A Newton-Type Method with Non-equivalence Deflation for Nonlinear Eigenvalue Problems Arising in Photonic Crystal Modeling

Tsung-Ming Huang
Department of Mathematics, National Taiwan Normal University

The First Annual Meeting of Applied Mathematics: Frontier Aspects of Applied Mathematics, December 6-7, 2015, Taipei, Taiwan

Joint work

Wen-Wei Lin
Department of Applied Mathematics
National Chiao-Tung University

Volker Mehrmann
Institute fur Mathematik
TU Berlin

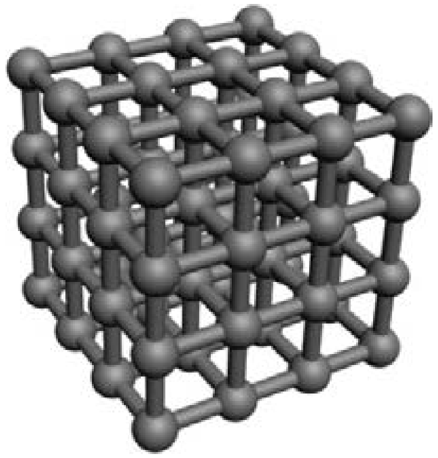
Plan

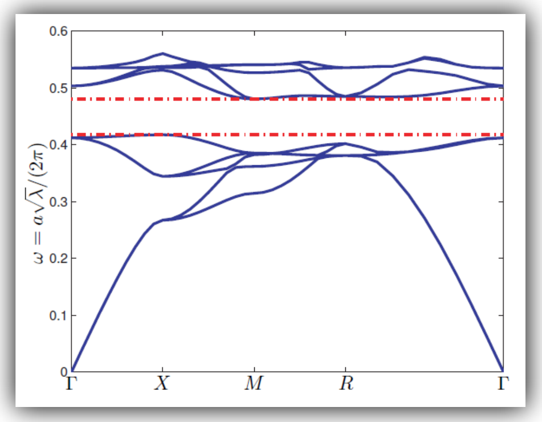
- Maxwell’s equations with dispersive metallic materials
- Nonlinear eigenvalue problems
- Newton-type method for solving nonlinear eigenvalue problems
- Numerical results

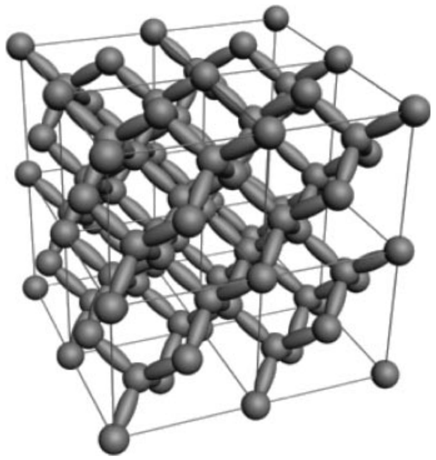
Dispersive Maxwell equations

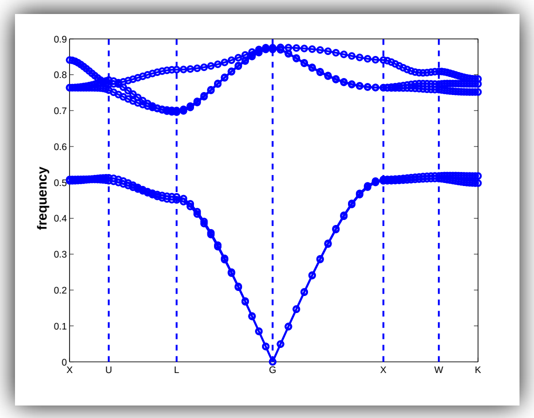
Photonic Crystals

- Periodic lattice composed of dielectric or metallic materials









- If we design a three-dimensional photonic crystal appropriately, there appears a frequency range where no electromagnetic eigenmode exists. Frequency ranges of this kind are called photonic band gaps.

- Light waves can be reflected, trapped, transported in photonic crystals.

$$\varepsilon(\mathbf{r}) = \begin{cases} \varepsilon_1, & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$

- Governing equation:

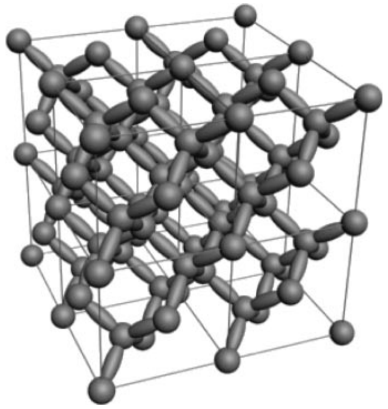
$$\nabla \times \nabla \times E(\mathbf{r}) = \omega^2 \varepsilon(\mathbf{r}) E(\mathbf{r})$$

Maxwell’s Equations for dispersive isotropic material

$$\nabla \times \nabla \times E(\mathbf{r}) = \omega^2 \varepsilon(\mathbf{r},\omega) E(\mathbf{r})$$

- $E(\mathbf{r})$ denotes the electric field at position $\mathbf{r} \in \mathbb{R}^3$
- $\varepsilon(\mathbf{r},\omega)$ denotes the permittivity, which is dependent on the position $\mathbf{r}$ and the frequency ω
- Drude model

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} 1 - \dfrac{\omega_p^2}{\omega^2 + \mathrm{i}\,\Gamma_p \omega}, & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$



- Drude-Lorentz model

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} \varepsilon_\infty - \dfrac{\omega_p^2}{\omega^2 + \mathrm{i}\,\Gamma_p \omega} + \displaystyle\sum_{j=1}^{2} \Omega_j A_j \left(\dfrac{e^{\imath\phi_j}}{\Omega_j - \omega - \mathrm{i}\,\Gamma_j} + \dfrac{e^{-\imath\phi_j}}{\Omega_j + \omega + \mathrm{i}\,\Gamma_j} \right), & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$

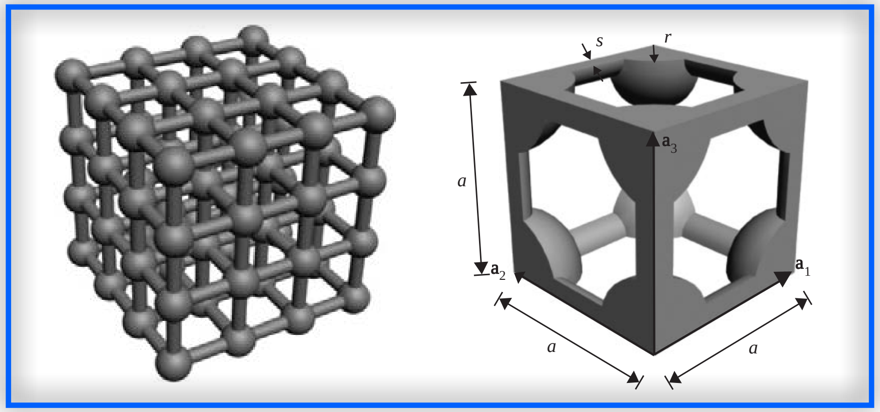
Bloch Theorem

- We are interested in finding E satisfying the quasi-periodic condition

$$E(\mathbf{r}+\mathbf{a}_\ell)=e^{i2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r}),\ \ \ell=1,2,3$$

$\mathbf{a}_1,\mathbf{a}_2,\mathbf{a}_3$ are the lattice translation vectors, k is a wave vector.

- Simple cubic (SC)

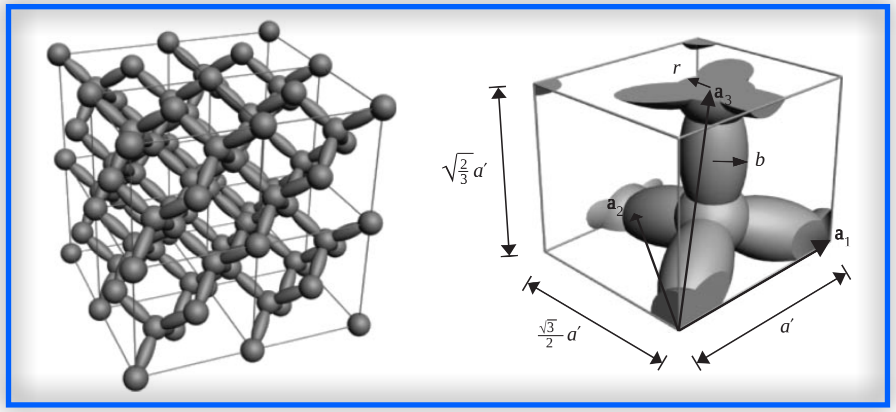


$$\mathbf{a}_1=a\begin{bmatrix}1&0&0\end{bmatrix},\quad \mathbf{a}_2=a\begin{bmatrix}0&1&0\end{bmatrix},$$

$$\mathbf{a}_3=a\begin{bmatrix}0&0&1\end{bmatrix}$$

pairwise angles formed *by these vectors are 60 degree*

- Face-centered cubic (FCC)



$$\mathbf{a}_1=\frac{a}{\sqrt{2}}[1,0,0]^\top,\ \mathbf{a}_2=\frac{a}{\sqrt{2}}\left[\frac{1}{2},\frac{\sqrt{3}}{2},0\right]^\top$$

$$\mathbf{a}_3=\frac{a}{\sqrt{2}}\left[\frac{1}{2},\frac{1}{2\sqrt{3}},\sqrt{\frac{2}{3}}\right]^\top$$

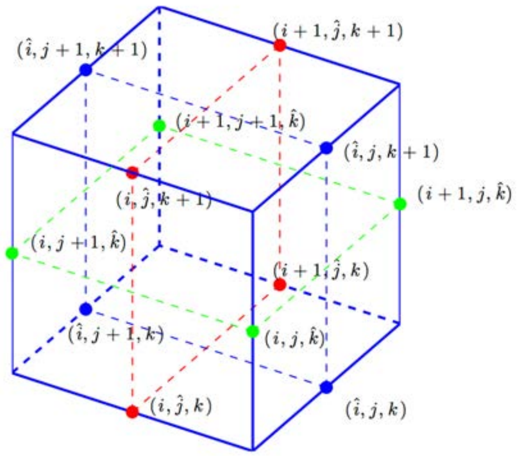
Figures taken from Chern, Chang, Chang, Hwang, 2004

Finite difference Yee's scheme

$\nabla\times\nabla\times E(\mathbf{r})=\omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})$

- Curl operator

$$\nabla\times E=\begin{bmatrix}0 & -\frac{\partial}{\partial z} & \frac{\partial}{\partial y}\\ \frac{\partial}{\partial z} & 0 & -\frac{\partial}{\partial x}\\ -\frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0\end{bmatrix}\begin{bmatrix}E_1\\ E_2\\ E_3\end{bmatrix}$$



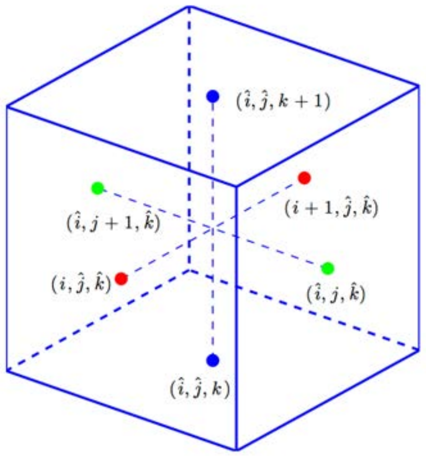
- Central edge points

$$\nabla\times H(\mathbf{r})=\omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})\quad\Rightarrow\quad C^*\mathbf{h}=\omega^2B(\omega)\mathbf{e}$$

- Central face points

$$\nabla\times E(\mathbf{r})=H(\mathbf{r})\quad\Rightarrow\quad C\mathbf{e}=\mathbf{h}$$

where



$$C=\begin{bmatrix}0 & -C_3 & C_2\\ C_3 & 0 & -C_1\\ -C_2 & C_1 & 0\end{bmatrix}\in\mathbb{C}^{3n\times3n}$$

$$n=n_1n_2n_3$$

$$\boxed{C_1=I_{n_2n_3}\otimes K_1\in\mathbb{C}^{n\times n},C_2=I_{n_3}\otimes K_2\in\mathbb{C}^{n\times n},C_3=K_3\in\mathbb{C}^{n\times n}}$$

Finite Diff. Assoc. with Quasi-Periodic Cond.

$$K_1 = \frac{1}{\delta_x}\begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_1} & & & -1 \end{bmatrix} \in \mathbb{C}^{n_1\times n_1},$$

$$K_2 = \frac{1}{\delta_y}\begin{bmatrix} -I_{n_1} & I_{n_1} & & \\ & \ddots & \ddots & \\ & & -I_{n_1} & I_{n_1} \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_2} J_2 & & & -I_{n_1} \end{bmatrix} \in \mathbb{C}^{(n_1 n_2)\times(n_1 n_2)},$$

$$K_3 = \frac{1}{\delta_z}\begin{bmatrix} -I_{n_1 n_2} & I_{n_1 n_2} & & \\ & \ddots & \ddots & \\ & & -I_{n_1 n_2} & I_{n_1 n_2} \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_3} J_3 & & & -I_{n_1 n_2} \end{bmatrix} \in \mathbb{C}^{n\times n}$$

Finite Diff. Assoc. with Quasi-Periodic Cond.

$$E(\mathbf{r}+\mathbf{a}_\ell)=e^{\mathrm{i}2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r})$$

$$K_1=\frac{1}{\delta_x}\begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \\ e^{\imath 2\pi\mathbf{k}\cdot\mathbf{a}_1} & & & -1 \end{bmatrix}\in\mathbb{C}^{n_1\times n_1},$$

$$K_2=\frac{1}{\delta_y}\begin{bmatrix} -I_{n_1} & I_{n_1} & & \\ & \ddots & \ddots & \\ & & -I_{n_1} & I_{n_1} \\ e^{\imath 2\pi\mathbf{k}\cdot\mathbf{a}_2}J_2 & & & -I_{n_1} \end{bmatrix}\in\mathbb{C}^{(n_1n_2)\times(n_1n_2)},$$

$$K_3=\frac{1}{\delta_z}\begin{bmatrix} -I_{n_1n_2} & I_{n_1n_2} & & \\ & \ddots & \ddots & \\ & & -I_{n_1n_2} & I_{n_1n_2} \\ e^{\imath 2\pi\mathbf{k}\cdot\mathbf{a}_3}J_3 & & & -I_{n_1n_2} \end{bmatrix}\in\mathbb{C}^{n\times n}$$

Finite Diff. Assoc. with Quasi-Periodic Cond.

$$K_1 = \frac{1}{\delta_x}\begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_1} & & & -1 \end{bmatrix} \in \mathbb{C}^{n_1\times n_1},$$

$$K_2 = \frac{1}{\delta_y}\begin{bmatrix} -I_{n_1} & I_{n_1} & & \\ & \ddots & \ddots & \\ & & -I_{n_1} & I_{n_1} \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_2} J_2 & & & -I_{n_1} \end{bmatrix} \in \mathbb{C}^{(n_1 n_2)\times(n_1 n_2)},$$

$$K_3 = \frac{1}{\delta_z}\begin{bmatrix} -I_{n_1 n_2} & I_{n_1 n_2} & & \\ & \ddots & \ddots & \\ & & -I_{n_1 n_2} & I_{n_1 n_2} \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_3} J_3 & & & -I_{n_1 n_2} \end{bmatrix} \in \mathbb{C}^{n\times n}$$

Finite Diff. Assoc. with Quasi-Periodic Cond.

$$K_1 = \frac{1}{\delta_x}\begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_1} & & & -1 \end{bmatrix} \in \mathbb{C}^{n_1\times n_1},$$

$$K_2 = \frac{1}{\delta_y}\begin{bmatrix} -I_{n_1} & I_{n_1} & & \\ & \ddots & \ddots & \\ & & -I_{n_1} & I_{n_1} \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_2} J_2 & & & -I_{n_1} \end{bmatrix} \in \mathbb{C}^{(n_1 n_2)\times(n_1 n_2)},$$

$$K_3 = \frac{1}{\delta_z}\begin{bmatrix} -I_{n_1 n_2} & I_{n_1 n_2} & & \\ & \ddots & \ddots & \\ & & -I_{n_1 n_2} & I_{n_1 n_2} \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_3} J_3 & & & -I_{n_1 n_2} \end{bmatrix} \in \mathbb{C}^{n\times n}$$

- For SC lattice

$$J_2 = I_{n_1}, \quad J_3 = I_{n_1 n_2}$$

Finite Diff. Assoc. with Quasi-Periodic Cond.

$$K_1 = \frac{1}{\delta_x}\begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_1} & & & -1 \end{bmatrix} \in \mathbb{C}^{n_1 \times n_1},$$

$$K_2 = \frac{1}{\delta_y}\begin{bmatrix} -I_{n_1} & I_{n_1} & & \\ & \ddots & \ddots & \\ & & -I_{n_1} & I_{n_1} \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_2} J_2 & & & -I_{n_1} \end{bmatrix} \in \mathbb{C}^{(n_1 n_2)\times(n_1 n_2)},$$

$$K_3 = \frac{1}{\delta_z}\begin{bmatrix} -I_{n_1 n_2} & I_{n_1 n_2} & & \\ & \ddots & \ddots & \\ & & -I_{n_1 n_2} & I_{n_1 n_2} \\ e^{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_3} J_3 & & & -I_{n_1 n_2} \end{bmatrix} \in \mathbb{C}^{n \times n}$$

- For SC lattice

$$J_2 = I_{n_1}, \quad J_3 = I_{n_1 n_2}$$

- For FCC lattice

$$J_2 = \begin{bmatrix} 0 & e^{-\imath 2\pi \mathbf{k}\cdot\mathbf{a}_1} I_{n_1/2} \\ I_{n_1/2} & 0 \end{bmatrix} \in \mathbb{C}^{n_1 \times n_1},$$

$$J_3 = \begin{bmatrix} 0 & e^{-\imath 2\pi \mathbf{k}\cdot\mathbf{a}_2} I_{\frac{1}{3}n_2} \otimes I_{n_1} \\ I_{\frac{2}{3}n_2} \otimes J_2 & 0 \end{bmatrix} \in \mathbb{C}^{(n_1 n_2)\times(n_1 n_2)}$$

$\nabla \times \nabla \times E(\mathbf{r}) = \omega^2 \varepsilon(\mathbf{r},\omega) E(\mathbf{r})$

- Resulting nonlinear eigenvalue problem

$$F(\omega)\mathbf{x} \equiv \left(C^* C - \omega^2 B(\omega)\right)\mathbf{x} \equiv \left(A - \omega^2 B(\omega)\right)\mathbf{x} = 0$$

with

$$B(\omega) = \varepsilon_0 B_n + \varepsilon(\omega) B_d$$

where B_n and B_d are diagonal, $B_n + B_d = I$

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} 1 - \dfrac{\omega_p^2}{\omega^2 + \imath\Gamma_p \omega}, & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} \varepsilon_\infty - \dfrac{\omega_p^2}{\omega^2 + \imath\Gamma_p \omega} + \displaystyle\sum_{j=1}^{2} \Omega_j A_j \left(\dfrac{e^{\imath\phi_j}}{\Omega_j - \omega - \imath\Gamma_j} + \dfrac{e^{-\imath\phi_j}}{\Omega_j + \omega + \imath\Gamma_j} \right), & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$

$\nabla \times \nabla \times E(\mathbf{r}) = \omega^2 \varepsilon(\mathbf{r},\omega) E(\mathbf{r})$

- Resulting nonlinear eigenvalue problem

$$F(\omega)\mathbf{x} \equiv \left(C^* C - \omega^2 B(\omega)\right)\mathbf{x} \equiv \left(A - \omega^2 B(\omega)\right)\mathbf{x} = 0$$

with

$$B(\omega) = \boxed{\varepsilon_0 B_n} + \varepsilon(\omega) B_d$$

where B_n and B_d are diagonal, $B_n + B_d = I$

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} 1 - \dfrac{\omega_p^2}{\omega^2 + \imath\Gamma_p \omega}, & \text{in material domain} \\ \boxed{\varepsilon_0,} & \text{otherwise} \end{cases}$$

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} \varepsilon_\infty - \dfrac{\omega_p^2}{\omega^2 + \imath\Gamma_p \omega} + \sum_{j=1}^{2} \Omega_j A_j \left(\dfrac{e^{\imath\phi_j}}{\Omega_j - \omega - \imath\Gamma_j} + \dfrac{e^{-\imath\phi_j}}{\Omega_j + \omega + \imath\Gamma_j} \right), & \text{in material domain} \\ \boxed{\varepsilon_0,} & \text{otherwise} \end{cases}$$

$\nabla \times \nabla \times E(\mathbf{r}) = \omega^2 \varepsilon(\mathbf{r},\omega) E(\mathbf{r})$

- Resulting nonlinear eigenvalue problem

$$F(\omega)\mathbf{x} \equiv \left(C^* C - \omega^2 B(\omega)\right)\mathbf{x} \equiv \left(A - \omega^2 B(\omega)\right)\mathbf{x} = 0$$

with

$$B(\omega) = \varepsilon_0 B_n + \varepsilon(\omega) B_d$$

where B_n and B_d are diagonal, $B_n + B_d = I$

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} 1 - \dfrac{\omega_p^2}{\omega^2 + \imath\Gamma_p \omega}, & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} \varepsilon_\infty - \dfrac{\omega_p^2}{\omega^2 + \imath\Gamma_p \omega} + \sum_{j=1}^{2} \Omega_j A_j \left(\dfrac{e^{\imath\phi_j}}{\Omega_j - \omega - \imath\Gamma_j} + \dfrac{e^{-\imath\phi_j}}{\Omega_j + \omega + \imath\Gamma_j} \right), & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$

$\nabla \times \nabla \times E(\mathbf{r}) = \omega^2 \varepsilon(\mathbf{r},\omega) E(\mathbf{r})$

- Resulting nonlinear eigenvalue problem

$$F(\omega)\mathbf{x} \equiv \left(C^* C - \omega^2 B(\omega)\right)\mathbf{x} \equiv \left(A - \omega^2 B(\omega)\right)\mathbf{x} = 0$$

with

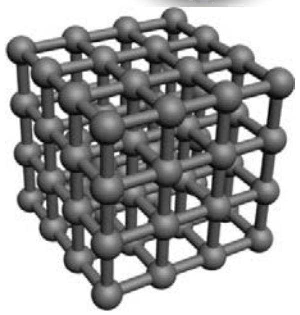
$$B(\omega) = \varepsilon_0 B_n + \boxed{\varepsilon(\omega) B_d}$$

where B_n and B_d are diagonal, $B_n + B_d = I$

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} \boxed{1 - \dfrac{\omega_p^2}{\omega^2 + \imath\Gamma_p \omega},} & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$

$$\varepsilon(\mathbf{r},\omega) = \begin{cases} \boxed{\varepsilon_\infty - \dfrac{\omega_p^2}{\omega^2 + \imath\Gamma_p \omega} + \sum_{j=1}^{2} \Omega_j A_j \left(\dfrac{e^{\imath\phi_j}}{\Omega_j - \omega - \imath\Gamma_j} + \dfrac{e^{-\imath\phi_j}}{\Omega_j + \omega + \imath\Gamma_j} \right)}, & \text{in material domain} \\ \varepsilon_0, & \text{otherwise} \end{cases}$$

Eigen-decomp. of C$_1$, C$_2$, C$_3$ for SC lattice



- Define

$$D_{\mathbf{a},m} = \mathrm{diag}\left(1, e^{\theta_{\mathbf{a},m}}, \cdots, e^{(m-1)\theta_{\mathbf{a},m}}\right),$$

$$U_m = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ e^{\theta_{m,1}} & e^{\theta_{m,2}} & \cdots & 1 \\ \vdots & \vdots & & \vdots \\ e^{(m-1)\theta_{m,1}} & e^{(m-1)\theta_{m,2}} & \cdots & 1 \end{bmatrix} \in \mathbb{C}^{m\times m}, \quad \theta_{\mathbf{a},m} = \frac{\imath 2\pi \mathbf{k}\cdot\mathbf{a}}{m}, \quad \theta_{m,i} = \frac{\imath 2\pi i}{m}$$

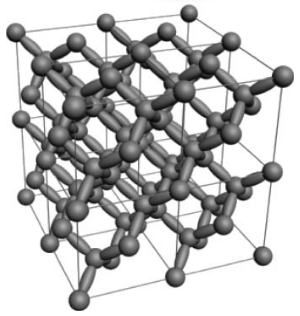
- Define unitary matrix T as

$$T = \frac{1}{\sqrt{n}}\left(D_{\mathbf{a}_3,n_3} \otimes D_{\mathbf{a}_2,n_2} \otimes D_{\mathbf{a}_1,n_1}\right)\left(U_{n_3} \otimes U_{n_2} \otimes U_{n_1}\right)$$

Then it holds that

$$C_1 T = T\Lambda_1, \quad C_2 T = T\Lambda_2, \quad C_3 T = T\Lambda_3$$

Eigen-decomp. of C_1, C_2, C_3 for FCC lattice



- Define

$$\psi_{\mathbf{x}} = \frac{\imath 2\pi \mathbf{k}\cdot\mathbf{a}_1}{n_1}, \qquad D_{\mathbf{x}} = \operatorname{diag}\left(1, e^{\psi_{\mathbf{x}}}, \cdots, e^{(n_1-1)\psi_{\mathbf{x}}}\right),$$

$$\psi_{\mathbf{y},i} = \frac{\imath 2\pi}{n_2}\left\{\mathbf{k}\cdot\left(\mathbf{a}_2 - \frac{\mathbf{a}_1}{2}\right) - \frac{i}{2}\right\}, \qquad D_{\mathbf{y},i} = \operatorname{diag}\left(1, e^{\psi_{\mathbf{y},i}}, \cdots, e^{(n_2-1)\psi_{\mathbf{y},i}}\right),$$

$$\psi_{\mathbf{z},i+j} = \frac{\imath 2\pi}{n_3}\left\{\mathbf{k}\cdot\left(\mathbf{a}_3 - \frac{\mathbf{a}_1+\mathbf{a}_2}{3}\right) - \frac{i+j}{3}\right\}, \quad D_{\mathbf{z},i+j} = \operatorname{diag}\left(1, e^{\psi_{\mathbf{y},i+j}}, \cdots, e^{(n_3-1)\psi_{\mathbf{y},i+j}}\right)$$

$$\mathbf{x}_i = D_{\mathbf{x}} U_{n_1}(:,i), \quad \mathbf{y}_{i,j} = D_{\mathbf{y},i} U_{n_2}(:,j)$$

- Define unitary matrix T as

$$T = \frac{1}{\sqrt{n}}\begin{bmatrix} T_1 & T_2 & \cdots & T_{n_1} \end{bmatrix} \in \mathbb{C}^{n\times n}, \quad T_i = \begin{bmatrix} T_{i,1} & T_{i,2} & \cdots & T_{i,n_2} \end{bmatrix} \in \mathbb{C}^{n\times(n_2 n_3)},$$

$$T_{i,j} = \left(D_{\mathbf{z},i+j} U_{n_3}\right) \otimes \left(\mathbf{y}_{i,j} \otimes \mathbf{x}_i\right)$$

Then it holds that

$$C_1 T = T\Lambda_1, \quad C_2 T = T\Lambda_2, \quad C_3 T = T\Lambda_3$$

Eigen-decomposition

- Eigen-decomposition of A:

$$\begin{bmatrix} Q_0 & Q \end{bmatrix}^* A \begin{bmatrix} Q_0 & Q \end{bmatrix} = \operatorname{diag}\left(0, \Lambda_q, \Lambda_q\right) \equiv \operatorname{diag}\left(0, \Lambda\right)$$

where

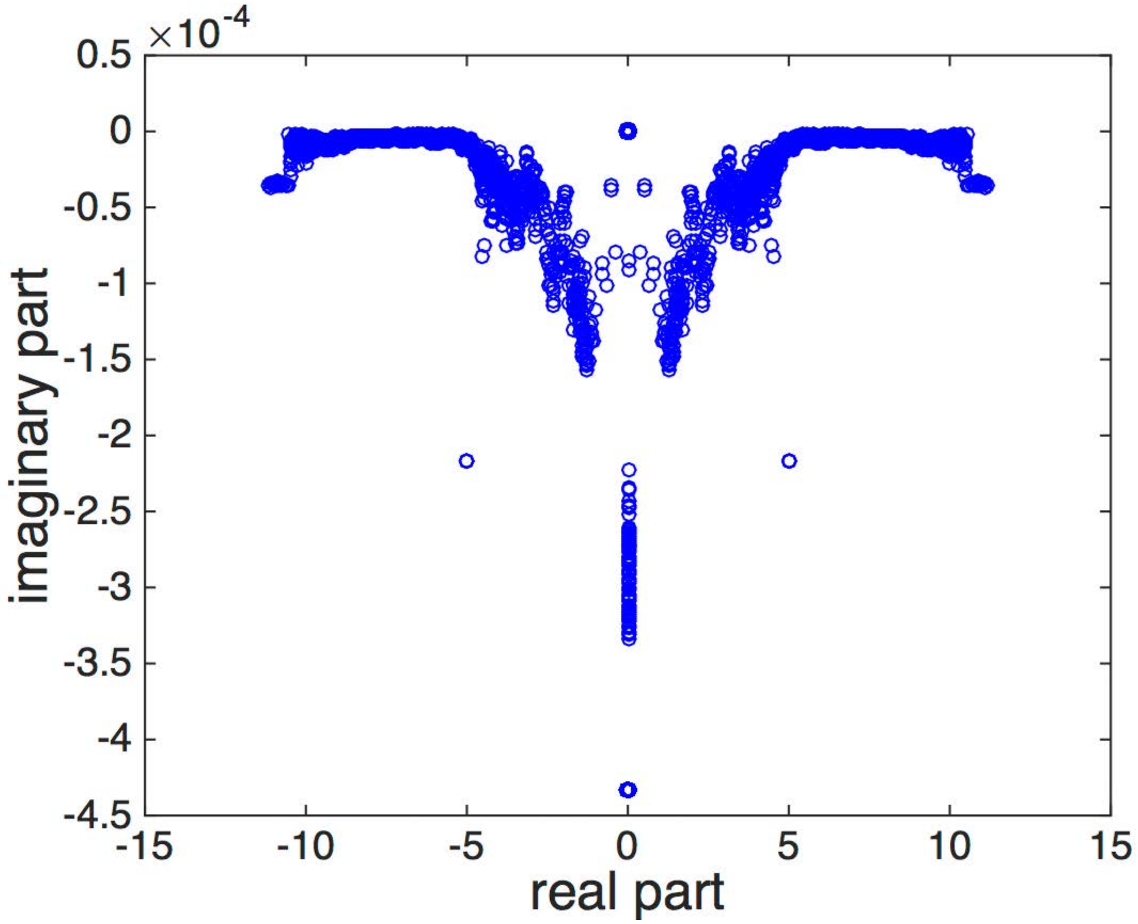
$$\begin{bmatrix} Q_0 & Q \end{bmatrix} := \left(I_3 \otimes T\right) \begin{bmatrix} \Pi_0 & \Pi_1 \end{bmatrix} \equiv \left(I_3 \otimes T\right) \begin{bmatrix} \Pi_{0,1} & \Pi_{1,1} & \Pi_{1,2} \\ \Pi_{0,2} & \Pi_{1,3} & \Pi_{1,4} \\ \Pi_{0,3} & \Pi_{1,5} & \Pi_{1,6} \end{bmatrix}$$

is unitary and $\Lambda_q = \Lambda_1^* \Lambda_1 + \Lambda_2^* \Lambda_2 + \Lambda_3^* \Lambda_3$

