$dwk++$ User Manual

Feng Wang (fwang@pppl.gov)

March 8, 2016

PPPL, Princeton, NJ, 08540,USA.

Contents

1 Introduction

$dwk++$ is a small code to calculate δW_k and solver fishbone dispersion relation for toakmak plasmas. The δW_k is calculate by 3D integration in phase space (r, Λ, E). A slowing down distribution function and a kink like mode structure is used. With a small growth rate (imag part of Ω) and tokamak parameters as a input, this code find out the fishbone mode frequency and fast ion $\beta_{h,0}$.

2 The defintion of δW_k

2.1 The normalized δW_k in the code:

$$\delta W_k = \sum^{p} \int_0^1 \frac{J}{q} dr \int d\Lambda \int E^3 dE \tau_b (\omega - \omega_\star) \frac{\partial F}{\partial E} \frac{|Y_p|^2}{n\omega_\phi + p\omega_b - \omega} \tag{1}$$

In current version, we only keep signle n. $\sigma = \pm 1$ is the direction of $v_\parallel$, m and n is poloidal and toroidal mode number, $\Lambda = \frac{\mu}{E}$, E is the fast ion energy, τ_b is the particle bounce time. The slowing down distribution function of fast ions is:

$$F = \frac{2^{3/2}}{C_f} \hat{F}(r, \epsilon, \Lambda) = \frac{2^{3/2}}{C_f} \frac{1}{E^{3/2} + E_c^{3/2}} erfc\left(\frac{E - E_0}{\Delta E}\right) e^{-(\frac{r-r_0}{\Delta r})^2} e^{-(\frac{\Lambda - \Lambda_0}{\Delta\Lambda})^2} \tag{2}$$

$$C_f = \int d^3\mathbf{v} \frac{1}{E^{3/2} + E_c^{3/2}} erfc\left(\frac{E - E_0}{\Delta E}\right) e^{-(\frac{\Lambda - \Lambda_0}{\Delta\Lambda})^2} \tag{3}$$

And $d^3\mathbf{v} = \frac{\sqrt{2}\pi}{b\sqrt{1-\frac{\Lambda}{b}}} d\Lambda E^{1/2} dE$.

$$C_f = \int \frac{\sqrt{2}\pi}{b\sqrt{1-\frac{\Lambda}{b}}} \frac{1}{E^{3/2}+E_c^{3/2}} erfc\left(\frac{E-E_0}{\Delta E}\right) e^{-(\frac{\Lambda-\Lambda_0}{\Delta\Lambda})^2} d\Lambda E^{1/2} dE \tag{4}$$

$$\frac{\partial F}{\partial E} = \frac{-2^{3/2}}{C_f}\left[\frac{2\exp(-(\frac{E-E_0}{\Delta E})^2)}{\sqrt{\pi}\Delta E(E^{3/2}+E_c^{3/2})} + \frac{3\sqrt{E}erfc(\frac{E-E_0}{\Delta E})}{2(E^{3/2}+E_c^{3/2})^2} - \frac{2\Lambda(\Lambda-\Lambda_0)erfc\left(\frac{E-E_0}{\Delta E}\right)}{E\Delta\Lambda^2(E^{3/2}+E_c^{3/2})}\right] e^{-(\frac{r-r_0}{\Delta r})^2} e^{-(\frac{\Lambda-\Lambda_0}{\Delta\Lambda})^2} \tag{5}$$

$$\frac{\partial F}{\partial r} = \frac{2^{3/2}}{C_f}\frac{erfc\left(\frac{E-E_0}{\Delta E}\right)}{E^{3/2}+E_c^{3/2}}\frac{2(r_0-r)}{\Delta r^2} e^{-(\frac{r-r_0}{\Delta r})^2} e^{-(\frac{\Lambda-\Lambda_0}{\Delta\Lambda})^2} \tag{6}$$

The diamagnetic frequency:

$$\omega_* = \frac{nq}{2r}\frac{\rho_0}{\varepsilon_0}\frac{\partial F/\partial r}{\partial F/\partial E} \tag{7}$$

where ρ_0 is the gyro radius with injection energy. ε_0 is the inverse aspect-ratio, and $\varepsilon = \frac{r}{R_0}$. The transit frequency for passing particle is given below:

$$\omega_b = \frac{\pi\sqrt{\kappa}}{K(\kappa^{-1})}\frac{\sqrt{\varepsilon\Lambda/2}}{q}\sqrt{E} \tag{8}$$

where $\kappa = \frac{1-\Lambda(1-\varepsilon)}{2\varepsilon\Lambda}$, and K denotes the complete elliptic integral of the first kind. The transit frequency in toroidal:$\omega_\phi = q\omega_b$. Particle bounce time: $\tau_b = \frac{2\pi}{\omega_b}$. The integral along the particle orbits:

$$Y(r,\Lambda,E) = \frac{1}{2\pi}\int_0^{2\pi} \chi d\theta B_\Lambda\left(\Lambda_b + 2\left(1-\Lambda_b\right)\right) G(r,\theta,E) e^{-i\chi p\Theta} \tag{9}$$

where $\chi(r,\Lambda) = \frac{\sigma\pi\sqrt{\kappa}\sqrt{\varepsilon\Lambda/2}}{K(\kappa^{-1})}$, $\kappa = \frac{1-\Lambda(1-\varepsilon)}{2\varepsilon\Lambda}$. K denotes the complete elliptic integral of the first kind. $\Lambda_b = \frac{\Lambda}{b}$, $B_\Lambda(r,\Lambda,\theta) = \frac{1}{b\sqrt{(1-\Lambda_b)}}$, $b = B/B_0 = 1 + (r/R_0)cos\theta$.

$$G(\text{r,}\Lambda\text{,E,}\theta) = (g^{\theta\theta}\kappa_\theta + g^{r\theta}\kappa_r)\xi_\theta(\hat{r}(\bar{r},\rho_d,\theta),\theta) + (g^{rr}\kappa_r + g^{r\theta}\kappa_\theta)\xi_r(\hat{r}(\bar{r},\rho_d,\theta),\theta) \tag{10}$$

where $\hat{r} = \bar{r} + \rho_d cos\theta$.

$$\rho_d(r,\Lambda,E) = \frac{q}{2}\rho_0\sqrt{\frac{E}{1-\Lambda/b}}\left[\frac{\Lambda}{b} + 2(1-\frac{\Lambda}{b})\right] = \frac{q}{2}\rho_0\sqrt{\frac{E}{1-\Lambda_b}}\left[\Lambda_b + 2(1-\Lambda_b)\right] \tag{11}$$

To simplify the code, we use Λ_0 instead of Λ in ρ_d.

$$\rho_d(r,E) = \frac{q}{2}\rho_0\sqrt{\frac{E}{1-\Lambda_{0,b}}}\left[\Lambda_{0b} + 2(1-\Lambda_{ob})\right] \tag{12}$$

For $\Lambda_0 = 0$

$$\rho_d(r,E) = q\rho_0\sqrt{E}$$

$$G(\text{r,E,}\theta) = (g^{\theta\theta}\kappa_\theta + g^{r\theta}\kappa_r)\xi_\theta(\hat{r}(\bar{r},\rho_d,\theta),\theta) + (g^{rr}\kappa_r + g^{r\theta}\kappa_\theta)\xi_r(\hat{r}(\bar{r},\rho_d,\theta),\theta) \tag{13}$$

$g^{rr} = 1 + \frac{\varepsilon cos\theta}{2}$, $g^{\theta\theta} = \frac{1}{r^2}\left(1 - \frac{5}{2}\varepsilon cos\theta\right)$, $g^{r\theta} = -\frac{3}{2r}\varepsilon sin\theta$.

$$\Theta(\theta,r,\Lambda) = \int_0^\theta d\theta'\frac{1}{b\sqrt{(1-\Lambda_b)}} = \int_0^\theta B_\Lambda \tag{14}$$

2.2 The mode structure

In the current version, the mode structure is source code, and it is a kink structure with a fixed boundary at $r = 1$, and with a finite resonance layer width Δr.

$\xi_r(r,\theta) = \xi_{r0}(r)exp(-i\theta)$, $\xi_\theta(r,\theta) = -i\xi_{\theta 0}(r)rexp(-i\theta)$

$$\xi_{r0}(r) = \begin{cases} 1 & r \le r_s - \Delta r/2 \\ \frac{\Delta r - r + r_s - \Delta r/2}{\Delta r} & r_s - \frac{\Delta r}{2} < r < r_s + \frac{\Delta r}{2} \\ 0 & r \ge r_s + \frac{\Delta r}{2} \end{cases}$$

$$\xi_{\theta 0}(r) = \begin{cases} 1 & r \le r_s - \Delta r/2 \\ \frac{\Delta r - 2r + r_s - \Delta r/2}{\Delta r} & r_s - \frac{\Delta r}{2} < r < r_s + \frac{\Delta r}{2} \\ 0 & r \ge r_s + \frac{\Delta r}{2} \end{cases}$$

2.3 The normalized quantities used for δW_k:

$v_0 = \sqrt{2T_h/M}$, T_h is the fast ions injection energy, M is the fast ion's mass. $\omega_0 = \frac{v_0}{R_0}$. $F_0 = \frac{n_0}{v_0^3}$, n_0 is the fast ion density at axis. $r_0 = a$ is the minor radius, $\varepsilon = a/R_0$, $E_0 = T_h/M$, B_0 is the torodial magnetic field at magnetic axis. $\delta W_{k,0} = \pi^2 a^2 R_0 n_0 T_h$.

