur rotation  $\mathcal{R}_{AB}$  qui transforme la base  $\mathcal{B}_A$  en base  $\mathcal{B}_B$ . Il est tel que :

$$\begin{aligned} \vec{i}_B &= \mathcal{R}_{AB} (\vec{i}_A) \\ \vec{j}_B &= \mathcal{R}_{AB} (\vec{j}_A) \\ \vec{k}_B &= \mathcal{R}_{AB} (\vec{k}_A) \end{aligned}$$

## 1.4 Enchaînement d'opérateurs rotations

Considérons également l'opérateur rotation  $\mathcal{R}_{BC}$  qui transforme la base  $\mathcal{B}_B$  en base  $\mathcal{B}_C$  et l'opérateur  $\mathcal{R}_{AC}$  qui transforme la base  $\mathcal{B}_A$  en base  $\mathcal{B}_C$

L'opérateur  $\mathcal{R}_{AB}$  transforme le vecteur  $\vec{u}$  en vecteur  $\vec{v}$  et l'opérateur  $\mathcal{R}_{BC}$  transforme le vecteur  $\vec{v}$  en vecteur  $\vec{w}$ . On a donc :

$$\begin{aligned} \vec{v} &= \mathcal{R}_{AB} (\vec{u}) \\ \vec{w} &= \mathcal{R}_{BC} (\vec{v}) \\ \vec{w} &= \mathcal{R}_{BC} (\vec{v}) = \mathcal{R}_{BC} (\mathcal{R}_{AB} (\vec{u})) = \mathcal{R}_{AC} (\vec{u}) \end{aligned}$$

En utilisant l'opérateur  $\circ$  pour la composition des transformations, la dernière ligne est notée :

$$\vec{w} = \mathcal{R}_{BC} \circ \mathcal{R}_{AB} (\vec{u}) = \mathcal{R}_{AC} (\vec{u})$$

ce qui fait apparaître la relation classique de composition des opérateurs rotations :

$$\mathcal{R}_{AC} = \mathcal{R}_{BC} \circ \mathcal{R}_{AB}$$

## 1.5 Matrices des coefficients de changement de base et des composantes d'opérateur rotation

Considérons la relation vectorielle  $\vec{v} = \mathcal{R}_{AB}(\vec{u})$ . Elle peut s'écrire en composantes dans des bases quelconque  $\mathcal{B}_A, \mathcal{B}_B, \mathcal{B}_C$  ou  $\mathcal{B}_D$  comme ceci :

$$\begin{aligned}(v)_A &= [\mathcal{R}_{AB}]_A (u)_A \\ (v)_B &= [\mathcal{R}_{AB}]_B (u)_B \\ (v)_C &= [\mathcal{R}_{AB}]_C (u)_C \\ (v)_D &= [\mathcal{R}_{AB}]_D (u)_D\end{aligned}$$

relations qui font apparaître les matrices  $3 \times 3$   $[\mathcal{R}_{AB}]_A, [\mathcal{R}_{AB}]_B, [\mathcal{R}_{AB}]_C$  et  $[\mathcal{R}_{AB}]_D$  des composantes de l'opérateur rotation  $\mathcal{R}_{AB}$  dans les bases  $\mathcal{B}_A, \mathcal{B}_B, \mathcal{B}_C$  et  $\mathcal{B}_D$ .

Considérons la dernière ligne. Comme  $(u)_D = R_{DC} (u)_C$  et  $(v)_D = R_{DC} (v)_C$ , elle s'écrit :

$$R_{DC} (v)_C = [\mathcal{R}_{AB}]_D R_{DC} (u)_C$$

En pré-multipliant par  $R_{DC}^{-1} = R_{DC}^T = R_{CD}$  il vient

$$(v)_C = R_{CD} [\mathcal{R}_{AB}]_D R_{DC} (u)_C$$

en comparant avec l'avant-dernière ligne, on trouve :

$$[\mathcal{R}_{AB}]_C = R_{CD} [\mathcal{R}_{AB}]_D R_{CD}^T$$

où on a remplacé le dernier  $R_{DC}$  par  $R_{CD}^T$  pour faire apparaître la relation classique de changement de base des opérateurs.

Cette relation est à comparer avec la relation de changement de base des vecteurs :

$$(u)_C = R_{CD} (u)_D$$

Dans les deux cas, on calcule les composantes dans la base  $\mathcal{B}_C$  à partir de celles qui sont connues dans la base  $\mathcal{B}_D$ . Dans les deux cas on pré-multiplie les composantes par la matrice de changement  $R_{CD}$  mais dans le cas de l'opérateur, on post-multiplie par la matrice de changement de base inverse (ou transposée).

Jusqu'ici, tous ces résultats sont généralement bien acquis. Rappelons pour fixer les idées que les éléments des matrices  $3 \times 3$   $[\mathcal{R}_{AB}]_C$  sont appelés les **composantes** de l'opérateur rotation  $\mathcal{R}_{AB}$  dans la base  $\mathcal{B}_C$  alors que les scalaires  $a_{ij}$  de la matrice de changement de base  $R_{AB} = [a_{ij}]$  sont appelés les **coefficients** de cette matrice. *Ces éléments ne représentent pas des scalaires de même nature.* Mais il se trouve que les composantes de l'opérateur rotation  $\mathcal{R}_{AB}$  dans la base  $\mathcal{B}_A$  sont *numériquement égales* aux coefficients de la matrice de changement de base  $R_{AB}$ :

$$[\mathcal{R}_{AB}]_A = R_{AB}$$

Montrons cette identité qui n'est que numérique, entre des éléments qui ne sont pas de même nature.

$\mathcal{R}_{AB}$  transforme les vecteurs unitaires de base  $\vec{i}_A, \vec{j}_A, \vec{k}_A$  en vecteurs unitaires  $\vec{i}_B, \vec{j}_B$  et  $\vec{k}_B$ . On a donc en composantes dans la base  $\mathcal{B}_A$  :

$$\begin{aligned}(i_B)_A &= [\mathcal{R}_{AB}]_A (i_A)_A \\ (j_B)_A &= [\mathcal{R}_{AB}]_A (j_A)_A \\ (k_B)_A &= [\mathcal{R}_{AB}]_A (k_A)_A\end{aligned}$$

Mais  $(i_A)_A = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$ ,  $(j_A)_A = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$  et  $(k_A)_A = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$  ce qui montre que les colonnes de la matrice  $[\mathcal{R}_{AB}]_A$  sont égales à  $(i_B)_A, (j_B)_A$  et  $(k_B)_A$  :

$$[\mathcal{R}_{AB}]_A = [ (i_B)_A \quad (j_B)_A \quad (k_B)_A ]$$

Considérons maintenant les changements de bases suivants :

$$\begin{aligned}(i_B)_A &= R_{AB}(i_B)_B \\ (j_B)_A &= R_{AB}(j_B)_B \\ (k_B)_A &= R_{AB}(k_B)_B\end{aligned}$$

comme Mais  $(i_B)_B = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$ ,  $(j_B)_B = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$  et  $(k_B)_B = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ , les colonnes de la matrice  $R_{AB}$  sont égales à  $(i_B)_A$ ,  $(j_B)_A$  et  $(k_B)_A$  :

$$R_{AB} = [ (i_B)_A \quad (j_B)_A \quad (k_B)_A ]$$

Il y a bien identité numérique entre ces deux matrices  $[\mathcal{R}_{AB}]_A$  et  $R_{AB}$  qui ne représente pas les mêmes éléments.

