Introduction to Eigenvalue Solver for ITG and Gyrofluid Simulation using BOUT++ GLF code in Core Region

S. S. Kim, H. Jhang, P. H. Diamond, T. Rhee, G. Y. Park

WCI Center for Fusion Theory, NFRI, Korea

X. Q. Xu, P. W. Xi, A. Dimits, M. Umansky

LLNL

BOUT++ 2013 Workshop

Introduction to

Eigenvalue Solver for ITG

Governing equations for ITG

(Ottaviani et. al., PoP '99)

- Vorticity equation

$$\frac{\partial \Omega_i}{\partial t} + \mathbf{V}_E \cdot \nabla \Omega_i = -n \nabla_\parallel V_\parallel + n\left(\mathbf{V}_E + \mathbf{V}_{p_i}\right) \cdot \left(\boldsymbol{\kappa} + \nabla \ln B\right) + n \mathbf{V}_{p_i} \cdot \nabla \left(\frac{n_1 - \Omega_i}{n} \right) - \mathbf{V}_E \cdot \nabla n_{eq} + D_c \Omega_i$$

where $\Omega_i = n_1 - n_{pol}$: generalized vorticity, $n_{pol} = \frac{ne}{m_i \omega_{ci}^2} \nabla_\perp^2 \varphi$, $n_1 = n_{eq} \frac{e\varphi_1}{T_e}$: adiabatic response

$$\mathbf{V}_E = \frac{c}{B} \mathbf{b} \times \nabla \varphi, \quad \mathbf{V}_{p_i} = \frac{c}{neB} \mathbf{b} \times \nabla p_i$$

- Ion temperature equation

$$\frac{\partial T_i}{\partial t} + \mathbf{V}_E \cdot \nabla T_i = -\frac{2}{3} T_i \nabla_\parallel V_\parallel + D_c T_i + D_{glf} T_i$$

- Ion parallel velocity equation

$$\frac{\partial V_\parallel}{\partial t} + \mathbf{V}_E \cdot \nabla V_\parallel = -\frac{e}{m_i} \nabla_\parallel \varphi - \frac{1}{m_i n} \nabla_\parallel p_i + D_c V_\parallel$$

where $D_c F = \mu_1 \nabla_\perp^2 F - \mu_2 \nabla_\perp^4 F$: viscous damping, $D_{glf} T_i = -\sqrt{\frac{8 T_{i,eq}}{\pi m_i}} \left| \nabla_\parallel \right| T_{i1}$: Landau damping.

Linearized equations

$$\frac{\partial \tilde{\Omega}_i}{\partial \bar{t}} = -\hat{n}_i \hat{\nabla}_{\parallel} \tilde{V}_{i\parallel} + i\left(\hat{n}_i \omega_{di} \tilde{\varphi} + \omega_{di} \tilde{p}_i\right) + \left[\frac{\hat{p}_i}{\rho_*}, \tilde{\nabla}_{\perp}^2 \tilde{\varphi}\right] - \left[\tilde{\Psi}, \frac{\hat{n}_i}{\rho_*}\right] + \overline{D}_c \tilde{\Omega}_i$$

$$\frac{\partial \tilde{T}_i}{\partial \bar{t}} + \left[\tilde{\Psi}, \frac{\hat{T}_i}{\rho_*}\right] = -\frac{2}{3} \hat{T}_i \nabla_{\parallel} \tilde{V}_{i\parallel} + \overline{D}_c \tilde{T}_i + \overline{D}_{glf} \tilde{T}_i$$

$$\frac{\partial \tilde{V}_{i\parallel}}{\partial \bar{t}} = -\hat{\nabla}_{\parallel} \tilde{\varphi} - \frac{1}{\hat{n}_i} \hat{\nabla}_{\parallel} \tilde{p}_i + \overline{D}_c \tilde{V}_{i\parallel}$$

where

$$\tilde{\Omega}_i = \tilde{n}_i - \hat{n}_i \tilde{\nabla}_{\perp}^2 \tilde{\varphi}, \quad \tilde{n}_i = \frac{\hat{n}_i}{\hat{T}_e} \tilde{\varphi},$$

$$\tilde{\Psi} = \Gamma_0^{1/2} \tilde{\varphi} = \left(1 - \frac{1}{2} \tilde{\nabla}_{\perp}^2\right)^{-1} \tilde{\varphi}$$

is gyroaveraged potential.

following normalizations are used,

$$F = F_{eq} + F_1, \quad F = \left(\Omega, V_{\parallel}, n, T, \varphi\right), \quad F_0 = \left(n_0, c_{s0}, n_0, T_0, \frac{T_0}{e}\right)$$

$$\hat{F} = \frac{F_{eq}}{F_0}, \quad \tilde{F} = \frac{F_1}{\rho_* F_0}, \quad \tilde{\nabla}_{\perp} = \rho_* a \nabla_{\perp}, \quad \hat{\nabla}_{\parallel} = a \nabla_{\parallel}, \quad \rho = \frac{r}{a}, \quad \bar{t} = \frac{t}{t_0}$$

$$\rho_* = \frac{\rho_{s0}}{a}, \quad \rho_{s0} = \frac{c_{s0}}{\omega_{c0}}, \quad c_{s0} = \sqrt{\frac{T_0}{m_i}}, \quad \omega_{c0} = \frac{e_i B_0}{m_i c}, \quad t_0 = \frac{a}{c_{s0}}$$

$$\left[f, g\right] \equiv \rho_{s0}^2 \mathbf{b} \times \nabla f \cdot \nabla g = \frac{\rho_*^2}{\rho} \left(\frac{\partial f}{\partial \rho} \frac{\partial g}{\partial \theta} - \frac{\partial g}{\partial \rho} \frac{\partial f}{\partial \theta}\right)$$

$$i \omega_{di} \tilde{F} \equiv 2 \rho_{s0} a \mathbf{b} \times \nabla \tilde{F} \cdot \nabla \ln B = 2 \rho_* \frac{a}{R} \left(\frac{1}{\rho} \cos\theta \frac{\partial \tilde{F}}{\partial \theta} + \sin\theta \frac{\partial \tilde{F}}{\partial \rho}\right)$$

Note: F_0 is constant (not the value on magnetic axis)

$$\frac{\partial \tilde{\Omega}_i}{\partial \bar{t}} = -\hat{n}_i \hat{\nabla}_{\parallel} \tilde{V}_{i\parallel} + i\left(\hat{n}_i \omega_{di} \tilde{\varphi} + \omega_{di} \tilde{p}_i\right) + \left[\frac{\hat{p}_i}{\rho_*}, \tilde{\nabla}_{\perp}^2 \tilde{\varphi}\right] + \left[\frac{\hat{n}_i}{\rho_*}, \tilde{\Psi}\right] + \bar{D}_c \tilde{\Omega}_i$$

$$\Rightarrow \quad \omega M \tilde{\varphi} = \hat{n}_i k_{\parallel} \tilde{V}_{i\parallel} - a_W \tilde{\varphi} - b_W \tilde{T}_i + G_{\hat{p}_i} L \tilde{\varphi} + G_{\hat{n}_i} J_0 \tilde{\varphi} + i\bar{D}_c M \tilde{\varphi}$$

$$\frac{\partial \tilde{T}_i}{\partial \bar{t}} = \left[\frac{\hat{T}_i}{\rho_*}, \tilde{\Psi}\right] - \frac{2}{3} \hat{T}_i \nabla_{\parallel} \tilde{V}_{i\parallel} + \bar{D}_c \tilde{T}_i + \bar{D}_{glf} \tilde{T}_i$$

$$\Rightarrow \quad \omega \tilde{T}_i = G_{\hat{T}_i} J_0 \tilde{\varphi}_i + \frac{2}{3} \hat{T}_i k_{\parallel} \tilde{V}_{i\parallel} + i\bar{D}_c \tilde{T}_i + i\bar{D}_{glf} \tilde{T}_i$$

$$\frac{\partial \tilde{V}_{i\parallel}}{\partial \bar{t}} = -\hat{\nabla}_{\parallel} \tilde{\varphi} - \frac{1}{\hat{n}_i} \hat{\nabla}_{\parallel} \tilde{p}_i + \bar{D}_c \tilde{V}_{i\parallel}$$

$$\Rightarrow \quad \omega \tilde{V}_{i\parallel} = \left(1 + \frac{\hat{T}_i}{\hat{T}_e}\right) k_{\parallel} \tilde{\varphi} + k_{\parallel} \tilde{T}_i + i\bar{D}_c \tilde{V}_{i\parallel}$$

where we introduce

$$\frac{\partial}{\partial \bar{t}} \equiv -i\omega, \qquad \tilde{\nabla}_{\perp}^2 \tilde{\varphi} \equiv L\tilde{\varphi}, \qquad \hat{\nabla}_{\parallel} \equiv -ik_{\parallel}$$

$$\tilde{\Omega}_i = \left(\frac{\hat{n}_i}{\hat{T}_e} - \hat{n}_i \tilde{\nabla}_{\perp}^2\right) \tilde{\varphi} \equiv M\tilde{\varphi}$$

$$\tilde{\Psi} = \left(1 - \frac{1}{2} \tilde{\nabla}_{\perp}^2\right)^{-1} \tilde{\varphi} \equiv J_0 \tilde{\varphi}$$

$$\left[\frac{\hat{F}}{\rho_*}, \quad\right] = \rho_{s0}^2 \mathbf{b} \times \nabla \frac{\hat{F}}{\rho_*} \cdot \nabla \equiv -iG_{\hat{F}}$$

$$i\left(\hat{n}_i \omega_{di} \tilde{\varphi} + \omega_{di} \tilde{p}_i\right) \equiv i\left(a_W \tilde{\varphi} + b_W \tilde{T}_i\right)$$

and use

$$\tilde{p}_i = \frac{\hat{p}_i}{\hat{T}_e} \tilde{\varphi}_i + \hat{n}_i \tilde{T}_i.$$

$$\frac{\partial \tilde{\Omega}_i}{\partial \bar{t}} = -\hat{n}_i \hat{\nabla}_{\parallel} \tilde{V}_{i\parallel} + i\left(\hat{n}_i \omega_{di} \tilde{\varphi} + \omega_{di} \tilde{p}_i\right) + \left[\frac{\hat{p}_i}{\rho_*}, \tilde{\nabla}_{\perp}^2 \tilde{\varphi}\right] + \left[\frac{\hat{n}_i}{\rho_*}, \tilde{\Psi}\right] + \bar{D}_c \tilde{\Omega}_i$$

$$\Rightarrow \quad \omega M \tilde{\varphi} = \hat{n}_i k_{\parallel} \tilde{V}_{i\parallel} - a_W \tilde{\varphi} - b_W \tilde{T}_i + G_{\hat{p}_i} L \tilde{\varphi} + G_{\hat{n}_i} J_0 \tilde{\varphi} + i\bar{D}_c M \tilde{\varphi}$$

$$\Rightarrow \quad \boxed{\omega \tilde{\varphi} = M^{-1}\left(-a_W + G_{\hat{p}_i} L + G_{\hat{n}_i} J_0 + i\bar{D}_c M\right)\tilde{\varphi} - M^{-1} b_W \tilde{T}_i + M^{-1} \hat{n}_i k_{\parallel} \tilde{V}_{i\parallel}}$$

$$\frac{\partial \tilde{T}_i}{\partial \bar{t}} = \left[\frac{\hat{T}_i}{\rho_*}, \tilde{\Psi}\right] - \frac{2}{3} \hat{T}_i \nabla_{\parallel} \tilde{V}_{i\parallel} + \bar{D}_c \tilde{T}_i + \bar{D}_{glf} \tilde{T}_i$$

$$\Rightarrow \quad \omega \tilde{T}_i = G_{\hat{T}_i} J_0 \tilde{\varphi}_i + \frac{2}{3} \hat{T}_i k_{\parallel} \tilde{V}_{i\parallel} + i\bar{D}_c \tilde{T}_i + i\bar{D}_{glf} \tilde{T}_i$$