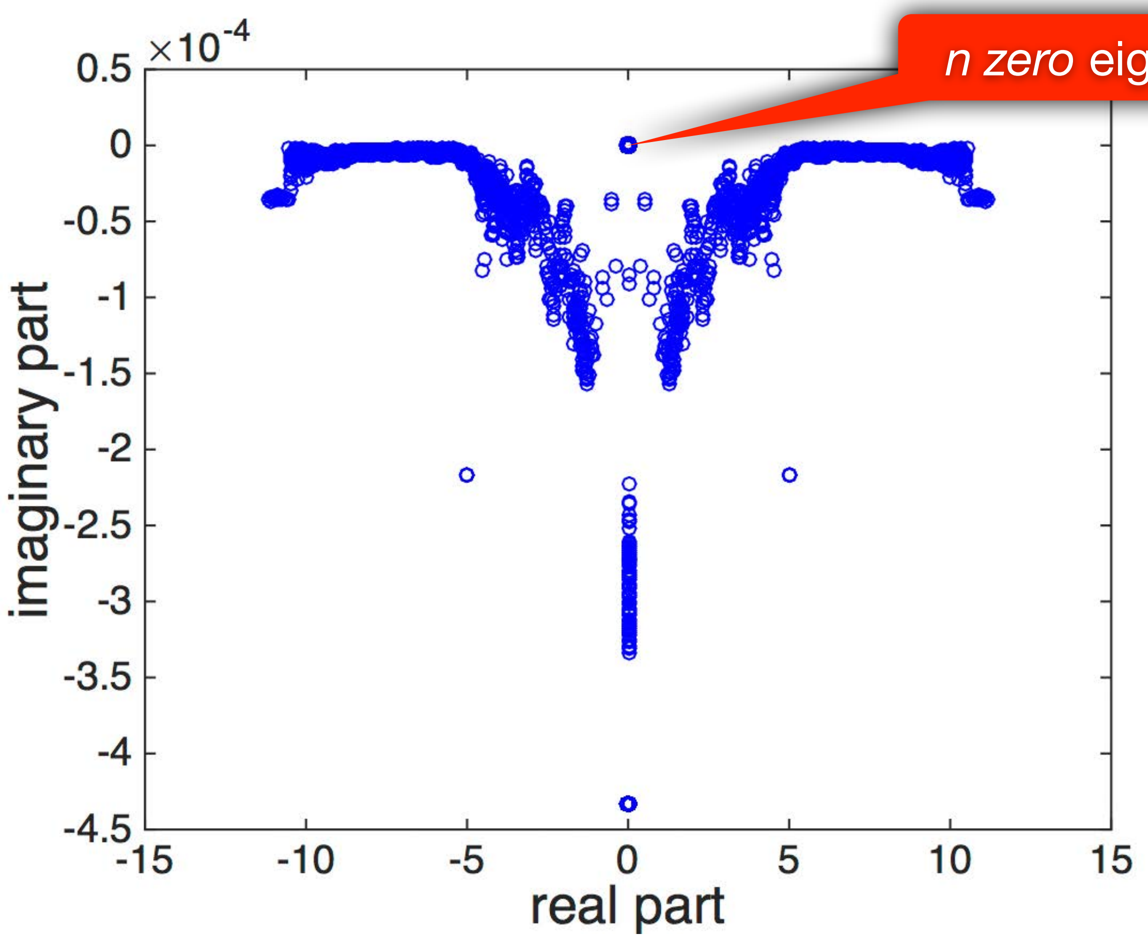
- $F(\omega)$ has n zero eigenvalues and no eigenvalue at infinity

$$F(\omega)\mathbf{x} \equiv \left(A - \omega^2 B(\omega)\right)\mathbf{x} = 0$$

Numerical Challenging

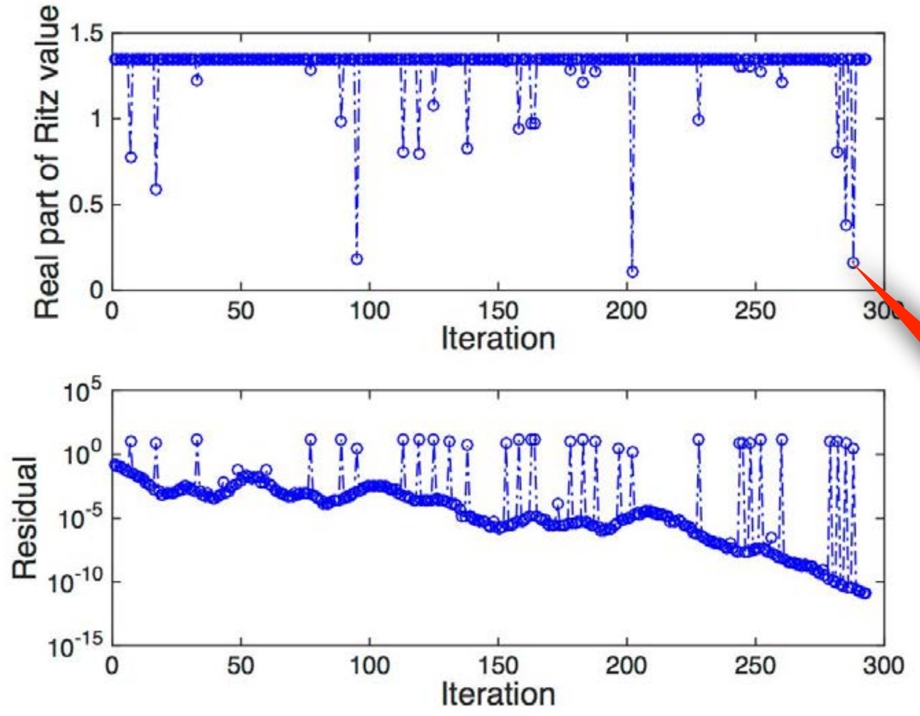


Numerical Challenging

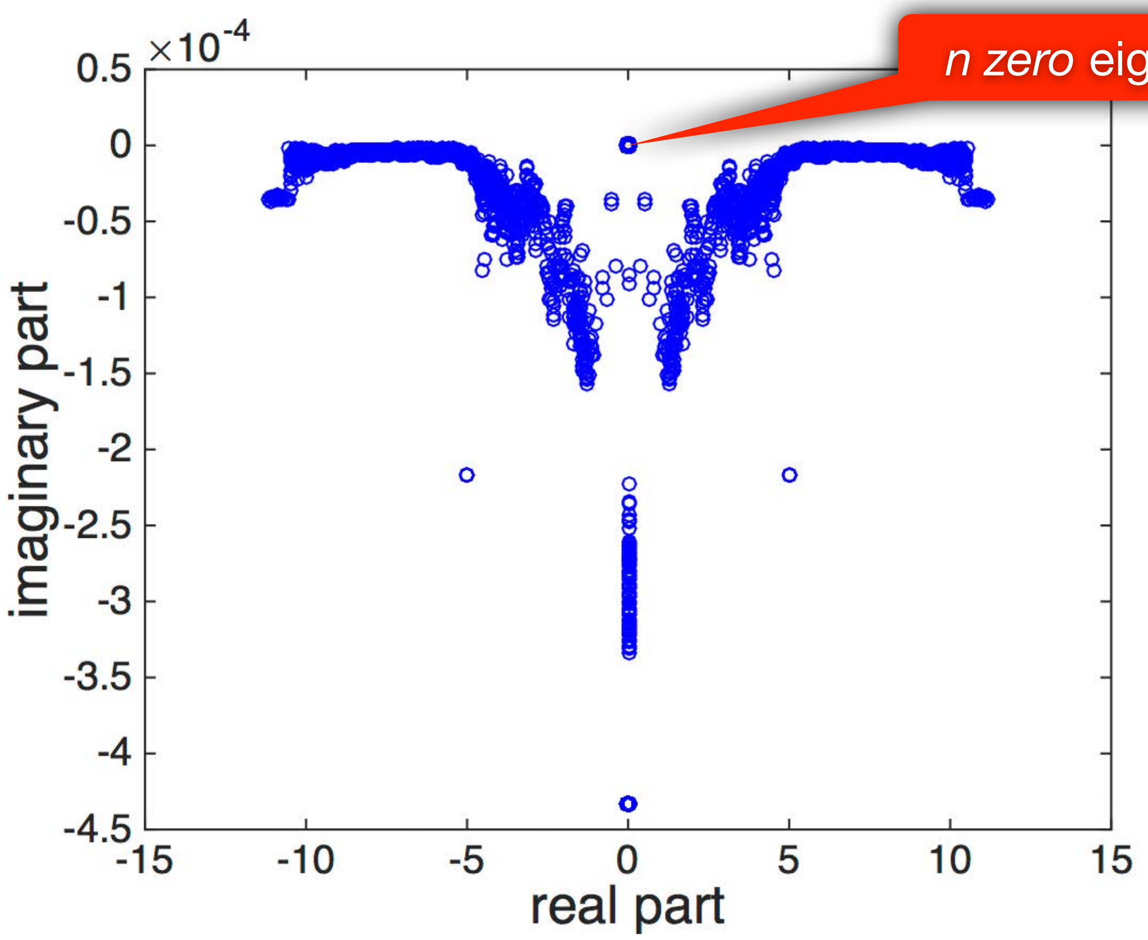


n zero eigenvalues

Numerical Challenging

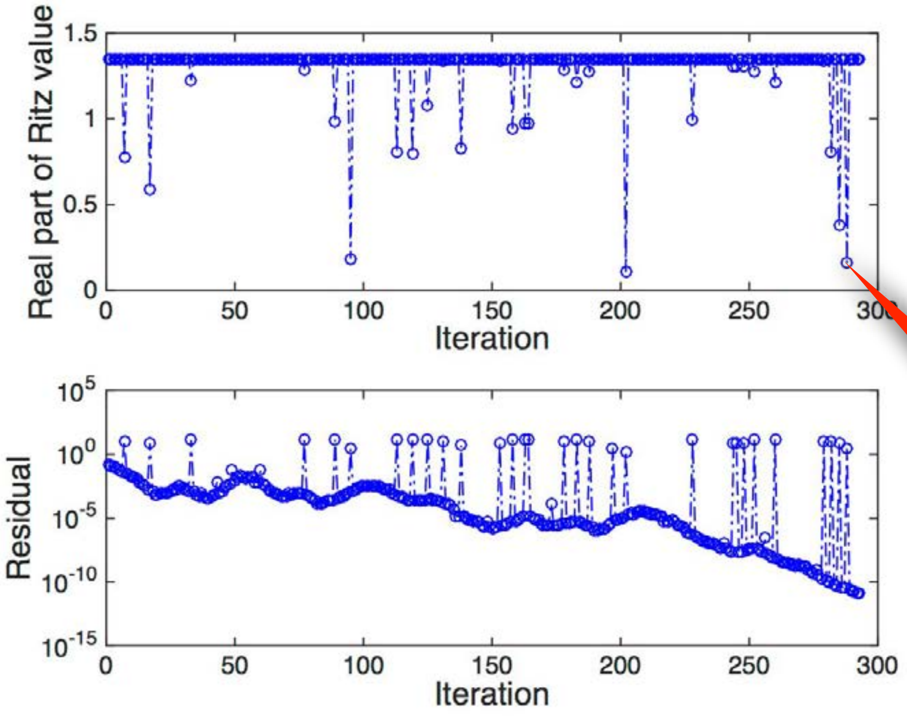


Ritz values are dragged toward zero during the iteration

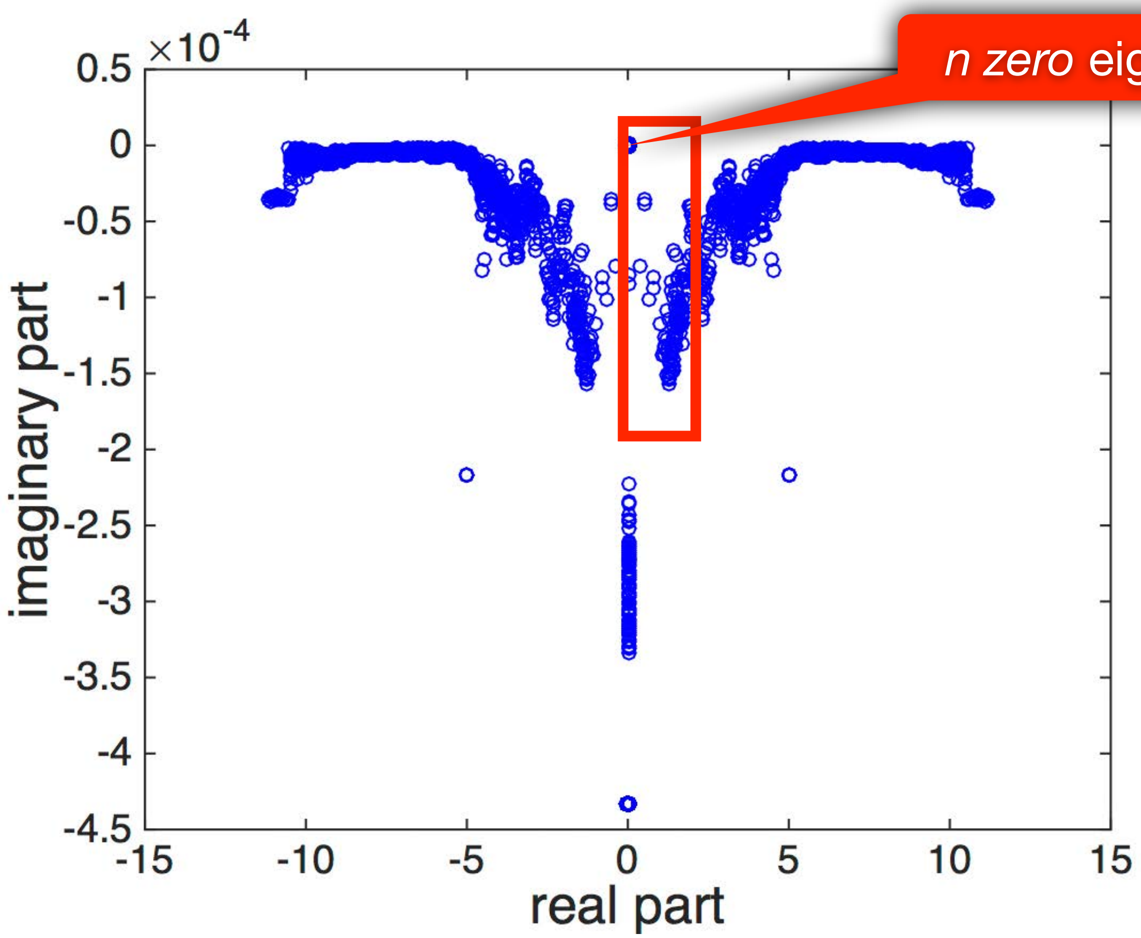


n zero eigenvalues

Numerical Challenging

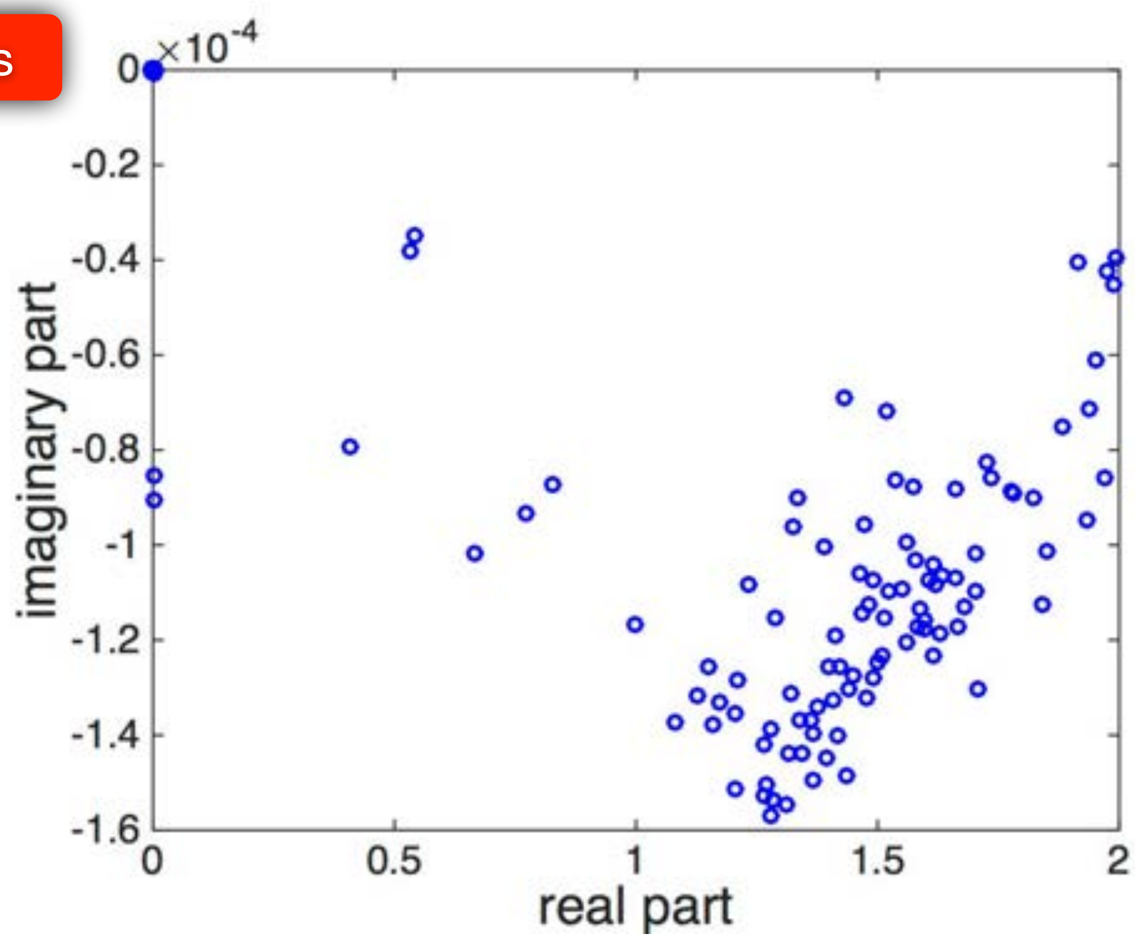


Ritz values are dragged toward zero during the iteration

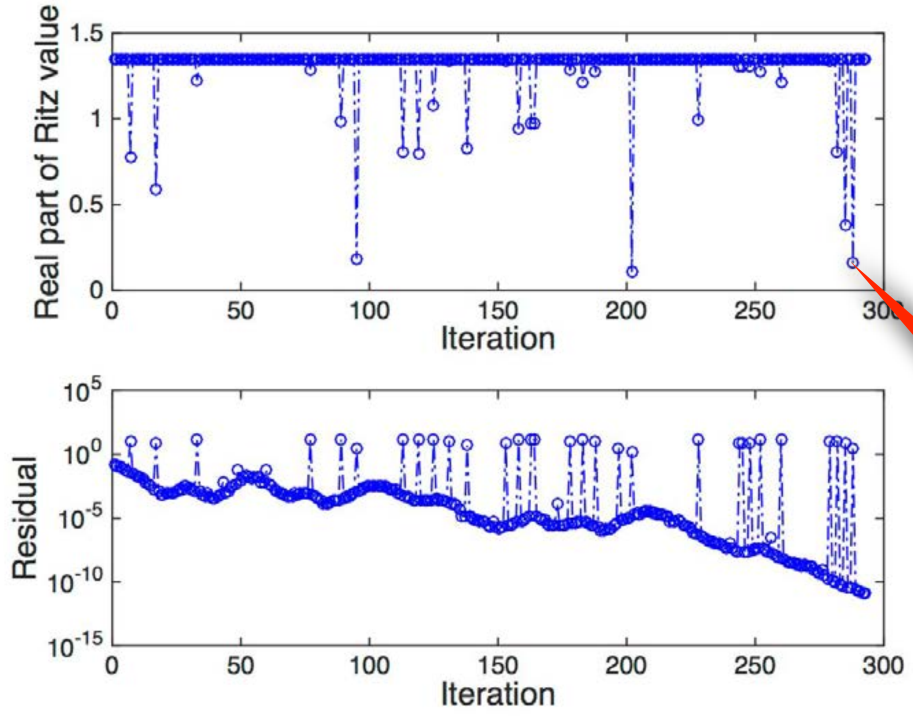


n zero eigenvalues

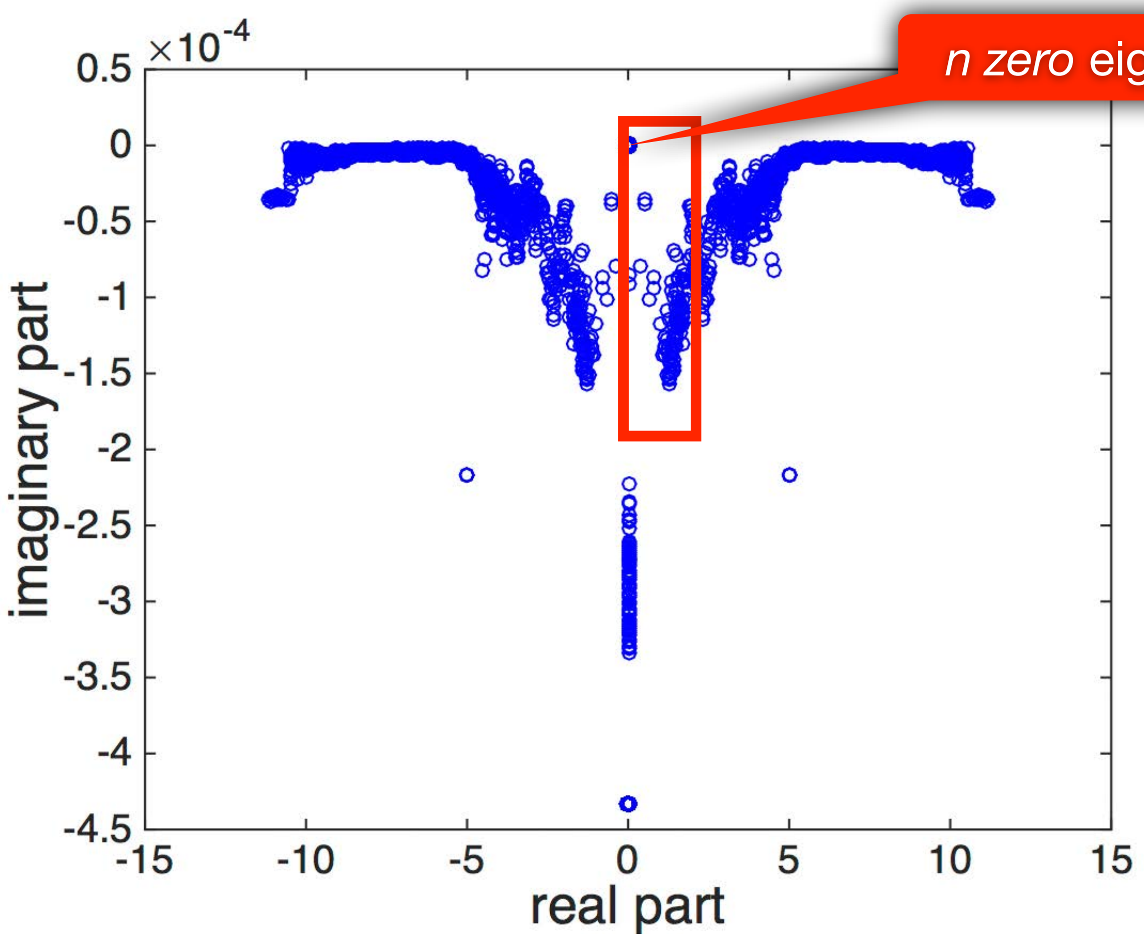
The eigenvalues with smallest positive real part are of interest



Numerical Challenging

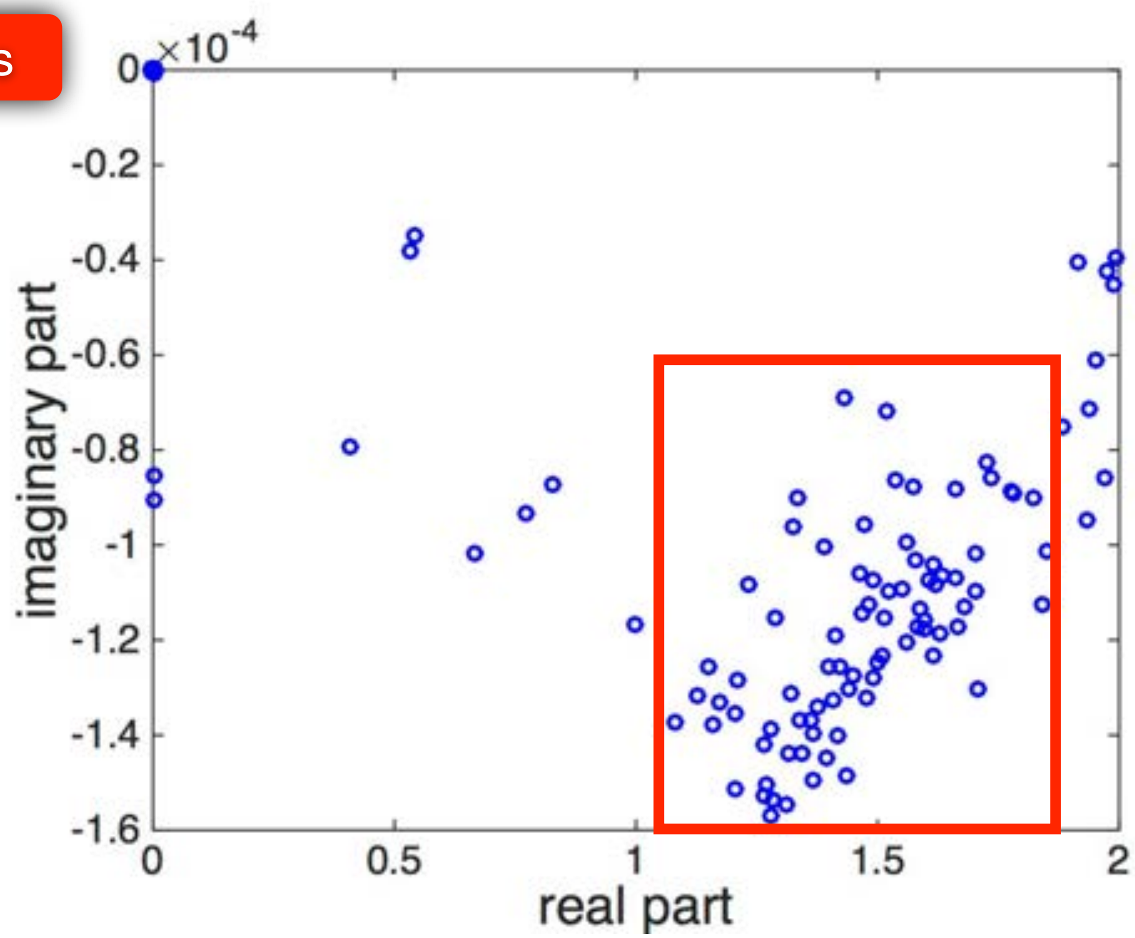


Ritz values are dragged toward zero during the iteration



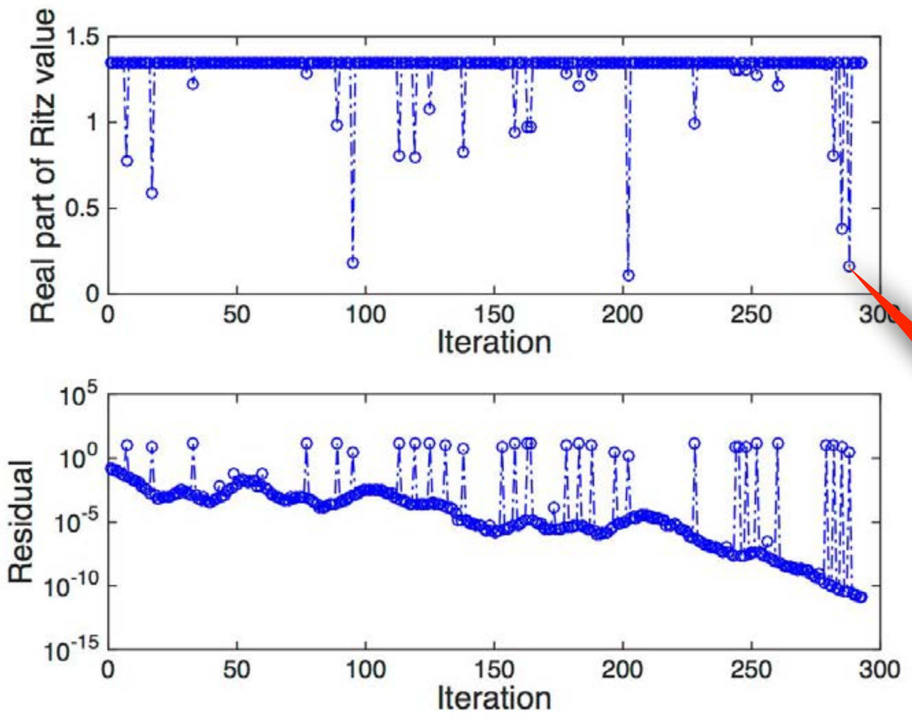
n zero eigenvalues

The eigenvalues with smallest positive real part are of interest



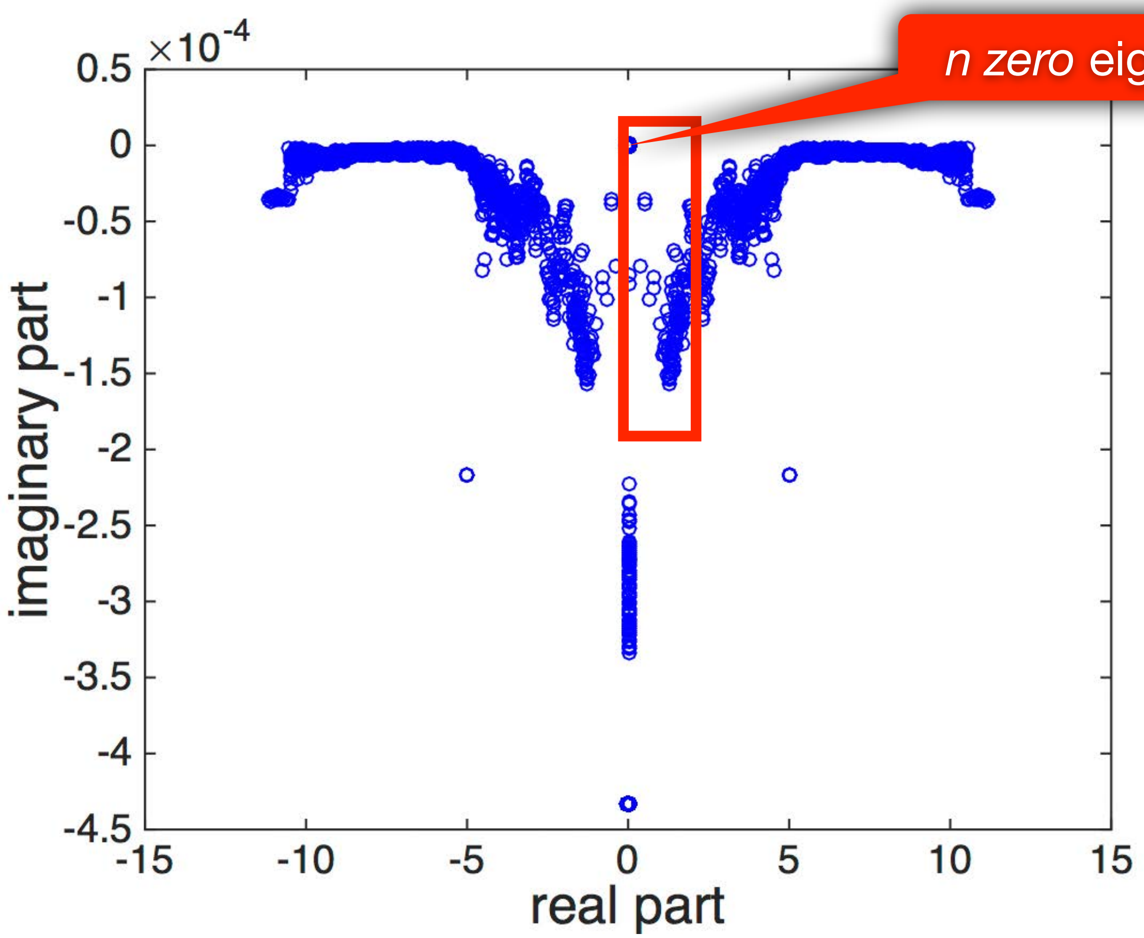
Numerical Challenging

- How to efficiently solve NLEVP
- How to tackle the effect of huge zero eigenvalues
- How to compute clustering eigenvalues

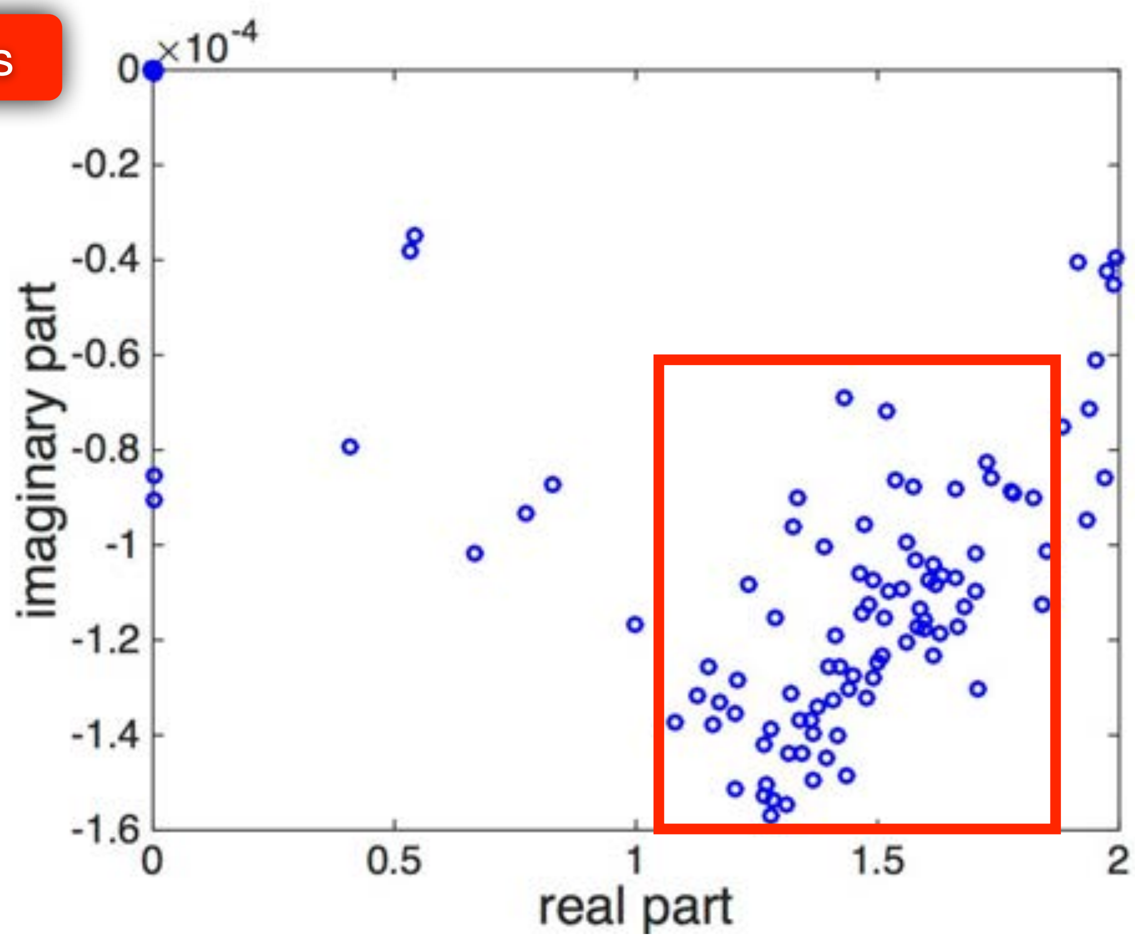


Ritz values are dragged toward zero during the iteration

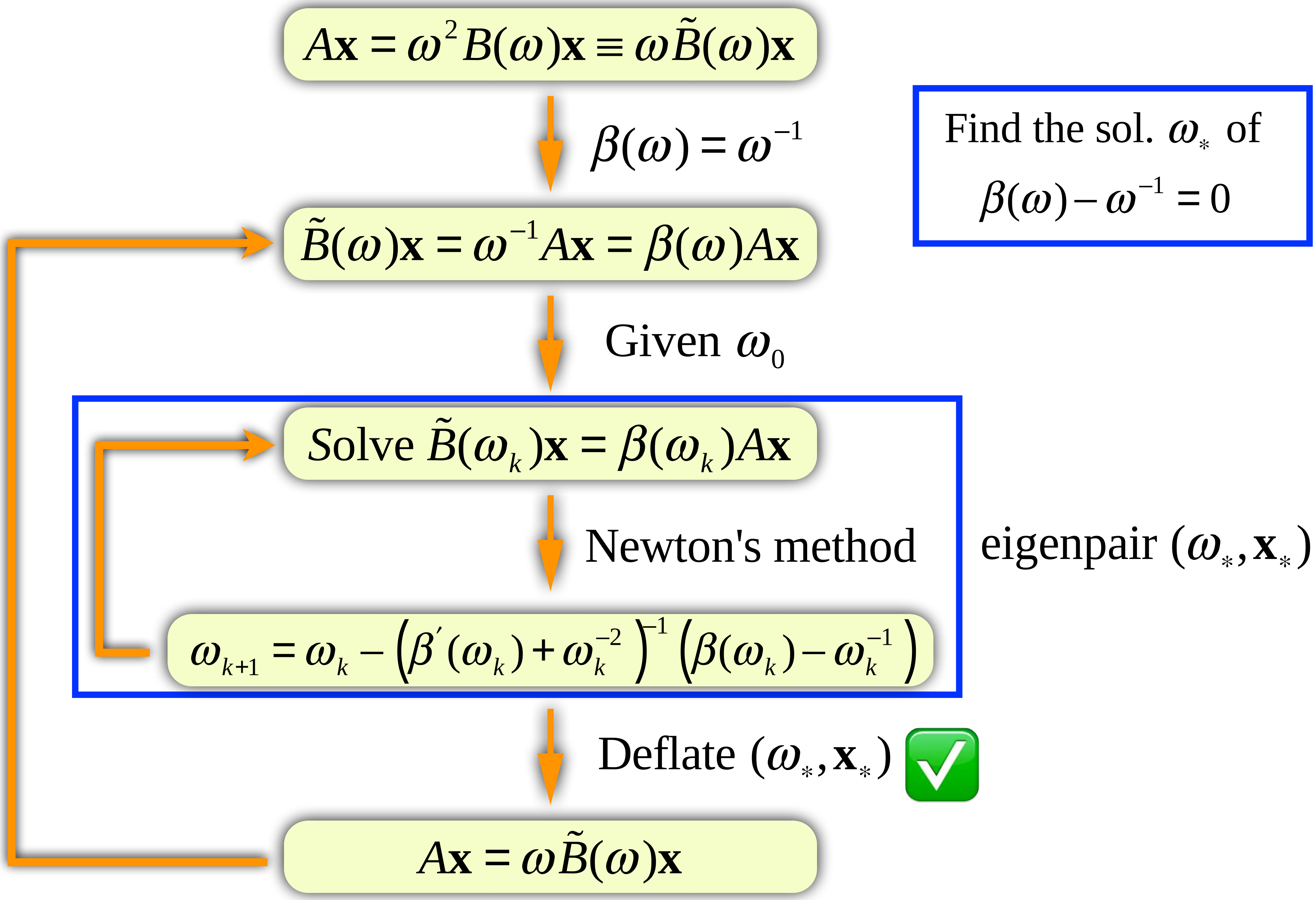
The eigenvalues with smallest positive real part are of interest



n zero eigenvalues



Flow chart of our proposed method



$$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x} \equiv \omega \tilde{B}(\omega)\mathbf{x}$$

