Centre National de la Recherche Scientifique

Laboratoire de Physique des Plasmas
UMR 7648
Ecole Polytechnique, route de Saclay, 91128 PALAISEAU CEDEX

An easy method to compute plane waves polarisation from a three components waveform

by

Patrick ROBERT

CNRS/CRPE, 1980

Revised August 2008

Writing achieved May 2015

Contents

1 Method **1**
1.1 Goal . . . 1
1.2 Geometric considerations . . . 2
1.3 Problem to solve . . . 2
1.4 Analytical solution . . . 3
1.4.1 Principle . . . 3
1.4.2 Calculations . . . 4
A) Computation of X_{ia} and Y_{ia} . . . 4
B) Computation of φ . . . 5
C) Computation of major and minor axes . . . 5
D) Computation of the axes' direction : . . . 6
E) Computation of $\vec{k}$ direction . . . 6
1.4.3 Verification . . . 6
1.5 Practical application . . . 8
1.6 Summary of useful formulae . . . 8

2 Code source to compute polarisation parameters **11**
2.1 Fortran 77 code . . . 11
2.2 Fortran 90 code . . . 13
2.3 C code . . . 15
2.4 IDL code . . . 19
2.5 IDL Example of use . . . 21
2.5.1 IDL Test Procedure . . . 21
2.5.2 Result . . . 23

3 Examples of experimental results **25**
3.1 Description of plotted parameters . . . 25
3.1.1 Total power . . . 25
3.1.2 Exentricity . . . 25
3.1.3 K direction . . . 25
3.1.4 Major axis direction . . . 25
3.1.5 Remarks . . . 25
3.2 Circular polarisation with K // Bo . . . 26
3.3 Linear polarisation with K // or $\perp$ to Bo . . . 28
3.4 Linear monochromatic wave with harmonics with K $\perp$ to Bo . . . 30
3.5 Same, but unpolarised waves before event . . . 32
3.6 Linear monochromatic wave with numberous harmonics . . . 34
3.7 Perfect linear waves . . . 36
3.8 Same with time dependent frequency . . . 38
3.9 Large band circular polarisation, with k // Bo . . . 40
3.10 Other stable linear polarisation with constant frequency . . . 42
3.11 Perfect large band circular waves . . . 44

Chapter 1

Method

1.1 Goal

A search-coil magnetometer provides 3 waveforms B_x, B_y, B_z, depending on time. Main assumption in the following method is to consider the signal $B_i(t)$ as a sum of plane waves of various frequency, each wave corresponding to a single $\vec{k}$ vector, i.e. an elliptic wave (see fig 1.1).

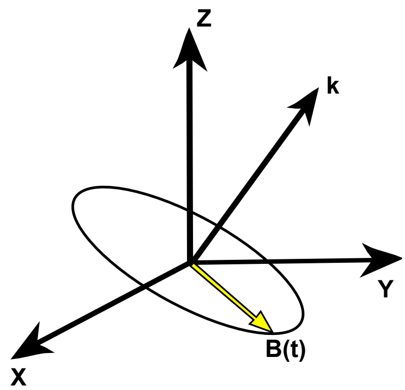

Figure 1.1: Elliptic plane wave for a single frequency, in a fixed measurement coordinate system.

The heart of the method is to do a Fourier analysis of the B_i(t) signals, assuming that each ray of the spectrum $\widetilde{B_i}(f_k)$ corresponds to a plane wave. So, the goal is to determine all the parameters of the ellipse corresponding to the ray f_k from the $B_i(t)$ signals, whatever the initial coordinate system.

Of course, for each frequency f corresponds an ellipse whose parameters are to be computed :

• $\vec{X_i}, \vec{Y_i}, \vec{Z_i}$: the 3 axes of the ellipse, (i = 1,3 i.e. 9 parameters)
• (a, b, φ) : the magnitude of the major and minor axes, and the phase of the spining $\vec{B}$ vector at the given frequency f. (see fig 1.1)
So a total of 12 parameters.

These 12 parametres can be found from the 3 input signals : $x_i(t) = \alpha_i \cos(\omega t + \Phi_i)$ $\quad i$=1,3 assuming that a Fourier analysis provides the 6 coefficients (α_i, Φ_i) for each frequency.

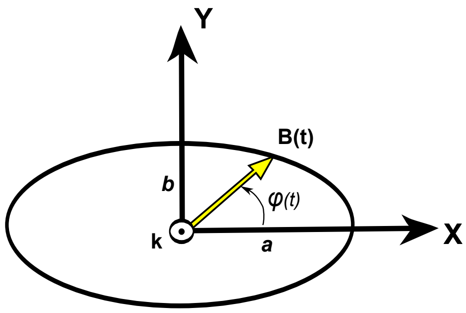

Figure 1.2: Ellipse of polarisation for a $\vec{B}$ spining vector.

1.2 Geometric considerations

We begin with the equation of the ellipse reported on its axes (fig 2) :

$$\begin{cases} X = a\cos(\omega t + \varphi) \\ Y = b\sin(\omega t + \varphi) \\ Z = 0 \end{cases} \tag{1.1}$$

φ defines the initial position of the spining vector at angular frequency ω, for $t = 0$.

In the measurement coordinat system, the three axes $(\vec{X}, \vec{Y}, \vec{Z})$ of the ellipse are defined by their components :

$$\vec{X} = \begin{pmatrix} X_x \\ X_y \\ X_z \end{pmatrix} \quad \vec{Y} = \begin{pmatrix} Y_x \\ Y_y \\ Y_z \end{pmatrix} \quad \vec{Z} = \begin{pmatrix} Z_x \\ Z_y \\ Z_z \end{pmatrix}$$

The measured components (x, y, z) at a given frequency can be deduced from (X, Y, Z) by the following coordinate system transform :

$$\begin{pmatrix} x(t) \\ y(t) \\ z(t) \end{pmatrix} = \begin{pmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{pmatrix} \begin{pmatrix} X(t) \\ Y(t) \\ Z(t) \end{pmatrix}$$

1.3 Problem to solve

X, Y, Z are sinusoidal signals defined in section 1.1. x, yz are sinusoidals signals too, which can be expressed as :

$$\alpha_i \cos(\omega t + \Phi_i) \tag{1.2}$$

So we have one matrix equation :

$$\boxed{\begin{pmatrix} \alpha_x \cos(\omega t + \Phi_x) \\ \alpha_y \cos(\omega t + \Phi_y) \\ \alpha_z \cos(\omega t + \Phi_z) \end{pmatrix} = \begin{pmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{pmatrix} \begin{pmatrix} a\cos(\omega t + \varphi) \\ b\sin(\omega t + \varphi) \\ 0 \end{pmatrix}} \tag{1.3}$$

Signals x y z are deduced from Fourier's analys which gives directly the coefficients (α_i, Φ_i) at the angular frequency $\omega = 2\pi f_k$, for the ray f_k considered.

So we have to determine the ellipse's parameters, ie (a, b, φ) and the direction of its axes, thus the vector's coordinates $\vec{X}, \vec{Y}$ and $\vec{Z}$ from equation (3), so a total of 12 unknown.

The previous matrix equation can be expressed as :

$$\boxed{\alpha_i \cos(\omega t + \Phi_i) = X_i a \cos(\omega t + \varphi) + Y_i b \sin(\omega t + \varphi)} \qquad i = x, y, z$$

Developping :

$$\alpha_i[\cos(\omega t)\cos(\Phi_i) - \sin(\omega t)\sin(\Phi_i)] = X_i a[\cos(\omega t)\cos(\varphi) - \sin(\omega t)\sin(\varphi)] + Y_i b \sin(\omega t + \varphi)$$
$$\cos(\omega t)(\alpha_i \cos(\Phi_i) - X_i a \cos(\varphi) - Y_i b \sin(\varphi)) = \sin(\omega t)(-\alpha_i \sin(\Phi_i) + X_i a \sin(\varphi) - Y_i b \cos(\varphi))$$

And we obtain the 6 equations's system (i=x, y, z) :

$$\boxed{\begin{array}{rl} & \alpha_i \cos(\Phi_i) - X_i a \cos(\varphi) - Y_i b \sin(\varphi) = 0 \\ - & \alpha_i \sin(\Phi_i) + X_i a \sin(\varphi) - Y_i b \cos(\varphi) = 0 \end{array}}$$

We can add the 6 orthonormalisation's relations of axes $\vec{X}, \vec{Y}, \vec{Z}$ (3 for orthogonal axes, and 3 for normalisation to 1) .

So we have to resolve <u>a system of 12 equations with 12 unknown</u> :

$$\alpha_i \cos(\Phi_i) - X_i a \cos(\varphi) - Y_i b \sin(\varphi) = 0 \tag{1.4}$$
$$-\alpha_i \sin(\Phi_i) + X_i a \sin(\varphi) - Y_i b \cos(\varphi) = 0 \tag{1.5}$$
$$\sum X_i Y_i = \sum Y_i Z_i = \sum Z_i X_i = 0 \tag{1.6}$$
$$\sum X_i^2 = \sum Y_i^2 = \sum Z_i^2 = 1 \tag{1.7}$$

1.4 Analytical solution

1.4.1 Principle

A) We can compute $X_i a$ and $Y_i b$, in fonction of φ, only from the equations 1.4 and 1.5.

B) If $\sum X_i Y_i = 0$ (see 1.6) we can write $\sum (X_i a)(Y_i b) = 0$, which, considered to the result of **A)**, allows the comparation of φ. Thus we know at this moment φ, $(X_i a)$ and $(Y_i b)$.

C) Axes a and b can be obtained by the expression :

$$\begin{cases} a^2 = \sum (X_i a)^2 \\ b^2 = \sum (Y_i a)^2 \end{cases}$$

D) Knowing a and b, and $X_i a$ and $Y_i b$, we can compute X_i and Y_i

E) At last Z_i is calculated by $\vec{Z} = \vec{X} \times \vec{Y}$

1.4.2 Calculations

A) Computation of X_{ia} and Y_{ia}

equation 1.4 and 1.5 are :

$$\begin{cases} \quad \alpha_i \cos(\Phi_i) - X_i a \cos(\varphi) - Y_i b \sin(\varphi) = 0 \\ - \quad \alpha_i \sin(\Phi_i) + X_i a \sin(\varphi) - Y_i b \cos(\varphi) = 0 \end{cases} \qquad \text{for } i\text{=x, y, z}$$

Multiply equation 1.4 by $\cos\varphi$ and equation 1.5 by $\sin\varphi$:

$$\begin{cases} \quad \alpha_i \cos(\Phi_i)\cos(\varphi) - X_i a \cos(\varphi)^2 - Y_i b \sin(\varphi)\cos(\varphi) = 0 \\ - \quad \alpha_i \sin(\Phi_i)\sin(\varphi) + X_i a \sin(\varphi)^2 - Y_i b \cos(\varphi)\sin(\varphi) = 0 \end{cases} \qquad \text{for } i\text{=x, y, z}$$

Adding the two equations :

$$\alpha_i(\cos(\Phi_i)\cos(\varphi) + \sin(\Phi_i)\sin(\varphi)) - X_i a = 0$$

And we obtain :

$$\boxed{X_i a = \alpha_i \cos(\varphi - \Phi_i)} \tag{1.8}$$

Multiply 1.4 by $\sin\varphi$ and 1.5 by $\cos\varphi$:

$$\begin{cases} \quad \alpha_i \cos(\Phi_i)\sin(\varphi) - X_i a \cos(\varphi)\sin(\varphi) - Y_i b \sin(\varphi)^2 = 0 \\ - \quad \alpha_i \sin(\Phi_i)\cos(\varphi) + X_i a \sin(\varphi)\cos(\varphi) - Y_i b \cos(\varphi)^2 = 0 \end{cases}$$

Adding the two equations :

$$\alpha_i[\cos(\Phi_i)\sin(\varphi) - \sin(\Phi_i)\cos(\varphi)] - Y_i b = 0$$

And we obtain :

$$\boxed{Y_i b = \alpha_i \sin(\varphi - \Phi_i)} \tag{1.9}$$

B) Computation of φ

From equation 1.6 we have : $\sum X_i Y_i = 0$

We can also write :

$\sum X_i a Y_i b = 0$

Replacing $X_i a$ and $Y_i a$ by the results given by equation 1.6 and equation 1.7, we obtain :

$\sum [\alpha_i^2 sin(\varphi - \Phi_i)\cos(\varphi - \Phi_i)] = 0$

or

$\frac{1}{2}\sum \alpha_i^2 \sin 2(\varphi - \Phi_i) = 0$

By developping we found :

$$\alpha_x^2 \sin(2\varphi - 2\Phi_x) + \alpha_y^2 \sin(2\varphi - 2\Phi_y) + \alpha_z^2 \sin(2\varphi - 2\Phi_z) = 0$$

$$\begin{aligned}
&\alpha_x^2[\sin 2\varphi \cos 2\Phi_x - \sin 2\Phi_x \cos 2\varphi] + \\
&\alpha_y^2[\sin 2\varphi \cos 2\Phi_y - \sin 2\Phi_y \cos 2\varphi] + \\
&\alpha_z^2[\sin 2\varphi \cos 2\Phi_z - \sin 2\Phi_z \cos 2\varphi] = 0
\end{aligned}$$

$$\sin 2\varphi[\alpha_x^2 \cos 2\Phi_x + \alpha_y^2 \cos 2\Phi_y + \alpha_z^2 \cos 2\Phi_z] - \cos 2\varphi[\alpha_x^2 \sin 2\Phi_x + \alpha_y^2 \sin 2\Phi_y + \alpha_z^2 \sin 2\Phi_z] = 0$$

$$\sin 2\varphi \sum \alpha_i^2 \cos 2\Phi_i = \cos 2\varphi \sum \alpha_i^2 \sin 2\Phi_i$$

Thus φ is given by :

$$\boxed{\tan 2\varphi = \frac{\sum_i \alpha_i^2 \sin 2\Phi_i}{\sum_i \alpha_i^2 \cos 2\Phi_i}} \qquad i = x, y, z \tag{1.10}$$

C) Computation of major and minor axes

Equation 1.7 give : $\sum X_i^2 = 1$ and $\sum Y_i^2 = 1$

So we can write : $\begin{cases} \sum (X_i a)^2 & = a^2 \\ \sum (Y_i b)^2 & = b^2 \end{cases}$

But from equation 1.6 and 1.7 we have : $\begin{cases} X_i a & = \alpha_i \cos(\varphi - \Phi_i) & \textit{from equation } 1.7 \\ Y_i b & = \alpha_i \sin(\varphi - \Phi_i) & \textit{from equation } 1.6 \end{cases}$

And we obtain the length of the axes :

$$\boxed{\begin{aligned} a^2 &= \sum \alpha_i^2 \cos^2(\varphi - \Phi_i) \\ b^2 &= \sum \alpha_i^2 \sin^2(\varphi - \Phi_i) \end{aligned}} \tag{1.11}$$

D) Computation of the axes' direction :

Knowing a and b, and using equation 1.8 and 1.9 we deduce directly :

$$\boxed{\begin{aligned} X_i &= \frac{1}{a}\alpha_i \cos(\varphi - \Phi_i) \\ Y_i &= \frac{1}{b}\alpha_i \sin(\varphi - \Phi_i) \end{aligned}} \tag{1.12}$$

E) Computation of $\vec{k}$ direction

We have : $\vec{k} = \vec{Z} = \vec{X} \times \vec{Y}$
So :

$$\begin{aligned} Z_x &= X_y Yz - X_z Yy \\ Z_y &= X_z Yx - X_x Yz \\ Z_z &= X_y Yz - X_z Yy \end{aligned}$$

It's possible to developp :

$$X_i Y_j = \frac{1}{ab}\alpha_i \alpha j \cos(\varphi - \Phi_i)\sin(\varphi - \Phi_j)$$
$$X_i Y_j - X_j Y_i = \frac{1}{ab}(\alpha_i \alpha_j \cos(\varphi - \Phi_i)\sin(\varphi - \Phi_j) - \alpha_j \alpha_i \cos(\varphi - \Phi_j)\sin(\varphi - \Phi_i))$$
$$X_i Y_j - X_j Y_i = \frac{\alpha_i \alpha_j}{ab}\sin(\Phi_i - \Phi_j)$$

Thus $\vec{k}$ is given by :

$$\boxed{Z_k = \frac{\alpha_i \alpha_j}{ab}\sin(\Phi_i - \Phi_j)} \qquad i, j, k = x, y, z \tag{1.13}$$

