The GENRAY Ray Tracing Code

A.P. Smirnov (Moscow State University) & R.W. Harvey (CompX)
CompX, P.O. Box 2672, Del Mar, CA 92014
Report CompX-2000-01, Ver. 2

March 17, 2003

Contents

1 Ray tracing equations **6**
- 1.1 Coordinate system . . . 6
- 1.2 Geometric optics equations . . . 6
- 1.3 Normalized variables and geometrical optics equations . . . 7

2 Magnetic field **9**
- 2.1 Axisymmetric magnetic field . . . 9
- 2.2 Variables in the equilib.dat file . . . 9
- 2.3 Ripple magnetic field . . . 10
- 2.4 General radial coordinates . . . 11

3 Plasma components **12**
- 3.1 Plasma species . . . 12
- 3.2 Density, temperature and Z_{eff} profiles . . . 12
  - 3.2.1 Axisymmetric radial profiles . . . 12
  - 3.2.2 Non axisymmetric density variations . . . 13
  - 3.2.3 Plasma neutrality and plasma charge Z_{eff} . . . 13

4 Dispersion relations **19**
- 4.1 Magnetized cold plasma . . . 19
- 4.2 Magnetized hot non-relativistic plasma (id=6) . . . 20
  - 4.2.1 General formula . . . 20
  - 4.2.2 Computation . . . 22
- 4.3 Magnetized electron relativistic plasma in Mazzucato approximation . 24
- 4.4 Magnetized electron relativistic plasma in Shkarofsky approximation . 24
- 4.5 Ono dispersion function for fast waves. . . . 24
- 4.6 The dispersion relation for non-Hermitian dielectric tensors. . . . 31
- 4.7 Change in the dispersion function along the ray. . . . 31

5 Wave power absorption **33**
- 5.1 Mazzucato approximation of the relativistic dielectric tensor for electron plasma . . . 33
- 5.2 Anti-Hermitian relativistic dielectric tensor for electron plasma . . . 34
  - 5.2.1 The tensor formula. . . . 34

5.2.2 Cyclotron resonance curves 36
5.2.3 The distribution functions. 38
5.3 Anti-Hermitian dielectric tensor for hot non-relativistic electron-ion plasma 40
5.4 Fast wave absorption 40
5.5 Lower hybrid wave absorption 40

6 Stochastic wave scattering 41

7 Wave launch and reflection 42
7.1 Ray cone 42
7.2 Approximation of grill antenna launch. 43
7.2.1 Location on the grill. 43
7.2.2 Power distribution along one grill. 44
7.2.3 Power spectrum. 45
7.2.4 Approximation of one *i* grill 46
7.2.5 The given N_parallel determines N_toroidal and N_poloidal. . 47
7.2.6 The given N_parallel and N_poloidal. 47
7.2.7 The given N_parallel and the angle between N_perpendicular and $\nabla\psi$. 48
7.2.8 The given toroidal and poloidal refractive index components. . 49
7.2.9 The determination of cutoff LH and FW point near the plasma edge. 50
7.3 EC wave launch for O-X-B mode conversion scenarios 50
7.4 Initial refractive index vector inside the plasma 51
7.5 Reflection from the plasma vacuum boundary 54

8 Electric field polarization 55

9 Current drive 57

10 Power and current drive radius 58

11 Numerical methods 59
11.1 Numerical solution of ray-tracing equations 59
11.2 Hamiltonian conservation 59
11.3 Differentiation of the dispersion function 61
11.4 Algorithm giving the output data at the set radial steps 61

12 Electron cyclotron emission. 63
12.1 Radiation transport equation for the the radiative power. 63
12.2 Radiation temperature. 67
12.3 Emission plots. 67

13 GENRAY usage **69**
13.1 Preliminaries 69
13.2 Genray flowchart 70
13.3 Parameter input 73

14 Examples **76**
14.1 Fast wave in D3D in cold plasma 76

Abstract/Overview

GENRAY (A.P.Smirnov and R.W.Harvey, 1995) [1] is a general ray tracing code for the calculation of electromagnetic wave propagation and absorption in the geometrical optics approximation.

Physical model:

1) A solution of the ray tracing equations (I.B.Bernstein and L.Friedland, 1983) [2] in general non-axisymmetric geometry is obtained for an arbitrary form of the magnetic surfaces in toroidal geometry. The magnetic field is input with a numerical **eqdsk** file giving the axisymmetric flux surfaces. Additional perturbations of the toroidal and poloidal magnetic field may be specified in order **t**o study the influenc**e** of the magnetic ripple.

2) The code provides several alternative dispersion functions D: (1) cold plasma dispersion with different numbers of species, (2) full (T.H.Stix, 1992)[3] hot non-relativistic plasma dispersion with shifted two-temperature $(T_{\|}, T_{\perp})$ Maxwellian distributions and multiple plasma species coded by Forest, and (3) the relativistic electron plasma using the (E.Mazzucato et al., 1987) [5] or I.P.Shkarofsky dispersion functions. Using these dispersion functions, ray tracing is performed for several types of radio waves: ECR, LH, FW, EBW, IBW....

3) The radial profiles of density, temperature and effective charge Z_{eff} can be input in pseudo-parabolic form or in tables for spline fits. In order to interface flexibly with related codes, a choice of variables for the minor radius is provided: the square root of toroidal volume, toroidal or poloidal flux, and poloidal cross-sectional area.

4) The code can recalculate the plasma density or Z_{eff} using the given temperature and plasma pressure specified in the **eqdsk** file. Plasma density perturbations in the form of the toroidal and poloidal harmonics may be specified.

5) A model is provided for angular scattering of the perpendicular refractive index $\overrightarrow{N}_{\perp}$ of the rays. This enables, for example, study of the effects of wave scattering by drift waves at the reflection point near the plasma boundary.

6) The calculation of absorption of wave power along the wave trajectory is determined by the imaginary part of the perpendicular refractive index $ImN_{\perp}$ using : (1) the anti-hermitian part of the dielectric tensor for hot non-relativistic plasma; (2) the relativistic electron plasma Mazzucato approximation for the fully relativistic electron plasma; or (3) the asymptotic formula for $ImN_{\perp}$ for LH and FW.

7) The code has two models for wave launch: (1) a cone of rays launched from an arbitrary point outside the plasma (for EC waves); and (2) grill launch of a specified $n_{\|}$-power spectrum $P(n_{\|})$ of waves from points inside the plasma (for LH, FW, BW). The code then calculates the set of trajectories for these types of wave launch.

8) Along the trajectory the code calculates the current drive efficiency using: (1) the asymptotic formula for FW and LH waves; (2) the CURBA code (R.H.Cohen) that takes into consideration relativistic resonance conditions and the influence of trapped electrons.

9) Along the trajectory, the code calculates the wave electric field polarization (for coupling to the CQL3D Fokker-Planck code) using several approximations for the dielectric tensor.

10) Based on linear damping of the wave energy, the code calculates radial profiles of absorbed power and current drive**.**

Numerical method

1) For the numerical solution of the ray-tracing equations the code can use: a 4th order Runge-Kutta method with constant and variable time step, or a 5th order Runge-Kutta method with variable time step.

2) The ray-tracing equations have the form of Hamiltonian equations. The dispersion function plays the role of a Hamiltonian in these equations. For Hamiltonian conservation the code can use different forms of corrections along the trajectory.

3) The code has the capability to form the numerical derivatives required for differencing of the ray equation, given only an expression for the local wave dispersion relation. This enables ready use of given dispersion relation solvers for ray tracing. The cold plasma and Stix dispersion relation in the code also have explicit calculation of derivatives from analytic expressions.

The code has a modular structure that permits adding new forms of magnetic fields, dispersion functions, absorption formula, and numerical methods.

Chapter 1

Ray tracing equations

1.1 Coordinate system

We use the cylindrical space coordinates $\vec{R} = (r, \varphi, z)$; r is the major radius, φ is the toroidal angle, and z is along the vertical axis. The conjugate coordinates for the refractive index $\vec{N} = \frac{c\vec{k}}{\omega}$ are $\vec{N} = (N_r, M = rN_\varphi, N_z)$. Here $\vec{k}$ is wave vector; c is speed of light; $\omega = 2\pi f$; f is wave frequency. We will use t for time.

1.2 Geometric optics equations

If the wave length is smaller than the local space scale and the wave frequency is bigger than the inverse time of the change of the plasma parameters, then we can use the geometric optics equations for the wave description (B.I.Bernstein and L.Friedland,1983) [2]. In the geometric optic approximation the wave electric field is composed of the slowly varying amplitude $\vec{E}(\vec{R}, t)$ and the fast variation $\exp(i\psi(\vec{R}, t))$. The function ψ determines the wave vector $\vec{k}(\vec{R}, t)$ and the wave frequency $\omega = \omega(\vec{R}, t)$; $\vec{k} = \nabla\psi$, $\omega = -\partial\psi/\partial t$. The quantities $\vec{k}$ and ω are the slowly varying functions. The wave electric field is determined by the dielectric tensor $\hat{\varepsilon}(\omega, \vec{k})$, written for the uniform plasma with local parameters along the ray.

Let $N_\parallel = (\vec{N} \cdot \vec{B})/B$ be the longitudinal component of the refractive index along the magnetic field $\vec{B}$; $\vec{N}_\perp = (\vec{N} - \vec{N}_\parallel)$ is the perpendicular component of the refractive index. The local orthogonal coordinate system $(\vec{e}_x, \vec{e}_y, \vec{e}_z)$[3] for the wave electric field polarization $\vec{E}$ can be introduced by the following formula:

$$\vec{e}_z = \frac{\vec{B}}{B}, \ \vec{e}_x = \frac{\vec{N}_\perp}{N_\perp}, \ \vec{e}_y = [\vec{e}_z \times \vec{e}_x]. \tag{1.1}$$

In this system of coordinates, the wave electric field and the refractive index have the form:

$$\overrightarrow{E} = E_x \overrightarrow{e}_x + E_y \overrightarrow{e}_y + E_z \overrightarrow{e}_z, \tag{1.2}$$

$$\overrightarrow{N} = N_\perp \overrightarrow{e}_x + N_\parallel \overrightarrow{e}_z. \tag{1.3}$$

The amplitude of the wave electric field $\overrightarrow{E}$ is the solution of the linear system:

$$\widehat{D} \cdot \overrightarrow{E} = \overrightarrow{0}, \tag{1.4}$$

$$\widehat{D} = D_{\alpha\beta} = \varepsilon_{\alpha\beta} + N_\alpha N_\beta - N^2 \delta_{\alpha\beta}, \tag{1.5}$$

$$\begin{pmatrix} \varepsilon_{xx} - N_\parallel^2 & \varepsilon_{xy} & \varepsilon_{xz} + N_\parallel N_\perp \\ \varepsilon_{yx} & \varepsilon_{yy} - N^2 & \varepsilon_{yz} \\ \varepsilon_{zx} + N_\parallel N_\perp & \varepsilon_{zy} & \varepsilon_{zz} - N_\perp^2 \end{pmatrix} \begin{pmatrix} E_x \\ E_y \\ E_z \end{pmatrix} = 0. \tag{1.6}$$

The condition for the existence of the non-trivial solution of the system (1.6) gives the dispersion relation:

$$D(\overrightarrow{R}, N_\parallel, N_\perp, \omega) = \det D_{\alpha\beta} = 0. \tag{1.7}$$

This dispersion function, calculated for the Hermitian part of the dielectric tensor $\widehat{\varepsilon}$, determines the ray trajectory from the following system of the ODE:

$$\begin{array}{ll} \frac{dr}{dt} = -\frac{c}{\omega}\frac{\partial D/\partial N_r}{\partial D/\partial \omega}, & \frac{dN_r}{dt} = \frac{c}{\omega}\frac{\partial D/\partial r}{\partial D/\partial \omega} \\ \frac{d\varphi}{dt} = -\frac{c}{\omega}\frac{\partial D/\partial M}{\partial D/\partial \omega}, & \frac{dM}{dt} = \frac{c}{\omega}\frac{\partial D/\partial \varphi}{\partial D/\partial \omega} \\ \frac{dz}{dt} = -\frac{c}{\omega}\frac{\partial D/\partial N_z}{\partial D/\partial \omega}, & \frac{dN_z}{dt} = \frac{c}{\omega}\frac{\partial D/\partial z}{\partial D/\partial \omega}. \end{array} \tag{1.8}$$

1.3 Normalized variables and geometrical optics equations

In the code we use normalized variables. We will introduce the following characteristic parameters:

$$r_0,\ t_0 = r_0/c,\ \omega_0 = 2\pi f_0. \tag{1.9}$$

The normalized variables take the form:

$$\widehat{t} = t/t_0, \tag{1.10}$$

$$\widehat{R} = (\widehat{z} = z/r_0, \widehat{r} = r/r_0, \varphi), \tag{1.11}$$

$$\overrightarrow{N} = (N_z, N_r, \widehat{M} = M/r_0 = \widehat{r} N_\varphi), \tag{1.12}$$

$$\widehat{\omega} = \omega/\omega_0. \tag{1.13}$$

We use the following equations, taking the length $r_0 = 1\ m$:

$$\begin{array}{ll} \frac{d\widehat{r}r_0}{d\widehat{t}t_0} = -\frac{c}{\widehat{\omega}}\frac{\partial D/\partial N_r}{\partial D/\partial\widehat{\omega}}, & \frac{dN_r}{d\widehat{t}t_0} = \frac{c}{\widehat{\omega}}\frac{\partial D/\partial\widehat{r}r_0}{\partial D/\partial\widehat{\omega}} \\ \frac{d\widehat{\varphi}}{d\widehat{t}t_0} = -\frac{c}{\widehat{\omega}}\frac{\partial D/\partial Mr_0}{\partial D/\partial\widehat{\omega}}, & \frac{d\widehat{M}r_0}{d\widehat{t}t_0} = \frac{c}{\widehat{\omega}}\frac{\partial D/\partial\varphi}{\partial D/\partial\widehat{\omega}} \\ \frac{d\widehat{z}r_0}{d\widehat{t}t_0} = -\frac{c}{\widehat{\omega}}\frac{\partial D/\partial N_z}{\partial D/\partial\widehat{\omega}}, & \frac{dN_z}{d\widehat{t}t_0} = \frac{c}{\widehat{\omega}}\frac{\partial D/\partial\widehat{z}r_0}{\partial D/\partial\widehat{\omega}}. \end{array} \tag{1.14}$$

Thus, we have this system of he ray-tracing equations in normalized variables:

$$\begin{array}{ll} \frac{d\widehat{r}}{dt} = -\frac{1}{\widehat{\omega}}\frac{\partial D/\partial Nr}{\partial D/\partial\widehat{\omega}}, & \frac{dN_r}{dt} = \frac{1}{\widehat{\omega}}\frac{\partial D/\partial\widehat{r}}{\partial D/\partial\widehat{\omega}} \\ \frac{d\varphi}{d\widehat{t}} = -\frac{1}{\widehat{\omega}}\frac{\partial D/\partial\widehat{M}}{\partial D/\partial\widehat{\omega}}, & \frac{d\widehat{M}}{d\widehat{t}} = \frac{1}{\widehat{\omega}}\frac{\partial D/\partial\varphi}{\partial D/\partial\widehat{\omega}}. \\ \frac{d\widehat{z}}{d\widehat{t}} = -\frac{1}{\widehat{\omega}}\frac{\partial D/\partial N_z}{\partial D/\partial\widehat{\omega}}, & \frac{dN_z}{d\widehat{t}} = \frac{1}{\widehat{\omega}}\frac{\partial D/\partial\widehat{z}}{\partial D/\partial\widehat{\omega}} \end{array} \tag{1.15}$$

In these variables, the group velocity $\overrightarrow{V}_{gr} = (\frac{dr}{dt}, r\frac{d\varphi}{dt}, \frac{dz}{dt})$ is normalized by the speed of light:

$$\widehat{\overrightarrow{V}_{gr}} = (\frac{d\widehat{r}}{d\widehat{t}}, \widehat{r}\frac{d\varphi}{d\widehat{t}}, \frac{d\widehat{z}}{d\widehat{t}}) = \overrightarrow{V}_{gr}/c. \tag{1.16}$$

In the normalized form the right-hand side of the ray-tracing equations and the values of the variables may calculated with values near the unity. This may useful as a means to reduce numerical error due to finite computer word length.

Chapter 2

Magnetic field

2.1 Axisymmetric magnetic field

An axisymmetric magnetic field is specified using data in the input file **equilib.dat.** This file has an **eqdsk** file format {Lang Lao]. The specified magnetic field is of the form:

$$\overrightarrow{B} = \overrightarrow{e}_{\varphi} B_{\varphi} + \overrightarrow{B}_{p}. \tag{2.1}$$

The toroidal magnetic field is given in **equilib.dat** through the function f,

$$B_{\varphi} = \frac{1}{r} f(\psi), \tag{2.2}$$

and the poloidal magnetic field the a tabulation of the the poloidal flux function ψ,

$$\overrightarrow{B}_{p} = \frac{\overrightarrow{e}_{r}}{r}\frac{\partial \psi}{\partial z} - \frac{\overrightarrow{e}_{z}}{r}\frac{\partial \psi}{\partial r}. \tag{2.3}$$

$(\overrightarrow{e}_{r}, \overrightarrow{,e}_{\varphi}, \overrightarrow{e}_{z})$ are unit vectors of the configuration space cylindrical coordinate system.

2.2 Variables in the equilib.dat file

The **equilib.dat** file contains the array $peqd(i,j) = \psi_{ij}$ for the poloidal flux function given at the points on an equi-spaced rectangular mesh $r = x$, $z = y$,

$$(x_i, y_j),\ i = 1, ...nxeqd, j = 1, ...nyeqd. \tag{2.4}$$

It also contains the array $f_j = f(\psi_j)$ given at equi-spaced points of the poloidal flux mesh ψ_j, $j = 1, nxeqd$, values varying form the magnetic axis to the limiting flux surface. The variables in the **equilib.dat** file are given in the following units: lengths (z,r) are in (m); toroidal magnetic field B_{φ} is in (T); poloidal magnetic flux function ψ is in $(Wb = T * m^2)$ weber; toroidal function $f = rB_{\varphi}$ is in $(T * m)$; pressure p in

Pa. The code reads the arrays $peqd(i,j) = \psi_{ij}$ and $f_j = f(\psi_j)$ and calculates spline approximations for the poloidal function $\psi(z,r)$ and the toroidal function $f = f(\psi)$.

2.3 Ripple magnetic field

We take into account the influence of ripple magnetic field $\delta\overrightarrow{B} = (\delta B_r, \delta B_\varphi, \delta B_z)$ only on the value of the total magnetic field $\overrightarrow{B}_{total} = \overrightarrow{B} + \delta\overrightarrow{B}$; we use the approximation that the flux surfaces remain axisymmetric, for purposes of specifying plasma density, Z_{eff}, and temperature, and the flux surfaces are calculated using axisymmetric data from **equilib.dat** file. The stationary vacuum ripple magnetic field must have the potential F, $\delta\overrightarrow{B} = \nabla F$. The ripple field potential can be found in the form:

$$F = \delta_{ripple}\sin(N_{loop}\varphi)g(r,z). \tag{2.5}$$

This potential gives the toroidal and poloidal ripple fields:

$$B_r = \frac{\partial F}{\partial r},\ B_\varphi = \frac{1}{r}\frac{\partial F}{\partial \varphi},\ B_z = \frac{\partial F}{\partial z}. \tag{2.6}$$

We use this model form for the toroidal ripple field:

$$F = \delta_{ripple}\sin(N_{loop}\varphi)\frac{b_0R_0}{N_{loop}}(\frac{r}{R_{max}})^{N_{loop}}. \tag{2.7}$$

This formula is suitable for approximation of the ripple field in DIII-D. The corresponding function $g(r,z)$ has the form:

$$g(r) = (b_0R_0/N_{loop})(r/R_{max})^{N_{loop}}. \tag{2.8}$$

Here, N_{loop} is the number of toroidal field coils; b_0 is the value of the toroidal magnetic field at the magnetic axis; R_0 is the magnetic axis major radius; R_{max} is the parameter which is taken equal to the major radius at the outer plasma edge; δ_{ripple} gives the relative amplitude of the ripple field.

Alternatively, the code uses another form of the ripple potential:

$$g(r,z) = (b_0R_0/N_{loop})I_0(N_{loop}\rho(r,z)/I_0(N_{loop}). \tag{2.9}$$

Here, I_0 is the modified Bessel function; $\rho(z,r)$ is the small radius; δ_{ripple} is the amplitude of the ripple field at the last flux surface at small radius $\rho = 1$.

The input data **nloop=**N_{loop} and **deltripl=**δ_{ripple} and the switch **i_ripple** are set in the **genray.dat** file.

The given switch chooses the model for the ripple field:

if **i_ripple**=1 **c**ode uses the formula 2.8

if i_ripple=2 code uses the formula 2.9

2.4 General radial coordinates

Any positive monotonic function on the poloidal flux can be used as a radial coordinate. The switch **indexrho** in the **genray.dat** file identifies the choice of four normalized radial coordinates implemented in the code:

indexrho=1

$$\rho_s(\psi) = \sqrt{S(\psi)/S(\psi_{lim})}. \tag{2.10}$$

indexrho=2

$$\rho_{\Phi_t}(\psi) = \sqrt{\Phi_{tor}(\psi)/\Phi_{tor}(\psi_{lim})}. \tag{2.11}$$

indexrho=3

$$\rho_{V_t}(\psi) = \sqrt{V_{tor}(\psi)/V_{tor}(\psi_{lim})}. \tag{2.12}$$

indexrho=4

$$\rho_{\psi^{1/2}}(\psi) = \sqrt{|(\psi - \psi_{mag})/(\psi_{lim} - \psi_{mag})|}. \tag{2.13}$$

indexrho=5

$$\rho_{\psi}(\psi) = (\psi - \psi_{mag})/(\psi_{lim} - \psi_{mag}) \tag{2.14}$$

Here, $S(\psi)$ is the area of the poloidal cross-section; $V_{tor}(\psi)$ is torus volume within the poloidal cross-section $\psi = const$; $\Phi_{tor}(\psi)$ is the toroidal flux inside the surface $\psi = const$; ψ_{mag} is the poloidal flux at the magnetic axis and ψ_{lim} is the poloidal flux at the plasma edge. To find $S(\psi)$, $V_{tor}(\psi)$, and $\Phi_{tor}(\psi)$, the code uses poloidal and toroidal fields calculated using $\psi(r,z)$ and $f(\psi)$.

Chapter 3

Plasma components

3.1 Plasma species

The plasma consists of the number *nbulk* plasma species. The s^{th} plasma species ($s = 1$ for electrons, $s = 2, 3, .., nbulk$ for ion species) has density n_s, charge $Z_s > 0$, and mass m_s. We use the following notation:

$$X_s \equiv \frac{\omega_{ps}^2}{\omega^2}, \quad Y_s \equiv \frac{\omega_{cs}}{\omega}. \tag{3.1}$$

Here,

$\omega_{ps}^2 \equiv \frac{4\pi n_s Z_s^2 e^2}{m_s}$ is the plasma frequency ;

$\omega_{cs} \equiv \frac{Z_s eB}{m_s c}$ is the gyro-frequency ;

$e > 0$ is the electron charge;

$Z_{eff} \equiv \sum_{i=2}^{i=nbulk} Z_i^2 n_i / n_e$ is the effective charge.

3.2 Density, temperature and Z_{eff} profiles

3.2.1 Axisymmetric radial profiles

In the axisymmetric case, there are two ways to input density, temperature and Z_{eff} profiles as a function of the small radius.

Quasi-parabolic profiles

$$n_s(\rho) = (n_{0_s} - n_{b_s})(1 - \rho^{k1_{n_s}})^{k2_{n_s}} + n_{b_s}, \tag{3.2}$$

$$T_s(\rho) = (T_{0_s} - T_{b_s})(1-\rho^{k1_{T_s}})^{k2_{T_s}} + T_{b_s}, \tag{3.3}$$

$$Z_{eff}(\rho) = (Z_{eff_0} - Z_{eff_b})(1-\rho^{k1_z})^{k2_z} + Z_{eff_b}. \tag{3.4}$$

Profile arrays

One dimensional arrays for the density $n_s(\rho_k)$, temperature $T_s(\rho_k)$ and $Z_{eff}(\rho_k)$ profiles are specified on a radial mesh ρ_k. k=1,...ndens. A spline approximation of this data is used to create the radial profiles as a function of the radius.

3.2.2 Non axisymmetric density variations

To add plasma density δn fluctuations we use the following form for the full plasma density:

$$n(z,r,\varphi) = n(\rho)(1+\delta n_0 \beta(\rho,\theta,\varphi)). \tag{3.5}$$

The function β is assumed to be of the form:

$$\beta(\rho,\theta,\varphi) = \beta_\rho(\rho)\beta_\theta(\theta)\cos(l_\varphi\varphi), \tag{3.6}$$

$$\beta_\theta(\theta) = 0.5(1+\cos(l_\theta\vartheta)), \tag{3.7}$$

$$\beta_\rho(\rho) = 0.5*(1+\frac{2}{\pi}\arctan(\frac{(\rho-\rho_0)}{\sigma_n^2}))\frac{\rho-\rho_0}{1-_{\rho_0}}. \tag{3.8}$$

Here

δn_0 is the the relative amplitude of the density fluctuations,

$\beta_\rho(\rho)$ sets the radial form of the fluctuation, For the small values of σ_n it is the smooth differentiable approximation of the following function $G(\rho,\rho_0)$.

$$\beta_\rho(\rho) \approx G(\rho,\rho_0) = \begin{cases} 0, 0 \le \rho < \rho_0 \\ (\rho-\rho_0)/(1-\rho_0), \rho_0 < \rho \le 1 \end{cases}$$

The function $\beta_\theta(\theta)$ gives the poloidal form of the density fluctuations. The input data for the density fluctuations are given in **genray.dat** file: **var0=**δn_0, **sigman=**σ_n**, an=**ρ_0**, denm=**l_θ, **denn=**l_φ

3.2.3 Plasma neutrality and plasma charge Z_{eff}.

The ion charges Z_s and densities $n_s(\rho)$ should satisfy the plasma neutrality and generate the given plasma charge $Z_{eff}(\rho)$. Thus these input data are dependent upon one other.

There are several possibilities in GENRAY for the generation of electron and ion density radial profiles and plasma plasma charge. The variable **izeff** in the **GENRAY.dat** file specifies these possibilities:

1) izeff=0

Input data from **genray.dat** file:
nbulk is the number of plasma species;
$charge(s) = Z_s > 0$, $s = 2,...,,nbulk$ are the ion charges;
$n_s(\rho)$, $s = 2,...,,nbulk$ are the ion density radial profiles**;**
$T_s(\rho)$, $s = 1,...,,nbulk$ are the electron and ion temperature radial profiles**.**

GENRAY calculates electron density profile $n_e(\rho_k)$ and effective charge profile $Z_{eff} = Z_{eff}(\rho_k)$ at the points of the radial mesh $\rho_k = (k-1)/(ndens-1), k = 1,...,ndens$. The parameter *ndens* (given in param.i file) is the number of points of the radial ρ mesh. Using the given ion charges Z_s and the ion density profiles $n_s(\rho)$ (analytic or tabular), we have:

$$n_e(\rho_k) = n_{s=1}(\rho_k) = \sum_{s=2}^{s=nbulk} Z_s n_s(\rho_k) \tag{3.9}$$

$$Z_{eff}(\rho_k) = \sum_{s=2}^{s=nbulk} Z_s^2 n_s(\rho_k)/n_1(\rho_k). \tag{3.10}$$

Then, for general radius ρ GENRAY calculates the functions $n_{e=}n_1(\rho)$ and $Z_{eff}(\rho)$ as spline approximations of the data on the given radial mesh ρ_k. k=1,...ndens.