$$\Rightarrow \quad \boxed{\omega \tilde{T}_i = G_{\hat{T}_i} J_0 \tilde{\varphi}_i + \left(i\bar{D}_c + i\bar{D}_{glf}\right)\tilde{T}_i + \frac{2}{3} \hat{T}_i k_{\parallel} \tilde{V}_{i\parallel}}$$

$$\frac{\partial \tilde{V}_{i\parallel}}{\partial \bar{t}} = -\hat{\nabla}_{\parallel} \tilde{\varphi} - \frac{1}{\hat{n}_i} \hat{\nabla}_{\parallel} \tilde{p}_i + \bar{D}_c \tilde{V}_{i\parallel}$$

$$\Rightarrow \quad \boxed{\omega \tilde{V}_{i\parallel} = \left(1 + \frac{\hat{T}_i}{\hat{T}_e}\right) k_{\parallel} \tilde{\varphi} + k_{\parallel} \tilde{T}_i + i\bar{D}_c \tilde{V}_{i\parallel}}$$

where we introduce

$$\frac{\partial}{\partial \bar{t}} \equiv -i\omega, \qquad \tilde{\nabla}_{\perp}^2 \tilde{\varphi} \equiv L\tilde{\varphi}, \qquad \hat{\nabla}_{\parallel} \equiv -ik_{\parallel}$$

$$\tilde{\Omega}_i = \left(\frac{\hat{n}_i}{\hat{T}_e} - \hat{n}_i \tilde{\nabla}_{\perp}^2\right)\tilde{\varphi} \equiv M\tilde{\varphi}$$

$$\tilde{\Psi} = \left(1 - \frac{1}{2}\tilde{\nabla}_{\perp}^2\right)^{-1} \tilde{\varphi} \equiv J_0 \tilde{\varphi}$$

$$\left[\frac{\hat{F}}{\rho_*}, \quad\right] = \rho_{s0}^2 \mathbf{b} \times \nabla \frac{\hat{F}}{\rho_*} \cdot \nabla \equiv -iG_{\hat{F}}$$

$$i\left(\hat{n}_i \omega_{di} \tilde{\varphi} + \omega_{di} \tilde{p}_i\right) \equiv i\left(a_W \tilde{\varphi} + b_W \tilde{T}_i\right)$$

and use

$$\tilde{p}_i = \frac{\hat{p}_i}{\hat{T}_e} \tilde{\varphi}_i + \hat{n}_i \tilde{T}_i.$$

$$\frac{\partial \tilde{\Omega}_i}{\partial \bar{t}} = -\hat{n}_i \hat{\nabla}_{\parallel} \tilde{V}_{i\parallel} + i\left(\hat{n}_i \omega_{di} \tilde{\varphi} + \omega_{di} \tilde{p}_i\right) + \left[\frac{\hat{p}_i}{\rho_*}, \tilde{\nabla}_{\perp}^2 \tilde{\varphi}\right] + \left[\frac{\hat{n}_i}{\rho_*}, \tilde{\Psi}\right] + \bar{D}_c \tilde{\Omega}_i$$

$$\Rightarrow \quad \omega M \tilde{\varphi} = \hat{n}_i k_{\parallel} \tilde{V}_{i\parallel} - a_W \tilde{\varphi} - b_W \tilde{T}_i + G_{\hat{p}_i} L \tilde{\varphi} + G_{\hat{n}_i} J_0 \tilde{\varphi} + i\bar{D}_c M \tilde{\varphi}$$

$$\Rightarrow \quad \boxed{\omega \tilde{\varphi} = M^{-1}\left(-a_W + G_{\hat{p}_i} L + G_{\hat{n}_i} J_0 + i\bar{D}_c M\right)\tilde{\varphi} - M^{-1} b_W \tilde{T}_i + M^{-1} \hat{n}_i k_{\parallel} \tilde{V}_{i\parallel}}$$

Those operators can be represented as matrices in a functional space.

$$\frac{\partial \tilde{T}_i}{\partial \bar{t}} = \left[\frac{\hat{T}_i}{\rho_*}, \tilde{\Psi}\right] - \frac{2}{3} \hat{T}_i \nabla_{\parallel} \tilde{V}_{i\parallel} + \bar{D}_c \tilde{T}_i + \bar{D}_{glf} \tilde{T}_i$$

$$\Rightarrow \quad \omega \tilde{T}_i = G_{\hat{T}_i} J_0 \tilde{\varphi}_i + \frac{2}{3} \hat{T}_i k_{\parallel} \tilde{V}_{i\parallel} + i\bar{D}_c \tilde{T}_i + i\bar{D}_{glf} \tilde{T}_i$$

$$\Rightarrow \quad \boxed{\omega \tilde{T}_i = G_{\hat{T}_i} J_0 \tilde{\varphi}_i + \left(i\bar{D}_c + i\bar{D}_{glf}\right)\tilde{T}_i + \frac{2}{3} \hat{T}_i k_{\parallel} \tilde{V}_{i\parallel}}$$

$$\frac{\partial \tilde{V}_{i\parallel}}{\partial \bar{t}} = -\hat{\nabla}_{\parallel} \tilde{\varphi} - \frac{1}{\hat{n}_i} \hat{\nabla}_{\parallel} \tilde{p}_i + \bar{D}_c \tilde{V}_{i\parallel}$$

$$\Rightarrow \quad \boxed{\omega \tilde{V}_{i\parallel} = \left(1 + \frac{\hat{T}_i}{\hat{T}_e}\right) k_{\parallel} \tilde{\varphi} + k_{\parallel} \tilde{T}_i + i\bar{D}_c \tilde{V}_{i\parallel}}$$

where we introduce

$$\frac{\partial}{\partial \bar{t}} \equiv -i\omega, \qquad \tilde{\nabla}_{\perp}^2 \tilde{\varphi} \equiv L\tilde{\varphi}, \qquad \hat{\nabla}_{\parallel} \equiv -ik_{\parallel}$$

$$\tilde{\Omega}_i = \left(\frac{\hat{n}_i}{\hat{T}_e} - \hat{n}_i \tilde{\nabla}_{\perp}^2\right)\tilde{\varphi} \equiv M\tilde{\varphi}$$

$$\tilde{\Psi} = \left(1 - \frac{1}{2}\tilde{\nabla}_{\perp}^2\right)^{-1} \tilde{\varphi} \equiv J_0 \tilde{\varphi}$$

$$\left[\frac{\hat{F}}{\rho_*}, \quad\right] = \rho_{s0}^2 \mathbf{b} \times \nabla \frac{\hat{F}}{\rho_*} \cdot \nabla \equiv -iG_{\hat{F}}$$

$$i\left(\hat{n}_i \omega_{di} \tilde{\varphi} + \omega_{di} \tilde{p}_i\right) \equiv i\left(a_W \tilde{\varphi} + b_W \tilde{T}_i\right)$$

and use

$$\tilde{p}_i = \frac{\hat{p}_i}{\hat{T}_e} \tilde{\varphi}_i + \hat{n}_i \tilde{T}_i.$$

Final form of eigenvalue equation

$$\omega\tilde{\varphi} = A_{11}\tilde{\varphi} + A_{12}\tilde{T}_i + A_{13}\tilde{V}_{i\|}$$

$$\omega\tilde{T}_i = A_{21}\tilde{\varphi} + A_{22}\tilde{T}_i + A_{23}\tilde{V}_{i\|}$$

$$\omega\tilde{V}_{i\|} = A_{31}\tilde{\varphi} + A_{32}\tilde{T}_i + A_{33}\tilde{V}_{i\|}$$

$$\boxed{\omega\mathbf{F} = \mathbf{A}\cdot\mathbf{F}}$$

$$[\mathbf{F},\omega] = \mathrm{eig}(\mathbf{A});$$

in *matlab*

$$\mathbf{F} = \begin{pmatrix} \tilde{\boldsymbol{\varphi}} \\ \tilde{\mathbf{T}}_{\mathbf{i}} \\ \tilde{\mathbf{V}}_{\mathbf{i}\|} \end{pmatrix}, \quad \mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \mathbf{A}_{13} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \mathbf{A}_{23} \\ \mathbf{A}_{31} & \mathbf{A}_{32} & \mathbf{A}_{33} \end{pmatrix}$$

$$\tilde{\boldsymbol{\varphi}} = \sum_k \tilde{\varphi}_k |k\rangle = \begin{pmatrix} \tilde{\varphi}_1 \\ \tilde{\varphi}_2 \\ \tilde{\varphi}_3 \\ \vdots \end{pmatrix}, \quad \ldots$$

$|k\rangle$ is a basis function

where

$$A_{11} = M^{-1}\left(-a_W + G_{\hat{p}_i}L + G_{\hat{n}_i}J_0 + i\overline{D}_c M\right), \qquad A_{12} = -M^{-1}b_w, \qquad A_{13} = M^{-1}\hat{n}_i k_\|$$

$$A_{21} = G_{\hat{T}_i}J_0, \qquad A_{22} = i\overline{D}_c + i\overline{D}_{glf}, \qquad A_{23} = \frac{2}{3}\hat{T}_i k_\|$$

$$A_{31} = \left(1 + \frac{\hat{T}_i}{\hat{T}_e}\right)k_\|, \qquad A_{32} = k_\|, \qquad A_{33} = i\overline{D}_c$$

Basis functions

Field F can represented as $F(\rho,\theta,\zeta,t)=e^{-i\omega t}\sum_{k=mnp}F_k W_k(\rho)e^{i(m\theta-n\zeta)}$ in concentric - circular geometry :

$$\mathbf{F}=|F\rangle=e^{-i\omega t}\sum_k F_k|k\rangle,\quad \boxed{|k\rangle=|mnp\rangle=W_{mnp}(\rho)e^{i(m\theta-n\zeta)}},$$

inner product:

$$\langle k|F\rangle\equiv\frac{1}{4\pi^2}\int_0^1\rho d\rho\int_0^{2\pi}d\theta\int_0^{2\pi}d\zeta W_{mnp}(\rho)e^{-i(m\theta-n\zeta)}F,\qquad \langle k|k'\rangle=\delta_{kk'}$$

- Radial basis function

For Hermite basis ,

$$W_{mnp}(\rho)=\frac{1}{\sqrt{2\rho w_{mn}}\nu_p}H_p(x)e^{-x^2/2}$$

where $x=\dfrac{\rho-\rho_{mn}}{w_{mn}}$ $\left(\rho_{mn}\text{ is radial position of rational surface satisfying } q=\dfrac{m}{n}\right)$, $\nu_p=2^{p/2}\Gamma(p+1)^{1/2}\pi^{1/4}$

For Bessel basis,

$$W_{mnp}(\rho)=\frac{\sqrt{2}}{J_{m+1}(\alpha_{mp})}J_m(\alpha_{mp}\rho)\quad\text{where } J_m(\alpha_{mp})=0$$