Et maintenant, un résultat étonnant, mais qui s'explique physiquement :

$$[\mathcal{R}_{AB}]_A = [\mathcal{R}_{AB}]_B$$

Les composantes de l'opérateur qui transforme  $\mathcal{B}_A$  en  $\mathcal{B}_B$  sont les mêmes dans ces deux bases. Il suffit pour le démontrer d'appliquer la formule de changement de bases des opérateurs :

$$[\mathcal{R}_{AB}]_B = R_{BA} [\mathcal{R}_{AB}]_A R_{AB}$$

Mais puisque  $R_{BA} = R_{AB}^{-1} = [\mathcal{R}_{AB}]_A^{-1}$ , d'où :

$$[\mathcal{R}_{AB}]_B = R_{AB} = [\mathcal{R}_{AB}]_A$$

Bien qu'étonnante, l'identité :

$$[\mathcal{R}_{AB}]_B = [\mathcal{R}_{AB}]_A$$

se comprend physiquement. L'opérateur  $\mathcal{R}_{AB}$  représente une rotation d'un certain angle  $\alpha$  autour d'un vecteur unitaire  $\vec{u}$ . On peut donc écrire :

$$\mathcal{R}_{AB} = \mathcal{R}(\alpha, \vec{u})$$

et exprimer les matrices des composantes en fonction de ces éléments :  $[\mathcal{R}_{AB}]_A = [\mathcal{R}(\alpha, (u)_A)]$  et  $[\mathcal{R}_{AB}]_B = [\mathcal{R}(\alpha, (u)_B)]$ . Si  $\alpha = 0$  les bases  $\mathcal{B}_A$  et  $\mathcal{B}_B$  sont confondues. Dans ce cas on a  $(u)_A = (u)_B$ . Lorsque  $\alpha$  croit la direction  $\vec{u}$  est fixe dans  $\mathcal{B}_A$  (autrement dit  $(u)_A = cte$ ) et  $\mathcal{B}_B$  tourne autour de cette direction de l'angle  $\alpha$ . Mais on peut aussi considérer que  $\vec{u}$  est fixe dans  $\mathcal{B}_B$  (autrement dit  $(u)_B = cte$ ) et que  $\mathcal{B}_A$  tourne autour de cette direction de l'angle  $-\alpha$ . En fait quelque soit  $\alpha$ , la direction  $\vec{u}$  est la direction unique qui est fixe dans les deux bases, et on a  $(u)_A = (u)_B$  en permanence, quelque soit  $\alpha$ . Il en résulte que  $[\mathcal{R}(\alpha, (u)_A)] = [\mathcal{R}(\alpha, (u)_B)]$  et que  $[\mathcal{R}_{AB}]_A = [\mathcal{R}_{AB}]_B$ .

## 1.6 Enchaînement des rotations

Considérons un enchaînement de rotations transformant les bases  $\mathcal{B}_A$  en  $\mathcal{B}_B$  puis  $\mathcal{B}_B$  en  $\mathcal{B}_C$ , puis  $\mathcal{B}_C$  en  $\mathcal{B}_D$  qui transforme également un vecteur  $\vec{u}$  en vecteur  $\vec{v}$ . On a :

$$\vec{v} = \mathcal{R}_{CD}(\mathcal{R}_{BC}(\mathcal{R}_{AB}(\vec{u}))) = \mathcal{R}_{CD} \circ \mathcal{R}_{BC} \circ \mathcal{R}_{AB} \circ \vec{u}$$

Écrivons cette relation en composantes dans le base  $\mathcal{B}_A$ . Il vient :

$$(v)_A = [\mathcal{R}_{CD}]_A [\mathcal{R}_{BC}]_A [\mathcal{R}_{AB}]_A (u)_A$$

C'est l'écriture théorique telle qu'on la trouve dans les cours de mathématiques, avec les matrices de rotation dans l'ordre inverse des opérations, et c'est cette formule qui reste dans la mémoire des personnes qui ne font pas les calculs numériques pratiques associés à ces opérations. Dans cette relation, on sait les exprimer les composantes de

$$[\mathcal{R}_{AB}]_A = R_{AB} = [ (i_B)_A \quad (j_B)_A \quad (k_B)_A ]$$

Par contre pour obtenir celles de  $[\mathcal{R}_{BC}]_A$  et  $[\mathcal{R}_{CD}]_A$  à partir des matrices  $[\mathcal{R}_{BC}]_B$  et  $[\mathcal{R}_{CD}]_C$  que l'on sait calculer, on doit procéder à des changements de bases :

$$\begin{aligned} [\mathcal{R}_{BC}]_A &= R_{AB} [\mathcal{R}_{BC}]_B R_{BA} = R_{AB} R_{BC} R_{BA} \\ [\mathcal{R}_{CD}]_A &= R_{AC} [\mathcal{R}_{CD}]_C R_{CA} = R_{AB} R_{BC} [\mathcal{R}_{CD}]_C R_{CB} R_{BA} = R_{AB} R_{BC} R_{CD} R_{CB} R_{BA} \end{aligned}$$

Ainsi, il vient :

$$(v)_A = R_{AB} R_{BC} R_{CD} R_{CB} R_{BA} R_{AB} R_{BC} R_{BA} R_{AB} (u)_A$$

soit après simplification :

$$(v)_A = R_{AB} R_{BC} R_{CD} (u)_A$$

que l'on peut également écrire sous la forme :

$$(v)_A = [\mathcal{R}_{AB}]_A [\mathcal{R}_{BC}]_B [\mathcal{R}_{CD}]_C (u)_A$$

Sous cette forme, où l'ordre des matrices est le même que l'ordre des opérations, on sait exprimer directement les diverses matrices qui composent l'enchaînement.

Finalement, on a deux formes pour exprimer un enchaînement de matrices de rotations :

$$\begin{aligned} [R_{AD}]_A &= [\mathcal{R}_{CD}]_A [\mathcal{R}_{BC}]_A [\mathcal{R}_{AB}]_A \\ [R_{AD}]_A &= [\mathcal{R}_{AB}]_A [\mathcal{R}_{BC}]_B [\mathcal{R}_{CD}]_C \end{aligned}$$

La première forme est celle qui figure dans les cours de mathématiques. En effet, on y fait des calculs théoriques sans se soucier des valeurs numériques. On n'a pas besoin de calculer effectivement les valeurs numériques des composantes, et pour démontrer des formules, on sous-entend une base unique  $\mathcal{B}_0$  et c'est dans cette base unique que toutes les composantes seraient éventuellement exprimées. L'ordre de la multiplication des matrices se fait de droite à gauche dans l'ordre de l'application des opérateurs.

La deuxième forme est celle qui est utilisée dans les programmes de modélisation 3D où on travaille sur des valeurs numériques, et l'ordre utilisé est ordre composite qui correspond aussi à l'ordre de l'application des opérateurs mais de gauche à droite cette fois-ci.

## 1.7 Les matrices de rotation élémentaires

Pour calculer les composantes de  $[\mathcal{R}_{AB}]_A = R_{AB} = [ (i_B)_A \quad (j_B)_A \quad (k_B)_A ]$  on considère les 3 cas particuliers où la rotation qui transforme  $\mathcal{B}_A$  en  $\mathcal{B}_B$  correspond à une rotation autour d'un des 3 vecteurs de la base  $\mathcal{B}_A$  (qui restera fixe dans les deux bases). Les expressions pour le cas où cette rotation se fait autour d'un vecteur quelconque  $\vec{u}$  sont présentées dans [https://libre.fr/michel/copie\\_cert/DCSD-2009\\_008-NOT-003-1.1.pdf](https://libre.fr/michel/copie_cert/DCSD-2009_008-NOT-003-1.1.pdf).