1.4.3 Verification

If $\alpha_x = a\alpha_y = b\alpha_z = 0$

$$\begin{cases} \Phi_x = \varphi \\ \Phi_y = \varphi - \frac{\pi}{2} \end{cases}$$

We must find :

$$\begin{cases} X_x = 1 \\ X_y = 0 \\ X_z = 0 \end{cases}$$

$$\begin{cases} Y_x = 0 \\ Y_y = 1 \\ Y_z = 0 \end{cases}$$

$$\begin{cases} Z_x = 0 \\ Z_y = 0 \\ Z_z = 1 \end{cases}$$

Verification of φ :

$$\tan(2\varphi) = \frac{\sum(\alpha_i)^2 \sin(2\Phi_i)}{\sum(\alpha_i)^2 \cos(2\Phi_i)} = \frac{a^2\sin(2\varphi) + b^2\sin(2\varphi - \pi)}{a^2\cos(2\varphi) + b^2\cos(2\varphi - \pi)}$$

$$\tan(2\varphi) = \frac{a^2\sin(2\varphi) - b^2\sin(2\varphi)}{a^2\sin(2\varphi) - b^2\sin(2\varphi)} = \frac{(a^2 - b^2)\sin(2\varphi)}{(a^2 - b^2)\cos(2\varphi)} = \tan(2\varphi)$$

Verification of a^2 and b^2 :

$$\begin{cases} a^2 = \sum \alpha_i^2 \cos^2(\varphi - \Phi_i) = a^2 + b^2\cos^2(\varphi - (\varphi + \frac{\pi}{2})) = a^2 \\ b^2 = \sum \alpha_i^2 \sin^2(\varphi - \Phi_i) = b^2\sin^2(\varphi - (\varphi + \frac{\pi}{2})) = b^2 \end{cases}$$

Verification of X_i, Y_i, Z_i :

$X_i = \frac{1}{a}\alpha_i \cos(\varphi - \Phi_i)$

$$\begin{cases} X_x = \dfrac{1}{a}a = 1 \\ X_y = \dfrac{1}{a}b\cos(\dfrac{-\pi}{2}) = 0 \\ X_z = 0 \end{cases}$$

$$Y_i = \frac{1}{a}\alpha_i \sin(\varphi - \Phi_i)$$

$$\begin{cases} Y_x = \dfrac{1}{a}a\sin(0) = 0 \\ Y_y = \dfrac{1}{b}b\sin(\dfrac{\pi}{2}) = 1 \\ Y_z = 0 \end{cases}$$

$$Z_k = \frac{\alpha_i\alpha_j}{ab}\sin(\Phi_i - \Phi_j)$$

$$\begin{cases} Z_x = \dfrac{\alpha_y\alpha_z}{ab}\sin(\Phi_y - \Phi_z) = 0 \\ Z_y = \dfrac{\alpha_z\alpha_x}{ab}\sin(\Phi_z - \Phi_x) = 0 \\ Z_z = \dfrac{\alpha_x\alpha_y}{ab}\sin(\Phi_x - \Phi_y) = 1 \end{cases}$$

1.5 Practical application

We have a fixed coordinate's system x y z in which we record signals x(t), y(t), z(t).
By complex FFT we obtain coefficients $\mathbf{C}^{\mathbf{x}}_{\eta}, \mathbf{C}^{\mathbf{y}}_{\eta}, \mathbf{C}^{\mathbf{z}}_{\eta}$

$$\begin{cases} \mathbf{C}^{\mathbf{x}}_{\eta} = \rho_{\mathbf{x}} \mathbf{e}^{\mathbf{i}\theta_{\mathbf{x}}} \\ \mathbf{C}^{\mathbf{y}}_{\eta} = \rho_{\mathbf{y}} \mathbf{e}^{\mathbf{i}\theta_{\mathbf{y}}} \\ \mathbf{C}^{\mathbf{z}}_{\eta} = \rho_{\mathbf{z}} \mathbf{e}^{\mathbf{i}\theta_{\mathbf{z}}} \end{cases}$$

With $(\mathbf{C}^{\mathbf{x}}_{-\eta}) = (\mathbf{C}^{\mathbf{x}}_{\eta})^*$ because x(t) is a real signal. Idem for y and z.

Signals x(t), y(t), z(t) are then decomposed into a sum of monochromatic's signals :

$$\begin{cases} x_{\eta\omega_0}(t) = 2\rho_x \cos(\eta\omega_0 t + \Phi_x) \Longrightarrow \begin{cases} \alpha_x = 2\rho_x \\ \Phi_x = \Theta_x \end{cases} \\ y_{\eta\omega_0}(t) = 2\rho_y \cos(\eta\omega_0 t + \Phi_y) \Longrightarrow \begin{cases} \alpha_y = 2\rho_y \\ \Phi_y = \Theta_y \end{cases} \\ z_{\eta\omega_0}(t) = 2\rho_y \cos(\eta\omega_0 t + \Phi_z) \Longrightarrow \begin{cases} \alpha_z = 2\rho_z \\ \Phi_z = \Theta_z \end{cases} \end{cases}$$

For each frequency $f = \dfrac{\omega}{2\pi} = \dfrac{\eta\omega_0}{2\pi}$ we determine completely an ellipse's parameters, with frequency $\frac{\omega}{2\pi}$, and a projection three axes x, y, z would give the three monochromatic's signals $x_\omega(t), y_\omega(t), z_\omega(t)$ exactly similar to those recorded.
(Nevertheless we have to prove this decomposition is single).

<u>So ellipse parameters are</u> :

1) (a, b, φ) = major axe, minor axe, ellipse's phase centered in the system XY

2) $(\vec{X}, \vec{Y}, \vec{Z})$ = coordinates of the main axis of the ellipse in the (x, y, z) system : $\vec{X}$=major axe, $\vec{Y}$=minor axe, $\vec{Z}$=normal to the wave plane.

1.6 Summary of useful formulae

We begin with a mesure's system xyz in which by a Fourier's analysis, we know the 3 components of monochromatic signal.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \alpha_x \cos(\omega t + \Phi_x) \\ \alpha_y \cos(\omega t + \Phi_y) \\ \alpha_z \cos(\omega t + \Phi_z) \end{pmatrix}$$

So we know α_i and Φ_i

This 3 components correspond to the projection on xyz of an ellipse, which reported to its main axes XYZ written :

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} a\cos(\omega t + \varphi) \\ b\sin(\omega t + \varphi) \\ 0 \end{pmatrix}$$

The previous computations allows us to determine :

1) The absolute phase φ :

$$\boxed{\tan(2\varphi) = \frac{\sum_i \alpha_i^2 \sin(2\Phi_i)}{\sum_i \alpha_i^2 \cos(2\Phi_i)}}$$

2) Major and minor axes' length :

$$\boxed{a^2 = \sum_i \alpha_i^2 \cos(\varphi - \Phi_i) \qquad b^2 = \sum_i \alpha_i^2 \sin(\varphi - \Phi_i)}$$

3) Ellipse's axes directions :

$$\boxed{X_i = \frac{1}{a}\alpha_i \cos(\varphi - \Phi_i) \quad Y_i = \frac{1}{b}\alpha_i \sin(\varphi - \Phi_i) \quad Z_i = \frac{1}{ab}\alpha_j \alpha_k \sin(\Phi_j - \Phi_k)}$$

The implementation of these formulas is made in the next chapter 2, in the form of codes written in FORTRAN 77, FORTRAN 90, C and IDL. Chapter 3 show various examples of wave polarisation observed by the CLUSTER/STAFF-SC experiment which valid the method.

Chapter 2

Code source to compute polarisation parameters

2.1 Fortran 77 code

The Fortran code was initially written in F77, as given below. Conversion in F90 has been done on next section, with a minimum of modifications, only to be compiled as a polarlib.f90 code.

```
CXXXXXXXXOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOXX

C BEGIN polarlib.f

C   ----------------------------------------------------------------
C   |                                                              |
C   |                      polarlib                                |
C   |                                                              |
C   | Bibliotheque de calcul de la polarisation d'une onde plane.  |
C   | Extrait des bibliotheques developpees pour le satellite GEOS |
C   | au CRPE en 1977-1980. Conversion succinte des .f en .f90     |
C   |                                                              |
C   |           P. Robert, CNRS/CETP, Novembre 2001                |
C   |                                                              |
C   ----------------------------------------------------------------

CXXXXXXXXOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOXX

      subroutine cpolar(sx,sy,sz,ptot,a,b,rkx,rky,rkz,ax,ay,az,phase)

C     ---------------------------------------------------------------+--
C     calcul de l'ellipse de polarisation d'une onde plane, a partir
C     de ses coefficients de Fourrier sur 3 axes orthogonaux.
C     input : complex sx,sy,sz
C     output: a,b        = axes de l'ellipse
C             rkx,rky,rkz= normale au plan de l'ellipse
C             ax ,ay ,az = direction du grand axe
C             phase      = phase absolue du champ le long de l'ellipse

C     P. Robert, CETP, Novembre 2001
C     ---------------------------------------------------------------+--

      complex sx,sy,sz

      real aii(3),fii(3),GXI(3),GZI(3)


C *** calcul des amplitudes et phase sur chaque composante

      call CALAIFI(sx,sy,sz,aii,fii)

C *** puissance totale

      ptot= 0.5*(aii(1)**2 +aii(2)**2 + aii(3)**2)

C *** calcul des parametres de l'ellipse
```

```
      call ELLIPSE(aii,fii,phase,a,b,EXC,GXI,GZI)

      rkx=GZI(1)
      rky=GZI(2)
      rkz=GZI(3)

      ax=GXI(1)
      ay=GXI(2)
      az=GXI(3)

      return
      end

CXXXXXXXXOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOXX

      SUBROUTINE CALAIFI(SX,SY,SZ,AII,FII)

      DIMENSION AII(3),FII(3)
      COMPLEX SX,SY,SZ

C     *****************************************************************
C     AUTEUR   : P. ROBERT CRPE-CNET, 1980  (revu Novembre 2001)
C     CATEGORIE: DEPOUILLEMENT SPECIFIQUE UBF/GEOS
C     OBJET    : CALCUL DES COEFFICIENTS AI ET FII DU PROGRAMME POLAR
C     *****************************************************************


      AII(1)= 2.*CABS(SX)
      AII(2)= 2.*CABS(SY)
      AII(3)= 2.*CABS(SZ)

      RR= REAL(SX)
      RI=AIMAG(SX)
      FII(1)=0.
      IF(ABS(RR).GT.1.E-30.OR.ABS(RI).GT.1.E-30) FII(1)=ATAN2(RI,RR)

      RR= REAL(SY)
      RI=AIMAG(SY)
      FII(2)=0.
      IF(ABS(RR).GT.1.E-30.OR.ABS(RI).GT.1.E-30) FII(2)=ATAN2(RI,RR)

      RR= REAL(SZ)
      RI=AIMAG(SZ)
      FII(3)=0.
      IF(ABS(RR).GT.1.E-30.OR.ABS(RI).GT.1.E-30) FII(3)=ATAN2(RI,RR)

      RETURN
      END

CXXXXXXXXOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOXX

      SUBROUTINE ELLIPSE(AII,FII,PHA,AA,BB,EXC,GXI,GZI)

      DIMENSION AII(3),FII(3),GXI(3),GZI(3)
      double precision SN, CN, dblePHA, dbleAA, dbleBB

C     *****************************************************************
C     AUTEUR   : P. ROBERT CRPE-CNET, 1980
C     CATEGORIE: TRAITEMENT DU SIGNAL
C     OBJET    : CALCUL D UNE ELLIPSE A PARTIR DE SES 3 PROJECTIONS
C     *****************************************************************

C      CALCUL DE L"ELLIPSE DE POLARISATION A PARTIR DE SES 3 PROJECTIONS
C      DONNEES PAR LE MODULE AII ET LA PHASE FII DE CHAQUE COMPOSANTE


C *   CALCUL DE LA PHASE ABSOLUE

      SN=0.d0
      CN=0.d0
      dblePHA=0.d0

      DO 76 J=1,3
         SN=SN+(dble(AII(J))**2)*SIN(2.d0*dble(FII(J)))
         CN=CN+(dble(AII(J))**2)*COS(2.d0*dble(FII(J)))
   76 CONTINUE
      IF(DABS(SN).GT.1.D-30.AND.DABS(CN).GT.1.D-30)
     &     dblePHA=DATAN2(SN,CN)/2.D0
```

```
C *   CALCUL DU MODULE DU GRAND AXE, DU PETIT AXE ET EXENTRICITE

      dbleAA=1.D-30
      dbleBB=1.D-30

      DO 78 J=1,3
         dbleAA=dbleAA+(dble(AII(J))*DCOS(dblePHA-dble(FII(J))))**2
         dbleBB=dbleBB+(dble(AII(J))*DSIN(dblePHA-dble(FII(J))))**2
   78 CONTINUE

      dbleAA=DSQRT(dbleAA)
      dbleBB=DSQRT(dbleBB)
      EXC=sngl(dbleBB/dbleAA)

C *   CALCUL DE LA DIRECTION DES AXES DE L"ELLIPSE

      DO 82 J=1,3
         GXI(J)=sngl(dble(AII(J))*DCOS(dblePHA-dble(FII(J)))/dbleAA)
   82 CONTINUE

      GZI(1)=sngl(dble(AII(2))*dble(AII(3))*
     &       DSIN(dble(FII(2))-dble(FII(3)))/(dbleAA*dbleBB))
      GZI(2)=sngl(dble(AII(3))*dble(AII(1))*
     &       DSIN(dble(FII(3))-dble(FII(1)))/(dbleAA*dbleBB))
      GZI(3)=sngl(dble(AII(1))*dble(AII(2))*
     &       DSIN(dble(FII(1))-dble(FII(2)))/(dbleAA*dbleBB))

C *   CALCUL DU GRAND ET PETIT AXE, ET DE LA PHASE ABSOLUE

      PHA=sngl(dblePHA)
      AA=sngl(dbleAA)
      BB=sngl(dbleBB)

      RETURN
      END

CXXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX

C END polarlib.f
```