3 Fishbone dispersion relation

Assume $\delta W_{mhd} = 0$, the normalized dispersion relation is:

$$\frac{4}{\pi}(\frac{r_s}{R_0})^2|\frac{\xi_s}{\xi_0}|^2(-i\frac{\omega}{\omega_A}) + \beta_h \delta W_k = 0 \tag{15}$$

or:

$$i\omega = C\beta_{h,0}\delta W_k \tag{16}$$

where:

$$C = \frac{\omega_A}{\omega_0}\frac{1}{4(\frac{r_s}{R_0})^2|\frac{\xi_s}{\xi_0}|^2} \tag{17}$$

here $\xi_s/\xi_0 = 1$, $\omega_A = \frac{v_A}{3^{1/2}R_0 s}$, $v_A = \frac{B}{\sqrt{\mu_0\rho_m}}$, $s = r_s\frac{dq}{dr(r=r_s)}$, and$\beta_{h,0} = 8\pi n_0 T_h/B_t^2$.
Considering MHD contribution from $m = 1, n = 1$:

$$i\omega = C\beta_{h,0}\delta W_k + \frac{\omega_A}{\omega_0}\delta W_T \tag{18}$$

where $\delta W_T = 3\pi\left(\frac{r_s}{R}\right)^2(1-q_0)\left(\frac{13}{144} - \beta_{ps}^2\right)$.

4 How to run dwk++

4.1 Compile the code

dwk++ code is using c++ language as the main language, so it need a c++ compiler to compile the code. Gnu/g++ on max os and Linux and intel compiler on Linux was tested. To compile the code, libconfig with version >1.5 is needed, and set environment variable LIBCONFIG_DIR to the path where libconfig located. Set CXX to the c++ compiler (g++, icpc), and run 'make' at 'dwk-/' directory. If everything correct, a executable file 'dwk++' should be generated in 'dwk-/src/' directory and be copied to current directory. Then you can run dwk++ with dwk.cfg input file. A example of dwk.cfg can be find in 'dwk++/examples'.

4.2 Arguments

- -h print help information.
- -i input file name, dwk.cfg by default.
- -o outfile name, omega_dwk.out by default.
- -s scan dwk(omega), only find Ω_0 and $\beta_{h,0}$.

4.3 Input file

Here is an example of input file:

Listing 1: a input file example

```
/*the input file for dwk++, parameters units is in ( ) */
//tokamak parameters
tokamak=
{
        a=0.38;             //minor radius (m).
        R0=1.30;            //major radius (m).
        Bt =0.84;           //Toroidal magnetic field at axis without plasma, (Tesla).
        n0 =1.7e19;         //thermal plasma density, (m^-3).
        mi =2.0;            //ion mass, (protom mass,m_p).
        E_i0 =21.90995;     //Fost ion injection energy, (KeV).
        m_ep =2.0;          //fat ion mass (protom mass,m_p).
```

```
        //qc=[9.053654243264999e-01,1.229631602293500e+00,-1.854061034868290e+01,1.0859895461
        //qc=[8.053654243264999e-01,1.229631602293500e+00,-1.854061034868290e+01,1.0859895461
        qc=[0.8, 0.0, 1.3843];
        q_s=1.0;        //resonance surface

}
//grid parameters
grid=
{
        nx=202; //grid size should be 3n + 1, n is a positive integer.
        nL=202;
        nE=202;
        ntheta=202;
}
//fast ion distribution
slowing=
{
        r0=0.0;         //(a)
        rd=0.3;         //(a)
        L0=0.01;
        Ld=0.01;
        E0=1.0;         //(E_i0)
        Ed=0.02;                //(E_i0)
        Ec=0.3; //(E_i0)
        sigma=1;        //co(1) ? count (-1)
}
//perturbation and omega range for the soultion
mode=
{
        n=1;    //toroidal mode number
        m=1;    //poloidal mode number
        pa=0;   //resounces: sum Yps/(n*omega_phi +p*omega_theta -\omega).
        pb=0;   //sum from pa to pb
        delta_r=0.001; //step function width for kink. (a)
        omega_0=0.1;   //find omega_0(the solution) and scan dwk between omega_0 and omega_1,
        omega_1=0.99;   //omega unit is (v_0/R_0), v_0 is the fast ion injection speed.
        omega_i=0.005; //image part of omega, the growth rate
        omega_n=100;     //scan steps
        omega_err=1.0e-5; //residual of omega_0
        max_iter =100;    //maximum iteration number to find the omega_0
        max_iterg =2;
        dw_f=0.00;        //dw_mhd
        zero_rhod=0;    //1: with drift orbit width effect, 0 without
}

dwkopt=
{
        omega_star_off=1;       //1:  omega_star term on, 0: omega_star term off
        omega_off=1;            //1: omega term on, 0; omega term off
}
```

4.4 Output file

- omega_dwk.out,
- Yps.nc.