Nous considérerons les 3 matrices de rotation élémentaires suivantes :

$$R_X(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha \\ 0 & \sin \alpha & \cos \alpha \end{pmatrix} \quad R_Y(\alpha) = \begin{pmatrix} \cos \alpha & 0 & \sin \alpha \\ 0 & 1 & 0 \\ -\sin \alpha & 0 & \cos \alpha \end{pmatrix} \quad R_Z(\alpha) = \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Nous aurions pu associer les expressions  $R_X(\alpha)$ ,  $R_Y(\alpha)$  et  $R_Z(\alpha)$  aux transposées de ces matrices, comme le font d'autres auteurs. Les deux choix sont arbitraires et strictement équivalents, comme la conduite à droite ou la conduite à gauche, mais lorsque le choix est fait il faut bien le respecter car il conditionne tout ce qui suit.

Avec ce choix :

- $[\mathcal{R}_{AB}]_A = R_{AB} = R_X(\alpha)$ ,  $\mathcal{B}_B$  se déduit de  $\mathcal{B}_A$  par une rotation d'angle  $\alpha$  autour de  $\vec{i}_A = \vec{i}_B$  (qui reste inchangé).
- $[\mathcal{R}_{AB}]_A = R_{AB} = R_Y(\alpha)$ ,  $\mathcal{B}_B$  se déduit de  $\mathcal{B}_A$  par une rotation d'angle  $\alpha$  autour de  $\vec{j}_A = \vec{j}_B$  (qui reste inchangé).
- $[\mathcal{R}_{AB}]_A = R_{AB} = R_Z(\alpha)$ ,  $\mathcal{B}_B$  se déduit de  $\mathcal{B}_A$  par une rotation d'angle  $\alpha$  autour de  $\vec{k}_A = \vec{k}_B$  (qui reste inchangé).

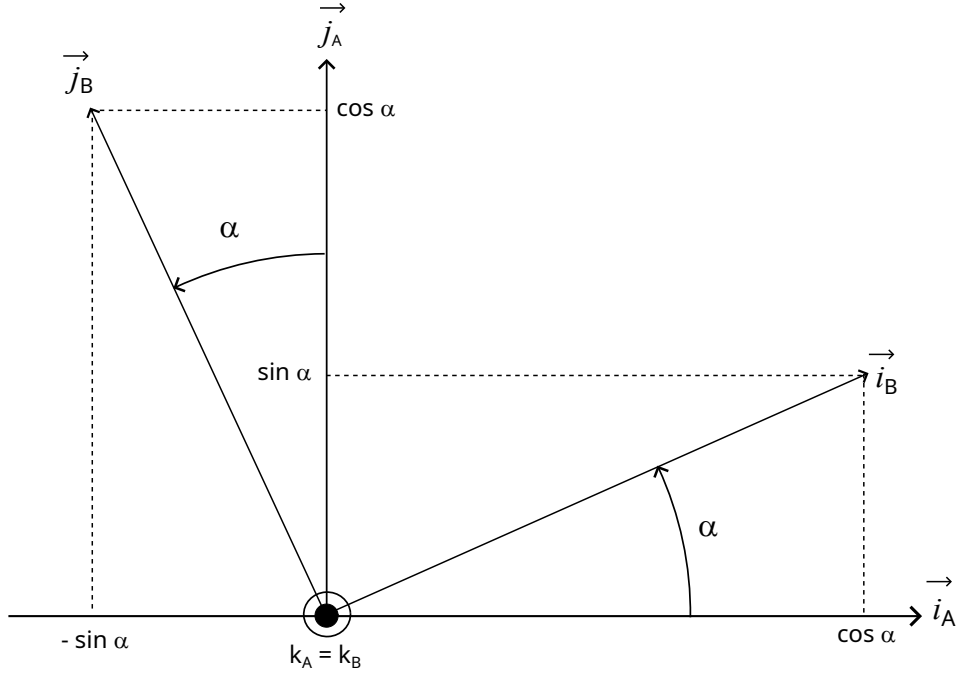


Figure 1: Changement de base élémentaire

Pour s'en persuader, considérons le dernier cas représenté sur la figure 1 :

Sur cette figure, tracée dans le plan orthogonal à  $\vec{k}_A = \vec{k}_B$ , il est facile de voir que :

$$(i_B)_A = \begin{pmatrix} \cos \alpha \\ \sin \alpha \\ 0 \end{pmatrix} \quad (j_B)_A = \begin{pmatrix} -\sin \alpha \\ \cos \alpha \\ 0 \end{pmatrix} \quad \text{et} \quad (k_B)_A = (k_A)_A = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

Comme  $[\mathcal{R}_{AB}]_A = R_{AB} = [ (i_B)_A \quad (j_B)_A \quad (k_B)_A ]$ , on retrouve bien  $R_Z(\alpha)$ .

## 1.8 Les micro-rotations

On appelle micro-rotation, d'un angle  $\varepsilon$  assez petit pour pouvoir considérer, au second ordre près que  $\sin \varepsilon \simeq \varepsilon$  et  $\cos \varepsilon \simeq 1$ . Alors que les rotations successives ne sont pas commutatives (l'ordre des rotations est significatif), **les micro-rotations sont commutatives**, c'est-à-dire que le résultat est indépendant de l'ordre dans lequel on les applique. Considérons par exemple le produit des 3 micro-rotations suivantes :

$$R_X(\varepsilon_X)R_Y(\varepsilon_Y)R_Z(\varepsilon_Z) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -\varepsilon_X \\ 0 & \varepsilon_X & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & \varepsilon_Y \\ 0 & 1 & 0 \\ -\varepsilon_Y & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -\varepsilon_Z & 0 \\ \varepsilon_Z & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & -\varepsilon_Z & \varepsilon_Y \\ \varepsilon_Z & 1 & -\varepsilon_X \\ -\varepsilon_Y & \varepsilon_X & 1 \end{pmatrix}$$

En changeant l'ordre des 3 rotations, on trouve toujours le même résultat (*en négligeant les termes d'ordre supérieur ou égal à deux*). On peut donc poser :

$$R_X(\varepsilon_X)R_Y(\varepsilon_Y)R_Z(\varepsilon_Z) = R_Z(\varepsilon_Z)R_X(\varepsilon_X)R_Y(\varepsilon_Y) = \dots = R_{XYZ}(\varepsilon_X, \varepsilon_Y, \varepsilon_Z) = \begin{pmatrix} 1 & -\varepsilon_Z & \varepsilon_Y \\ \varepsilon_Z & 1 & -\varepsilon_X \\ -\varepsilon_Y & \varepsilon_X & 1 \end{pmatrix}$$

## 1.9 Remarque

Comme nous l'avons signalé en introduction, certains auteurs utilisent un sens (matrices élémentaires transposées) et un ordre inverse au notre, ce qui est strictement équivalent. Il suffit que sens et ordre choisis soient cohérents entre eux.

## 2 Erreur de pointage d'un télescope

Nous considérons un télescope monté sur une monture équatoriale qui vise une étoile d'ascension droite  $\alpha$  et de déclinaison  $\delta$ .