2.2 Fortran 90 code

```
!XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX

! BEGIN polarlib.f

!   ----------------------------------------------------------------
!   |                                                              |
!   |                      polarlib                                |
!   |                                                              |
!   | Bibliotheque de calcul de la polarisation d'une onde plane.  |
!   | Extrait des bibliotheques developpees pour le satellite GEOS |
!   | au CRPE en 1977-1980. Conversion succinte des .f en .f90     |
!   |                                                              |
!   |           P. Robert, CNRS/CETP, Novembre 2001                |
!   |                                                              |
!   ----------------------------------------------------------------
!
!XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX

      subroutine cpolar(sx,sy,sz,ptot,a,b,rkx,rky,rkz,ax,ay,az,phase)

!     -----------------------------------------------------------------+--
!     calcul de l'ellipse de polarisation d'une onde plane, a partir
!     de ses coefficients de Fourrier sur 3 axes orthogonaux.
!     input : complex sx,sy,sz
!     output: a,b        = axes de l'ellipse
!             rkx,rky,rkz= normale au plan de l'ellipse
!             ax ,ay ,az = direction du grand axe
!             phase      = phase absolue du champ le long de l'ellipse
!
!     P. Robert, CETP, Novembre 2001
!     -----------------------------------------------------------------+--
```

```
      complex sx,sy,sz

      real aii(3),fii(3),GXI(3),GZI(3)


! *** calcul des amplitudes et phase sur chaque composante

      call CALAIFI(sx,sy,sz,aii,fii)

! *** puissance totale

      ptot= 0.5*(aii(1)**2 +aii(2)**2 + aii(3)**2)

! *** calcul des parametres de l'ellipse

      call ELLIPSE(aii,fii,phase,a,b,EXC,GXI,GZI)

      rkx=GZI(1)
      rky=GZI(2)
      rkz=GZI(3)

      ax=GXI(1)
      ay=GXI(2)
      az=GXI(3)

      return
      end

!XXXXXXXXOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOXX

      SUBROUTINE CALAIFI(SX,SY,SZ,AII,FII)

      DIMENSION AII(3),FII(3)
      COMPLEX SX,SY,SZ

!     *****************************************************************
!     AUTEUR   : P. ROBERT CRPE-CNET, 1980  (revu Novembre 2001)
!     CATEGORIE: DEPOUILLEMENT SPECIFIQUE UBF/GEOS
!     OBJET    : CALCUL DES COEFFICIENTS AI ET FII DU PROGRAMME POLAR
!     *****************************************************************


      AII(1)= 2.*CABS(SX)
      AII(2)= 2.*CABS(SY)
      AII(3)= 2.*CABS(SZ)

      RR= REAL(SX)
      RI=AIMAG(SX)
      FII(1)=0.
      IF(ABS(RR).GT.1.E-30.OR.ABS(RI).GT.1.E-30) FII(1)=ATAN2(RI,RR)

      RR= REAL(SY)
      RI=AIMAG(SY)
      FII(2)=0.
      IF(ABS(RR).GT.1.E-30.OR.ABS(RI).GT.1.E-30) FII(2)=ATAN2(RI,RR)

      RR= REAL(SZ)
      RI=AIMAG(SZ)
      FII(3)=0.
      IF(ABS(RR).GT.1.E-30.OR.ABS(RI).GT.1.E-30) FII(3)=ATAN2(RI,RR)

      RETURN
      END

!XXXXXXXXOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOXX

      SUBROUTINE ELLIPSE(AII,FII,PHA,AA,BB,EXC,GXI,GZI)

      DIMENSION AII(3),FII(3),GXI(3),GZI(3)
      double precision SN, CN, dblePHA, dbleAA, dbleBB

!     *****************************************************************
!     AUTEUR   : P. ROBERT CRPE-CNET, 1980
!     CATEGORIE: TRAITEMENT DU SIGNAL
!     OBJET    : CALCUL D UNE ELLIPSE A PARTIR DE SES 3 PROJECTIONS
!     *****************************************************************

!      CALCUL DE L"ELLIPSE DE POLARISATION A PARTIR DE SES 3 PROJECTIONS
```

```
!      DONNEES PAR LE MODULE AII ET LA PHASE FII DE CHAQUE COMPOSANTE


! *    CALCUL DE LA PHASE ABSOLUE

       SN=0.d0
       CN=0.d0
       dblePHA=0.d0

       DO 76 J=1,3
          SN=SN+(dble(AII(J))**2)*SIN(2.d0*dble(FII(J)))
          CN=CN+(dble(AII(J))**2)*COS(2.d0*dble(FII(J)))
    76 CONTINUE
       IF(DABS(SN).GT.1.D-30.AND.DABS(CN).GT.1.D-30) &
            dblePHA=DATAN2(SN,CN)/2.D0

! *    CALCUL DU MODULE DU GRAND AXE, DU PETIT AXE ET EXENTRICITE

       dbleAA=1.D-30
       dbleBB=1.D-30

       DO 78 J=1,3
          dbleAA=dbleAA+(dble(AII(J))*DCOS(dblePHA-dble(FII(J))))**2
          dbleBB=dbleBB+(dble(AII(J))*DSIN(dblePHA-dble(FII(J))))**2
    78 CONTINUE

       dbleAA=DSQRT(dbleAA)
       dbleBB=DSQRT(dbleBB)
       EXC=sngl(dbleBB/dbleAA)

! *    CALCUL DE LA DIRECTION DES AXES DE L"ELLIPSE

       DO 82 J=1,3
          GXI(J)=sngl(dble(AII(J))*DCOS(dblePHA-dble(FII(J)))/dbleAA)
    82 CONTINUE
!
       GZI(1)=sngl(dble(AII(2))*dble(AII(3))* &
              DSIN(dble(FII(2))-dble(FII(3)))/(dbleAA*dbleBB))
       GZI(2)=sngl(dble(AII(3))*dble(AII(1))* &
              DSIN(dble(FII(3))-dble(FII(1)))/(dbleAA*dbleBB))
       GZI(3)=sngl(dble(AII(1))*dble(AII(2))* &
              DSIN(dble(FII(1))-dble(FII(2)))/(dbleAA*dbleBB))

! *    CALCUL DU GRAND ET PETIT AXE, ET DE LA PHASE ABSOLUE

       PHA=sngl(dblePHA)
       AA=sngl(dbleAA)
       BB=sngl(dbleBB)

       RETURN
       END

!XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX

! END polarlib.f90
```

2.3 C code

The following C code has been obtained from the f77 code and the f2C tool. Then, aesthetics of the C code has been improved, and code itself checked by using C compiler on Ubuntu Linux machine.

```
/* polarlib.f -- translated by f2c (version 20100827).
   You must link the resulting object file with libf2c:
        on Microsoft Windows system, link with libf2c.lib;
        on Linux or Unix systems, link with .../path/to/libf2c.a -lm
        or, if you install libf2c.a in a standard place, with -lf2c -lm
        -- in that order, at the end of the command line, as in
                cc *.o -lf2c -lm
```

```
        Source for libf2c is in /netlib/f2c/libf2c.zip, e.g.,

                http://www.netlib.org/f2c/libf2c.zip
*/

#include "f2c.h"

/* XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX*/
/*                                                                           */
/* BEGIN polarlib.c                                                         */
/*   ---------------------------------------------------------------        */
/*   |                                                             |        */
/*   |                         polarlib                            |        */
/*   |                                                             |        */
/*   | Bibliotheque de calcul de la polarisation d'une onde plane.  |        */
/*   | Extrait des bibliotheques developpees pour le satellite GEOS |        */
/*   | au CRPE en 1977-1980. Conversion des .f en .c Avril 2015     |        */
/*   |                                                             |        */
/*   |           P. Robert, CNRS/CETP, Novembre 2001               |        */
/*   |                                                             |        */
/*   ---------------------------------------------------------------        */
/*                                                                           */
/* XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX*/

/* Subroutine */ int cpolar_(complex *sx, complex *sy, complex *sz, real *
        ptot, real *a, real *b, real *rkx, real *rky, real *rkz, real *ax,
        real *ay, real *az, real *phase)
{
    /* System generated locals */
    real r__1, r__2, r__3;

    /* Local variables */
    static real aii[3], fii[3], exc, gxi[3], gzi[3];
    extern /* Subroutine */ int calaifi_(complex *, complex *, complex *,
            real *, real *), ellipse_(real *, real *, real *, real *, real *,
            real *, real *, real *);

/*     ----------------------------------------------------------------+--*/
/*     calcul de l'ellipse de polarisation d'une onde plane, a partir    */
/*     de ses coefficients de Fourrier sur 3 axes orthogonaux.           */
/*     input : complex sx,sy,sz                                          */
/*     output: a,b        = axes de l'ellipse                            */
/*             rkx,rky,rkz= normale au plan de l'ellipse                 */
/*             ax ,ay ,az = direction du grand axe                       */
/*             phase      = phase absolue du champ le long de l'ellipse  */
/*                                                                        */
/*     P. Robert, CETP, Novembre 2001                                    */
/*     ----------------------------------------------------------------+--*/

/* *** calcul des amplitudes et phase sur chaque composante */

    calaifi_(sx, sy, sz, aii, fii);

/* *** puissance totale */

    r__1 = aii[0];
    r__2 = aii[1];
    r__3 = aii[2];

    *ptot = (r__1 * r__1 + r__2 * r__2 + r__3 * r__3) * .5f;

/* *** calcul des parametres de l'ellipse */

    ellipse_(aii, fii, phase, a, b, &exc, gxi, gzi);

    *rkx = gzi[0];
    *rky = gzi[1];
    *rkz = gzi[2];

    *ax = gxi[0];
    *ay = gxi[1];
    *az = gxi[2];

    return 0;

} /* cpolar_ */

/* XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX*/
```

```
/* Subroutine */ int calaifi_(complex *sx, complex *sy, complex *sz, real *
        aii, real *fii)
{
    /* Builtin functions */
    double c_abs(complex *), r_imag(complex *), atan2(doublereal, doublereal);

    /* Local variables */
    static real ri, rr;

/*     ******************************************************************/
/*     AUTEUR   : P. ROBERT CRPE-CNET, 1980  (revu Novembre 2001)       */
/*     CATEGORIE: DEPOUILLEMENT SPECIFIQUE UBF/GEOS                     */
/*     OBJET    : CALCUL DES COEFFICIENTS AI ET FII DU PROGRAMME POLAR   */
/*     ******************************************************************/

    /* Parameter adjustments */
    --fii;
    --aii;

    /* Function Body */

    aii[1] = c_abs(sx) * 2.f;
    aii[2] = c_abs(sy) * 2.f;
    aii[3] = c_abs(sz) * 2.f;

    rr = sx->r;
    ri = r_imag(sx);
    fii[1] = 0.f;
    if (dabs(rr) > 1e-30f || dabs(ri) > 1e-30f) {
        fii[1] = atan2(ri, rr);
    }

    rr = sy->r;
    ri = r_imag(sy);
    fii[2] = 0.f;
    if (dabs(rr) > 1e-30f || dabs(ri) > 1e-30f) {
        fii[2] = atan2(ri, rr);
    }

    rr = sz->r;
    ri = r_imag(sz);
    fii[3] = 0.f;
    if (dabs(rr) > 1e-30f || dabs(ri) > 1e-30f) {
        fii[3] = atan2(ri, rr);
    }

    return 0;

} /* calaifi_ */

/* XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX*/


/* Subroutine */ int ellipse_(real *aii, real *fii, real *pha, real *aa, real
        *bb, real *exc, real *gxi, real *gzi)
{
    /* System generated locals */
    doublereal d__1;

    /* Builtin functions */
    double sin(doublereal), cos(doublereal), atan2(doublereal, doublereal),
            sqrt(doublereal);

    /* Local variables */
    static integer j;
    static doublereal cn, sn, dbleaa, dblebb, dblepha;

/*     ******************************************************************/
/*     AUTEUR   : P. ROBERT CRPE-CNET, 1980                             */
/*     CATEGORIE: TRAITEMENT DU SIGNAL                                  */
/*     OBJET    : CALCUL D UNE ELLIPSE A PARTIR DE SES 3 PROJECTIONS     */
/*     ******************************************************************/

/*     CALCUL DE L"ELLIPSE DE POLARISATION A PARTIR DE SES 3 PROJECTIONS */
/*     DONNEES PAR LE MODULE AII ET LA PHASE FII DE CHAQUE COMPOSANTE    */

/*  CALCUL DE LA PHASE ABSOLUE */
```

```
    /* Parameter adjustments */
    --gzi;
    --gxi;
    --fii;
    --aii;

    /* Function Body */
    sn = 0.;
    cn = 0.;
    dblepha = 0.;

    for (j = 1; j <= 3; ++j) {
        d__1 = (doublereal) aii[j];
        sn += d__1 * d__1 * sin((doublereal) fii[j] * 2.);
        d__1 = (doublereal) aii[j];
        cn += d__1 * d__1 * cos((doublereal) fii[j] * 2.);
    }

    if (abs(sn) > 1e-30 && abs(cn) > 1e-30) {
        dblepha = atan2(sn, cn) / 2.;
    }

/*  CALCUL DU MODULE DU GRAND AXE, DU PETIT AXE ET EXENTRICITE */

    dbleaa = 1e-30;
    dblebb = 1e-30;

    for (j = 1; j <= 3; ++j) {
        d__1 = (doublereal) aii[j] * cos(dblepha - (doublereal) fii[j]);
        dbleaa += d__1 * d__1;
        d__1 = (doublereal) aii[j] * sin(dblepha - (doublereal) fii[j]);
        dblebb += d__1 * d__1;
    }

    dbleaa = sqrt(dbleaa);
    dblebb = sqrt(dblebb);
    *exc = (real) (dblebb / dbleaa);

/*  CALCUL DE LA DIRECTION DES AXES DE L"ELLIPSE */

    for (j = 1; j <= 3; ++j) {
        gxi[j] = (real) ((doublereal) aii[j] * cos(dblepha - (doublereal) fii[
                j]) / dbleaa);
    }

    gzi[1] = (real) ((doublereal) aii[2] * (doublereal) aii[3] * sin((
            doublereal) fii[2] - (doublereal) fii[3]) / (dbleaa * dblebb));
    gzi[2] = (real) ((doublereal) aii[3] * (doublereal) aii[1] * sin((
            doublereal) fii[3] - (doublereal) fii[1]) / (dbleaa * dblebb));
    gzi[3] = (real) ((doublereal) aii[1] * (doublereal) aii[2] * sin((
            doublereal) fii[1] - (doublereal) fii[2]) / (dbleaa * dblebb));

/*  CALCUL DU GRAND ET PETIT AXE, ET DE LA PHASE ABSOLUE */

    *pha = (real) dblepha;
    *aa = (real) dbleaa;
    *bb = (real) dblebb;

    return 0;

} /* ellipse_ */

/* XXXXXXXXOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOXX*/

/* END polarlib.c                                                          */
```

2.4 IDL code

IDL code has been deduced from F77 version in May 2015.

```
;XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX

; BEGIN polarlib.pro

;   ----------------------------------------------------------------
;   |                                                              |
;   |                       polarlib                               |
;   |                                                              |
;   | Bibliotheque de calcul de la polarisation d'une onde plane.  |
;   | Extrait des bibliotheques developpees pour le satellite GEOS |
;   | au CRPE en 1977-1980. Conversion succinte des .f en .f90     |
;   |                                                              |
;   |           P. Robert, CNRS/CETP, Novembre 2001                |
;   |                                                              |
;   ----------------------------------------------------------------

;XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX

      PRO CALAIFI, SX,SY,SZ,AII,FII

;     DIMENSION AII(3),FII(3) defined in calling procedure
;     COMPLEX SX,SY,SZ        defined in calling procedure

;     ******************************************************************
;     AUTEUR   : P. ROBERT CRPE-CNET, 1980  (revu Novembre 2001)
;     CATEGORIE: DEPOUILLEMENT SPECIFIQUE UBF/GEOS
;     OBJET    : CALCUL DES COEFFICIENTS AI ET FII DU PROGRAMME POLAR
;     ******************************************************************


      AII(0)= 2.*ABS(SX)
      AII(1)= 2.*ABS(SY)
      AII(2)= 2.*ABS(SZ)

      RR=REAL_PART(SX)
      RI=IMAGINARY(SX)
      FII(0)=0.
      IF(ABS(RR) GT 1.E-30 OR ABS(RI) GT 1.E-30) THEN FII(0)=ATAN(RI,RR)

      RR=REAL_PART(SY)
      RI=IMAGINARY(SY)
      FII(1)=0.
      IF(ABS(RR) GT 1.E-30 OR ABS(RI) GT 1.E-30) THEN FII(1)=ATAN(RI,RR)

      RR=REAL_PART(SZ)
      RI=IMAGINARY(SZ)
      FII(2)=0.
      IF(ABS(RR) GT 1.E-30 OR ABS(RI) GT 1.E-30) THEN FII(2)=ATAN(RI,RR)

      END

;XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX

      PRO ELLIPSE, AII,FII,PHA,AA,BB,EXC,GXI,GZI

;     DIMENSION AII(3),FII(3),GXI(3),GZI(3) defined in calling procedure

;     ******************************************************************
;     AUTEUR   : P. ROBERT CRPE-CNET, 1980
;     CATEGORIE: TRAITEMENT DU SIGNAL
;     OBJET    : CALCUL D UNE ELLIPSE A PARTIR DE SES 3 PROJECTIONS
;     ******************************************************************

;      CALCUL DE L"ELLIPSE DE POLARISATION A PARTIR DE SES 3 PROJECTIONS
;      DONNEES PAR LE MODULE AII ET LA PHASE FII DE CHAQUE COMPOSANTE


; *   CALCUL DE LA PHASE ABSOLUE

      SN=0.d0
      CN=0.d0
      dblePHA=0.d0
```

```
      FOR J=0,2 DO BEGIN
         SN=SN+(DOUBLE(AII(J))^2)*SIN(2.d0*DOUBLE(FII(J)))
         CN=CN+(DOUBLE(AII(J))^2)*COS(2.d0*DOUBLE(FII(J)))
      ENDFOR

      IF(ABS(SN) GT 1.D-30 AND ABS(CN) GT 1.D-30) THEN $
           dblePHA=ATAN(SN,CN)/2.D0

; *   CALCUL DU MODULE DU GRAND AXE, DU PETIT AXE ET EXENTRICITE

      dbleAA=1.D-30
      dbleBB=1.D-30

      FOR J=0,2 DO BEGIN
         dbleAA=dbleAA+(DOUBLE(AII(J))*COS(dblePHA-DOUBLE(FII(J))))^2
         dbleBB=dbleBB+(DOUBLE(AII(J))*SIN(dblePHA-DOUBLE(FII(J))))^2
      ENDFOR

      dbleAA=SQRT(dbleAA)
      dbleBB=SQRT(dbleBB)
      EXC=FLOAT(dbleBB/dbleAA)

; *   CALCUL DE LA DIRECTION DES AXES DE L"ELLIPSE

      FOR J=0,2 DO GXI(J)=FLOAT(DOUBLE(AII(J))*COS(dblePHA-DOUBLE(FII(J)))/dbleAA)

      GZI(0)=FLOAT(DOUBLE(AII(1))*DOUBLE(AII(2))*  $
               SIN(DOUBLE(FII(1))-DOUBLE(FII(2)))/(dbleAA*dbleBB))
      GZI(1)=FLOAT(DOUBLE(AII(2))*DOUBLE(AII(0))*  $
               SIN(DOUBLE(FII(2))-DOUBLE(FII(0)))/(dbleAA*dbleBB))
      GZI(2)=FLOAT(DOUBLE(AII(0))*DOUBLE(AII(1))*  $
               SIN(DOUBLE(FII(0))-DOUBLE(FII(1)))/(dbleAA*dbleBB))

; *   CALCUL DU GRAND ET PETIT AXE, ET DE LA PHASE ABSOLUE

      PHA=FLOAT(dblePHA)
      AA =FLOAT(dbleAA)
      BB =FLOAT(dbleBB)

      END

;XXXXXXXX0XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX0XX

      PRO cpolar, sx,sy,sz,ptot,a,b,rkx,rky,rkz,ax,ay,az,phase

;     ---------------------------------------------------------------+--
;     calcul de l'ellipse de polarisation d'une onde plane, a partir
;     de ses coefficients de Fourrier sur 3 axes orthogonaux.
;     input : complex sx,sy,sz
;     output: a,b        = axes de l'ellipse
;             rkx,rky,rkz= normale au plan de l'ellipse
;             ax ,ay ,az = direction du grand axe
;             phase      = phase absolue du champ le long de l'ellipse

;     P. Robert, CETP, Novembre 2001
;     ---------------------------------------------------------------+--

;     complex sx,sy,sz defined in calling procedure, containing Fourier coeff

      aii=FLTARR(3)
      fii=FLTARR(3)

      GXI=FLTARR(3)
      GZI=FLTARR(3)

; *** calcul des amplitudes et phase sur chaque composante

      CALAIFI, sx,sy,sz,aii,fii

; *** puissance totale

      ptot= 0.5*(aii(0)^2 +aii(1)^2 + aii(2)^2)

; *** calcul des parametres de l'ellipse

      ELLIPSE, aii,fii,phase,a,b,EXC,GXI,GZI

      rkx=GZI(0)
      rky=GZI(1)
      rkz=GZI(2)
```

```

       ax=GXI(0)
       ay=GXI(1)
       az=GXI(2)

       end

;XXXXXXXXOXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXOXX

; END polarlib.pro