2) izeff=1

Input data from **genray.dat** file:

nbulk is the number of the plasma species, *nbulk*≥ 3;
$charge(s) = Z_s > 0$, $s = 2,...,,nbulk$ are the ion charges;
$n_s(\rho)$, $s = 1,...,,nbulk-2$ are the electron and ion density radial profiles;
$Z_{eff}(\rho)$ is the plasma effective charge;
$T_s(\rho)$, $s = 1,...,,nbulk$ are the electron and ion temperature radial profiles**.**

In this case GENRAY calculates the ion density profiles $n_{nbulk-1}$ and n_{nbulk} using given $Z_{eff}(\rho)$, ion charges Z_s and given electron $n_1(\rho)$ and ion $n_s(\rho)$, $s = 2,...,(nbulk-2)$ density profiles from the following system:

$$Z_{eff}(\rho) = \sum_{i=2}^{i=nbulk-2} Z_i^2 \widehat{n}_i(\rho) + Z_{nbulk-1}^2 \widehat{n}_{nbulk-1}(\rho) + Z_{nbulk}^2 \widehat{n}_{nbulk}(\rho), \tag{3.11}$$

$$1 = \sum_{i=2}^{i=nbulk-2} Z_i \widehat{n}_i(\rho) + Z_{nbulk-1} \widehat{n}_{nbulk-1}(\rho) + Z_{nbulk} \widehat{n}_{nbulk}(\rho). \tag{3.12}$$

Here, $\widehat{n}_i(\rho) = n_i(\rho)/n_e(\rho)$ is the ratio of ion density ($i = 2,...,nbulk$) to electron density.
We see (from3.11, 3.12) that this case (**izeff=1**) does not make sense if $nbulk \leq 2$. The nbulk= 3 case is a standard one in which only the electron density and effective charge profiles $n_s(\rho)$, $Z_{eff}(\rho)$ and 2 ionic charges, Z_s, $s = 2,3$, need be given.

For $nbulk \geq 3$, then consider the system of equations (3.11) and (3.12).

a) If $Z_{nbulk} \neq Z_{nbulk-1}$, then from equations (3.11) and (3.12) we have:

$$\widehat{n}_{nbulk} = \frac{Z_{eff} - Z_{nbulk-1} + \sum_{i=2}^{i=nbulk-2} Z_i \widehat{n}_i (Z_{nbulk-1} - Z_i)}{Z_{nbulk}(Z_{nbulk} - Z_{nbulk-1})}, \tag{3.13}$$

and

$$\widehat{n}_{nbulk-1} = \frac{Z_{nbulk} - Z_{eff} - \sum_{i=2}^{i=nbulk-2} Z_i \widehat{n}_i (Z_{nbulk} - Z_i)}{Z_{nbulk-1}(Z_{nbulk} - Z_{nbulk-1})}. \tag{3.14}$$

If $nbulk = 3$, then the summations $\sum_{i=2}^{i=nbulk-2}$ in 3.21 and 3.22are equal to zero. If the equations 3.21, 3.22 give negative values $\widehat{n}_{nbulk}$ or $\widehat{n}_{nbulk-1}$, GENRAY generates a warning and stops.

b) $Z_{nbulk} = Z_{nbulk-1}$, then equations (3.11) and (3.12) could not determine $\widehat{n}_{nbulk}$ or $\widehat{n}_{nbulk-1}$, and code will give the warning and stop.

3) izeff=2

Input data from the **genray.dat** file is the following:

nbulk is the number of plasma species;
$charge(s) = Z_s > 0$, $s = 2, ..., , nbulk$ are the ion charges;
$n_s(\rho)$, $s = 1, ..., , nbulk$ are the electron and ion density radial profiles;
$Z_{eff}(\rho)$ is the plasma charge.

In this case the input $Z_{eff}(\rho)$ does not coincide with the effective plasma charge calculated by formula (3.10). This case can be useful when we use the electron Mazzucato dispersion but Z_{eff} will be necessary for the calculations of current drive efficiency.

4) izeff=3

The input data from **genray.dat** file is the following:

nbulk is the number of plasma species;
$charge(s) = Z_s > 0$ $s = 2, ..., nbulk$ are the ion charges;
$n_s(\rho)$, $s = 2, ..., , nbulk$ are the ion density radial profiles.

In this case we will use the **eqdsk** pressure $p(\rho)$.

It is assumed that the ion temperatures are equal to the electron temperature $T_e(\rho) = T_i(\rho)$**.** In this case GENRAY calculates the electron density $n_1(\rho)$ from the charge neutrality condition (3.9) , the plasma charge $Z_{eff}(\rho)$ from (3.10), and $T_e(\rho) = T_i(\rho)$ from (3.17). The plasma pressure is determined by the formula:

$$p = n_1 T_1 + \sum_{i=2}^{nbulk} n_i T_i. \tag{3.15}$$

Here, T_1 is electron temperature , T_i $(i = 2, nbulk)$ is ion temperature. Let all ion temperature be equal to electron temperature:

$$T_i = T_e,\ i = 2, nbulk. \tag{3.16}$$

From (3.16) we have:

$$T_1 = p/(n_1 + \sum_{i=2}^{nbulk} n_i). \tag{3.17}$$

We use the ion density radial profiles from input **equilib.dat** file $n_i(\rho),\ i = 2,...,nbulk$. $n_1(\rho)$ and $Z_{eff}(\rho)$ are calculated as stated above. The code calculates density n in $10^{19}m^{-3}$, pressure p in Pa and temperature T in KeV. To get the temperature in KeV, we change Eq. 3.17 to the following form:

$$T_1 = \frac{p}{1.6 \cdot 10^3 (n_1 + \sum_{i=2}^{nbulk} n_i)}. \tag{3.18}$$

For this case the limit of the number of the plasma species is $nbulk \geq 2$.

5) izeff=4

The input data from **genray.dat** file is the following:

$nbulk$ is the number of plasma species; cases for $nbulk \geq 1$ are enabled;
$charge(s) = Z_s > 0$, $s = 2,...,,nbulk$ are the ion charges;
$n_s(\rho)$, $s = 2,...,,nbulk-2$ are the ion density radial profiles;
$Z_{eff}(\rho)$ is the plasma charge;
$T_s(\rho)$, $s = 1,...,,nbulk$ are the electron and ion temperature radial profiles.

In this case we will use the **eqdsk** pressure $p(\rho)$.

In that case GENRAY calculates the electron $n_1(\rho)$ and two ion density radial profiles $n_{nbulk-1}(\rho)$ and $n_{nbulk}(\rho)$ using the given $Z_{eff}(\rho)$, ion charges Z_s , pressure $p(\rho)$ and the given ion $n_s(\rho)$, $s = 2,...,(nbulk-2)$ density profiles and temperature profiles $T_s(\rho)$, $s = 1,...,nbulk$. We introduce the normalized ion temperatures $\widehat{T_i} = T_i/T_1$. Eq. (3.23) gives:

$$p/T_1 = n_1 + \sum_{i=2}^{i=nbulk} n_i \widehat{T_i}. \tag{3.19}$$

For T in Kev , p in Pa and n in $10^{19}m^{-3}$, the formula (3.19) has the form:

$$p/T_1 = 1.6 \cdot 10^3 (n_1 + \sum_{i=2}^{i=nbulk} n_i \widehat{T_i}). \tag{3.20}$$

If $nbulk = 1$:
In this case GENRAY sets $Z_{eff}(\rho) = 1$ and $n_1(\rho) = (p(\rho)/2)/T_1(\rho)/(1.6 \cdot 10^3)$, accounting for ion pressure equal to electron pressure.

If $nbulk = 2$:

In this case GENRAY sets $Z_{eff}(\rho) = Z_2$, $n_2(\rho) = n_1(\rho)/Z_2$, using the plasma neutrality, and gives $n_1(\rho) = p(\rho)/T_1(\rho)/(1.6 \cdot 10^3)/(1 + \widehat{T}_2/Z_2)$, using3.20.

If $nbulk \geq 3$

a) If $Z_{nbulk} \neq Z_{nbulk-1}$

From equations (3.21), (3.22) and (3.23) we have:

$$\widehat{n}_{nbulk} = \frac{Z_{eff} - Z_{nbulk-1} + \sum_{i=2}^{i=nbulk-2} Z_i \widehat{n}_i (Z_{nbulk-1} - Z_i)}{Z_{nbulk}(Z_{nbulk} - Z_{nbulk-1})} \tag{3.21}$$

$$\widehat{n}_{nbulk-1} = \frac{Z_{nbulk} - Z_{eff} - \sum_{i=2}^{i=nbulk-2} Z_i \widehat{n}_i (Z_{nbulk} - Z_i)}{Z_{nbulk-1}(Z_{nbulk} - Z_{nbulk-1})} \tag{3.22}$$

$$p = n_1 T_1 + \sum_{i=2}^{nbulk} n_i T_i \tag{3.23}$$

$$\begin{gathered} p = n_1 T_1 + n_1 \sum_{i=2}^{nbulk-2} \widehat{n}_i T_i + T_{nbulk-1} n_1 \frac{Z_{nbulk} - Z_{eff} - \sum_{i=2}^{i=nbulk-2} Z_i \widehat{n}_i (Z_{nbulk} - Z_i)}{Z_{nbulk-1}(Z_{nbulk} - Z_{nbulk-1})} + \\ T_{nbulk} n_1 \frac{Z_{eff} - Z_{nbulk-1} + \sum_{i=2}^{i=nbulk-2} Z_i \widehat{n}_i (Z_{nbulk-1} - Z_i)}{Z_{nbulk}(Z_{nbulk} - Z_{nbulk-1})} \end{gathered} \tag{3.24}$$

$$\begin{gathered} n_1 (T_1 + T_{nbulk-1} \frac{Z_{nbulk} - Z_{eff}}{Z_{nbulk-1}(Z_{nbilk} - Z_{nbulk-1})} + T_{nulk} \frac{Z_{eff} - Z_{nbulk-1}}{Z_{nbulk}(Z_{nbulk} - Z_{nbulk-1})}) = p - \\ - sum3 + T_{nbulk-1} \frac{sum1}{Z_{nbulk-1}(Z_{nbulk} - Z_{nbulk-1})} - T_{nbulk} \frac{sum2}{Z_{nbulk}(Z_{nbulk} - Z_{nbulk-1})} \quad . \end{gathered} \tag{3.25}$$

We use the following definitions:

$$sum1 = \sum_{i=2}^{i=nbulk-2} Z_i n_i (Z_{nbulk} - Z_i), \tag{3.26}$$

$$sum2 = \sum_{i=2}^{i=nbulk-2} Z_i n_i (Z_{nbulk-1} - Z_i), \tag{3.27}$$

$$sum3 = \sum_{i=2}^{i=nbulk-2} n_i T_i. \tag{3.28}$$

If $nbulk = 3$ these sums are equal to zero.

Eq. (3.25) gives the electron density:

$$n_1 = \frac{p - sum3 + T_{nbulk-1} \frac{sum1}{Z_{nbulk-1}(Z_{nbulk} - Z_{nbulk-1})} - T_{nbulk} \frac{sum2}{Z_{nbulk}(Z_{nbulk} - Z_{nbulk-1})}}{T_1 + T_{nbulk-1} \frac{Z_{nbulk} - Z_{eff}}{Z_{nbulk-1}(Z_{nbulk} - Z_{nbulk})} + T_{nbulk} \frac{Z_{eff} - Z_{nbulk-1}}{Z_{nbulk}(Z_{nbulk} - Z_{nbulk-1})}}. \tag{3.29}$$

With density in $10^{19} m^{-3}$, pressure in Pa, and temperature in KeV, then the formula (3.29) has the following form:

$$n_1 = \frac{p/(1.6\cdot 10^3) - sum3 + T_{nbulk-1}\frac{sum1}{Z_{nbulk-1}(Z_{nbulk}-Z_{nbulk-1})} - T_{nbulk}\frac{sum2}{Z_{nbulk}(Z_{nbulk}-Z_{nbulk-1})}}{T_1 + T_{nbulk-1}\frac{Z_{nbulk}-Z_{eff}}{Z_{nbulk-1}(Z_{nbulk}-Z_{nbulk})} + T_{nbulk}\frac{Z_{eff}-Z_{nbulk-1}}{Z_{nbulk}(Z_{nbulk}-Z_{nbulk-1})}}. \tag{3.30}$$

GENRAY calculates electron density $n_1(\rho)$ using (3.30) and ion densities $n_{bulk}(\rho)$, $n_{bulk-1}(\rho)$ using (3.21), (3.22).

b) If $Z_{nbulk} = Z_{nbulk-1}$

In this case the plasma charge cannot be arbitrary. It must be equal to the value determined by equation 3.10. GENRAY will stop the calculation and will give a warning in this case.

Chapter 4

Dispersion relations

GENRAY uses several different dispersion function models:

1) Magnetized cold plasma with electrons and an arbitrary number of ions.

2) Magnetized hot non-relativistic plasma with electrons and an arbitrary number of ions. The distribution of each species is a shifted, two-temperature Maxwellian.

3)The Mazzucato relativistic approximation of the dielectric tensor for magnetized relativistic electron plasma.

4) The Shkarofsky relativistic approximation of the dielectric tensor for magnetized relativistic electron plasma.

5) The dispersion relations for non-Hermitian dielectric tensors proposed in [2]. This form of the dispersion function is realized for Mazzucato electron relativistic tensor and for hot non-relativistic plasma

The switch **id** specifies which dispersion relation model is to be used. (Ref.4)

4.1 Magnetized cold plasma

For magnetized cold plasma, we use the following form of the dielectric tensor $\varepsilon_{\alpha\beta}$ (Ref.3) (N.A.Krall and A.W.Trivelpiece, 1973) [4]:

$$\varepsilon_{cold_{\alpha\beta}} = \begin{bmatrix} \varepsilon_\perp & ig & \\ -ig & \varepsilon_\perp & \\ & & \varepsilon_\parallel \end{bmatrix}, \tag{4.1}$$

$\varepsilon_{xx} = \varepsilon_{yy} = \varepsilon_\perp$, $\varepsilon_{xy} = ig_\perp$, $\varepsilon_{yx} = -ig_\perp$, $\varepsilon_{zz} = \varepsilon_\parallel$.

$$\varepsilon_\perp = 1 - \sum_{s=1}^{s=nbulk} \frac{\omega_{ps}^2}{\omega^2 - \omega_s^2}, \tag{4.2}$$

$$g = -\frac{\omega_{pe}^2 \omega_{ce}}{\omega(\omega^2 - \omega_{ce}^2)} + \sum_{i=2}^{i=nbulk} \frac{\omega_{pi}^2 \omega_{ci}}{\omega(\omega^2 - \omega_{ci}^2)}, \tag{4.3}$$

$$\varepsilon_\parallel = 1 - \sum_{s=1}^{s=nbulk} \frac{\omega_{ps}^2}{\omega^2}. \tag{4.4}$$

Here, electron ω_{ce} and ion gyro-frequencies ω_{ci} have the same sign. We can rewrite the tensor components using (3.1):

$$\varepsilon_\perp = 1 - \sum_{s=1}^{s=nbulk} \frac{X_s}{1-Y_s^2}, \tag{4.5}$$

$$g = -\frac{X_e Y_e}{1-Y_e^2} + \sum_{i=2}^{i=nbulk} \frac{X_i Y_i}{1-Y_i^2}, \tag{4.6}$$

$$\varepsilon_\parallel = 1 - \sum_{s=1}^{s=nbulk} X_s. \tag{4.7}$$

Here electron Y and ion gyro-frequencies Y_i have the same sign. We will use θ to be the angle between the magnetic field $\overrightarrow{B}$ and the refractive index $\overrightarrow{N}$, $\cos\theta = (\overrightarrow{B}\cdot\overrightarrow{N})/BN$. For the given tensor, the dispersion relation (1.5) can be written in the following form:

id=1

$$D_{cold} = AN^4 + BN^2 + C = 0. \tag{4.8}$$

Here,

$$A = \varepsilon_1 \sin^2\theta + \varepsilon_3 \cos^2\theta, \tag{4.9}$$

$$B = -\varepsilon_1\varepsilon_3(1+\cos^2\theta) - (\varepsilon_1^2 - \varepsilon_2^2)\sin^2\theta, \tag{4.10}$$

$$C = \varepsilon_3(\varepsilon_1^2 - \varepsilon_2^2), \tag{4.11}$$

$\varepsilon_1 = \varepsilon_\perp$, $\varepsilon_2 = g$, $\varepsilon_3 = \varepsilon_\parallel$

The given tensor has singularity at the points of $Y_s^2 = 1$. To avoid these singularities, we multiply all tensor elements by $\Delta = (1 - Y_\beta)$. We can choose plasma designator β to avoid singularity along the trajectory. For example, if the ray can reach the electron-resonance point along the trajectory, we choose $\beta = 1$ to avoid singularity at this point.

For some cases, it is convenient to rewrite the dispersion function so that it describes only one given wave mode:

id=2

$$D_{cold} = N^2 - \frac{-B + ioxm\sqrt{B^2 - 4AC}}{2A} = 0. \tag{4.12}$$

Here, $ioxm = \pm 1$ determines the type of wave mode to be used.

4.2 Magnetized hot non-relativistic plasma (id=6)

4.2.1 General formula

For non-relativistic hot plasma, GENRAY contains hot dielectric tensor in the form (T.H.Stix, 1992) [3]:

$$\varepsilon_{hot_{\alpha\beta}}(\omega, \overrightarrow{k}) = \delta_{\alpha\beta} + \sum_s \chi_{s_{\alpha\beta}}(\omega, \overrightarrow{k}). \tag{4.13}$$

Here, the summation is the plasma species and susceptibilities s, $\chi_{s_{\alpha\beta}}(\omega, \overrightarrow{k})$. For a shifted Maxwellian distribution with different temperatures $T_{s_\parallel}$ parallel and $T_{s_\perp}$ perpendicular to magnetic field and the drifting velocity V_s(parallel to the magnetic field), the susceptibilities have the form:

$$\widehat{\chi}_s = \widehat{e}_\parallel \widehat{e}_\parallel \frac{2\omega_{ps}^2}{\omega k_\parallel w_{s_\perp}^2} V_s + \frac{\omega_{ps}^2}{\omega} \sum_{n=-\infty}^{n=\infty} \exp(-\lambda_s) \widehat{Y}_n(\lambda_s), \tag{4.14}$$

$$\widehat{Y}_n(\lambda) = \begin{bmatrix} \frac{nI_n}{\lambda} A_n & -in(I_n - I_n^{'})A_n & \frac{k_\perp}{\Omega_s}\frac{nI_n}{\lambda} B_n \\ in(I_n - I_n^{'})A_n & (\frac{n^2}{\lambda} I_n + 2\lambda I_n - 2\lambda I_n^{'})A_n & \frac{ik_\perp}{\Omega_s}(I_n - I_n^{'})B_n \\ \frac{k_\perp}{\Omega_s}\frac{nI_n}{\lambda} B_n & -\frac{ik_\perp}{\Omega_s}(I_n - I)_n^{'} B_n & \frac{2(\omega - n\Omega_s)}{k_\parallel w_{s_\perp}^2} I_n B_n \end{bmatrix}. \tag{4.15}$$

Here,

$V_s =< v_\parallel >_s$ is the drift velocity ;

$w_{s_\perp} = \sqrt{\frac{2T_{s_\perp}}{m_s}}$ is the a perpendicular thermal velocity;

$w_{s_\parallel} = \sqrt{\frac{2T_{s_\parallel}}{m_s}}$ is the parallel thermal velocity;

$T_s = (T_{s_\parallel} + 2T_{s_\perp})/3$ is the average thermal velocity;

$t_{pop_s} = \frac{T_{s_\perp}}{T_{s_\parallel}}$ is the ratio of perpendicular and parallel temperatures;

$\lambda_s = \frac{k_\perp^2 w_{s_\perp}^2}{\omega_{cs}^2} = \frac{N_\perp^2 w_{s_\perp}^2}{Y_s^2 c^2} = \frac{N_\perp^2 \beta_s^2 t_{pop_s}}{Y_s^2}$;

$\beta_s = \frac{w_{s_\parallel}}{c}$ is the normalized longitudinal thermal velocity;

Ω_s is the algebraic gyro-frequency; it has different signs for electrons $\Omega_e = -\omega_{ce}$ and for ions $\Omega_i = \omega_{ci}$.

$I_n(\lambda)$ is the modified Bessel function with argument λ and $I_n^{'} = dI_n(\lambda)/d\lambda$.

The coefficients A and B have the form;

$$A_n = \frac{1}{\omega}\frac{T_\perp - T_\parallel}{T_\parallel} + \frac{1}{k_\parallel w_\parallel}\frac{(\omega - k_\parallel V - n\Omega)T_\perp + n\Omega T_\parallel}{\omega T_\parallel} Z_0, \tag{4.16}$$

$$\begin{aligned} B_n = & \frac{1}{k_\parallel}\frac{(\omega - n\Omega)T_\perp - (k_\parallel V - n\Omega)T_\parallel}{\omega T_\parallel} \\ & + \frac{1}{k_\parallel}\frac{\omega - n\Omega}{k_\parallel w_\parallel}\frac{(\omega - k_\parallel V - n\Omega)T_\perp + n\Omega T_\parallel}{\omega T_\parallel} Z_0. \end{aligned} \tag{4.17}$$

$Z_0 = Z(\zeta_n)$ is the plasma dispersion function:

$$Z_0(\zeta_n) = i\sqrt{\pi} sgn(k_\parallel) \exp(-\zeta_n^2) - 2S(\zeta_n), \tag{4.18}$$

$$S(\zeta) \equiv \exp(-\zeta^2) \int_0^\zeta \exp(z^2) dz, \tag{4.19}$$

$$\zeta_n = \frac{\omega - k_\| V - n\Omega}{k_\| w_\|}. \tag{4.20}$$

We transform the formula (4.14) for the susceptibilities to the following:

$$\chi_{s_{\alpha\beta}} = \delta_{\alpha z}\delta_{\beta z}\frac{2X_s}{N_\|\beta_s^2 t_{pop_s}}\frac{V_s}{c} + X_s\sum_{n=-\infty}^{n=\infty}\exp(-\lambda_s)\widehat{\widehat{Y}}_n(\lambda_s), \tag{4.21}$$

$$\widehat{\widehat{Y}}_n = \omega\widehat{Y}_n(\lambda) = \begin{bmatrix} \frac{nI_n}{\lambda}\widetilde{A}_n & -in(I_n - I_n^{'})\widetilde{A}_n & \frac{\omega N_\perp}{\Omega}\frac{nI_n}{\lambda}\widetilde{B}_n \\ in(I_n - I_n^{'})\widetilde{A}_n & (\frac{n^2}{\lambda}I_n + 2\lambda I_n - 2\lambda I_n^{'})\widetilde{A}_n & \frac{i\omega N_\perp}{\Omega}(I_n - I_n^{'})\widetilde{B}_n \\ \frac{\omega N_\perp}{\Omega}\frac{nI_n}{\lambda}\widetilde{B}_n & -\frac{i\omega N_\perp}{\Omega}(I_n - I_n^{'})\widetilde{B}_n & \frac{2(1-n\frac{\Omega}{\omega})}{N_\|\beta^2 t_{pop}}I_n\widetilde{B}_n \end{bmatrix}, \tag{4.22}$$

$$\widetilde{A}_n = \omega A_n = (t_{pop-1}) + \frac{1}{N_\|\beta}((1 - N_\|\frac{V}{c} - n\frac{\Omega}{\omega})t_{pop} + n\frac{\Omega}{\omega})Z_0(\zeta_n), \tag{4.23}$$

$$\begin{aligned}\widetilde{B}_n = \omega B_n/c = \tfrac{1}{N_\|}((1 - n\tfrac{\Omega}{\omega})t_{pop} - (N_\|\tfrac{V}{c} - n\tfrac{\Omega}{\omega})) + \\ + \tfrac{1}{N_\|}\tfrac{(1-n\frac{\Omega}{\omega})}{N_\|\beta}((1 - N_\|\tfrac{\Omega}{\omega})t_{pop} + n\tfrac{\Omega}{\omega})Z_0(\zeta_n)\end{aligned}, \tag{4.24}$$

$$\zeta_n = \frac{1 - N_\|\frac{V}{c} - n\frac{\Omega}{\omega}}{N_\|\beta}. \tag{4.25}$$

It is essential to note that GENRAY uses a positive sign for electron and for ion charges $Z_s > 0$. The relation Y_s (3.1)of gyro-frequencies $\omega_{cs}$to wave frequency ω has the same sign for electrons and for ions. We have chosen this form for the cold plasma dispersion relation (4.3). For the hot plasma of electrons, we use:

$$\frac{\Omega_e}{\omega} = -\frac{\omega_{ce}}{\omega} = -Y_e = -Y_1; \tag{4.26}$$

but for the ions, we use:

$$\frac{\Omega_i}{\omega} = \frac{\omega_{ci}}{\omega} = Y_i,\ i = 2,...,nbulk. \tag{4.27}$$

The dispersion relation for hot non-relativistic plasma has the form:

id=6

$$D = D_{hot}(N_\|, N_\perp, \overrightarrow{X}, \overrightarrow{Y}, \overrightarrow{T}, \overrightarrow{t_{pop}}) = 0. \tag{4.28}$$

Here, $\overrightarrow{X} = X_s$, $\overrightarrow{Y} = Y_s$, $\overrightarrow{T} = T_s$, $\overrightarrow{t_{pop}} = t_{pop_s}$, $s = 1,...,nbulk$.

4.2.2 Computation

Calculation of the dispersion function (4.28) must estimate the number of gyro-harmonics used. We can estimate the number of harmonics using features of the plasma functions $Z(\zeta_n)$ and the product of the modified Bessel functions $I_n(\lambda)$ multiplied by $\exp(-\lambda)$.