### 2.1 Pointage équatorial idéal

Notons  $\mathcal{B}_0$  la base d'un repère équatorial avec l'axe  $Z_0$  vers le pôle céleste Nord et l'axe  $X_0$  dirigé vers le point  $\gamma$ . Pour amener l'axe  $X_0$  en direction d'une étoile  $E$  de coordonnées équatoriales  $(\alpha, \delta)$  on peut opérer une rotation  $R_Z(\alpha)$  qui transforme la base  $\mathcal{B}_0$  en base  $\mathcal{B}_1$  suivie d'une rotation  $R_Y(-\delta)$  qui transforme la base  $\mathcal{B}_1$  en base  $\mathcal{B}_E$  amenant l'axe  $X_1$  transformé de  $X_0$  dans la direction  $\overrightarrow{OE}$  de l'étoile. Ainsi le produit des rotations

$$R_{ideal} = R_Z(\alpha)R_Y(-\delta)$$

amène le vecteur  $X_0$  selon la direction  $\overrightarrow{OE}$  de l'étoile. Les composantes de  $\overrightarrow{OE}$  dans  $\mathcal{B}_E$  sont données par :

$$(e)_0 = \begin{pmatrix} \cos \alpha & -\sin \alpha & 0 \\ \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos \delta & 0 & -\sin \delta \\ 0 & 1 & 0 \\ \sin \delta & 0 & \cos \delta \end{pmatrix} (e)_E$$

où  $\vec{e}$  est le vecteur de base  $X$  de  $\mathcal{B}_E$  qui est donc tel que  $(e)_E = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$ . D'où :

$$(e)_0 = \begin{pmatrix} \cos \alpha \cos \delta \\ \sin \alpha \cos \delta \\ \sin \delta \end{pmatrix}$$

où on reconnaît les cosinus directeurs de la direction de l'étoile dans la base équatoriale. Cette approche est un peu lourde et laborieuse, mais elle sera utile pour la suite.

### 2.2 Pointage local parfait

Notre télescope est situé à la latitude  $\varphi$  sur un méridien qui fait un angle  $T_s$  (temps sidéral local) avec le méridien origine qui contient le point  $\gamma$  par lequel passe l'axe  $X_0$ . Notre base locale  $\mathcal{B}_L$  à son axe  $Z_L$  selon la verticale locale, en direction du zénith et son axe  $X_L$  à l'intersection du plan horizontal local et du méridien, en direction du Sud. Une rotation d'angle  $T_S$  autour de  $Z_0$  transforme  $\mathcal{B}_0$  en  $\mathcal{B}_2$  avec et une rotation d'angle  $\frac{\pi}{2} - \varphi$  autour de  $Y_2$  (non représenté sur la figure 2) amène  $\mathcal{B}_2$  selon  $\mathcal{B}_L$  :

$$\begin{aligned} \mathcal{B}_0 &\xrightarrow{R_Z(T_S)} \mathcal{B}_2 \\ \mathcal{B}_2 &\xrightarrow{R_Y(\frac{\pi}{2}-\varphi)} \mathcal{B}_L \end{aligned}$$

Notons  $\mathcal{B}_M$  la base associée à la monture. La mise en station consiste à aligner son axe  $Z_M$  avec l'axe  $Z_0$ . Pour cela, il suffit de faire tourner la base locale  $\mathcal{B}_L$  de l'angle  $\varphi - \frac{\pi}{2}$  autour de  $Y_L = Y_2$ , rotation inverse de la dernière rotation considérée :

$$\mathcal{B}_L \xrightarrow{R_Y(\varphi-\frac{\pi}{2})} \mathcal{B}_M // \mathcal{B}_2$$

En l'absence d'erreurs  $\mathcal{B}_M$  est parallèle à  $\mathcal{B}_2$ . Dans la base  $\mathcal{B}_M$ , en position  $H = 0$  (angle horaire nul),  $\delta = 0$  (déclinaison nulle), le télescope est selon  $\vec{i}_M$  (c'est-à-dire selon  $X_2$ ). Pour qu'il pointe l'étoile, on lui fait subir la rotation  $R_Z(-H)$  qui amène  $\vec{i}_M$  parallèle à  $X_1$  puis la rotation  $R_Y(-\delta)$  qui transforme la base  $\mathcal{B}'_1$  en base  $\mathcal{B}_T$  amenant  $\vec{i}_M$  dans la direction  $\overrightarrow{OE'}$  de l'étoile :

$$\begin{aligned} \mathcal{B}_M &\xrightarrow{R_Z(-H)} \mathcal{B}'_1 \\ \mathcal{B}'_1 &\xrightarrow{R_Y-\delta} \mathcal{B}_T \end{aligned}$$

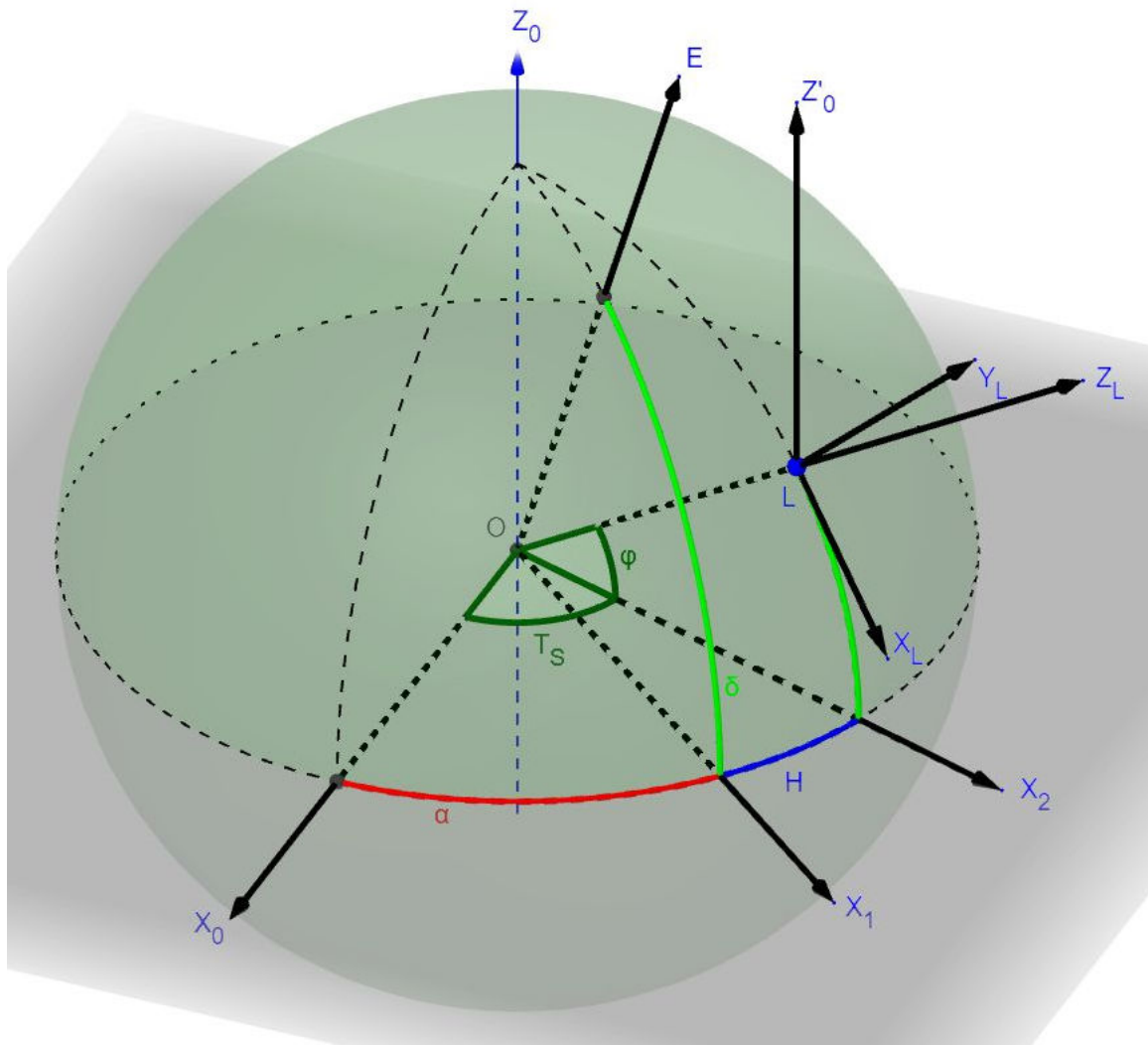


Figure 2: Les principaux repères



Il en résulte que les cosinus directeurs de la direction pointée sont donnés par :