```

2.5 IDL Example of use

2.5.1 IDL Test Procedure

In section 1.6 we saw that, after a Fourier transform of the 3 waveforms, we know the amplitude α_i and phase Φ_i of the 3 components of monochromatic signal, for each frequency f, by :

$$\begin{pmatrix} x_f(t) \\ y_f(t) \\ z_f(t) \end{pmatrix} = \begin{pmatrix} \alpha_x \cos(2\pi f t + \Phi_x) \\ \alpha_y \cos(2\pi f t + \Phi_y) \\ \alpha_z \cos(2\pi f t + \Phi_z) \end{pmatrix}$$

So we know α_k and Φ_k (with k= x, y, z) by the complex coefficient of the Fourier transform at the f frequency, ie :

$$X_k = \alpha_k e^{i\Phi_k}$$

The input complex coefficient X_k are named ‘Sx, Sy, Sz’ in the IDL test procedure below.

```
PRO test

PRINT, '============================================================='
PRINT, 'Test of copolarlib.pro  -  P. Robert, June 2015'
PRINT, '============================================================='
PRINT
PRINT, 'Ellipse parameters'
PRINT, '------------------'
PRINT
PRINT, 'One define here a right-handed elliptic wave, with K //Bo'
PRINT, 'Coordinates are: '
PRINT, '     X along the major axis of the ellipse'
PRINT, '     Y along the minor axis of the ellipse'
PRINT, '     Z along the DC magnetic field'

a=10.  ; length of the major axis (nT)
b=5.   ; length of the minor axis
F=4.   ; Frequency of the wave (Hz)
phi=45. ; initial phase of the B vector (degres)

PRINT
PRINT, 'a,b (nT)      : ',a,b
PRINT, 'exentricity   : ', SQRT(a*a -b*b)/a
PRINT, 'Frequency (Hz): ',F
PRINT, 'Phase (deg.)  : ',phi

PRINT
```

```
PRINT, 'Numerisation parameters'
PRINT, '----------------------'

Fr=20.48 ;Sampling rate (Hz)
N=512 ; number of point of the real input waveform
PI=3.141593
df=Fr/FLOAT(N)

; note that we choose F=4 and Sr=20.48 to have F=Kf*df
; with Kf integer (df=Fr/N).

Kf=FIX(FLOAT(N)*F/Fr)

PRINT, 'Sample rate (Hz)                : ',FR
PRINT, 'Number of point of the waveform : ',N
PRINT, 'Frequency resolution (Hz)       : ',df
PRINT, 'Ray number corresponding to F   : ',Kf
PRINT, 'Check than Kf*df = F            : ',Kf*df, F

x=FLTARR(N)
y=FLTARR(N)
z=FLTARR(N)

PRINT
PRINT, 'Waveform computation...

FOR i=0,N-2 DO BEGIN
    x(i)=a*COS(2.*PI*F*FLOAT(i)/Fr +phi*PI/180.)
    y(i)=b*SIN(2.*PI*F*FLOAT(i)/Fr +phi*PI/180.)
    z(i)=0.
ENDFOR

PRINT, 'Done!'

PRINT
PRINT, 'FFT of the 3 waveforms...'

Sx=FFT(x)
Sy=FFT(y)
Sz=FFT(z)

PRINT, 'Done!'

PRINT
PRINT, 'Polar computation for the frequency F...'
PRINT, '------------------------------------'

PRINT, 'input complex Fourier coefficient: (for ray # Kf)'
PRINT, Sx(Kf)
PRINT, Sy(Kf)
PRINT, Sz(Kf)

cpolar, Sx(Kf),Sy(Kf),Sz(Kf),ptot,a,b,rkx,rky,rkz,ax,ay,az,phase

PRINT, 'Done!'

PRINT
PRINT, 'Results for ray number ',Kf
PRINT, '--------------------------------'
PRINT
PRINT, 'output total power (nT2) : ',ptot
PRINT, 'major and minor axis of the ellipse (nT) : ',a,b
PRINT
PRINT, 'ouput K vector direction (modulo pi) normal to the ellipse plane:'
PRINT, 'Kx, Ky, Kz : ', rkx,rky,rkz
PRINT
PRINT, 'output major axis direction (modulo pi):'
PRINT, 'Ax, Ay, Az : ',ax,ay,az

PRINT
PRINT, 'output absolute phase (degree): ', phase*180./3.14159
PRINT
PRINT, '=============================================================='

END
```

2.5.2 Result

By lauching IDL and running the previous test procedure, after compilation of polarlib.pro library, we obtained the following results :

```
robert@lx-robert:/data/CACTUS_HOME/DOCS/20150424_Doc_Polar_Draft$ idl
IDL Version 8.2.3 (linux x86_64 m64). (c) 2013, Exelis Visual Information Solutions, Inc.
Installation number: 500640-1.
Licensed for use by: LPP- Palaiseau

IDL> .COMPILE polarlib
% Compiled module: CALAIFI.
% Compiled module: ELLIPSE.
% Compiled module: CPOLAR.

IDL> .COMPILE test
% Compiled module: TEST.

IDL> test
=================================================================
Test of copolarlib.pro  -  P. Robert, June 2015
=================================================================

Ellipse parameters
------------------

One define here a right-handed elliptic wave, with K //Bo
Coordinates are:
     X along the major axis of the ellipse
     Y along the minor axis of the ellipse
     Z along the DC magnetic field

a,b (nT)      :       10.0000      5.00000
exentricity   :      0.866025
Frequency (Hz):       4.00000
Phase (deg.)  :       45.0000

Numerisation parameters
-----------------------
Sample rate (Hz)                :       20.4800
Number of point of the waveform :      512
Frequency resolution (Hz)       :     0.0400000
Ray number corresponding to F   :      100
Check than Kf*df = F            :       4.00000      4.00000

Waveform computation...
Done!

FFT of the 3 waveforms...
Done!

Polar computation for the frequency F...
-------------------------------------
input complex Fourier coefficient: (for ray # Kf)
(      3.52944,      3.51906)
(      1.76925,     -1.76376)
(      0.00000,      0.00000)
% Compiled module: REAL_PART.
Done!

Results for ray number      100
--------------------------------

output total power (nT2) :       62.1636
major and minor axis of the ellipse (nT) :       9.96810      4.99641

ouput K vector direction (modulo pi) normal to the ellipse plane:
Kx, Ky, Kz :      -0.00000     -0.00000      1.00000

output major axis direction (modulo pi):
Ax, Ay, Az :      0.999998   0.00202547      0.00000

output absolute phase (degree):       44.8574

=================================================================
IDL>
```

We can see that we find the elliptical polarization defined as input, with $\vec{k}//\vec{Z}$, and the main characteristics:

$$
\begin{array}{lll}
a = & 9.96810 & \simeq 10. \\
b = & 4.99641 & \simeq 5. \\
e = & 0.865308 & \simeq 0.866025 \\
\varphi = & 44.8574 & \simeq 45. \\
 & & \\
Kz = & 1.0 & \\
Az = & 0.999998 & \simeq 1.0
\end{array}
$$

To have a complete spectrum, the `cpolar` routine must be called for each frequency of the complex arrays `Sx, Sy, Sz`. Note that the first $N/2$ points of the arrays `Sx(N), Sy(N), Sz(N)`, correspond to the frequency $f_i = (i-1)\delta f$ for $i = 1, N/2$.

The successive spectra are then plotted in the form of a time-frequency spectrogram, as we can see in the following chapter 3.

Chapter 3

Examples of experimental results

.

3.1 Description of plotted parameters

The following examples show the results of the visualisation program, which plot the main polarisation parameters in several pannels described below.

3.1.1 Total power

This is a classical spectrogram of the total power, in time-frequency diagram, given by the three input signal Sx(f), Sy(f) and Sz(f). Color scale is logarithmic, and corresponding values are given by the upper line of the color palet (top-right).

3.1.2 Exentricity

The corresponding spectrogram gives the value of the exentricity of the polarisation ellipse. Color code is dark-blue for e=0 (circular wave) and red for e=1 (degenerated ellipse along a line).

3.1.3 K direction

The K direction is in fact the normal to the plane of the ellipse, and is determined by the two polar angles θ and ϕ. Convention is following :

$$\theta < 90 : \text{ right handed mode}$$
$$\theta > 90 : \text{ left handed mode}$$

Color code for θ and ϕ are given in color palet top-right.

3.1.4 Major axis direction

As K vector, this direction is given by the two polar angles θ and ϕ.

3.1.5 Remarks

• If we have exentricity=0 (circular wave) the major axis direction is undetermined.
• If we have exentricity=1 (linear polarisation) K could be undetermined, but major axis direction is measured.
• If we have θ=0, ϕ is undetermined.

3.2 Circular polarisation with K // Bo

Very clear example showing a circular right handed polarisation (exentricity in blue, ie e=0, then circular wave, and θ_K in blue, ie $\theta_K = 0$, then right handed polarisation (see 3.1.3). As $\theta_K = 0$, ϕ_K is undetermined. As the circular polarization is slightly flattened, the direction of the major axis can nevertheless be determined ($\theta = 90°$).

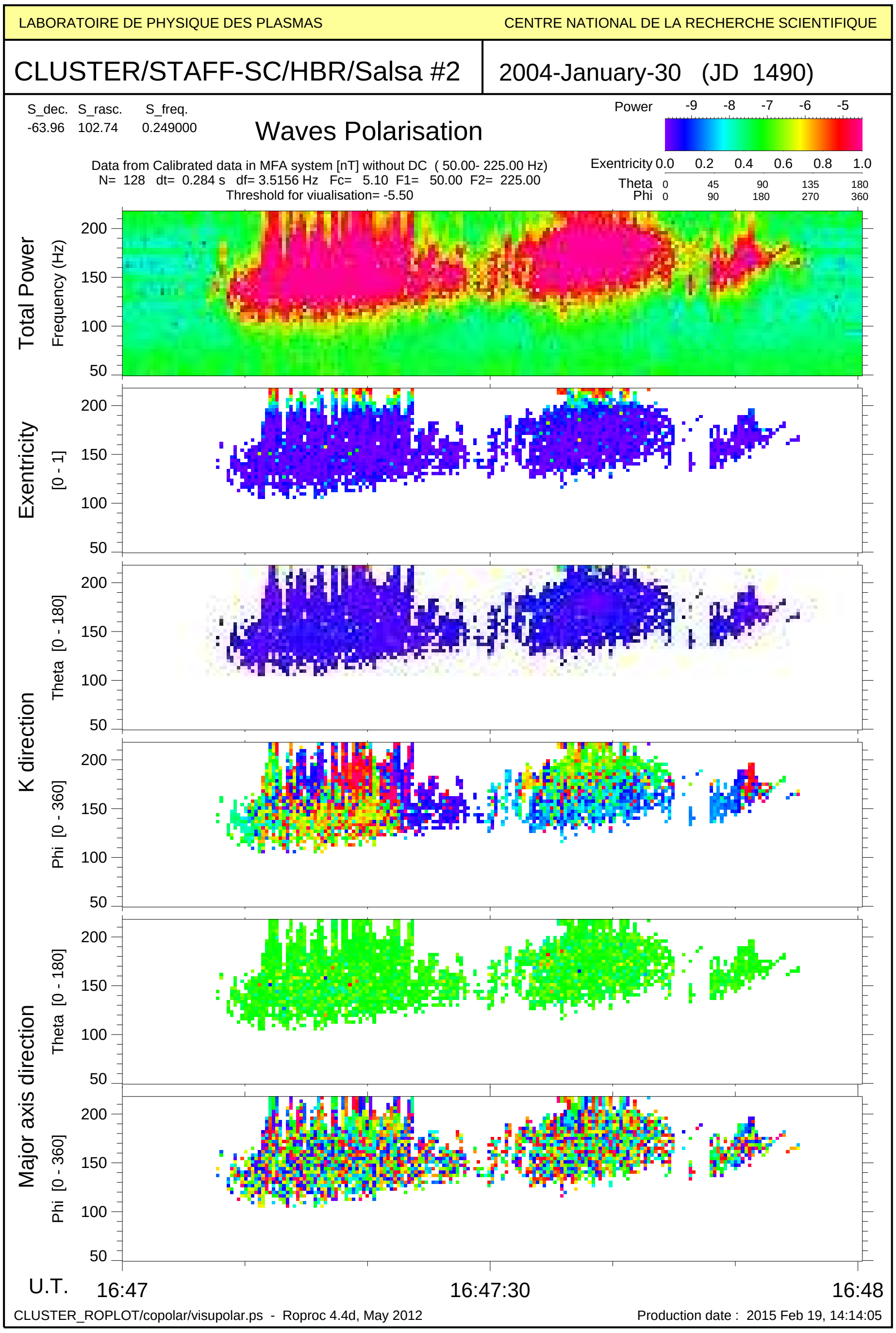

Figure 3.1: Polarisation parameters for a Whistler recorded in HBR mode on CLUSTER/STAFF-SC.

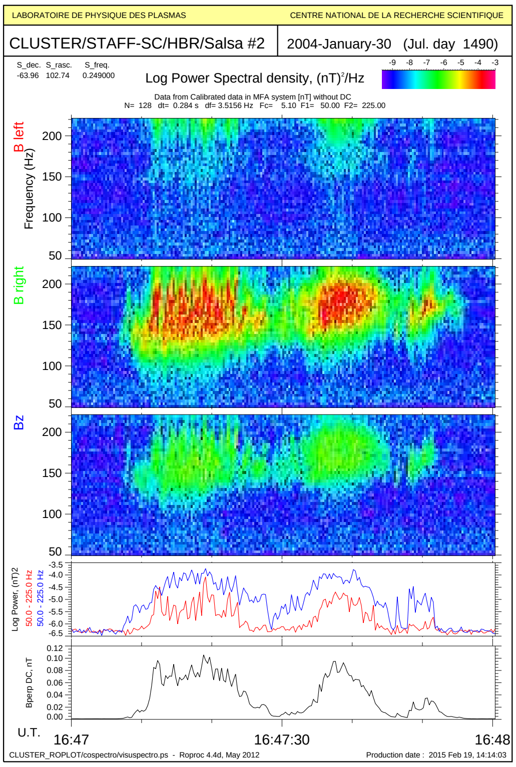

Figure 3.2: Classical spectrogram from the same event, in Left-Right-Z components.