For a large value of the argument $|x| > 1$, the asymptotic part of plasma dispersion function $S(x) = Re(Z_0(x))$ (1.5) has the form:

$$S(x) \simeq \frac{1}{2x}. \tag{4.29}$$

The number of necessary gyro-harmonics can be evaluated from the following condition:

$$|S(\zeta_n)| = \frac{1}{2|\zeta_n|} \geq \varepsilon_n. \tag{4.30}$$

Here , ε_n is accuracy. The condition (4.30) gives $|\zeta_n| \leq \frac{1}{2\varepsilon_n}$:

$$|1 - N_\parallel \frac{V}{c} + n\frac{\Omega}{\omega}| \leq \frac{|N_\parallel|\beta}{2\varepsilon_n}. \tag{4.31}$$

Here we introduce,

$$n_{min} = \min((-1 + N_\parallel \frac{V}{c} - \frac{|N_\parallel|\beta}{2\varepsilon_n})/\frac{\Omega}{\omega}), (-1 + N_\parallel \frac{V}{c} + \frac{|N_\parallel|\beta}{2\varepsilon_n})/\frac{\Omega}{\omega})), \tag{4.32}$$

$$n_{max} = \max((-1 + N_\parallel \frac{V}{c} - \frac{|N_\parallel|\beta}{2\varepsilon_n})/\frac{\Omega}{\omega}), (-1 + N_\parallel \frac{V}{c} + \frac{|N_\parallel|\beta}{2\varepsilon_n})/\frac{\Omega}{\omega})). \tag{4.33}$$

The interval of the numbers n of the gyro-harmonics (determined from the plasma dispersion function $Z_0(\zeta_n)$) has the form:

$$n_{min} < n < n_{max}. \tag{4.34}$$

In particular, the number of cyclotron resonance harmonic n_c can be obtained from condition $Z_0(\zeta_{n_c}) = 0$. This gives $\zeta_{n_c} = 0$ and

$$n_c = (-1 + N_\parallel \frac{V}{c})/\frac{\Omega}{\omega}. \tag{4.35}$$

We can see that $n_c = 0.5(n_{min} + n_{max})$. The production $\exp(-\lambda)I_n(\lambda)$ decreases sharply as gyro-harmonics number n increases. This results in the additional condition for the numbers n :

$$\exp(-\lambda)I_n(\lambda) \geq \varepsilon_n. \tag{4.36}$$

The code has parameter *nmax* (inside subroutine harmon_z)that determines the maximum number of modified Bessel functions calculated. Now this parameter is equal to $nmax = 3000$. If $|n_{min}| < nmax$ and $|n_{max}| < nmax$, the code calculates only $n_{Max} = max(|n_{min}|, |n_{max}|)$ modified Bessel functions. In some cases boundaries n_{min} or n_{max} can be larger than the parameter *nmax*. In such situations, the code changes n_{min} or n_{max} to the given parameter *nmax*. In this case, however, condition (4.34) will not be satisfied.

After determination of the interval for gyro-harmonics (n_{min}, n_{max}), the code calculates the productions $\exp(-\lambda)I_n(\lambda)$ from number $n = 0$ until $n = min(n_{max}, n_\star)$. At each n, the code checks the condition (4.36). If for some number $n_\star$ this production is smaller then the given accuracy $\exp(-\lambda)I_{n_\star}(\lambda) < \varepsilon_n$, the code takes this number as the maximum number of harmonics to be used.

4.3 Magnetized electron relativistic plasma in Mazzucato approximation

For relativistic electron plasma, GENRAY uses the approximation of the relativistic dielectric tensor and the dispersion relation proposed and realized as the routines by Mazzucato (E.Mazzucato et al., 1987) [5]. This tensor contains Hermitian and anti-Hermitian parts. It can be used for calculation of propagation and absorption of waves in the electron cyclotron region. The dispersion function for the Hermitian dielectric tensor has the form:

id=4

$$D_{Maz}(\overrightarrow{N}_{\parallel}, \overrightarrow{N}_{\perp}, X_e, Y_e, T_e). \tag{4.37}$$

4.4 Magnetized electron relativistic plasma in Shkarofsky approximation

For the relativistic electron plasma, GENRAY has the opportunity to use approximation of the relativistic dielectric tensor and the dispersion relation proposed and realized as the routines by Shkarofsky. This tensor contains Hermitian and anti-Hermitian parts. It can be used for calculation of propagation and absorption of waves in the electron cyclotron region. The dispersion function for the Hermitian dielectric tensor has the form:

id=7

$$D_{Shk}(\overrightarrow{N}_{\parallel}, \overrightarrow{N}_{\perp}, X_e, Y_e, T_e). \tag{4.38}$$

4.5 Ono dispersion function for fast waves.

For fast waves the code can use the dispersion function in the M.Ono (1995) [17]approximation for the wave frequencies ω in the interval $\Omega_i \ll \omega \ll \omega_{LH}$. To use the Ono dispersion, choose the option **id**=8. This dispersion relation works for the analytical and numerical derivatives.

The high harmonics fast wave (HHFW) dispersion relation was obtained from a nontrivial solution of the determinant of the wave equation tensor for Maxwellian plasmas [3]

$$det \begin{bmatrix} K_{xx} - n_{\parallel}^2 & -iK_{xyc} & K_{xz} + n_{\perp} n_{\parallel} \\ iK_{xyc} & K_{yy} - n_{\parallel}^2 - n_{\perp}^2 & iK_{yz} \\ K_{xz} + n_{\perp} n_{\parallel} & -iK_{yz} & K_{zz} - n_{\perp}^2 \end{bmatrix} = 0 \tag{4.39}$$

It assumes that the wave frequency is high compared to the ion cyclotron frequency, but well bellow the electron cyclotron frequency. Finite Larmor radius (FLR) effects on electrons $\lambda_e \approx (m_e/m_i)\lambda_i \ll$ are neglected. The terms that contribute to the electron damping are the n=0 terms in the K_{yy}, K_{yz}, K_{zy} and K_{zz} elements of the dielectric tensor. For the present case, it is sufficient to keep the lowest order terms in λ_e. If we neglect the ion FLR terms the dielectric tensor are simplified to

$$K_{xxc} = 1 + \frac{X_e}{Y_e^2} - \sum_i \frac{X_i}{1 - Y_i^2} \tag{4.40}$$

$$K_{xyc} = \frac{X_e}{Y_e} + \sum_i \frac{X_i Y_i}{1 - Y_i^2} \tag{4.41}$$

$$K_{yyc} = K_{xxc} + n_\perp^2 \frac{X_e}{Y_e^2} \frac{V_{Te}}{cn_\parallel} Z_0(y_0) = K_{xxc} + n_\perp^2 \delta_m \tag{4.42}$$

$$V_{T_\sigma} = \sqrt{2T_\sigma / m_\sigma}, \sigma = e, i$$

$$\delta_m = \frac{X_e}{Y_e^2} \frac{V_{Te}}{cn_\parallel} Z_0(y_0)$$

$$y_0 = \frac{1}{n_\parallel} \frac{c}{V_{Te}}$$

$$K_{xzc} = -n_\perp n_\parallel \sum_i \frac{X_i V_{Ti}^2}{(1 - Y_i^2)^2 c^2} = n_\perp n_\parallel \delta \tag{4.43}$$

$$\delta = -\sum_i \frac{X_i V_{Ti}^2}{(1 - Y_i^2)^2 c^2}$$

$$K_{yz} = -n_\perp \frac{V_{Te}^2 n_\parallel}{2c^2 Y_e} K_{zze} = -n_\perp \delta_x K_{zze} \tag{4.44}$$

and

$$K_{zz} \cong 1 - \sum_i X_i - \frac{X_e c^2}{n_\parallel^2 V_{Te}^2} \frac{dZ_0(y_0)}{dy_0} \cong -\frac{X_e c^2}{n_\parallel^2 V_{Te}^2} \frac{dZ_0(y_0)}{dy_0} = K_{zze} \tag{4.45}$$

$$\delta_x = \frac{V_{Te}^2 n_\parallel}{2c^2 Y_e}$$

Here Y_e and Y_i have the same sign.
The dispersion relation can be rewritten in the following simple form

$$D = det \begin{bmatrix} K_{xxc} - n_\parallel^2 & -iK_{xyc} & n_\perp n_\parallel (1 + \delta) \\ iK_{xyc} & K_{xxc} - n_\parallel^2 - n_\perp^2 (1 - \delta_m) & -in_\perp \delta_x K_{zz} \\ n_\perp n_\parallel (1 + \delta) & in_\perp \delta_x K_{zz} & K_{zz} - n_\perp^2 \end{bmatrix} = 0 \tag{4.46}$$

The above determinant is convenient for solving for $n_\perp$ for given $n_\parallel$ and ω, since all $K^{'}s$ and $\delta^{\cdot}s$ are independent of $n_\perp$. It has a form

$$D = an_\perp^4 + bn_\perp^2 + c = 0 \tag{4.47}$$

where

$$a = [(n_\parallel^2 - K_{xxc}) - n_\parallel^2(1+\delta)^2](1-\delta_m)$$

$$b = -K_{xy}^2 - \delta_x^2(n_\parallel^2 - K_{xxc})K_{zze}^2 - (n_\parallel^2 - K_{xxc})(1-\delta_m)K_{zze} + \\ +2\delta_x n_\parallel(1+\delta)K_{zze}K_{xyc} + (n_\parallel^2 - K_{xxc})^2 - \\ -n_\parallel^2(1+\delta)^2(n_\parallel^2 - K_{xxc})$$

$$c = [K_{xyc}^2 - (n_\parallel^2 - K_{xxc})^2]K_{zze}$$

The subscripts "c" and "e" denote cold ions and kinetic electrons.
Thermal velocity $V_{T\sigma} = \sqrt{2T_\sigma/m_\sigma}$, $\frac{\partial V_{T\sigma}}{\partial T_\sigma} = \frac{V_{T\sigma}}{2T_\sigma}$.
The simplified dielectric tensor has the form

$$\begin{bmatrix} \varepsilon_{11} = K_{xxc} & \varepsilon_{12} = -iK_{xyc} & \varepsilon_{13} = n_\perp n_\parallel \delta \\ \varepsilon_{21} = iK_{xyc} & \varepsilon_{22} = K_{xxc} + n_\perp^2 \delta_m & \varepsilon_{23} = -in_\perp \delta_x K_{zz} \\ \varepsilon_{31} = n_\perp n_\parallel \delta & \varepsilon_{32} = in_\perp \delta_x K_{zz} & \varepsilon_{33} = K_{zz} \end{bmatrix} \tag{4.48}$$

To calculate the right hand side in the ray-tracing equations we need the derivatives from the dispersion function D.

The derivatives of the dispersion function with respect to space coordinates

$$\frac{\partial D}{\partial r} = \sum_{ij} \frac{\partial D}{\partial \varepsilon_{ij}}\left[\frac{\partial \varepsilon_{ij}}{\partial n_\perp}\frac{\partial n_\perp}{\partial r} + \frac{\partial \varepsilon_{ij}}{\partial n_\parallel}\frac{\partial n_\parallel}{\partial r} + \sum_\sigma\left(\frac{\partial \varepsilon_{ij}}{\partial X_\sigma}\frac{\partial X_\sigma}{\partial r} + \frac{\partial \varepsilon_{ij}}{\partial Y_\sigma}\frac{\partial Y_\sigma}{\partial r} + \frac{\partial \varepsilon_{ij}}{\partial T_\sigma}\frac{\partial T_\sigma}{\partial r}\right)\right] + \\ + \frac{\partial D}{\partial n_\perp}\frac{\partial n_\perp}{\partial r} + \frac{\partial D}{\partial n_\parallel}\frac{\partial n_\parallel}{\partial r} \tag{4.49}$$

$$\frac{\partial D}{\partial z} = \sum_{ij} \frac{\partial D}{\partial \varepsilon_{ij}}\left[\frac{\partial \varepsilon_{ij}}{\partial n_\perp}\frac{\partial n_\perp}{\partial z} + \frac{\partial \varepsilon_{ij}}{\partial n_\parallel}\frac{\partial n_\parallel}{\partial z} + \sum_\sigma\left(\frac{\partial \varepsilon_{ij}}{\partial X_\sigma}\frac{\partial X_\sigma}{\partial z} + \frac{\partial \varepsilon_{ij}}{\partial Y_\sigma}\frac{\partial Y_\sigma}{\partial z} + \frac{\partial \varepsilon_{ij}}{\partial T_\sigma}\frac{\partial T_\sigma}{\partial z}\right)\right] + \\ + \frac{\partial D}{\partial n_\perp}\frac{\partial n_\perp}{\partial z} + \frac{\partial D}{\partial n_\parallel}\frac{\partial n_\parallel}{\partial z} \tag{4.50}$$

$$\frac{\partial D}{\partial \phi} = \sum_{ij} \frac{\partial D}{\partial \varepsilon_{ij}}\left[\frac{\partial \varepsilon_{ij}}{\partial n_\perp}\frac{\partial n_\perp}{\partial \phi} + \frac{\partial \varepsilon_{ij}}{\partial n_\parallel}\frac{\partial n_\parallel}{\partial \phi} + \sum_\sigma\left(\frac{\partial \varepsilon_{ij}}{\partial X_\sigma}\frac{\partial X_\sigma}{\partial \phi} + \frac{\partial \varepsilon_{ij}}{\partial Y_\sigma}\frac{\partial Y_\sigma}{\partial \phi} + \frac{\partial \varepsilon_{ij}}{\partial T_\sigma}\frac{\partial T_\sigma}{\partial \phi}\right)\right] + \\ + \frac{\partial D}{\partial n_\perp}\frac{\partial n_\perp}{\partial \phi} + \frac{\partial D}{\partial n_\parallel}\frac{\partial n_\parallel}{\partial \phi} \tag{4.51}$$

The derivatives of the dispersion function with respect to refractive index coordinates are

$$\frac{\partial D}{\partial n_r} = \sum_{ij} \frac{\partial D}{\partial \varepsilon_{ij}}\left[\frac{\partial \varepsilon_{ij}}{\partial n_\perp}\frac{\partial n_\perp}{\partial n_r} + \frac{\partial \varepsilon_{ij}}{\partial n_\parallel}\frac{\partial n_\parallel}{\partial n_r}\right] + \\ + \frac{\partial D}{\partial n_\perp}\frac{\partial n_\perp}{\partial n_r} + \frac{\partial D}{\partial n_\parallel}\frac{\partial n_\parallel}{\partial n_r} \tag{4.52}$$

$$\begin{aligned}\frac{\partial D}{\partial n_z} = \sum_{ij} \frac{\partial D}{\partial \varepsilon_{ij}} [\frac{\partial \varepsilon_{ij}}{\partial n_\perp} \frac{\partial n_\perp}{\partial n_z} + \frac{\partial \varepsilon_{ij}}{\partial n_\parallel} \frac{\partial n_\parallel}{\partial n_z}] + \\ + \frac{\partial D}{\partial n_\perp} \frac{\partial n_\perp}{\partial n_z} + \frac{\partial D}{\partial n_\parallel} \frac{\partial n_\parallel}{\partial n_z}\end{aligned} \tag{4.53}$$

$$\begin{aligned}\frac{\partial D}{\partial m} = \sum_{ij} \frac{\partial D}{\partial \varepsilon_{ij}} [\frac{\partial \varepsilon_{ij}}{\partial n_\perp} \frac{\partial n_\perp}{\partial n_m} + \frac{\partial \varepsilon_{ij}}{\partial n_\parallel} \frac{\partial n_\parallel}{\partial m}] + \\ + \frac{\partial D}{\partial n_\perp} \frac{\partial n_\perp}{\partial m} + \frac{\partial D}{\partial n_\parallel} \frac{\partial n_\parallel}{\partial m}\end{aligned} \tag{4.54}$$

The derivatives of the dispersion function with respect to frequency are

$$\begin{aligned}\frac{\partial D}{\partial \omega} = \sum_{ij} \frac{\partial D}{\partial \varepsilon_{ij}} [\frac{\partial \varepsilon_{ij}}{\partial n_\perp} \frac{\partial n_\perp}{\partial \omega} + \frac{\partial \varepsilon_{ij}}{\partial n_\parallel} \frac{\partial n_\parallel}{\partial \omega} + \sum_\sigma (\frac{\partial \varepsilon_{ij}}{\partial X_\sigma} \frac{\partial X_\sigma}{\partial \omega} + \frac{\partial \varepsilon_{ij}}{\partial Y_\sigma} \frac{\partial Y_\sigma}{\partial \omega})] + \\ + \frac{\partial D}{\partial n_\perp} \frac{\partial n_\perp}{\partial \omega} + \frac{\partial D}{\partial n_\parallel} \frac{\partial n_\parallel}{\partial \omega}\end{aligned} \tag{4.55}$$

The dielectric tensor elements are functions of the five parameters $\varepsilon_{i,j} = \varepsilon_{i,j}(\overrightarrow{X}, \overrightarrow{Y}, \overrightarrow{T}, n_\perp, n_\parallel)$. The derivatives of the dielectric tensor elements with respect to $X_{e,i}$, $Y_{e,i}$, $T_{e,i}$ and the refractive index components $N_\perp$, $N_\parallel$ have the following form

$$\begin{aligned}\frac{\partial \varepsilon_{11}}{\partial X_e} = \frac{1}{Y_e^2}, \frac{\partial \varepsilon_{11}}{\partial X_i} = -\frac{1}{1-Y_i^2} \\ \frac{\partial \varepsilon_{11}}{\partial Y_e} = -\frac{2X_e}{Y_e^3}, \frac{\partial \varepsilon_{11}}{\partial Y_i} = -\frac{2X_i Y_i}{(1-Y_i^2)^2} \\ \frac{\partial \varepsilon_{11}}{\partial T_e} = 0, \frac{\partial \varepsilon_{11}}{\partial T_i} = 0 \\ \frac{\partial \varepsilon_{11}}{\partial n_\perp} = 0 \\ \frac{\partial \varepsilon_{11}}{\partial n_\parallel} = 0\end{aligned} \tag{4.56}$$

$$\begin{aligned}\frac{\partial \varepsilon_{12}}{\partial X_e} = -i\frac{1}{Y_e}, \frac{\partial \varepsilon_{12}}{\partial X_i} = -i\frac{Y_i}{1-Y_i^2} \\ \frac{\partial \varepsilon_{12}}{\partial Y_e} = i\frac{X_e}{Y_e^2}, \frac{\partial \varepsilon_{12}}{\partial Y_i} = -iX_i\frac{1+Y_i^2}{(1-Y_i^2)^2} \\ \frac{\partial \varepsilon_{12}}{\partial T_e} = 0, \frac{\partial \varepsilon_{12}}{\partial T_i} = 0 \\ \frac{\partial \varepsilon_{12}}{\partial n_\perp} = 0 \\ \frac{\partial \varepsilon_{12}}{\partial n_\parallel} = 0\end{aligned} \tag{4.57}$$

$$\begin{aligned}\frac{\partial \varepsilon_{13}}{\partial X_e} = 0, \frac{\partial \varepsilon_{13}}{\partial X_i} = -n_\perp n_\parallel \frac{V_{Ti}^2}{(1-Y_i^2)^2 c^2} \\ \frac{\partial \varepsilon_{13}}{\partial Y_e} = 0, \frac{\partial \varepsilon_{13}}{\partial Y_i} = -n_\perp n_\parallel \frac{2X_i V_{Ti}^2 2Y_i}{(1-Y_i^2)^3 c^2} \\ \frac{\partial \varepsilon_{13}}{\partial T_e} = 0, \frac{\partial \varepsilon_{13}}{\partial T_i} = -n_\perp n_\parallel \frac{X_i V_{Ti}^2}{(1-Y_i^2)^2 c^2 T_i} \\ \frac{\partial \varepsilon_{13}}{\partial n_\perp} = n_\parallel \delta \\ \frac{\partial \varepsilon_{13}}{\partial n_\parallel} = n_\perp \delta\end{aligned} \tag{4.58}$$

$$\begin{gathered}
\frac{\partial \varepsilon_{22}}{\partial X_e} = \frac{\partial \varepsilon_{11}}{\partial X_e} + n_\perp^2 \frac{1}{Y_e^2}\frac{V_{Te}}{cn_\parallel} Z_0\left(\frac{c}{n_\parallel V_{Te}}\right) = \frac{\partial \varepsilon_{11}}{\partial X_e} + n_\perp^2 \frac{\delta_m}{X_e}, \frac{\partial \varepsilon_{22}}{\partial X_i} = \frac{\partial \varepsilon_{11}}{\partial X_i} \\
\frac{\partial \varepsilon_{22}}{\partial Y_e} = \frac{\partial \varepsilon_{11}}{\partial Y_e} - n_\perp^2 \frac{2X_e}{Y_e^3}\frac{V_{Te}}{cn_\parallel} Z_0\left(\frac{c}{n_\parallel V_{Te}}\right) = \frac{\partial \varepsilon_{22}}{\partial Y_e} - n_\perp^2 \frac{2\delta_m}{Y_e}, \frac{\partial \varepsilon_{22}}{\partial Y_i} = \frac{\partial \varepsilon_{11}}{\partial Y_i} \\
\frac{\partial \varepsilon_{22}}{\partial T_e} = \frac{\partial \varepsilon_{11}}{\partial T_e} + n_\perp^2 \frac{X_e}{Y_e^2}\left(\frac{V_{Te}}{cn_\parallel}\frac{Z_0(y_0)}{2T_e} - \frac{1}{n_\parallel^2}\frac{dZ_0}{dy_0}\frac{1}{2T_e}\right), \frac{\partial \varepsilon_{22}}{\partial T_i} = \frac{\partial \varepsilon_{11}}{\partial T_i} \\
\frac{\partial \varepsilon_{22}}{\partial n_\perp} = \frac{\partial \varepsilon_{11}}{\partial n_\perp} + 2n_\perp \frac{X_e}{Y_e^2}\frac{V_{Te}}{cn_\parallel} Z_0\left(\frac{c}{n_\parallel V_{Te}}\right) = \frac{\partial \varepsilon_{11}}{\partial n_\perp} + 2n_\perp \delta_m \\
\frac{\partial \varepsilon_{22}}{\partial n_\parallel} = \frac{\partial \varepsilon_{11}}{\partial n_\parallel} + n_\perp^2 \frac{X_e}{Y_e^2}\left[-\frac{V_{Te}}{cn_\parallel^2} Z_0 - \frac{V_{Te}}{cn_\parallel}\frac{dZ_0}{dy_0}\frac{c}{n_\parallel^2 V_{Te}}\right] = \frac{\partial \varepsilon_{11}}{\partial n_\parallel} - n_\perp^2\left(\frac{\delta_m}{n_\parallel} + \frac{X_e}{Y_e^2 n_\parallel^3}\frac{dZ_0}{dy_0}\right)
\end{gathered} \tag{4.59}$$

$$\begin{gathered}
\frac{\partial \varepsilon_{33}}{\partial X_e} = -\frac{1}{n_\parallel^2}\frac{c^2}{V_{Te}^2}\frac{dZ_0}{dy_0} = \frac{-K_{zz}}{X_e}, \frac{\partial \varepsilon_{33}}{\partial X_i} = 0 \\
\frac{\partial \varepsilon_{33}}{\partial Y_e} = 0, \frac{\partial \varepsilon_{33}}{\partial Y_i} = 0 \\
\frac{\partial \varepsilon_{33}}{\partial T_e} = -\frac{X_e}{n_\parallel^2}\left(-\frac{c^2}{V_{Te}^2 T_e}\frac{dZ_0(y_0)}{dy_0} - \frac{c^2}{V_{Te}^2}\frac{d^2 Z_0}{dy_0^2}\frac{c}{n_\parallel V_{Te} 2T_e}\right), \frac{\partial \varepsilon_{33}}{\partial T_i} = 0 \\
\frac{\partial \varepsilon_{33}}{\partial n_\perp} = 0 \\
\frac{\partial \varepsilon_{33}}{\partial n_\parallel} = 2\frac{X_e}{n_\parallel^3}\frac{c^2}{V_{Te}^2}\frac{dZ_0}{dy_0} + \frac{X_e c^2}{n_\parallel^2 V_{Te}^2}\frac{d^2 Z_0}{dy_0^2}\frac{c}{n_\parallel^2 V_{Te}}
\end{gathered} \tag{4.60}$$

$$\begin{gathered}
\frac{\partial \varepsilon_{23}}{\partial X_e} = -in_\perp \delta_x \frac{\partial K_{zz}}{\partial X_e} = in_\perp \delta_x \frac{c^2}{n_\parallel^2 V_{Te}^2}\frac{dZ_0}{dy_0} = -in_\perp \delta_x \frac{K_{zz}}{X_e}, \frac{\partial \varepsilon_{23}}{\partial X_i} = 0 \\
\frac{\partial \varepsilon_{23}}{\partial Y_e} = in_\perp \frac{\delta_x}{Y_e} K_{zz}, \frac{\partial \varepsilon_{23}}{\partial Y_i} = 0 \\
\frac{\partial \varepsilon_{23}}{\partial T_e} = -in_\perp\left(\delta_x \frac{1}{T_e} K_{zz} + \delta_x \frac{\partial \varepsilon_{33}}{\partial T_e}\right), \frac{\partial \varepsilon_{23}}{\partial T_i} = 0 \\
\frac{\partial \varepsilon_{23}}{\partial n_\perp} = -i\left(\delta_x K_{zz} + n_\perp \delta_x \frac{\partial \varepsilon_{33}}{\partial n_\perp}\right) \\
\frac{\partial \varepsilon_{23}}{\partial n_\parallel} = -in_\perp\left(\frac{\delta_x}{n_\parallel} K_{zz} + \delta_x \frac{\partial \varepsilon_{33}}{\partial n_\parallel}\right)
\end{gathered} \tag{4.61}$$

$$\begin{gathered}
\frac{\partial \varepsilon_{21}}{\partial X_\sigma} = -\frac{\partial \varepsilon_{12}}{\partial X_\sigma} \\
\frac{\partial \varepsilon_{21}}{\partial Y_\sigma} = -\frac{\partial \varepsilon_{12}}{\partial Y_\sigma} \\
\frac{\partial \varepsilon_{21}}{\partial T_\sigma} = -\frac{\partial \varepsilon_{12}}{\partial T_\sigma} \\
\frac{\partial \varepsilon_{21}}{\partial n_\perp} = -\frac{\partial \varepsilon_{12}}{\partial n_\perp} \\
\frac{\partial \varepsilon_{21}}{\partial n_\parallel} = -\frac{\partial \varepsilon_{12}}{\partial n_\parallel}
\end{gathered} \tag{4.62}$$

$$\begin{aligned}
\frac{\partial \varepsilon_{31}}{\partial X_\sigma} &= \frac{\partial \varepsilon_{13}}{\partial X_\sigma} \\
\frac{\partial \varepsilon_{31}}{\partial Y_\sigma} &= \frac{\partial \varepsilon_{13}}{\partial Y_\sigma} \\
\frac{\partial \varepsilon_{31}}{\partial T_\sigma} &= \frac{\partial \varepsilon_{13}}{\partial T_\sigma} \\
\frac{\partial \varepsilon_{31}}{\partial n_\perp} &= \frac{\partial \varepsilon_{13}}{\partial n_\perp} \\
\frac{\partial \varepsilon_{31}}{\partial n_\parallel} &= \frac{\partial \varepsilon_{13}}{\partial n_\parallel}
\end{aligned} \tag{4.63}$$

$$\begin{aligned}
\frac{\partial \varepsilon_{32}}{\partial X_\sigma} &= -\frac{\partial \varepsilon_{23}}{\partial X_\sigma} \\
\frac{\partial \varepsilon_{32}}{\partial Y_\sigma} &= -\frac{\partial \varepsilon_{23}}{\partial Y_\sigma} \\
\frac{\partial \varepsilon_{32}}{\partial T_\sigma} &= -\frac{\partial \varepsilon_{23}}{\partial T_\sigma} \\
\frac{\partial \varepsilon_{32}}{\partial n_\perp} &= -\frac{\partial \varepsilon_{23}}{\partial n_\perp} \\
\frac{\partial \varepsilon_{32}}{\partial n_\parallel} &= -\frac{\partial \varepsilon_{23}}{\partial n_\parallel}
\end{aligned} \tag{4.64}$$