Matrix (M, J0, Dc) involving Laplacian L

For Bessel basis,

$$L_{kk'} = \langle k | \tilde{\nabla}_\perp^2 | k' \rangle = \rho_*^2 \langle k | \left(\frac{\partial^2}{\partial \rho^2} + \frac{1}{\rho}\frac{\partial}{\partial \rho} - \frac{m'^2}{\rho^2} \right) | k' \rangle = -k_\perp^2 \delta_{kk'}$$

where $k_\perp = \rho_* \alpha_{mp}$

$$M_{kk'} \approx \left. \frac{\hat{n}_i}{\hat{T}_e} \right|_{\rho=\rho_{mn}} \delta_{kk'} - \hat{n}_i \big|_{\rho=\rho_{mn}} L_{kk'}$$

$$J_{0,kk'} = \left(1 - \frac{1}{2} L_{kk'} \right)^{-1}$$

$$\overline{D}_{c,kk'} = \overline{\mu}_1 L_{kk'} - \overline{\mu}_2 L_{kk'}^2$$

$$L\tilde{\varphi} = \tilde{\nabla}_\perp^2 \tilde{\varphi}$$

$$M\tilde{\varphi} = \left(\frac{\hat{n}_i}{\hat{T}_e} - \hat{n}_i \tilde{\nabla}_\perp^2 \right) \tilde{\varphi} = \tilde{\Omega}_i$$

$$J_0 \tilde{\varphi} = \left(1 - \frac{1}{2} \tilde{\nabla}_\perp^2 \right)^{-1} \tilde{\varphi} = \tilde{\Psi}$$

$$\overline{D}_c F = \overline{\mu}_1 \tilde{\nabla}_\perp^2 F - \overline{\mu}_2 \tilde{\nabla}_\perp^4 F$$

Matrix ($k_{\|}$, D_{glf}) involving parallel wavenumber

$$\hat{F}k_{\|,kk'} = i\langle k|\hat{F}\hat{\nabla}_{\|}|k'\rangle = \delta_{mm'}\delta_{nn'}\frac{a}{R}\int_0^1 \hat{F}(\rho)\left(\frac{m}{q(\rho)} - n\right)W_{mnp}(\rho)W_{mnp'}(\rho)\rho d\rho$$

$$\overline{D}_{glf,kk'} = -\langle k|\sqrt{\frac{8\hat{T}_i}{\pi}}\left|\hat{\nabla}_{\|}\right||k'\rangle = -\delta_{mm'}\delta_{nn'}\sqrt{\frac{8}{\pi}}\frac{a}{R}\int_0^1 \sqrt{\hat{T}_i(\rho)}\left|\frac{m}{q(\rho)} - n\right|W_{mnp}(\rho)W_{mnp'}(\rho)\rho d\rho$$

Matrix (G) for ExB and diamagnetic drifts

$$G_{\hat{F}}\tilde{\varphi} = i\left[\frac{\hat{F}}{\rho_*}, \tilde{\varphi}\right]:$$

$$G_{\hat{F},kk'} = i\langle k|\left[\frac{\hat{F}}{\rho_*}, \quad\right]|k'\rangle = i\rho_*\langle k|\frac{1}{\rho}\frac{\partial\hat{F}}{\partial\rho}\frac{\partial}{\partial\theta}|k'\rangle = -\delta_{mm'}\delta_{nn'}\rho_* m\int_0^1 \frac{\partial\hat{F}}{\partial\rho}W_{mnp}(\rho)W_{mnp'}(\rho)d\rho$$

Matrix (a_W, b_W) involving curvature

$$\hat{\beta}\omega_{di}\hat{\alpha}\left|k'\right\rangle = -2i\rho_* \frac{a}{R}\hat{\beta}\left(\frac{1}{\rho}\cos\theta\frac{\partial}{\partial\theta} + \sin\theta\frac{\partial}{\partial\rho}\right)\hat{\alpha}\left|k'\right\rangle$$

$$= \rho_* \frac{a}{R}\left[\left(\frac{m'}{\rho}\hat{\beta}\hat{\alpha}W_{k'} + \hat{\beta}\frac{\partial\hat{\alpha}W_{k'}}{\partial\rho}\right)e^{i(m'-1)\theta} + \left(\frac{m'}{\rho}\hat{\beta}\hat{\alpha}W_{k'} - \hat{\beta}\frac{\partial\hat{\alpha}W_{k'}}{\partial\rho}\right)e^{i(m'+1)\theta}\right]e^{-in'\zeta}$$

$$\Rightarrow \quad \left\langle k\left|\hat{\beta}\omega_{di}\hat{\alpha}\right|k'\right\rangle = \rho_* \frac{a}{R}\left[\begin{array}{l}\left\{(m+1)\int_0^1 \hat{\beta}\hat{\alpha}W_{mnp}W_{m+1,n,p'}d\rho + \int_0^1 \hat{\beta}W_{mnp}\frac{\partial\hat{\alpha}W_{m+1,n,p'}}{\partial\rho}\rho d\rho\right\}\delta_{m+1,m'}\delta_{nn'} \\ +\left\{(m-1)\int_0^1 \hat{\beta}\hat{\alpha}W_{mnp}W_{m-1,n,p'}d\rho - \int_0^1 \hat{\beta}W_{mnp}\frac{\partial\hat{\alpha}W_{m-1,n,p'}}{\partial\rho}\rho d\rho\right\}\delta_{m-1,m'}\delta_{nn'}\end{array}\right]$$

$$\Downarrow$$

$$a_{W,kk'} = \left\langle k\left|\left(\hat{n}_i\omega_{di} + \omega_{di}\frac{\hat{p}_i}{\hat{T}_e}\right)\right|k'\right\rangle = a_W^+(k,p')\delta_{m+1,m'}\delta_{nn'} + a_W^-(k,p')\delta_{m-1,m'}\delta_{nn'}$$

$$b_{W,kk'} = \left\langle k\left|\omega_{di}\hat{n}_i\right|k'\right\rangle = b_W^+(k,p')\delta_{m+1,m'}\delta_{nn'} + b_W^-(k,p')\delta_{m-1,m'}\delta_{nn'}$$

$$a_W\tilde{\varphi} = \hat{n}_i\omega_{di}\tilde{\varphi} + \omega_{di}\frac{\hat{p}_i}{\hat{T}_e}\tilde{\varphi}$$

$$b_W\tilde{T}_i = \omega_{di}\hat{n}_i\tilde{T}_i$$

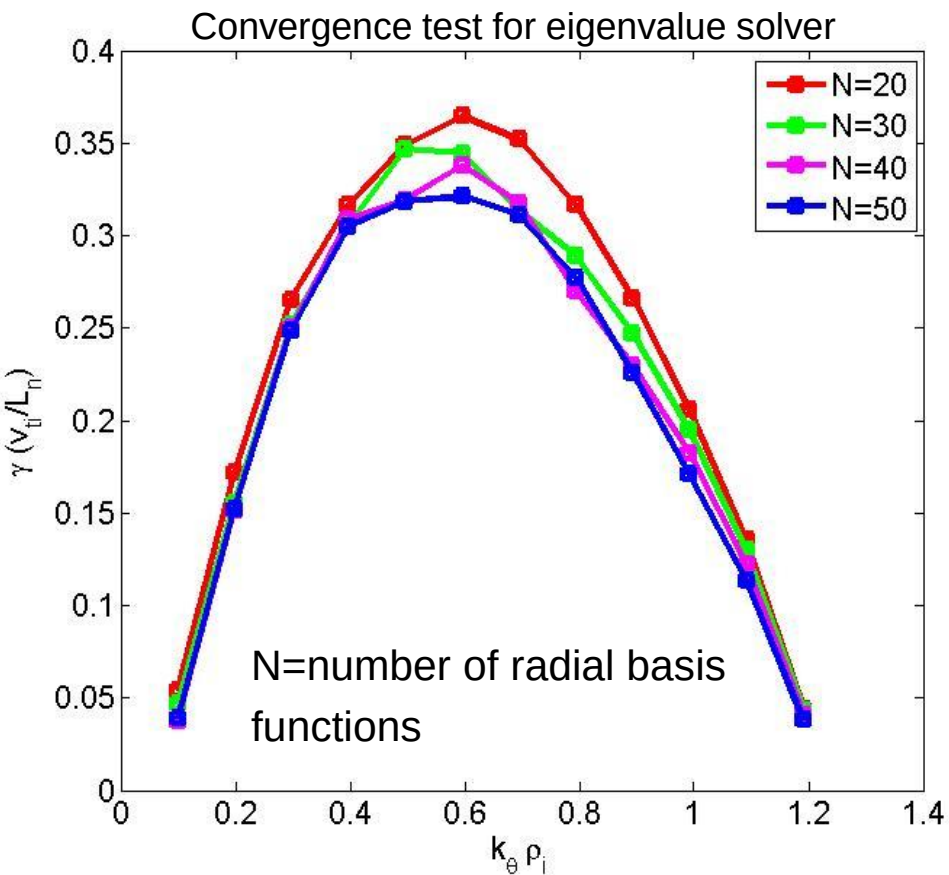
where

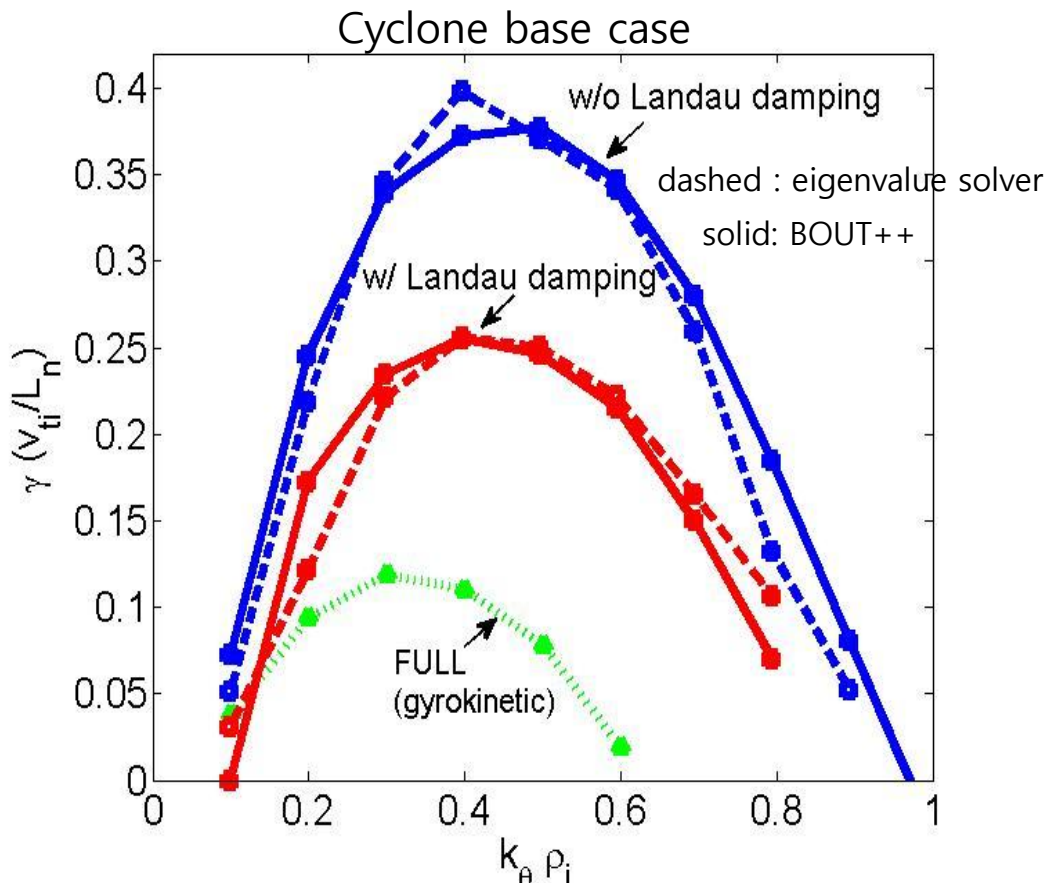
$$a_W^\pm(k,p') = \rho_* \frac{a}{R}\left[\int_0^1 \hat{n}_i\left\{(m\pm1)(1+\tau) \pm \left(\frac{\partial\ln\hat{n}_i}{\partial\rho} + \frac{\partial\ln\tau}{\partial\rho}\right)\tau\right\}W_{mnp}W_{m\pm1,n,p'}d\rho \pm \int_0^1 \hat{n}_i(1+\tau)W_{mnp}\frac{\partial W_{m\pm1,n,p'}}{\partial\rho}\rho d\rho\right]$$

$$b_W^\pm(k,p') = \rho_* \frac{a}{R}\left[\int_0^1\left(m\pm1+\frac{\partial\ln\hat{n}_i}{\partial\rho}\right)\hat{n}_i W_{mnp}W_{m\pm1,n,p'}d\rho \pm \int_0^1 \hat{n}_i W_{mnp}\frac{\partial W_{m\pm1,n,p'}}{\partial\rho}\rho d\rho\right], \quad \tau = \frac{\hat{T}_i}{\hat{T}_e}$$

Comparison of BOUT++ with eigenvalue solver

- In eigenvalue solver, $W_{mnp} = \frac{\sqrt{2}}{J_{m+1}(\alpha_{mp})} J_m\left(\frac{\alpha_{mp}}{a} r\right) e^{i(m\theta - n\zeta)}$ $(p = 1 \sim N)$ is used as basis function.
- The fluid equations are then projected on to the set of basis functions.
- Eigenvalues are obtained by using matlab.
- BOUT++ results agree well with eigenvalues, which means BOUT++ correctly solves the given equations.