$\beta(\omega) = \omega^{-1}$

Find the sol. ω_* of

$\beta(\omega) - \omega^{-1} = 0$

$$\tilde{B}(\omega)\mathbf{x} = \omega^{-1}A\mathbf{x} = \beta(\omega)A\mathbf{x}$$

Given ω_0

Solve $\tilde{B}(\omega_k)\mathbf{x} = \beta(\omega_k)A\mathbf{x}$

Newton's method

eigenpair $(\omega_*, \mathbf{x}_*)$

$$\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta(\omega_k) - \omega_k^{-1}\right)$$

Deflate $(\omega_*, \mathbf{x}_*)$ ✅

$$A\mathbf{x} = \omega \tilde{B}(\omega)\mathbf{x}$$

Non-equivalence deflated method for

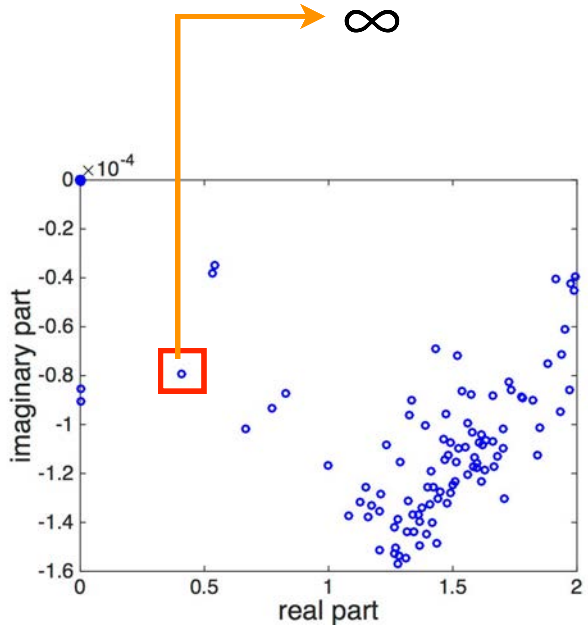
$$F(\omega)\mathbf{x} \equiv \left(A - \omega^2 B(\omega)\right)\mathbf{x} = 0$$

Deflation

$$F(\omega)\mathbf{x} \equiv \left(A - \omega^2 B(\omega)\right)\mathbf{x} = 0$$

- Let $\underbrace{\mu_1,\dots,\mu_1}_{m_1},\underbrace{\mu_2,\dots,\mu_2}_{m_2},\dots,\underbrace{\mu_\ell,\dots,\mu_\ell}_{m_\ell}$: eigenvalues of $F(\omega)$

  and $X = \begin{bmatrix} X_1 & X_2 & \cdots & X_\ell \end{bmatrix}, \quad X^* X = I_m, \quad X_j \in \mathbb{C}^{3n\times m_j}$



- Define non-equivalence deflated NLEVP as

$$\tilde{F}(\omega)\tilde{\mathbf{x}} := \left(F(\omega)\prod_{j=1}^{\ell}\left(I - \frac{\omega}{\omega-\mu_j} X_j X_j^* \right)\right)\tilde{\mathbf{x}}$$

- Theorem:

$$\left\{\omega \mid \tilde{F}(\omega)\tilde{\mathbf{x}} = 0,\ \tilde{\mathbf{x}} \neq 0\right\}$$

$$= \left\{\omega \mid F(\omega)\mathbf{x} = 0, \mathbf{x} \neq 0\right\} \setminus \left\{\mu_1,\cdots,\mu_1,\cdots,\mu_\ell,\cdots,\mu_\ell\right\} \cup \{\infty\}$$

  Furthermore, if $(\mu,\tilde{\mathbf{x}})$ is an eigenpair of $\tilde{F}(\omega)$, then $(\mu,\mathbf{x})$ is an eigenpair of $F(\omega)$ with

$$\mathbf{x} = \prod_{j=1}^{\ell}\left(I - \frac{\mu}{\mu-\mu_j} X_j X_j^* \right)\tilde{\mathbf{x}}$$

$$\tilde{F}(\omega) = \left(F(\omega) \prod_{j=1}^{\ell} \left(I - \frac{\omega}{\omega - \mu_j} X_j X_j^* \right) \right)$$

- Using the fact that $X^* X = I_m$, we obtain

$$\prod_{j=1}^{\ell} \left(I - \frac{\omega}{\omega - \mu_j} X_j X_j^* \right) = I - \sum_{j=1}^{\ell} \frac{\omega}{\omega - \mu_j} X_j X_j^* = I - \omega X D(\omega) X^*,$$

where

$$D(\omega) = \operatorname{diag}\left((\omega - \mu_1)^{-1} I_{m_1}, (\omega - \mu_2)^{-1} I_{m_2}, \cdots, (\omega - \mu_\ell)^{-1} I_{m_\ell} \right).$$

$$\tilde{F}(\omega) = \left(F(\omega) \prod_{j=1}^{\ell} \left(I - \frac{\omega}{\omega - \mu_j} X_j X_j^* \right) \right)$$

$$\boxed{F(\omega) \equiv A - \omega^2 B(\omega)}$$

- Using the fact that $X^* X = I_m$, we obtain

$$\prod_{j=1}^{\ell} \left(I - \frac{\omega}{\omega - \mu_j} X_j X_j^* \right) = I - \sum_{j=1}^{\ell} \frac{\omega}{\omega - \mu_j} X_j X_j^* = I - \omega X D(\omega) X^*,$$

where

$$D(\omega) = \mathrm{diag}\left((\omega - \mu_1)^{-1} I_{m_1}, (\omega - \mu_2)^{-1} I_{m_2}, \cdots, (\omega - \mu_\ell)^{-1} I_{m_\ell} \right).$$

- Reformulate $\tilde{F}(\omega)$ as

$$\tilde{F}(\omega) = A - \omega \left[\omega B(\omega) + (A - \omega^2 B(\omega)) X D(\omega) X^* \right]$$

$$\tilde{F}(\omega) = \left(F(\omega)\prod_{j=1}^{\ell}\left(I - \frac{\omega}{\omega-\mu_j} X_j X_j^* \right) \right)$$

$$\boxed{F(\omega) \equiv A - \omega^2 B(\omega)}$$

- Using the fact that $X^* X = I_m$, we obtain

$$\prod_{j=1}^{\ell}\left(I - \frac{\omega}{\omega-\mu_j} X_j X_j^* \right) = I - \sum_{j=1}^{\ell} \frac{\omega}{\omega-\mu_j} X_j X_j^* = I - \omega X D(\omega) X^*,$$

where

$$D(\omega) = \mathrm{diag}\left((\omega-\mu_1)^{-1} I_{m_1}, (\omega-\mu_2)^{-1} I_{m_2}, \cdots, (\omega-\mu_\ell)^{-1} I_{m_\ell} \right).$$

- Reformulate $\tilde{F}(\omega)$ as

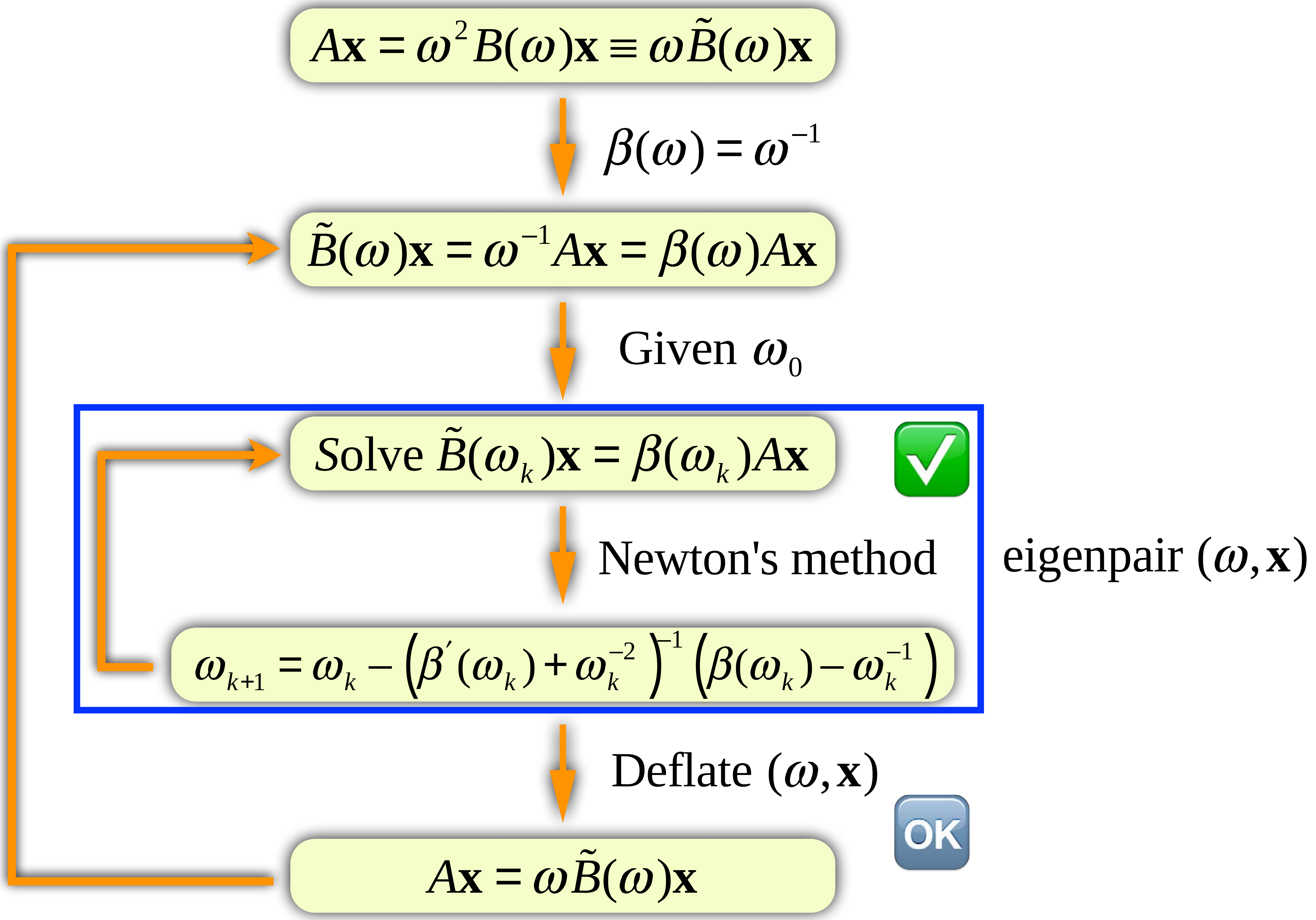
$$\tilde{F}(\omega) = A - \omega\left[\omega B(\omega) + (A - \omega^2 B(\omega)) X D(\omega) X^* \right]$$

- Define

$$\tilde{B}(\omega) = \begin{cases} \omega B(\omega) & \text{for } F(\omega), \\ \omega B(\omega) + (A - \omega^2 B(\omega)) X D(\omega) X^* & \text{for } \tilde{F}(\omega) \end{cases}$$

Then, these two NLEVP can be represented as the general form

$$\boxed{A\mathbf{x} = \omega \tilde{B}(\omega)\mathbf{x}.}$$



$$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x} \equiv \omega\tilde{B}(\omega)\mathbf{x}$$

$\beta(\omega) = \omega^{-1}$

$$\tilde{B}(\omega)\mathbf{x} = \omega^{-1}A\mathbf{x} = \beta(\omega)A\mathbf{x}$$

Given ω_0

Solve $\tilde{B}(\omega_k)\mathbf{x} = \beta(\omega_k)A\mathbf{x}$ ✓

Newton's method

$$\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta(\omega_k) - \omega_k^{-1}\right)$$

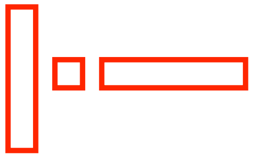
eigenpair $(\omega, \mathbf{x})$

Deflate $(\omega, \mathbf{x})$

OK

$$A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$$

$$\tilde{B}(\omega) = \omega B(\omega) + \left[(A - \omega^2 B(\omega))X\right]D(\omega)X^*$$



Null-space free method for

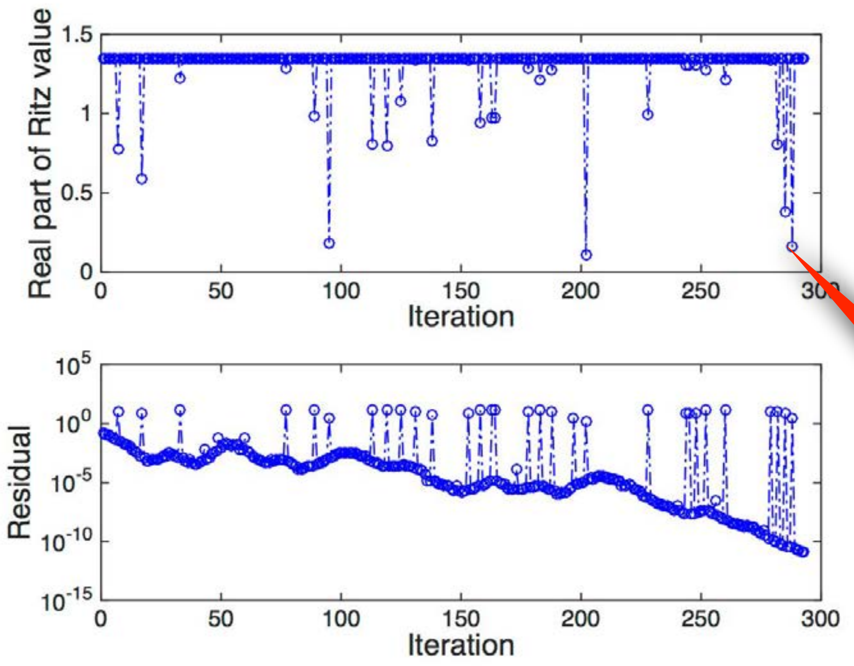
$$\boxed{\beta(\omega_k)A\mathbf{x} = \tilde{B}(\omega_k)\mathbf{x}}$$

Huge zero eigenvalues

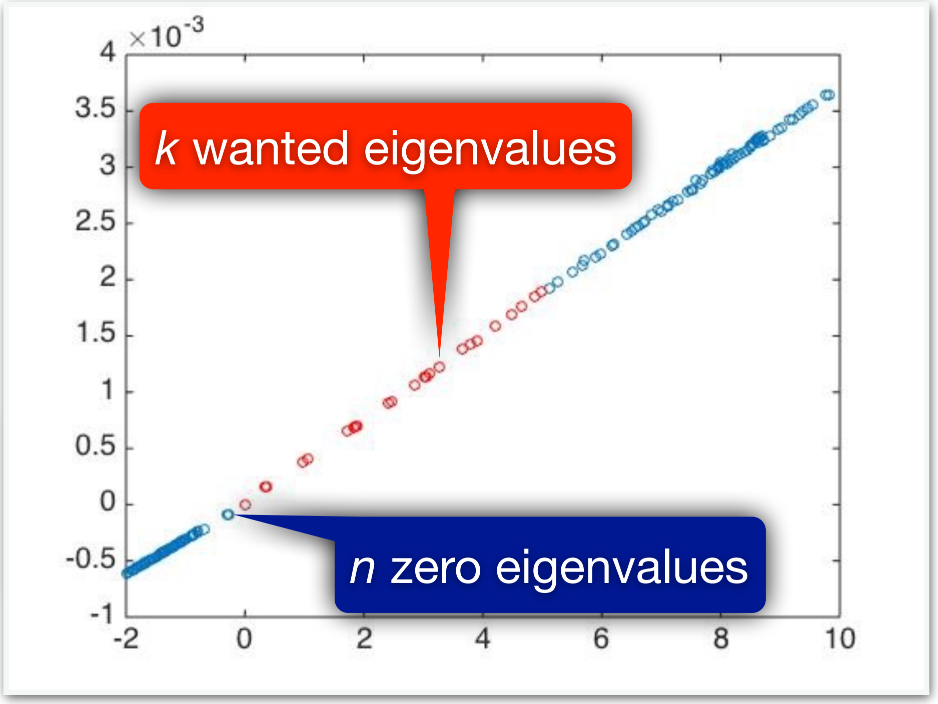
$$\begin{bmatrix} Q_0 & Q \end{bmatrix}^* A \begin{bmatrix} Q_0 & Q \end{bmatrix} = \text{diag}\left(0, \Lambda_q, \Lambda_q\right) \equiv \text{diag}\left(0, \Lambda\right)$$

$$A\mathbf{x} = \beta(\omega_k)^{-1} \tilde{B}(\omega_k)\mathbf{x} \equiv \lambda\, \tilde{B}(\omega_k)$$

n zero eigenvalues



Ritz values are dragged toward zero during the iteration



k wanted eigenvalues

n zero eigenvalues

Null-space free method

$$Q^* A Q = \Lambda$$

- Theorem:

$$\text{span}\left(\tilde{B}(\omega_k)^{-1} Q \Lambda^{1/2}\right) = \text{span}\left\{\mathbf{x} \mid A\mathbf{x} = \lambda \tilde{B}(\omega_k)\mathbf{x},\ \lambda \neq 0\right\}$$

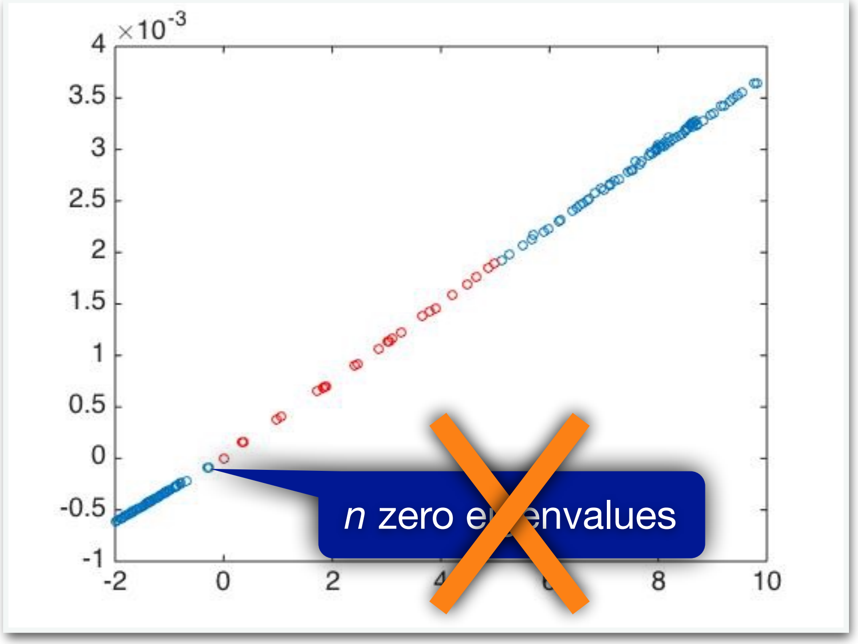
and

$$\left\{\lambda \neq 0 \middle| A\mathbf{x} = \lambda \tilde{B}(\omega_k)\mathbf{x}\right\} = \left\{\lambda \middle| \Lambda^{1/2} Q^* \tilde{B}(\omega_k)^{-1} Q \Lambda^{1/2} \mathbf{u} = \lambda \mathbf{u}\right\}$$

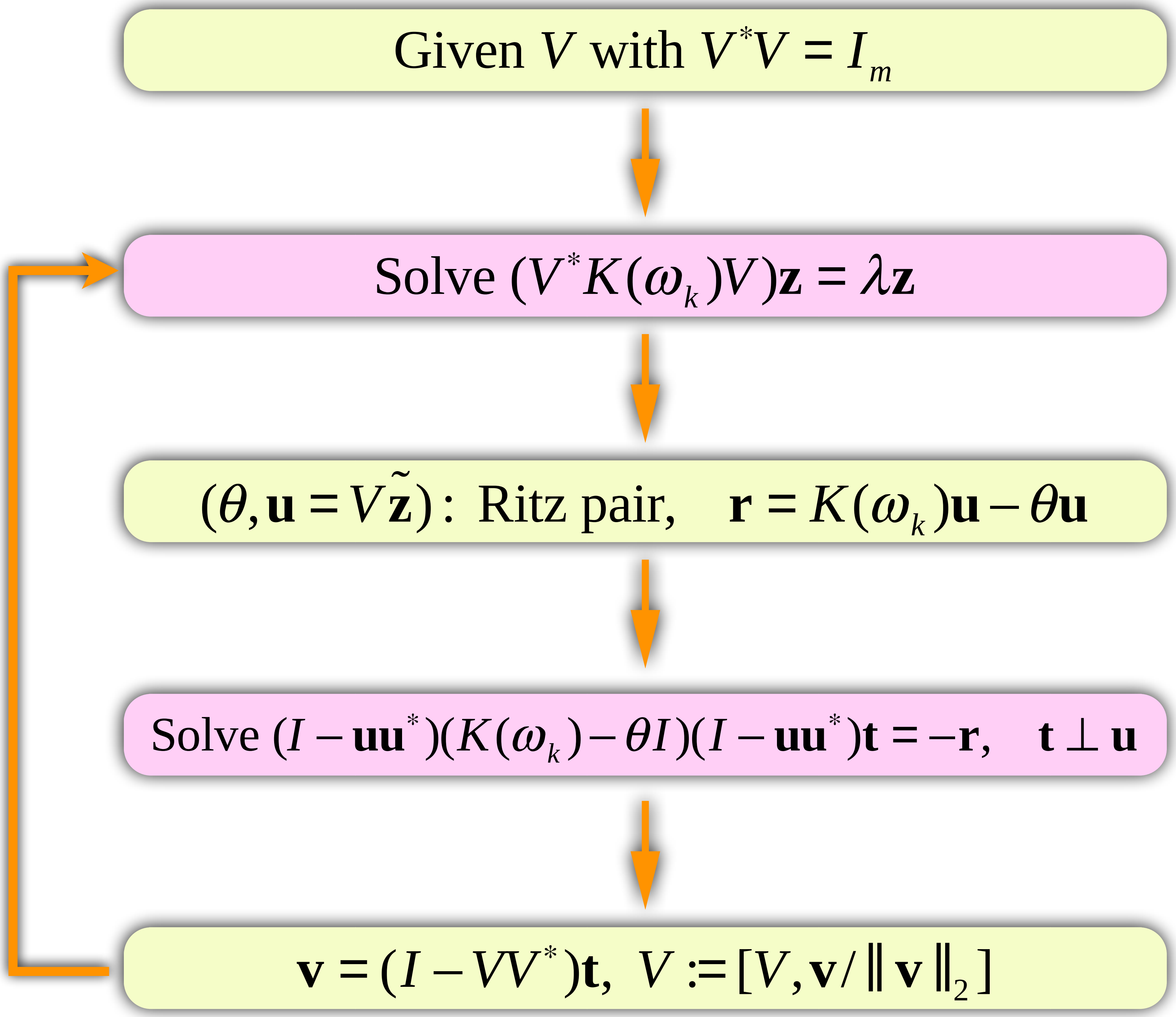
- Null-space free SEP

$$A\mathbf{x} = \lambda \tilde{B}(\omega_k)\mathbf{x} \longrightarrow K(\omega_k)\mathbf{u} \equiv \left(\Lambda^{1/2} Q^* \tilde{B}(\omega_k)^{-1} Q \Lambda^{1/2}\right)\mathbf{u} = \lambda \mathbf{u}$$

- Dim. of GEP and SEP are 3n and 2n, respectively
- GEP and SEP have same 2n nonzero eigenvalues. SEP has no zero eigenvalues



Jacobi-Davidson method for $K(\omega_k)\mathbf{u} = \lambda \mathbf{u}$



Given V with $V^* V = I_m$

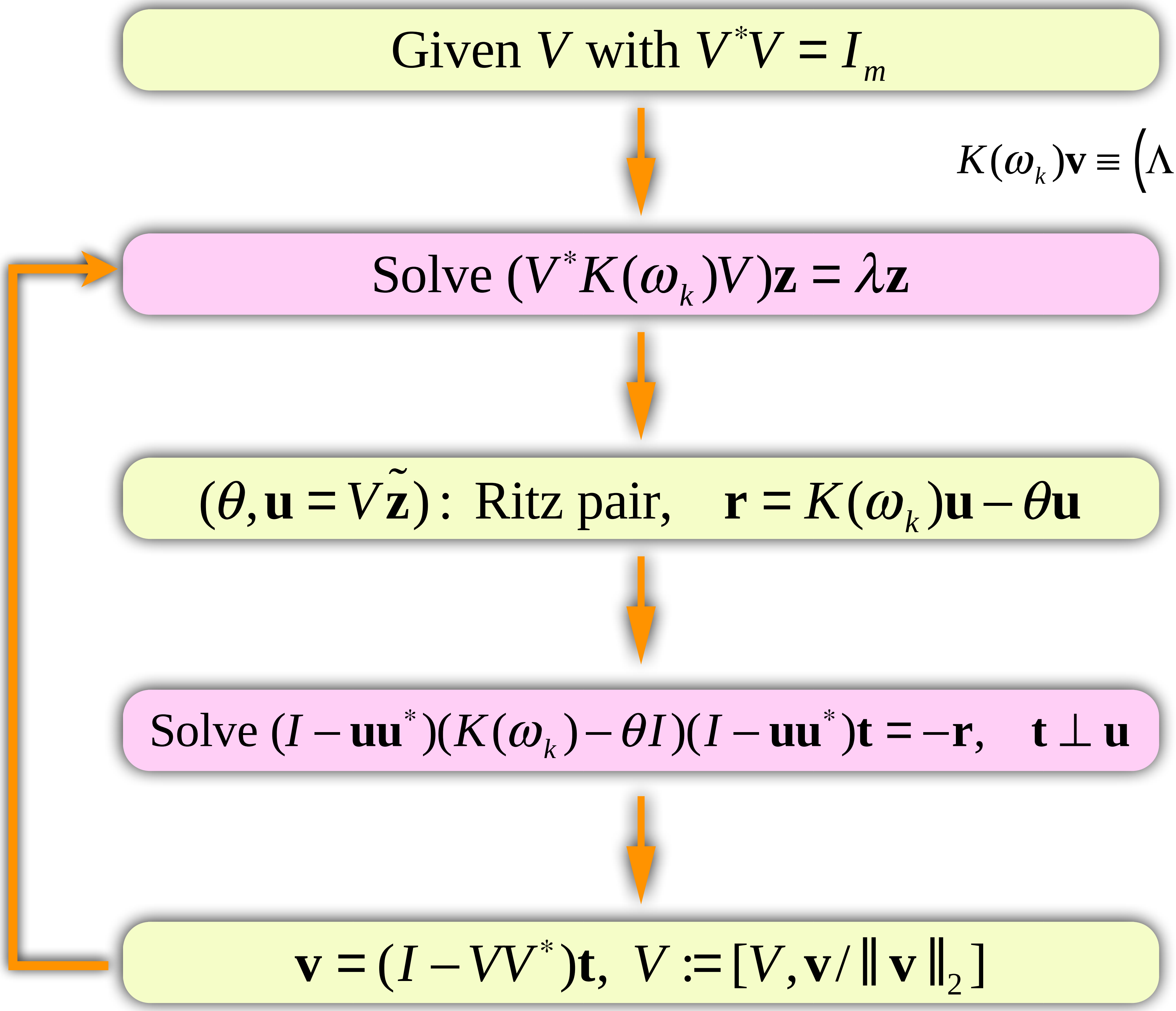
Solve $(V^* K(\omega_k) V)\mathbf{z} = \lambda \mathbf{z}$

$(\theta, \mathbf{u} = V\tilde{\mathbf{z}}):$ Ritz pair, $\quad \mathbf{r} = K(\omega_k)\mathbf{u} - \theta \mathbf{u}$

Solve $(I - \mathbf{u}\mathbf{u}^*)(K(\omega_k) - \theta I)(I - \mathbf{u}\mathbf{u}^*)\mathbf{t} = -\mathbf{r}, \quad \mathbf{t} \perp \mathbf{u}$

$\mathbf{v} = (I - VV^*)\mathbf{t},\ \ V := [V, \mathbf{v} / \| \mathbf{v} \|_2]$

Jacobi-Davidson method for $K(\omega_k)\mathbf{u} = \lambda\mathbf{u}$



Given V with $V^*V = I_m$

$K(\omega_k)\mathbf{v} \equiv \left(\Lambda^{1/2}Q^*\tilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}\right)\mathbf{v}$

Solve $(V^*K(\omega_k)V)\mathbf{z} = \lambda\mathbf{z}$ ✓

$(\theta, \mathbf{u} = V\tilde{\mathbf{z}})$: Ritz pair, $\mathbf{r} = K(\omega_k)\mathbf{u} - \theta\mathbf{u}$

Solve $(I - \mathbf{u}\mathbf{u}^*)(K(\omega_k) - \theta I)(I - \mathbf{u}\mathbf{u}^*)\mathbf{t} = -\mathbf{r}, \quad \mathbf{t} \perp \mathbf{u}$

$\mathbf{v} = (I - VV^*)\mathbf{t}, \ V := [V, \mathbf{v}/\|\mathbf{v}\|_2]$

Efficient computation $K(\omega_k)\mathbf{v} = \left(\Lambda^{1/2} Q^* \tilde{B}(\omega_k)^{-1} Q \Lambda^{1/2}\right)\mathbf{v}$

- It is required to compute $Q^*\tilde{\mathbf{p}}$, $Q\tilde{\mathbf{q}}$, and $\tilde{B}(\omega)^{-1}\mathbf{d}$ for given vectors $\tilde{\mathbf{p}}, \tilde{\mathbf{q}}, \mathbf{d}$

Efficient computation $K(\omega_k)\mathbf{v} = \left(\Lambda^{1/2} Q^* \tilde{B}(\omega_k)^{-1} Q \Lambda^{1/2}\right)\mathbf{v}$

- It is required to compute $Q^* \tilde{\mathbf{p}}$, $Q\tilde{\mathbf{q}}$, and $\tilde{B}(\omega)^{-1}\mathbf{d}$ for given vectors $\tilde{\mathbf{p}}, \tilde{\mathbf{q}}, \mathbf{d}$

$$Q \equiv \left(I_3 \otimes T\right)\begin{bmatrix} \ddots & \ddots \\ \ddots & \ddots \\ \ddots & \ddots \end{bmatrix}$$

- For computing $Q^* \tilde{\mathbf{p}}$ and $Q\tilde{\mathbf{q}}$, the matrix Q itself does not need to be formed explicitly because the matrix-vector products $T^* \mathbf{p}$ and $T\mathbf{q}$ can be evaluated by the fast Fourier transform efficiently

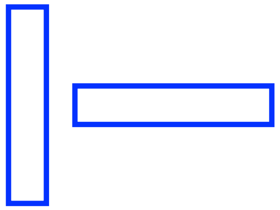
Efficient computation $K(\omega_k)\mathbf{v} = \left(\Lambda^{1/2}Q^*\tilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}\right)\mathbf{v}$

- It is required to compute $Q^*\tilde{\mathbf{p}}$, $Q\tilde{\mathbf{q}}$, and $\tilde{B}(\omega)^{-1}\mathbf{d}$ for given vectors $\tilde{\mathbf{p}}, \tilde{\mathbf{q}}, \mathbf{d}$

$$Q \equiv \left(I_3 \otimes T\right)\begin{bmatrix} \ddots & \ddots \\ \ddots & \ddots \\ \ddots & \ddots \end{bmatrix}$$

- For computing $Q^*\tilde{\mathbf{p}}$ and $Q\tilde{\mathbf{q}}$, the matrix Q itself does not need to be formed explicitly because the matrix-vector products $T^*\mathbf{p}$ and $T\mathbf{q}$ can be evaluated by the fast Fourier transform efficiently

- For computing $\tilde{B}(\omega)^{-1}\mathbf{d}$, represent $\tilde{B}(\omega)$ as



$$\tilde{B}(\omega) = \omega B(\omega) + Y(\omega)X^*, \quad Y(\omega) = (A - \omega^2 B(\omega))XD(\omega)$$

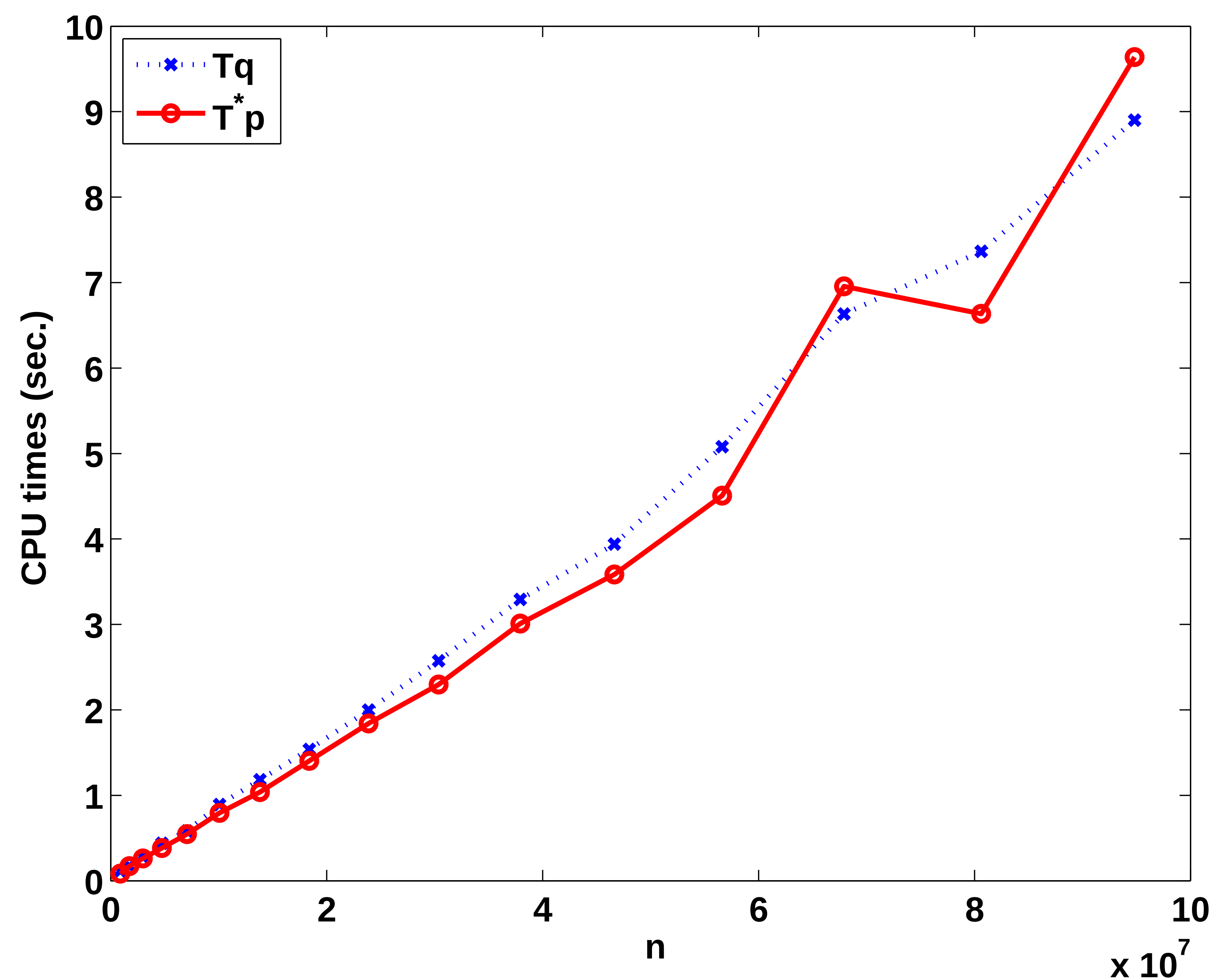
$$\boxed{\tilde{B}(\omega)^{-1} = \omega^{-1}B(\omega)^{-1}\left\{I - Y(\omega)\left(\omega I + X^*B(\omega)^{-1}Y(\omega)\right)^{-1} X^*B(\omega)^{-1}\right\}}$$