The derivatives of the refractive index with respect to the space coordinates are

$$n_\parallel = (b_z n_z + b_r n_r + b_\phi \frac{m}{r})/b$$

where,

$$n_\phi = m/r$$

$$n^2 = n_z^2 + n_r^2 + (\frac{m}{r})^2$$

$$n_\perp = \sqrt{n^2 - n_\parallel^2}$$

$$b = \sqrt{b_z^2 + b_r^2 + b_\phi^2}$$

$$\frac{\partial b}{\partial r} = \frac{1}{b}(b_r \frac{\partial b_r}{\partial r} + b_z \frac{\partial b_z}{\partial r} + b_\phi \frac{\partial b_\phi}{\partial r})$$

$$\frac{\partial b}{\partial z} = \frac{1}{b}(b_r \frac{\partial b_r}{\partial z} + b_z \frac{\partial b_z}{\partial z} + b_\phi \frac{\partial b_\phi}{\partial z})$$

$$\frac{\partial b}{\partial \phi} = \frac{1}{b}(b_r \frac{\partial b_r}{\partial \phi} + b_z \frac{\partial b_z}{\partial \phi} + b_\phi \frac{\partial b_\phi}{\partial \phi})$$

$$\frac{\partial n}{\partial r} = -\frac{1}{n}(\frac{m^2}{r^3})$$

$$\frac{\partial n}{\partial z} = 0$$

$$\frac{\partial n}{\partial \phi} = 0$$

$$\frac{\partial n_\parallel}{\partial r} = \frac{1}{b}(\frac{\partial b_z}{\partial r}n_z + \frac{\partial b_r}{\partial r}n_r + \frac{\partial b_\phi}{\partial r}\frac{m}{r} - \frac{\partial b_\phi}{\partial r}\frac{m}{r^2}) - \frac{n_\parallel}{b}\frac{\partial b}{\partial r}$$

$$\frac{\partial n_\parallel}{\partial z} = \frac{1}{b}(\frac{\partial b_z}{\partial z}n_z + \frac{\partial b_r}{\partial z}n_r + \frac{\partial b_\phi}{\partial z}\frac{m}{r}) - \frac{n_\parallel}{b}\frac{\partial b}{\partial z}$$

$$\frac{\partial n_\parallel}{\partial \phi} = \frac{1}{b}(\frac{\partial b_z}{\partial \phi}n_z + \frac{\partial b_r}{\partial \phi}n_r + \frac{\partial b_\phi}{\partial \phi}\frac{m}{r}) - \frac{n_\parallel}{b}\frac{\partial b}{\partial \phi}$$

$$\frac{\partial n_\parallel}{\partial n_r} = \frac{b_r}{b}$$

$$\frac{\partial n_\parallel}{\partial n_z} = \frac{b_z}{b}$$

$$\frac{\partial n_\parallel}{\partial m} = \frac{b_\phi}{rb}$$

$$\frac{\partial n_\perp}{\partial r} = \frac{1}{n_\perp}(n\frac{\partial n}{\partial r} - n_\parallel\frac{\partial n_\parallel}{\partial r}) = \frac{1}{n_\perp}(-\frac{m^2}{r^3} - n_\parallel\frac{\partial n_\parallel}{\partial r})$$

$$\frac{\partial n_\perp}{\partial z} = \frac{1}{n_\perp}(n\frac{\partial n}{\partial z} - n_\parallel\frac{\partial n_\parallel}{\partial z}) = -\frac{1}{n_\perp}(n_\parallel\frac{\partial n_\parallel}{\partial z})$$

$$\frac{\partial n_\perp}{\partial \phi} = \frac{1}{n_\perp}(n\frac{\partial n}{\partial \phi} - n_\parallel\frac{\partial n_\parallel}{\partial \phi}) = -\frac{1}{n_\perp}(n_\parallel\frac{\partial n_\parallel}{\partial \phi})$$

$$\frac{\partial n_\perp}{\partial n_r} = \frac{1}{n_\perp}(n_r - n_\parallel\frac{\partial n_\parallel}{\partial n_r})$$

$$\frac{\partial n_\perp}{\partial n_z} = \frac{1}{n_\perp}(n_z - n_\parallel\frac{\partial n_\parallel}{\partial n_z})$$

$$\frac{\partial n_\perp}{\partial m} = \frac{1}{n_\perp}(\frac{m}{r^2} - n_\parallel\frac{\partial n_\parallel}{\partial m})$$

The dielectric tensor element K_{zze}depends on the derivative of the plasma dispersion function $dZ_0(y_0)/dy_0$.The real part of Z_0 function has the maximal and the minimal values at y_0 equal to the critical value $y_0 = y_{0cr} = \pm 0.924$. The right hand side of the ray-tracing equations contains the first and the second derivatives from $Z_0(y_0)$. These derivatives change the signs near y_{0cr}. This creates problems when the ray trajectory goes through the points with y_0 close to y_{0cr}. To avoid this problem the code can change the type of the dispersion relation from the Ono dispersion, **id**=8, to the cold plasma dispersion relation, **id**=2, in the vicinity of y_{0cr}. For the option **iswitch**=1 the code will use cold plasma dispersion for $\mid y_0 - 0.95 * sign(y_0) \mid < 0.03$.

4.6 The dispersion relation for non-Hermitian dielectric tensors.

We used the dispersion relation for non-Hermitian dielectric plasma proposed in [2]. This form of the dispersion function is realized for the Mazzucato electron relativistic tensor **id=5** and for the hot non-relativistic plasma **id=9.**
For the plasma with non-Hermitian dielectric tensors $\widehat{\varepsilon} \neq \widehat{\varepsilon}^H$ the dispersion relation is

$$0 = D(\overrightarrow{k}, \omega, \overrightarrow{r}, t) = det\widehat{\varepsilon}(\overrightarrow{k}, \omega, \overrightarrow{r}, t) = P(\overrightarrow{k}, \omega, \overrightarrow{r}, t) + iQ(\overrightarrow{k}, \omega, \overrightarrow{r}, t) \quad (4.65)$$

Here the function D has non-zero imaginary part. The equation 4.65 gives a couple of the equations for the real functions

$$P(\overrightarrow{k}, \omega, \overrightarrow{r}, t) = 0, Q(\overrightarrow{k}, \omega, \overrightarrow{r}, t) = 0 \quad (4.66)$$

In this case we have two equations for one function $\omega = \omega(\overrightarrow{k}, \overrightarrow{r}, t)$ so this system has not the solution. In [2] it was proposed to introduce the complex frequency $\Omega = \omega + i\nu$ with small imaginary part $\nu \ll \omega$. In this case the condition, that the determinant of the wave amplitude system should be equal zero, gives the equation

$$D(\overrightarrow{k}, \omega + i\nu, \overrightarrow{r}, t) = det\widehat{\varepsilon}(\overrightarrow{k}, \omega + i\nu, \overrightarrow{r}, t) = 0 \quad (4.67)$$

For small ν this equation can be approximately written as

$$D(\overrightarrow{k}, \omega, \overrightarrow{r}, t) + iD_\omega(\overrightarrow{k}, \omega, \overrightarrow{r}, t) \approx 0 \quad (4.68)$$

Using the real and imaginary parts of D 4.65 in 4.68 we get

$$\nu = P/Q_\omega = -Q/P_\omega \quad (4.69)$$

Let

$$\Phi(\overrightarrow{k}, \omega, \overrightarrow{r}, t) = P^2(\overrightarrow{k}, \omega, \overrightarrow{r}, t) + Q^2(\overrightarrow{k}, \omega, \overrightarrow{r}, t) \quad (4.70)$$

From 4.69

$$\Phi_\omega = 2PP_\omega + QQ_\omega = 0 \quad (4.71)$$

Equation 4.71 is the dispersion relation that determines $\omega = \omega(\overrightarrow{k}, \overrightarrow{r}, t)$. This dispersion function 4.71 is used in the ray tracing equations.

4.7 Change in the dispersion function along the ray.

In some conditions the ray trajectory reaches a point where the dispersion function does not permit the trajectory to go through this point. For example, this can occur for hot plasma when the trajectory goes through the gyro-resonance area, where the dispersion relation has points of bifurcation or conversion from X to EBW mode. In this case the code can change the form of dispersion function used. The code changes:

a) the dispersion function used in ray-tracing equations ;

b) the formula for absorption calculations.

Away from the resonance point, the dispersion relation and absorption are specified by parameters **id** and **iabsorp** given in the input **genray.dat** file. If **iswitch=1,** the code changes these parameters to **idswitch** and **iabswitch** in the vicinity of the cyclotron-resonance points $|1 - nY_j| < del_y, n = 0, \pm 1, \pm 2....$ The type of plasma species j is determined by parameter $j =$**jy_d.** Parameters **iswitch, jy_d, del_y, idswitch** and **iabswitch** are given in **genray.dat** file. Parameter **iy_d** can be equal =1 for electrons or **jy_d=2**.,...**nbulk** for ions.

Chapter 5

Wave power absorption

Wave power P along the ray trajectory can be found in the following formula:

$$P(l) = P(0)\exp(-2\int_0^l Im(\overrightarrow{k})d\overrightarrow{l}) \tag{5.1}$$

Here $\overrightarrow{l}$ is the vector directed parallel the trajectory, l is the distance along the ray, $P(0)$ is the wave power at launch point. The integral is calculated along the ray trajectory.

$$Im(\overrightarrow{k})\cdot d\overrightarrow{l} = \frac{(Im\overrightarrow{k})\cdot\overrightarrow{V}_{gr}}{V_{gr}}dl = \frac{\omega}{c}\frac{Im\overrightarrow{N}\cdot\overrightarrow{V}_{gr}}{V_{gr}}dl = \frac{\omega}{c}\frac{ImN_\perp V_{gr_\perp}}{V_{gr_p}}dl_p. \tag{5.2}$$

$\overrightarrow{V}_{gr}$ is wave group velocity, V_{gr_p} is the poloidal-plane component of the group velocity, $V_{gr_\perp}$ is group velocity perpendicular to the magnetic field.

To calculate wave absorption we need to know the imaginary part of perpendicular refractive index $ImN_\perp$. GENRAY has several possibilities for calculation of $ImN_\perp$. For calculation $ImN_\perp$, the code uses the anti-Hermitian parts of different forms of the dielectric tensor. The switch **iabsorp** allows various choices.

5.1 Mazzucato approximation of the relativistic dielectric tensor for electron plasma

(**iabsorp=1**)

In this case the code uses the Mazzucato solver[5] for determination $ReN\perp + iImN_\perp$ as a root of dispersion relation $D_{Maz}(N_\parallel, ReN\perp + iImN_\perp, X_e, Y_e, T_e) = 0$.

5.2 Anti-Hermitian relativistic dielectric tensor for electron plasma

(**iabsorp=6**)

5.2.1 The tensor formula.

In this case we use the numerical calculations of anti-Hermitian dielectric tensor $\widehat{\varepsilon}_a$ for electron plasma with relativistic Maxwellian electrons (R.W.Harvey et al., 1993)[6], using formulas given in (M.Bornatici et al., 1985) [7],

$$\widehat{\varepsilon}_{relativist_a} = -\pi\frac{\omega_p^2}{\omega^2}\sum_{n=-\infty}^{n=\infty}\int d^3pU^{(n)}(f)\widehat{S}^{(n)}\delta(\gamma-\frac{k_\parallel v_\parallel}{\omega}-\frac{n\omega_{ce}}{\omega}), \tag{5.3}$$

where,

$$U^{(n)}(f)\equiv\frac{1}{\gamma}\left[\frac{n\omega_{ce}}{\omega}\frac{\partial f}{\partial p_\perp}+N_\parallel\frac{p_\perp}{mc}\frac{\partial f}{\partial p_\parallel}\right], \tag{5.4}$$

$$\widehat{S}^{(n)}\equiv\begin{bmatrix} p_\perp(\frac{nJ_n}{b})^2 & -ip_\perp\frac{nJ_nJ_n'}{b} & p_\parallel\frac{nJ_n^2}{b} \\ ip_\perp\frac{nJ_nJ_n'}{b} & p_\perp(J_n')^2 & ip_\parallel J_nJ_n' \\ p_\parallel\frac{nJ_n^2}{b} & -ip_\parallel J_nJ_n' & \frac{p_\parallel^2}{p_\perp}J_n^2 \end{bmatrix}. \tag{5.5}$$

Here, ω_{ce} is rest mass electron cyclotron frequency, m is electron rest mass, p is momentum, $u=p/m$ is momentum per unit mass, $\gamma=\sqrt{1+(p/mc)^2}$ is the relativistic factor, $v=p/\gamma m$ is velocity, symbols $\parallel$ and $\perp$ refer to direction with respect to ambient magnetic field, and $f(p_\perp,p_\parallel)$ is the relativistic electron distribution function normalized to $\int d^3pf=1$. Bessel function $J_n(b)$ and its derivatives $J_n'=dJ_n/db$ at argument $b=|\frac{k_\perp p_\perp}{m\omega_{ce}}|$ measure the ratio of the electron Larmor radius and the perpendicular wave length. Tensor $\widehat{S}^{(n)}$ is represented in a Cartesian coordinate system with the z-axis along magnetic field $\overrightarrow{B}$ and y -axis in the $\overrightarrow{k}\times\overrightarrow{B}$ -direction. Here n is the number of the gyro-harmonic.

It is convenient to introduce normalized momentum $\overline{p}=p/mc$, functions $\overline{U}^{(n)}(f(\overline{p}_\perp,\overline{p}_\parallel))=mcU^{(n)}(f(\overline{p}_\perp,\overline{p}_\parallel))$ and $\overline{\widehat{S}}^{(n)}(\overline{p}_\perp,\overline{p}_\parallel)=\frac{\overline{p}_\perp\widehat{S}^{(n)}(\overline{p}_\perp,\overline{p}_\parallel)}{mc}$. For variable $\gamma=\sqrt{1+\overline{p}^2}$, $b=N_\perp\overline{p}/|Y|$ (??sign of Y),

$$\overline{U}^{(n)}(f(\overline{p}_\perp,\overline{p}_\parallel))\equiv\frac{1}{\gamma}\left[nY\frac{\partial f}{\partial\overline{p}_\perp}+N_\parallel\overline{p}_\perp\frac{\partial f}{\partial\overline{p}_\parallel}\right], \tag{5.6}$$

$$\overline{\widehat{S}}^{(n)}\equiv\begin{bmatrix} \overline{p}_\perp^2(\frac{nJ_n}{b})^2 & -i\overline{p}_\perp^2\frac{nJ_nJ_n'}{b} & \overline{p}_\parallel\overline{p}_\perp\frac{nJ_n^2}{b} \\ i\overline{p}_\perp^2\frac{nJ_nJ_n'}{b} & \overline{p}_\perp^2(J_n')^2 & i\overline{p}_\parallel\overline{p}_\perp J_nJ_n' \\ \overline{p}_\parallel\overline{p}_\perp\frac{nJ_n^2}{b} & -i\overline{p}_\parallel\overline{p}_\perp J_nJ_n' & \overline{p}_\parallel^2J_n^2 \end{bmatrix}, \tag{5.7}$$

$$\widehat{\varepsilon}_{relativist_a} = -\pi X (mc)^3 \sum_{n=-\infty}^{n=\infty} \int d^3\overline{p}\overline{U}^{(n)}(f(\overline{p}_\perp,\overline{p}_\parallel))\overline{\overline{S}}^{(n)}(\overline{p}_\perp,\overline{p}_\parallel)\delta(\gamma - N_\parallel\overline{p}_\parallel - nY)/\overline{p}_\perp. \tag{5.8}$$

The relativistic cyclotron resonance condition is:

$$\gamma - N_\parallel\overline{p}_\parallel - nY = 0. \tag{5.9}$$

We calculate the anti-Hermitian part of the relativistic dielectric tensor for the normalized relativistic electron distribution function f , $\int d^3pf = 1$. The dielectric tensor 5.3 for the normalized distribution function is

$$\widehat{\varepsilon}_{relativist_a} = -\pi X \sum_{n=-\infty}^{n=\infty} \widehat{I}^{(n)}. \tag{5.10}$$

$$\begin{aligned} \widehat{I}^{(n)} &= (mc)^3 \int d^3\overline{p}\overline{U}^{(n)}(f)\overline{\overline{S}}^{(n)}\delta(\gamma - N_\parallel\overline{p}_\parallel - nY)/\overline{p}_\perp = \\ &= 2\pi(mc)^3 \int d\overline{p}_\perp d\overline{p}_\parallel \overline{U}^{(n)}(f)\overline{\overline{S}}^{(n)}\delta(\gamma - N_\parallel\overline{p}_\parallel - nY) = \\ &= 2\pi \int d\overline{p}_\perp d\overline{p}_\parallel \overline{U}^{(n)}(\overline{f})\overline{\overline{S}}^{(n)}\delta(\gamma - N_\parallel\overline{p}_\parallel - nY). \end{aligned} \tag{5.11}$$

Here delta function $\delta(g^{(n)})$ in 5.11 has the argument:

$$g^{(n)}(\overline{p}_\perp,\overline{p}_\parallel) = \gamma - N_\parallel\overline{p}_\parallel - nY. \tag{5.12}$$

The derivative from the $g^{(n)}$ by $\overline{p}_\parallel$ is:

$$\frac{\partial g^{(n)}}{\partial \overline{p}_\parallel} = \frac{\overline{p}_\parallel - N_\parallel\gamma}{\gamma}. \tag{5.13}$$

Here we introduce the tensor:

$$\widehat{F}^{(n)}(\overline{p}_\perp,\overline{p}_\parallel) = 2\pi\overline{U}^{(n)}(\overline{f}(\overline{p}_\perp,\overline{p}_\parallel))\overline{\overline{S}}^{(n)}(\overline{p}_\perp,\overline{p}_\parallel). \tag{5.14}$$

The expression 5.11 can be rewritten using 5.12, 5.13 and 5.14:

$$\begin{aligned} \widehat{I}^{(n)} &= \int \widehat{F}^{(n)}(\overline{p}_\perp,\overline{p}_\parallel)d\overline{p}_\perp d\overline{p}_\parallel \delta(g^{(n)}(\overline{p}_\perp,\overline{p}_\parallel)) = \\ &= \int_0^{\overline{p}^{(n)}_{\perp_0}} d\overline{p}_\perp \sum_{k=1,2} \frac{\widehat{F}^{(n)}(\overline{p}_\perp,\overline{p}^{(n)}_{\parallel_k}(\overline{p}_\perp))}{|\frac{\partial}{\partial\overline{p}_\parallel}g^{(n)}(\overline{p}_\perp,\overline{p}^{(n)}_{\parallel_k}(\overline{p}_\perp))|.} \end{aligned} \tag{5.15}$$

Here, $\overline{p}_{\parallel_k}(\overline{p}_\perp)$ are the roots of resonance condition 5.9. For different cyclotron resonance conditions, the sum over index k has one or two terms, and the upper integral limit $\overline{p}^{(n)}_{\perp_0}$ can be finite $\overline{p}^{(n)}_{\perp_0} = \frac{N_\parallel^2 + n^2Y^2 - 1}{1 - N_\parallel^2}$ or infinite $\overline{p}^{(n)}_{\perp_0} = \infty$ number. We introduce the tensor:

$$\widehat{G}^{(n)}_k(\overline{p}_\perp) = \frac{\widehat{F}^{(n)}(\overline{p}_\perp,\overline{p}^{(n)}_{\parallel_k}(\overline{p}_\perp))}{|\frac{\partial}{\partial\overline{p}_\parallel}g(\overline{p}_\perp,\overline{p}^{(n)}_{\parallel_k}(\overline{p}_\perp))|.} \tag{5.16}$$

The final formula for $\widehat{\varepsilon}_a$ is:

$$\widehat{\varepsilon} = -\pi X \sum_{n=-\infty}^{n=\infty} \widehat{I}^{(n)}, \tag{5.17}$$

$$\widehat{I}^{(n)} = \int_0^{\overline{p}_{\perp 0}} d\overline{p}_\perp \sum_{k=1,2} \widehat{G}_k^{(n)}(\overline{p}_\perp), \tag{5.18}$$

$$\widehat{G}_k^{(n)}(\overline{p}_\perp) = \frac{2\pi \overline{U}^{(n)}(\overline{f}(\overline{p}_\perp, \overline{p}_\parallel))}{|\frac{\partial}{\partial \overline{p}_\parallel} g(\overline{p}_\perp, \overline{p}_{\parallel_k}^{(n)}(\overline{p}_\perp))|} \overline{\widehat{S}}^{(n)}(\overline{p}_\perp, \overline{p}_{\parallel_k}^{(n)}), \tag{5.19}$$

$$\gamma_k^{(n)}(\overline{p}_\perp) = \sqrt{1 + (\overline{p}_\perp)^2 + (\overline{p}_{\parallel_k}^{(n)}(\overline{p}_\perp))^2}. \tag{5.20}$$

We calculate anti-Hermitian relativistic tensor $\widehat{\varepsilon}_{relativist_a}$ numerically on the $\overline{p}_\perp$ mesh.

For **iabsorp=6,** GENRAY calculates the imaginary part $ImN_\perp$ using the Hermitian part of dielectric tensor 4.13 for the hot non-relativistic plasma and the anti-Hermitian dielectric tensor 5.3

$$ImN_\perp = \frac{ImD(N_\parallel, ReN_\perp, \widehat{\varepsilon}_{hot_H} + i\widehat{\varepsilon}_{relativist_a})}{\frac{\partial}{\partial Re(N_\perp)} D(N_\parallel, ReN_\perp, \widehat{\varepsilon}_{hot_H})}. \tag{5.21}$$

Here, $\widehat{\varepsilon}_{hot_H} = 0.5(\widehat{\varepsilon}_{hot} + \widehat{\varepsilon}_{hot}^{T\star})$ is the Hermitian part of the hot non-relativistic dielectric tensor.

5.2.2 Cyclotron resonance curves

The cyclotron resonance condition 5.9 gives resonance curves at $(\overline{p}_\perp, \overline{p}_\parallel)$ plate. The resonance curve can be an ellipse, parabola or hyperbola . The resonance curve is the solution of the equation:

$$\gamma = N_\parallel \overline{p}_\parallel + nY. \tag{5.22}$$

This equation gives the following conditions:

$$N_\parallel \overline{p}_\parallel + nY > 1 > 0, \tag{5.23}$$

$$\overline{p}_\perp^2 (1 - N_\parallel^2) - 2N_\parallel \overline{p}_\parallel nY + \overline{p}_\perp^2 = n^2 Y^2 - 1. \tag{5.24}$$

For $N_\parallel^2 \neq 1$, the resonance curve 5.24 can be rewritten in the following form:

$$A^2 sign(1 - N_\parallel^2)(\overline{p}_\parallel - \frac{N_\parallel nY}{1 - N_\parallel^2})^2 + \overline{p}_\perp^2 = R^2 sign(\frac{n^2Y^2 + N_\parallel^2 - 1}{1 - N_\parallel^2}), \tag{5.25}$$

where

$$R^2 = |\frac{n^2Y^2 + N_\parallel^2 - 1}{1 - N_\parallel^2}|, \tag{5.26}$$

$$A^2 = |1 - N_\parallel^2|. \tag{5.27}$$

If $N_\parallel^2 < 1$, the resonance curve 5.25 is an ellipse:

If $n^2Y^2 + N_\parallel^2 - 1 > 0$, equation 5.25 has no solution.

If $n^2Y^2 + N_\parallel^2 - 1 \geq 0$, the ellipse resonance curve is:

$$\frac{(\overline{p}_\parallel - \frac{N_\parallel nY}{1-N_\parallel^2})^2}{(\frac{R}{A})^2} + \frac{\overline{p}_\perp^2}{R^2} = 1. \tag{5.28}$$

The two roots of equation 5.28 are:

$$\overline{p}_{\parallel_{1,2}}^{(n)}(\overline{p}_\perp) = \frac{N_\parallel nY \pm \sqrt{N_\parallel^2 + n^2Y^2 - 1 + (N_\parallel^2 - 1)\overline{p}_\perp^2}}{1 - N_\parallel^2}. \tag{5.29}$$

These two roots exist for the following $\overline{p}_\perp$ values:

$$0 \leq \overline{p}_\perp \leq \overline{p_{\perp_0}^{(n)}} = \sqrt{\frac{n^2Y^2 + N_{parallel}^2 - 1}{1 - N_\parallel^2}}. \tag{5.30}$$

If $N_\parallel^2 > 1$, the resonance curve 5.25 is a hyperbola:

$$\frac{(\overline{p}_\parallel - \frac{N_\parallel nY}{1-N_\parallel^2})^2}{(\frac{R}{A})^2} - \frac{\overline{p}_\perp^2}{R^2} = 1. \tag{5.31}$$

This equation has only one root which obeys condition 5.23.

If $N_\parallel > 1$, the root of equation 5.31 is:

$$\overline{p}_{\parallel_2}^{(n)}(\overline{p}_\perp) = \frac{N_\parallel nY - \sqrt{N_\parallel^2 + n^2Y^2 - 1 + (N_\parallel^2 - 1)\overline{p}_\perp^2}}{1 - N_\parallel^2}, \tag{5.32}$$

$$0 \leq \overline{p}_\perp < \infty, \tag{5.33}$$

$$\overline{p}_{\parallel_2}^{(n)}(\overline{p}_\perp) > \overline{p}_{\parallel_2}^{(n)}(\overline{p}_\perp = 0) = \frac{N_\parallel nY - \sqrt{N_\parallel^2 + n^2Y^2 - 1}}{1 - N_\parallel^2}. \tag{5.34}$$

If $N_\parallel < -1$, the root of equation 5.31 is:

$$\overline{p}_{\parallel_1}^{(n)}(\overline{p}_\perp) = \frac{N_\parallel nY + \sqrt{N_\parallel^2 + n^2Y^2 - 1 + (N_\parallel^2 - 1)\overline{p}_\perp^2}}{1 - N_\parallel^2}, \tag{5.35}$$

$$0 \leq \overline{p}_\perp < \infty, \tag{5.36}$$

$$\overline{p}_{\parallel_1}^{(n)}(\overline{p}_\perp) < \overline{p}_{\parallel_1}^{(n)}(\overline{p}_\perp = 0) = \frac{N_\parallel nY + \sqrt{N_\parallel^2 + n^2Y^2 - 1}}{1 - N_\parallel^2}. \tag{5.37}$$

For $N_\parallel^2 = 1$, the resonance curve 5.24 is transformed to a parabola:

$$2N_\parallel \overline{p}_\parallel nY = \overline{p}_\perp^2 - n^2Y^2 + 1. \tag{5.38}$$

If $nY \leq 0$, equation 5.40 with condition 5.23 has no solution. .
If $nY > 0$, equation 5.40 with condition 5.23 has one root.

$$\overline{p}_{\parallel_1}^{(n)}(\overline{p}_\perp) = \frac{1}{2nYN_\parallel}(\overline{p}_\perp^2 - n^2Y^2 + 1), \tag{5.39}$$

$$0 \leq \overline{p}_\perp < \infty. \tag{5.40}$$

For $N_\parallel = 1$,

$$\overline{p}_{\parallel_1}^{(n)}(\overline{p}_\perp) \geq \overline{p}_{\parallel_1}^{(n)}(\overline{p}_\perp = 0) = \frac{1}{2nYN_\parallel}(-n^2Y^2 + 1). \tag{5.41}$$

For $N_\parallel = -1$,

$$\overline{p}_{\parallel_1}^{(n)}(\overline{p}_\perp) \leq \overline{p}_{\parallel_1}^{(n)}(\overline{p}_\perp = 0) = \frac{1}{2nYN_\parallel}(-n^2Y^2 + 1). \tag{5.42}$$

5.2.3 The distribution functions.

GENRAY has the several possibilities to choose the form of the relativistic electron distribution function. The option **i_diskf** (set in genray.dat file) determines the form of the used distribution.