Gyrofluid Simulation using BOUT++ GLF Code in Core Region

Gyrofluid equation for ITG

(M. A. Beer and G. W. Hammett, PoP '96)

- **Guiding center density equation**

$$\frac{\partial \tilde{n}_i}{\partial t}+\mathbf{V}_{\tilde{\Phi}}\cdot\nabla\tilde{n}_i+\frac{n_0}{2T_{i0}}\left[\hat{\nabla}_\perp^2\mathbf{V}_{\tilde{\Phi}}\right]\cdot\nabla\tilde{T}_{i\perp}+B\nabla_\parallel\frac{n_0\tilde{V}_{i\parallel}}{B}-\frac{en_0}{T_{i0}}\left(1+\frac{1}{2}\eta_i\hat{\nabla}_\perp^2\right)i\omega_*\tilde{\Phi}+\frac{en_0}{T_{i0}}\left(2+\frac{1}{2}\hat{\nabla}_\perp^2\right)i\omega_d\tilde{\Phi}+\frac{1}{T_{i0}}i\omega_d\left(\tilde{p}_{i\parallel}+\tilde{p}_{i\perp}\right)=0$$

- **Guiding center parallel velocity equation**

$$\frac{\partial \tilde{V}_{i\parallel}}{\partial t}+\mathbf{V}_{\tilde{\Phi}}\cdot\nabla\tilde{V}_{i\parallel}+\frac{e}{m_i}\nabla_\parallel\tilde{\Phi}+\frac{B}{m_in_0}\nabla_\parallel\frac{\tilde{p}_{i\parallel}}{B}+\left(\frac{\tilde{T}_{i\perp}}{m_i}+\frac{e}{2m_i}\hat{\nabla}_\perp^2\tilde{\Phi}\right)\nabla_\parallel\ln B+\frac{1}{p_{i0}}i\omega_d\left(\tilde{q}_{i\parallel}+\tilde{q}_{i\perp}\right)+4i\omega_d\tilde{V}_{i\parallel}=0$$

- **Guiding center parallel & perpendicular pressure equation**

$$\frac{\partial \tilde{p}_{i\parallel}}{\partial t}+\mathbf{V}_{\tilde{\Phi}}\cdot\nabla\tilde{p}_{i\parallel}+\frac{n_0}{2}\left[\hat{\nabla}_\perp^2\mathbf{V}_{\tilde{\Phi}}\right]\cdot\nabla\tilde{T}_{i\perp}+B\nabla_\parallel\frac{1}{B}\left(\tilde{q}_{i\parallel}+p_{i0}\tilde{V}_{i\parallel}\right)+2p_{i0}B\nabla_\parallel\frac{\tilde{V}_{i\parallel}}{B}+2\tilde{q}_{i\perp}\nabla_\parallel\ln B$$
$$-en_0\left(1+\eta_i+\frac{1}{2}\eta_i\hat{\nabla}_\perp^2\right)i\omega_*\tilde{\Phi}+en_0\left(4+\frac{1}{2}\hat{\nabla}_\perp^2\right)i\omega_d\tilde{\Phi}+4i\omega_d\tilde{p}_{i\parallel}+n_0\,i\omega_d\left(3\tilde{T}_{i\parallel}+\tilde{T}_{i\perp}\right)+2n_0\left|\omega_d\right|\left(\nu_1\tilde{T}_{i\parallel}+\nu_2\tilde{T}_{i\perp}\right)=0$$

$$\frac{\partial \tilde{p}_{i\perp}}{\partial t}+\mathbf{V}_{\tilde{\Phi}}\cdot\nabla\tilde{p}_{i\perp}+\frac{1}{2}\left[\hat{\nabla}_\perp^2\mathbf{V}_{\tilde{\Phi}}\right]\cdot\nabla\tilde{p}_{i\perp}+n_0\left[\hat{\hat{\nabla}}_\perp^2\mathbf{V}_{\tilde{\Phi}}\right]\cdot\nabla\tilde{T}_{i\perp}+B^2\nabla_\parallel\frac{1}{B^2}\left(\tilde{q}_{i\perp}+p_{i0}\tilde{V}_{i\parallel}\right)$$
$$-en_0\left[1+\frac{1}{2}\hat{\nabla}_\perp^2+\eta_i\left(1+\frac{1}{2}\hat{\nabla}_\perp^2+\hat{\hat{\nabla}}_\perp^2\right)\right]i\omega_*\tilde{\Phi}+en_0\left(3+\frac{3}{2}\hat{\nabla}_\perp^2+\hat{\hat{\nabla}}_\perp^2\right)i\omega_d\tilde{\Phi}+3i\omega_d\tilde{p}_{i\perp}+n_0i\omega_d\left(\tilde{T}_{i\parallel}+2\tilde{T}_{i\perp}\right)+2n_0\left|\omega_d\right|\left(\nu_3\tilde{T}_{i\parallel}+\nu_4\tilde{T}_{i\perp}\right)=0$$

- **Quasi-neutrality**

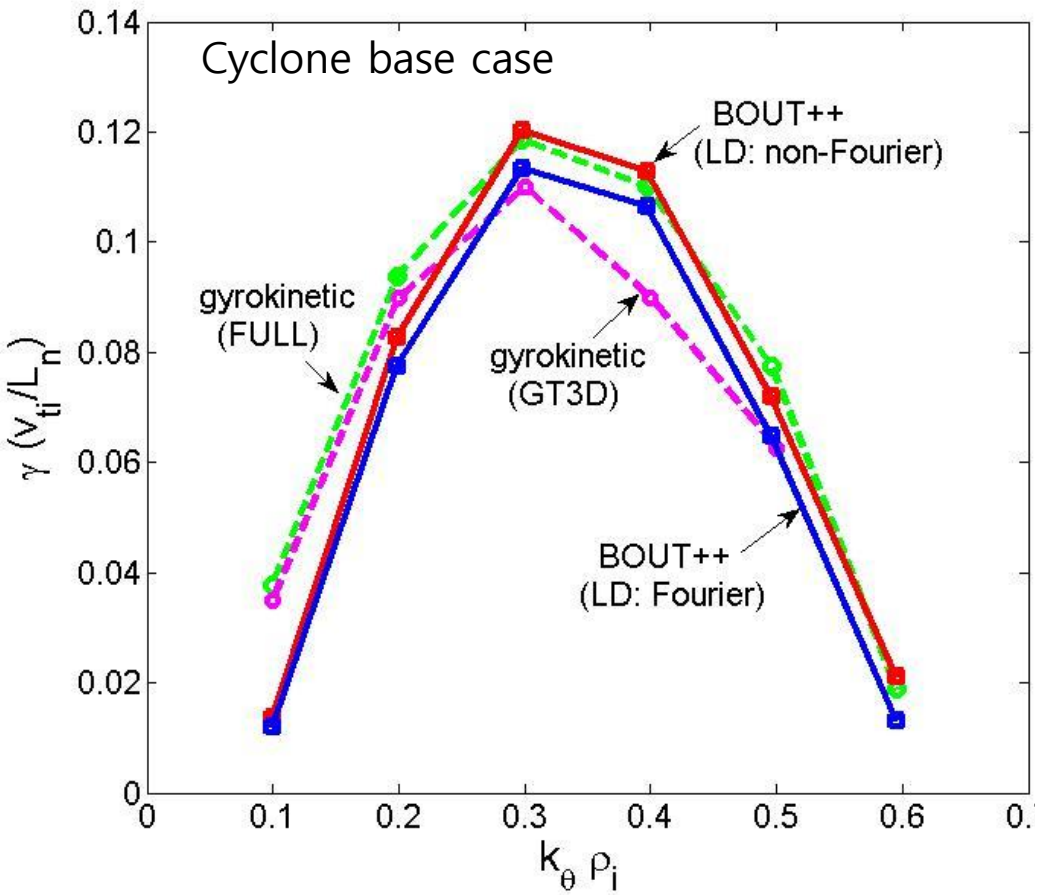
$$\frac{n_0}{T_{e0}}\tilde{\varphi}+\frac{n_0}{T_{i0}}\left(1-\Gamma_0\right)\tilde{\varphi}=\Gamma_0^{1/2}\tilde{n}_i+\frac{n_0}{T_{i0}}b\frac{\partial\Gamma_0^{1/2}}{\partial b}\tilde{T}_i$$

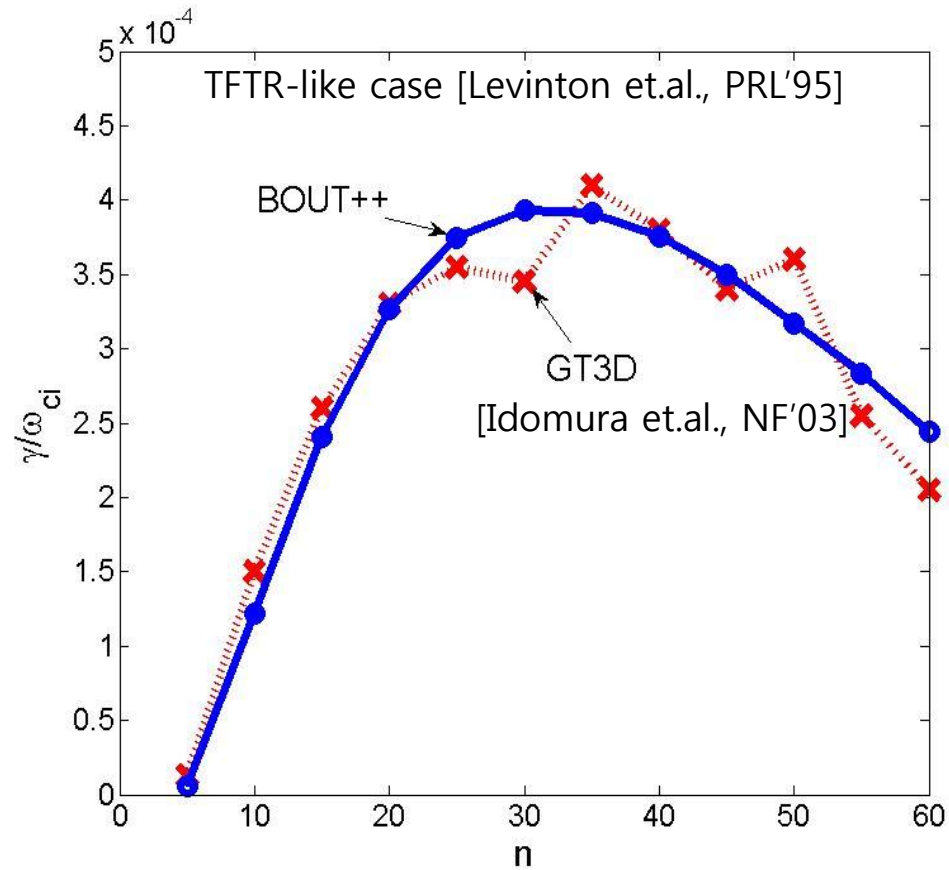
where $\tilde{\Phi}=\Gamma_0^{1/2}\tilde{\varphi}$, $\frac{1}{2}\hat{\nabla}_\perp^2\tilde{\Phi}=\left(b\frac{\partial\Gamma_0^{1/2}}{\partial b}\right)\tilde{\varphi}$, $\hat{\hat{\nabla}}_\perp^2\tilde{\Phi}=b\frac{\partial^2}{\partial b^2}\left(b\Gamma_0^{1/2}\right)\tilde{\varphi}$, $\Gamma_0^{1/2}\cong\frac{1}{1+b/2}$

$$i\omega_d=\left(v_t^2/\omega_c\right)\mathbf{b}\times\nabla\ln B\cdot\nabla,\quad b=-\rho_i^2\nabla_\perp^2$$