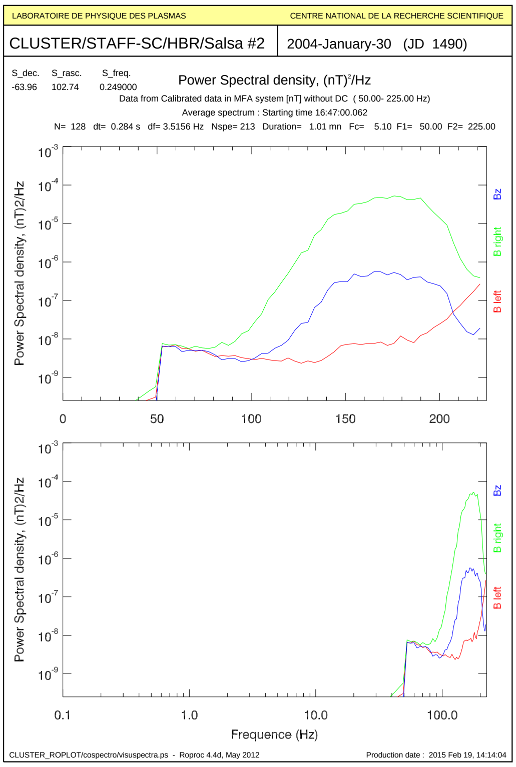

Figure 3.3: Average spectra of the above spectrogram, in Left-Right-Z components.

3.3 Linear polarisation with K // or ⊥ to Bo

First event at ∼ 06:58 is a large band event, with linear polarisation in the plane orthogonal to B_0, and a K direction parallel or antiparallel to B_0.
Second event at ∼ 07:10 is a linear monochromatic wave at ∼ 2 Hz, of same characteristics.
Third event, at ∼ 07:20-07:30 is a pseudo monochromatic wave, at about 10 Hz, always linear polarised (e∼1) but with K perpendicular to B_0 ($\theta_K = 90°$).

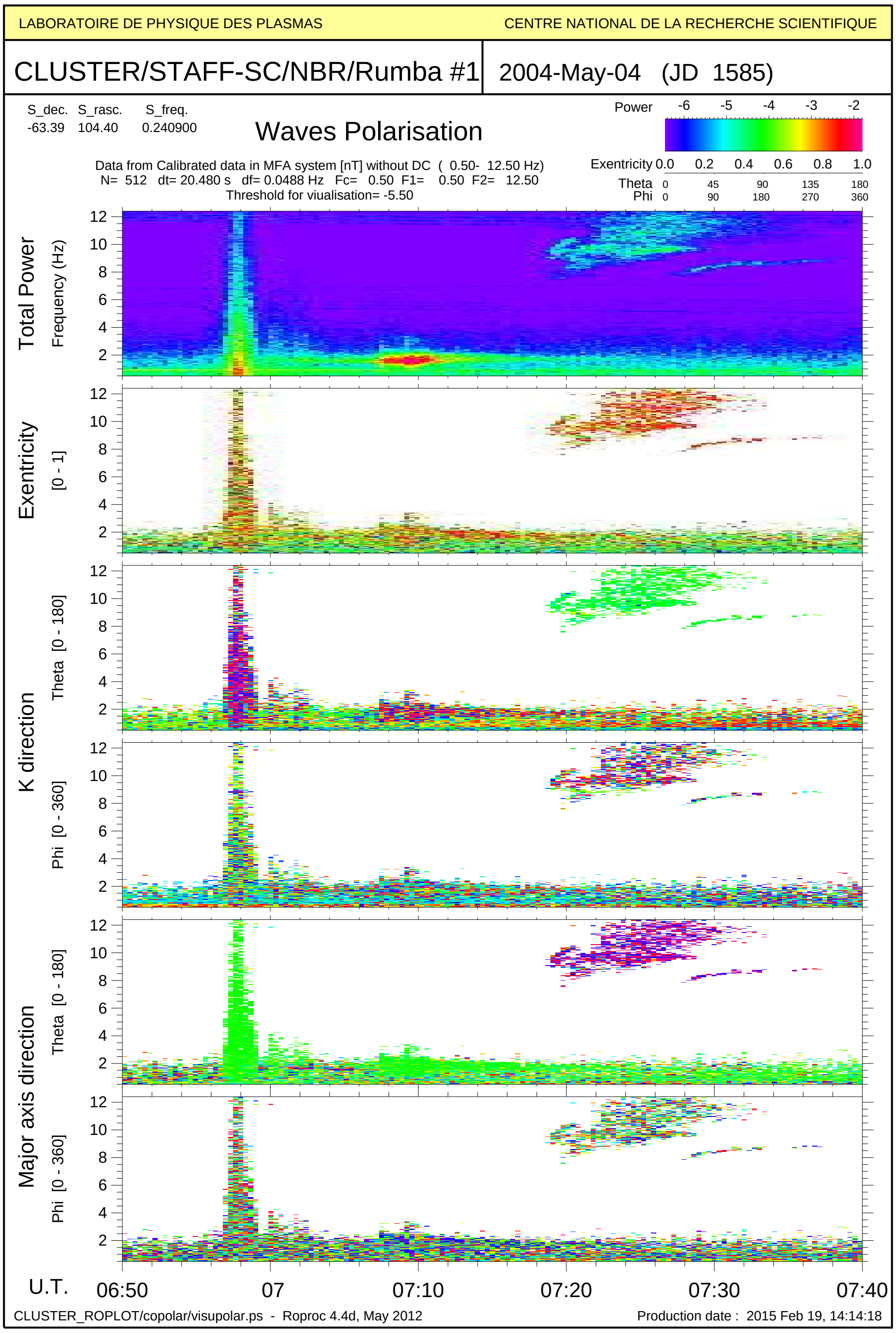

Figure 3.4: Polarisation parameters for 3 events recorded in NBR mode on CLUSTER/STAFF-SC.

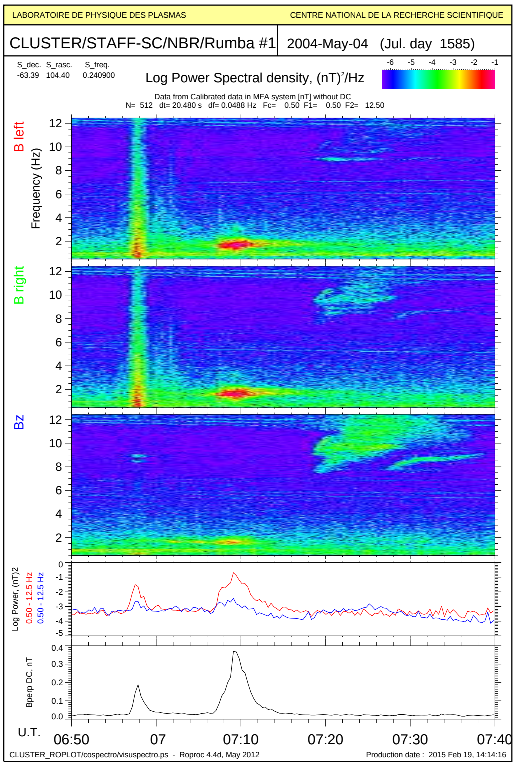

Figure 3.5: Classical spectrogram from the same event, in Left-Right-Z components.

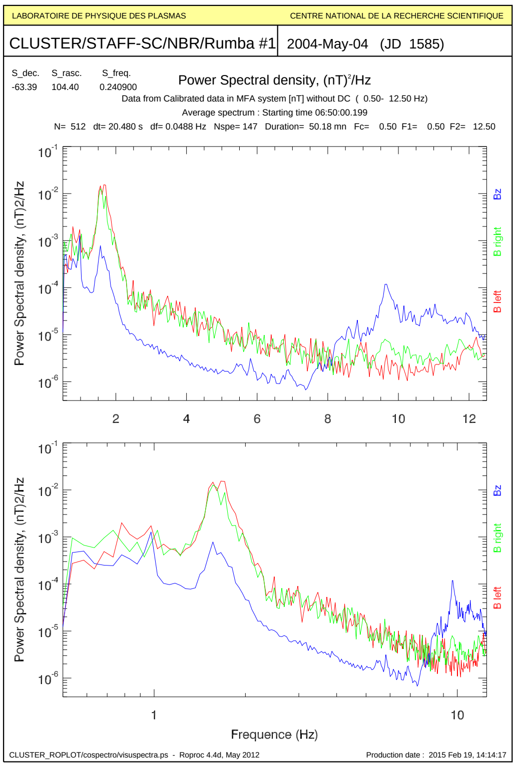

Figure 3.6: Average spectra of the above spectrogram, in Left-Right-Z components.

3.4 Linear monochromatic wave with harmonics with K $\perp$ to Bo

Nice case of monochomatic wave, with linear polarisation (e∼1) and a K vector perpendicular to B_0 ($\theta_K = 90°$). Note that this value of θ_K is confirmed by the direction of major axis along B_0 ($\theta = 0$ or 180).

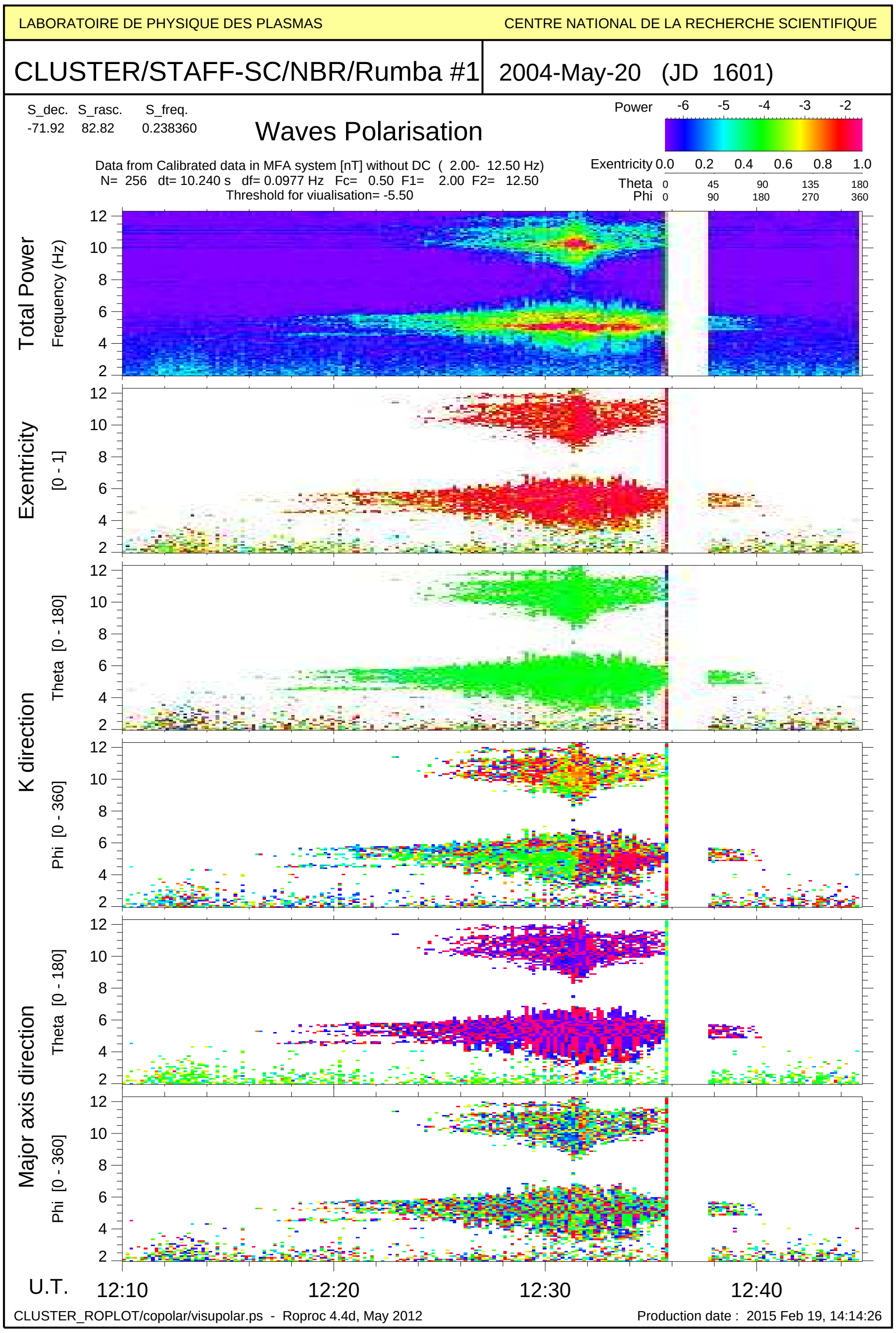

Figure 3.7: Polarisation parameters for a linear monochromatic wave along Bo, with harmonics.

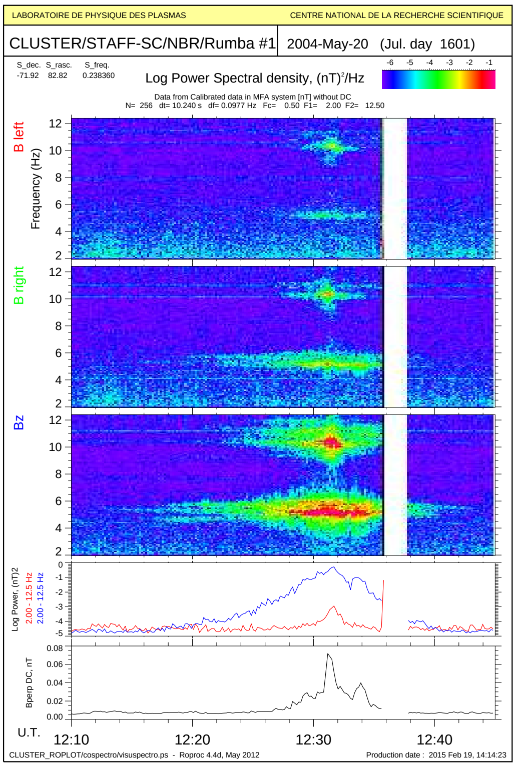

Figure 3.8: Classical spectrogram from the same event, in Left-Right-Z components.

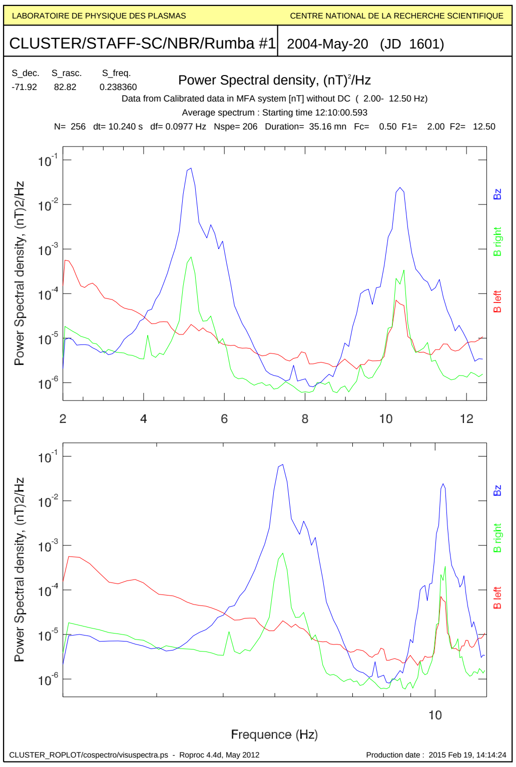

Figure 3.9: Average spectra of the above spectrogram, in Left-Right-Z components.

3.5 Same, but unpolarised waves before event

Same kind of wave than previous event, but with a slight frequency variation (from 09:30 to 09:50) and harmonics.
Before, and during the begining of the wave, unpolarised event appears at low frequency (∼2 Hz) .