Analytical relativistic Maxwellian distribution (i_diskf=0). In this case the code uses the analytical relativistic Maxwellian distribution with the radial electron temperature profile.

$$f_m = \frac{1}{4\pi m^2 cTK_2(\theta)} \exp(-\theta\gamma) = \frac{1}{(mc)^3}\overline{f}_m(\theta,\gamma), \tag{5.43}$$

$$\overline{f}_m(\theta,\gamma) = \frac{\theta}{4\pi K_2(\theta)} \exp(-\theta\gamma) = \frac{\theta}{4\pi K_2(\theta)\exp(\theta)} \exp(\theta(1-\gamma)) \tag{5.44}$$

This function $\overline{f}_m$ is normalized $\int d^3\overline{p}\overline{f}_m = 1$. Here, $T(\rho)$ is the electron temperature, $\theta = \frac{mc^2}{T}$, K_2 is the Macdonald function. For the Maxwellian distribution 5.44, function 5.6 has the form:

$$\overline{U}^{(n)}(\overline{f}_m(\overline{p}_\perp,\overline{p}_\parallel)) = -\frac{\overline{p}_\perp \theta \overline{f}_m}{\gamma^2}(nY + N_\parallel \overline{p}_\parallel). \tag{5.45}$$

Relativistic distribution specified on a mesh, from diskf file written by CQL3D (i_diskf=1). In this case GENRAY reads the mesh 3D distribution from the ASCII file **diskf** and creates the function that approximates this mesh distribution.

Relativistic distribution specified on a mesh, from netcdfnm.nc file written by CQL3D (i_diskf=2). In this case GENRAY reads the mesh 3D distribution from netcdf file **netcdfnm.nc** and creates the function that approximates this mesh distribution.

Model analytic relativistic non-Maxwellian distribution (i_diskf=3). In this case GENRAY creates the model distribution, using the input variables from namelist **/read_diskf/** in **genray.dat** file.

$$f(\overline{v},\chi,\rho) = n(\rho)(1-rtail-rhot-rbeam)f_m(T(\rho)) + rtail * f_{tail} + rhot * f_{hot} + rbeam * f_{beam} \tag{5.46}$$

Where $n(\rho)$ is the electron density radial profile, f_m is the normalized Maxwellian relativistic electron distribution with the electron temperature radial profile, $\int f_m d^3v = 1$, f_{tail} is the tail distribution, f_{beam} is the distribution created by beam, f_{hot} is the tail distribution , rtail, rhot and rbeam are the ratios of the tail, hot and beam densities to the total plasma density.
The **tail** distribution

$$f_{tail} = c_{tail} H(\rho, rt1, rt2) * f_m(ttail) \tag{5.47}$$

Where H(x,x1,x2) is the box function

$$H(x,x1,x2) = \begin{cases} 1, x \subset (x1,x2) \\ 0, x \notin (x1,x2) \end{cases} \tag{5.48}$$

rt1 and **rt2** are the normalized small radii for tail localization, $f_m(ttail)$ is the relativistic Maxwellian distribution, $ttail$ is the tail temperature.
The **beam** distribution

$$f_{beam} = c_{beam} H(\rho, rb1, rb2) * \exp(-\frac{mc^2}{2 * tbeam} \frac{(p_\parallel - p_{beam_\parallel})^2 + (p_\perp - p_{beam_\perp})^2}{(mc)^2}) \tag{5.49}$$

$$p_{beam} = m\sqrt{\frac{e_{beam}}{m}} \tag{5.50}$$

$$p_{beam_\parallel} = p_{beam}\cos\theta_{beam}, p_{beam_\perp} = p_{beam}\sin\theta_{beam} \tag{5.51}$$

Where **rb1** and **rb2** are the small radii for beam localization, $tbeam$ is the beam component temperature, θ_{beam}=**thbeam** (degree) is the beam pitch angle.
The **hot** function approximates the distribution created by the quasi-linear RF heating

$$f_{hot} = c_{hot} H(\rho, rh1, rh2) H(epar, hotmnpar, hotmxnpar) H(epar, hotmnper, hotmxnpar) \\ (\frac{p_\perp}{mc})^{hotexp} \exp(-\frac{1}{2}(\frac{mc^2}{thotpar}\left(\frac{p_\parallel}{mc}\right)^2 + \frac{mc^2}{thotper}\left(\frac{p_\perp}{mc}\right)^2)) \tag{5.52}$$

Where **rh1** and **rh2** are the small radii for the hot component localization, **thotpar** and **thotper** are the parallel and perpendicular temperatures of the hot component,

$epar = (\sqrt{1+\frac{v_\parallel^2}{c^2}}-1)mc^2$, $eper = (\sqrt{1+\frac{v_\perp^2}{c^2}}-1)mc^2$ are the parallel and perpendicular energies, **(hotmnpar,hotmxnpar)** and **(hotmnper,hotmxnper)** are the boundaries of the parallel and perpendicular energy boxes **hotmnpar<** epar **<hotmxnpar**, **hotmnper<** eper **<hotmxnper**.
The tail, beam and hot distributions are normalized to unity $\int f_{tail} d^3v = 1$, $\int f_{beam} d^3v = 1$, $\int f_{hot} d^3v = 1$.

5.3 Anti-Hermitian dielectric tensor for hot non-relativistic electron-ion plasma

(**iabsorp=4**)

For **iabsorp=4** , GENRAY calculates the imaginary part $ImN_\perp$ using the dielectric tensor 4.13 for hot non-relativistic plasma.

$$ImN_\perp = \frac{ImD(N_\parallel, ReN_\perp, \widehat{\varepsilon}_{hot})}{\frac{\partial}{\partial Re(N_\perp)} D(N_\parallel, ReN_\perp, \widehat{\varepsilon}_{hot_H})}. \tag{5.53}$$

Here $\widehat{\varepsilon}_{hot_H} = 0.5(\widehat{\varepsilon}_{hot} + \widehat{\varepsilon}^{T\star}_{hot})$ is the Hermitian part of the hot non-relativistic dielectric tensor.

5.4 Fast wave absorption

iabsorp=3

For fast waves, the code can calculate the imaginary part on the perpendicular component of the refractive index $ImN_\perp$ using results from (S. C. Chiu et al., 1989)[8]. The code calculates electron and ion absorption separately $ImN_\perp = ImN_{\perp_e} + ImN_{\perp_i}$.

5.5 Lower hybrid wave absorption

iabsorp=2

For lower hybrid waves, the code can calculate the imaginary part of the perpendicular component of the refractive index $ImN_\perp$ using the results from (P. Bonoli, 1984) [9]. The code calculates the electron $ImN_{\perp_e}$, ion $ImN_{\perp_i}$, and collisional $ImN_{\perp_{cl}}$ absorption separately giving total absorption $ImN_\perp = ImN_\perp + ImN_{\perp_i} + ImN_{\perp_{cl}}$.

Chapter 6

Stochastic wave scattering

The code has the capability of switching on the stochastic scattering of perpendicular refractive index $N_\perp$. For example, this can be used for modeling the scattering of low hybrid waves by drift-wave density fluctuations (P.L. Anderson and F.W.Perkins, 1983) [10]. We use angle scattering of refractive vector $\overrightarrow{N}_\perp$ in the plate perpendicular to the magnetic field $\overrightarrow{B}$ in small radius points ρ_{sc} along the ray trajectories. If the ray trajectory passes this radius or the ray is reflected from the plasma boundary, then polar angle ϑ will be scattered with the deviation $\sqrt{2\sigma_{sc}(\rho_{sc})}$

$$\triangle\vartheta = \sqrt{2\sigma_{sc}}\zeta. \tag{6.1}$$

Where :

ϑ is a polar angle in the plate perpendicular to the magnetic field ;

$\zeta(0,1)$ is the random number with normal distribution $M(\zeta) = 0$, $M(\zeta^2) = 1$;

$\sigma_{sc}(\rho)$ is the given angle dispersion.

The input parameters for $\overrightarrow{N}_\perp$ angle scattering are:

iscat is the switch of the scattering:

iscat=1 to switch on the scattering, **=0** to switch out;

nscat_n is the number of the small radius points at which the code will scatter $\overrightarrow{N}_\perp$; this parameter is given in param.i file;

scatd(0) is the mean square scattering angle $(radian^2)$ for the reflection near the plasma boundary;

rhoscat(**1:nscat_n**) are the small radii for the scattering location;

scatd(i)=$\sigma_{sc}(rhoscat(i))$, i=1**,...,nscat_n** are the mean square scattering angles $(radian^2)$ at the small radii.

Chapter 7

Wave launch and reflection

The code has several possibilities for ray launch. It can launch a ray from a vacuum point outside the plasma or from an interior point inside the plasma. The switch **istart** chooses the launch method.

7.1 Ray cone

istart=1

The cone pencil of rays launched in the vacuum outside the plasma is a model of the ECR launch. In this case the rays are launched on a set of circular cones with the same cone vertex. All cone axes coincide with one central ray.

The input parameters for the cone are in **genray.dat** file:

1) (**rst,phist,zst**) are cylindrical coordinates (r,φ,z) of the cone vertex. Here, lengths **rst, phist** are in (m), the toroidal angle **phist** is in degrees.

2) The direction of the central ray, going from the cone vertex, is given by two angles (in degrees):

alfast is toroidal angle α of the central ray , measured from r-vector at the source, counter clockwise around z axis;

betast is poloidal angle β of the central ray, β is the angle between the central ray and the plate $z = const$.

The refractive unit index $\overrightarrow{N}$ of the central ray has the coordinates :

$N_r = \sin(\pi/2-\beta)\cos\alpha, N_\varphi = \sin(\pi/2-\beta)\sin\alpha, N_z = \cos(\pi/2-\beta)$

3)The specifications of the cones are:

alpha1 is angle width of the power distribution (the maximum cone angle) measured from the central ray;

na1 is number of cones (if **na1**=0, the wave cone consists from the central ray only);

na2 is number of rays per cone;

alpha2 is cone angle;

total number of rays i_{rays}=**na1*na2**+1.

4) Power distribution among rays:

The total power from antenna is **powtot** (MWt). The power distribution over cone angle α has the following form:

$$P(\alpha) = P_0 \exp(-2(\frac{\alpha}{\alpha_1})^2). \tag{7.1}$$

Distribution 7.1 is normalized on the total antenna power:

$$\int_{\alpha=0}^{\alpha=\alpha_1} P(\alpha) \sin\alpha d\alpha = powtot. \tag{7.2}$$

For the cone pencil, the code calculates the coordinates of the intersection of each ray cone $i = 1,..,i_{rays}$ with the plasma boundary $(R_{0i}.\varphi_{0i}, z_{0i})$. At the points of intersection $(R_0.\varphi_0, z_0)$, the code calculates the toroidal N_φ and poloidal N_θ components of the refractive vector.

$$N_\varphi = -N_x \sin\varphi_0 + N_y \cos\varphi_0, \tag{7.3}$$

$$N_\theta = (-\frac{\partial\psi}{\partial z}(N_x\cos\varphi_0 + N_y\sin\varphi_0) + \frac{\partial\psi}{\partial r}\sin\varphi_0)/|\nabla\psi|. \tag{7.4}$$

Here , the derivatives from the poloidal flux ψ are taken at the point of intersection $(R_0.\varphi_0, z_0)$.

7.2 Approximation of grill antenna launch.

istart=2

The rays are launched from points interior to the plasma, since the wave mode may be evanescent in the immediate vicinity of the low density plasma edge. The spectral distribution of the rays is specified parametrically. The parameters of the grill are given in **genray.dat** file.

7.2.1 Location on the grill.

Location on the grill is determined by the following parameters:

ngrilla (in **param.i** file) is the maximal total number of $N_\parallel$ spectra used for setting of the arrays**,**

ngrill ($\leq$**ngrilla, given** in **genray.dat** file) is the total number of $N_\parallel$ spectra to be used; $i = 1,...,ngrill$ are the given $N_\parallel$ spectra. The different spectra can have the same or different poloidal locations.

rhopsi0(ngrill) is the initial small radius ρ_i for the front wave of each grill spectrum;

thgrill(ngrill) is poloidal angle θ_i (degrees) of each grill measured counter clockwise from the horizontal through the magnetic axis;

phigrill(ngrill) is toroidal angle φ_i (degrees) of each grill;

height(ngrill) is poloidal length h_i (m) of each grill (giving the poloidal power distribution of each grill);

nthin(ngrill) is the number of rays simulating a grill.

The space power distribution is determined by the following variables

powers(ngrill) is power in one grill (MWatts). The total power is powetot=$\sum_{i=1}^{ngrill} powers(i)$.
The form of the grill spectrum is determined by the following parameters:
The code has the several types of the refractive index setting. The variable **i_n_poloidal** chooses the type (see latter).
For **i_n_poloidal=1,2,3** cases:

anmin(ngrill) is position of left bound $N_{\|_{min}}$ of the power spectrum $P(N_\|)$; it can be negative;

anmax(ngrill) is position of right bound $N_{\|_{max}}$ of the power spectrum $P(N_\|)$; it can be negative;

nnkpar(ngrill) is number of points of power spectrum $P(N_\|)$;
For **i_n_poloidal=4** case:

antormin(ngrill) is position of left bound $N_{\phi_{min}}$ of the power spectrum $P(N_\phi, N_\theta)$; it can be negative;

antormax(ngrill) is position of right bound $N_{\phi_{max}}$ of the power spectrum $P(N_\phi, N_\theta)$; it can be negative;

nnktor(ngrill) is number of points of power spectrum $P(N_\phi, N_\theta)$ in N_ϕ direction;

anpolmin(ngrill) is position of left bound $N_{\theta_{min}}$ of the power spectrum $P(N_\phi, N_\theta)$ it can be negative;

anpolmax(ngrill) is position of right bound $N_{\theta_{max}}$ of the power spectrum $P(N_\theta, N_\theta)$ it can be negative;

nnkpol(ngrill) is number of points of power spectrum $P(N_\phi, N_\theta)$ in N_θ direction;

7.2.2 Power distribution along one grill.

Poloidal angle θ along a grill with the number i is in interval $(\theta_{i_{min}}, \theta_{i_{max}})$. Where $\theta_{i_{min}} = \theta_i + 0.5\triangle\theta_i$, $\theta_{i_{max}} = \theta_i - 0.5\triangle\theta_i$ and

$$\triangle\theta_i = 2\arctan(0.5h_i/\rho_i) \approx h_i/\rho_i. \tag{7.5}$$

We use the following power $P_{\theta i}(\theta)$ distribution along one i grill

$$P_{\theta i}(\theta) = p_{\theta i}\cos^2(\pi(\theta - \theta_i)/\triangle\theta_i). \tag{7.6}$$

This distribution $P_{\theta i}(\theta)$ is normalized on **powers**(i)

$$\frac{1}{\triangle\theta_i}\int_{\theta_{i_{min}}}^{\theta_{i_{max}}} P_{\theta i}(\theta)d\theta = powers(i). \tag{7.7}$$

For each i grill we introduce poloidal mesh

$$\theta_{ij} = \theta_i - 0.5\triangle\theta_i + (j-1)\triangle\theta_i/(nthin(i)-1)), j = 1,...,nthin(i). \tag{7.8}$$

The codes has several possibilities to set the initial refractive index. The index **i_n_poloidal=1,2,3,4** (see latter) chooses the type of the refractive index setting.

7.2.3 Power spectrum.

Power spectrum over parallel refractive index.

For the cases **i_n_poloidal=1,2,3 o**ne of the input variables is the parallel refractive index $N_\parallel$. Hear we describe the used forms of the power spectrum over the parallel refractive index $P_N(N_\parallel)$. For one i grill, the parallel component of refractive vector $N_\parallel$ is in the interval $(N_{\parallel_{i_{min}}}, N_{\parallel_{i_{max}}})$, where $N_{\parallel_{i_{min}}} = anmin(i), N_{\parallel_{i_{max}}} = anmax(i)$. We use two different forms of power spectrum:

if **igrillpw=1**

$$P_{Ni}(N_\parallel) = p_{Ni.} \tag{7.9}$$

if **igrillpw=2**

$$P_{Ni}(N_\parallel) = p_{Ni}\left(\frac{\sin(x)}{x}\right)^2, x = \frac{2\pi}{\triangle N_{\parallel_i}}(N_\parallel - N_{\parallel_{0_i}}). \tag{7.10}$$

Here, $\triangle N_{\parallel_i} = N_{\parallel_{i_{max}}} - N_{\parallel_{i_{min}}}$, $N_{\parallel_{0_i}} = 0.5(N_{\parallel_{i_{max}}} + N_{\parallel_{i_{min}}})$ is the center value of $N_\parallel$. The power spectrum 7.9 and 7.10 are normalized at unit

$$\frac{1}{\triangle N_{\parallel_i}} \int_{N_\parallel = N_{\parallel_{i_{min}}}}^{N_\parallel = N_{\parallel_{i_{max}}}} P_{Ni}(N_\parallel) dN_\parallel = 1. \tag{7.11}$$

For each i grill we introduce uniform $N_\parallel$ mesh:

$$N_{\parallel_{i,n}} = anmin(i) + nh_{N_{\parallel_i}}, n = 1, ..., nnkpar(i) \tag{7.12}$$

Where $h_{N_{\parallel_i}} = (anmax(i) - anmin(i))/(nnkpar(i) + 1)$ is the step of this mesh.

if **igrillpw=3**

$$P_{Ni}(N_\parallel) = p_{Ni} exp((\frac{N_\parallel - anmin(i)}{anmax(i)})^2). \tag{7.13}$$

This power spectrum is normalized by unit using eq.7.11.

Power spectrum over toroidal and poloidal refractive indexes.

For the cases **i_n_poloidal=4** the refractive index $\overrightarrow{N}$ is set by its toroidal and poloidal components N_ϕ, N_θ. Hear we describe the used forms of the power spectrum over the toroidal and poloidal refractive index components $P_N(N_\phi, N_\theta)$. For one i grill, the toroidal and poloidal refractive vector components are in the intervals $N_\phi \subset (N_{\phi_{i_{min}}}, N_{\phi_{i_{max}}})$,$N_\theta \subset (N_{\theta_{i_{min}}}, N_{\theta_{i_{max}}})$, where $N_{\phi_{i_{min}}} = antormin(i), N_{\phi_{imax}} = antormax(i)$ and $N_{\theta_{imin}} = anpolmin(i), N_{\theta_{imax}} = anpolmax(i)$. We use two different forms of power spectrum:

if **igrillpw=1**

$$P_{Ni}(N_\phi, N_\theta) = p_{Ni.} \tag{7.14}$$

if **igrillpw=2**

$$P_{Ni}(N_\phi, N_\theta) = p_{Ni} f_\phi\left(N_\phi\right) f_\theta\left(N_{\theta\phi}\right) \tag{7.15}$$

$$f_\phi(N_\phi) = \left(\frac{\sin(x)}{x}\right)^2, x = \frac{2\pi}{\triangle N_{\phi_i}}(N_\phi - N_{\phi_{0_i}}) \tag{7.16}$$

$$f_\theta(N_\theta) = \left(\frac{\sin(y)}{y}\right)^2, y = \frac{2\pi}{\triangle N_{\theta_i}}(N_\theta - N_{\theta_{0_i}}) \tag{7.17}$$

Here $\triangle N_{\phi_i} = N_{\phi_{i_{max}}} - N_{\phi_{i_{min}}}$ and $\triangle N_{\theta_i} = N_{\theta_{i_{max}}} - N_{\theta_{i_{min}}}$, $N_{\phi_{0_i}} = 0.5(N_{\phi_{i_{max}}} + N_{\phi_{i_{min}}})$ and $N_{\theta_{0_i}} = 0.5(N_{\theta_{i_{max}}} + N_{\theta_{i_{min}}})$ are the center values of N_ϕ and N_θ . The power spectrum 7.14, 7.15, 7.16 and 7.17 are normalized at unit

$$\frac{1}{\triangle N_{\phi_i}\triangle N_{\theta_i}} \int_{N_{\phi_{i_{min}}}}^{N_{\phi_{i_{max}}}} \int_{N_{\theta_{i_{min}}}}^{N_{\theta_{i_{max}}}} P_{Ni}(N_\phi, N_\theta) dN_\phi dN_\theta = 1. \tag{7.18}$$

For each i grill we introduce uniform (N_ϕ, N_θ) mesh:

$$N_{\phi_{i,ntor}} = antormin(i) + ntor * h_{N_{\phi_i}}, ntor = 1, ..., nnktor(i) \tag{7.19}$$

$$N_{\theta_{i,npol}} = anpolmin(i) + npol * h_{N_{\theta_i}}, npol = 1, ..., nnkpol(i) \tag{7.20}$$

Where $h_{N_{\phi_i}} = (antormax(i) - antormin(i))/(nnktor(i)+1)$ and $h_{N_{\theta_i}} = (anpolmax(i) - anpolmin(i))/(nnkpol(i)+1)$ are the steps of this mesh.

if **igrillpw=3**

$$P_{Ni}(N_\phi, N_\theta) = p_{Ni} exp(-(\frac{N_\phi - antormin(i)}{antormax(i)})^2 - (\frac{N_\theta - anpolmin(i)}{anpolmax(i)})^2) \tag{7.21}$$

This power spectrum is normalized by unit using eq. 7.18.

7.2.4 Approximation of one *i* grill

Each i grill is divided into **nthin**(i) cells with poloidal length equal to $\delta l_{ij} = \rho_i \triangle\theta_i/(nthin(i) - 1))$. The centers of j cells are at points with coordinates $(\rho_i, \theta_{ij}, \varphi_i)$ where θ_{ij} is given by 7.8. The local power in j cell is:

$$P_{\theta ij} = p_{\theta i} \cos^2(\pi(\theta_{ij} - \theta_i)/\triangle\theta_i)). \tag{7.22}$$

Amplitude $p_{\theta i}$ is calculated from normalization condition 7.7. For **nthin(i)=1** , we use $\theta_{i1} = \theta_i$, $P_{i1} = powers(i)$.

For i_n_poloidal=1,2,3 cases. From all cell centers, we launch **nnkpar**(i) rays . The ray with number $n = 1, ..., nnkpar(i)$ has parallel refractive index $N_{\|_{in}}$ 7.12 and spectrum power $P_{Ni}(N_{\|_{i,n}})$ from 7.9 or 7.10. For **nnkpar**(i)=1, we use $N_{\|_{i,1}} = N_{\|_{0_i}} = 0.5(anmax(i) + anmin(i))$ and $P_{Ni}(N_{\|_{i1}}) = powers(i)$. The partial power in the ray with indexes (i, j, n) is $P_{ijn} = P_{\theta i}(\theta_{ij}) P_{Ni}(N_{\|_{in}})$.

The number of rays per i grill is equal to **nthin**(i)***nnkpar**(i) . The total number of rays is $i_{ray} = \sum_{\mathbf{i}=1}^{i=ngrill} nthin(i) nnkpar(i)$**.**

So, in each space (i, j) point, the code calculates coordinates $(R_0.\varphi_0, z_0)$ of the initial ray location. At this point for each (i, j, n) ray, the code determines parallel to magnetic field component of the refractive index $N_{\|_{i,n}}$

For i_n_poloidal=4 case. From all cell centers, we launch **nnktor**(i)***nnkpol**(i) rays . The ray with numbers $ntor = 1,...,nnktor(i)$ and $npol = 1,...,nnkpol(i)$ has the toroidal and poloidal refractive index components $(N_{\phi_{i,ntor}}, N_{\theta_{i,npol}})$ 7.19, 7.20 and spectrum power $P_{Ni}(N_{\phi_{i,ntor}} N_{\theta_{i,npol}})$ from 7.14 or 7.15. For **nnktor**(i)=1, we use $N_{\phi_{i,1}} = N_{\phi_{0_i}} = 0.5(antormax(i) + antormin(i))$, for **nnkpol**(i)=1, we use $N_{\theta_{i,1}} = N_{\theta_{0_i}} = 0.5(anpolmax(i) + anpolmin(i))$. If the spectrum consists from one point **nnktor**(i)=1 and **nnkpol**(i)=1 then $P_{Ni}(N_{\phi_{i,1}}, N_{\theta_{i,1}}) = powers(i)$. The partial power in the ray with indexes $(i, j, ntor, npol)$ is $P_{i,j,ntor,npol} = P_{\theta i}(\theta_{ij}) P_{Ni}(N_{\phi_{i,ntor}}, N_{\theta_{i,npol}})$.

The number of rays per i grill is equal to **nthin**(i)***nnktor**(i) ***nnkpol**(i). The total number of rays is $i_{ray} = \sum_{\mathbf{i}=1}^{i=ngrill} ntin(i) nnktor(i) nnrpol(i)$**.**

So, in each space (i, j) point, the code calculates coordinates $(R_0.\varphi_0, z_0)$ of the initial ray location. At this point for each $(i, j, ntor, npol)$ ray, the code determines the toroidal and poloidal refractive index components $N_{\phi_{i,ntor}}$ and $N_{\theta_{i,npol}}$

There are the several possibilities to set the direction of the perpendicular refractive index component in the code.