$$(e')_0 = R_Z(T_S) R_Y\left(\frac{\pi}{2} - \varphi\right) R_Y\left(\varphi - \frac{\pi}{2}\right) R_Z(-H) R_Y(-\delta) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Or  $R_Y\left(\frac{\pi}{2} - \varphi\right) R_Y\left(\varphi - \frac{\pi}{2}\right) = I_d$  et  $R_Z(T_S) R_Z(-H) = R_Z(T_S - H) = R_Z(\alpha)$ , D'où

$$(e')_0 = R_Z(\alpha) R_Y(-\delta) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = (e)_0$$

En l'absence d'erreurs le pointage local donne le même résultat que le pointage idéal.

### 2.3 Modélisation des erreurs

Lors de l'orientation de la base  $\mathcal{B}_M$  de la monture on peut introduire les erreurs suivantes :

- $\varepsilon_R$  angle d'erreur d'horizontalité autour de  $X_L$  (R comme roulis),
- $\varepsilon_Y$  angle d'erreur d'horizontalité autour de  $Y_L$
- $\varepsilon_A$  angle d'erreur d'azimut autour de  $Z_L$

Par ailleurs dans les renseignements que l'on fournit à la monture pour déterminer la direction du zénith on peut faire une erreur  $\varepsilon_\varphi$  sur la latitude, qui va s'ajouter à  $\varepsilon_Y$  sans qu'il soit possible de distinguer l'un de l'autre. En conséquence, nous ne considérerons que leur somme que nous noterons  $\varepsilon_Y$ .

Lors du pointage local on peut introduire les erreurs suivantes :

- $\varepsilon_H$  une erreur sur l'heure fournie à la monture qui va se cumuler à l'erreur de zéro du moteur d'ascension et qui de plus est sujette au jeu sur cet axe. Cette erreur vient modifier la valeur de  $H$  dans  $R_Z(-H - \varepsilon_H)$ .
- $\varepsilon_D$  l'erreur de zéro du moteur de déclinaison qui se cumule avec le jeu de ce moteur. Cette erreur vient modifier la valeur de  $\delta$  dans  $R_Y(-\delta - \varepsilon_D)$ .

Enfin l'erreur d'alignement du télescope avec son axe théorique. Si on considère les 3 micro-rotations théoriques possibles, celle autour de l'axe optique du télescope n'est pas significative. Reste les deux erreurs en rotation autour de 2 axes orthogonaux à l'axe optique. L'un vient modifier la valeur de la déclinaison. Il sera donc inclus dans  $\varepsilon_D$  sans que l'on puisse faire la part entre les deux, et qui peut être compensée en modifiant l'angle de déclinaison. L'autre que nous noterons  $\varepsilon_C$  empêchera d'atteindre le pointage polaire laissant un cône inatteignable de cet angle au sommet. Il correspond à une micro-rotation  $R_X(\varepsilon_C)$  qui s'intercale entre  $R_Z(-H - \varepsilon_H)$  et  $R_Y(-\delta - \varepsilon_D)$ .

Finalement, les cosinus directeurs de la direction pointée seront donnés par :

$$(e')_0 = R_Z(T_S) R_Y\left(\frac{\pi}{2} - \varphi\right) R_{XYZ}(\varepsilon_R, \varepsilon_Y, \varepsilon_A) R_Y\left(\varphi - \frac{\pi}{2}\right) R_Z(-H - \varepsilon_H) R_X(\varepsilon_C) R_Y(-\delta - \varepsilon_D) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Calculons ces coordonnées dans le repère  $B_E$ . Il vient :

$$(e')_E = R_Y(\delta) R_Z(-\alpha) (e')_0$$

soit :

$$(e')_E = R_Y(\delta) R_Z(-\alpha) R_Z(T_S) R_Y\left(\frac{\pi}{2} - \varphi\right) R_{XYZ}(\varepsilon_R, \varepsilon_Y, \varepsilon_A) R_Y\left(\varphi - \frac{\pi}{2}\right) R_Z(-H - \varepsilon_H) R_X(\varepsilon_C) R_Y(-\delta - \varepsilon_D) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

or  $T_S - \alpha = H$ , d'où :

$$(e')_E = R_Y(\delta) R_Z(H) R_Y\left(\frac{\pi}{2} - \varphi\right) R_{XYZ}(\varepsilon_R, \varepsilon_Y, \varepsilon_A) R_Y\left(\varphi - \frac{\pi}{2}\right) R_Z(-H - \varepsilon_H) R_X(\varepsilon_C) R_Y(-\delta - \varepsilon_D) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Dans la base  $\mathcal{B}_E$  si le pointage est parfait, on doit obtenir  $(e')_E = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$ . En présence d'erreurs on va

obtenir  $(e')_E = \begin{pmatrix} 1 + \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}$ . L'erreur  $\Delta x$ , si elle existe n'a pas de signification. L'erreur  $\Delta y$  correspond à une rotation de la direction  $X_E$  autour  $Z_0$  (pondérée par le bras de levier  $\cos \delta$ ) et l'erreur, c'est-à-dire à une erreur en ascension droite pondérée par  $\cos \delta$ . L'erreur  $\Delta z$  correspond à une rotation de la direction  $X_E$  autour de  $Y_1$  c'est-à-dire à une erreur en déclinaison. Ainsi on peut écrire :

$$\begin{pmatrix} 1 + \Delta x \\ \Delta \alpha \cos \delta \\ \Delta \delta \end{pmatrix} = R_Y(\delta) R_Z(H) R_Y\left(\frac{\pi}{2} - \varphi\right) R_{XYZ}(\varepsilon_R, \varepsilon_Y, \varepsilon_A) R_Y\left(\varphi - \frac{\pi}{2}\right) R_Z(-H - \varepsilon_H) R_X(\varepsilon_C) R_Y(-\delta - \varepsilon_D) \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

Seules les lignes 2 et 3 sont à significatives.

### 3 Identification des erreurs

Pour tenter d'identifier les 6 inconnues  $\varepsilon_H, \varepsilon_D, \varepsilon_A, \varepsilon_Y, \varepsilon_R, \varepsilon_C$  on procède à la linéarisation de l'expression des erreurs.