CPU Times for T*p and Tq with FCC

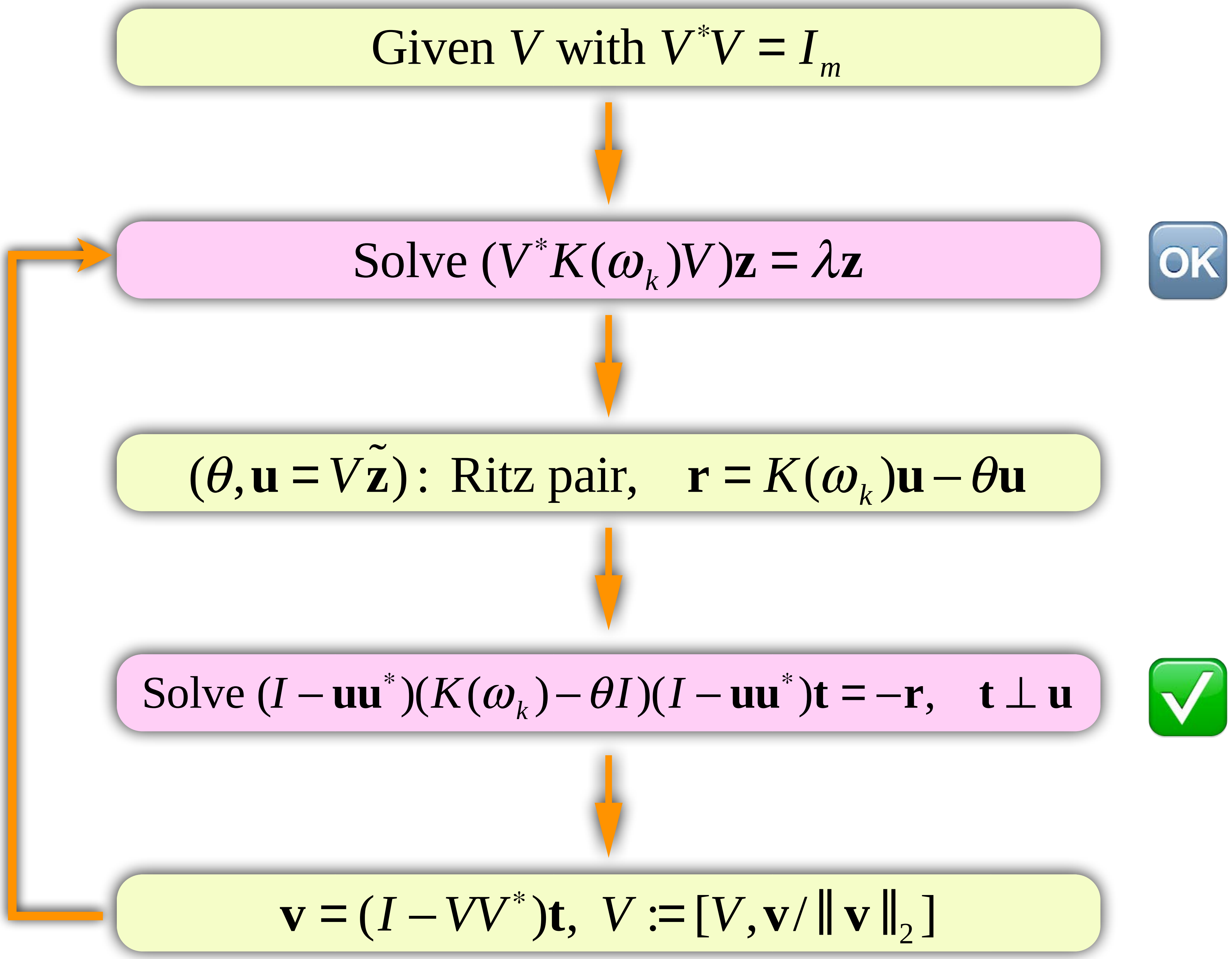
MATLAB

T*p : fft

Tq : ifft



Jacobi-Davidson method for $K(\omega_k)\mathbf{u} = \lambda \mathbf{u}$



Given V with $V^*V = I_m$

Solve $(V^*K(\omega_k)V)\mathbf{z} = \lambda \mathbf{z}$ OK

$(\theta, \mathbf{u} = V\tilde{\mathbf{z}})$: Ritz pair, $\quad \mathbf{r} = K(\omega_k)\mathbf{u} - \theta\mathbf{u}$

Solve $(I - \mathbf{u}\mathbf{u}^*)(K(\omega_k) - \theta I)(I - \mathbf{u}\mathbf{u}^*)\mathbf{t} = -\mathbf{r}, \quad \mathbf{t} \perp \mathbf{u}$ ✅

$\mathbf{v} = (I - VV^*)\mathbf{t}, \; V := [V, \mathbf{v}/\|\mathbf{v}\|_2]$

Solving correction equation

- In solving correction equation

$$\left(I-\mathbf{u}\mathbf{u}^*\right)\left(K(\omega_k)-\theta I\right)\left(I-\mathbf{u}\mathbf{u}^*\right)\mathbf{t}=-\mathbf{r}, \quad \mathbf{t}\perp\mathbf{u}$$

we need to solve a preconditioning linear system

$$\left(I-\mathbf{u}\mathbf{u}^*\right)M_K\left(I-\mathbf{u}\mathbf{u}^*\right)\mathbf{z}=\mathbf{d}, \quad \mathbf{z}\perp\mathbf{u}$$

→ $$\mathbf{z}=M_K^{-1}\mathbf{d}+\eta M_K^{-1}\mathbf{u} \;\text{ with }\; \eta=-\frac{\mathbf{u}^*M_K^{-1}\mathbf{d}}{\mathbf{u}^*M_K^{-1}\mathbf{u}}$$

with M_K being the preconditioner of $K(\omega_k)-\theta I$.

Preconditioner M_K

$$K(\omega_k) - \theta I = \Lambda^{1/2} Q^* \tilde{B}(\omega_k)^{-1} Q \Lambda^{1/2} - \theta I$$

$$\tilde{B}(\omega)^{-1}$$
$$= \omega^{-1} B(\omega)^{-1} \left\{ I - Y(\omega) \left(\omega I + X^* B(\omega)^{-1} Y(\omega) \right)^{-1} X^* B(\omega)^{-1} \right\}$$

$$U(\omega_k) = \omega_k^{-1} \Lambda^{1/2} Q^* B(\omega_k)^{-1} (A - \omega_k^2 B(\omega_k)) X$$

$$V(\omega_k) = \left[\omega_k^{-1} X^* B(\omega_k)^{-1} Q \Lambda^{1/2} \right]^*$$

$$\Psi(\omega_k) = D(\omega_k)^{-1} + \omega_k^{-1} X^* B(\omega_k)^{-1} (A - \omega_k^2 B(\omega_k)) X$$

Preconditioner M_K

$$K(\omega_k) - \theta I = \Lambda^{1/2} Q^* \tilde{B}(\omega_k)^{-1} Q \Lambda^{1/2} - \theta I$$

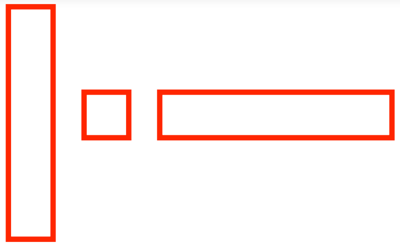
$$\tilde{B}(\omega)^{-1}$$
$$= \omega^{-1} B(\omega)^{-1} \left\{ I - Y(\omega)\left(\omega I + X^* B(\omega)^{-1} Y(\omega)\right)^{-1} X^* B(\omega)^{-1} \right\}$$

$$U(\omega_k) = \omega_k^{-1} \Lambda^{1/2} Q^* B(\omega_k)^{-1} (A - \omega_k^2 B(\omega_k)) X$$

$$V(\omega_k) = \left[\omega_k^{-1} X^* B(\omega_k)^{-1} Q \Lambda^{1/2}\right]^*$$

$$\Psi(\omega_k) = D(\omega_k)^{-1} + \omega_k^{-1} X^* B(\omega_k)^{-1} (A - \omega_k^2 B(\omega_k)) X$$

$$K(\omega_k) - \theta I = \left(\Lambda^{1/2} Q^* (\omega_k^{-1} B(\omega_k)^{-1}) Q \Lambda^{1/2} - \theta I\right) - U(\omega_k) \Psi(\omega_k)^{-1} V(\omega_k)^*$$



Preconditioner M_K

$$K(\omega_k)-\theta I = \Lambda^{1/2}Q^*\tilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}-\theta I$$

$$\tilde{B}(\omega)^{-1}$$
$$= \omega^{-1}B(\omega)^{-1}\left\{I - Y(\omega)\left(\omega I + X^*B(\omega)^{-1}Y(\omega)\right)^{-1} X^*B(\omega)^{-1}\right\}$$

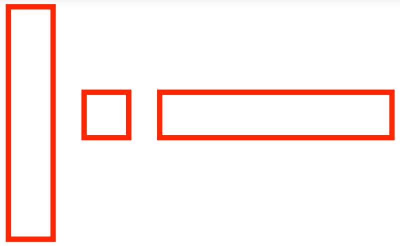
$$U(\omega_k) = \omega_k^{-1}\Lambda^{1/2}Q^*B(\omega_k)^{-1}(A-\omega_k^2B(\omega_k))X$$

$$V(\omega_k) = \left[\omega_k^{-1}X^*B(\omega_k)^{-1}Q\Lambda^{1/2}\right]^*$$

$$\Psi(\omega_k) = D(\omega_k)^{-1} + \omega_k^{-1}X^*B(\omega_k)^{-1}(A-\omega_k^2B(\omega_k))X$$

$$K(\omega_k)-\theta I = \left(\Lambda^{1/2}Q^*(\omega_k^{-1}B(\omega_k)^{-1})Q\Lambda^{1/2}-\theta I\right)-U(\omega_k)\Psi(\omega_k)^{-1}V(\omega_k)^*$$

$$\alpha_{a,k}I \approx \omega_k^{-1}B(\omega_k)^{-1}$$



Preconditioner M_K

$$K(\omega_k)-\theta I=\Lambda^{1/2}Q^*\tilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}-\theta I$$

$$\tilde{B}(\omega)^{-1}$$

$$=\omega^{-1}B(\omega)^{-1}\left\{I-Y(\omega)\left(\omega I+X^*B(\omega)^{-1}Y(\omega)\right)^{-1}X^*B(\omega)^{-1}\right\}$$

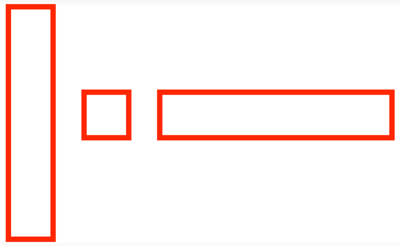
$$U(\omega_k)=\omega_k^{-1}\Lambda^{1/2}Q^*B(\omega_k)^{-1}(A-\omega_k^2B(\omega_k))X$$

$$V(\omega_k)=\left[\omega_k^{-1}X^*B(\omega_k)^{-1}Q\Lambda^{1/2}\right]^*$$

$$\Psi(\omega_k)=D(\omega_k)^{-1}+\omega_k^{-1}X^*B(\omega_k)^{-1}(A-\omega_k^2B(\omega_k))X$$

$$K(\omega_k)-\theta I=\left(\Lambda^{1/2}Q^*(\omega_k^{-1}B(\omega_k)^{-1})Q\Lambda^{1/2}-\theta I\right)-U(\omega_k)\Psi(\omega_k)^{-1}V(\omega_k)^*$$

$$\alpha_{a,k}I\approx\omega_k^{-1}B(\omega_k)^{-1}$$



$$M_K=\left(\Lambda^{1/2}Q^*(\alpha_{a,k}I)Q\Lambda^{1/2}-\theta I\right)-U(\omega_k)\Psi(\omega_k)^{-1}V(\omega_k)^*:=\Omega_k-U(\omega_k)\Psi(\omega_k)^{-1}V(\omega_k)^*$$

Preconditioner M_K

$$K(\omega_k)-\theta I = \Lambda^{1/2}Q^*\tilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}-\theta I$$

$$\tilde{B}(\omega)^{-1}$$

$$=\omega^{-1}B(\omega)^{-1}\left\{I-Y(\omega)\left(\omega I+X^*B(\omega)^{-1}Y(\omega)\right)^{-1}X^*B(\omega)^{-1}\right\}$$

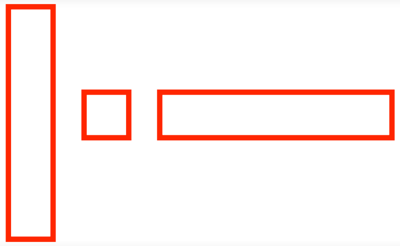
$$U(\omega_k)=\omega_k^{-1}\Lambda^{1/2}Q^*B(\omega_k)^{-1}(A-\omega_k^2B(\omega_k))X$$

$$V(\omega_k)=\left[\omega_k^{-1}X^*B(\omega_k)^{-1}Q\Lambda^{1/2}\right]^*$$

$$\Psi(\omega_k)=D(\omega_k)^{-1}+\omega_k^{-1}X^*B(\omega_k)^{-1}(A-\omega_k^2B(\omega_k))X$$

$$K(\omega_k)-\theta I = \left(\Lambda^{1/2}Q^*(\omega_k^{-1}B(\omega_k)^{-1})Q\Lambda^{1/2}-\theta I\right)-U(\omega_k)\Psi(\omega_k)^{-1}V(\omega_k)^*$$

$$\alpha_{a,k}I \approx \omega_k^{-1}B(\omega_k)^{-1}$$



$$M_K = \left(\Lambda^{1/2}Q^*(\alpha_{a,k}I)Q\Lambda^{1/2}-\theta I\right)-U(\omega_k)\Psi(\omega_k)^{-1}V(\omega_k)^* := \Omega_k - U(\omega_k)\Psi(\omega_k)^{-1}V(\omega_k)^*$$

$$\mathbf{z} = M_K^{-1}\mathbf{d}+\eta M_K^{-1}\mathbf{u}$$

$$M_K^{-1}=\Omega_k^{-1}\left\{I+U(\omega_k)\left(\Psi(\omega_k)-V(\omega_k)^*\Omega_k^{-1}U(\omega_k)\right)^{-1}V(\omega_k)^*\Omega_k^{-1}\right\}$$

Efficiency of preconditioner M_K

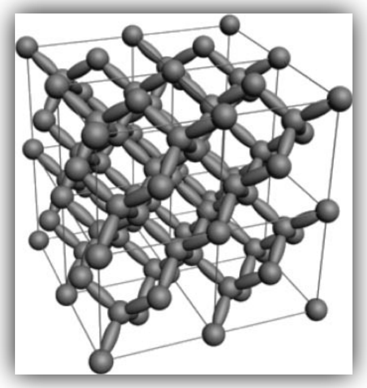
bicgstabl

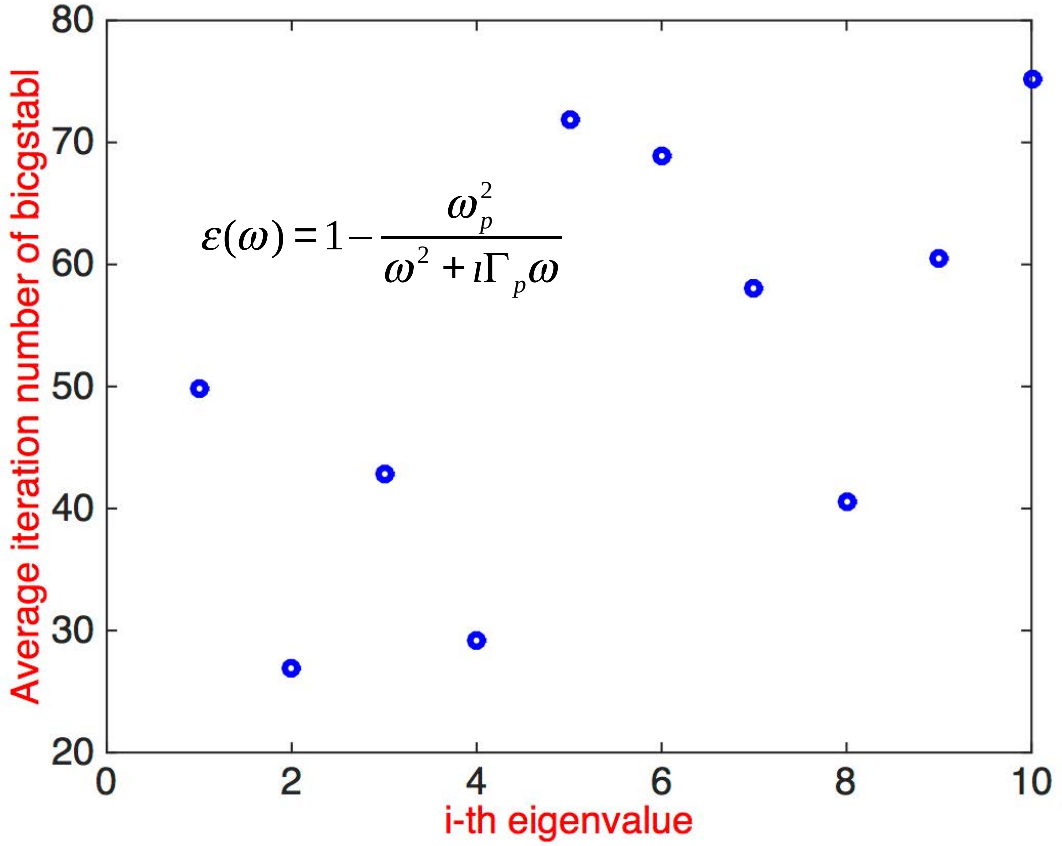
$$M_K = \Omega_k - U(\omega_k)\Psi(\omega_k)^{-1}V(\omega_k)^*$$

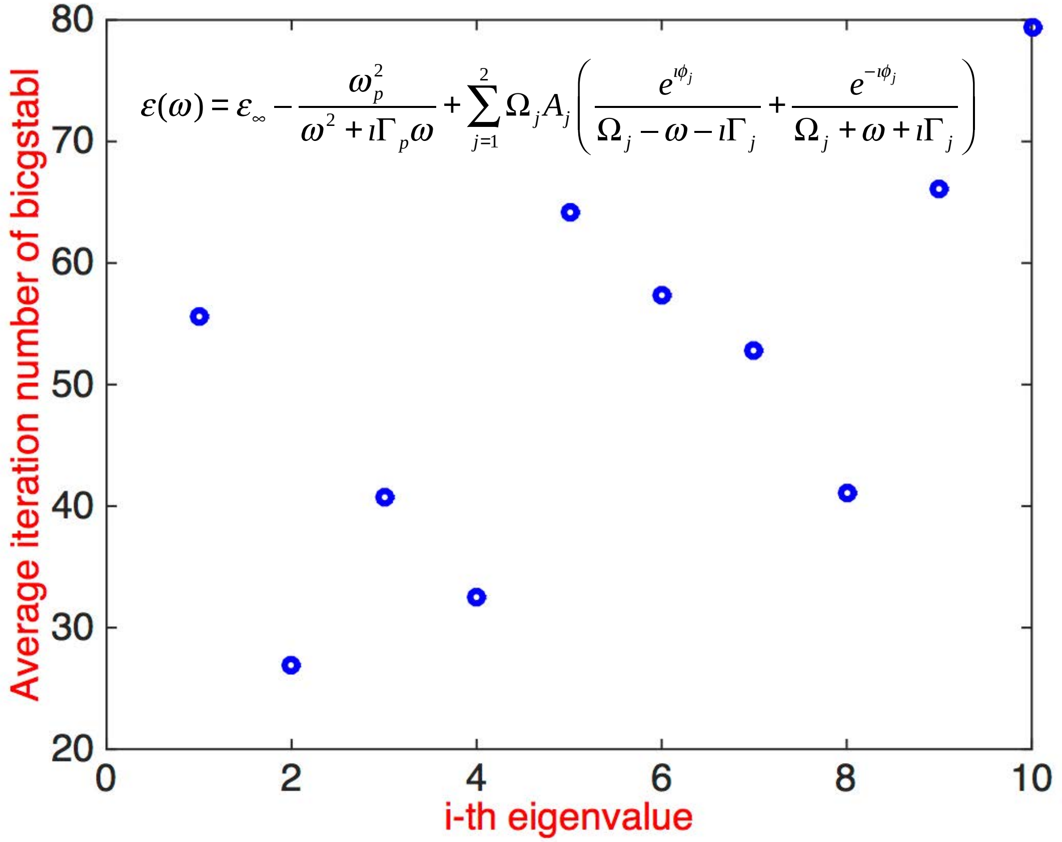
tol = 1.0e-3

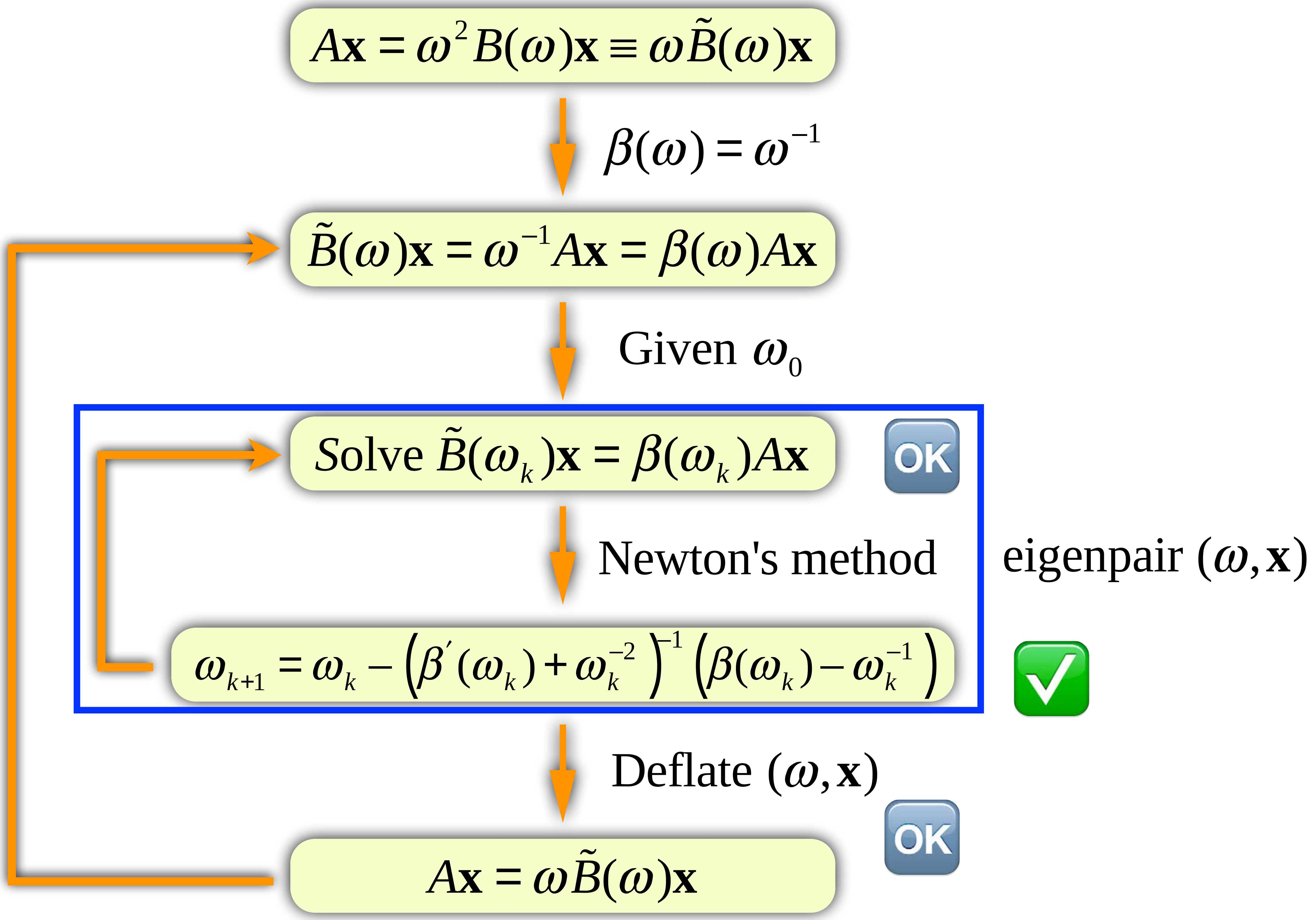
$$\left(I - \mathbf{u}\mathbf{u}^*\right)\left(K(\omega_k) - \theta I\right)\left(I - \mathbf{u}\mathbf{u}^*\right)\mathbf{t} = -\mathbf{r}, \quad \mathbf{t} \perp \mathbf{u}$$

Dimension = 1,769,472









$$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x} \equiv \omega\tilde{B}(\omega)\mathbf{x}$$

$\beta(\omega) = \omega^{-1}$

$$\tilde{B}(\omega)\mathbf{x} = \omega^{-1}A\mathbf{x} = \beta(\omega)A\mathbf{x}$$

Given ω_0

Solve $\tilde{B}(\omega_k)\mathbf{x} = \beta(\omega_k)A\mathbf{x}$ OK

Newton's method

eigenpair $(\omega, \mathbf{x})$

$$\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta(\omega_k) - \omega_k^{-1}\right)$$ ✓

Deflate $(\omega, \mathbf{x})$

OK

$$A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$$

Computing $\beta'(\omega)$

$K(\omega_k)\mathbf{u} = \beta^{-1}(\omega_k)\mathbf{u}$

- Let $\mathbf{u}(\omega)$ and $\mathbf{v}(\omega)$ with $\mathbf{v}^*(\omega)\mathbf{u}(\omega)=1$ be the right and the left eigenvectors of $K(\omega)^{-1}$, respectively, corresponding to the eigenvalue $\beta(\omega)$

$$K(\omega)^{-1}\mathbf{u}(\omega) = \beta(\omega)\mathbf{u}(\omega), \quad \mathbf{v}^*(\omega)K(\omega)^{-1} = \beta(\omega)\mathbf{v}^*(\omega)$$

Computing $\beta'(\omega)$

$K(\omega_k)\mathbf{u} = \beta^{-1}(\omega_k)\mathbf{u}$

- Let $\mathbf{u}(\omega)$ and $\mathbf{v}(\omega)$ with $\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1$ be the right and the left eigenvectors of $K(\omega)^{-1}$, respectively, corresponding to the eigenvalue $\beta(\omega)$

$$K(\omega)^{-1}\mathbf{u}(\omega) = \beta(\omega)\mathbf{u}(\omega), \quad \mathbf{v}^*(\omega)K(\omega)^{-1} = \beta(\omega)\mathbf{v}^*(\omega)$$

$$\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1$$

Computing $\beta'(\omega)$

$$K(\omega_k)\mathbf{u} = \beta^{-1}(\omega_k)\mathbf{u}$$

- Let $\mathbf{u}(\omega)$ and $\mathbf{v}(\omega)$ with $\mathbf{v}^*(\omega)\mathbf{u}(\omega)=1$ be the right and the left eigenvectors of $K(\omega)^{-1}$, respectively, corresponding to the eigenvalue $\beta(\omega)$

$$K(\omega)^{-1}\mathbf{u}(\omega) = \beta(\omega)\mathbf{u}(\omega), \quad \mathbf{v}^*(\omega)K(\omega)^{-1} = \beta(\omega)\mathbf{v}^*(\omega)$$

$$\mathbf{v}^*(\omega)\mathbf{u}(\omega)=1$$

$$\boxed{\beta(\omega) = \mathbf{v}^*(\omega)K(\omega)^{-1}\mathbf{u}(\omega)}$$

Computing $\beta'(\omega)$

$$K(\omega_k)\mathbf{u} = \beta^{-1}(\omega_k)\mathbf{u}$$

- Let $\mathbf{u}(\omega)$ and $\mathbf{v}(\omega)$ with $\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1$ be the right and the left eigenvectors of $K(\omega)^{-1}$, respectively, corresponding to the eigenvalue $\beta(\omega)$

$$K(\omega)^{-1}\mathbf{u}(\omega) = \beta(\omega)\mathbf{u}(\omega), \quad \mathbf{v}^*(\omega)K(\omega)^{-1} = \beta(\omega)\mathbf{v}^*(\omega)$$

$$\Downarrow \qquad \mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1 \Rightarrow \boxed{\mathbf{v}^*(\omega)'\mathbf{u}(\omega) + \mathbf{v}^*(\omega)\mathbf{u}(\omega)' = 0}$$

$$\boxed{\beta(\omega) = \mathbf{v}^*(\omega)K(\omega)^{-1}\mathbf{u}(\omega)}$$

Computing $\beta'(\omega)$

$$K(\omega_k)\mathbf{u} = \beta^{-1}(\omega_k)\mathbf{u}$$

- Let $\mathbf{u}(\omega)$ and $\mathbf{v}(\omega)$ with $\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1$ be the right and the left eigenvectors of $K(\omega)^{-1}$, respectively, corresponding to the eigenvalue $\beta(\omega)$

$$K(\omega)^{-1}\mathbf{u}(\omega) = \beta(\omega)\mathbf{u}(\omega), \quad \mathbf{v}^*(\omega)K(\omega)^{-1} = \beta(\omega)\mathbf{v}^*(\omega)$$

$$\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1$$

$$\boxed{\mathbf{v}^*(\omega)'\mathbf{u}(\omega) + \mathbf{v}^*(\omega)\mathbf{u}(\omega)' = 0}$$

$$\boxed{\beta(\omega) = \mathbf{v}^*(\omega)K(\omega)^{-1}\mathbf{u}(\omega)}$$

$$K(\omega)K(\omega)^{-1} = I$$

Computing $\beta'(\omega)$

$K(\omega_k)\mathbf{u} = \beta^{-1}(\omega_k)\mathbf{u}$

- Let $\mathbf{u}(\omega)$ and $\mathbf{v}(\omega)$ with $\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1$ be the right and the left eigenvectors of $K(\omega)^{-1}$, respectively, corresponding to the eigenvalue $\beta(\omega)$

$$K(\omega)^{-1}\mathbf{u}(\omega) = \beta(\omega)\mathbf{u}(\omega), \quad \mathbf{v}^*(\omega)K(\omega)^{-1} = \beta(\omega)\mathbf{v}^*(\omega)$$

$$\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1 \rightarrow \boxed{\mathbf{v}^*(\omega)'\mathbf{u}(\omega) + \mathbf{v}^*(\omega)\mathbf{u}(\omega)' = 0}$$

$$\downarrow$$

$$\boxed{\beta(\omega) = \mathbf{v}^*(\omega)K(\omega)^{-1}\mathbf{u}(\omega)}$$

$$K(\omega)K(\omega)^{-1} = I \rightarrow \boxed{(K(\omega)^{-1})' = -K(\omega)^{-1}K(\omega)'K(\omega)^{-1}}$$

Computing $\beta'(\omega)$

$K(\omega_k)\mathbf{u} = \beta^{-1}(\omega_k)\mathbf{u}$

- Let $\mathbf{u}(\omega)$ and $\mathbf{v}(\omega)$ with $\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1$ be the right and the left eigenvectors of $K(\omega)^{-1}$, respectively, corresponding to the eigenvalue $\beta(\omega)$

$$K(\omega)^{-1}\mathbf{u}(\omega) = \beta(\omega)\mathbf{u}(\omega), \quad \mathbf{v}^*(\omega)K(\omega)^{-1} = \beta(\omega)\mathbf{v}^*(\omega)$$

$$\mathbf{v}^*(\omega)\mathbf{u}(\omega) = 1 \rightarrow \boxed{\mathbf{v}^*(\omega)'\mathbf{u}(\omega) + \mathbf{v}^*(\omega)\mathbf{u}(\omega)' = 0}$$

$$\boxed{\beta(\omega) = \mathbf{v}^*(\omega)K(\omega)^{-1}\mathbf{u}(\omega)}$$

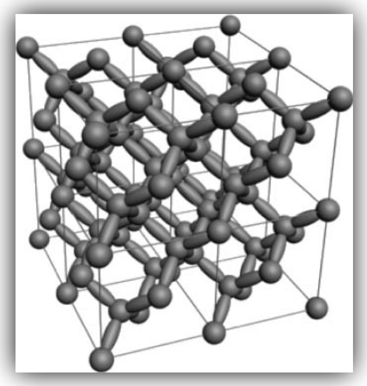
$$K(\omega)K(\omega)^{-1} = I \rightarrow \boxed{(K(\omega)^{-1})' = -K(\omega)^{-1}K(\omega)'K(\omega)^{-1}}$$

- Using these three results, we can derive

$$\boxed{\beta'(\omega) = \beta(\omega)^2\mathbf{v}^*(\omega)\Lambda^{1/2}Q^*\tilde{B}(\omega)^{-1}\tilde{B}(\omega)'\tilde{B}(\omega)^{-1}Q\Lambda^{1/2}\mathbf{u}(\omega)}$$