7.2.5 The given N_parallel determines N_toroidal and N_poloidal.

i_n_poloidal=1.
Using the given value of $N_{\|_{i,n}}$ the code calculates the toroidal $N_{\|\varphi}$ and poloidal $N_{\|\theta}$ components of the refractive vector. Then it solves the dispersion relation $N_\perp = N_\perp(N\|)$ or $N = N(N\|)$. After that the code directs the perpendicular component of the refractive index along the perpendicular to the magnetic surface along $\pm\nabla\psi$.

$$N_\rho = \sigma\sqrt{N^2 - N_\|^2} = \sigma\sqrt{N^2 - N_{\|\phi}^2 - N_{\|\theta}^2} \tag{7.23}$$

Here $\sigma = \pm 1$. Here $\sigma = \pm 1$. The variable **σ=i_ vgr_ini** is set in **genray.dat** file in namelist **/wave/**. To choose the different possibilities of the grill type wave launch the code has the switch **i_n_poloidal** that is determined in **genray.dat** file in **grill** namelist**.** For the given case **i_n_poloidal=1.** By default **i_n_poloidal=1.**

7.2.6 The given N_parallel and N_poloidal.

i_n_poloidal=2.
The other possibility of the direction of the perpendicular refractive index is to set parallel $N_{\|_{i,n}}$ and poloidal N_θ refractive indexes **i_n_poloidal=2** In the given case the **grill** namelist has the input variable N_θ**=n_poloidal**. By default **n_poloidal**=0.

The initial refractive vector can be written as

$$\overrightarrow{N}_{i,n} = N_{\|_{i,n}} \frac{\overrightarrow{B}}{B} + N_\perp \overrightarrow{e}_\perp \tag{7.24}$$

Here $\overrightarrow{e}_\perp$ is a unit vector that determines the direction of the perpendicular refractive index

$$\overrightarrow{e}_\perp = \alpha_{\perp\phi} \overrightarrow{e}_\phi + \alpha_{\perp\theta} \overrightarrow{e}_\theta + \alpha_{\perp\rho} \overrightarrow{e}_\psi \tag{7.25}$$

The coordinates of $\overrightarrow{e}_{\perp}$ vector can be determined from the following equations

$$\alpha_{\perp\phi}^2 + \alpha_{\perp\theta}^2 + \alpha_{\perp\rho}^2 = 1 \tag{7.26}$$

$$B_\phi \alpha_{\perp\phi} + B_p \alpha_{\perp\theta} = 0 \tag{7.27}$$

From 7.26, 7.27

$$\alpha_{\perp\phi} = -B_p \alpha_{\perp\theta}/B_\phi \tag{7.28}$$

$$\alpha_{\perp\rho}^2 = 1 - \alpha_{\perp\phi}^2 - \alpha_{\perp\theta}^2 = 1 - \alpha_{\perp\theta}^2 \frac{B^2}{B_\phi^2} \tag{7.29}$$

The eq.7.29 have sense at

$$|\alpha_{\perp\theta}| < \frac{B_\phi}{B} \tag{7.30}$$

Using 7.28, 7.29 we can rewrite 7.25

$$\overrightarrow{e}_{\perp} = -B_p \alpha_{\perp\theta}/B_\phi \overrightarrow{e}_\phi + \alpha_{\perp\theta} \overrightarrow{e}_\theta + \sigma \sqrt{1 - \alpha_{\perp\theta}^2 \frac{B^2}{B_\phi^2}}\, \overrightarrow{e}_\psi \tag{7.31}$$

Here $\sigma = \pm 1$. From 7.24 and 7.31

$$\begin{aligned} \overrightarrow{N}_{i,n} &= N_{\parallel_{i,n}} (\overrightarrow{e}_\phi \tfrac{B_\phi}{B} + \overrightarrow{e}_\theta \tfrac{B_p}{B}) + N_\perp (-B_p \alpha_{\perp\theta}/B_\phi \overrightarrow{e}_\phi + \alpha_{\perp\theta} \overrightarrow{e}_\theta + \sigma \sqrt{1 - \alpha_{\perp\theta}^2 \tfrac{B^2}{B_\phi^2}}\, \overrightarrow{e}_\psi) = \\ &= \overrightarrow{e}_\phi (N_{\parallel_{i,n}} \tfrac{B_\phi}{B} - N_\perp B_p \alpha_{\perp\theta}/B_\phi) + \overrightarrow{e}_\theta (N_{\parallel_{i,n}} \tfrac{B_p}{B} + N_\perp \alpha_{\perp\theta}) + \overrightarrow{e}_\psi \sigma \sqrt{1 - \alpha_{\perp\theta}^2 \tfrac{B^2}{B_\phi^2}} \end{aligned} \tag{7.32}$$

Eq.7.32 gives the poloidal component of the initial refractive index $N_\theta = N_\parallel \frac{B_p}{B} + N_\perp \alpha_{\perp\theta}$. Here the poloidal refractive index N_θ is the part of the projection of the refractive index on the. So

$$\alpha_{\perp\theta} = \frac{N_\theta}{N_\perp} - \frac{N_\parallel}{N_\perp} \frac{B_p}{B} \tag{7.33}$$

Using the condition 7.30 we can get the condition for N_θ

$$\frac{N_\parallel B_p}{B} - \frac{N_\perp B_\phi}{B} < N_\theta < \frac{N_\parallel B_p}{B} + \frac{N_\perp B_\phi}{B} \tag{7.34}$$

7.2.7 The given N_parallel and the angle between N_perpendicular and $\nabla\psi$.

The other possibility of the direction of the perpendicular refractive index is to set the parallel refractive index $N_{\parallel_{i,n}}$ and the angle χ between the perpendicular refractive index $\overrightarrow{N}_{\perp}$and the gradient from the poloidal flux $\widehat{\chi} = \widehat{\overrightarrow{N}_{\perp} \nabla\psi}$, $0 \le \chi \le \pi$. For the given case **i_n_poloidal=3.** In this given case the **grill** namelist has the input parameter **ksi_nperp**. By default **ksi_nperp**=0.

The initial refractive vector is determined be the equations 7.24, 7.25, 7.26 and 7.28. Where

$$\alpha_{\perp\rho} = \cos\chi \tag{7.35}$$

Using 7.35 in 7.26,7.28 we will get

$$\alpha_{\perp\theta} = \sin\chi\frac{B_\phi}{B} \tag{7.36}$$

$$\alpha_{\perp\phi} = -\sin\chi\frac{B_p}{B} \tag{7.37}$$

Using 7.36, 7.37 we can rewrite 7.25

$$\overrightarrow{e}_\perp = -\frac{B_p}{B}\sin\chi\overrightarrow{e}_\phi + \frac{B_\phi}{B}\sin\chi\overrightarrow{e}_\theta + \cos\chi\overrightarrow{e}_\psi \tag{7.38}$$

It gives

$$\begin{aligned}\overrightarrow{N}_{i,n} &= N_{\|_{i,n}}(\overrightarrow{e}_\phi\frac{B_\phi}{B} + \overrightarrow{e}_\theta\frac{B_p}{B}) + N_\perp(-\frac{B_p}{B}\sin\chi\overrightarrow{e}_\phi + \frac{B_\phi}{B}\sin\chi\overrightarrow{e}_\theta + \cos\chi\overrightarrow{e}_\psi) = \\ &= \overrightarrow{e}_\phi(N_{\|_{i,n}}\frac{B_\phi}{B} - N_\perp\frac{B_p}{B}\sin\chi) + \overrightarrow{e}_\theta(N_{\|_{i,n}}\frac{B_p}{B} + N_\perp\frac{B_\phi}{B}\sin\chi) + \overrightarrow{e}_\psi N_\perp\cos\chi\end{aligned} \tag{7.39}$$

7.2.8 The given toroidal and poloidal refractive index components.

i_n_poloidal=4. The other possibility to set the refractive index is to give its toroidal and poloidal components N_ϕ and N_ϕ. In this case the refractive index is

$$\overrightarrow{N} = N_\phi\overrightarrow{e}_\phi + N_\theta\overrightarrow{e}_\theta + N_\rho\overrightarrow{e}_\psi \tag{7.40}$$

We will find the parallel refractive index for the given case. The vector of the parallel refractive index component is

$$\overrightarrow{N}_\| = N_\|(\overrightarrow{e}_\phi\frac{B_\phi}{B} + \overrightarrow{e}_\theta\frac{B_\theta}{B}) \tag{7.41}$$

The vector of the perpendicular refractive index is

$$\overrightarrow{N}_\perp = N_\perp\sin\chi(\overrightarrow{e}_\phi\sin\xi + \overrightarrow{e}_\theta\cos\xi) + N_\perp\cos\chi\overrightarrow{e}_\psi \tag{7.42}$$

Here χ is the angle between the perpendicular refractive index $\overrightarrow{N}_\perp$and the poloidal flux gradient $\widehat{\chi} = \widehat{\overrightarrow{N}_\perp\nabla\psi}$, $0 \le \chi \le \pi$, ξ is the angle between the projection of the perpendicular refractive index on the plane perpendicular the $\nabla\psi$ and the poloidal unit vector $\overrightarrow{e}_\theta$. Using the condition $0 = \overrightarrow{N}_\perp\cdot\overrightarrow{N}_\| = \sin\xi\frac{B_\phi}{B} + \cos\xi\frac{B_\theta}{B}$ we have $\tan\xi = -\frac{B_\theta}{B_\phi}$, so $\sin\xi = \pm\frac{B_\theta}{B}$ and $\cos\xi = \mp\frac{B_\phi}{B}$. The equation 7.42 will give

$$\overrightarrow{N}_\perp = \pm N_\perp\sin\chi(\overrightarrow{e}_\phi\frac{B_\theta}{B} - \overrightarrow{e}_\theta\frac{B_\phi}{B}) + N_\perp\cos\chi\overrightarrow{e}_\psi \tag{7.43}$$

Using 7.40, 7.41 and 7.43 we will have

$$N_\rho = N_\perp \cos\chi \tag{7.44}$$

$$N_\phi = \pm N_\perp \sin\chi \frac{B_\theta}{B} + N_\parallel \frac{B_\phi}{B} \tag{7.45}$$

$$N_\theta = \mp N_\perp \sin\chi \frac{B_\phi}{B} + N_\parallel \frac{B_\theta}{B} \tag{7.46}$$

Equations 7.45 and 7.46 determine the parallel refractive index

$$N_\parallel = N_\phi \frac{B_\phi}{B} + N_\theta \frac{B_\theta}{B} \tag{7.47}$$

and

$$\sin\chi = \pm \frac{N_\phi B_\theta - N_\theta B_\phi}{B N_\perp} \tag{7.48}$$

From the condition $|\sin\chi| \leq 1$ equation 7.48 gives the condition for N_ϕ and N_θ. Equations 7.44, 7.48 determine the small radial refractive index component

$$N_\rho = \sigma\sqrt{1 - sin^2\chi} \tag{7.49}$$

Here $\sigma = \pm 1$. The variable **σ=i_ vgr_ini** is set in **genray.dat** file in namelist /wave/.

7.2.9 The determination of cutoff LH and FW point near the plasma edge.

In some cases it is necessary to launch LH or FW in the cutoff point most close to the plasma edge. GENRAY has the possibility to shift the given launch point to the cutoff point near the plasma edge. If the variable **i_rho_cutoff=1** (in **GENRAY.dat** file) then the the code will find the cutoff point. By default **i_rho_cutoff=0.**
From the input grill data the code determines the launch point $(R_0.\varphi_0, z_0)$.The code can shift this point to the cutoff point near the plasma edge along the straight line. This line is directed from the magnetic axis to the given point $(R_0.\varphi_0, z_0)$.

7.3 EC wave launch for O-X-B mode conversion scenarios

istart=3

For dense plasma with $X_e(\rho = 0) > 1$ 3.1, the electron cyclotron O (ordinary) mode launched from the periphery where $X_e(\rho) < 1$, reflects at the points where $X_e = 1$. In this case O mode cannot reach the center of the plasma. A way to overcome the density limit is the O-X-EBW mode conversion processes proposed by (J. Preinhaelter and V.Kopecky, 1973) [11]. X and EBW refer to the extraordinary and the electron Bernstein modes. These authors proposed the scenario that O mode be transformed to

X mode at the point where $X_e = 1$. The O-X mode conversion requires the launch of O mode with optimal parallel refractive index:

$$N_{\parallel_{opt}} = \sqrt{Y_e(P_{conv})/Y_e(P_{conv}) + 1.} \tag{7.50}$$

Here,

$P_{conv} = (\rho_{conv}, \theta_{conv})$ is O-X mode conversion point, where

$$X_e(\rho_{conv}, \theta_{conv}) = 1; \tag{7.51}$$

$(\rho_{conv}, \theta_{conv})$ are the small radius and the poloidal angle of O-X mode conversion point P_{conv}.

O-mode is transformed to X mode near O-X conversion point P_{conv}. The X-mode enters the plasma with greater density, $X_e > 1$. Then X-mode is reflected and goes towards the plasma periphery. If the upper hybrid resonance (UHR) layer (where $X_e + Y_e^2 = 1$) is inside the plasma , the X-mode is transformed into the electrostatic electron Bernstein mode. The EBW mode is reflected near the UHR layer and can propagate into the plasma core. The EBW mode is then absorbed by the plasma near the electron cyclotron resonance (ECR) point. This scenario enables a wave launched in the O-mode to heat the central part of an over-dense ($X_e > 1$) plasma.

This special X-mode launch has been implemented for studying O-X-EBW mode conversion scenarios (C.B. Forest et al., 2000) [12]. The X-mode is launched from a space point inside the plasma. To specify the location of this point we use input data from the grill namelist parameters, θ_{conv}**=thgrill**(1).

First, we calculate the small radius ρ_{conv} of O-X conversion point P_{conv} for the given poloidal angle θ_{conv} using equation 7.51. In the point found, the X-mode wave is launched just inside the $X_e = 1$-layer in the direction of denser plasma with the optimal value of the parallel refractive index $N_\parallel = N_{\parallel_{conv}}$7.50. To launch this wave, we determine the perpendicular component of refractive vector $N_{\perp X}$ for X-mode from the solution of the dispersion relation:

$$D_{hot}(\widehat{X}(\rho_{conv}), \widehat{Y}(\rho_{conv}), \widehat{T}(\rho_{conv}), N_{\parallel_{opt}}, N_{\perp X}) = 0. \tag{7.52}$$

For these conditions, $N_{\perp X}$ should be close to zero.

The code calculates the input grill parameters **rhopsi0**(1)=ρ_{conv} **anmin**(1)=$N_{\parallel_{opt}} -$ 0.01 and **anmax(1)=** $N_{\parallel_{opt}} + 0.01$. For this case.\, the code uses one ray launch. So we see in **genray.dat** file: **ngrill=1**, **nthin(1)=1**, **nnkpar(1)=1**. The parameters **rhopsi0(1), height(1) anmin(1), anmin(1)** in **genray.dat** file can be arbitrary; they will not be used.

7.4 Initial refractive index vector inside the plasma

The refractive vector $\overrightarrow{N}_0$ at the initial space point P_0 consists of the vector parallel to the flux surface $\overrightarrow{N}_{0_{\parallel_\psi}}$ ($\overrightarrow{N}_{0_{\parallel_\psi}} \cdot \nabla\psi = 0$) and the vector perpendicular to flux surface $\overrightarrow{N}_{0_{\perp_\psi}}$:

$$\overrightarrow{N}_0 = \overrightarrow{N}_{0_{\parallel_\psi}} + \overrightarrow{N}_{0_{\perp_\psi}}. \tag{7.53}$$

At the plasma-vacuum boundary Γ , the parallel refractive index $\overrightarrow{N}_{0_{\|\psi}}$ should be the same for both the vacuum and plasma sides. We determine the parallel refractive index (that is, in the flux surface) at the vacuum side of boundary :Γ

$$\overrightarrow{N}_{0_{\|\psi_\Gamma}} = N_{0_\varphi}\overrightarrow{e}_\varphi + N_{0_\theta}\overrightarrow{e}_\theta. \tag{7.54}$$

Here, N_{0_φ} is the toroidal component of the refractive index at the initial point; N_{0_θ} is the poloidal component of the refractive index at the initial point; $\overrightarrow{e}_\varphi$ is the unit vector in the toroidal direction; $\overrightarrow{e}_\theta = \nabla\psi \times \overrightarrow{e}_\varphi/|\nabla\psi|$ is the unit vector in the poloidal direction. The perpendicular to the magnetic surface refractive index vector directed into the plasma is equal to:

$$\overrightarrow{N}_{0_{\perp\psi}} = -N_{0_{\perp_\psi}}\nabla\psi/|\nabla\psi|. \tag{7.55}$$

The poloidal flux function has been assumed to be an increasing function from the magnetic axis, the usual convention in the EQDSK files. Here, $N_{0_{\perp_\psi}}$ is the small radial component of the refractive index that can be calculated from the dispersion relation for given $N_{0_\|}$ and N_{0_φ}, N_{0_θ}.

The different models of the wave launch described in the previous sections 7.1, 7.2, 7.3 create the initial toroidal N_{0_φ} and poloidal N_{0_θ} components of the refractive index. Using these components the code calculates the parallel to the magnetic field refractive index 7.54.

Then the code solves the dispersion relation to find the perpendicular to the magnetic field refractive index $N_{0_\perp}$. The dispersion relation $D(N_\|, N_\perp) = 0$ has a set of roots $N_{\perp_k}$. Each root corresponds to a specified wave mode. **Genray.dat** file input has two switches for root choice.

ioxm=$\pm$1 determines the sign before the root in the cold plasma dispersion relation 4.12, and O-mode (**ioxm=+1**) or X-mode (**ioxm=-1**) for the solution of the Mazzucato dispersion relation 4.37.

ibw specifies the launch of EBW. If **ibw=1, the** code launches a Bernstein wave using the hot plasma dispersion relation and grill conditions **istart=2**. If **ibw=0,** the code launches O or X modes.

Using the found root $N_{0_\perp}$ the code calculates the module of the small radial component of the refractive vector directed perpendicular to the magnetic surface

$$|\overrightarrow{N}_{0_{\perp\psi}}| = \sqrt{N_{0_\perp}^2 + N_{0_\|}^2 - N_{0_\varphi}^2 - N_{0_\theta}^2}$$

The code has the possibility to launch the wave towards the interior of the plasma or in the outwards direction. The direction of the small radial component of the wave group velocity $\overrightarrow{v}_{gr_\psi}$ is determined by the direction of the small radial refractive vector $\overrightarrow{N}_{0_{\perp\psi}}$. The code chooses the sign of $\overrightarrow{N}_{0_{\perp\psi}}$ to get the necessary direction of $\overrightarrow{v}_{gr_\psi}$ .The input parameter **i_vgr_ini** given in **genray.dat** determines the direction of $\overrightarrow{v}_{gr_\psi}$ in the initial point. For the wave directed into the plasma **i_vgr_ini**=+1 and $\overrightarrow{v}_{gr_\psi}$ is anti-parallel to $\nabla\psi$. For **i_vgr_ini**=-1 the small radial wave group velocity directed out of plasma. It is assumed that the poloidal flux has a minimum inside the plasma.

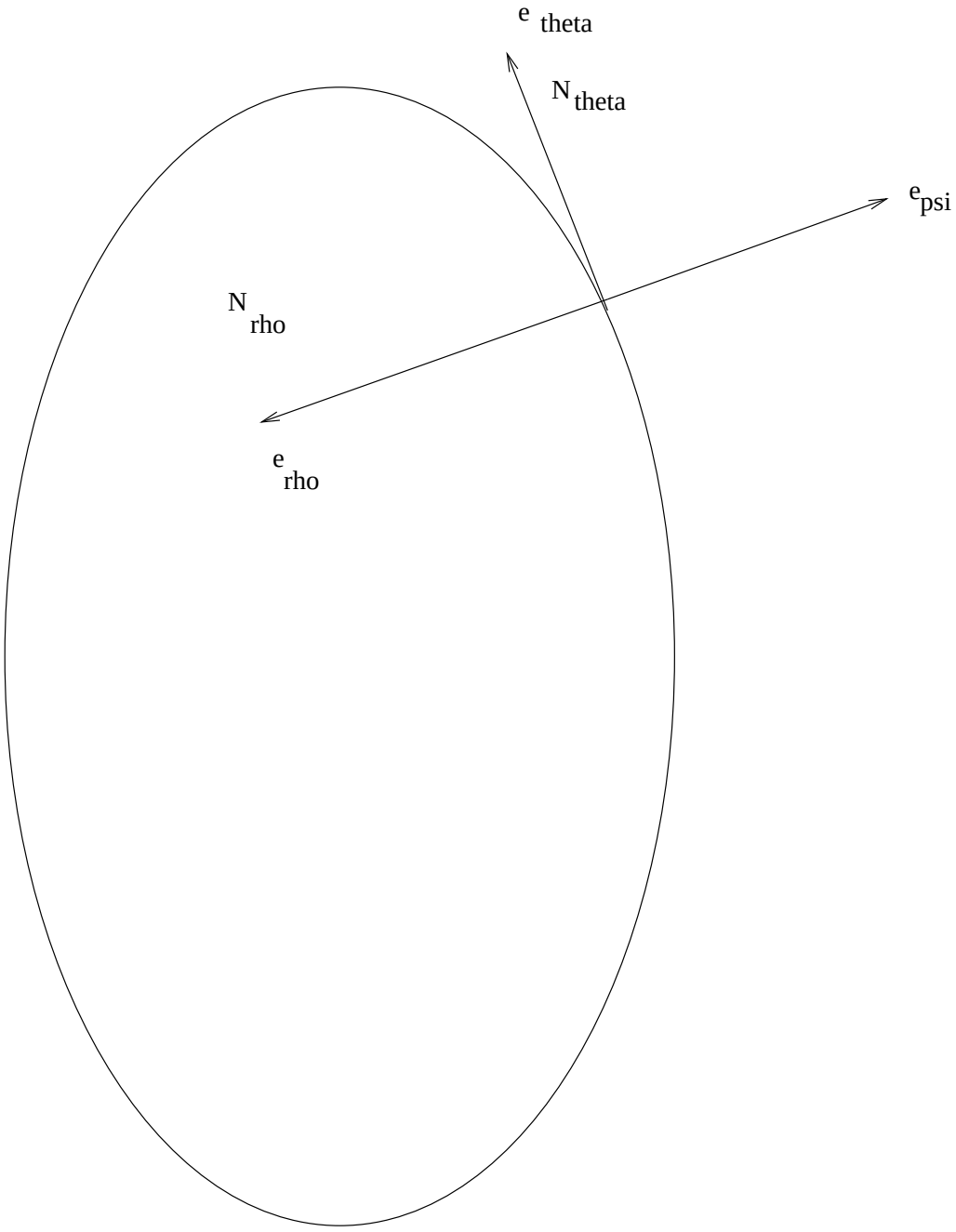


Figure 7.1: The refractive index components in the poloidal pane.

7.5 Reflection from the plasma vacuum boundary

Near the plasma vacuum boundary, the code can reflect the rays. The variable **ireflm** determines the maximum number of reflections for each ray. It is proposed that plasma boundary Γ can be represented by two functions versus major radius $z_{min}(r)$, $z_{max}(r)$, where $r_{max} < r < r_{min}$. If the point of the ray trajectory $P_{reflect}(r,z)$ goes close to the plasma boundary:

$$\begin{array}{lll} r < r_{min} + epsbnd & or & r_{max} - epsbnd < r \\ z < z_{min} + epsbnd & or & z_{max} - epsbnd < z \end{array}, \tag{7.56}$$

the subroutine **bound** reflects the ray at this point. Here *epsbdn* is a parameter given inside subroutine **bound.** At the reflection point $P_{reflect}(r,z)$, the code changes the sign of the normal to the flux surface component of the refractive index,

$$\overrightarrow{N}_{reflect_{\perp\psi}} = N_{reflect_{\perp}} \nabla\psi / |\nabla\psi|, \tag{7.57}$$

and conserves the parallel to the flux surface refractive index component,

$$\overrightarrow{N}_{reflect_{\parallel\psi}} = N_{reflect_{\varphi}} \overrightarrow{e}_{\varphi} + N_{reflect_{\theta}} \overrightarrow{e}_{\theta}. \tag{7.58}$$

The refractive vector before reflection has the form:

$$\overrightarrow{N}_{reflect} = \overrightarrow{N}_{reflect_{\parallel\psi}} + \overrightarrow{N}_{reflect_{\perp\psi}}. \tag{7.59}$$

The refractive vector after reflection has the form:

$$\overrightarrow{N}_{reflect} = \overrightarrow{N}_{reflect_{\parallel\psi}} - \overrightarrow{N}_{reflect_{\perp\psi}}. \tag{7.60}$$

If reflection is not needed, we set the parameter **ireflm** equal to unit (**ireflm=1**).

Chapter 8

Electric field polarization

The code calculates electric field polarization along ray trajectories. The electric field $\overrightarrow{E}$ is the unit vector which satisfies equations 1.6 for different forms of the dielectric tensor. The condition for this solution is the dispersion relation $D = 0$ 1.7. For any given computational accuracy, the numerical solution of the ray-tracing equations1.8 gives the numerical error for the ray variables. This error leads to non-zero dispersion function $D \neq 0$ along the ray trajectory. So, formally the system 1.6 will have no solution. Therefore, we must determine the polarization for equations 1.8 with non-zero dispersion function. We use the following approach:

$$\begin{array}{c} d_{11}E_1 + d_{12}E_2 + d_{13}E_3 = 0 \\ d_{21}E_1 + d_{22}E_2 + d_{23}E_3 = 0. \\ d_{31}E_1 + d_{32}E_2 + d_{33}E_3 = 0 \end{array} \tag{8.1}$$

System 8.1 determinant is $D = \sum_{i=1,3} d_{ji}a_{ji}$. Where j is the number of chosen matrix row and a_{ji} is a cofactor of the element d_{ji}. The solution of system 8.1 has the following form:

$$E_k = a_{jk}. \tag{8.2}$$

To check it, we put 8.2 into 8.1

$$\begin{array}{c} d_{11}a_{j1} + d_{12}a_{j2} + d_{13}a_{j3} = 0 \\ d_{21}a_{j1} + d_{22}a_{j2} + d_{23}a_{j3} = 0. \\ d_{31}a_{j1} + d_{32}a_{j2} + d_{33}a_{j3} = 0 \end{array} \tag{8.3}$$

The condition of the solution is $D = 0$. Let p be the number of the row in the system. The equation 8.3 has the form:

$$\sum_{i=1,3} d_{pi}a_{ji} = 0. \tag{8.4}$$

If index j coincides with the number of row $j = p$, then,

$$\sum_{i=1,3} d_{pi}a_{pi} = D = 0. \tag{8.5}$$

If $j \neq p$, then $\sum_{i=1,3} d_{pi} a_{ji} = 0$ as the determinant of the matrix with two coincided rows (d_{p1}, d_{p2}, d_{p3}).

Due to numerical errors, determinant D does not equal zero. In this case we use equation 8.2 for the electric field determination. The number of matrix row j is chosen from condition:

$$a_{ji} = \max_{pk} |a_{pk}|. \tag{8.6}$$

The electric field is normalized to unit length.