#### 3.1 Linéarisation de l'erreur de pointage

Exprimons les différents éléments de l'expression précédente. Il vient :

$$\begin{pmatrix} 1 + \Delta x \\ \Delta \alpha \cos \delta \\ \Delta \delta \end{pmatrix} = \begin{pmatrix} \cos \delta & 0 & \sin \delta \\ 0 & 1 & 0 \\ -\sin \delta & 0 & \cos \delta \end{pmatrix} \begin{pmatrix} \cos H & -\sin H & 0 \\ \sin H & \cos H & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \sin \varphi & 0 & \cos \varphi \\ 0 & 1 & 0 \\ -\cos \varphi & 0 & \sin \varphi \end{pmatrix} \times \\ \begin{pmatrix} 1 & -\varepsilon_A & \varepsilon_Y \\ \varepsilon_A & 1 & -\varepsilon_R \\ -\varepsilon_Y & \varepsilon_R & 1 \end{pmatrix} \begin{pmatrix} \sin \varphi & 0 & -\cos \varphi \\ 0 & 1 & 0 \\ \cos \varphi & 0 & \sin \varphi \end{pmatrix} \begin{pmatrix} \cos(H + \varepsilon_H) & \sin(H + \varepsilon_H) & 0 \\ -\sin(H + \varepsilon_H) & \cos(H + \varepsilon_H) & 0 \\ 0 & 0 & 1 \end{pmatrix} \times \\ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -\varepsilon_C \\ 0 & \varepsilon_C & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -\varepsilon_C \\ 0 & \varepsilon_C & 1 \end{pmatrix} \begin{pmatrix} \cos(\delta + \varepsilon_D) & 0 & -\sin(\delta + \varepsilon_D) \\ 0 & 1 & 0 \\ \sin(\delta + \varepsilon_D) & 0 & \cos(\delta + \varepsilon_D) \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

On effectue ces produits. Dans les résultats intermédiaires on considère que  $\sin \varepsilon \simeq \varepsilon$  et  $\cos \varepsilon \simeq 1$  et on ne conserve que les termes du premier ordre. Tous calculs faits on obtient les relations suivantes :

$$\begin{aligned} \Delta x &= 0 \\ \Delta \alpha \cos \delta &= (\varepsilon_A \sin \varphi - \varepsilon_R \cos \varphi) \cos \delta - (\varepsilon_A \cos \varphi + \varepsilon_R \sin \varphi) \cos H \sin \delta + \varepsilon_Y \sin H \sin \delta - \varepsilon_H \cos \delta - \varepsilon_C \sin \delta \\ \Delta \delta &= -(\varepsilon_A \cos \varphi + \varepsilon_R \sin \varphi) \sin H - \varepsilon_Y \cos H + \varepsilon_D \end{aligned}$$

En posant :

$$\begin{aligned} \varepsilon_Z &= \varepsilon_A \sin \varphi - \varepsilon_R \cos \varphi \\ \varepsilon_X &= \varepsilon_A \cos \varphi + \varepsilon_R \sin \varphi \end{aligned}$$

qui, avec  $\varepsilon_Y$  sont les 3 composantes de l'erreur d'alignement polaire de la monture dans la base  $\mathcal{B}_2$ , on obtient :

$$\begin{aligned} \Delta \alpha \cos \delta &= \varepsilon_Z \cos \delta - \varepsilon_X \cos H \sin \delta + \varepsilon_Y \sin H \sin \delta - \varepsilon_H \cos \delta - \varepsilon_C \sin \delta \\ \Delta \delta &= -\varepsilon_X \sin H - \varepsilon_Y \cos H + \varepsilon_D \end{aligned}$$

que l'on peut mettre sous la forme matricielle suivante :

$$\begin{pmatrix} \Delta\alpha \cos \delta \\ \Delta\delta \end{pmatrix} = \begin{pmatrix} -\cos H \sin \delta & \sin H \sin \delta & \cos \delta & -\sin \delta & -\cos \delta & 0 \\ -\sin H & -\cos H & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \varepsilon_X \\ \varepsilon_Y \\ \varepsilon_Z \\ \varepsilon_C \\ \varepsilon_H \\ \varepsilon_D \end{pmatrix}$$

Sous cette forme on voit que la colonne 3 :  $\begin{pmatrix} \cos \delta \\ 0 \end{pmatrix}$  de la matrice de sensibilité est proportionnelle à la colonne 5 :  $\begin{pmatrix} -\cos \delta \\ 0 \end{pmatrix}$ , ce qui signifie que les erreurs inconnues  $\varepsilon_Z$  et  $\varepsilon_H$  sont liées. Effectivement ces deux erreurs correspondent toutes deux à une rotation autour de l'axe  $Z_0 = Z_1 = Z_2$ . On ne peut donc les différencier. En revenant aux 6 inconnues initiales,  $\varepsilon_H, \varepsilon_D, \varepsilon_A, \varepsilon_Y, \varepsilon_R, \varepsilon_C$ , il en résulte que les 3 inconnues  $\varepsilon_H, \varepsilon_A$  et  $\varepsilon_R$  sont liées. On ne peut les identifier toutes les trois. Il faut donc en supprimer une. J'ai choisi de supprimer  $\varepsilon_R$ , l'angle d'erreur d'horizontalité en rotation autour de l'horizontale Nord-Sud.

Avec les 5 inconnues  $\varepsilon_H, \varepsilon_D, \varepsilon_A, \varepsilon_Y, \varepsilon_C$  l'expression matricielle de l'erreur s'écrit :

$$\begin{pmatrix} \Delta\alpha \cos \delta \\ \Delta\delta \end{pmatrix} = \begin{pmatrix} -\cos \delta & 0 & (\sin \varphi \cos \delta - \cos \varphi \cos H \sin \delta) & \sin H \sin \delta & -\sin \delta \\ 0 & 1 & -\cos \varphi \sin H & -\cos H & 0 \end{pmatrix} \begin{pmatrix} \varepsilon_H \\ \varepsilon_D \\ \varepsilon_A \\ \varepsilon_Y \\ \varepsilon_C \end{pmatrix}$$

### 3.2 Méthode d'identification

L'expression précédente se met sous la forme  $Y_i = A_i X$  où :

- $Y_i = \begin{pmatrix} \Delta\alpha \cos \delta \\ \Delta\delta \end{pmatrix}_i$  est la  $i$ -ième mesure de l'erreur de pointage,
- $A_i = \begin{pmatrix} -\cos \delta & 0 & (\sin \varphi \cos \delta - \cos \varphi \cos H \sin \delta) & \sin H \sin \delta & -\sin \delta \\ 0 & 1 & -\cos \varphi \sin H & -\cos H & 0 \end{pmatrix}_i$  est la  $i$ -ième matrice de mesure
- $X = \begin{pmatrix} \varepsilon_H \\ \varepsilon_D \\ \varepsilon_A \\ \varepsilon_Y \\ \varepsilon_C \end{pmatrix}$  est le vecteur des 5 inconnues.

Si on dispose de  $N$  mesures, le système à résoudre se met sous la forme :

$$Y = AX \text{ avec } Y = \begin{pmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_N \end{pmatrix} \text{ et } A = \begin{pmatrix} A_1 \\ A_2 \\ \vdots \\ A_N \end{pmatrix}$$

#### 3.2.1 Résolution classique des moindres carrés (cas sur-déterminé)

$Y$  est un vecteur à  $2N$  composantes et  $A$  une matrice  $2N \times 5$ .