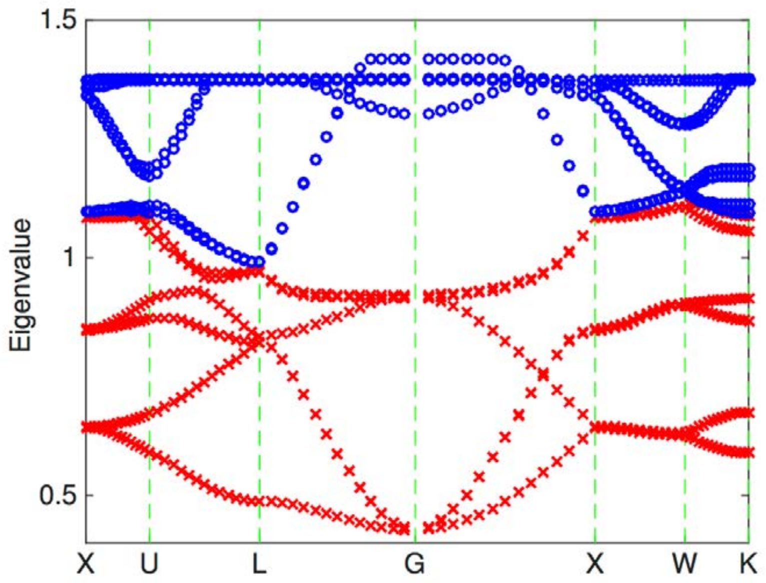
Numerical results

Benchmark problems



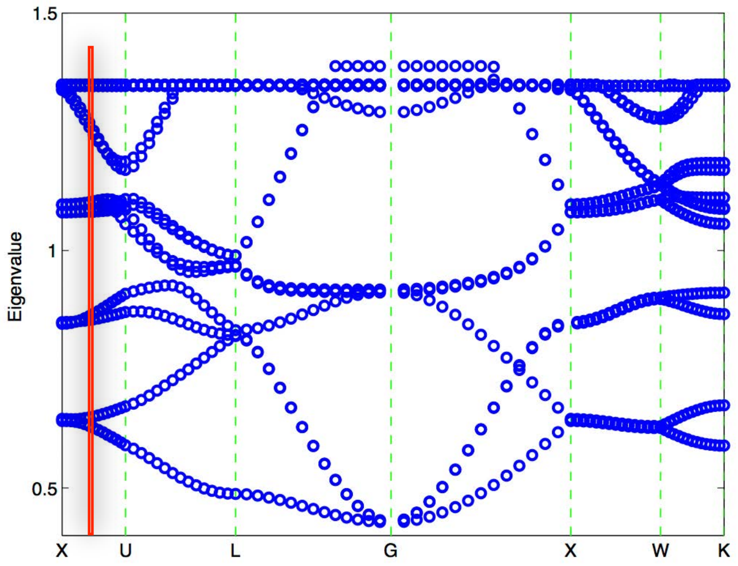
- Face-centered cubic (FCC) lattice
- Matrix dimension = 3 * 96^3 = 2,654,208
- Using MATLAB function bicgstabl with stopping tolerance 1.0e-3 to solve correction equation

$$\varepsilon(\omega)=1-\frac{\omega_p^2}{\omega^2+\imath\Gamma_p\omega}$$



Band structure diagram for Drude model

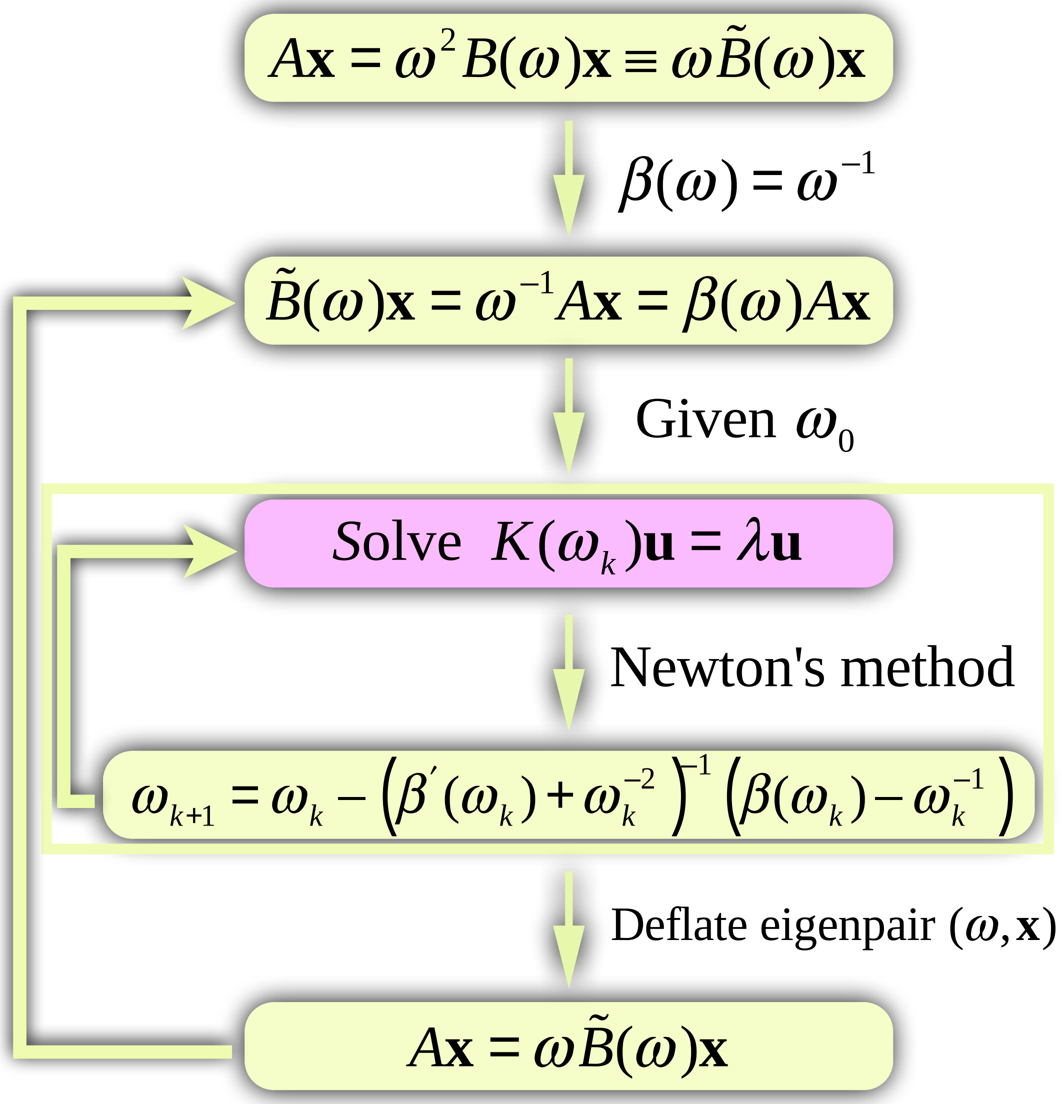
$$\varepsilon(\omega)=\varepsilon_\infty-\frac{\omega_p^2}{\omega^2+\imath\Gamma_p\omega}+\sum_{j=1}^{2}\Omega_j A_j\left(\frac{e^{\imath\phi_j}}{\Omega_j-\omega-\imath\Gamma_j}+\frac{e^{-\imath\phi_j}}{\Omega_j+\omega+\imath\Gamma_j}\right)$$



Band structure diagram for Drude-Lorentz model

Total iteration of JD

$$\sum_{k=1}^{m} \# \mathrm{JD}(K(\omega_k^{(i)}))$$



$$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x} \equiv \omega \tilde{B}(\omega)\mathbf{x}$$

$$\beta(\omega) = \omega^{-1}$$

$$\tilde{B}(\omega)\mathbf{x} = \omega^{-1} A\mathbf{x} = \beta(\omega) A\mathbf{x}$$

Given ω_0

Solve $K(\omega_k)\mathbf{u} = \lambda \mathbf{u}$

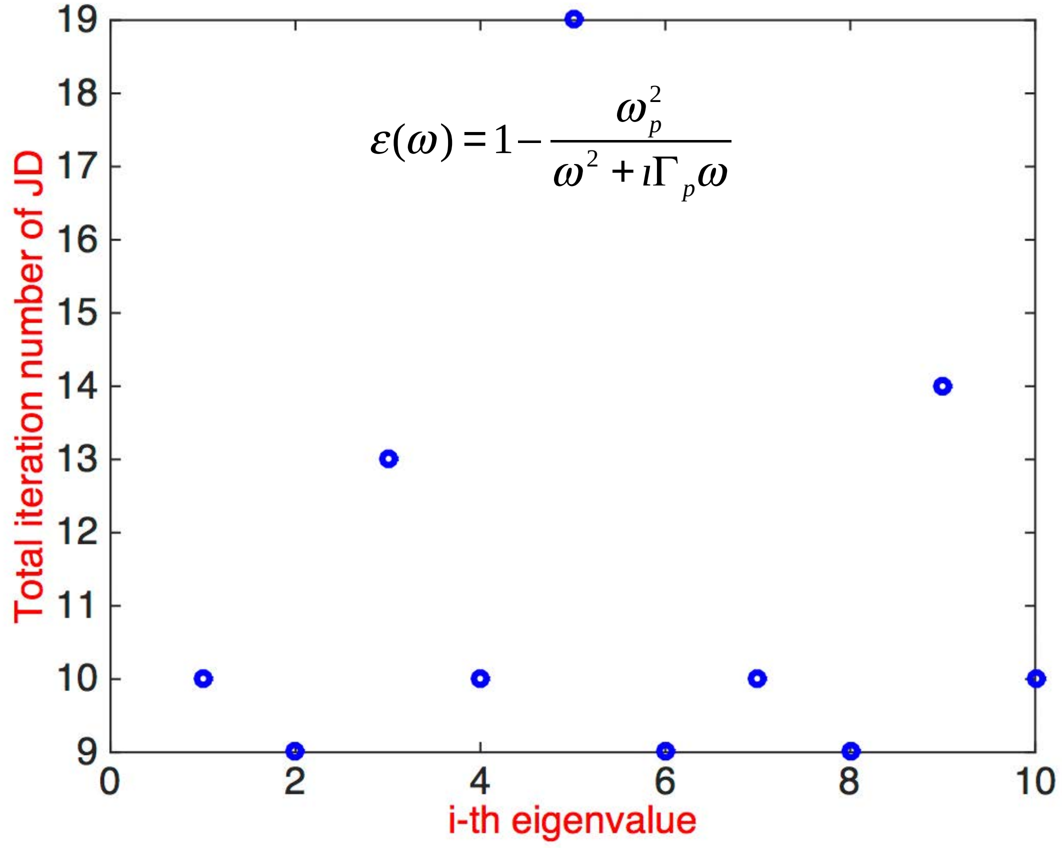
Newton's method

$$\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta(\omega_k) - \omega_k^{-1}\right)$$

Deflate eigenpair $(\omega, \mathbf{x})$

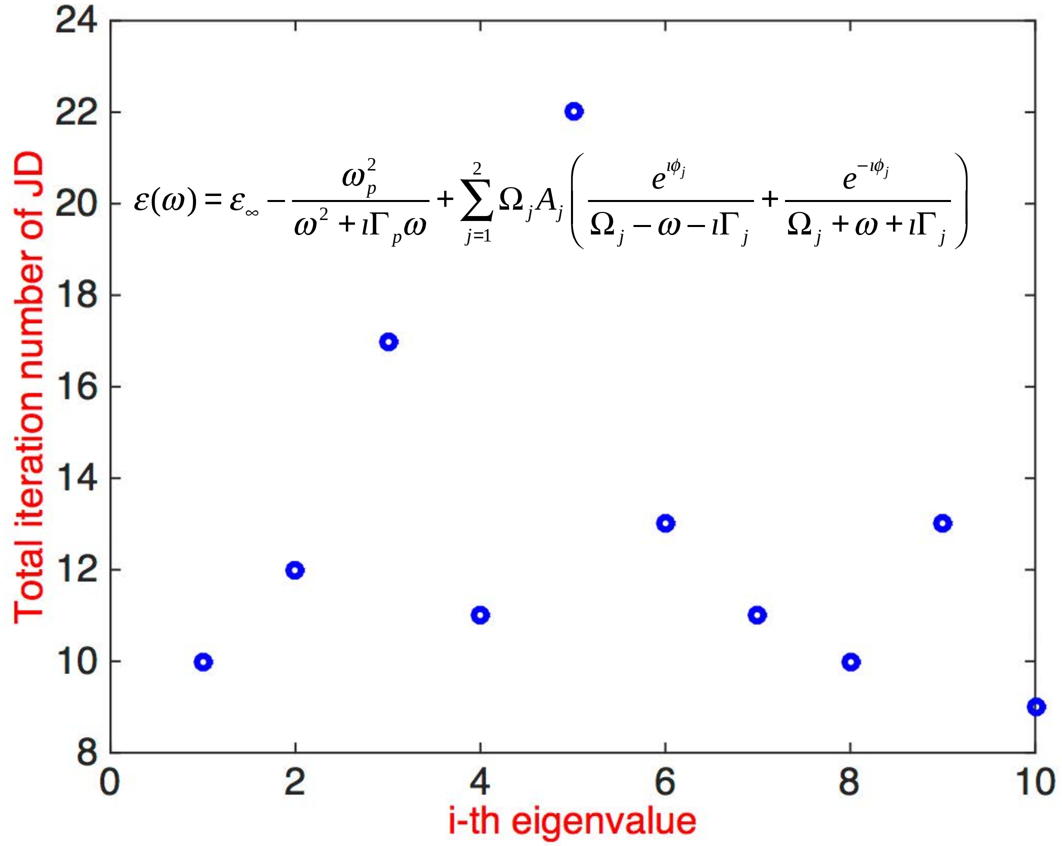
$$A\mathbf{x} = \omega \tilde{B}(\omega)\mathbf{x}$$

Dimension = 1,769,472



$$\varepsilon(\omega) = 1 - \frac{\omega_p^2}{\omega^2 + \imath \Gamma_p \omega}$$

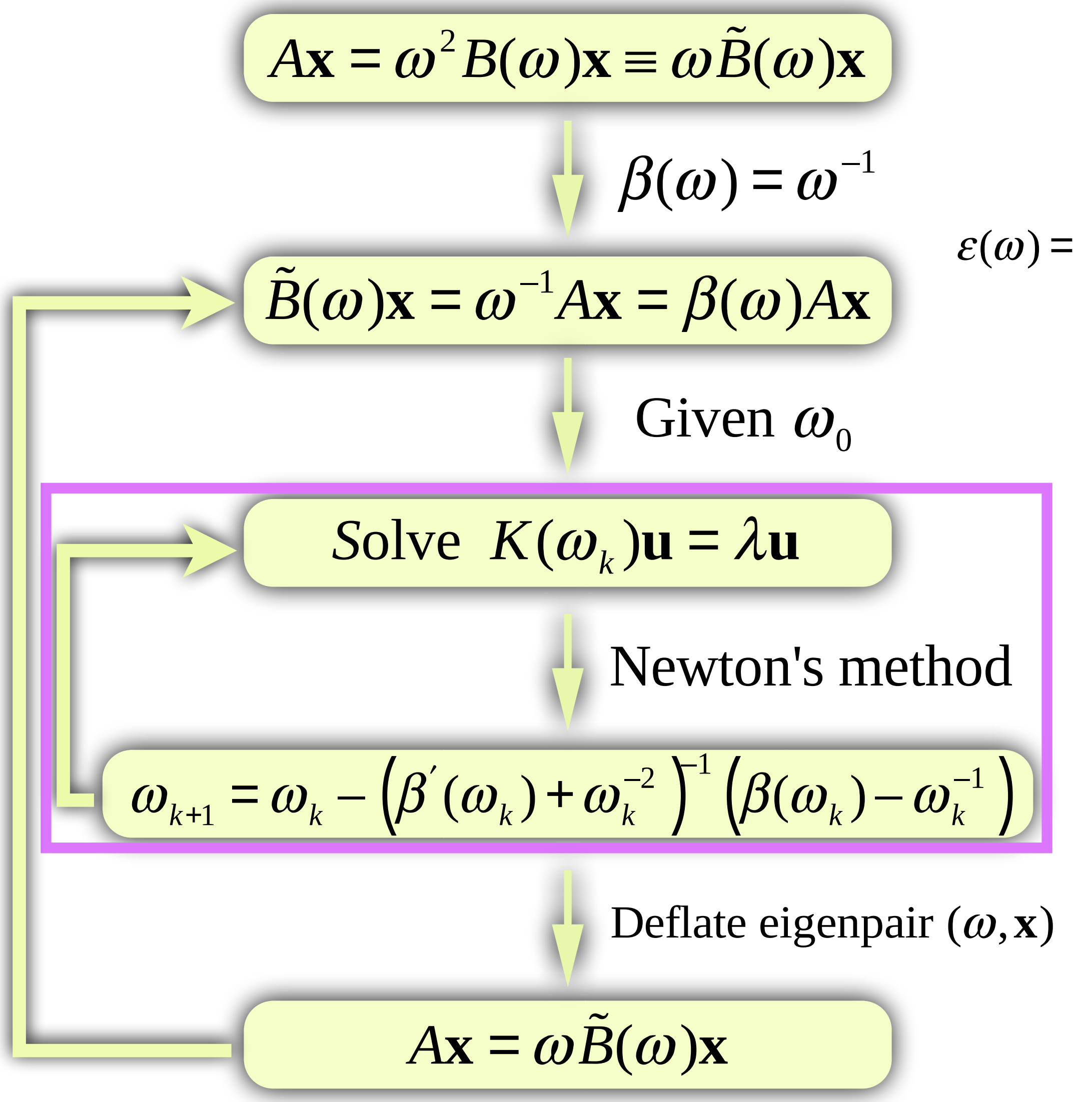
1	10
2	9
3	13
4	10
5	19
6	9
7	10
8	9
9	14
10	10



$$\varepsilon(\omega) = \varepsilon_\infty - \frac{\omega_p^2}{\omega^2 + \imath \Gamma_p \omega} + \sum_{j=1}^{2} \Omega_j A_j \left(\frac{e^{\imath \phi_j}}{\Omega_j - \omega - \imath \Gamma_j} + \frac{e^{-\imath \phi_j}}{\Omega_j + \omega + \imath \Gamma_j} \right)$$

1	10
2	12
3	17
4	11
5	22
6	13
7	11
8	10
9	13
10	9

Convergence of Newton-type method



$$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x} \equiv \omega\tilde{B}(\omega)\mathbf{x}$$

$$\beta(\omega) = \omega^{-1}$$

$$\tilde{B}(\omega)\mathbf{x} = \omega^{-1}A\mathbf{x} = \beta(\omega)A\mathbf{x}$$

Given ω_0

Solve $K(\omega_k)\mathbf{u} = \lambda\mathbf{u}$

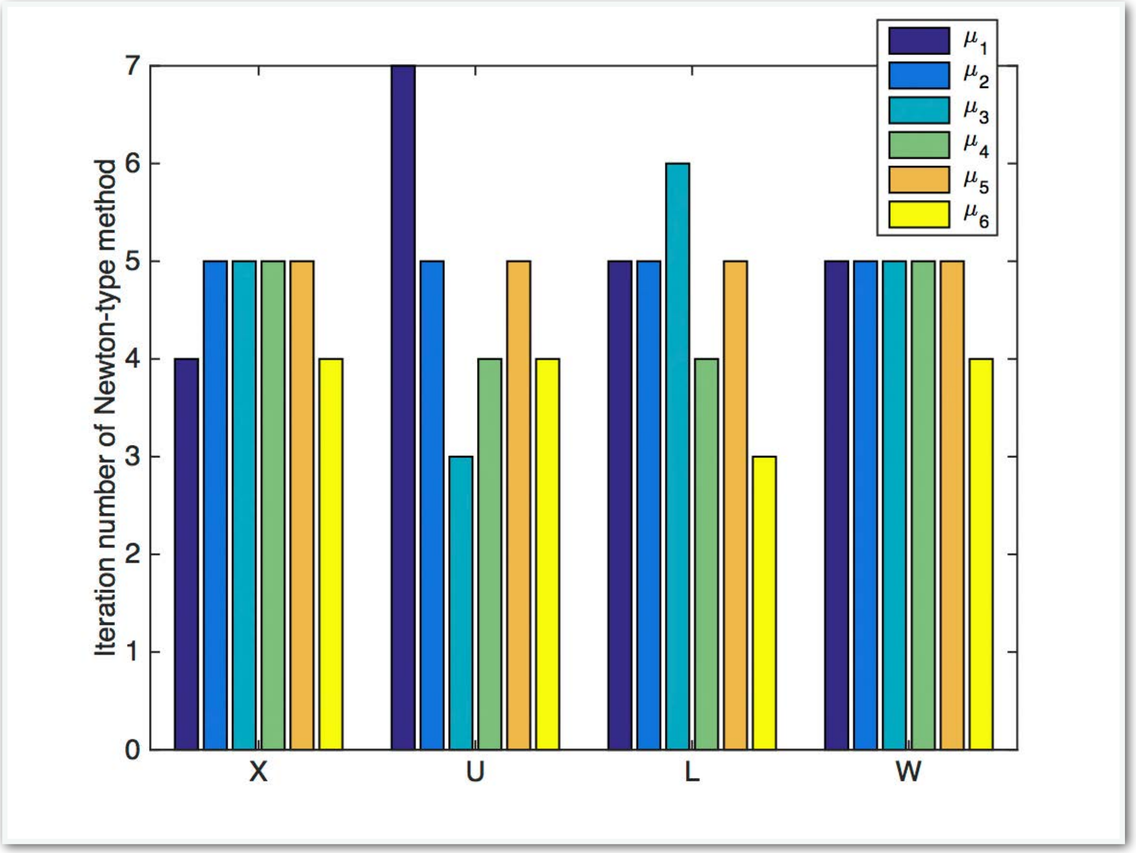
Newton's method

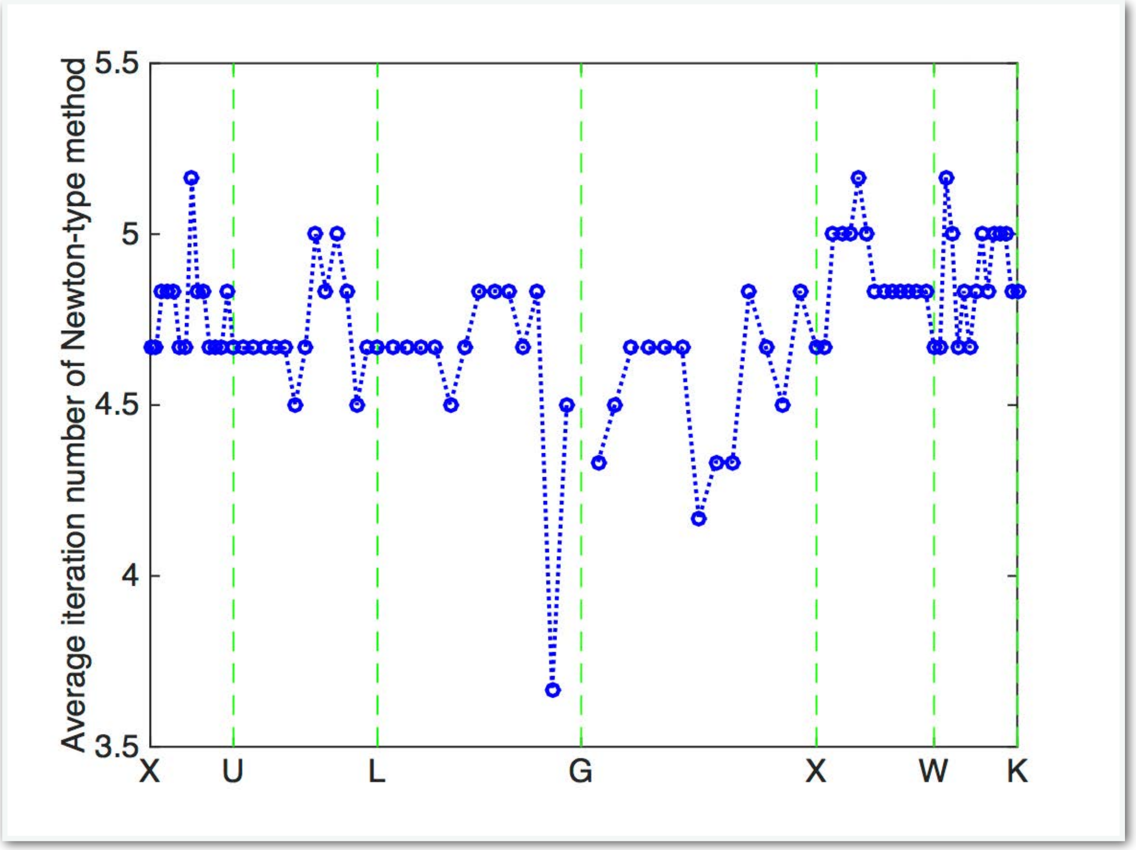
$$\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta(\omega_k) - \omega_k^{-1}\right)$$

Deflate eigenpair $(\omega, \mathbf{x})$

$$A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$$

$$\varepsilon(\omega) = 1 - \frac{\omega_p^2}{\omega^2 + \imath\Gamma_p\omega}$$

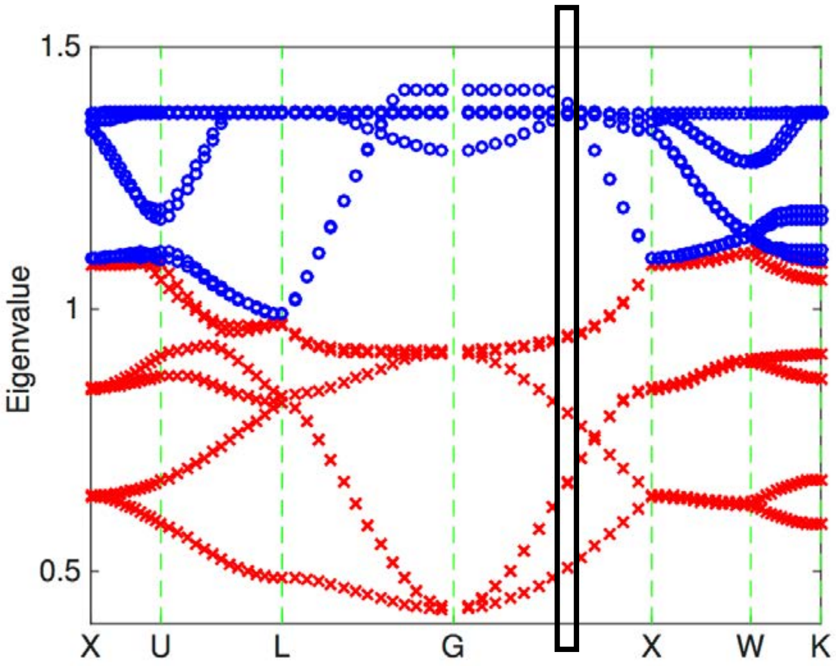




- Only 3 to 7 iterations are needed for computing each eigenvalue
- The average ranges from 3.6 to 5.2 for all benchmark problems.
- Quadratic convergence of Newton-type method

Clustering eigenvalues

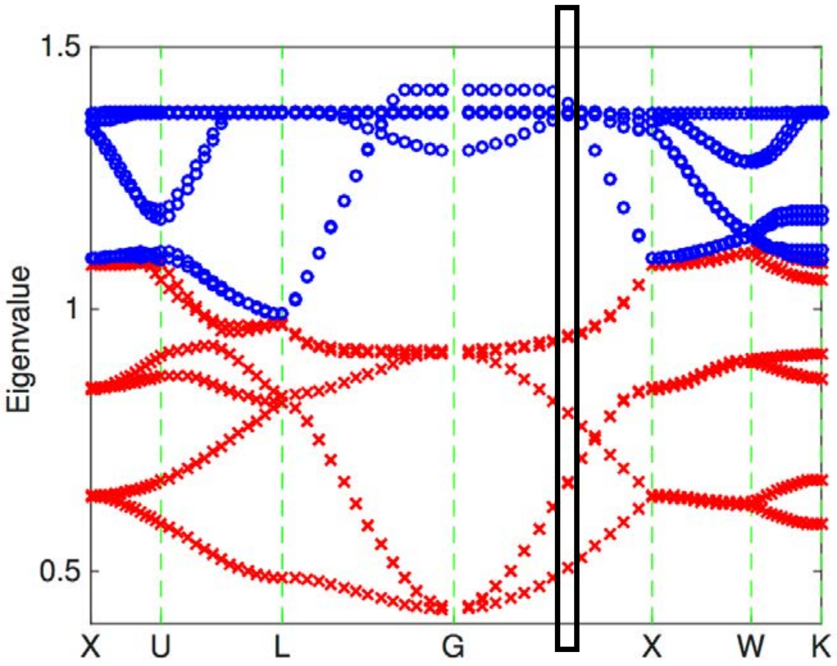
Clustering eigenvalues



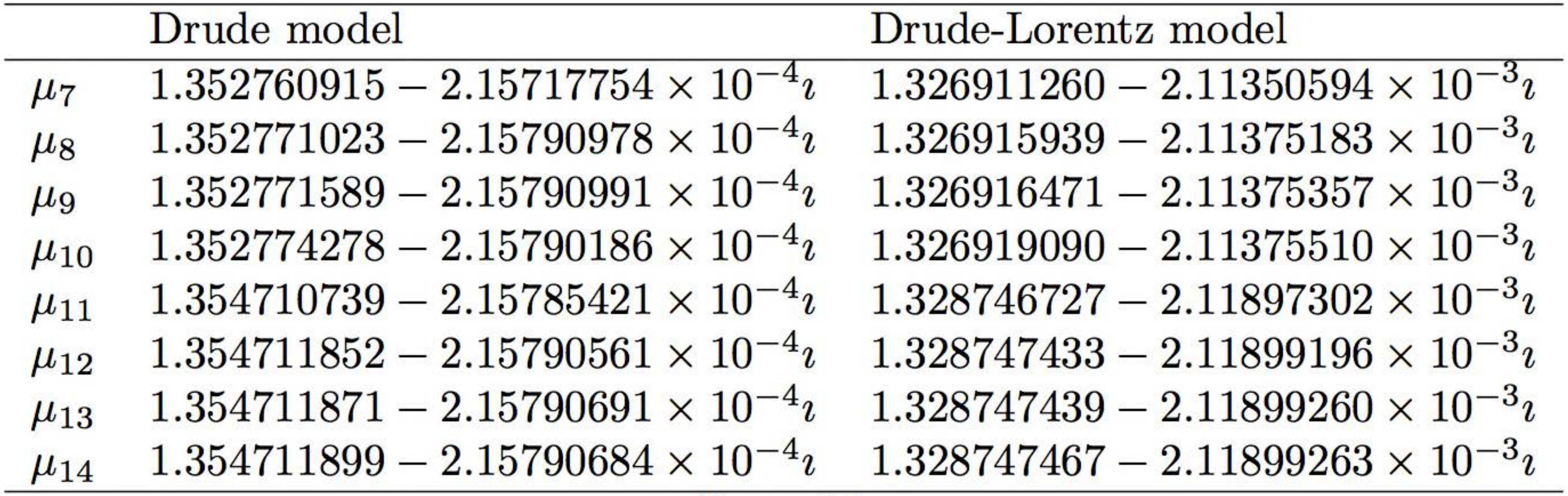
	Drude model	Drude-Lorentz model
μ_7	$1.352760915 - 2.15717754 \times 10^{-4}\imath$	$1.326911260 - 2.11350594 \times 10^{-3}\imath$
μ_8	$1.352771023 - 2.15790978 \times 10^{-4}\imath$	$1.326915939 - 2.11375183 \times 10^{-3}\imath$
μ_9	$1.352771589 - 2.15790991 \times 10^{-4}\imath$	$1.326916471 - 2.11375357 \times 10^{-3}\imath$
μ_{10}	$1.352774278 - 2.15790186 \times 10^{-4}\imath$	$1.326919090 - 2.11375510 \times 10^{-3}\imath$
μ_{11}	$1.354710739 - 2.15785421 \times 10^{-4}\imath$	$1.328746727 - 2.11897302 \times 10^{-3}\imath$
μ_{12}	$1.354711852 - 2.15790561 \times 10^{-4}\imath$	$1.328747433 - 2.11899196 \times 10^{-3}\imath$
μ_{13}	$1.354711871 - 2.15790691 \times 10^{-4}\imath$	$1.328747439 - 2.11899260 \times 10^{-3}\imath$
μ_{14}	$1.354711899 - 2.15790684 \times 10^{-4}\imath$	$1.328747467 - 2.11899263 \times 10^{-3}\imath$

TABLE 6.1

Clustering eigenvalues



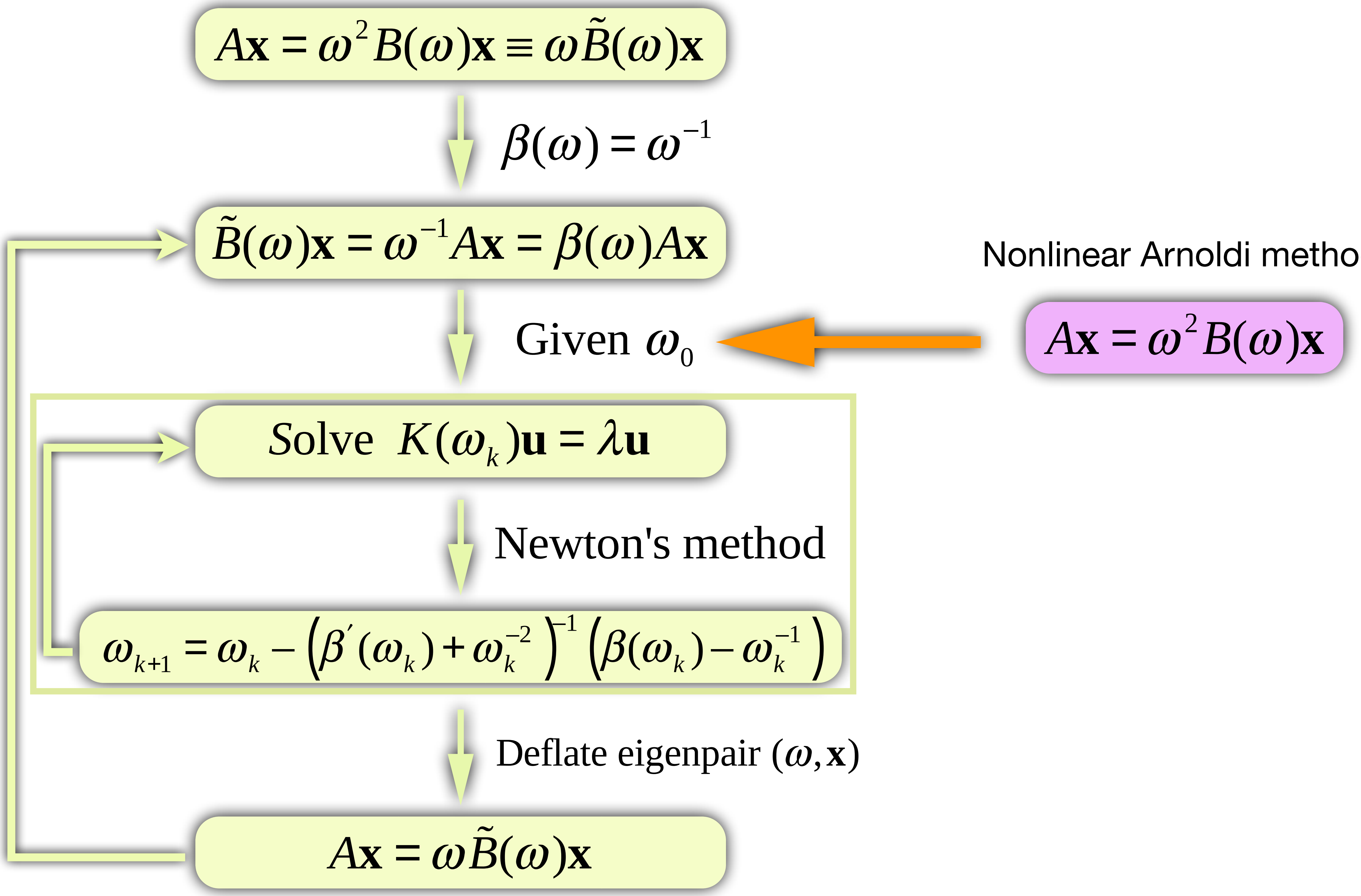
- Convergence is heavily dependent on the choice of the initial value $\omega_0^{(d)}$
- It is important to provide a good initial value to guarantee convergence
- Switch to solve a new approximate eigenpair of NLEVP roughly by nonlinear Arnoldi method



	Drude model	Drude-Lorentz model
μ_7	$1.352760915 - 2.15717754 \times 10^{-4}\imath$	$1.326911260 - 2.11350594 \times 10^{-3}\imath$
μ_8	$1.352771023 - 2.15790978 \times 10^{-4}\imath$	$1.326915939 - 2.11375183 \times 10^{-3}\imath$
μ_9	$1.352771589 - 2.15790991 \times 10^{-4}\imath$	$1.326916471 - 2.11375357 \times 10^{-3}\imath$
μ_{10}	$1.352774278 - 2.15790186 \times 10^{-4}\imath$	$1.326919090 - 2.11375510 \times 10^{-3}\imath$
μ_{11}	$1.354710739 - 2.15785421 \times 10^{-4}\imath$	$1.328746727 - 2.11897302 \times 10^{-3}\imath$
μ_{12}	$1.354711852 - 2.15790561 \times 10^{-4}\imath$	$1.328747433 - 2.11899196 \times 10^{-3}\imath$
μ_{13}	$1.354711871 - 2.15790691 \times 10^{-4}\imath$	$1.328747439 - 2.11899260 \times 10^{-3}\imath$
μ_{14}	$1.354711899 - 2.15790684 \times 10^{-4}\imath$	$1.328747467 - 2.11899263 \times 10^{-3}\imath$

TABLE 6.1

Initial data



$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x} \equiv \omega\tilde{B}(\omega)\mathbf{x}$

$\beta(\omega) = \omega^{-1}$

$\tilde{B}(\omega)\mathbf{x} = \omega^{-1}A\mathbf{x} = \beta(\omega)A\mathbf{x}$