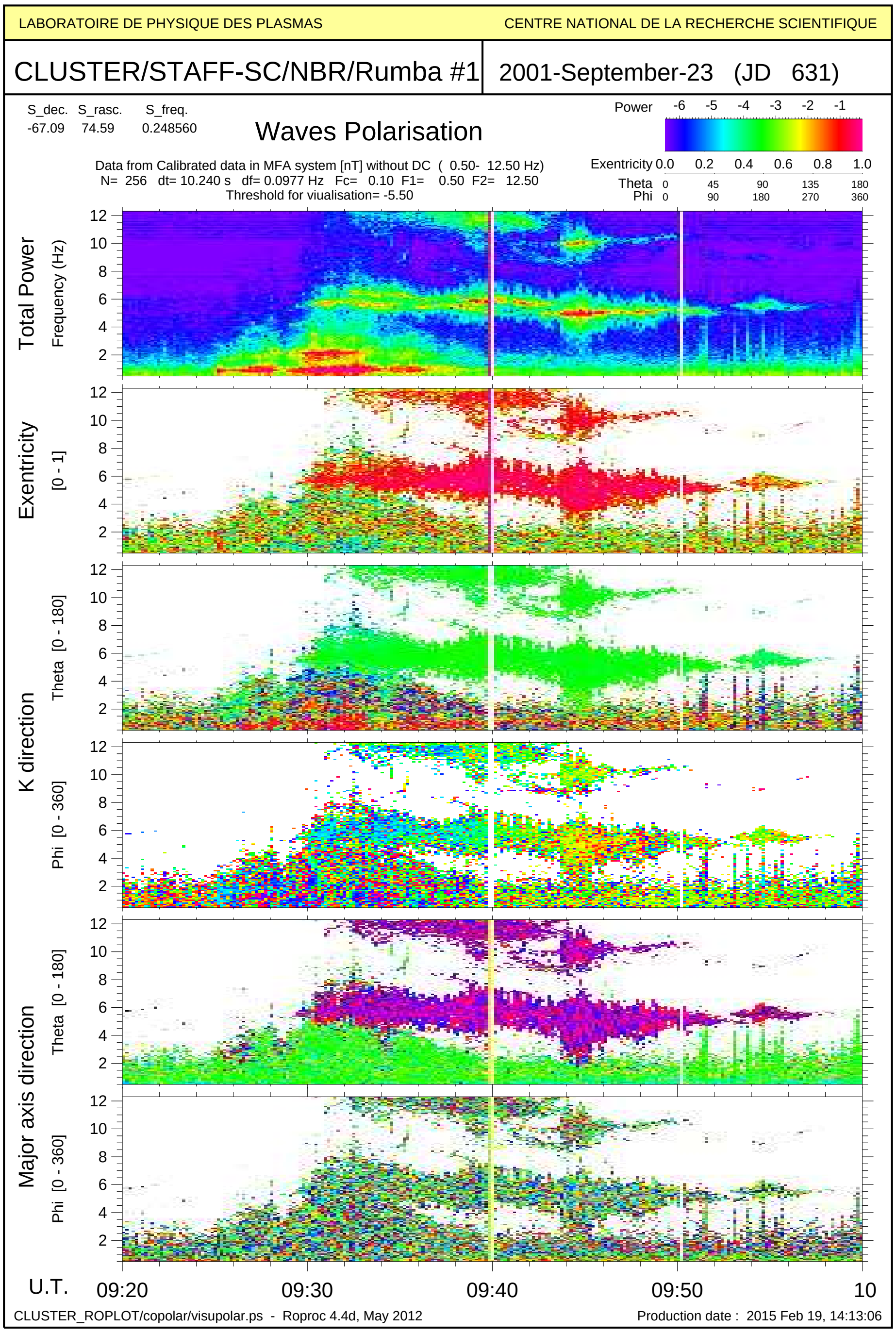

Figure 3.10: Polarisation parameters for a linear monochromatic wave along Bo.

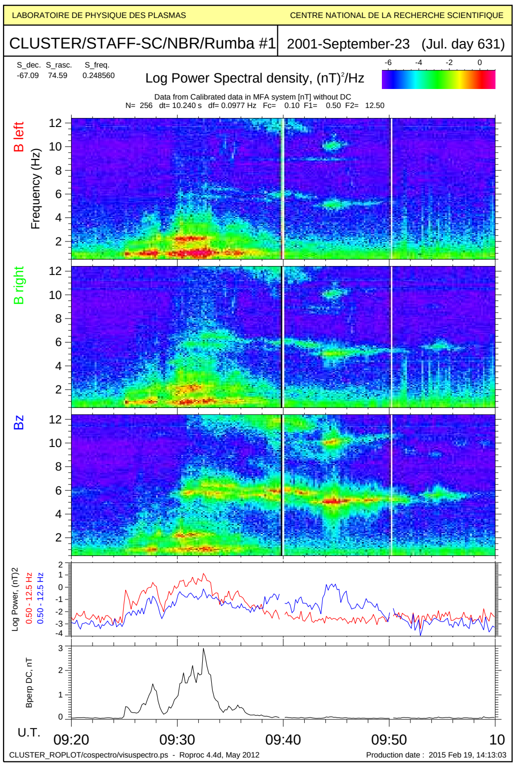

Figure 3.11: Classical spectrogram from the same event, in Left-Right-Z components.

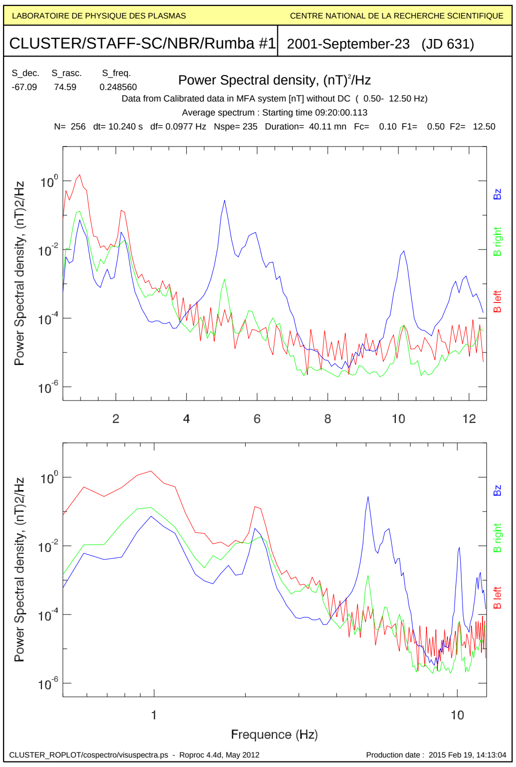

Figure 3.12: Average spectra of the above spectrogram, in Left-Right-Z components.

3.6 Linear monochromatic wave with numberous harmonics

Same kind of wave than previous events (linear polarisation along B_0), with a slight frequency variation and numberous harmonics.

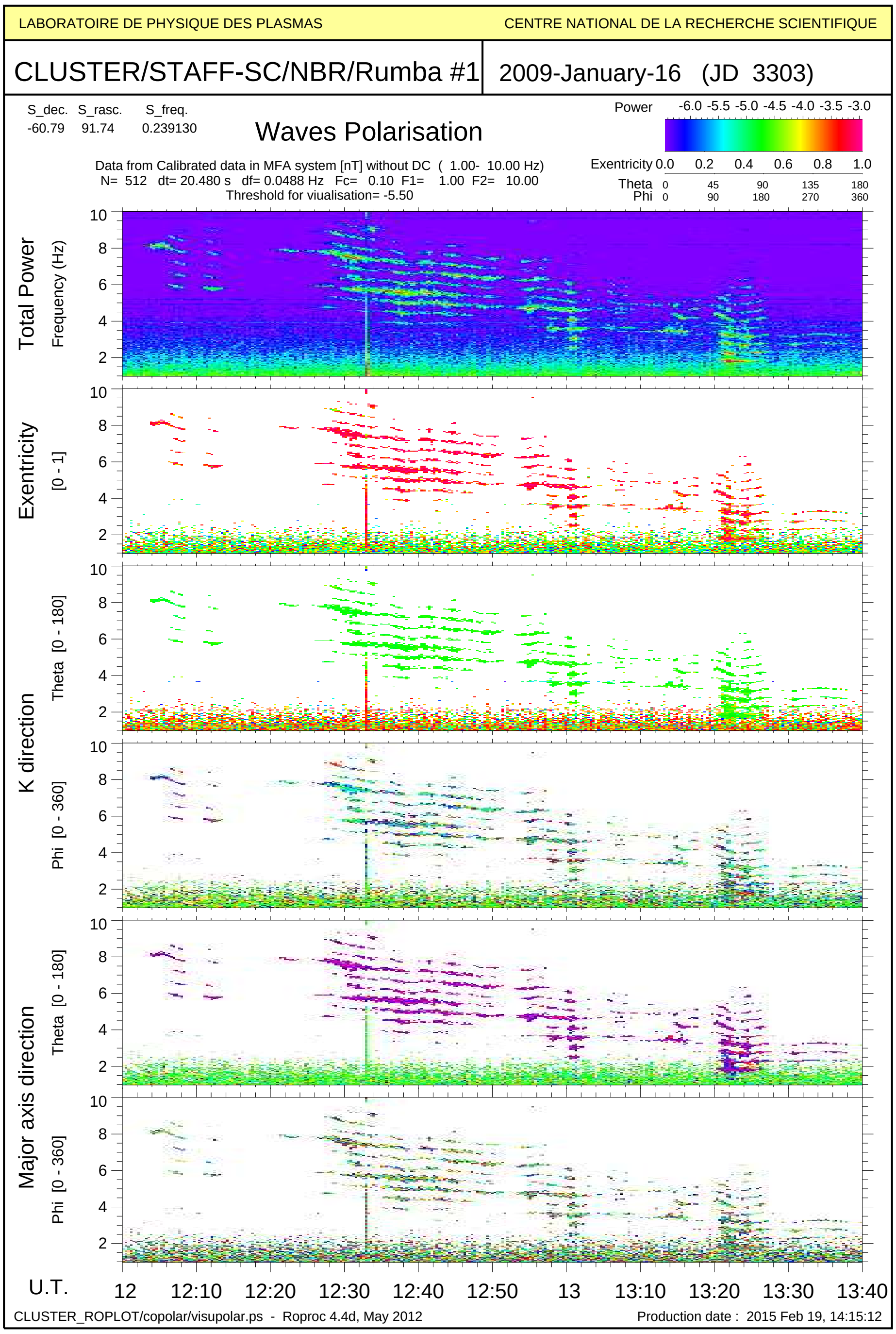

Figure 3.13: Polarisation parameters for a linear monochromatic wave along Bo, with numberous harmonics.

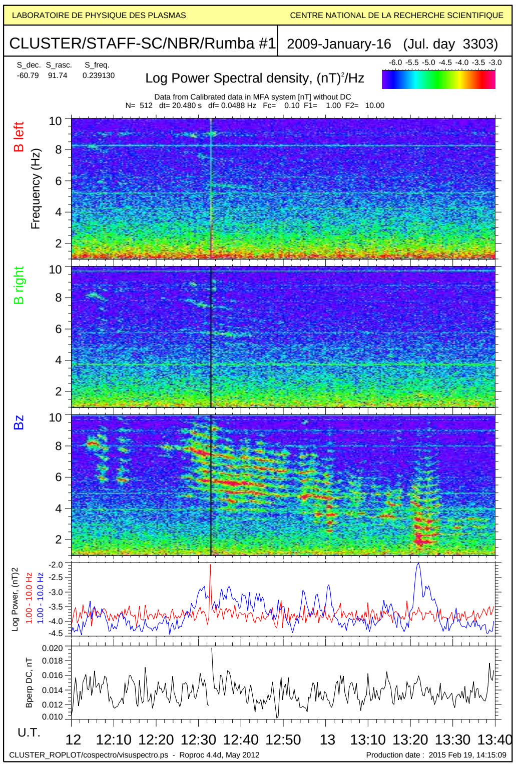

Figure 3.14: Classical spectrogram from the same event, in Left-Right-Z components.

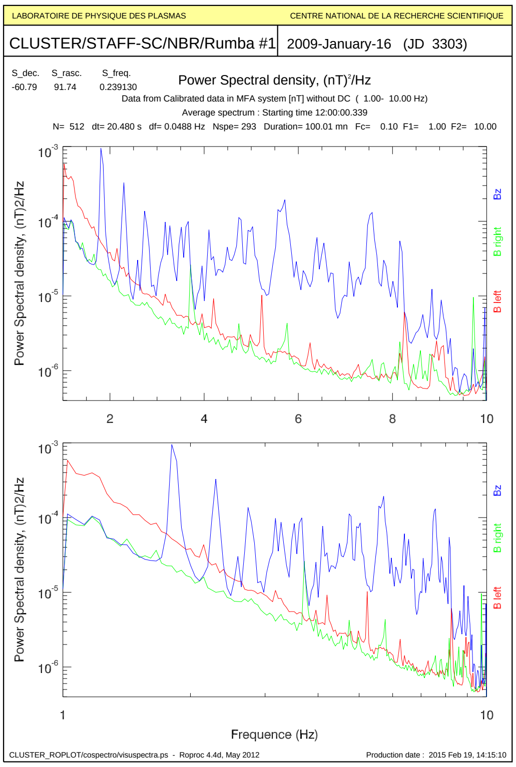

Figure 3.15: Average spectra of the above spectrogram, in Left-Right-Z components.

3.7 Perfect linear waves

Very clear example of linear polarisation along B_0 (exentricity=1, $\theta_K = 90°$).

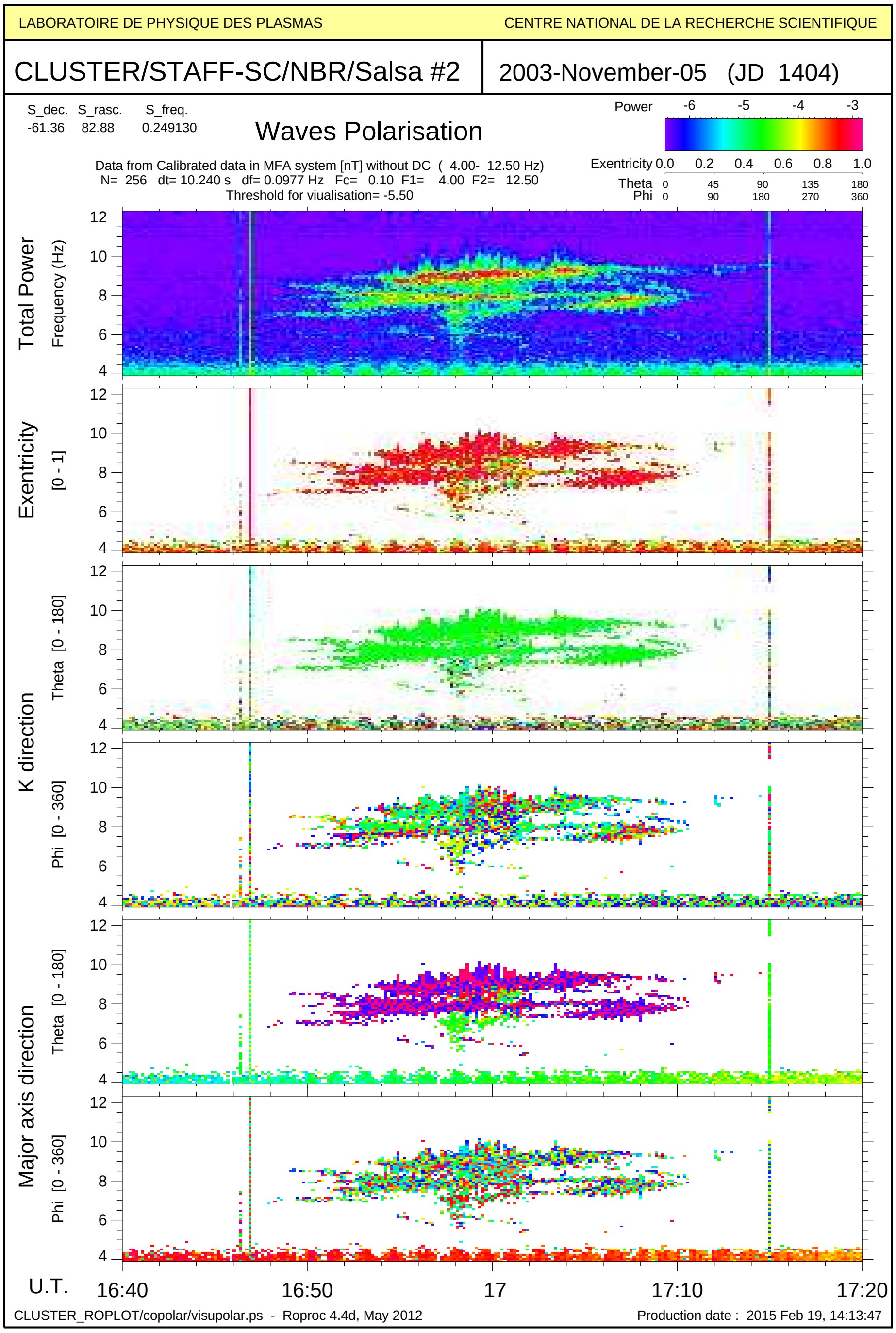

Figure 3.16: Polarisation parameters for perfect linear monochromatic wave along Bo.