Si  $2N > 5$ , c'est-à-dire si  $N \geq 3$  il y a plus d'équations que d'inconnues. Le système étant sur-déterminé, on cherche une solution approchée qui minimise l'erreur quadratique  $c = \frac{1}{2} (Y - AX)^T (Y - AX)$ . Elle annule le gradient  $\frac{\partial c}{\partial X} = A^T AX - A^T Y = 0$ , d'où la solution dite des moindres carrés :

$$X = (A^T A)^{-1} A^T Y$$

qui doit résoudre un système  $6 \times 6$ .

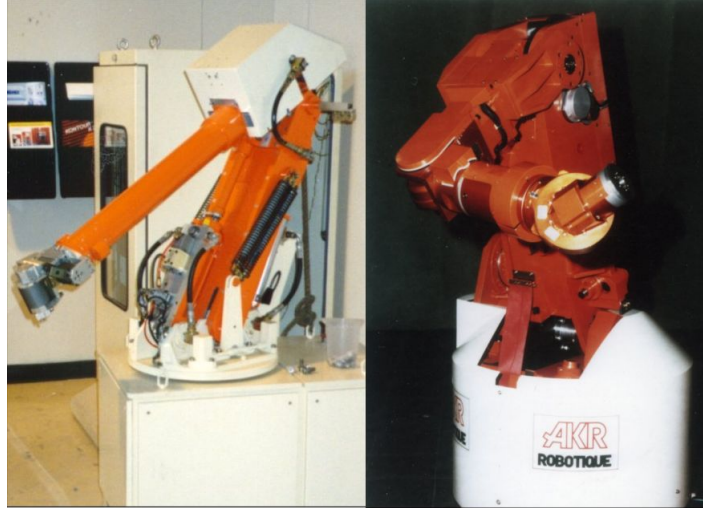


Figure 3: Robots AKR 3000 et C4000

### 3.2.2 Cas particulier sous-déterminé

Si  $2N < 5$ , c'est-à-dire si  $N \leq 2$  il y a plus d'inconnues que d'équations. Il y a donc une infinité de solutions. Dans cette infinité on choisit la solution de norme minimale  $X^T X$  sous la contrainte  $(AX - Y)^T \lambda = 0$ . On considère le critère  $c = \frac{1}{2} X^T X + (AX - Y)^T \lambda$  et on annule son gradient  $\frac{\partial c}{\partial X} = X - A^T \lambda = 0$ . En reportant  $X = A^T \lambda$  dans la contrainte  $Y = AX = AA^T \lambda$ , on en déduit  $\lambda = (AA^T)^{-1} Y$ . D'où la solution dite de **Moore-Penrose** :

$$X = A^T (AA^T)^{-1} Y$$

qui doit résoudre un système  $2N \times 2N$ , c'est-à-dire  $4 \times 4$  lorsque  $N = 2$  et  $2 \times 2$  lorsque  $N = 1$ . Dans ces deux derniers cas, la méthode, qui est destinée à identifier les erreurs de la monture, perd tout son intérêt. Par contre elle peut être intéressante pour évaluer au mieux  $\varepsilon_H$  et  $\varepsilon_D$  pour un pointage ultérieur.

## 4 Mise en œuvre informatique

Dans les années 1980 j'avais mis en œuvre cette méthode pour identifier les paramètres et les erreurs sur les paramètres de construction des robots AKR-3000 (6 ddl) et AKR-C-4000 (7 ddl) photographiés Figure 3.

Cela avait permis d'améliorer significativement la précision de positionnement de ces robots. A l'époque la programmation était effectuée en langage C. Dans ce document je présente une version écrite en Octave (très proche de Matlab) qui est un langage particulièrement approprié aux opérations matricielles.

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Test de la méthode d'identification des erreurs monture avec un jeu de
% données artificielles. Il prend en entrée le vecteur des 6 écarts inconnus
% qui sont à identifier, dans l'ordre :
% ErA_dg, ErS_dg, ErR_dg, ErC_dg, ErH_dg, ErD_dg
% L'ecart ErR n'est pas recherché. Il est considéré nul par la méthode.
% Il en résulte que les écarts ErA et ErH trouvés incluent sa contribution.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function simuestim(ecarts_dg)
if nargin < 1 % Jeu d'erreurs test si appel de simuestim sans argument
    ErA_dg = 0.7; % Erreur d'azimut/Oz (zenith)
    ErS_dg = 1.1; % Erreur de site/Oy (horizon Est)
    ErR_dg = -1.; % Erreur de niveau/Ox (horizon Sud)
    ErC_dg = 0.9; % Erreur de conicité/Ox équatorial apres la rot AD)
    ErH_dg = 1.9; % Erreur de zéro en angle horaire
    ErD_dg = 1.5; % Erreur de zéro en Déclinaison
    ErR_dg *= 0; % Annulé pour trouver des ErA et ErH proches des donnés
    ecarts_dg = [ErA_dg, ErS_dg, ErR_dg, ErC_dg, ErH_dg, ErD_dg];
endif
disp(["Erreurs inconnues en degres: " num2str(ecarts_dg)]);
% Distribution des écarts en radians
errs = ecarts_dg*pi/180;

```

```

Ea = errs(1); Es = errs(2); En = errs(3);
Ec = errs(4); Eh = errs(5); Ed = errs(6);
%Coordonnees du site d'observation
lati = 45*pi/180;
cl = cos(lati); sl = sin(lati);
% LES COORDONNEES DES 9 ETOILES UTILISEES
% Dubhe
adr(1) = (11 + 5/60 + 6.13/3600)*pi/12;
dec(1) = (61 + 37/60 + 52.6/3600)*pi/180;
% Capella
adr(2) = (5 + 18/60 + 20.84/3600)*pi/12;
dec(2) = (46 + 1/60 + 10.6/3600)*pi/180;
% Mirfak
adr(3) = (3 + 25/60 + 55.68/3600)*pi/12;
dec(3) = (49 + 56/60 + 24.1/3600)*pi/180;
% Caph
adr(4) = (0 + 10/60 + 21.45/3600)*pi/12;
dec(4) = (59 + 16/60 + 42.1/3600)*pi/180;
% Alderamin
adr(5) = (21 + 19/60 + 6.04/3600)*pi/12;
dec(5) = (62 + 40/60 + 44.6/3600)*pi/180;
% Deneb
adr(6) = (20 + 42/60 + 11.51/3600)*pi/12;
dec(6) = (45 + 21/60 + 36.3/3600)*pi/180;
% Vega
adr(7) = (18 + 37/60 + 41.57/3600)*pi/12;
dec(7) = (38 + 48/60 + 16.5/3600)*pi/180;
% Alkaid
adr(8) = (13 + 48/60 + 25.58/3600)*pi/12;
dec(8) = (49 + 12/60 + 5.0/3600)*pi/180;
% Arcturus
adr(9) = (14 + 16/60 + 41/3600)*pi/12;
dec(9) = (19 + 03/60 + 53/3600)*pi/180;

%Formation des matrices à partir des 9 étoiles
Nval = 9;
for k = 1:Nval
    % Temps sidéral local arbitraire à l'instant de la mesure
    lst = (1+k/60)*pi/12;
    ad = adr(k); de = dec(k);
    sb = sin(de); cb = cos(de);
    ah = lst - ad; sh = sin(ah); ch = cos(ah);

    % Erreur de pointage monture réelle en radians
    uvw = Rx(En)*Rz(Ea)*Ry(Es+lati-pi/2)*Rz(-ah-Eh)*Rx(Ec)*Ry(-de-Ed)*[1;0;0];
    uvw = Ry(de)*Rz(ah)*Ry(pi/2-lati)*uvw;
    Y(2*k-1:2*k,1) = uvw(2:3,1);