Given ω_0

Nonlinear Arnoldi method

$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x}$

Solve $K(\omega_k)\mathbf{u} = \lambda\mathbf{u}$

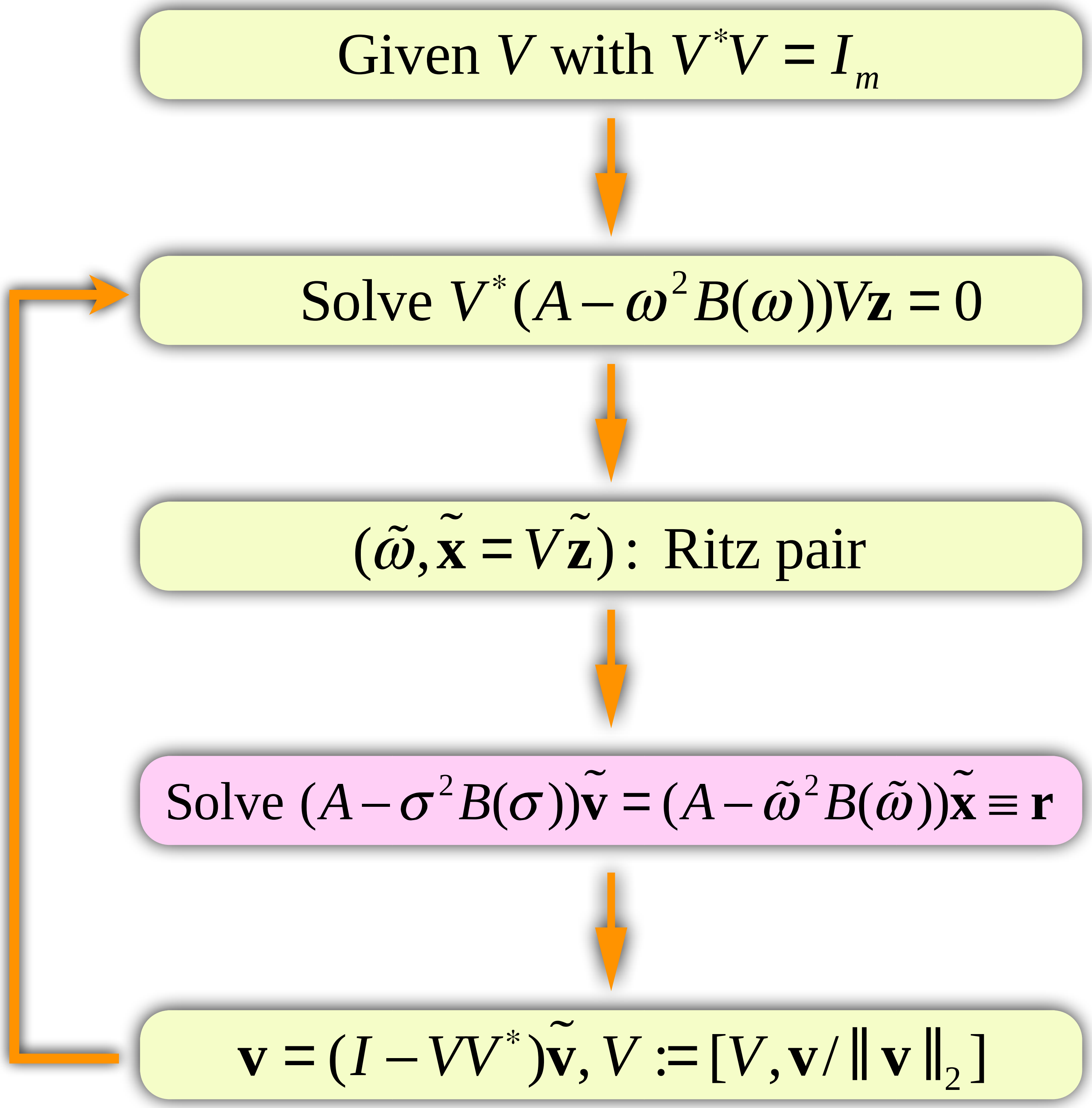
Newton's method

$\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta(\omega_k) - \omega_k^{-1}\right)$

Deflate eigenpair $(\omega, \mathbf{x})$

$A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$

Nonlinear Arnoldi method (NAr)



Given V with $V^*V = I_m$

Solve $V^*(A - \omega^2 B(\omega))V\mathbf{z} = 0$

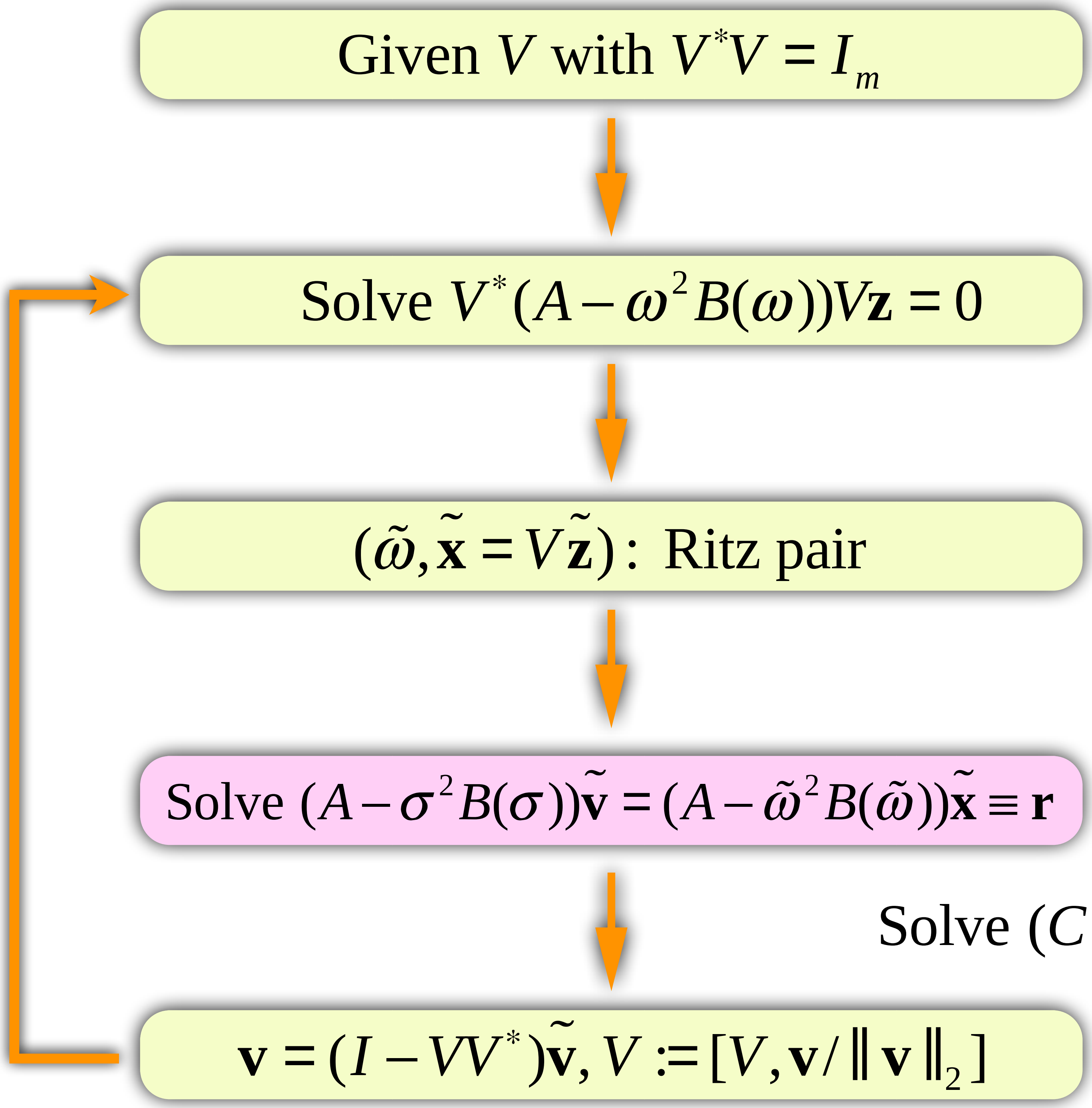
$(\tilde{\omega}, \tilde{\mathbf{x}} = V\tilde{\mathbf{z}})$: Ritz pair

Solve $(A - \sigma^2 B(\sigma))\tilde{\mathbf{v}} = (A - \tilde{\omega}^2 B(\tilde{\omega}))\tilde{\mathbf{x}} \equiv \mathbf{r}$

$\mathbf{v} = (I - VV^*)\tilde{\mathbf{v}}, V := [V, \mathbf{v}/\|\mathbf{v}\|_2]$

$$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x}$$

Nonlinear Arnoldi method (NAr)



Given V with $V^*V = I_m$

$A\mathbf{x} = \omega^2 B(\omega)\mathbf{x}$

Solve $V^*(A-\omega^2 B(\omega))V\mathbf{z} = 0$

$(\tilde{\omega}, \tilde{\mathbf{x}} = V\tilde{\mathbf{z}})$: Ritz pair

Solve $(A-\sigma^2 B(\sigma))\tilde{\mathbf{v}} = (A-\tilde{\omega}^2 B(\tilde{\omega}))\tilde{\mathbf{x}} \equiv \mathbf{r}$

$M = A - \tau I$

Solve $(C^*C - \tau I)\mathbf{y} = \mathbf{d}$ ✅

$\mathbf{v} = (I - VV^*)\tilde{\mathbf{v}}, V := [V, \mathbf{v}/\|\mathbf{v}\|_2]$

Solving preconditioning linear system

$$\boxed{(C^*C - \tau I)\mathbf{y} = \mathbf{d}}$$

Solving preconditioning linear system

$$\boxed{(C^*C - \tau I)\mathbf{y} = \mathbf{d}}$$

$$G = [C_1^\top, C_2^\top, C_3^\top]^\top$$

$$C^*C = I_3 \otimes \left(G^*G\right) - GG^*$$

$$C = \begin{bmatrix} 0 & -C_3 & C_2 \\ C_3 & 0 & -C_1 \\ -C_2 & C_1 & 0 \end{bmatrix}$$

Solving preconditioning linear system

$$(C^{*}C-\tau I)\mathbf{y}=\mathbf{d}$$

$$G=[C_1^{\top},C_2^{\top},C_3^{\top}]^{\top}$$

$$C^{*}C=I_3\otimes\left(G^{*}G\right)-GG^{*}$$

$$\left\{I_3\otimes(G^{*}G)-\tau I\right\}\mathbf{y}=\mathbf{d}+GG^{*}\mathbf{y}$$

$$C=\begin{bmatrix}0 & -C_3 & C_2\\ C_3 & 0 & -C_1\\ -C_2 & C_1 & 0\end{bmatrix}$$

Solving preconditioning linear system

$$\boxed{(C^*C - \tau I)\mathbf{y} = \mathbf{d}}$$

$$G = [C_1^\top, C_2^\top, C_3^\top]^\top$$

$$C^*C = I_3 \otimes \left(G^*G\right) - GG^*$$

$$\boxed{\left\{I_3 \otimes (G^*G) - \tau I\right\}\mathbf{y} = \mathbf{d} + GG^*\mathbf{y}}$$

$$C = \begin{bmatrix} 0 & -C_3 & C_2 \\ C_3 & 0 & -C_1 \\ -C_2 & C_1 & 0 \end{bmatrix}$$

$$CG = 0$$

$$GG^*\mathbf{y} = -\tau^{-1}GG^*\mathbf{d}$$

Solving preconditioning linear system

$$\boxed{(C^*C-\tau I)\mathbf{y}=\mathbf{d}}$$

$$G=[C_1^\top,C_2^\top,C_3^\top]^\top \qquad C^*C=I_3\otimes\left(G^*G\right)-GG^*$$

$$\boxed{\left\{I_3\otimes(G^*G)-\tau I\right\}\mathbf{y}=\mathbf{d}+GG^*\mathbf{y}}$$

$$C=\begin{bmatrix}0 & -C_3 & C_2\\ C_3 & 0 & -C_1\\ -C_2 & C_1 & 0\end{bmatrix} \qquad CG=0 \qquad GG^*\mathbf{y}=-\tau^{-1}GG^*\mathbf{d}$$

$$\boxed{\left\{I_3\otimes(G^*G)-\tau I\right\}\mathbf{y}=\mathbf{d}-\tau^{-1}GG^*\mathbf{d}}$$

Solving preconditioning linear system

$$\boxed{(C^*C-\tau I)\mathbf{y}=\mathbf{d}}$$

$$\Downarrow$$

$$G=[C_1^\top,C_2^\top,C_3^\top]^\top \qquad C^*C=I_3\otimes\left(G^*G\right)-GG^*$$

$$\boxed{\left\{I_3\otimes(G^*G)-\tau I\right\}\mathbf{y}=\mathbf{d}+GG^*\mathbf{y}}$$

$$\Downarrow$$

$$C=\begin{bmatrix}0 & -C_3 & C_2\\ C_3 & 0 & -C_1\\ -C_2 & C_1 & 0\end{bmatrix} \qquad CG=0 \qquad GG^*\mathbf{y}=-\tau^{-1}GG^*\mathbf{d}$$

$$\boxed{\left\{I_3\otimes(G^*G)-\tau I\right\}\mathbf{y}=\mathbf{d}-\tau^{-1}GG^*\mathbf{d}}$$

$$\Lambda_q=\Lambda_1^*\Lambda_1+\Lambda_2^*\Lambda_2+\Lambda_3^*\Lambda_3 \qquad C_1T=T\Lambda_1,\quad C_2T=T\Lambda_2,\quad C_3T=T\Lambda_3$$

Solving preconditioning linear system

$$(C^*C - \tau I)\mathbf{y} = \mathbf{d}$$

$$G = [C_1^\top, C_2^\top, C_3^\top]^\top \qquad C^*C = I_3 \otimes \left(G^*G\right) - GG^*$$

$$\left\{I_3 \otimes (G^*G) - \tau I\right\}\mathbf{y} = \mathbf{d} + GG^*\mathbf{y}$$

$$C = \begin{bmatrix} 0 & -C_3 & C_2 \\ C_3 & 0 & -C_1 \\ -C_2 & C_1 & 0 \end{bmatrix} \qquad CG = 0 \qquad GG^*\mathbf{y} = -\tau^{-1}GG^*\mathbf{d}$$

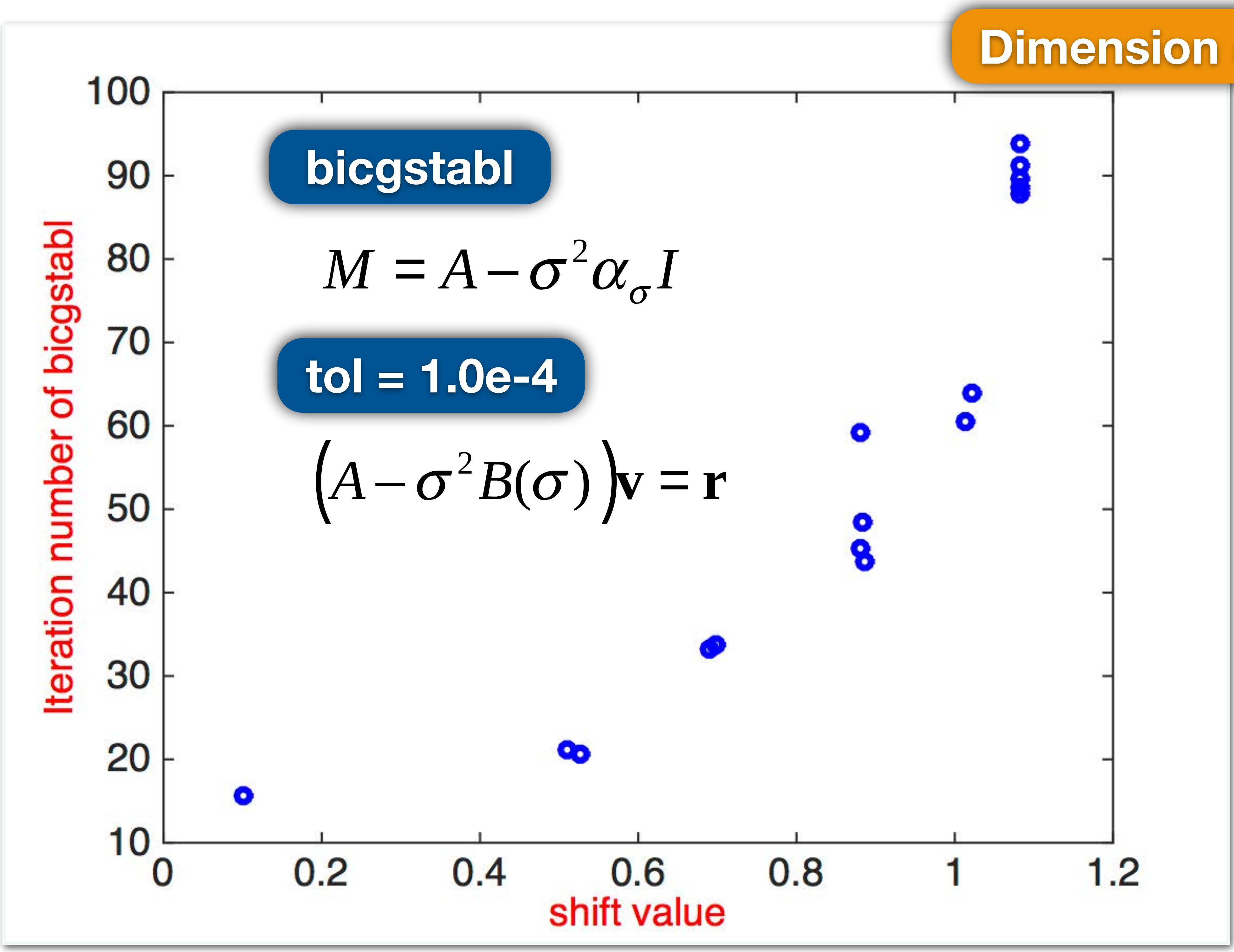
$$\left\{I_3 \otimes (G^*G) - \tau I\right\}\mathbf{y} = \mathbf{d} - \tau^{-1}GG^*\mathbf{d}$$

$$\Lambda_q = \Lambda_1^*\Lambda_1 + \Lambda_2^*\Lambda_2 + \Lambda_3^*\Lambda_3 \qquad C_1T = T\Lambda_1, \quad C_2T = T\Lambda_2, \quad C_3T = T\Lambda_3$$

$$\left(I_3 \otimes \Lambda_q - \tau I\right)\tilde{\mathbf{y}} = \left(I - \tau^{-1}\begin{bmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \end{bmatrix}\begin{bmatrix} \Lambda_1^* & \Lambda_2^* & \Lambda_3^* \end{bmatrix} \right)\left(I_3 \otimes T\right)^*\mathbf{d}, \quad \mathbf{y} = \left(I_3 \otimes T\right)\tilde{\mathbf{y}}$$

Efficiency of preconditioner

Dimension = 2,654,208



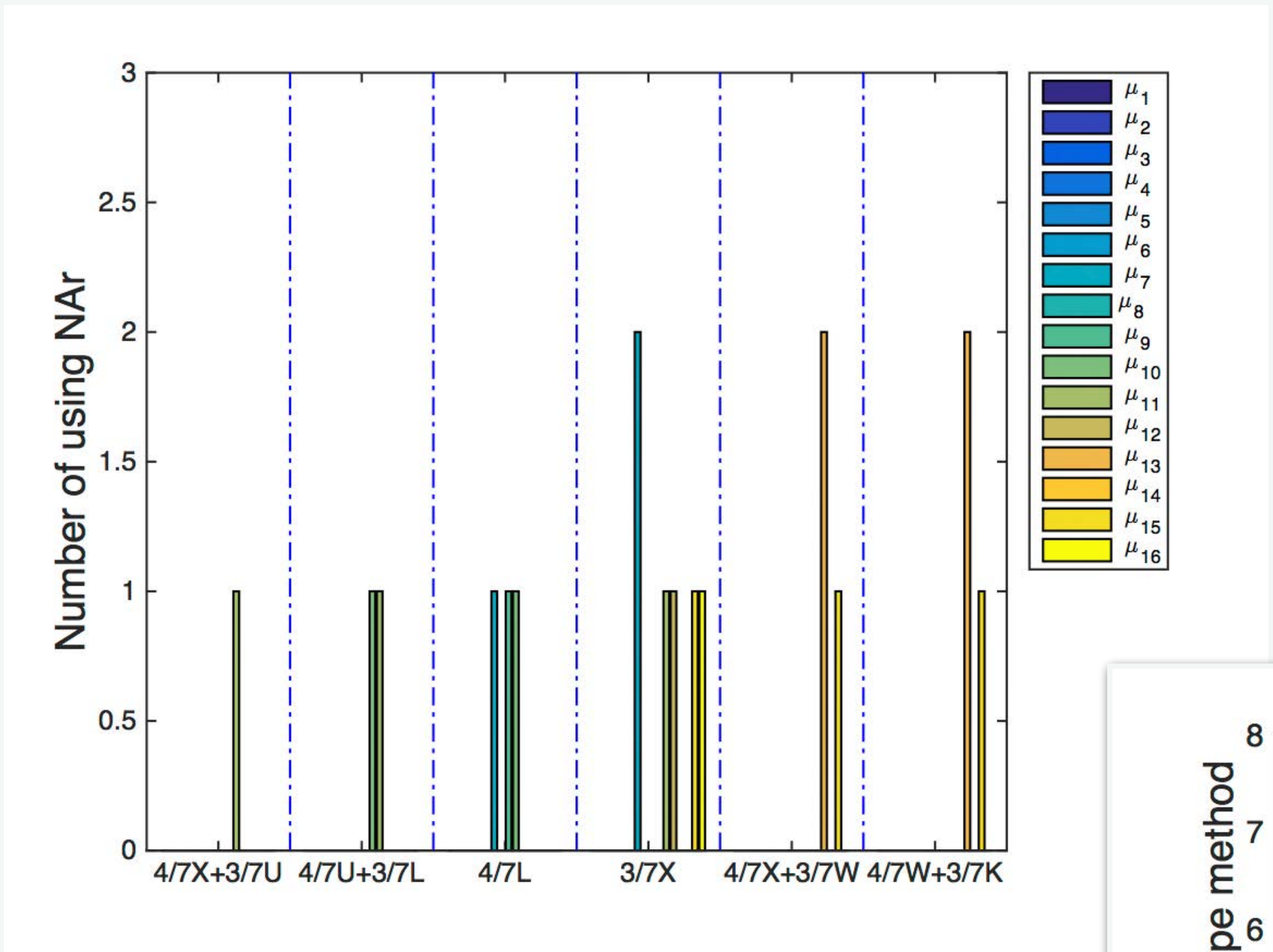
bicgstabl

$$M = A - \sigma^2 \alpha_\sigma I$$

tol = 1.0e-4

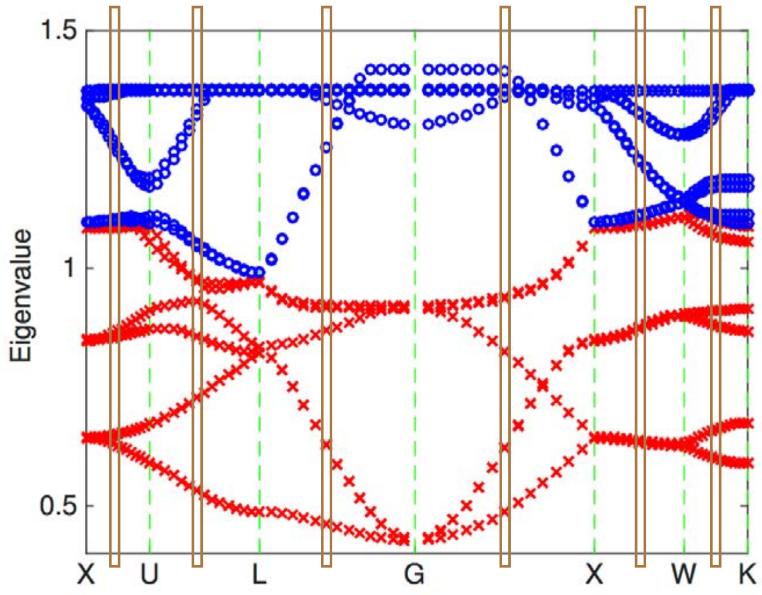
$$\left(A - \sigma^2 B(\sigma)\right)\mathbf{v} = \mathbf{r}$$

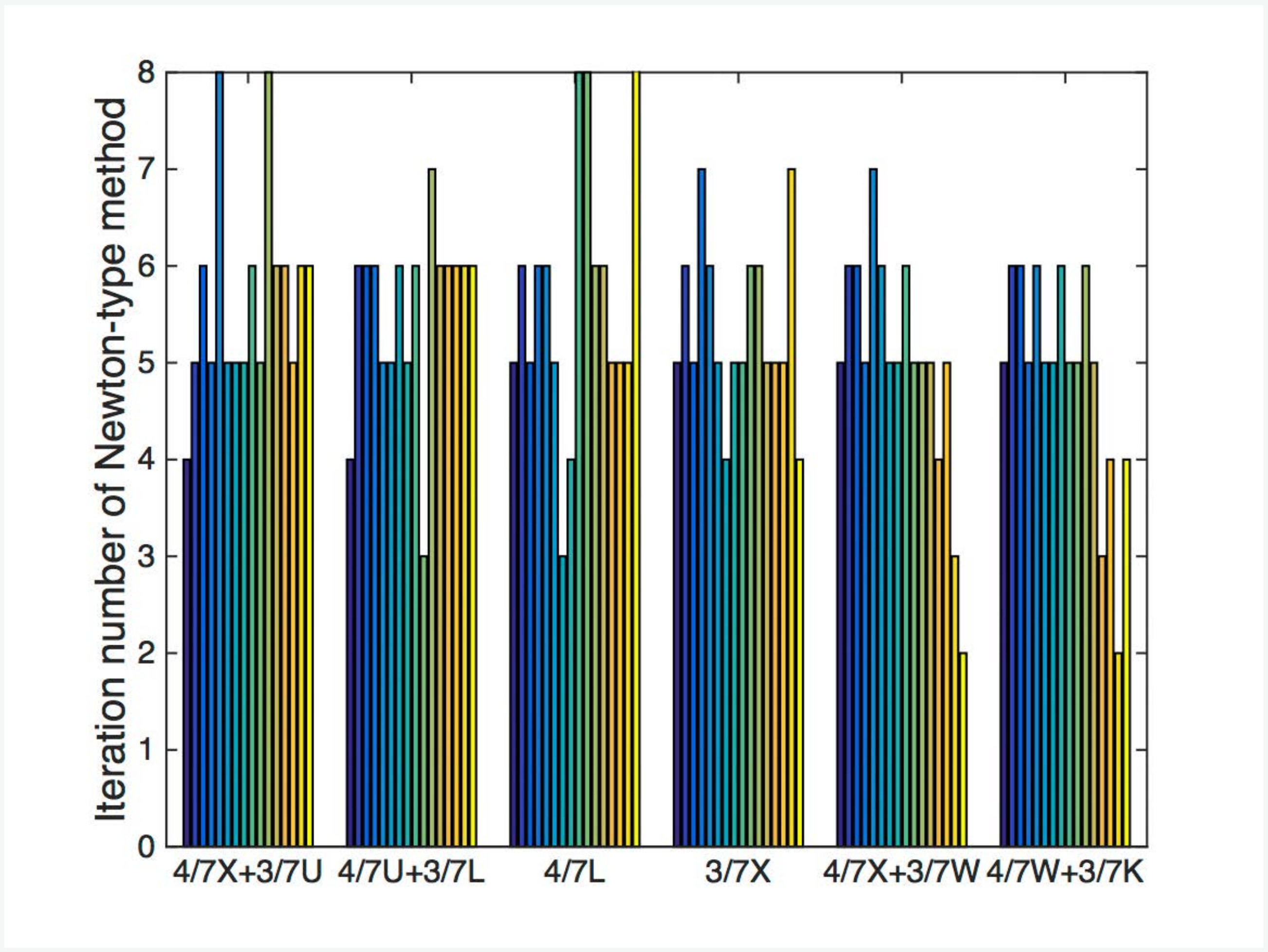
Results for Drude model



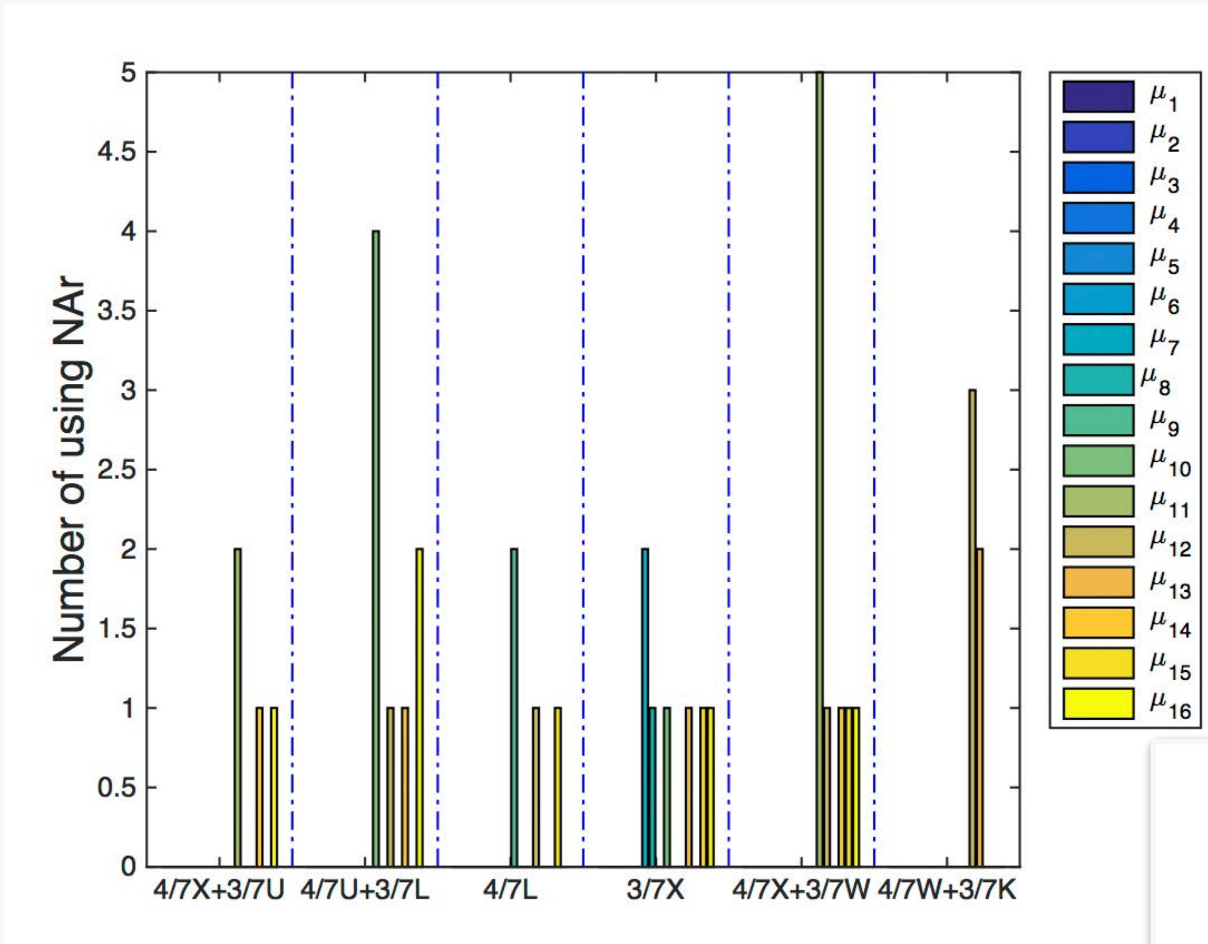
$$\varepsilon(\omega) = 1 - \frac{\omega_p^2}{\omega^2 + \imath\Gamma_p\omega}$$

$$K(\omega_k^{(d)})\mathbf{u} = \lambda\mathbf{u}, \ \text{for } k = 1,\ldots,m$$

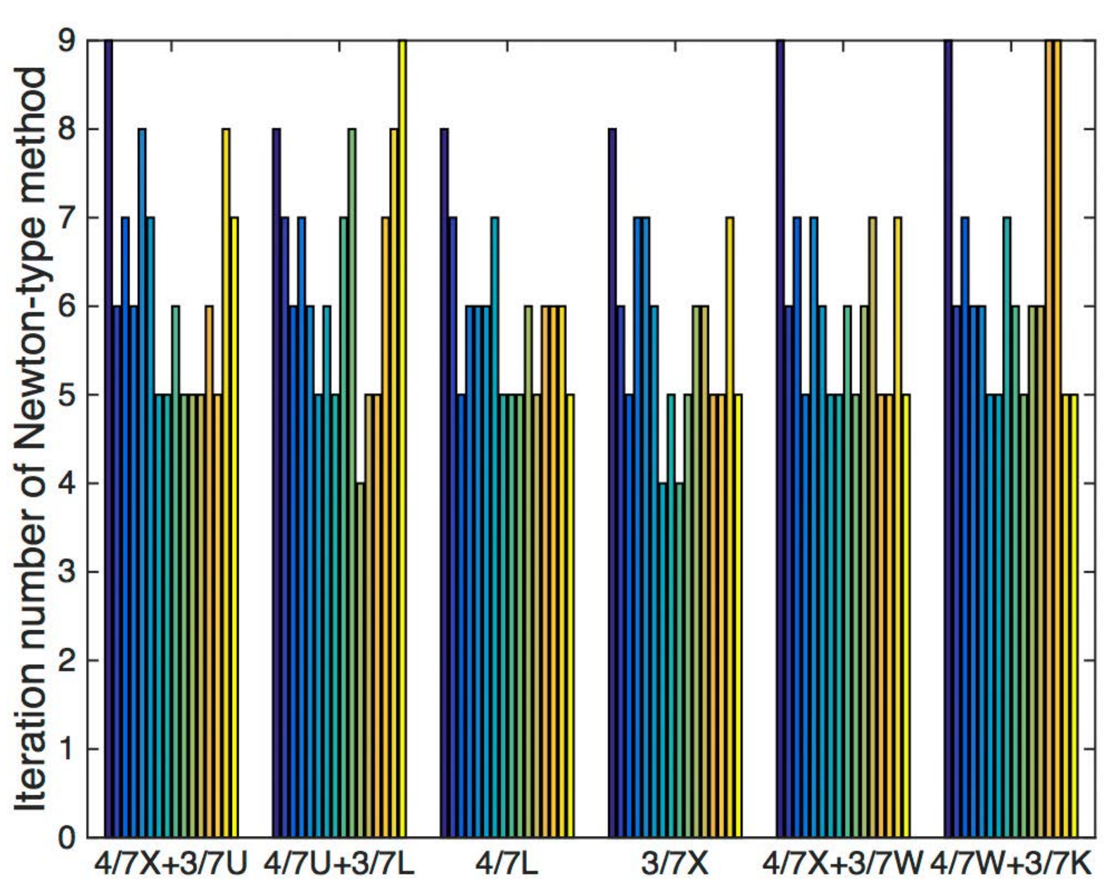




Results for Drude-Lorentz model



$$\varepsilon(\omega)=\varepsilon_\infty-\frac{\omega_p^2}{\omega^2+\imath\Gamma_p\omega}+\sum_{j=1}^{2}\Omega_jA_j\left(\frac{e^{\imath\phi_j}}{\Omega_j-\omega-\imath\Gamma_j}+\frac{e^{-\imath\phi_j}}{\Omega_j+\omega+\imath\Gamma_j}\right)$$



Summary

Conclusion

- Solving the nonlinear eigenvalue problem (NLEVP) arising from Yee's discretization of a three-dimensional dispersive metallic photonic crystal is a computational challenge.
- We have proposed a Newton-type method to compute one desired eigenpair of the NLEVP at a time.
- Once the desired eigenvalue is converged, it is then transformed to infinity by the proposed non-equivalence deflation scheme, while all other eigenvalues remain unchanged. The next successive eigenvalue thus becomes the smallest nonzero real part eigenvalue of the transformed NLEVP which is then again solved by the Newton-type method.
- In order to compute the clustering eigenvalues of the NLEVP, we propose a hybrid method by using the Jacobi-Davidson to solve the standard eigenvalue problems in the Newton-type method and the NAr to compute the initial data.
- The numerical results demonstrate that our proposed method is robust for solving both of well-separated and clustering eigenvalues of the NLEVP for the Drude and Drude-Lorentz models.

Thank you.