Linear benchmarks

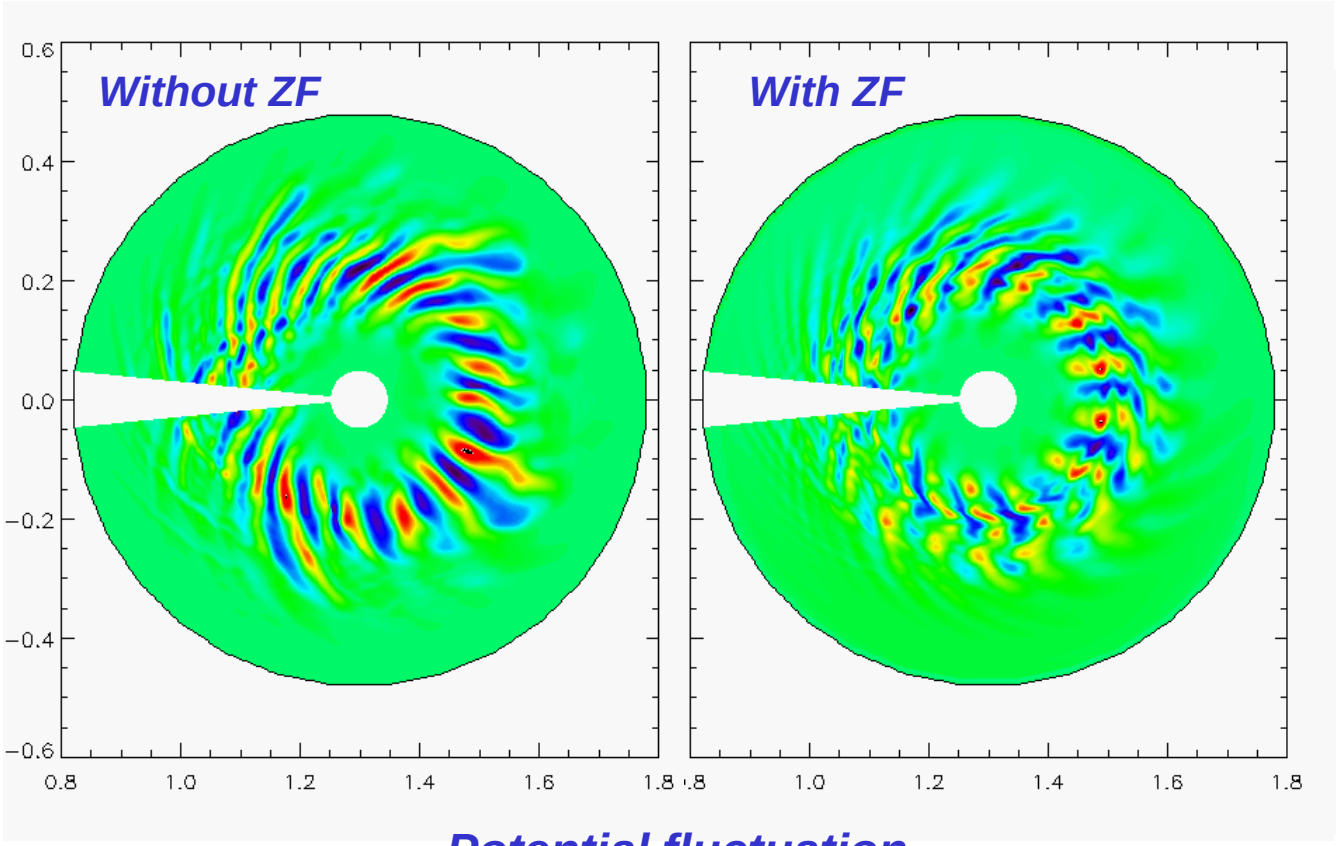
- BOUT++ using Beer's "3+1" field model agrees well with gyrokinetic codes.
- Non-Fourier method for Landau damping shows good agreement with Fourier method.



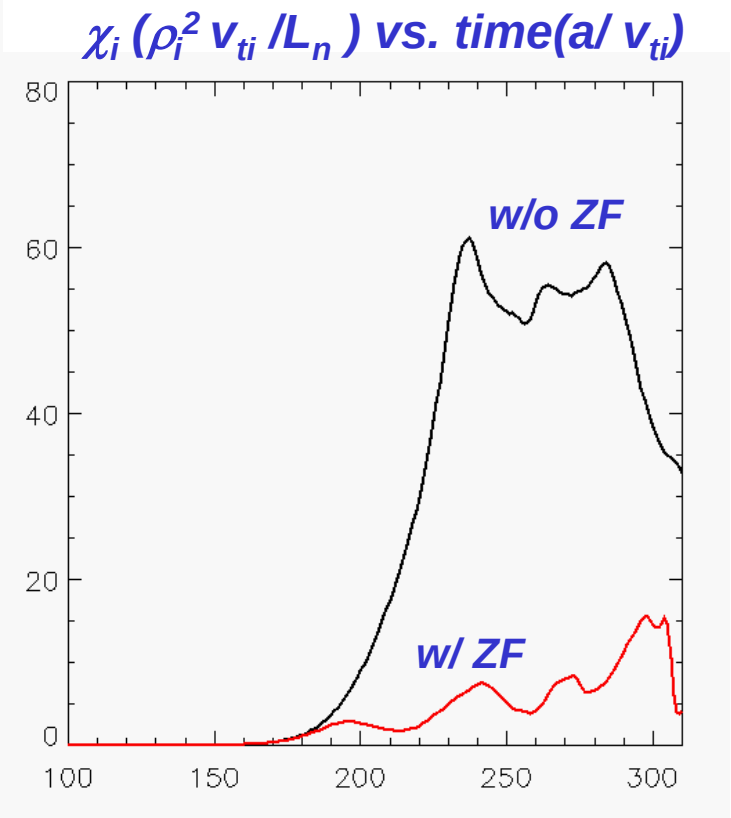


Nonlinear simulations of ITG

- Global nonlinear simulations using Beer model performed at fixed profile
- Turbulence suppression by zonal flow observed



Potential fluctuation



Hands-on Exercise for Eigenvalue Solver using MATLAB

0) Get eigenvalue solver and Run matlab :

1) Run eigensolver_init.m and eigensolver_ITG.m in matlab:

>> eigensolver_init : assign equilibrium profiles, calculate radial basis functions,
arrange mode index, evaluate matrix elements involving integrals

>> eigensolver_ITG : set up matrix and solve the eigenvalue equation for ITG

2) Run plot_eigenvalue.m and plot_eigenmode.m in matlab:

>> plot_eigenvalue : plot linear ITG growth rate and real frequency vs. k_theta*rho_i
>> plot_eigenmode : choose toroidal mode number and plot eigenmode structure

Get eigenvalue solver using git and run matlab :

```
> mkdir eigensolver-ex
> cd eigensolver-ex
> git clone https://github.com/boutproject/BOUT-2.0.git
> cd BOUT-2.0/tools/eigensolver/GLF_ITG
> module load matlab-nofonts
> matlab
```

File Edit Debug Parallel Desktop Window Help

Current Folder: /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG

Shortcuts How to Add What's New

Current Folder

« GLF_ITG

Name

- eigensolver_init.m
- eigensolver_ITG.m
- eigensolver_ITG.pdf
- plot_eigenmode.m
- plot_eigenvalue.m
- README

Details

Select a file to view details

Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
>>
```

Workspace

Select dat...

Name	Value

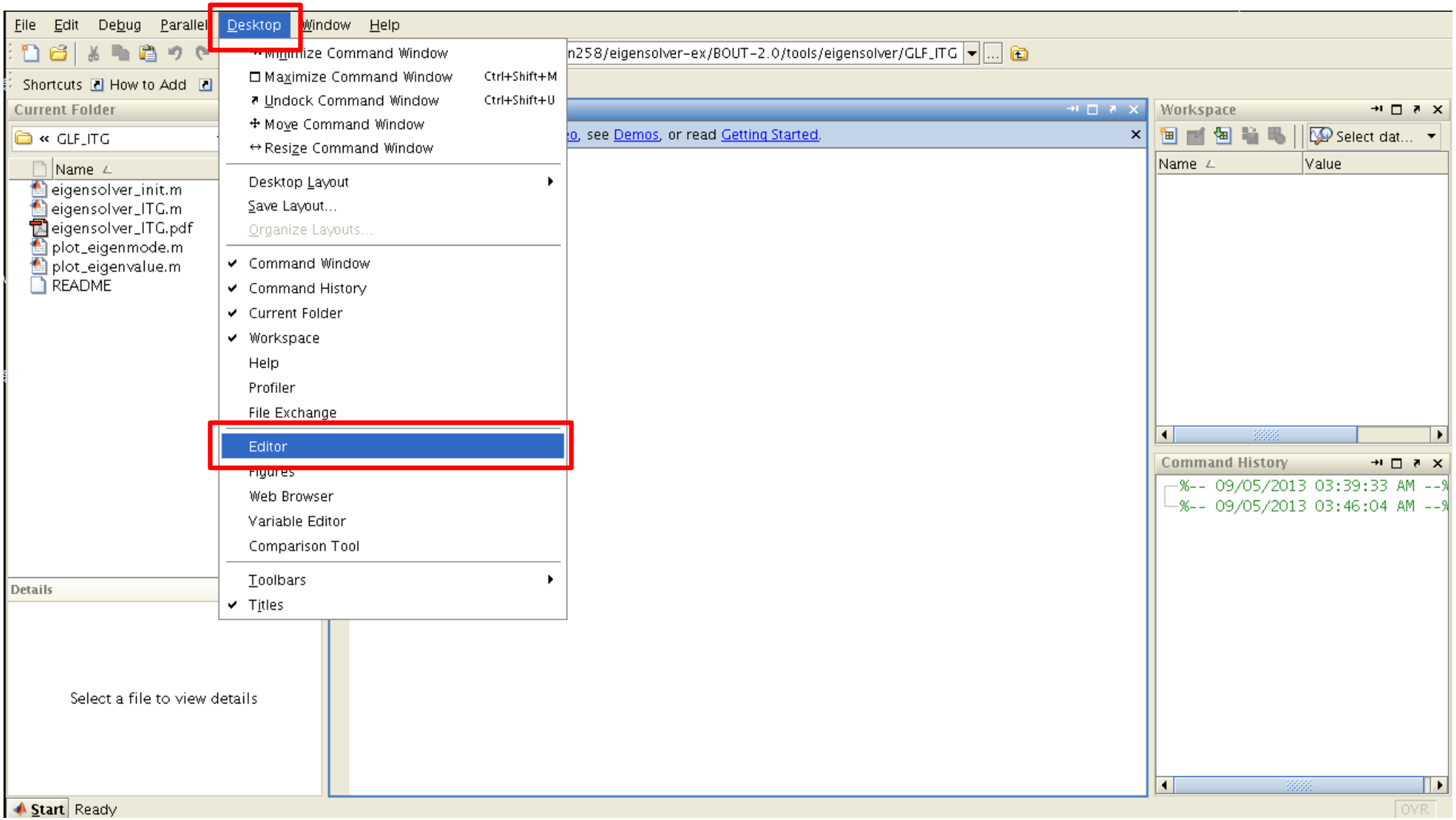
Command History