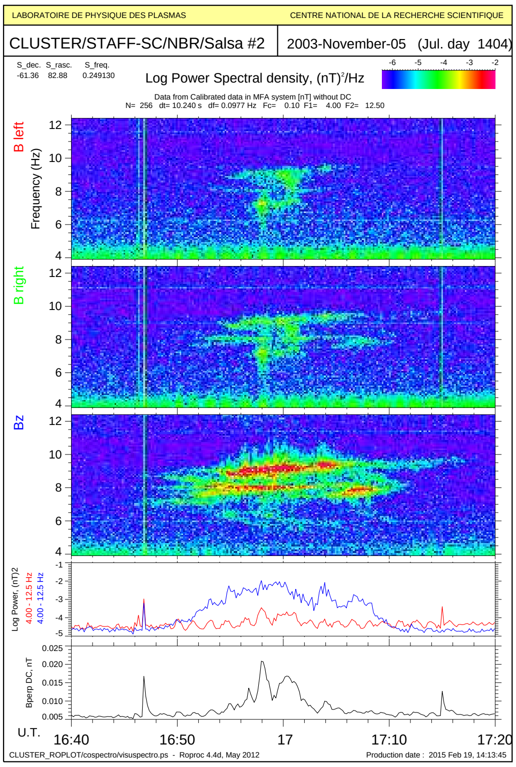

Figure 3.17: Classical spectrogram from the same event, in Left-Right-Z components.

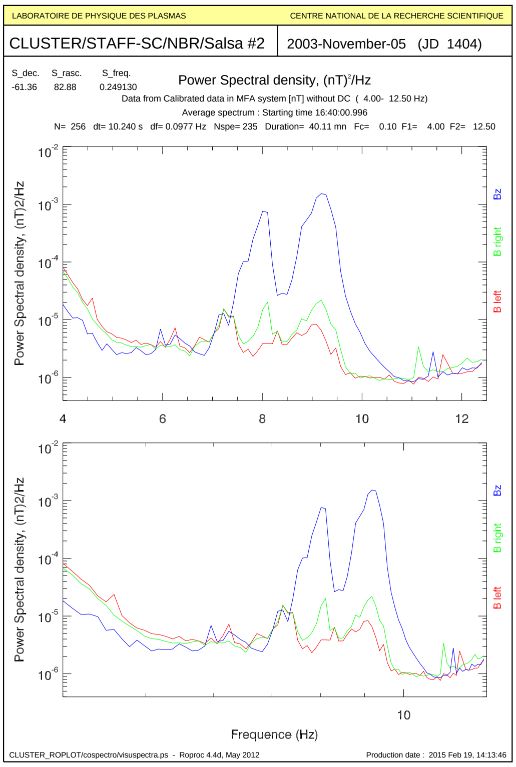

Figure 3.18: Average spectra of the above spectrogram, in Left-Right-Z components.

3.8 Same with time dependent frequency

Same kind of wave than previous example, but with a time variation of the main frequency.

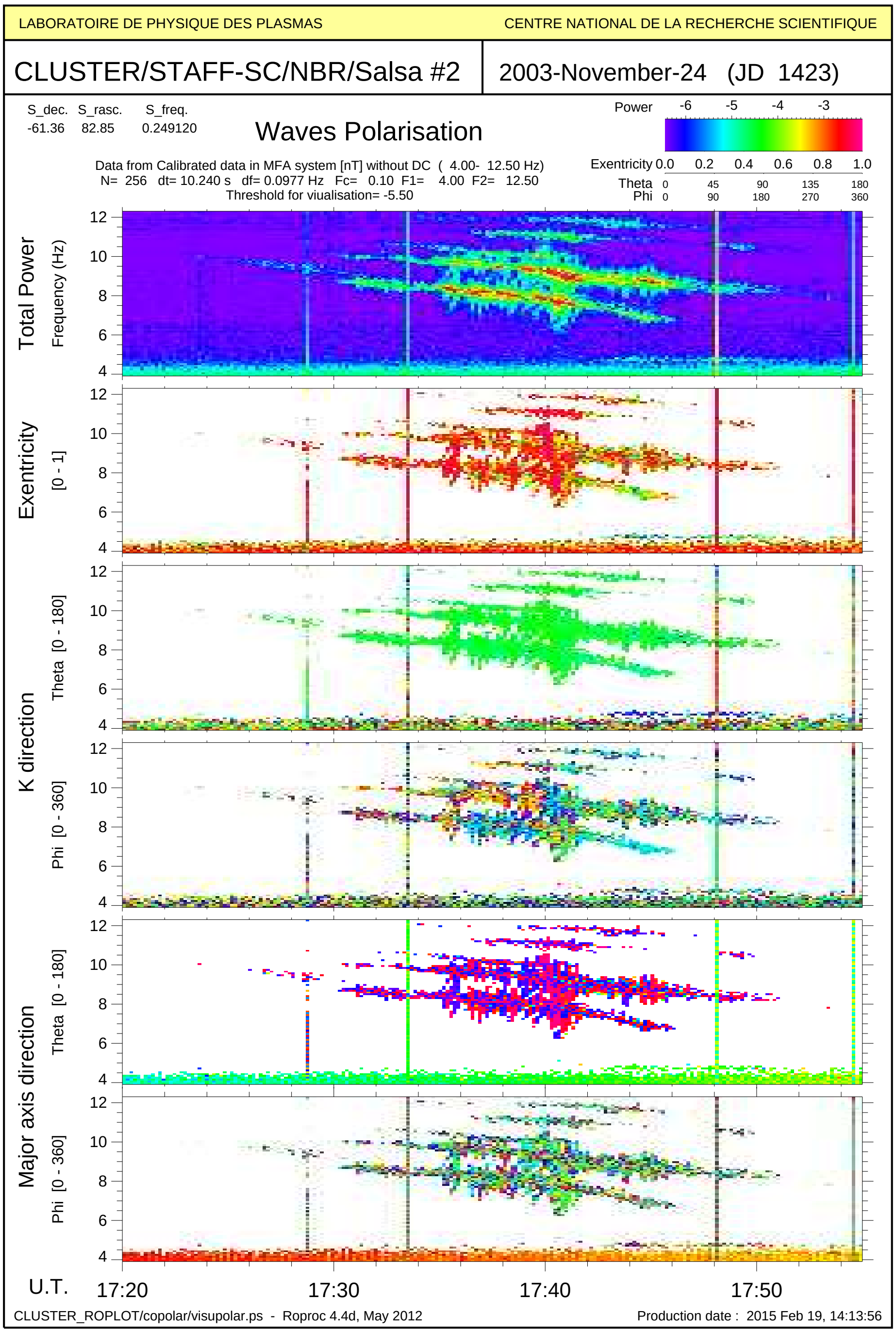

Figure 3.19: Polarisation parameters for a linear monochromatic wave along Bo with frequency variation.

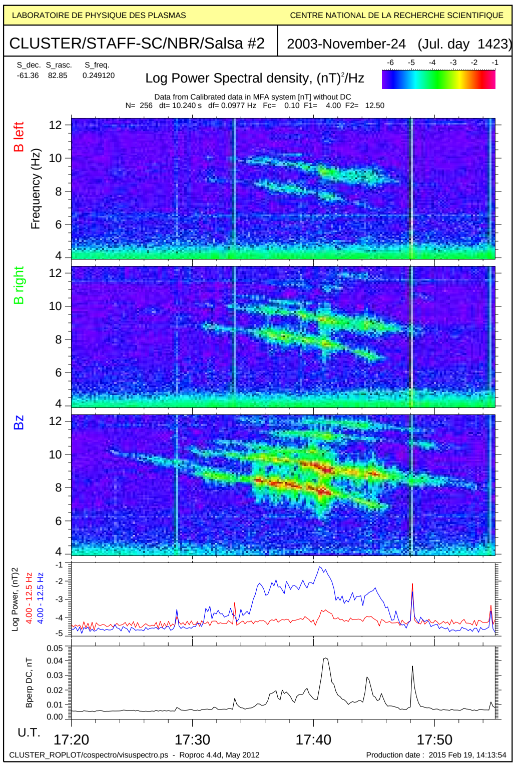

Figure 3.20: Classical spectrogram from the same event, in Left-Right-Z components.

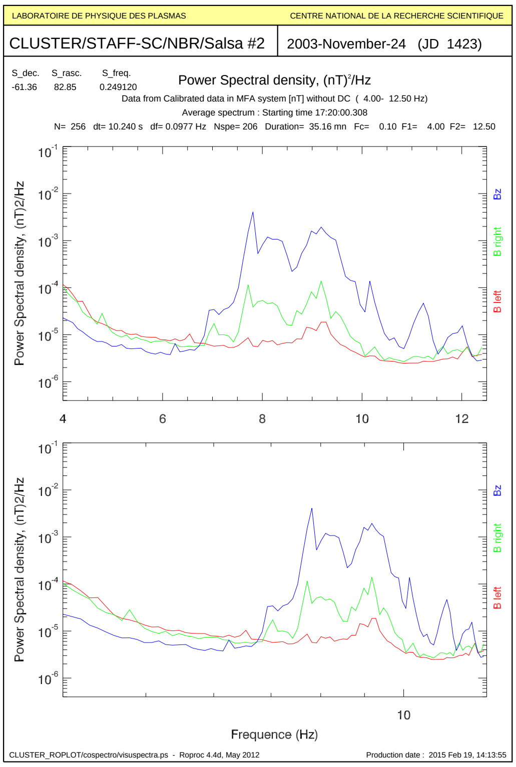

Figure 3.21: Average spectra of the above spectrogram, in Left-Right-Z components.

3.9 Large band circular polarisation, with k // Bo

Example of large frequency band event, with circular polarisation (exentricity ~ 0) and K$//B_0$ ($\theta_K = 0$).

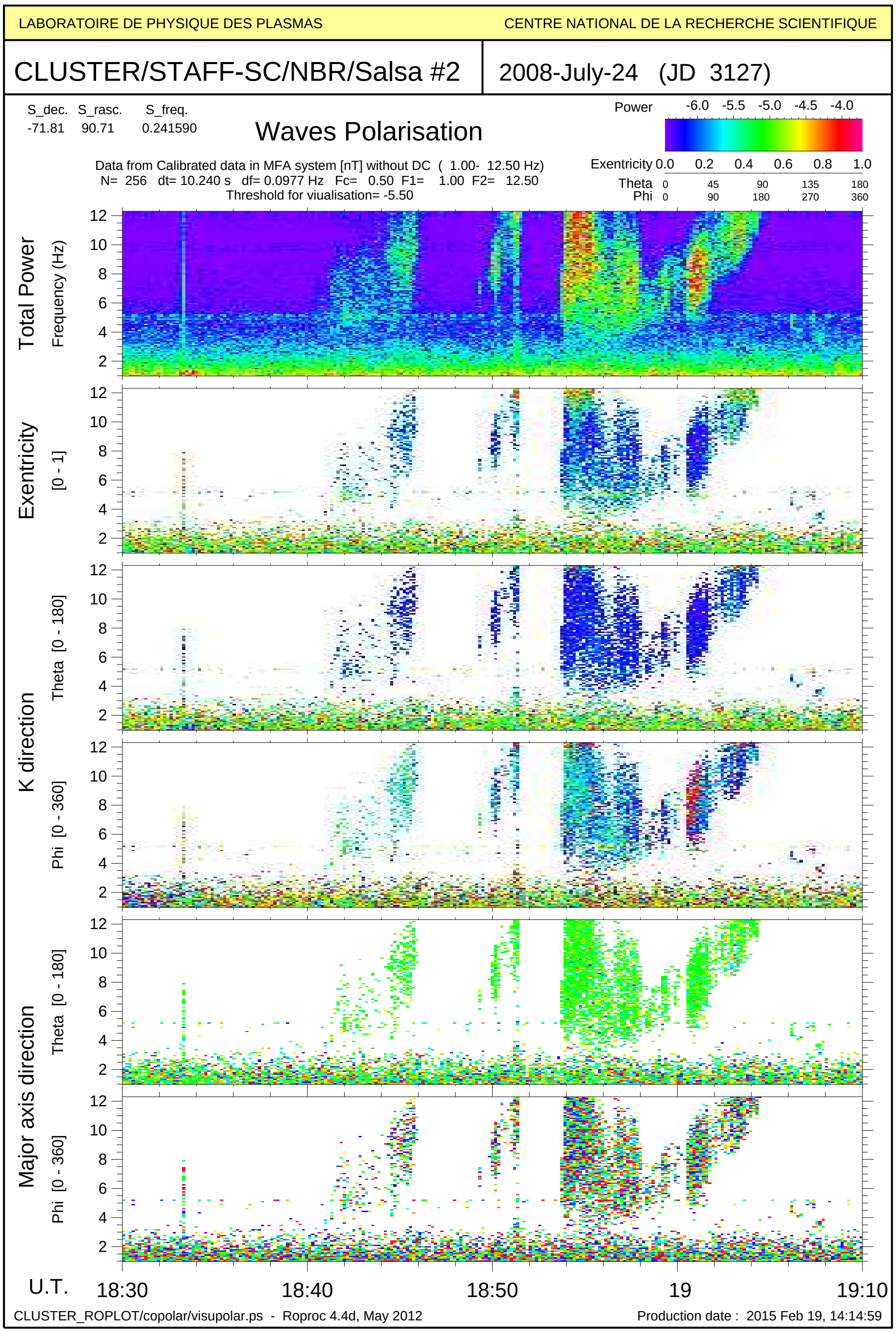

Figure 3.22: Polarisation parameters for large band circular polarisation, with k // Bo.

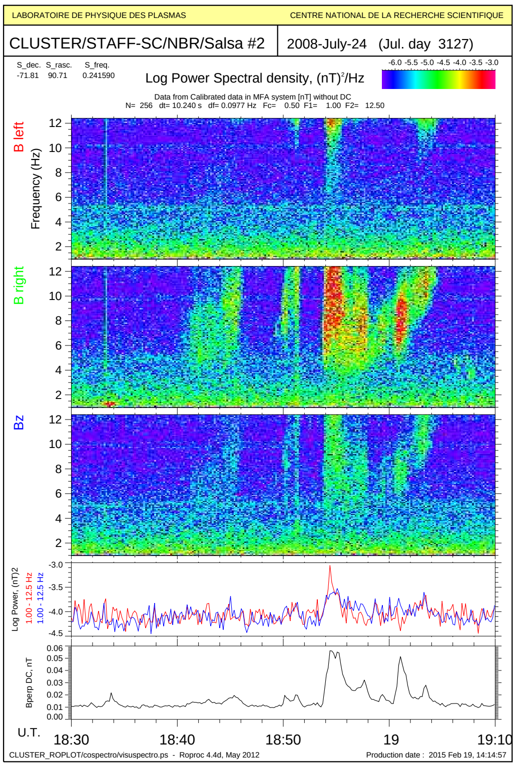

Figure 3.23: Classical spectrogram from the same event, in Left-Right-Z components.

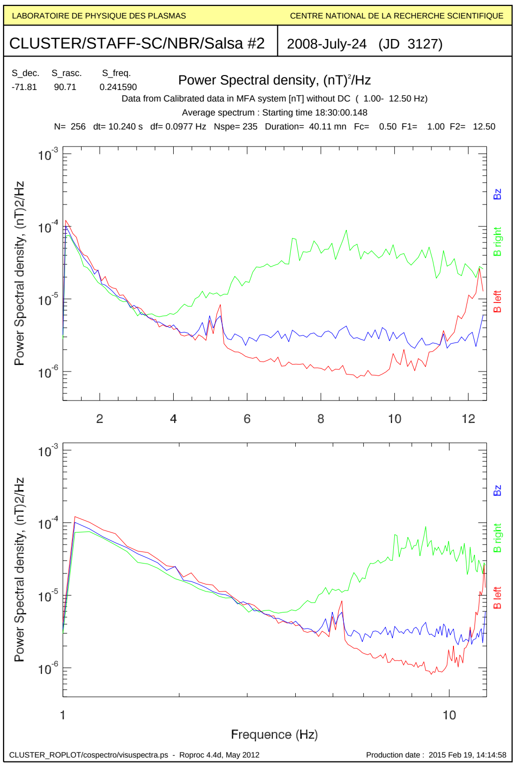

Figure 3.24: Average spectra of the above spectrogram, in Left-Right-Z components.

3.10 Other stable linear polarisation with constant frequency

Another example of linear polarisation along B_0 (exentricity=1, $\theta_K = 90°$) with constant frequency and harmonics.