Backup Slides

Dispersive Maxwell equations

Nonlinear eigenvalue problems

Newton-type Methods for Solving

$$A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$$

Numerical results

Dispersive Maxwell equations

Nonlinear eigenvalue problems

Newton-type Methods for Solving

$$A\mathbf{x} = \omega \tilde{B}(\omega)\mathbf{x}$$

Numerical results

Dispersive Maxwell equations

Nonlinear eigenvalue problems

Newton-type Methods for Solving

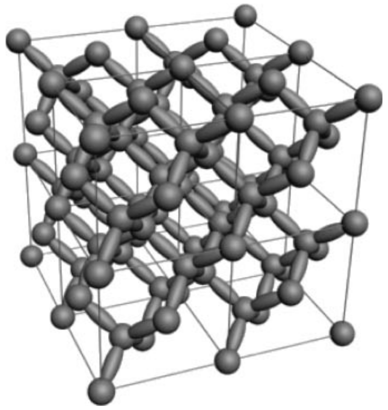
$$A\mathbf{x} = \omega \tilde{B}(\omega)\mathbf{x}$$

Numerical results

Wave vector k

$$E(\mathbf{r}+\mathbf{a}_\ell)=e^{\mathrm{i}2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r})$$

- Compute the band structure along the irreducible Brillouin zone for the lattice



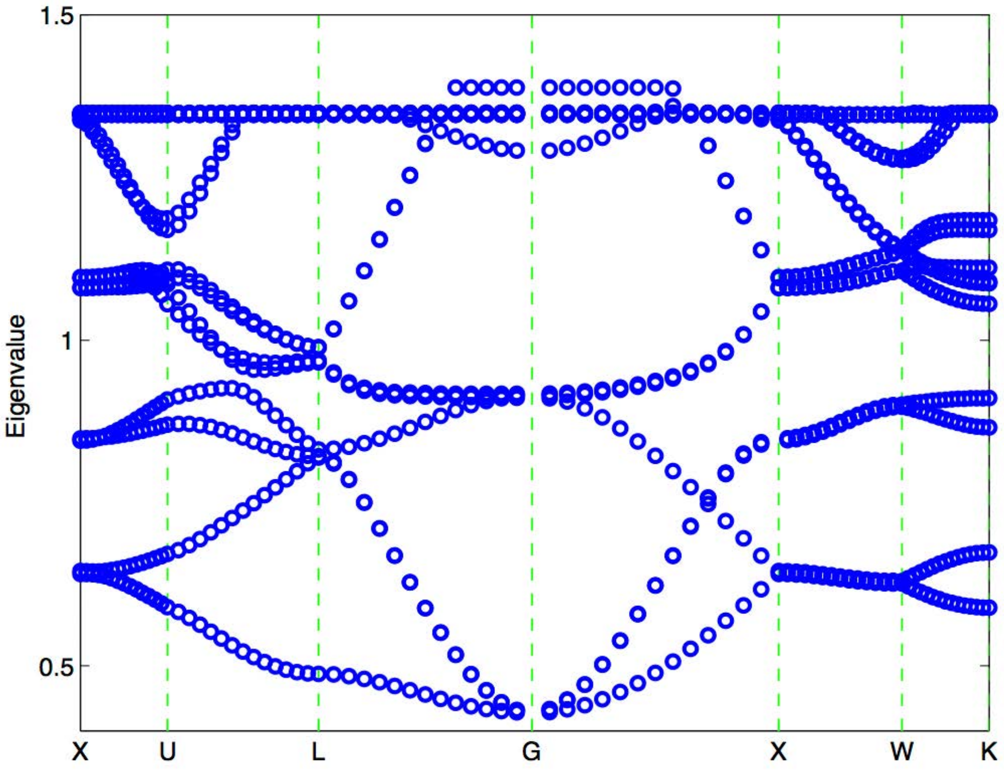
- FCC

$$X=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\to U=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{4}\\1\\\frac{1}{4}\end{bmatrix}\to L=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\\frac{1}{2}\\\frac{1}{2}\end{bmatrix}\to G=\begin{bmatrix}0\\0\\0\end{bmatrix}\to X\to W=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\to K=\frac{2\pi}{a}\begin{bmatrix}\frac{3}{4}\\\frac{3}{4}\\0\end{bmatrix}$$

- Eigenvalue problem depends on wave vector k

$$\nabla\times\nabla\times E(\mathbf{r})=\omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})$$

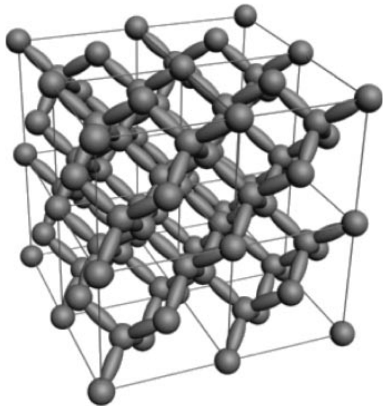
- A sequence of EVP need to solve



Wave vector k

$$E(\mathbf{r}+\mathbf{a}_\ell) = e^{i2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r})$$

- Compute the band structure along the irreducible Brillouin zone for the lattice



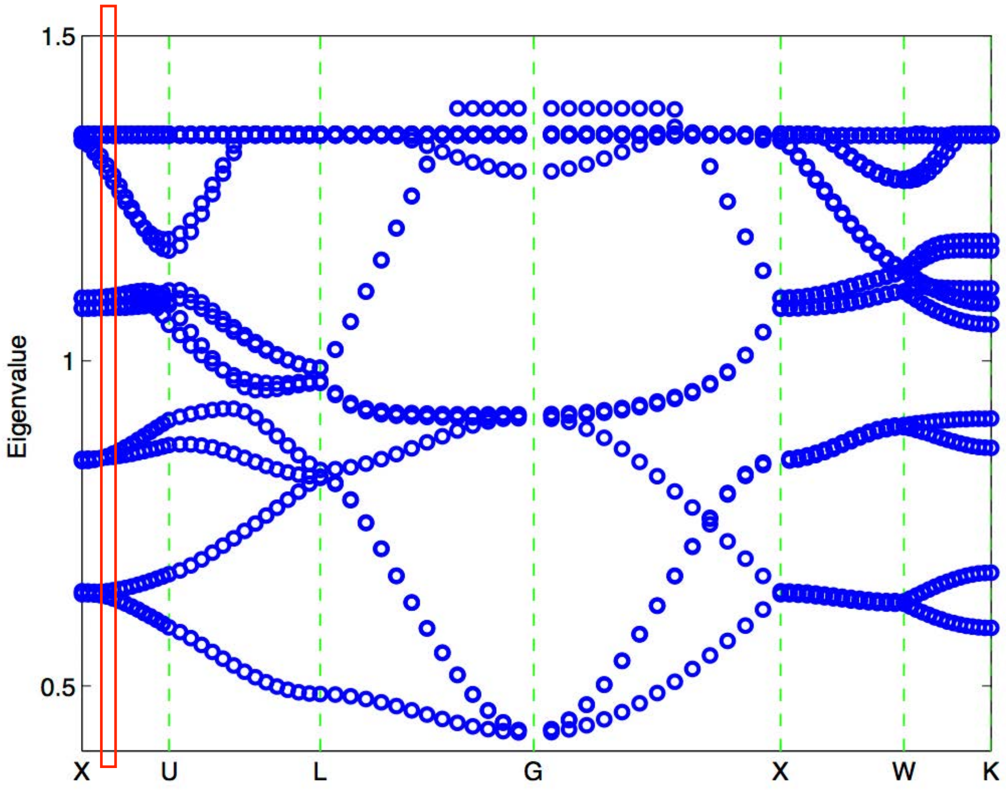
- FCC

$$X = \frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\ 1\\ 0\end{bmatrix} \to U = \frac{2\pi}{a}\begin{bmatrix}\frac{1}{4}\\ 1\\ \frac{1}{4}\end{bmatrix} \to L = \frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\ \frac{1}{2}\\ \frac{1}{2}\end{bmatrix} \to G = \begin{bmatrix}0\\ 0\\ 0\end{bmatrix} \to X \to W = \frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\ 1\\ 0\end{bmatrix} \to K = \frac{2\pi}{a}\begin{bmatrix}\frac{3}{4}\\ \frac{3}{4}\\ 0\end{bmatrix}$$

- Eigenvalue problem depends on wave vector k

$$\nabla\times\nabla\times E(\mathbf{r}) = \omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})$$

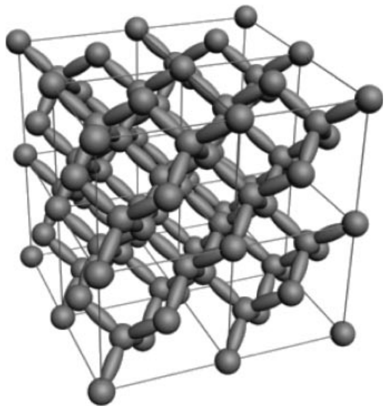
- A sequence of EVP need to solve



Wave vector k

$$E(\mathbf{r}+\mathbf{a}_\ell) = e^{\mathrm{i}2\pi\mathbf{k}\cdot\mathbf{a}_\ell} E(\mathbf{r})$$

- Compute the band structure along the irreducible Brillouin zone for the lattice



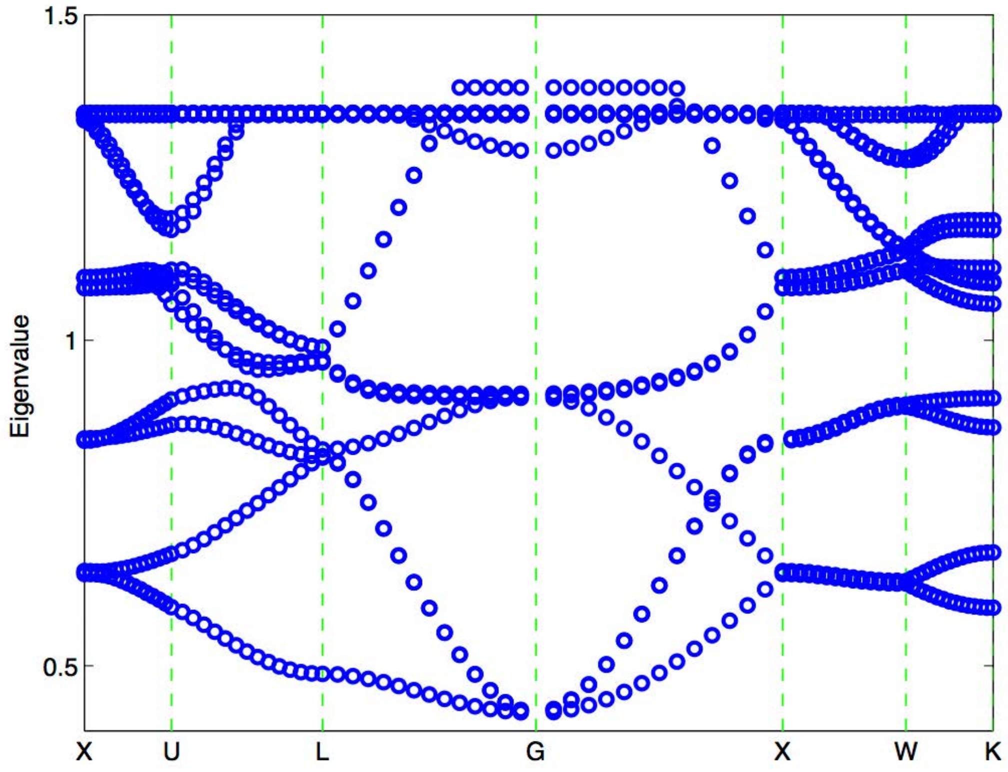
- FCC

$$X = \frac{2\pi}{a}\begin{bmatrix} \frac{1}{2} \\ 1 \\ 0 \end{bmatrix} \to U = \frac{2\pi}{a}\begin{bmatrix} \frac{1}{4} \\ 1 \\ \frac{1}{4} \end{bmatrix} \to L = \frac{2\pi}{a}\begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{bmatrix} \to G = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \to X \to W = \frac{2\pi}{a}\begin{bmatrix} \frac{1}{2} \\ 1 \\ 0 \end{bmatrix} \to K = \frac{2\pi}{a}\begin{bmatrix} \frac{3}{4} \\ \frac{3}{4} \\ 0 \end{bmatrix}$$

- Eigenvalue problem depends on wave vector k

$$\nabla \times \nabla \times E(\mathbf{r}) = \omega^2 \varepsilon(\mathbf{r},\omega) E(\mathbf{r})$$

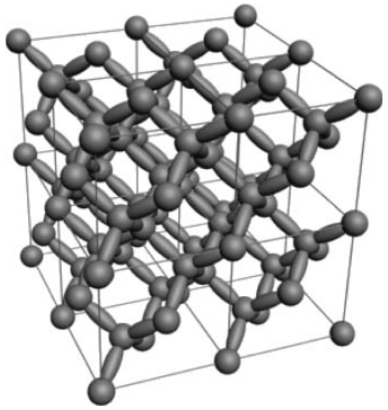
- A sequence of EVP need to solve



Wave vector k

$$E(\mathbf{r}+\mathbf{a}_\ell)=e^{i2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r})$$

- Compute the band structure along the irreducible Brillouin zone for the lattice



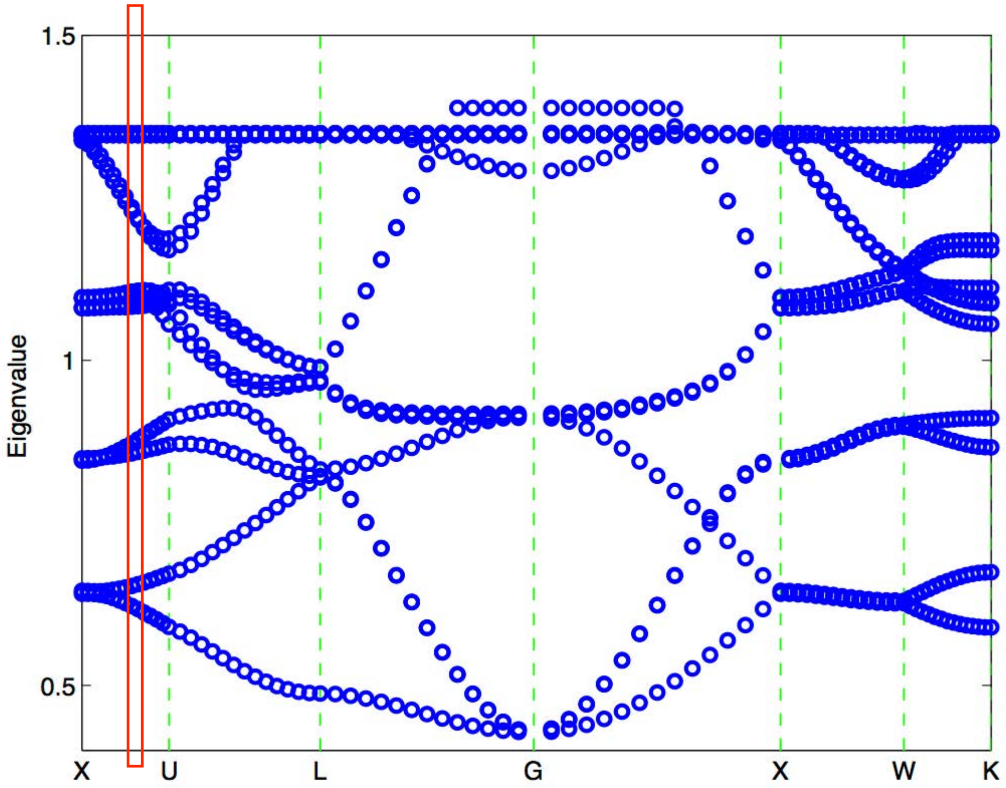
- FCC

$$X=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\rightarrow U=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{4}\\1\\\frac{1}{4}\end{bmatrix}\rightarrow L=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\\frac{1}{2}\\\frac{1}{2}\end{bmatrix}\rightarrow G=\begin{bmatrix}0\\0\\0\end{bmatrix}\rightarrow X\rightarrow W=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\rightarrow K=\frac{2\pi}{a}\begin{bmatrix}\frac{3}{4}\\\frac{3}{4}\\0\end{bmatrix}$$

- Eigenvalue problem depends on wave vector k

$$\nabla\times\nabla\times E(\mathbf{r})=\omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})$$

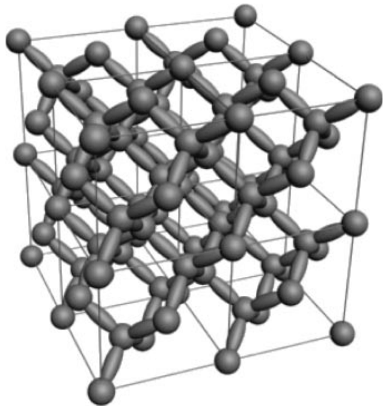
- A sequence of EVP need to solve



Wave vector k

$$E(\mathbf{r}+\mathbf{a}_\ell)=e^{\mathrm{i}2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r})$$

- Compute the band structure along the irreducible Brillouin zone for the lattice



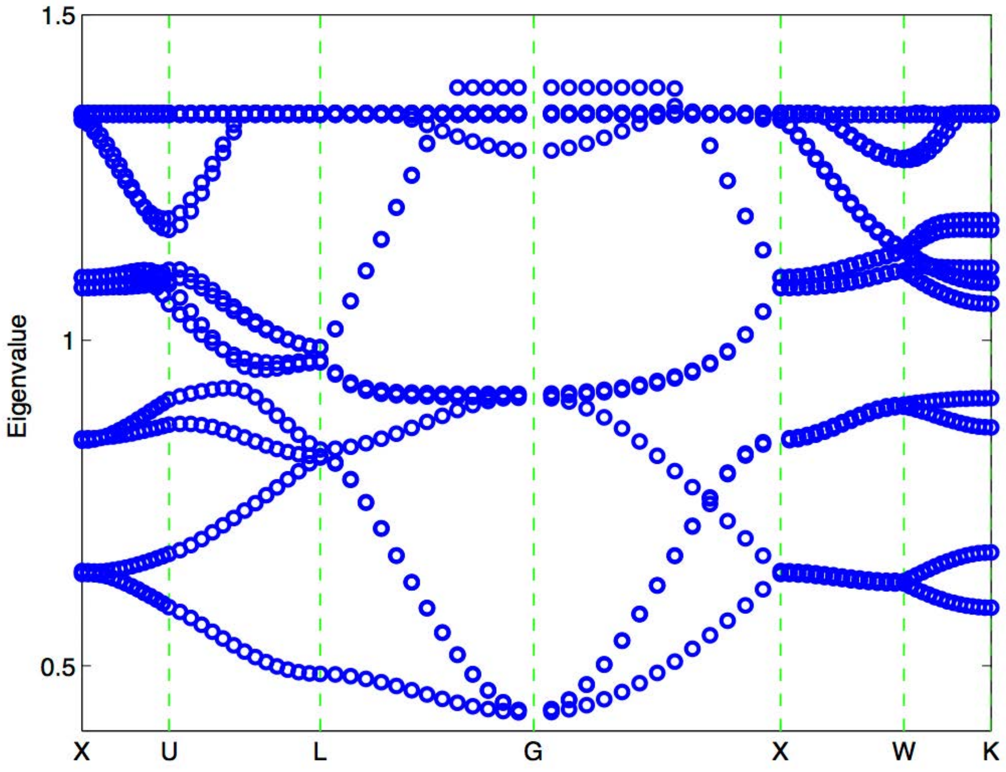
- FCC

$$X=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\rightarrow U=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{4}\\1\\\frac{1}{4}\end{bmatrix}\rightarrow L=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\\frac{1}{2}\\\frac{1}{2}\end{bmatrix}\rightarrow G=\begin{bmatrix}0\\0\\0\end{bmatrix}\rightarrow X\rightarrow W=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\rightarrow K=\frac{2\pi}{a}\begin{bmatrix}\frac{3}{4}\\\frac{3}{4}\\0\end{bmatrix}$$

- Eigenvalue problem depends on wave vector k

$$\nabla\times\nabla\times E(\mathbf{r})=\omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})$$

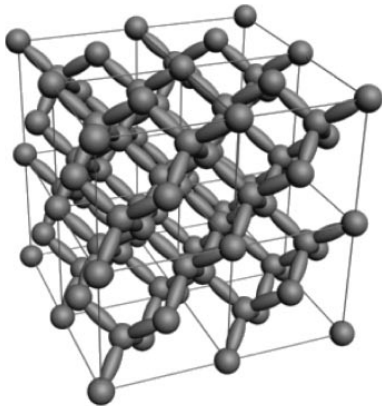
- A sequence of EVP need to solve



Wave vector k

$$E(\mathbf{r}+\mathbf{a}_\ell)=e^{\mathrm{i}2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r})$$

- Compute the band structure along the irreducible Brillouin zone for the lattice



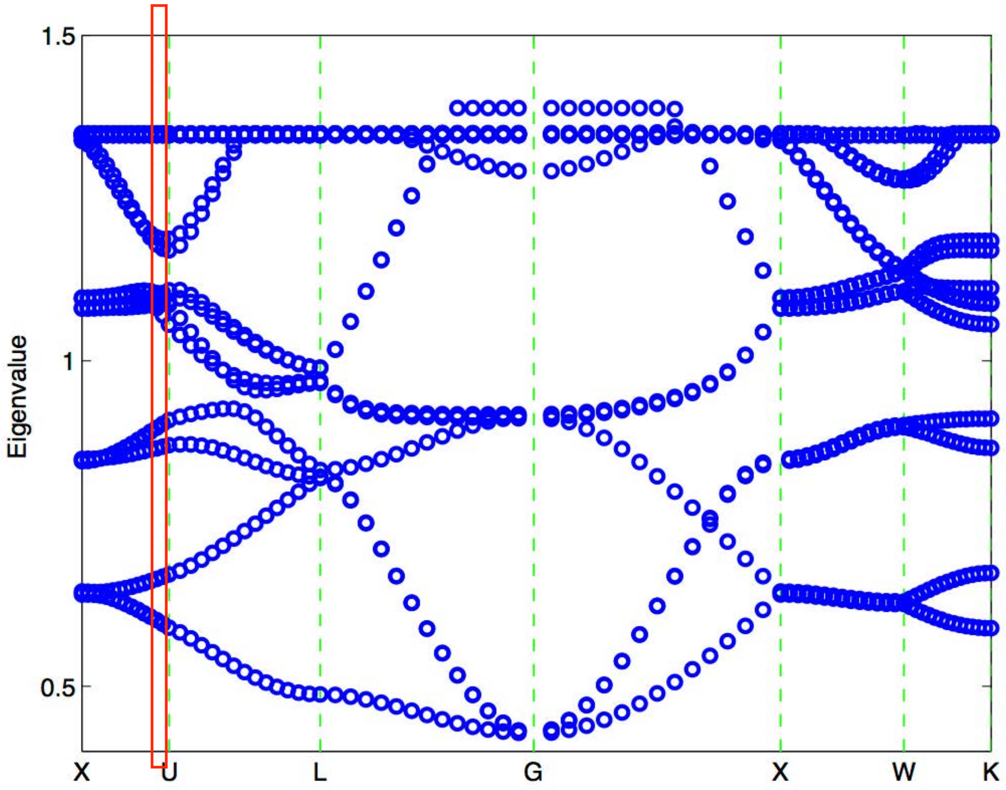
- FCC

$$X=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\rightarrow U=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{4}\\1\\\frac{1}{4}\end{bmatrix}\rightarrow L=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\\frac{1}{2}\\\frac{1}{2}\end{bmatrix}\rightarrow G=\begin{bmatrix}0\\0\\0\end{bmatrix}\rightarrow X\rightarrow W=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\rightarrow K=\frac{2\pi}{a}\begin{bmatrix}\frac{3}{4}\\\frac{3}{4}\\0\end{bmatrix}$$

- Eigenvalue problem depends on wave vector k

$$\nabla\times\nabla\times E(\mathbf{r})=\omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})$$

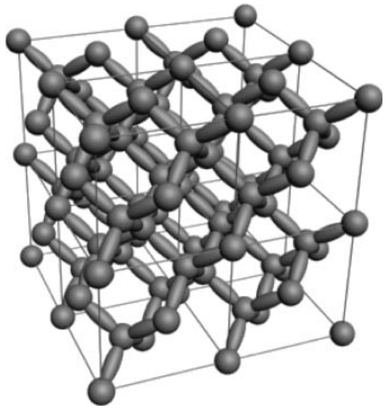
- A sequence of EVP need to solve



Wave vector k

$$E(\mathbf{r}+\mathbf{a}_\ell)=e^{i2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r})$$

- Compute the band structure along the irreducible Brillouin zone for the lattice



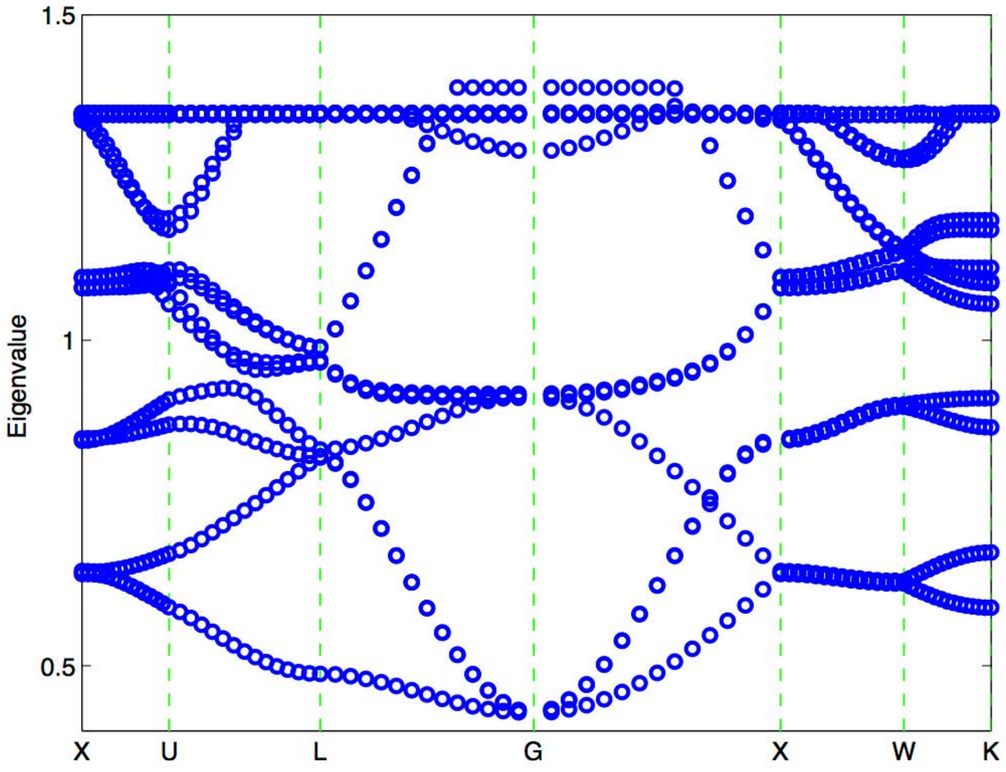
- FCC

$$X=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\rightarrow U=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{4}\\1\\\frac{1}{4}\end{bmatrix}\rightarrow L=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\\frac{1}{2}\\\frac{1}{2}\end{bmatrix}\rightarrow G=\begin{bmatrix}0\\0\\0\end{bmatrix}\rightarrow X\rightarrow W=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\1\\0\end{bmatrix}\rightarrow K=\frac{2\pi}{a}\begin{bmatrix}\frac{3}{4}\\\frac{3}{4}\\0\end{bmatrix}$$

- Eigenvalue problem depends on wave vector k

$$\nabla\times\nabla\times E(\mathbf{r})=\omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})$$

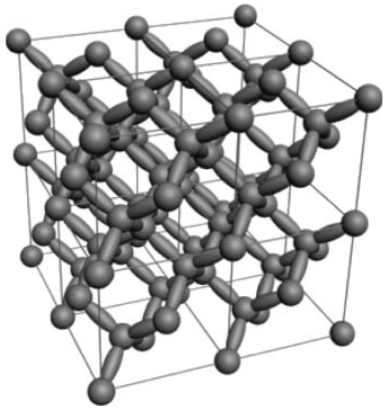
- A sequence of EVP need to solve



Wave vector k

$$E(\mathbf{r}+\mathbf{a}_\ell)=e^{i2\pi\mathbf{k}\cdot\mathbf{a}_\ell}E(\mathbf{r})$$

- Compute the band structure along the irreducible Brillouin zone for the lattice



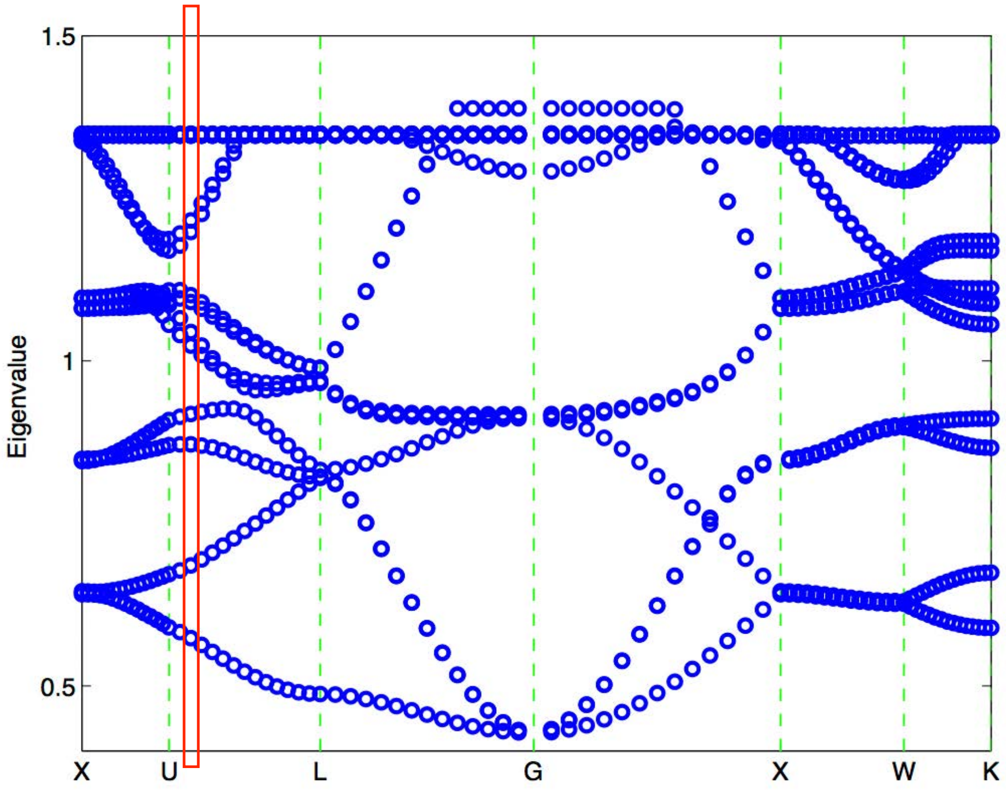
- FCC

$$X=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\ 1\\ 0\end{bmatrix}\to U=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{4}\\ 1\\ \frac{1}{4}\end{bmatrix}\to L=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\ \frac{1}{2}\\ \frac{1}{2}\end{bmatrix}\to G=\begin{bmatrix}0\\ 0\\ 0\end{bmatrix}\to X\to W=\frac{2\pi}{a}\begin{bmatrix}\frac{1}{2}\\ 1\\ 0\end{bmatrix}\to K=\frac{2\pi}{a}\begin{bmatrix}\frac{3}{4}\\ \frac{3}{4}\\ 0\end{bmatrix}$$

- Eigenvalue problem depends on wave vector k

$$\nabla\times\nabla\times E(\mathbf{r})=\omega^2\varepsilon(\mathbf{r},\omega)E(\mathbf{r})$$

- A sequence of EVP need to solve



Nonlinear Jacobi-Davidson method (NJD)

- For a given search subspace V, let $(\tilde{\omega}, \tilde{\mathbf{z}})$ be an eigenpair of

$$V^*(A-\omega^2 B(\omega))V\mathbf{z} = 0$$

and let $\tilde{\mathbf{x}} = V\tilde{\mathbf{z}}$ be the associated Ritz vector

- The new search direction v is chosen as

$$\left(I - \frac{(2\tilde{\omega}B(\tilde{\omega}) + \tilde{\omega}^2 B'(\tilde{\omega}))\tilde{\mathbf{x}}\tilde{\mathbf{x}}^*}{\tilde{\mathbf{x}}^*(2\tilde{\omega}B(\tilde{\omega}) + \tilde{\omega}^2 B'(\tilde{\omega}))\tilde{\mathbf{x}}} \right)(A-\tilde{\omega}^2 B(\tilde{\omega}))\left(I - \frac{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^*}{\tilde{\mathbf{x}}^*\tilde{\mathbf{x}}} \right)\mathbf{v} = -\mathbf{r}, \quad \mathbf{v} \perp \tilde{\mathbf{x}}$$

where σ is a given shift value. We employ a preconditioner

$$M_J = \left(I - \frac{(2\tilde{\omega}B(\tilde{\omega}) + \tilde{\omega}^2 B'(\tilde{\omega}))\tilde{\mathbf{x}}\tilde{\mathbf{x}}^*}{\tilde{\mathbf{x}}^*(2\tilde{\omega}B(\tilde{\omega}) + \tilde{\omega}^2 B'(\tilde{\omega}))\tilde{\mathbf{x}}} \right)(A-\tilde{\omega}^2 \alpha_\sigma I)\left(I - \frac{\tilde{\mathbf{x}}\tilde{\mathbf{x}}^*}{\tilde{\mathbf{x}}^*\tilde{\mathbf{x}}} \right)$$

- After re-orthogonalizing v against V, the vector is appended to V and one repeats this process until $(\tilde{\omega}, \tilde{\mathbf{x}})$ converges to the desired eigenpair.

Definitions

- Represent $F(\omega)$ as

$$F(\omega) = P(\omega) + R(\omega)$$

where $P(\omega)$ is a polynomial matrix of degree r and $R(\omega)$ is a rational polynomial matrix with entries being proper rational polynomial.

Definitions

- Represent $F(\omega)$ as

$$F(\omega) = P(\omega) + R(\omega)$$

where $P(\omega)$ is a polynomial matrix of degree r and $R(\omega)$ is a rational polynomial matrix with entries being proper rational polynomial.

- If $\omega_0 \in \mathbb{C}$ and nonzero vector $\mathbf{x}$ satisfy

$$\det(F(\omega_0)) = 0, \quad F(\omega_0)\mathbf{x} = 0$$

then $(\omega_0, \mathbf{x})$ is called an eigenpair of $F(\omega)$

Definitions

- Represent $F(\omega)$ as

$$F(\omega) = P(\omega) + R(\omega)$$

where $P(\omega)$ is a polynomial matrix of degree r and $R(\omega)$ is a rational polynomial matrix with entries being proper rational polynomial.