```
%-- 09/05/2013 03:39:33 AM --%
```

Start Ready

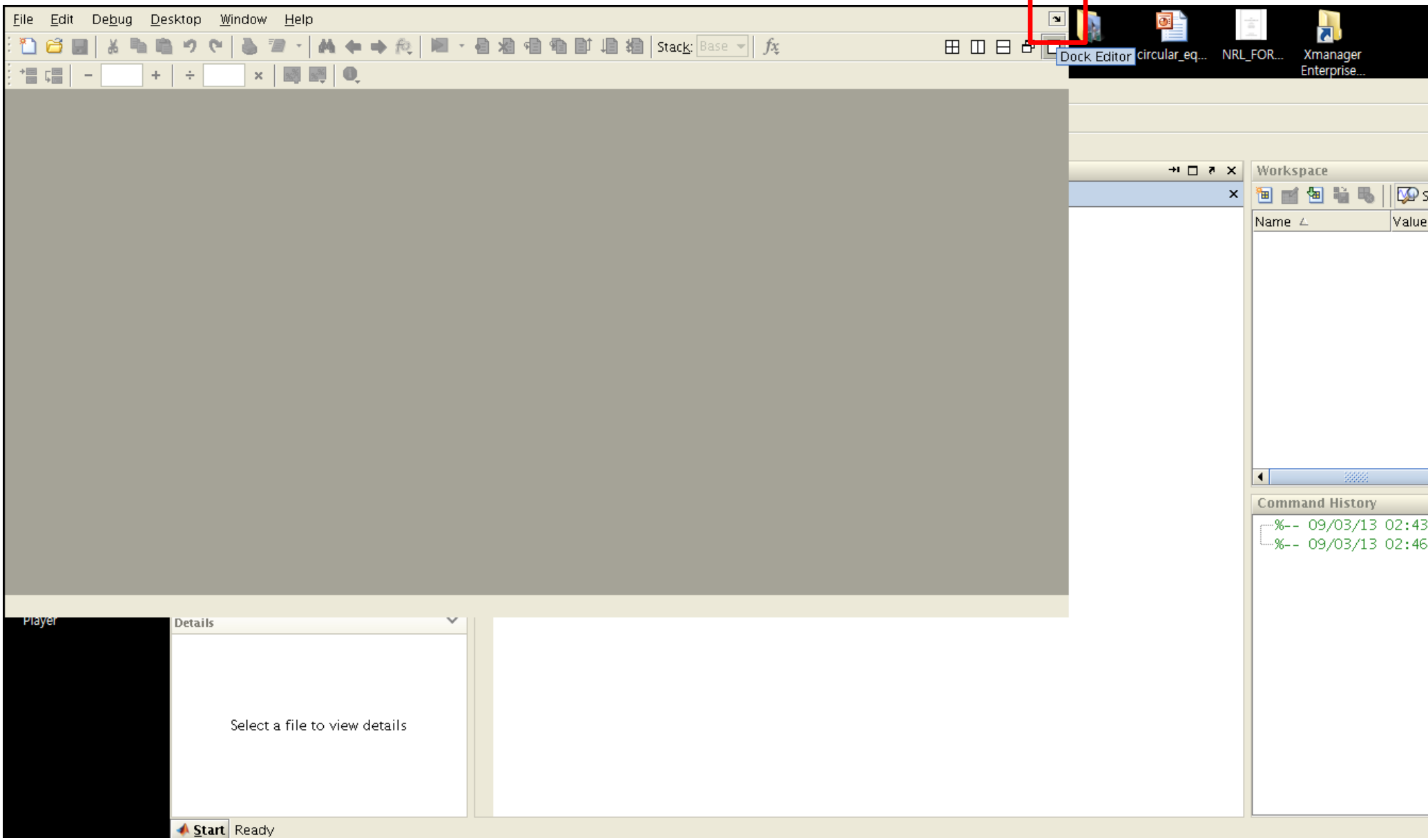
OVR

Open Editor window:



Dock Editor window:

Push this button



File Edit Debug Parallel Desktop Window Help

Current Folder: /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG

Shortcuts How to Add What's New

Current Folder

« GLF_ITG

Name
eigensolver_init.m
eigensolver_ITG.m
eigensolver_ITG.pdf
plot_eigenmode.m
plot_eigenvalue.m
README

Details

Select a file to view details

Editor

Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

>>

Workspace

Select dat...

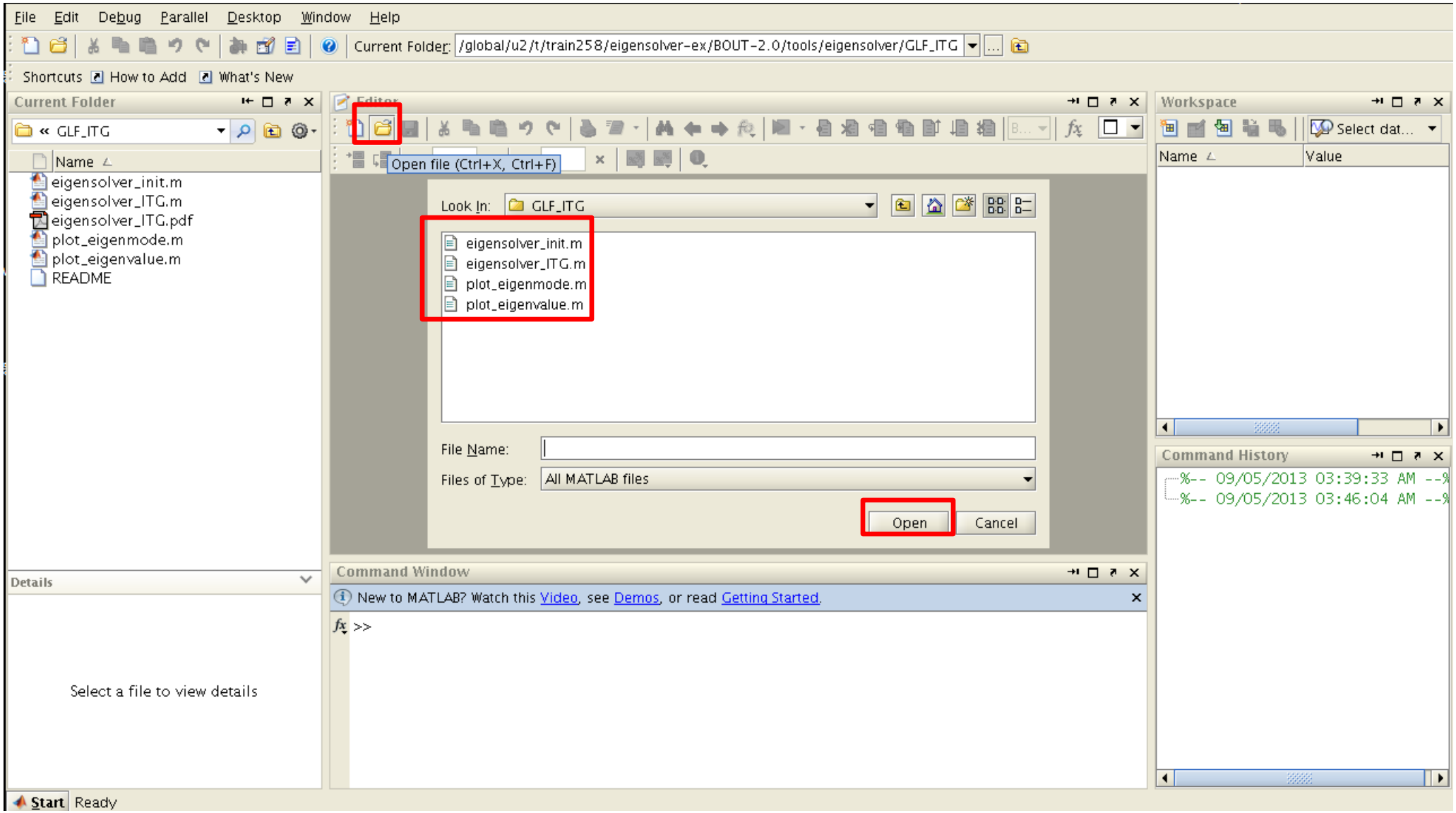
Name	Value

Command History

```
%-- 09/05/2013 03:39:33 AM --%
%-- 09/05/2013 03:46:04 AM --%
```

Start Ready

Open eigensolver_init.m, eigensolver_ITG.m, plot_eigenmode.m, plot_eigenvalue.m :



Run eigensolver_init.m: it takes about 5 min when nmax=15.

File Edit Text Go Cell Tools Debug Parallel Desktop Window Help

Current Folder: /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG

Shortcuts How to Add What's New

Current Folder

« GLF_ITG

Name
- eigensolver_init.m
- eigensolver_ITG.m
- eigensolver_ITG.pdf
- plot_eigenmode.m
- plot_eigenvalue.m
- README

Details

Select a file to view details

Editor - /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG/eigensolv...

Run eigensolver_init.m (F5)

```
7     %==================================================================
8
9 -         jmax=256; % number of radial grids
10 -        nmax=15;  % maximum of toroidal mode number
11 -        nmin=5;   % minimum of toroidal mode number
12 -        ndel=5;   % toroidal mode number spacing
13 -        mmax=150; % maximum of poloidal mode number
14 -        mmin=1;   % minimum of poloidal mode number
15 -        mdel=1;   % poloidal mode number spacing
16 -        kyrhoscut=1.3;  % maximal ktheta*rhoi allowed in the calculation
17 -        xmu1=0.1; % viscous damping coefficient
18 -        xmu2=0.0; % hyper-viscous damping coefficient
19 -        r0=0.2;   % inner boundary
20 -        r1=0.9;   % outer boundary
21
22 -        hermite=1;
23 -        bessel=2;
24 -        basis=bessel;     % choose radial basis function
```

eigensolver_init.m | eigensolver_ITG.m | plot_eigenmode.m | plot_eigenvalue.m

Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
status =

   Calculating radial basis functions:


status =

   ... finding zeros of Bessel functions
```