Chapter 9

Current drive

The code calculates the current drive (CD) $j = \eta P_{absorp}$ (current density) along the ray trajectories using various possibilities to determine CD efficiency η and absorbed wave power P_{absorp}. CD efficiency η is normalized at $(A/cm^2)/(erg/(c*cm^3))$. The switch for choosing the CD efficiency form is **ieffic** . The specification of the wave harmonic is determined by variable **jwave**.

ieffic=1

CD efficiency is calculated using the following simplest asymptotic formula for uniform non-relativistic plasma.

jwave=1 for EC wave first harmonic;

$\eta_{EC} = \frac{3}{2(5+Z_{eff})}(u^2 + (2 + \frac{3}{2(3+Z_{eff})}))$;

jwave=0 for lower hybrid (LH) wave;

$\eta_{LH} = \frac{2}{(5+Z_{eff})}(u^2 + (\frac{7}{4} + \frac{9}{4(3+Z_{eff})}))$;

where u is the resonance velocity normalized by electron thermal velocity $v_{Te} = 1.87 * 10^7 \sqrt{T_e}$ (m/c), T_e (keV), $u = \frac{c}{v_{Te}}(1 - j_w Y_e)/N_\parallel$.

ieffic=2

In this case CD efficiency is calculated using the asymptotic formulas (D.A. Ehst and F.F. Karney, 1991) [13]

a) **jwave=0**

Landau damping of LH waves resonant at parallel velocities above the electron thermal velocity.

b) **jwave=-1**

Slow frequency fast (compressional Alfven) waves (AW) resonant with low phase velocity electrons via combined Landau damping and transit time magnetic damping.

ieffic=3

CD efficiency is calculated using the CURBA subroutine (R.H. Cohen, 1987) [14], [15] for relativistic electron plasma in toroidal plasma with passed and trapped particles.

Chapter 10

Power and current drive radius

The code can calculate the absorbed wave power and CD profiles versus the chosen small radius 2.10-2.13. To switch on the profile calculation, we use **ionetwo=1**. In other cases, **ionetwo=0**. The code calculates the absorbed power along all ray trajectories. It uses this power and the CD efficiency to calculate the current generated in the plasma. Using these data, the code creates profiles averaged over the magnetic surfaces of the absorbed power and power density, the current and current density.

Chapter 11

Numerical methods

11.1 Numerical solution of ray-tracing equations

The code has several possibilities for the numerical solution of ray tracing ODE equations. The switch **irkmeth** specifies the method.

irkmeth=0

This is the Runge-Kutta 4th order method with constant time step. The code controls the position of the trajectory near the plasma vacuum boundary and reduces the time step if the ray point goes outside the plasma 7.56 in one of the sub-steps of the Runge-Kutta procedure. This control is produced in **boundc** subroutine. The parameter *epsbnd* is set inside **boundc** subroutine . The value of this parameter can differ from the value *epsbnd* in **bound** subroutine.

irkmeth=1

This is the Runge-Kutta 5th order method with variable time step. In this case, the code changes the time step to get accuracy. It uses the algorithm described in (W.H.Press et al., 1999)[16] (chapter 16.2).

irkmeth=2

This is the Runge-Kutta 4th order method with variable time step. In this case, the code changes the time step to get accuracy. In this case, the code controls the position of the point near the plasma vacuum boundary as done in **irkmeth=0**.

11.2 Hamiltonian conservation

The ray tracing equations 1.8 are of the Hamiltonian type, where the dispersion function D plays the role of the Hamiltonian. Along the ray trajectory the dispersion function conserves zero value $D = 0$. Inaccuracy in the numerical integration of the ray tracing equations leads to errors in ray coordinates $\overrightarrow{q} = (\overrightarrow{R}, \overrightarrow{N})$. Small deviation in these coordinates can lead to significant deviation in the dispersion function from zero. The non-zero value of the dispersion function D, for example, can give a wrong electric field polarization. The code has two possibilities to conserve the Hamiltonian.

a) The Hamiltonian D has error ε at each time step of the numerical solution of ray-tracing system $D(\overrightarrow{q}) = \varepsilon$. The small correction $\triangle\overrightarrow{q}$ can conserve the zero Hamiltonian value

$$D(\overrightarrow{q} + \triangle\overrightarrow{q}) = 0. \tag{11.1}$$

There is one equation for six variables. The additional condition for the correction $\triangle\overrightarrow{q}$ can be the minimum of the norm $|\triangle\overrightarrow{q}|$. For this condition, the full problem for the correction has the following form:

$$\begin{cases} J = |\triangle\overrightarrow{q}|^2 = \sum_{i=1,6}(\triangle q_i)^2 \to \min \\ D(\overrightarrow{q} + \triangle\overrightarrow{q}) = 0 \end{cases}. \tag{11.2}$$

We use the Lagrange function $L = J(\triangle\overrightarrow{q}) + \lambda D(\overrightarrow{q} + \triangle\overrightarrow{q})$ for solution 11.2. The derivative from this function gives the equations for $\triangle\overrightarrow{q}$ and λ :

$$\begin{array}{c} 2\triangle q_i + \lambda\frac{\partial D(\overrightarrow{q}+\triangle\overrightarrow{q})}{\partial q_i} = 0,\ i = 1,...,6. \\ D(\overrightarrow{q} + \triangle\overrightarrow{q}) = 0 \end{array} \tag{11.3}$$

The first equation in 11.3 gives:

$$\triangle q_i = -\frac{\lambda}{2}\frac{\partial D(\overrightarrow{q} + \triangle\overrightarrow{q})}{\partial q_i},\ i = 1,...6. \tag{11.4}$$

After substitution 11.4 for the second equation 11.3 we have:

$$D(\overrightarrow{q} - \frac{\lambda}{2}\nabla D(\overrightarrow{q} + \triangle\overrightarrow{q})) = 0. \tag{11.5}$$

To solve 11.5 we use the iteration process based on its linearization:

$$D(\overrightarrow{q} - \frac{\lambda}{2}\nabla D(\overrightarrow{q} + \triangle\overrightarrow{q})) = D(\overrightarrow{q}) - \frac{\lambda}{2}\sum_{i=1,6}\frac{\partial D(\overrightarrow{q})}{\partial q_i}\frac{\partial D(\overrightarrow{q} + \triangle\overrightarrow{q})}{\partial q_i} = 0. \tag{11.6}$$

Let l be the number of iterations with the initial parameters $\overrightarrow{q}^{(0)} = \overrightarrow{q}$, $\triangle\overrightarrow{q}^{(0)} = 0$ and $\lambda^{(0)} = 0$. The iteration process ($l = 1,2,...$) has the following form:

$$\overrightarrow{q}^{(l)} = \overrightarrow{q} + \triangle\overrightarrow{q}^{(l-1)} \tag{11.7}$$

$$\lambda^{(l)} = \frac{2D(\overrightarrow{q})}{\sum_{i=1,6}\frac{\partial D(\overrightarrow{q})}{\partial q_i}\frac{\partial D(\overrightarrow{q}^{(l-1)})}{\partial q_i}} \tag{11.8}$$

$$\triangle q_i^{(l)} = -\frac{\lambda^{(l)}}{2}\frac{\partial D(\overrightarrow{q}^{(l-1)})}{\partial q_i} = -\frac{D(\overrightarrow{q})}{\sum_{i=1,6}\frac{\partial D(\overrightarrow{q})}{\partial q_i}\frac{\partial D(\overrightarrow{q}^{(l-1)})}{\partial q_i}}\frac{\partial D(\overrightarrow{q}^{(l-1)})}{\partial q_i}. \tag{11.9}$$

The condition for the final iteration is $|D(\overrightarrow{q}^{(l+1)})| < \varepsilon_{correction}$.

b) For large values of the perpendicular refractive index $N_\perp$, the previous method does not reach accuracy in some conditions. In this case, we use the correction procedure based on the solution of dispersion relation $D(N_\parallel, N_\parallel, X, Y, T) = 0$. After each

time step of the numerical solution of the ray-tracing equation, we get ray coordinates $\overrightarrow{q} = (\overrightarrow{R}, \overrightarrow{N})$. These coordinates give the values of the parallel and perpendicular refractive vectors $\overrightarrow{N}_{\|}$, $\overrightarrow{N}_{\perp}$. Using given $N_{\|}$, we calculate the corrected perpendicular refractive index $N_{\perp_{corrected}}$ from the solution of the dispersion relation. We correct only two components of refractive index $(N_{r_{corrected}}, N_{z_{corrected}})$. The toroidal component is not corrected. Then the corrected values of the refractive index coordinates $(N_{r_{corrected}}, N_{z_{corrected}})$ are calculated with the following system:

$$\begin{array}{c} B_r N_{r_{corrcted}} + B_z N_{z_{corrected}} + B_\varphi M/r = BN_{\|} \\ N^2_{r_{corrected}} + N^2_{z_{corrected}} + (M/r)^2 = N^2_{\|} + N^2_{\perp_{corrected}}. \end{array} \tag{11.10}$$

This system can give two solutions. We choose the solution that has minimum distance from the non-corrected values. For some conditions the system 11.10 has no solution. In this case the code gives a warning.

11.3 Differentiation of the dispersion function

The right hand side of ray-tracing equations 1.8 contains a set of the first derivatives from dispersion function D. The dispersion functions have many difficult formulas; therefore, the calculation of analytical derivatives from these formulas can be a source of error. Another problem is that in some cases the subroutine calculates the dispersion function but does not calculate derivatives from this function. The code can calculate these derivatives using the analytical (**idif=1**) or numerical (**idif=2**) differentiation. The switch **idif** is the input parameter in **genray.dat** file.

11.4 Algorithm giving the output data at the set radial steps

It is convenient to generate the output data from GENRAY at the poloidal distances close to the mesh points $s^i = ih_s$ specified by the poloidal step size h_s. The input step size **prmt6=** h_s is given in **genray.dat** file.

The code calculates the poloidal distance $l^n_{poloidal} = \sum_{i=1}^{n} \triangle l^n_{poloidal}$ along the trajectory at the each time t^n. The time step $\tau = t^n - t^{n-1}$ is automatically determined by the given Runge-Kutta method accuracy . Here $\triangle l^n_{poloidal} = \sqrt{(\triangle r^n)^2 + (\triangle z^n)^2}$, $\triangle r^n = r^n - r^{n-1}$ and $\triangle z^n = z^n - z^{n-1}$ are the increments of the poloidal distance and r, z variables for one time step. If $l^n_{poloidal} > s^i + \varepsilon_s h_s$ then the trajectory have jumped throw the necessary output mesh point s^i. Here $\varepsilon_s > 0$ is the given small positive number. In this situation GENRAY goes back to the previous time t^{n-1} point $(\overrightarrow{R}^{n-1}, \overrightarrow{N}^{n-1})$ takes the new time step

$$\tau_{new} = \tau(s^{i-1} + \varepsilon_s h_s l - l^{n-1}_{poloidal})/\triangle l^n_{poloidal} \tag{11.11}$$

and calculates the position of the new ray point $(\overrightarrow{R}^n, \overrightarrow{N}^n)$ using the Runge-Kutta method with the constant time step. The formula 11.11 was obtained from the linear

approximation of the poloidal distance dependence from the time. After that GENRAY calculates the new value of the poloidal distance that will be used as the output poloidal point. If the automatically calculated time step τ is much bigger than τ_{new}, the used algorithm can slow the calculations significantly.

GENRAY has the possibility to switch off writing the output dat to the 3d.dat file. The input variable **iout3d** that makes it is in **genray.dat** file. For **iout3d=**'enable' GENRAY writes **3d.dat** file, for **iout3d=**'disable' GENRAY does not write it.

Chapter 12

Electron cyclotron emission.

It works if the variable **i_emission=1** in **GENRAY.dat** file.

12.1 Radiation transport equation for the the radiative power.

Electron cyclotron emission can be a sensitive indicator of nonthermal electron distribution. GENRAY calculates the emission follow the model proposed in [6]. In the WKB approximation, the radiation transport equation [18] gives I, the power in each wave mode per unit area-radian frequency-steradian following along rays in the plasma:

$$n_r^2 \overrightarrow{s} \cdot \nabla(n_r^{-2} I) = j - \alpha I \tag{12.1}$$

Were $\widehat{s}$ is a unit vector along in the ray the ray trajectory mode under consideration, n_r is the ray refractive index [18], emissivity j is the power radiated by the the plasma per unit volume radian frequency-steradian, and α is the inverse damping length along the ray of rf wave energy. The ray emission trajectories can be calculated using the different types of the dispersion functions specified by the option **id**.
The formula for the ray refractive index is given in [18]

$$n_r^2 = \left| n^2 \sin\theta \frac{\left[1 + \left(\frac{1}{n}\frac{\partial n}{\partial \theta}\right)_\omega^2\right]^{1/2}}{\frac{\partial}{\partial \theta}\left\{\frac{\cos\theta + \left(\frac{1}{n}\frac{\partial n}{\partial \theta}\right)_\omega \sin\theta}{\left[1+\left(\frac{1}{n}\frac{\partial n}{\partial \theta}\right)_\omega^2\right]^{1/2}}\right\}}\right| \tag{12.2}$$

GENRAY has the several possibilities to calculate the ray refractive index n_r. The input variable **i_rrind** in **genray.day** file determines the used formula for n_r. For **i_rrind=1** $n_r = n$; for **i_rrind=2** the ray refractive index n_r will be calculated using 12.2 and the cold electron plasma dispersion function; **i_rrind=3** the ray refractive index n_r will be calculated using 12.2 and the hot non-relativistic plasma dispersion function.

The absorption and emission coefficients are given by Refs.[6], [19] and [20]:

$$\alpha = \frac{\omega}{4\pi}\frac{\overrightarrow{E}^* \cdot \widehat{\varepsilon} \cdot \overrightarrow{E}}{|S|} \tag{12.3}$$

$$j = \pi n_r^2 (\frac{\omega}{c})^2 \frac{\overrightarrow{E}^* \cdot \widehat{G} \cdot \overrightarrow{E}}{|S|} \tag{12.4}$$

Where $\overrightarrow{E} = \overrightarrow{E}(\omega, \overrightarrow{k})$ is the space-time Fourier transform of the electric field, $\overrightarrow{S}(\omega, \overrightarrow{k})$ is the energy flux density per frequency and per unit volume in $\overrightarrow{k}$ space,

$$\overrightarrow{S} = \overrightarrow{v}_g U \tag{12.5}$$

and $\overrightarrow{v}_{gr}$ is the wave group velocity 1.8. We use the conventions in Refs. [6], [19] and [20], which give spectral energy density U,

$$U = \frac{1}{8\pi}(\overrightarrow{B}^* \cdot \overrightarrow{B} + \overrightarrow{E}^* \cdot \frac{\partial(\omega\widehat{\varepsilon_h})}{\partial\omega} \cdot \overrightarrow{E}) \tag{12.6}$$

In this equation we use Hermitian dielectric tensor coefficients $\widehat{\varepsilon_h}$ for different dielectric tensors. It is essential that relativistic expressions be used for anti-Hermitian dielectric tensor elements $\widehat{\varepsilon_a}$ and the corresponding the correlation tensor **G** for the fluctuating current density. These are obtained from the momentum relativistic electron distribution function f [18], [6],[7]

$$\widehat{\varepsilon}_{relativist_a} = -\pi\frac{\omega_p^2}{\omega^2}\sum_{n=-\infty}^{n=\infty}\int d^3p U^{(n)}(f)\widehat{S}^{(n)}\delta(\gamma - \frac{k_\parallel v_\parallel}{\omega} - \frac{n\omega_{ce}}{\omega}), \tag{12.7}$$

$$\widehat{G} = -\frac{\pi}{(2\pi)^5}\frac{\omega_p^2}{\omega^2}\frac{1}{m_e}\sum_{n=-\infty}^{n=\infty}\int d^3p\frac{1}{\gamma}f p_\perp \widehat{S}^{(n)}\delta(\gamma - \frac{k_\parallel v_\parallel}{\omega} - \frac{n\omega_{ce}}{\omega}) \tag{12.8}$$

The formula 5.3 for anti-Hermitian part of the relativistic dielectric tensor were discussed previously in the chapter "Wave power absorption". The formula for the corresponding the correlation tensor **G** for the fluctuating current are used the same approach.

The emission equation along the ray trajectory s starting at the detector has the following form.

$$-n_r^2\frac{d}{s}(n_r^{-2}I) = j - \alpha I \tag{12.9}$$

Here $\widehat{s}$ is a unit vector in the opposite ray direction, $s = 0$ at the detector . The solution of the eq. 12.9 is

$$n_r^{-2}(s)I(s) = C_0\exp(\int_{s_0}^{s}\alpha(t)dt) - \int_{s_0}^{s}\frac{j(t)}{n_r^2(t)}dt\exp(\int_t^s\alpha(q)dq) \tag{12.10}$$

Let we will calculate the power of the emission along the ray with the length L and

$$I(L) = 0 \tag{12.11}$$

Using 12.11 we get from 12.10

$$0 = C_0 \exp(\int_{s_0}^{L} \alpha(t)dt) - \int_{s_0}^{L} \frac{j(t)}{n_r^2(t)} dt \exp(\int_{t}^{L} \alpha(q)dq) \tag{12.12}$$

It gives the value of the constant

$$C_0 = \exp(-\int_{s_0}^{L} \alpha(t)dt) \int_{s_0}^{L} \frac{j(t)}{n_r^2(t)} dt \exp(\int_{t}^{L} \alpha(q)dq) \tag{12.13}$$

Using 12.13 in 12.10 we get the emission power at the s_0 point

$$\begin{array}{c} n_r^{-2}(s_0)I(s_0) = C_0 = \exp(-\int_{s_0}^{L} \alpha(t)dt) \int_{s_0}^{L} \frac{j(t)}{n_r^2(t)} dt \exp(\int_{t}^{L} \alpha(q)dq) = \\ = \int_{s_0}^{L} \frac{j(t)}{n_r^2(t)} dt \exp(-\int_{s_0}^{t} \alpha(q)dq) \end{array} \tag{12.14}$$

So, the emission power at s_0 has the form

$$I(s_0) = n_r^2(s_0) \int_{s_0}^{L} \frac{j(t)}{n_r^2(t)} dt \exp(-\int_{s_0}^{t} \alpha(q)dq) \tag{12.15}$$

The emission at the detector $s_0 = 0$ is

$$I(0) = n_r^2(0) \int_{0}^{L} \frac{j(t)}{n_r^2(t)} dt \exp(-\int_{0}^{t} \alpha(q)dq) \tag{12.16}$$

Let introduce the mesh along the ray s_n, $n = 1, ..., N$, where the value s at the detector is $s_1 = 0$. The emission I_n from nth bin $s_n < s \leq s_{n+1}$ can be obtained from the equation

$$-n_r^2 \frac{d}{ds} \nabla(n_r^{-2} I_n) = j\Pi_n(s) - \alpha I_n \tag{12.17}$$

with the condition

$$I_n(s_{n+1}) = 0 \tag{12.18}$$

Here

$$\Pi_n(s) = \left\{ \begin{array}{c} 1, s_n < s \leq s_{n+1} \\ 0, elsewhere \end{array} \right) \tag{12.19}$$

The solution of 12.17 can be obtained from12.14

$$n_r^{-2}(s) I_n(s) = C_{0_n} \exp(\int_{s_{n+1}}^{s} \alpha(t)dt) - \int_{s_{n+1}}^{s} \frac{j(t)\Pi_n(t)}{n_r^2(t)} dt \exp(\int_{t}^{s} \alpha(q)dq) \tag{12.20}$$

Using 12.18 we can find C_{0_n}

$$n_r^{-2}(s_{n+2}) I_n(s_{n+1}) = C_{0_n} = 0 \tag{12.21}$$

It gives the emission

$$n_r^{-2}(s) I_n(s) = \int_{s}^{s_{n+1}} \frac{j(t)\Pi_n(t)}{n_r^2(t)} dt \exp(\int_{t}^{s} \alpha(q)dq) \tag{12.22}$$

Using 12.19 we get the following form for the emission $I_{\rm n}$

$$n_r^{-2}(s)I_n(s) = \begin{cases} \int_s^{s_{n+1}} \frac{j(t)}{n_r^2(t)} dt \exp(\int_t^s \alpha(q)dq), s_n < s \le s_{n+1} \\ \int_{s_n}^{s_{n+1}} \frac{j(t)}{n_r^2(t)} dt \exp(\int_t^s \alpha(q)dq), s \le s_n \end{cases} \tag{12.23}$$

This formula gives the value of $I_{\rm n}$ at the the detector side of nth bin at $s = s_n$

$$n_r^{-2}(s = s_n)I_n(s = s_n) = \int_{s_n}^{s_{n+1}} \frac{j(t)}{n_r^2(t)} dt \exp(-\int_{s_n}^{t} \alpha(q)dq) \tag{12.24}$$

This integral 12.24 can be evaluated as

$$\begin{gathered} n_r^{-2}(s = s_n)I_n(s = s_n) \approx \frac{j(s_{n+1/2})}{n_r^2(s_{n+1/2})} \int_{s_n}^{s_{n+1}} \exp(-\alpha(s_{n+1/2})(t - s_n))dt = \\ = \frac{j(s_{n+1/2})}{n_r^2(s_{n+1/2})\alpha(s_{n+1/2})}(1 - \exp(-\alpha(s_{n+1/2})(s_{n+1} - s_n)) \end{gathered} \tag{12.25}$$

The emission $I_{\rm n}$ at the detector $s = 0$ I_{0n} has the the form

$$\begin{gathered} n_r^{-2}(s = 0)I_n(s = 0) = \int_{s_n}^{s_{n+1}} \frac{j(t)}{n_r^2(t)} dt \exp(-\int_{s_n}^{t} \alpha(q)dq - \int_0^{s_n} \alpha(q)dq)) = \\ = n_r^{-2}(s = s_n)I_n(s = s_n) \exp(-\int_0^{s_n} \alpha(q)dq) \end{gathered} \tag{12.26}$$

The last integral in 12.26 can be evaluated as

$$\tau_n = \int_0^{s_n} \alpha(q)dq \approx \begin{cases} \sum_{m=1}^{n} (s_m - s_{m-1})\alpha_{m-1/2}, n \ge 2 \\ 0, n = 1 \end{cases} \tag{12.27}$$

It gives

$$n_r^{-2}(s = 0)I_n(s = 0) = n_r^{-2}(s = s_n)I_n(s = s_n) \exp(-\tau_n) \tag{12.28}$$

The specific intensity equal to $I_n(s = 0)/\triangle s$ for each spatial bin $\triangle s = s^{n+1} - s^n$ can has a very narrow and top peaks along the ray [6]. In [6] it was proposed the procedure that adds the points along the calculated ray in the vicinity of these peaks.

Typically the number of points along the ray trajectory N=100 is sufficient to coarsely resolve the main contribution of all peaks. The mesh along the ray is then refined by calculating additional values of α and j have been calculated, and where the resultant $I_n(s = 0)$ are grater than $t \cdot I(s = 0)$; t is a tolerance parameter, typically t=0.01. This refinement procedure thus adds calculations of α and j only at those points along the ray where they contribute significantly to the signal arriving at the detector.

The code can obtain emission from multiple passes across the tokamak; the ray is assumed to specularly reflect off the vacuum chamber wall. Mode conversion at the wall is not considered. Usually, though, a simple procedure [6] is employed for optically thin frequencies. This is done in view of the very complicated, and changing, nature of the vacuum vessel. A single pass is considered to fully represent any of the passes across the tokamak and back. The total radiation $I_n(s = 0)$ is given by

$$I_{0t} = I_0(1 + r\exp^{-\tau} + r^2 \exp^{-2\tau} + ...) = I_0/(1 - r\exp^{-\tau}) \tag{12.29}$$

The variables **tol_emiss=**τ **(0<tol_emiss<**1) and **wallr=**r **(0<wallr<**1) are given in **genray.dat** file in namelist /emission/ input data.
GENRAY can calculate the emission at the detector point for the set of the the wave frequencies. The variable **nfreq (in GENRAY.dat** file in namelist /emission/) determines the number of the used frequencies. The maximal value of **nfreq** is set by the parameter **nfreqa** in **param.i** file. For **nfreq=1** the emission frequency determined by the variable **frqncy**(GHZ) given in **GENRAY.dat** file in namelist /wave/. For **nfreq>1** the code uses the following frequencies

$$\omega_{ifreq} = \omega_{0EC}(freqncy00 + \frac{freq01 - freq00}{nfreq - 1}(ifreq - 1)) \tag{12.30}$$

Where ω_{0EC} is the electron cyclotron gyro-frequency at the magnetic axis, **freq00**$\leq$ **freq01** are the ratios of the minimal and maximal emission frequencies to the central electron-cyclotron frequency.

12.2 Radiation temperature.

The radiation temperature along the ray is calculated by the following formula [18]

$$T_r = \frac{1}{n_r^2}\frac{j}{\alpha}\frac{8\pi^3 c^2}{\omega^2} \tag{12.31}$$

The specific emission intensity $I_n(s=0)/\triangle s$ at the detector has the sharp maximum in some point along the ray [6]. The code determines the point $P_0 = (r_0, z)$ where the specific intensity $I_n(s=0)/\triangle s$ reaches the maximal value on the given ray. This point gives the maximum impact to the total emission and to the radiation temperature from the ray. The the ray radiation temperature can be calculated using the total radiation at the plasma edge [18]

$$T_{i_{rad}} = \frac{2\pi c^2 I_{0i}}{f_i^2} \tag{12.32}$$

For the ray radiation temperature with the wall reflection we use the following formula

$$T_{ti_{rad}} = \frac{2\pi c^2 I_{0ti}}{f_i^2} \tag{12.33}$$

were $i \equiv ifreq$ is the number of the ray, and f_i is the wave frequency that was used for the ray with the number **i=1**,...**nfreq**. Due to the sharp form of the specific emission intensity the ray radiation temperature should coincide with the plasma temperature in the point $P_0 = (r_0, z)$. For comparison the code calculates the plasma temperature $T_{plasma}(P_0)$.

12.3 Emission plots.

GENRAY launches the several rays differer by the frequencies from detector points. It can use the several detector points set by EC cone or grill launch input data. The

detector point specified by the space position and the vector refractive index.
Firstly the code calculates the ray trajectories and writes the ray coordinates along the rays in GENRAY.bin file. To get xdraw plots of these trajectories we should use the commands **xdraw genr** or **xdraw emis**.

Along all trajectories the code calculates several variables that can be plotted by xdraw using the command **xdraw em:**

$I_n(s = s_n)/\triangle s$ is the specific intensity from one bin $s_n < s < s_{n+1}$ at detector side of n_th bin at $s = s_n$, $\triangle s = s_{n+1} - s_n$,

$I_n(s = 0)/\triangle s$ is the specific intensity from one bin $s_n < s < s_{n+1}$ at the plasma boundary $s = 0$,

$I_{0sn} = \sum_{k=1}^{n} I_k(s = 0)$ is the emission at the plasma boundary at $s = 0$ from the part of the ray $0 < s < s_n$,

$\tau_n = \int_0^{s_n} \alpha(s)ds$ is the optical depth,

T_{pl} is the plasma temperature along the ray,

T_{rad} is the radiation plasma temperature 12.31 along the ray.