4.5 Utilities to plot results

There are some matlab & python scripts in 'dwk++/utilities' directory.

5 Examples

5.1 Compare with (WANG, Destabilization of internal kink modes at high frequency by energetic circulating ions. Physical Review Letters, 2001, 86) case:

The δW_k analytical results:

$$\delta W_k = C(\delta W_{k,d} + \delta W_{k,s})$$

where $C = 4 \cdot \frac{2^{3/2} B m_h \pi^2 \xi_0^2}{R}$.

$$\delta W_{k,d} = \frac{-\pi R^2 v_0^2}{\Omega_c} \int drqH^2(r_s - r)\frac{\partial c_0(r)}{\partial r}[\Omega^3 \log(1-\frac{1}{\Omega}) + \Omega^2 + \frac{1}{2}\Omega + \frac{1}{3}] \tag{19}$$

$$\delta W_{k,s} = -\pi v_0^2 \int rdrc_0(r)H^2(r_s - r)[-\frac{\Omega}{\Omega - 1} + \Omega + \Omega^2 log(1-\frac{1}{\Omega})]$$

$c_0(r) = p_h(r)/(2^{3/2}\pi m_h B E_0)$
Assume $p_h = p_0 exp(-(\frac{r}{\Delta r})^2)$, $q = 0.5 + 2r^2$, $r_s = 0.5$
$\frac{\partial c_0(r)}{\partial r} = -\frac{2r}{\Delta r^2} p_0 exp(-(\frac{r}{\Delta r})^2)/(2^{3/2}\pi m_h B E_0)$

$$\begin{aligned}
\delta W_{k,d} &= \frac{-\pi R^2 v_0^2}{\Omega_c} \int drqH^2(r_s - r)\frac{\partial c_0(r)}{\partial r}[\Omega^3 \log(1-\frac{1}{\Omega}) + \Omega^2 + \frac{1}{2}\Omega + \frac{1}{3}] \\
&= \frac{-\pi R^2 v_0^2}{\Omega_c}[\Omega^3 \log(1-\frac{1}{\Omega}) + \Omega^2 + \frac{1}{2}\Omega + \frac{1}{3}] \int_0^{r_s} drq\frac{\partial c_0(r)}{\partial r} \\
&= \frac{\pi R^2 v_0^2}{\Omega_c}[\Omega^3 \log(1-\frac{1}{\Omega}) + \Omega^2 + \frac{1}{2}\Omega + \frac{1}{3}]\frac{2p_0}{(2^{3/2}\pi m_h B E_0)\Delta r^2} \int_0^{r_s} dr(0.5 + 2r^2)[r * exp(-(\frac{r}{\Delta r})^2)] \\
&= \frac{\pi R^2 v_0^2}{\Omega_c}[\Omega^3 \log(1-\frac{1}{\Omega}) + \Omega^2 + \frac{1}{2}\Omega + \frac{1}{3}]\frac{-2p_0}{(2^{3/2}\pi m_h B E_0)}[e^{\frac{-r_s^2}{\Delta r^2}}(\Delta r^2 + r_s^2 + 0.25) - \Delta r^2 - 0.25)]
\end{aligned} \tag{20}$$

$$\begin{aligned}
\delta W_{k,s} &= -\pi v_0^2 \int rdrc_0(r)H^2(r_s - r)[-\frac{\Omega}{\Omega - 1} + \Omega + \Omega^2 log(1-\frac{1}{\Omega})] \\
&= -\pi v_0^2[-\frac{\Omega}{\Omega - 1} + \Omega + \Omega^2 log(1-\frac{1}{\Omega})]\frac{p_0}{(2^{3/2}\pi m_h B E_0)} \int_0^{r_s} rdrexp(-(\frac{r}{\Delta r})^2) \\
&= -\pi v_0^2[-\frac{\Omega}{\Omega - 1} + \Omega + \Omega^2 log(1-\frac{1}{\Omega})]\frac{p_0 \frac{1}{2}\Delta r^2 (1 - e^{\frac{-r_s^2}{\Delta r^2}})}{(2^{3/2}\pi m_h B E_0)}
\end{aligned} \tag{21}$$

In test case (examples/dwk_p0.cfg), $\Delta r = 0.2a = 0.08m$, $r_s = 0.5a = 0.2m$

6 Appendix

For more details about formula derivation, please read liming_kinetic.pdf