Workspace

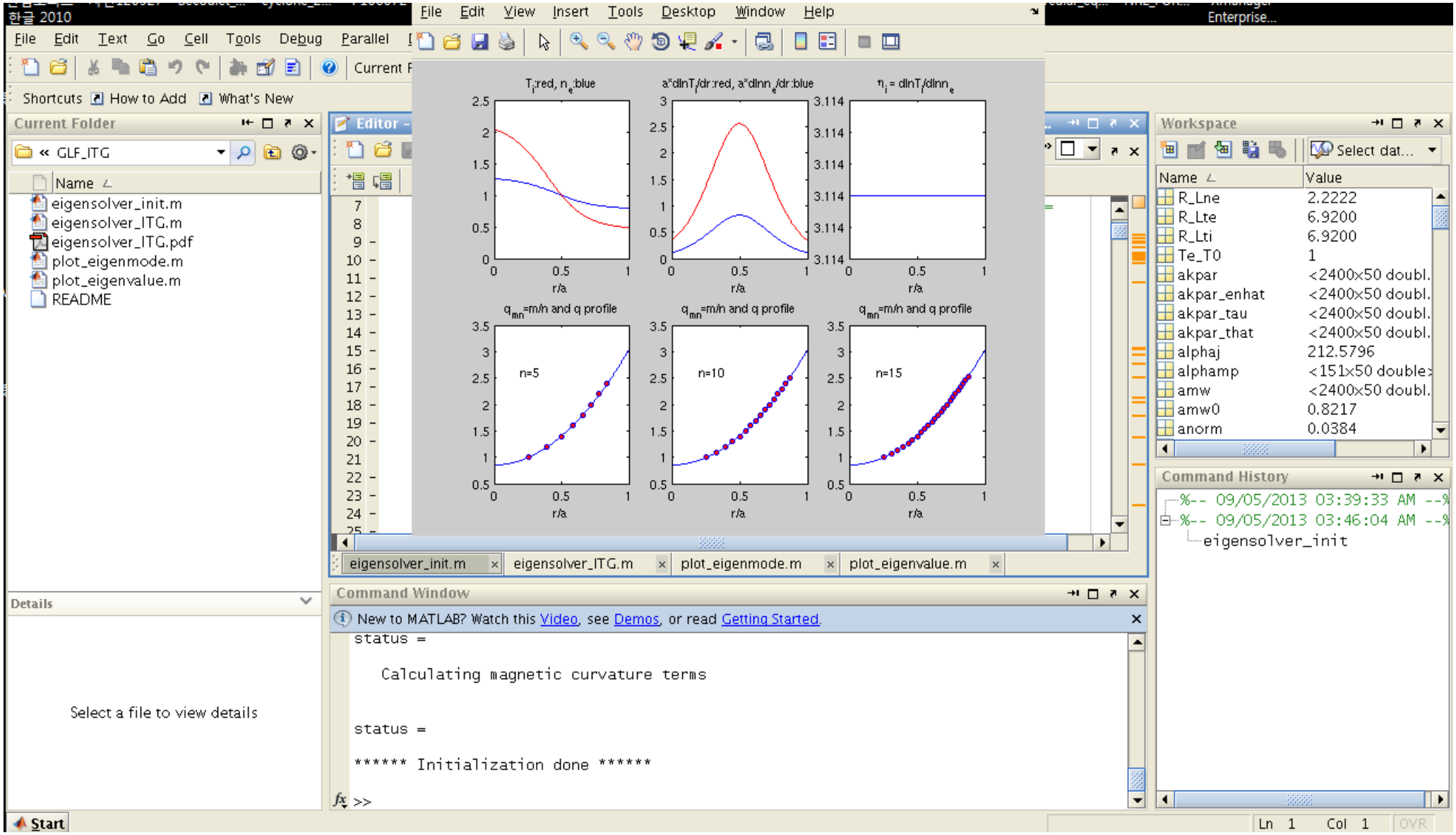
Select dat...

Name	Value

Command History

```
%-- 09/05/2013 03:39:33 AM --%
%-- 09/05/2013 03:46:04 AM --%
    eigensolver_init
```

Start Busy Ln 1 Col 1 OVR


File Edit View Insert Tools Desktop Window Help
File Edit Text Go Cell Tools Debug Parallel
Shortcuts How to Add What's New
Current Folder
GLF_ITG
Name
eigensolver_init.m
eigensolver_ITG.m
eigensolver_ITG.pdf
plot_eigenmode.m
plot_eigenvalue.m
README
Details
Select a file to view details
Editor
eigensolver_init.m
eigensolver_ITG.m
plot_eigenmode.m
plot_eigenvalue.m
T_i:red, n_e:blue
a*dlnT_i/dr:red, a*dlnn_e/dr:blue
η_i = dlnT_i/dlnn_e
q_mn=m/n and q profile
n=5
n=10
n=15
r/a
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
status =
Calculating magnetic curvature terms
status =
****** Initialization done ******
>>
Workspace
Select dat...
Name
Value
R_Lne 2.2222
R_Lte 6.9200
R_Lti 6.9200
Te_T0 1
akpar <2400x50 doubl.
akpar_enhat <2400x50 doubl.
akpar_tau <2400x50 doubl.
akpar_that <2400x50 doubl.
alphaj 212.5796
alphamp <151x50 double>
amw <2400x50 doubl.
amw0 0.8217
anorm 0.0384
Command History
%-- 09/05/2013 03:39:33 AM --%
%-- 09/05/2013 03:46:04 AM --%
eigensolver_init
Start
Ln 1 Col 1 OVR

Run eigensolver_ITG.m : it takes about 10 min.

File Edit Text Go Cell Tools Debug Parallel Desktop Window Help

Current Folder: /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG

Shortcuts How to Add What's New

Current Folder

« GLF_ITG

Name
- eigensolver_init.m
- eigensolver_ITG.m
- eigensolver_ITG.pdf
- plot_eigenmode.m
- plot_eigenvalue.m
- README

Editor - /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG/eigensolv...

Run eigensolver_ITG.m (F5)

```
%==================================================================
% Solve eigenvalue equation for ITG :
% Ottaviani model used [M. Ottaviani et.al., Phys. Plasmas 6, 3267 (1999)]
%==================================================================

  gyroaverage=1; % turn-on(1) or off(0) gyroaverage
  damping_glf=1; % turn-on(1) or off(0) Landau damping term
  damping_c=0;   % turn-on(1) or off(0) viscous damping term

  l_k=0*ones(1,kmax);
  lmaxn=0*ones(1,nmax/ndel);

  l=0;
  for k=1:kmax
     if(nfb(k)==nmax)
       l=l+1;
     end
  end
  lmax=l;
```

eigensolver_init.m | eigensolver_ITG.m | plot_eigenmode.m | plot_eigenvalue.m

Workspace

Name	Value
R_Lne	2.2222
R_Lte	6.9200
R_Lti	6.9200
Te_T0	1
akpar	<2400x50 doubl.
akpar_enhat	<2400x50 doubl.
akpar_tau	<2400x50 doubl.
akpar_that	<2400x50 doubl.
alphaj	212.5796
alphamp	<151x50 double>
amw	<2400x50 doubl.
amw0	0.8217
anorm	0.0384

Command History

```
%-- 09/05/2013 03:39:33 AM --%
%-- 09/05/2013 03:46:04 AM --%
  eigensolver_init
  eigensolver_ITG
```

Command Window

New to MATLAB? Watch this Video, see Demos, or read Getting Started.

```
status =

   Setting up matrix for eigenvalue equation


status =

   Calculating eigenvalues and eigenvectors
```

Details

Select a file to view details

Start Busy script Ln 8 Col 61 OVR

Run plot_eigenvalue.m;

Current Folder: /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG

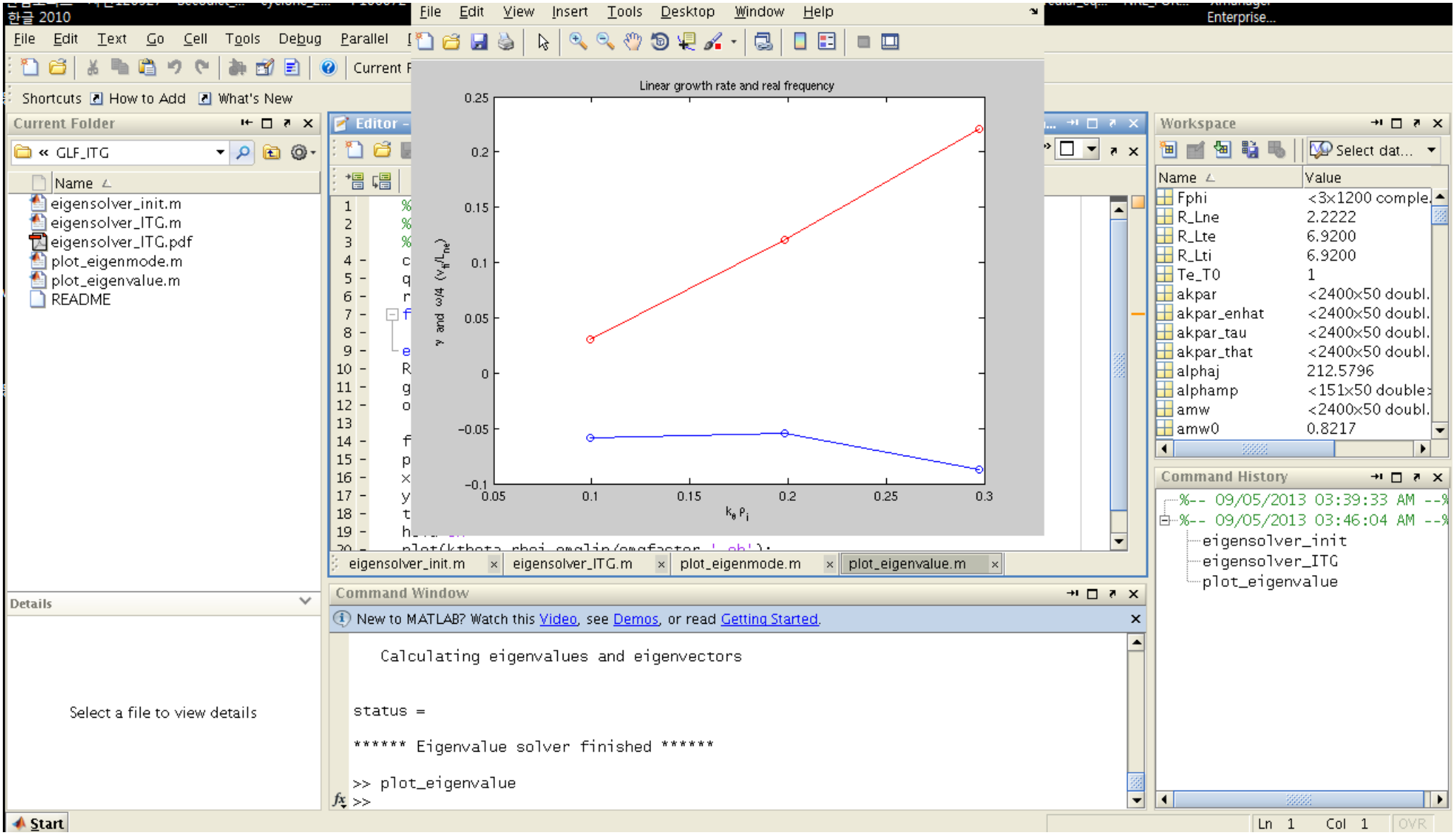
Editor - /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG/plot_eigen...

Run plot_eigenvalue.m (F5)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% plot linear growth rate and real frequency vs. ktheta*rhoi
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear ktheta_rhoi
qval=1.4;
rval=0.5;
for n1=nmin:ndel:nmax
    ktheta_rhoi(n1/ndel)=n1*qval/rval*rhos0;
end
R=1.3; a=0.48;  % for cyclone base case
gamfactor=R_Lne*a/R;  % change time unit from a/v_ti to L_ne/v_ti
omgfactor=gamfactor*4;

figure(2)
plot(ktheta_rhoi,gamlin/gamfactor,'-or');
xlabel('k_\theta \rho_i')
ylabel('\gamma  and  \omega/4  (v_{ti}/L_{ne})')
title('Linear growth rate and real frequency')
hold on
plot(ktheta_rhoi,omglin/omgfactor,'-ob');
```

Command Window