    % Matrice de sensibilité
    A(2*k-1:2*k,:) = [(sl*cb - cl*ch*sb) (sh*sb) -(cl*cb+sl*ch*sb) -sb -cb 0
                    -cl*sh -ch -sl*sh 0 0 1];
endfor

A = A(:, [1 2 4 5 6]); % On supprime la colonne relative à En
X = (A'*A)\(A'*Y);
E = Y - A*X;

X *= 180/pi; % en degrés
E *= 180/pi; % en degrés
disp(["Erreurs trouvees en degres : " num2str(X(1:2,1)') " 0 " num2str(X(3:5,1)')]);
disp(["Ecart quadratique initial en degres ; " num2str(sqrt(Y'*Y)/Nval)]);
disp(["Ecart quadratique residuel en degres ; " num2str(sqrt(E'*E)/Nval)]);
endfunction

% Matrices de rotation élémentaires
function m = Rx(a)
    c = cos(a); s = sin(a);
    m = [1 0 0; 0 c -s; 0 s c];
endfunction

function m = Ry(a)
    c = cos(a); s = sin(a);
    m = [c 0 s; 0 1 0; -s 0 c];

```

```

endfunction

function m = Rz(a)
    c = cos(a); s = sin(a);
    m = [c -s 0; s c 0; 0 0 1];
endfunction

```

Ce programme de simulation d'identification des erreurs est appelé par la commande suivante : `simuestim(ecarts_dg)`, où `ecarts_dg` est le vecteur des 6 erreurs inconnues à trouver, à savoir dans l'ordre : l'erreur d'azimut, de site, de niveau, de conicité, d'angle horaire et de déclinaison, toutes exprimées en degrés. Si ces inconnues ne sont pas données, le programme prend, par défaut le vecteur suivant : [0.7, 1.1, 0, 0.9, 1.9, 1.5], avec 0 en erreur de niveau (troisième valeur).

Comme les erreurs d'azimut, de niveau et d'angle horaire sont liées, la méthode ne peut pas les identifier toutes les trois, sauf si l'une est nulle et que la méthode ignore son calcul. Ici, avec la ligne :

```
A = A(:, [1 2 4 5 6]); % On supprime la colonne relative à En
```

nous avons choisi d'ignorer le calcul de l'erreur de niveau (en sautant la colonne 3). Il suffit de modifier cette ligne pour ignorer l'erreur d'azimut ou l'erreur d'angle horaire à la place de l'erreur de niveau.

Au début du programme, après avoir généré les inconnues à trouver si elles n'ont pas été fournies, nous générons un lot de 9 lignes de visées vers les étoiles suivantes : Dubhe, Capella, Mirfak, Caph, Alderamin, Deneb, Véga, Alkaid et Arcturus.

Ensuite nous générons, dans une boucle, les erreurs de visées et les matrices de sensibilité associées à ces étoiles, et nous concaténons les erreurs et les matrices de sensibilité

A la sortie de la boucle nous supprimons de la matrice de sensibilité la colonne de l'inconnue liée que nous ignorons, puis résolvons le système par la formule des moindres carrés. Remarquez qu'au lieu d'inverser la matrice  $A^T A$ , par l'expression  $X = (A' * A) \setminus (A' * Y)$  nous indiquons à Octave de résoudre le système  $(A^T A)X = A^T Y$  ou  $A^T A$  est une matrice symétrique  $5 \times 5$  et  $A^T Y$  un vecteur  $5 \times 1$ . Dans le programme en C cette résolution était effectuée par la méthode de Cholesky sans racine carrée qui est très précise et très rapide.

Nous imprimons ensuite les inconnues  $X$  trouvées, l'erreur de visée  $Y$  (en valeur quadratique) et l'erreur résiduelle  $E = Y - AX$  (en valeur quadratique).

On trouve ensuite les 3 fonctions relatives aux matrices 3 matrices élémentaires. La matrice de micro-rotation n'est pas codée : nous effectuons le produit des 3 rotations élémentaires standards.

Voici une sortie du programme :

```

>> simuestim
Erreurs inconnues en degres : 0.7      1.1    -0  0.9    1.9    1.5
Erreurs trouvees en degres : 0.67508 1.1132 0  0.8872 1.8694 1.5166
Ecart quadratique initial en degres ; 0.013335
Ecart quadratique residuel en degres ; 0.0051011

```

La résolution est moyennement bonne : l'écart quadratique n'est réduit que de 133 à 51. Ceci s'explique par la forte non linéarité du problème due à l'importance des écarts. Si nous les divisons 10, on trouve :

```

>> simuestim([0.07, 0.11, 0, 0.09, 0.19, 0.15])
Erreurs inconnues en degres : 0.07    0.11   -0  0.09    0.19    0.15
Erreurs trouvees en degres : 0.069755 0.11014 0  0.089878 0.1897 0.15017
Ecart quadratique initial en degres ; 0.001333
Ecart quadratique residuel en degres ; 5.1544e-05

```

La résolution est excellente. L'écart quadratique est réduit de 133.3 à 5.1.

Mais ces bons résultats sont dus au fait qu'une des erreurs liées a été prise nulle et que c'est justement cette erreur que nous avons ignoré. Examinons le cas où cette erreur n'est pas nulle, mais égale à  $-0.1^\circ$  : Il vient :

```

>> simuestim([0.07, 0.11, -0.1, 0.09, 0.19, 0.15])
Erreurs inconnues en degres : 0.07    0.11   -0.1  0.09    0.19    0.15
Erreurs trouvees en degres : -0.030456 0.11009 0  0.089922 0.048358 0.15003
Ecart quadratique initial en degres ; 0.0011974
Ecart quadratique residuel en degres ; 3.6554e-05

```

On voit que les erreurs non liées sont toujours trouvées avec une grande précision (la même), par contre les trois erreurs liées sont très mal identifiées. Mais *cela est sans importance sur la correction de l'erreur de pointage*. qui est toujours aussi bonne. En effet, l'erreur quadratique initiale de 119.7 est ramenée à 3.6, et c'est là l'essentiel. On attribue aux erreurs d'azimut et d'angle horaire des valeurs inexactes, mais compte tenu de l'erreur de niveau non identifiée, leurs valeurs corrigent parfaitement l'erreur de pointage.

## Conclusion

La méthode d'identification des imperfections d'une monture équatoriale présentée dans ce document doit permettre d'améliorer significativement la précision du pointage d'un télescope vers des cibles difficiles à localiser. Sa mise en œuvre suppose la mesure des erreurs de pointages  $\Delta Y$  et  $\Delta Z$  dans le repère équatorial. Si le télescope est équipé d'une caméra avec ses axes bien orientés, avec le mouvement de déclinaison qui se fait selon  $\Delta Z$  (par exemple), ces deux erreurs pourront être mesurées directement sur la caméra en convertissant les écarts en pixels en écarts en radians. Mais l'identification suppose que les erreurs cherchées sont des constantes. Malheureusement ce n'est pas le cas, du moins pour les erreurs sur les angles horaires et déclinaison à cause du jeu et des défauts des pignonneries. Pour le jeu, on peut diviser la voûte céleste en quatre régions séparées par les méridiens Nord-Sud et Est-Ouest et identifier un lot d'erreur pour chaque région. Finalement cette méthode est assez lourde à mettre en œuvre ce qui restreint son intérêt aux montures à poste fixe.