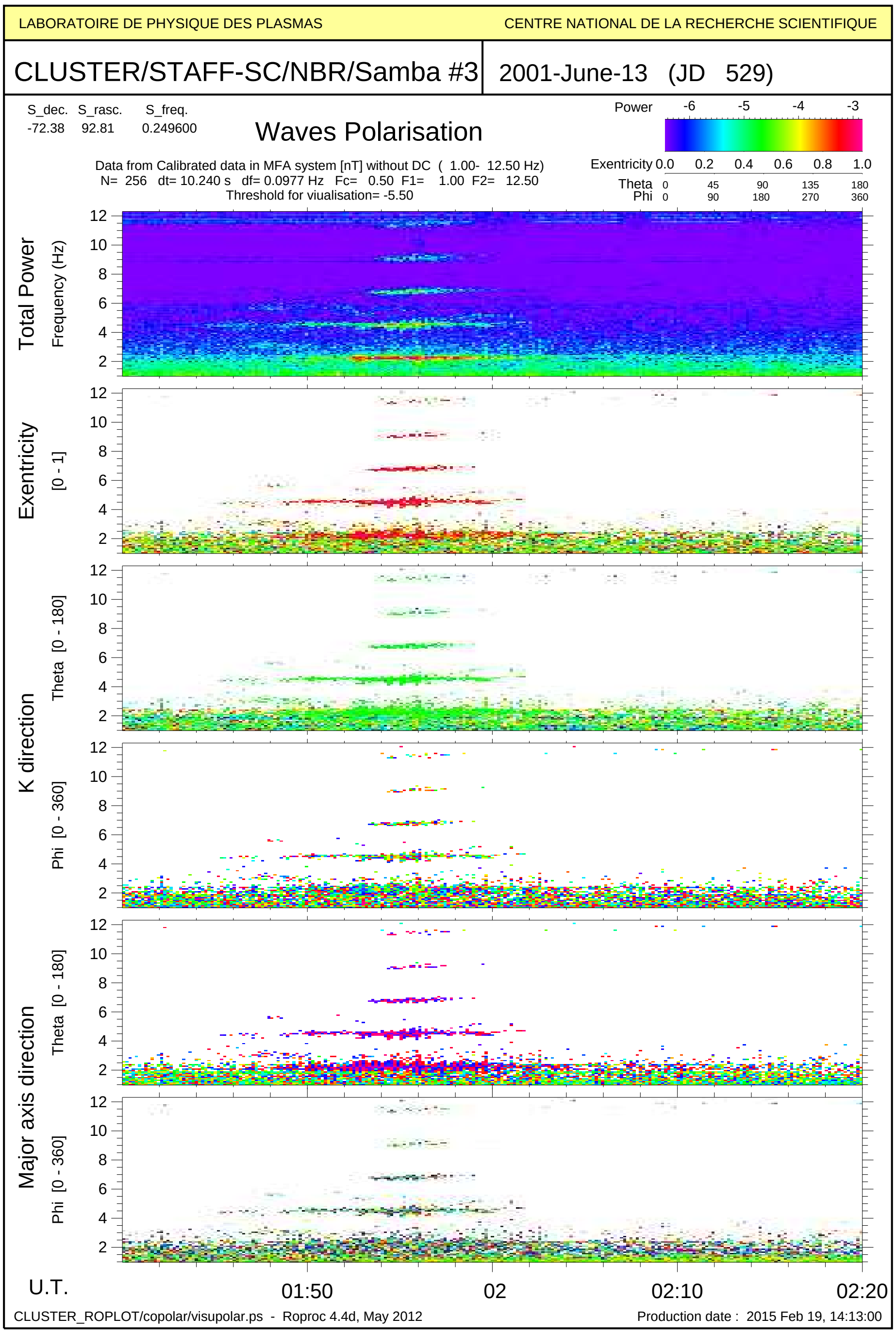

Figure 3.25: Polarisation parameters for stable linear polarisation with constant frequency.

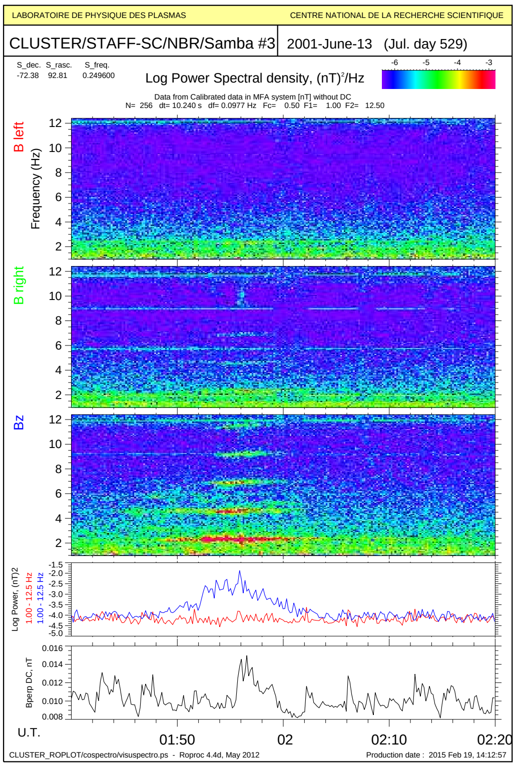

Figure 3.26: Classical spectrogram from the same event, in Left-Right-Z components.

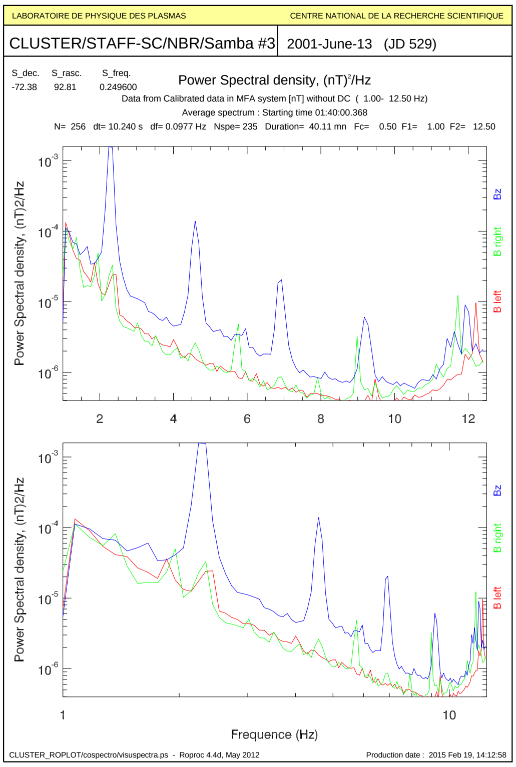

Figure 3.27: Average spectra of the above spectrogram, in Left-Right-Z components.

3.11 Perfect large band circular waves

Perfect example of large frequency band circular wave, with K // B_0. Exentricity=0, $\theta_K = 0$ (then right handed polarisation).
As the circular polarization is slightly flattened, the direction of the major axis can nevertheless be determined ($\theta = 90°$).

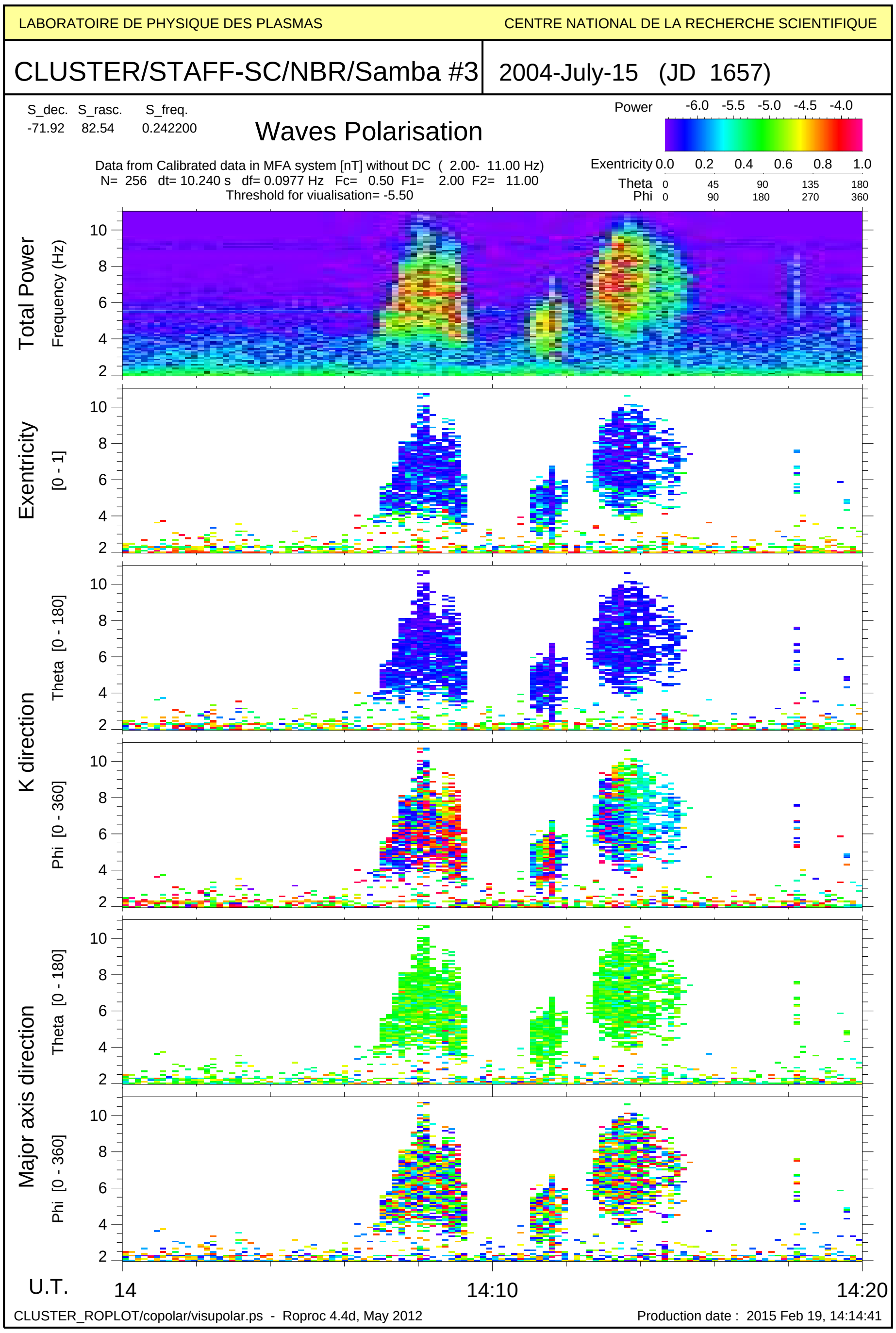

Figure 3.28: Polarisation parameters for perfect large band circular waves with K // B_0.

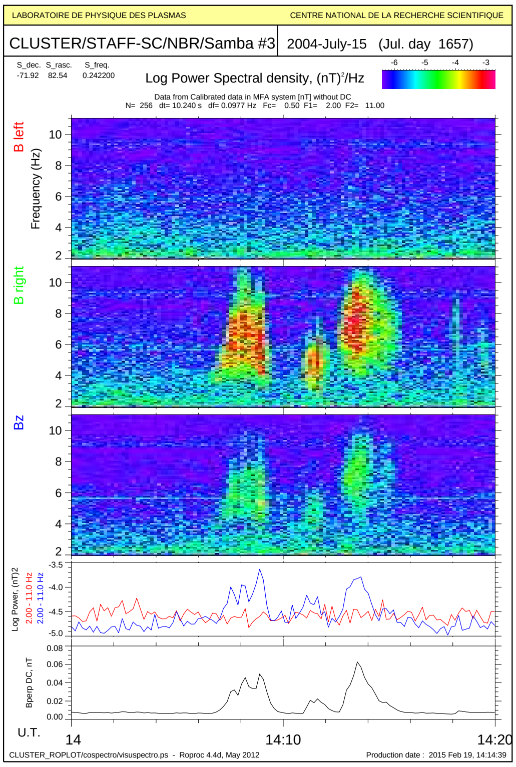

Figure 3.29: Classical spectrogram from the same event, in Left-Right-Z components.

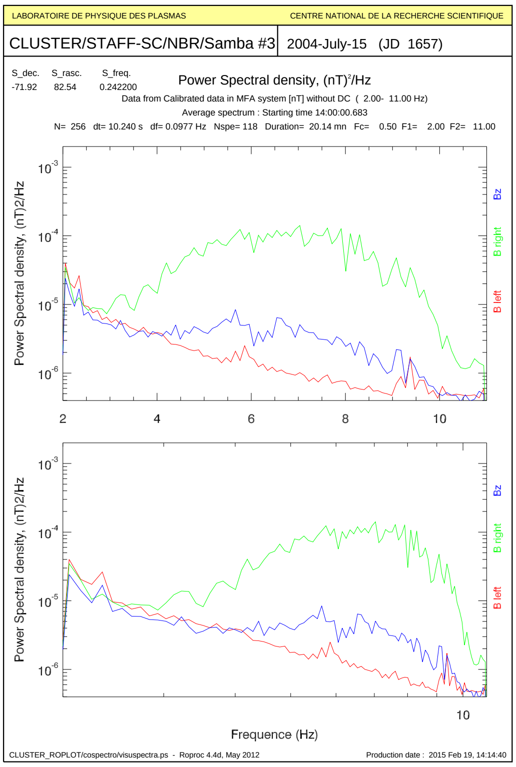

Figure 3.30: Average spectra of the above spectrogram, in Left-Right-Z components.

List of Figures

1.1 Elliptic plane wave for a single frequency, in a fixed measurement coordinate system. 1
1.2 Ellipse of polarisation for a $\vec{B}$ spining vector. 2

3.1 Polarisation parameters for a Whistler recorded in HBR mode on CLUSTER/STAFF-SC. 26
3.2 Classical spectrogram from the same event, in Left-Right-Z components. 27
3.3 Average spectra of the above spectrogram, in Left-Right-Z components. 27
3.4 Polarisation parameters for 3 events recorded in NBR mode on CLUSTER/STAFF-SC. 28
3.5 Classical spectrogram from the same event, in Left-Right-Z components. 29
3.6 Average spectra of the above spectrogram, in Left-Right-Z components. 29
3.7 Polarisation parameters for a linear monochromatic wave along Bo, with harmonics. 30
3.8 Classical spectrogram from the same event, in Left-Right-Z components. 31
3.9 Average spectra of the above spectrogram, in Left-Right-Z components. 31
3.10 Polarisation parameters for a linear monochromatic wave along Bo. 32
3.11 Classical spectrogram from the same event, in Left-Right-Z components. 33
3.12 Average spectra of the above spectrogram, in Left-Right-Z components. 33
3.13 Polarisation parameters for a linear monochromatic wave along Bo, with numberous harmonics. . 34
3.14 Classical spectrogram from the same event, in Left-Right-Z components. 35
3.15 Average spectra of the above spectrogram, in Left-Right-Z components. 35
3.16 Polarisation parameters for perfect linear monochromatic wave along Bo. 36
3.17 Classical spectrogram from the same event, in Left-Right-Z components. 37
3.18 Average spectra of the above spectrogram, in Left-Right-Z components. 37
3.19 Polarisation parameters for a linear monochromatic wave along Bo with frequency variation. . . . 38
3.20 Classical spectrogram from the same event, in Left-Right-Z components. 39
3.21 Average spectra of the above spectrogram, in Left-Right-Z components. 39
3.22 Polarisation parameters for large band circular polarisation, with k // Bo. 40
3.23 Classical spectrogram from the same event, in Left-Right-Z components. 41
3.24 Average spectra of the above spectrogram, in Left-Right-Z components. 41
3.25 Polarisation parameters for stable linear polarisation with constant frequency. 42
3.26 Classical spectrogram from the same event, in Left-Right-Z components. 43
3.27 Average spectra of the above spectrogram, in Left-Right-Z components. 43
3.28 Polarisation parameters for perfect large band circular waves with K // B_0. 44
3.29 Classical spectrogram from the same event, in Left-Right-Z components. 45
3.30 Average spectra of the above spectrogram, in Left-Right-Z components. 45

NOTES

Write here your personnal notes.

Last words

The examples shown here were made on the CLUSTER data. In the past, many linear and circular polarizations have also been put in evidence on the GEOS data. The programs given here can be applied to any waves data of any source, provided that the assumption "plane wave" can be made.

The "plane wave" assumption can be checked by using the method of the Minimum Variance Analysis (MVA).