```
status =

   Calculating eigenvalues and eigenvectors


status =

****** Eigenvalue solver finished ******

>>
```


Linear growth rate and real frequency
0.25
0.2
0.15
0.1
0.05
0
−0.05
−0.1
0.05
0.1
0.15
0.2
0.25
0.3
Current Folder
GLF_ITG
Name
eigensolver_init.m
eigensolver_ITG.m
eigensolver_ITG.pdf
plot_eigenmode.m
plot_eigenvalue.m
README
Details
Select a file to view details
Editor
eigensolver_init.m
eigensolver_ITG.m
plot_eigenmode.m
plot_eigenvalue.m
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
Calculating eigenvalues and eigenvectors
status =
****** Eigenvalue solver finished ******
>> plot_eigenvalue
>>
Workspace
Name
Value
Fphi
<3x1200 comple.
R_Lne
2.2222
R_Lte
6.9200
R_Lti
6.9200
Te_T0
1
akpar
<2400x50 doubl.
akpar_enhat
<2400x50 doubl.
akpar_tau
<2400x50 doubl.
akpar_that
<2400x50 doubl.
alphaj
212.5796
alphamp
<151x50 double
amw
<2400x50 doubl.
amw0
0.8217
Command History
%-- 09/05/2013 03:39:33 AM --%
%-- 09/05/2013 03:46:04 AM --%
eigensolver_init
eigensolver_ITG
plot_eigenvalue

Run plot_eigenmode.m:

Current Folder: /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG

Editor - /global/u2/t/train258/eigensolver-ex/BOUT-2.0/tools/eigensolver/GLF_ITG/plot_eigen...

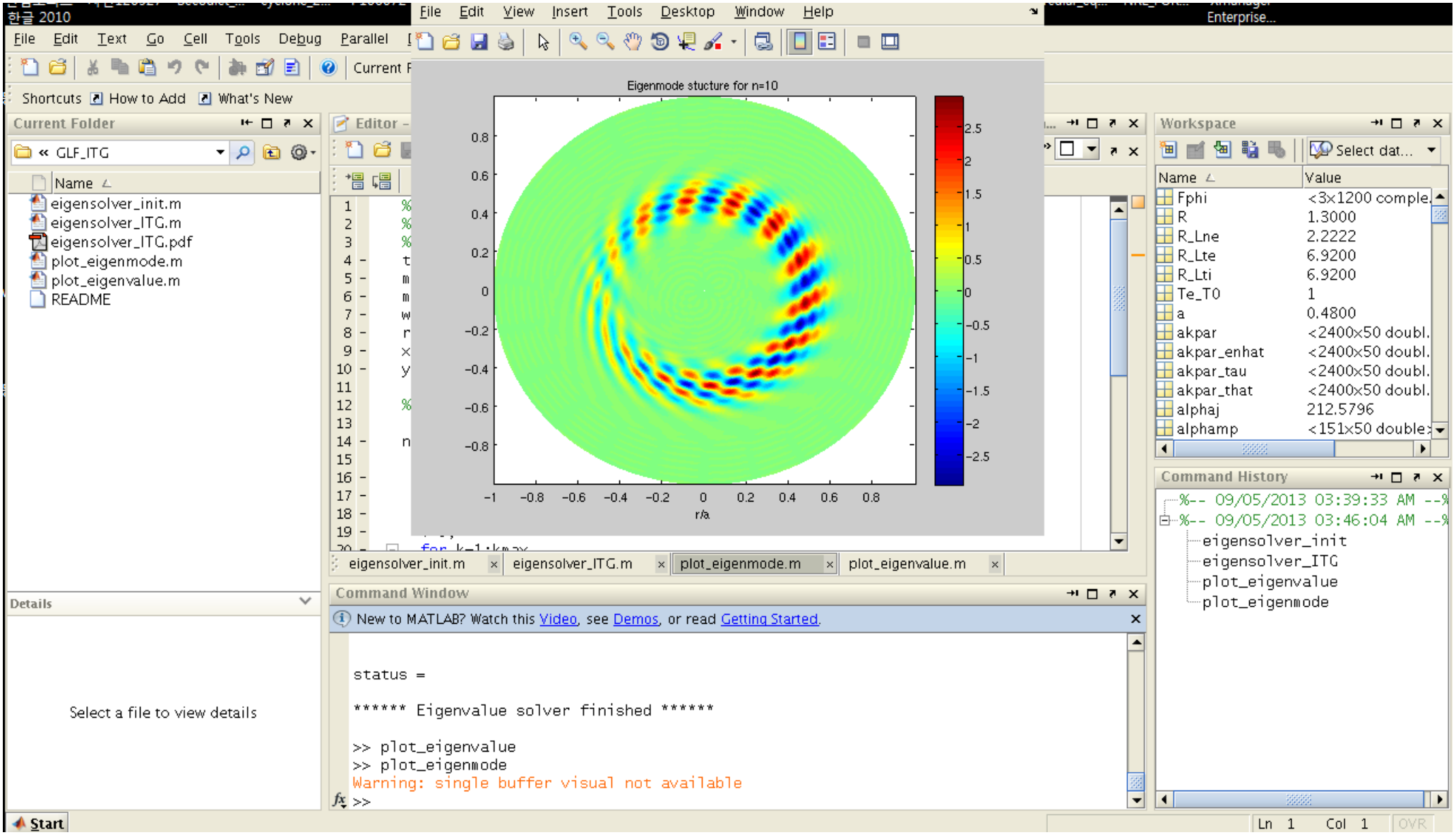
Run plot_eigenmode.m (F5)

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Contour plot of eigenmode structure for potential
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
theta=-pi:0.01:pi;
mft=mf';
matexp=exp(i*mft(1:kmax)*theta);
wavemap=wave(:,1:kmax);
rcore=r';
x=rcore*cos(theta);
y=rcore*sin(theta);

%for n1=nmin:ndel:nmax

n1=10; % choose toroidal mode number

  nn=n1/ndel;
  figure(2+nn)
  ph=0*ones(kmax,1);
  l=0;
```

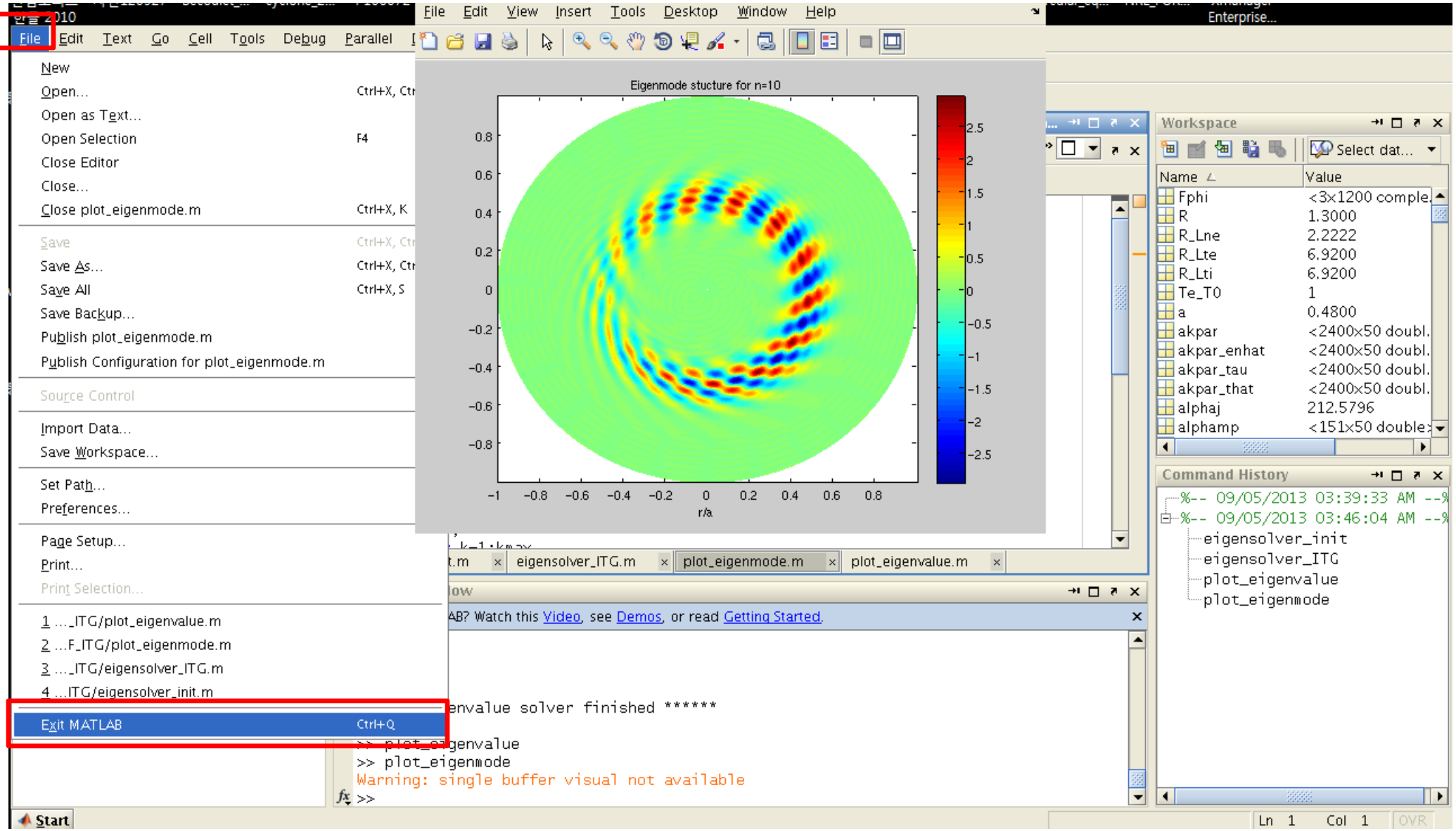
You can choose toroidal mode number.

Command Window

```
Calculating eigenvalues and eigenvectors


status =

****** Eigenvalue solver finished ******

>> plot_eigenvalue
>>
```


Eigenmode stucture for n=10
r/a
Current Folder
GLF_ITG
eigensolver_init.m
eigensolver_ITG.m
eigensolver_ITG.pdf
plot_eigenmode.m
plot_eigenvalue.m
README
Details
Select a file to view details
Editor
eigensolver_init.m
eigensolver_ITG.m
plot_eigenmode.m
plot_eigenvalue.m
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.
status =
****** Eigenvalue solver finished ******
>> plot_eigenvalue
>> plot_eigenmode
Warning: single buffer visual not available
Workspace
Fphi <3x1200 comple.
R 1.3000
R_Lne 2.2222
R_Lte 6.9200
R_Lti 6.9200
Te_T0 1
a 0.4800
akpar <2400x50 doubl.
akpar_enhat <2400x50 doubl.
akpar_tau <2400x50 doubl.
akpar_that <2400x50 doubl.
alphaj 212.5796
alphamp <151x50 double
Command History
%-- 09/05/2013 03:39:33 AM --%
%-- 09/05/2013 03:46:04 AM --%
eigensolver_init
eigensolver_ITG
plot_eigenvalue
plot_eigenmode

Exit matlab:


File Edit Text Go Cell Tools Debug Parallel
New
Open... Ctrl+X, Ctr
Open as Text...
Open Selection F4
Close Editor
Close...
Close plot_eigenmode.m Ctrl+X, K
Save Ctrl+X, Ctr
Save As... Ctrl+X, Ctr
Save All Ctrl+X, S
Save Backup...
Publish plot_eigenmode.m
Publish Configuration for plot_eigenmode.m
Source Control
Import Data...
Save Workspace...
Set Path...
Preferences...
Page Setup...
Print...
Print Selection...
1 ..._ITG/plot_eigenvalue.m
2 ...F_ITG/plot_eigenmode.m
3 ..._ITG/eigensolver_ITG.m
4 ...ITG/eigensolver_init.m
Exit MATLAB Ctrl+Q
File Edit View Insert Tools Desktop Window Help
Eigenmode stucture for n=10
r/a
eigensolver_ITG.m
plot_eigenmode.m
plot_eigenvalue.m
Workspace
Select dat...
Name Value
Fphi <3x1200 comple.
R 1.3000
R_Lne 2.2222
R_Lte 6.9200
R_Lti 6.9200
Te_T0 1
a 0.4800
akpar <2400x50 doubl.
akpar_enhat <2400x50 doubl.
akpar_tau <2400x50 doubl.
akpar_that <2400x50 doubl.
alphaj 212.5796
alphamp <151x50 double
Command History
%-- 09/05/2013 03:39:33 AM --%
%-- 09/05/2013 03:46:04 AM --%
eigensolver_init
eigensolver_ITG
plot_eigenvalue
plot_eigenmode
envalue solver finished ******
>> plot_eigenvalue
>> plot_eigenmode
Warning: single buffer visual not available
Start
Ln 1 Col 1 OVR