The command **xdraw emf1** creates the plots of the ray radiation temperature, the ray radiation temperature with wall reflection, the coordinates of the point $P_0 = (r_0, z)$, the plasma temperature versus the ratio ω/ω_{c0}. Here $\omega = 2\pi f$ is the wave frequency at the given ray, ω_{C0} is the electron cyclotron frequency fat the magnetic axis.
In some cases it is useful to compare the ray radiation temperature with the temperature where the wave frequency coincide with the second EC harmonic. The code has the option **i_r_2nd_harm.** If **i_r_2nd_harm=1** it code will determine the major radius of the second EC harmonic resonance point point $P_{2dnEC}(r_{2nd}, z = 0)$ at the midplane. The command **xdraw emfr** plots of the plasma temperature at the second harmonic EC points.

Chapter 13

GENRAY usage

13.1 Preliminaries

The GENRAY code is written in standard FORTRAN-77 and runs without modifications on Cray, Sun Stations, IBM Pentium PC. There are no mathematical library calls. All necessary routines are parts of the source code. The Fortran source files have the extension *.f.

There are a number of INCLUDE files for setting parameters and specifying common blocks. The INCLUDE files have the extension *.i. The include file param.i sets all the parameters for the code.

MAKEFILE compiles the code at Linux g77 compiler for IBM PC, f77 compiler Sun stations and the CF77 Cray compiler.

The are a number of *.in files. They are the command files for plotting the data from *.bin files under XDRAW graphic system. The file **xdraw.ini** is for color printing of figures made under XDRAW system. The command xdraw files (*.in) have the names: **drawname**.**bin**. To run the command file, we use the following command: **xdraw -c name**. The option -c is for color printing.

The code uses two input files:

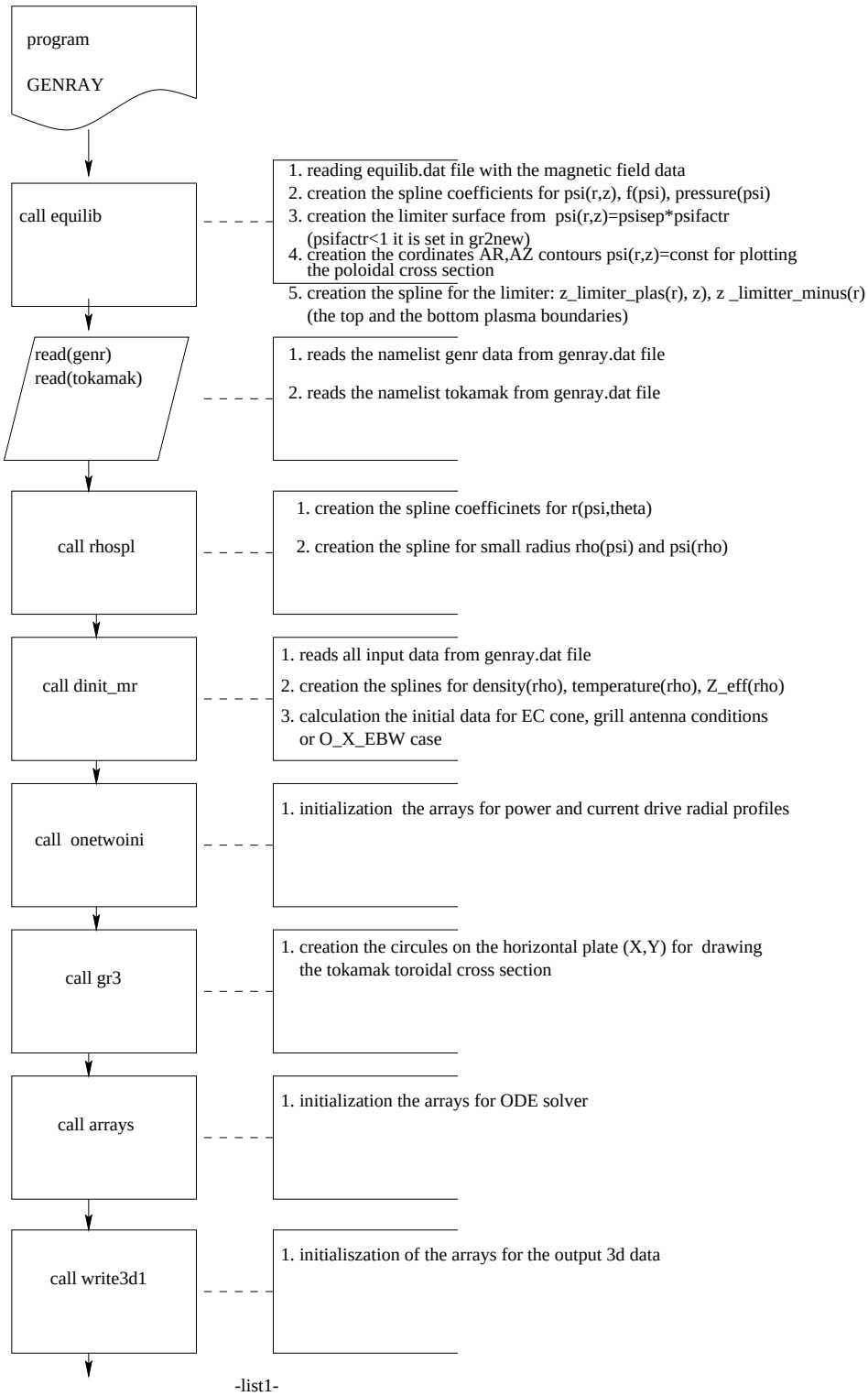
equilib.dat	This is a copy of eqdsk file
genray.dat	Inputs the data for all NAMELISTs used in the code.

The code creates the following output files:

con1	some input data from the eqdsk file
3d.dat	output data along all the rays for CQL3D code
genray.bin	output data for plotting the rays under XDRAW
onetwo.bin	for plotting power and CD radial profiles under XDRAW

13.2 Genray flowchart

The main code program is genray. It is in **genray.f** file. The flowchart of this program is the following figure:


program
GENRAY
call equilib
1. reading equilib.dat file with the magnetic field data
2. creation the spline coefficients for psi(r,z), f(psi), pressure(psi)
3. creation the limiter surface from psi(r,z)=psisep*psifactr
(psifactr<1 it is set in gr2new)
4. creation the cordinates AR,AZ contours psi(r,z)=const for plotting
the poloidal cross section
5. creation the spline for the limiter: z_limiter_plas(r), z), z _limitter_minus(r)
(the top and the bottom plasma boundaries)
read(genr)
read(tokamak)
1. reads the namelist genr data from genray.dat file
2. reads the namelist tokamak from genray.dat file
call rhospl
1. creation the spline coefficinets for r(psi,theta)
2. creation the spline for small radius rho(psi) and psi(rho)
call dinit_mr
1. reads all input data from genray.dat file
2. creation the splines for density(rho), temperature(rho), Z_eff(rho)
3. calculation the initial data for EC cone, grill antenna conditions
or O_X_EBW case
call onetwoini
1. initialization the arrays for power and current drive radial profiles
call gr3
1. creation the circules on the horizontal plate (X,Y) for drawing
the tokamak toroidal cross section
call arrays
1. initialization the arrays for ODE solver
call write3d1
1. initialiszation of the arrays for the output 3d data

-list1-

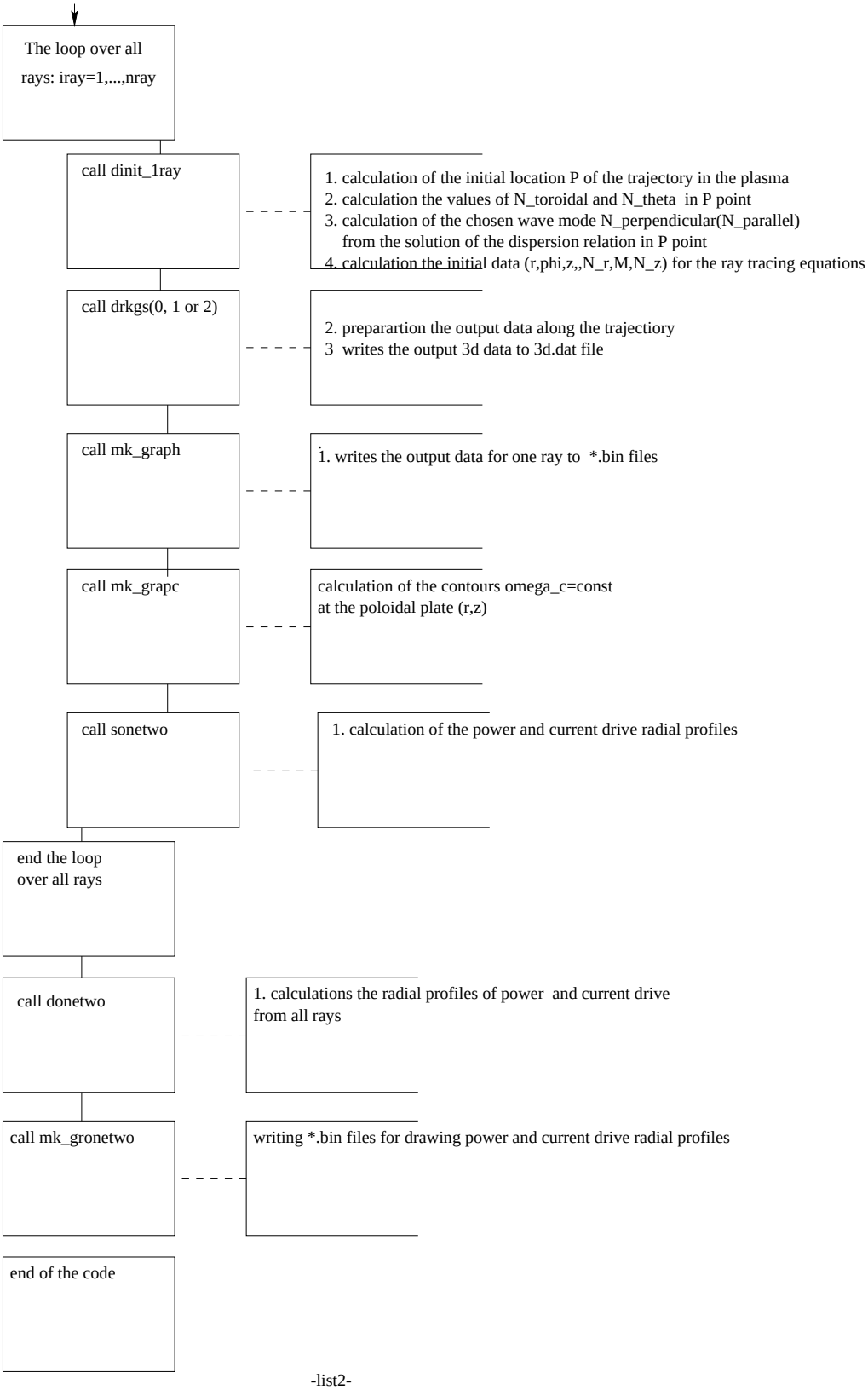


Figure 13.1: Flowchart of the main program.

13.3 Parameter input

The variables in the form of PARAMETERS are set in **param.i** file. The list of these PARAMETERS is given below.

——-for common block **cone.i**
nraymax is the maximum number of antenna rays. It must be greater or equal to the total number of rays **nray**, calculated in cone_ec routine.

——-for common block **five.i**
nxeqda is the maximal number of points in horizontal (X) direction in **equilib.dat (eqdsk)** file, it should be **nxeqda**.ge.**nxeqd**;
nyeqda is the maximal number of points in vertical (Y) direction in **equilib.dat (eqdsk)** file, it shoild be **nyeqda**.ge.**nyeqd**;
nlimit is the number of limiter points in **equilib.dat (eqdsk)** file or for calculation of the limiter surface;
nx4a=nxeqda+4, ny4a=nyeqda+4, nrya=ny4a, nlim4=nlimit+4 are the numbers of the points for the spline approximation of the poloidal flux and the plasma limiter. ATTENTION: this must be **nrya=max(nxeqda,nyeqda)+4.**

——- for common **fourb.i**
nves is the number of vacuum vessel points.

——- for common **gr.i**
npsi is the number of flux surface contours $\psi_j = \psi_{mag} + h_\psi(j-1)$, $h_\psi = (\psi_{limiter} - \psi_{mag})/(npsi-1)$, $j = 1,...,npsi$. These contours are used to determine the connection between the poloidal coordinate system (ψ,θ) and the Cartesian coordinates $r(\psi,\theta), z(\psi,\theta)$. Here ψ_{mag} is the poloidal flux at the magnetic axis (x_{mag}, y_{mag}) and $\psi_{limiter}$ is the poloidal flux at the limiter or the last closed flux surface.
nteta is the number of points along each flux contour in the poloidal direction $\psi(zpsi(\psi_j,\theta_i), rpsi(\psi_j,\theta_i)) = \psi_j$. The poloidal angle is measured in counter clockwise direction from the horizontal plate. This horizontal plate goes through the magnetic axis; $\theta_i = h_\theta(i-0.5)$, $h_\theta = 2\pi/nteta, i = 1, nteta1$
nteta1=nteta+1
epspsi is the accuracy for determination of the contour point coordinates $\psi(zpsi, rpsi) = \psi_j$

NL is the number of the poloidal flux contours for plotting the tokamak poloidal cross section,
NP=ntheta is used for plotting the tokamak toroidal cross section

——-for common **grill.i**
ngrilla is the maximal number of poloidal grills; see grill launch Section 7.2 **nnkprmax=**$\max_{i=1,ngrill} nnkpar(i)$
nnktormax=$\max_{i=1,ngrill} nnktor(i)$
nnkpolmrax=$\max_{i=1,ngrill} nnkpol(i)$
nthinmax=$\max_{i=1,ngrill} nthin(i)$

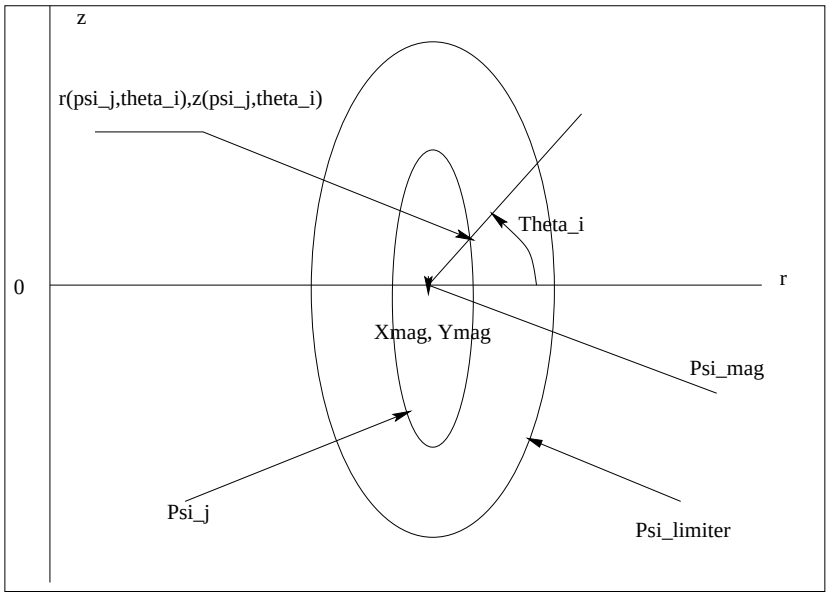


Figure 13.2: Flux coordinates

nraymaxl is the maximum number of rays from the grill antenna

———-for common **ions.i**
ncompa is the maximum number of plasma species; this must be $ncomp \geq nbulk \geq 1$

———-for common **onetwo.i**
NR is the number of points in the small radius direction for calculation of the power and current drive radial profiles.

———-for common **rho.i**
npsi4=npsi+4, nteta14=nteta1+4 are the number of points for the spline approximation of the small radius as a function of the poloidal flux .
———-for common **six.i**
ndensa is the maximal number of the points in the input arrays with density, temperature and Zeff small radial profiles.
ndens4a=ndensa+4
———-for common **write.i**
nrelta is maximum value for nrelt. Where nrelt is the maximum number of output ray points along every ray
nraya is the maximal value of the rays
nfreqa is the maximal value for **nfreq** the number of the emission frequencies
———-for common **scatnperp**.i (data for $N_\perp$ scattering)
nscat_n is the number of points in arrays rhoscat(**nscat_n**) and scatd(0:**nscat_n**)

——-for common dskincomm
iya is the maximal number of the points in the pitch angle mesh for the input distribution function**s**
ixa is the maximal number of the points in the momentum mesh for the input distribution function**s**
irza is the maximal number of the points in the small radial mesh for the input distribution function**s**
ngena is the maximal number of the plasma species for the input distribution functions

Chapter 14

Examples

14.1 Fast wave in D3D in cold plasma

We use the following input data for fast wave launch in D3D:

1) Dispersion relation: cold plasma, **id**=2

2) FW absorption **iabsorp**=3

3) Plasma species **nbulk**=3

electrons and two sorts of ions: Deuterium D(charge(2)=1, dmas(2)=3674) and Oxygen O (charge(3)=6, dmas(3)=22044)

4) Wave frequency - 60 MHZ

5)Center plasma density $n_{e=}4.46*10^{19}m^{-3}$, $n_{D=}4.46*10^{19}m^{-3}$, $n_{O=}0.446*10^{19}m^{-3}$

6) Center plasma temperature $T_e = 3.59\,KeV$, $T_D = 3.23\,KeV$, $T_O = 3.23\,KeV$

7) Wave power 0.683 MWT.

The results of GENRAY calculations for FW launch in D3D are plotted in these two sets of figures.

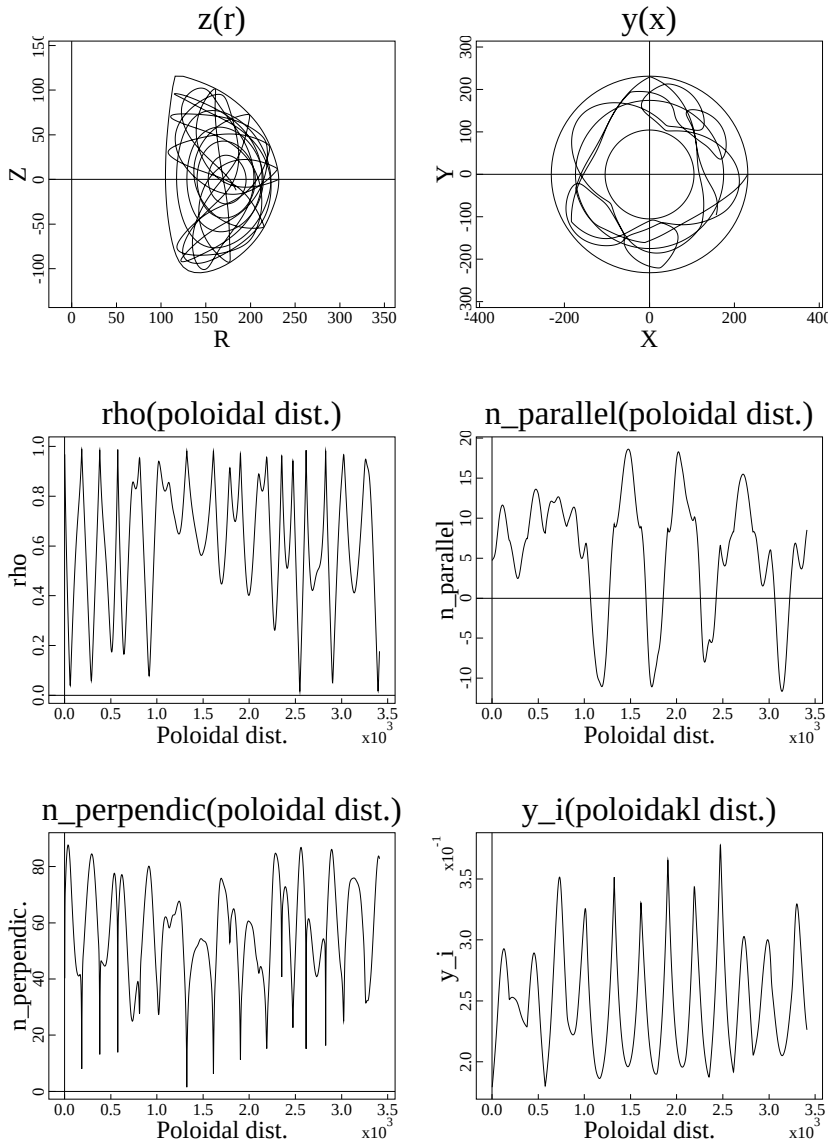


Figure 14.1: FW in D3D. a) Ray trajectory and refractive index and Y_i along the ray. The poloidal distance is the projection of the ray length on the poloidal cross-section.

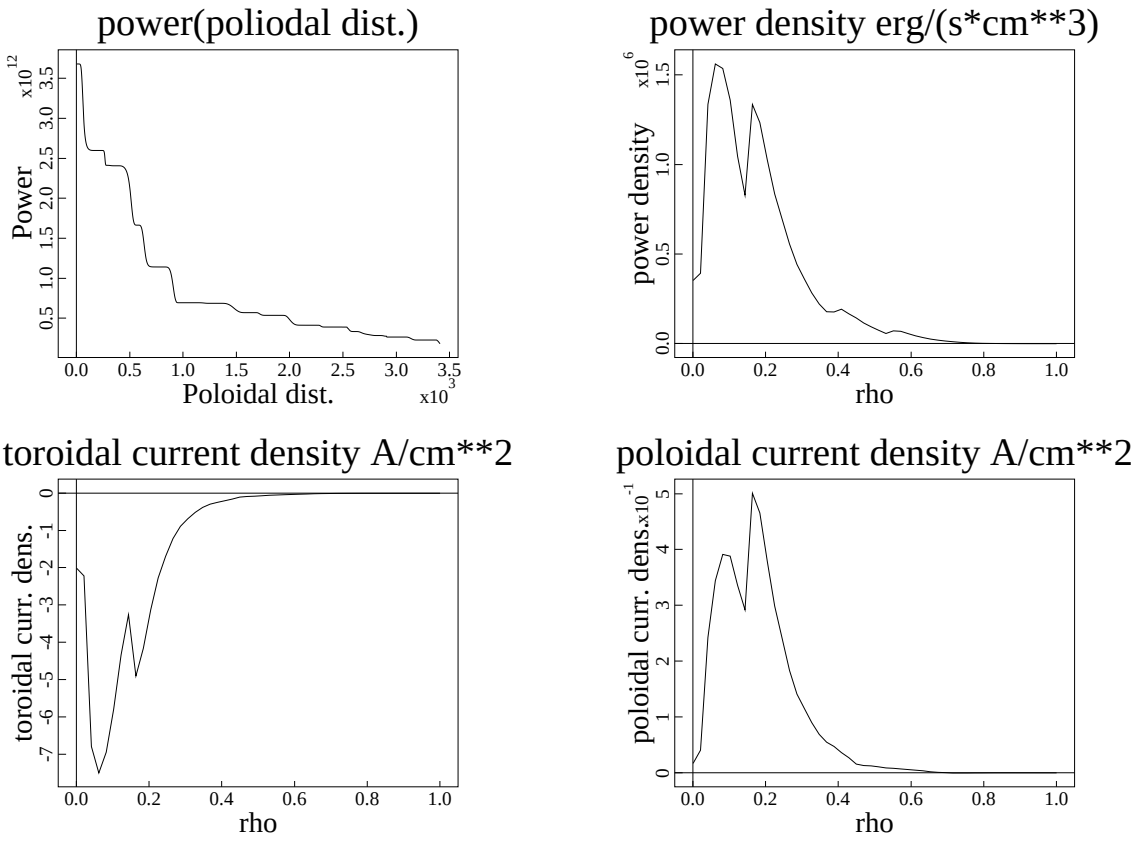


Figure 14.2: FW in D3D b) Wave power along the ray and small radius profiles of absorbed wave power density and current drive density.

Bibliography

[1] A. P. Smirnov and R. W. Harvey, Bull. Am. Phys. Soc. **40,** p. 1837, abstract 8p35, (1995).

[2] I. B. Bernstein , L. Friedland, in Handbook of Plasma Physics, edited by M. R. Rosenbluth and R.Z. Sagdeev (North-Holland, Amsterdam), Vo1. 1, pp. 367-418, 1983).

[3] T. H. Stix, Waves in Plasmas (American Institute of Physics, New York), (1992).

[4] N. A. Krall, A. W. Trivelpiece, Principles of Plasma Physics (McGraw-Hill book company, New York), (1973).

[5] E. Mazzucato, I. Fidone, G. Granata, Damping of electron cyclotron waves in dense plasmas of compact ignition tokamak, Phys. Fluids, **30** (12), pp. 3745-3751, (1987).

[6] R.W. Harvey, M. R. O'Brien, V.V Rozhdestvensky, T.C. Luce, Electron cyclotron emissions from nonthermal tokamak plasmas, Phys. Fluids B, **5** (2), pp. 446, (1993).

[7] M. Bornatici, U. Ruffina, Electron cyclotron absorption and emission in non-Maxwellian plasmas, Associate EURATOM-FOM, FOM Institute voor Plasmafysica, P. P. 85/06, (1985).

[8] S. C. Chiu, V. C. Chen, R, W. Harvey, M Porkolab, Theory of fast wave current drive for tokamak plasma, Nuclear Fusion, **29**, pp. 21750-2186, (1989).

[9] P. Bonoli, Linear theory of lower hybrid heating. IEEE transaction on plasma science, V. PS-12, pp. 95-107, (1984).

[10] P. L. Andrews and F. W. Perkins, Scattering of lower-hybrid waves by drift-wave density fluctuations: Solution of the radiative transfer equation, Phys. Fluids, **26**, (9), pp. 2537-2545, (1983

[11] J. Preinhaelter and V. Kopecky, Penetration of high-frequency waves into a weakly inhomogeneous magnetized plasma at oblique incidence and their transformation to Bernstein modes, J. Plasma Physics, **10,** part 1, pp.1-12, (1973).

[12] C. B. Forest, P. K. Chattopadhay, R. W. Harvey, A. P. Smirnov, Off-midplane launch of electron Bernstein waves for current drive in overdense plasma, Phys. Plasmas **7**, pp. 1352-13?, (2000).

[13] D. A. Ehst and C. F. F. Karney, Approximate formula for radio-frequency current drive efficiency with magnetic trapping, Nuclear Fusion, **31**, (10), pp. 1933-1938, (1991).

[14] R. H. Cohen, Effect of trapped electrons on current drive, Phys. Fluids, **30** (8), pp. 2442-2449, (1987).

[15] R. H. Cohen, Erratum: Effect of trapped electrons on current drive, Phys. Fluids, **31** (2), p. 421, (1988).

[16] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Numerical Recipes in Fortran 77 (Cambridge University Press), (1999).

[17] Masayuki Ono, High harmonics fast waves in high beta plasmas, Phys. Plasmas **2**, N11, pp. 4075-4082, (1995).

[18] G. Bekefi, Radiation Processes in Plasmas (Wiley, New York,1966).

[19] M. Bornatici, R. Cano, O. DeBarbieri and F. Engelmann, Nucl.Fusion, **23**, 1153 (1983)

[20] H. Weitzner and D. B. Batchelor, Phys. Fluids, **23**, 1359 (1980).