- If $\omega_0 \in \mathbb{C}$ and nonzero vector $\mathbf{x}$ satisfy

$$\det(F(\omega_0)) = 0, \quad F(\omega_0)\mathbf{x} = 0$$

then $(\omega_0, \mathbf{x})$ is called an eigenpair of $F(\omega)$

- $F(\omega)$ has an eigenvalue at infinity with eigenvector $\mathbf{x}$ if

$$\lim_{\omega\to\infty} \det(\omega^{-r} F(\omega)) = 0, \quad \lim_{\omega\to\infty} (\omega^{-r} F(\omega))\mathbf{x} = 0$$

$$\tilde{F}(\omega)\tilde{\mathbf{x}} := \left(F(\omega)\prod_{j=1}^{\ell}\left(I - \frac{\omega}{\omega-\mu_j} X_j X_j^* \right)\right)\tilde{\mathbf{x}}$$

- Theorem

$$\begin{aligned} &\left\{\omega \mid \tilde{F}(\omega)\tilde{\mathbf{x}} = 0, \tilde{\mathbf{x}} \neq 0\right\} \\ = &\left\{\omega \mid F(\omega)\mathbf{x} = 0, \mathbf{x} \neq 0\right\} \setminus \left\{\mu_1,\cdots,\mu_1,\cdots,\mu_\ell,\cdots,\mu_\ell\right\} \cup \left\{\infty\right\} \end{aligned}$$

Furthermore, if $(\mu, \tilde{\mathbf{x}})$ is an eigenpair of $\tilde{F}(\omega)$, then $(\mu, \mathbf{x})$ is an eigenpair of $F(\omega)$ with

$$\mathbf{x} = \prod_{j=1}^{\ell}\left(I - \frac{\mu}{\mu-\mu_j} X_j X_j^* \right)\tilde{\mathbf{x}}$$

- Remark: The orthonormal matrix X can be constructed by the convergent eigenvectors with using re-orthogonalization

Non-equivalence deflated algorithm

$(\mu_1, \mathbf{x}_1), \dots, (\mu_\ell, \mathbf{x}_\ell)$

```
Set $X = [\,]$ and $\widetilde{B}(\omega) = \omega B(\omega)$.
for $d = 1, \ldots, \ell$ do
    Compute the desired eigenvalue/eigenvector pair $(\mu_d, \mathbf{x}_d)$ of
    $A\mathbf{x} = \omega \widetilde{B}(\omega)\mathbf{x}$;
    % Retrieve the eigenvector of $Ax = \omega^2 B(\omega)x$
    for $i = 1, \ldots, d-1$ do
        Compute $\mathbf{x}_d = \left(I - \frac{\mu_d}{\mu_d - \mu_i}\tilde{\mathbf{x}}_i\tilde{\mathbf{x}}_i^*\right)\mathbf{x}_d$;
    end for
    % Compute the orthonormal matrix $X$ from the
    convergent eigenvectors
    Set $\tilde{\mathbf{x}}_d = \mathbf{x}_d$; Orthogonalize $\tilde{\mathbf{x}}_d$ against $X$ and normalize $\tilde{\mathbf{x}}_d$;
    Expand $X = [X, \tilde{\mathbf{x}}_d]$;
    % Create the coefficient matrix of the new deflated
    nonlinear eigenvalue problem
    Set
        $\widetilde{B}(\omega) = \omega B(\omega) + (A - \omega^2 B(\omega))XD(\omega)X^*,$
    where $D(\omega) = \mathrm{diag}\left((\omega - \mu_1)^{-1}, \cdots, (\omega - \mu_d)^{-1}\right)$;
end for
```

Non-equivalence deflated algorithm

$(\mu_1, \mathbf{x}_1), \ldots, (\mu_\ell, \mathbf{x}_\ell)$

Newton-type method

```
Set $X = [\,]$ and $\widetilde{B}(\omega) = \omega B(\omega)$.
for $d = 1, \ldots, \ell$ do
    Compute the desired eigenvalue/eigenvector pair $(\mu_d, \mathbf{x}_d)$ of $A\mathbf{x} = \omega \widetilde{B}(\omega)\mathbf{x}$;
    % Retrieve the eigenvector of $Ax = \omega^2 B(\omega) x$
    for $i = 1, \ldots, d-1$ do
        Compute $\mathbf{x}_d = \left( I - \frac{\mu_d}{\mu_d - \mu_i} \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^* \right) \mathbf{x}_d$;
    end for
    % Compute the orthonormal matrix $X$ from the convergent eigenvectors
    Set $\tilde{\mathbf{x}}_d = \mathbf{x}_d$; Orthogonalize $\tilde{\mathbf{x}}_d$ against $X$ and normalize $\tilde{\mathbf{x}}_d$;
    Expand $X = [X, \tilde{\mathbf{x}}_d]$;
    % Create the coefficient matrix of the new deflated nonlinear eigenvalue problem
    Set
        $$\widetilde{B}(\omega) = \omega B(\omega) + (A - \omega^2 B(\omega)) X D(\omega) X^*,$$
    where $D(\omega) = \mathrm{diag}\left( (\omega - \mu_1)^{-1}, \cdots, (\omega - \mu_d)^{-1} \right)$;
end for
```

Jacobi-Davidson method for solving

$$K(\omega_k)\mathbf{u} = \lambda\mathbf{u}$$

- Rewrite

$$A\mathbf{x} = \omega \tilde{B}(\omega)\mathbf{x} \quad \Rightarrow \quad \omega^{-1} A\mathbf{x} = \tilde{B}(\omega)\mathbf{x}$$

- For a given ω_k, consider GEP

$$\boxed{\beta(\omega_k) A\mathbf{x} = \tilde{B}(\omega_k)\mathbf{x}}$$

- To find an eigenvalue ω_* of $A\mathbf{x} = \omega \tilde{B}(\omega)\mathbf{x}$ is equivalent to determine a root of the nonlinear equation

$$\beta(\omega) - \omega^{-1} = 0$$

- Newton's method

$$\boxed{\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1} \left(\beta(\omega_k) - \omega_k^{-1}\right)}$$

- Rewrite

$$A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x} \quad \Rightarrow \quad \omega^{-1}A\mathbf{x} = \tilde{B}(\omega)\mathbf{x}$$

- For a given ω_k, consider GEP

$$\boxed{\beta(\omega_k)A\mathbf{x} = \tilde{B}(\omega_k)\mathbf{x}}$$

- To find an eigenvalue ω_* of $A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$ is equivalent to determine a root of the nonlinear equation

$$\beta(\omega) - \omega^{-1} = 0$$

- Newton's method

Need $\beta(\omega_k)$ *and* $\beta'(\omega_k)$

$$\boxed{\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta(\omega_k) - \omega_k^{-1}\right)}$$

Newton-type method for $A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$

```
Set $k = 0$.
repeat
    Compute the eigenvalue $\beta_k^{-1}$ with the smallest positive real part and
    the associated eigenvector $\mathbf{u}_k$ of
        $\boxed{\beta^{-1}\mathbf{u} = K(\omega_k)\mathbf{u} \equiv (\Lambda^{1/2}Q^*\widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2})\mathbf{u}}$    (1)
    Compute the left eigenvector $\mathbf{v}_k$ of (1) corresponding to $\beta_k$;
    Compute $\beta'(\omega_k)$ by
        $\boxed{\beta'(\omega_k) = \beta_k^2\mathbf{v}_k^*\Lambda^{1/2}Q^*\widetilde{B}(\omega_k)^{-1}\widetilde{B}(\omega_k)'\widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}\mathbf{u}_k;}$
    Compute $\omega_{k+1}$ by
        $\boxed{\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta_k - \omega_k^{-1}\right);}$
    Set $k = k + 1$;
until $|\omega_k - \omega_{k-1}| < tol$.
Set $\mu_d = \omega_k$;
Compute the eigenvector $\mathbf{x}_d = \widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}\mathbf{u}_k$.
```

Newton-type method for $A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$

Solve it by Jacobi-Davidson

```
Set $k = 0$.
repeat
    Compute the eigenvalue $\beta_k^{-1}$ with the smallest positive real part and the associated eigenvector $\mathbf{u}_k$ of
        $\boxed{\beta^{-1}\mathbf{u} = K(\omega_k)\mathbf{u} \equiv (\Lambda^{1/2}Q^*\widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2})\mathbf{u}}$    (1)
    Compute the left eigenvector $\mathbf{v}_k$ of (1) corresponding to $\beta_k$;
    Compute $\beta'(\omega_k)$ by
        $\boxed{\beta'(\omega_k) = \beta_k^2\mathbf{v}_k^*\Lambda^{1/2}Q^*\widetilde{B}(\omega_k)^{-1}\widetilde{B}(\omega_k)'\widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}\mathbf{u}_k;}$
    Compute $\omega_{k+1}$ by
        $\boxed{\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta_k - \omega_k^{-1}\right);}$
    Set $k = k + 1$;
until $|\omega_k - \omega_{k-1}| < tol$.
Set $\mu_d = \omega_k$;
Compute the eigenvector $\mathbf{x}_d = \widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}\mathbf{u}_k$.
```

Dispersive Maxwell equations

Nonlinear eigenvalue problems

Newton-type Methods for Solving

$$A\mathbf{x} = \omega\tilde{B}(\omega)\mathbf{x}$$

Numerical results

Computing derivative

$$\boxed{\beta^{'}(\omega)}$$

Nonlinear Arnoldi method (NAr)

- For a given search subspace V, let $(\tilde{\omega}, \tilde{\mathbf{z}})$ be an eigenpair of

  $$V^*(A-\omega^2 B(\omega))V\mathbf{z} = 0$$

  and let $\tilde{\mathbf{x}} = V\tilde{\mathbf{z}}$ be the associated Ritz vector

- The new search direction v is chosen as

  $$\mathbf{v} = \left(A-\sigma^2 B(\sigma)\right)^{-1}\left[(A-\tilde{\omega}^2 B(\tilde{\omega}))\tilde{\mathbf{x}}\right] \equiv \left(A-\sigma^2 B(\sigma)\right)^{-1}\mathbf{r}$$

  where σ is a given shift value

- After re-orthogonalizing v against V, the vector is appended to V and one repeats this process until $(\tilde{\omega}, \tilde{\mathbf{x}})$ converges to the desired eigenpair.

Preconditioner of Solving Linear Systems

- Solve linear system

$$\left(A-\sigma^2 B(\sigma)\right)\mathbf{v}=\mathbf{r}$$

- Since $B(\sigma)$ is diagonal, we employ a preconditioner

$$M=A-\sigma^2\alpha_\sigma I\equiv C^*C-\tau I$$

where α_σ is the average of the diagonal elements of $B(\sigma)$

- Apply the left-preconditioning M^{-1} to equation and obtain the system

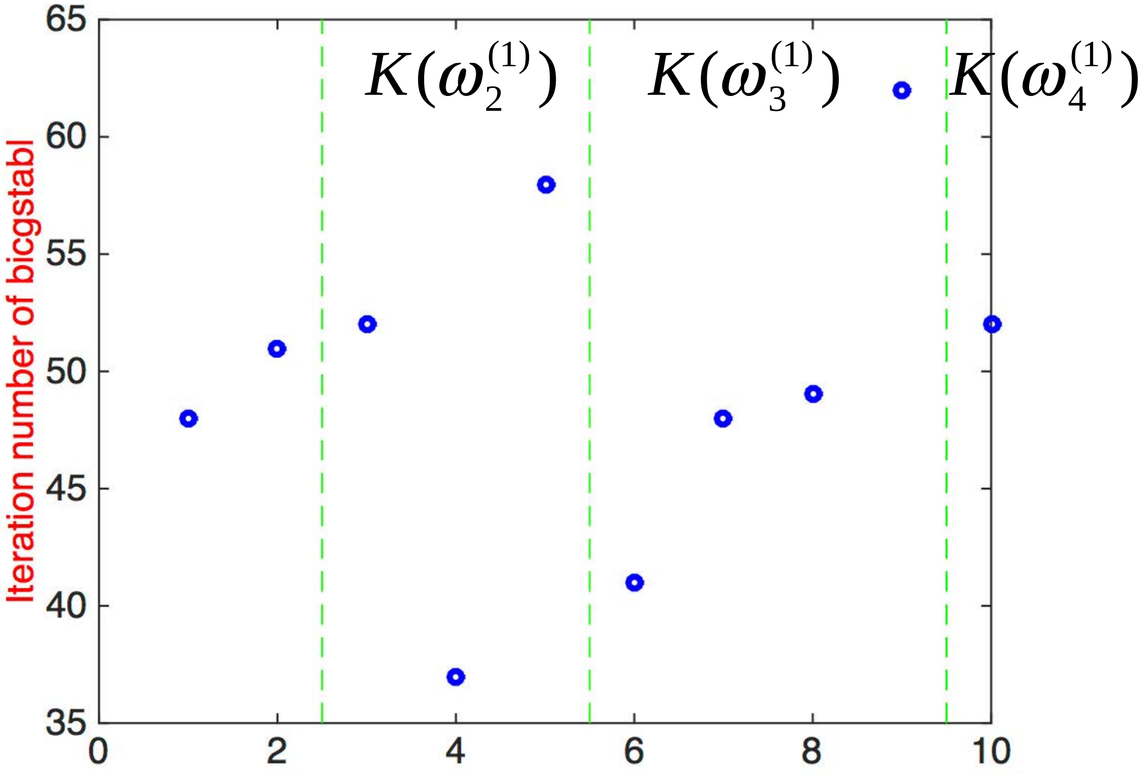
$$\left[I+\sigma^2 M^{-1}\left(\alpha_\sigma I-B(\sigma)\right)\right]\mathbf{v}=M^{-1}\mathbf{r}$$

- No need to compute a matrix-vector multiplication with A

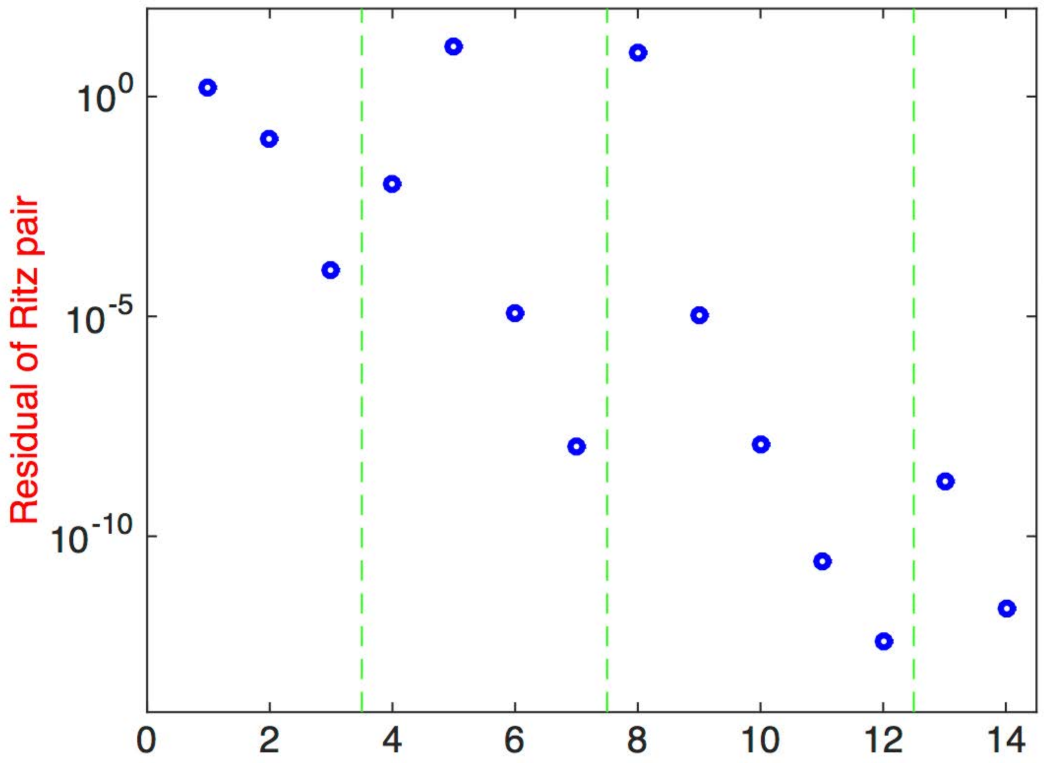
Well-separated eigenvalues

1st, 2nd eigenvalues for Drude model

$K(\omega_1^{(1)})\mathbf{u} = \lambda \mathbf{u}$

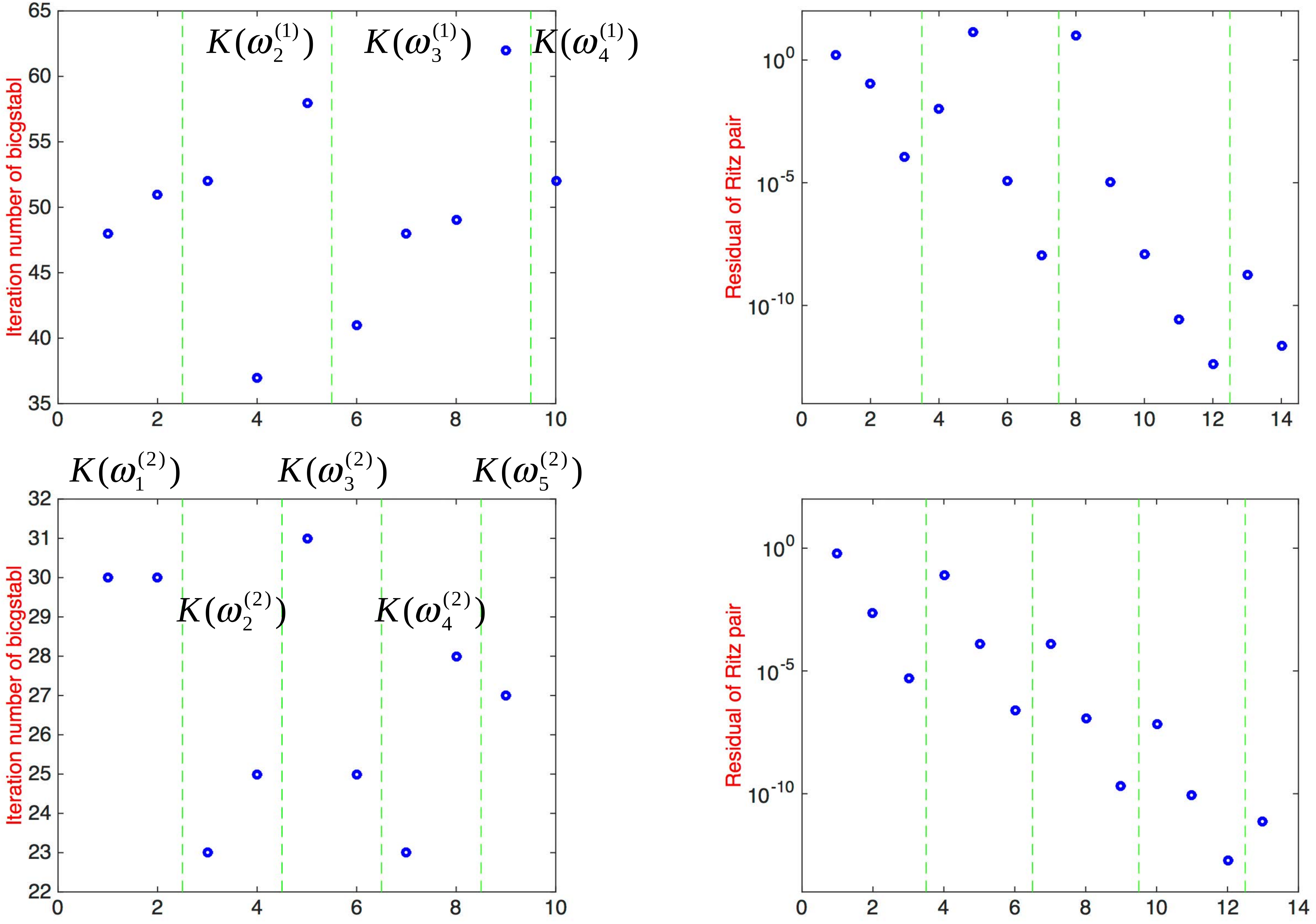


$K(\omega_2^{(1)})$ $K(\omega_3^{(1)})$ $K(\omega_4^{(1)})$



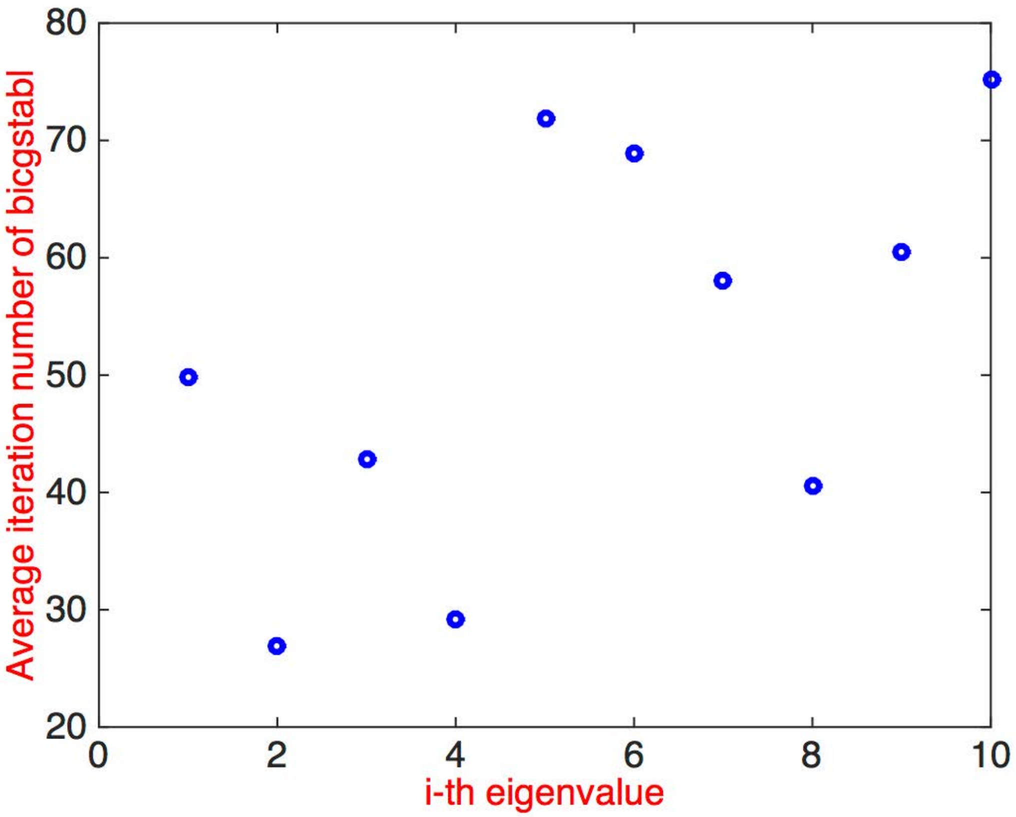
1st, 2nd eigenvalues for Drude model

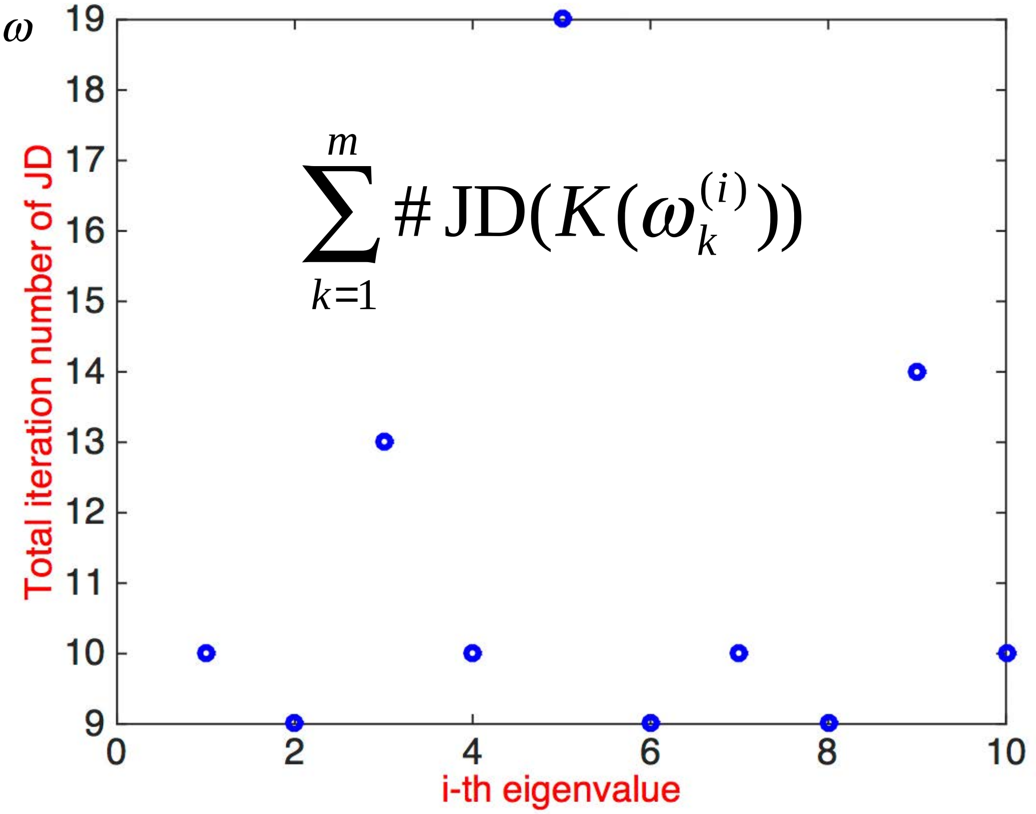
$K(\omega_1^{(1)})\mathbf{u} = \lambda \mathbf{u}$



Average iterations of bicgstabl and total iteration of JD

$$\varepsilon(\omega) = 1 - \frac{\omega_p^2}{\omega^2 + \imath\Gamma_p\omega}$$

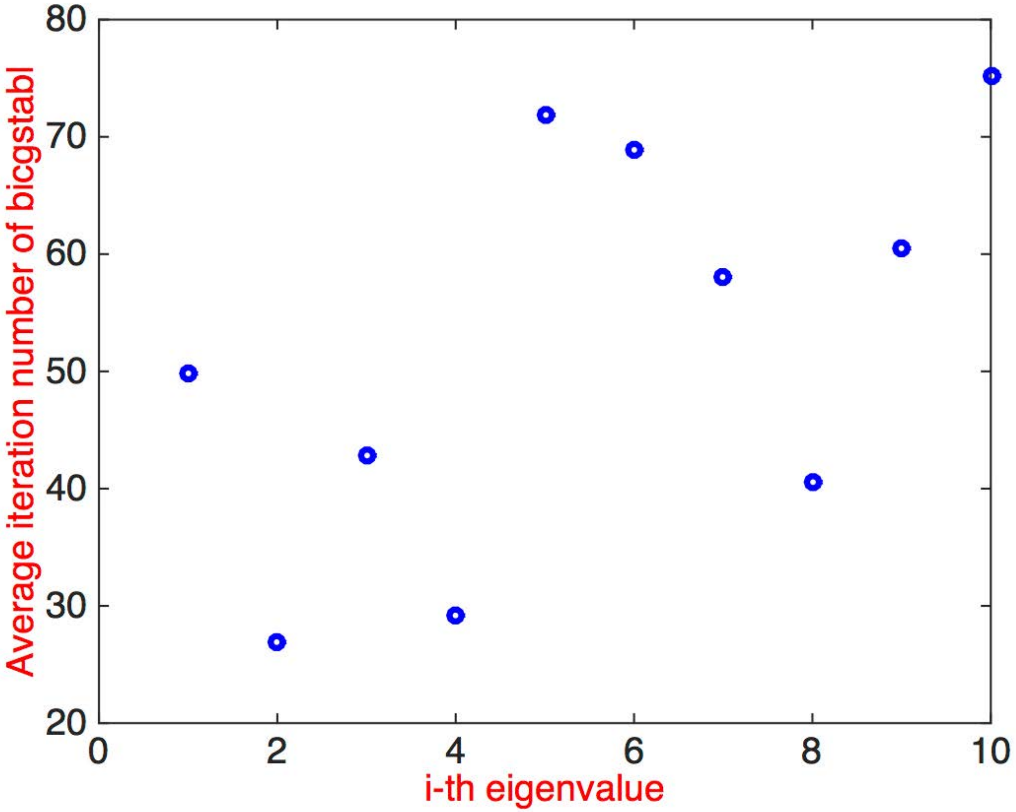


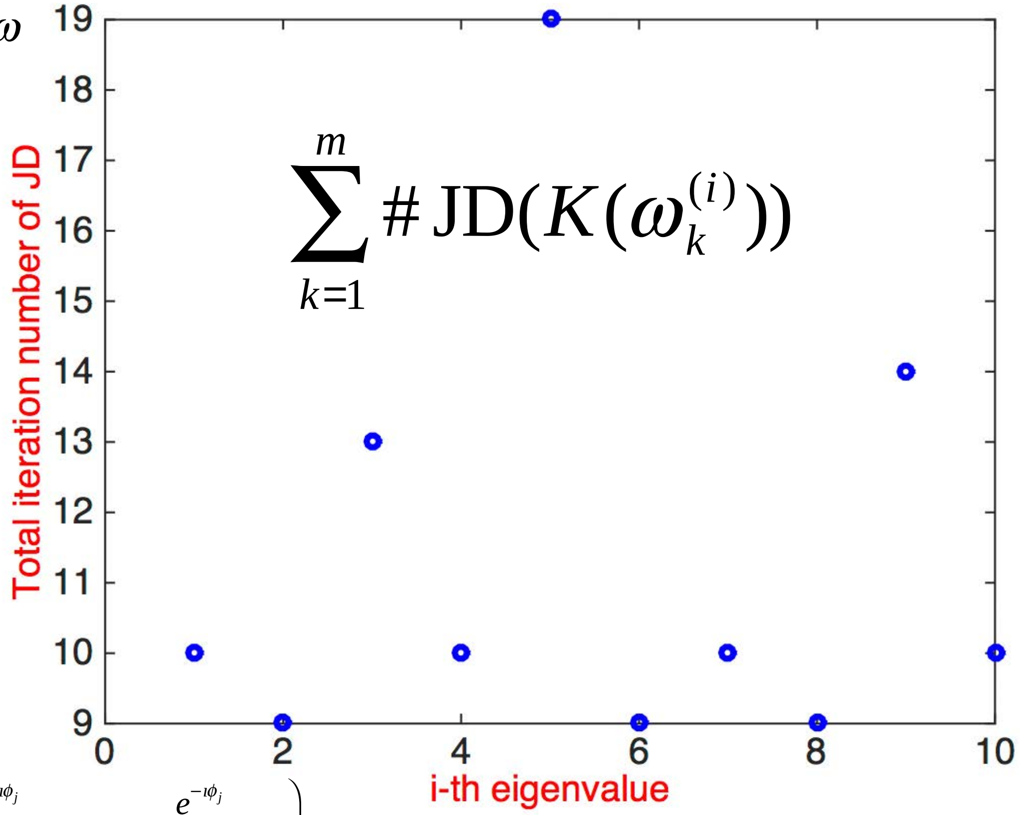


$$\sum_{k=1}^{m} \# \, \mathrm{JD}(K(\omega_k^{(i)}))$$

Average iterations of bicgstabl and total iteration of JD

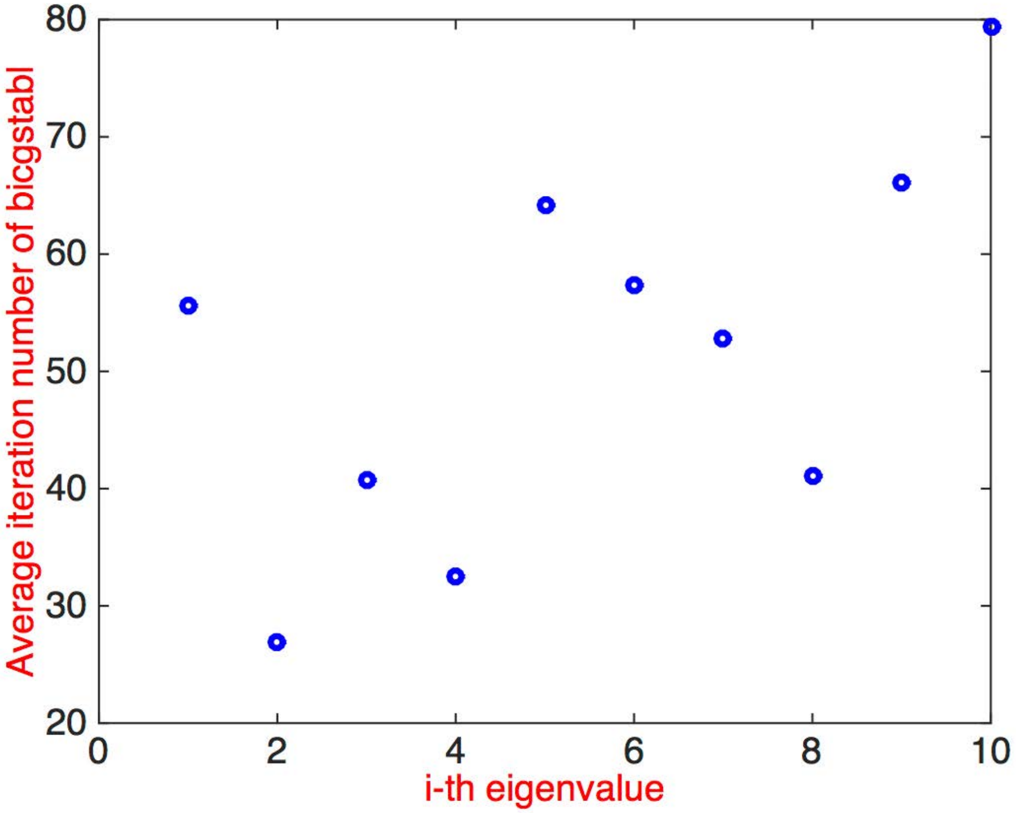
$$\varepsilon(\omega)=1-\frac{\omega_p^2}{\omega^2+\imath\Gamma_p\omega}$$

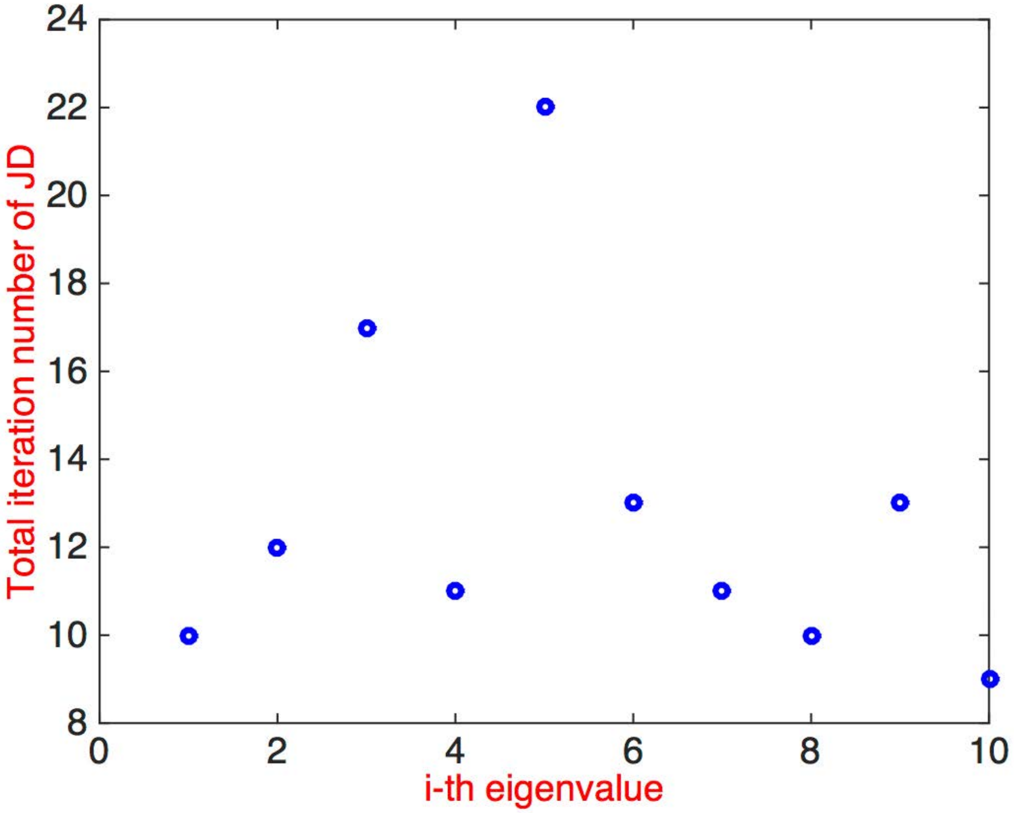




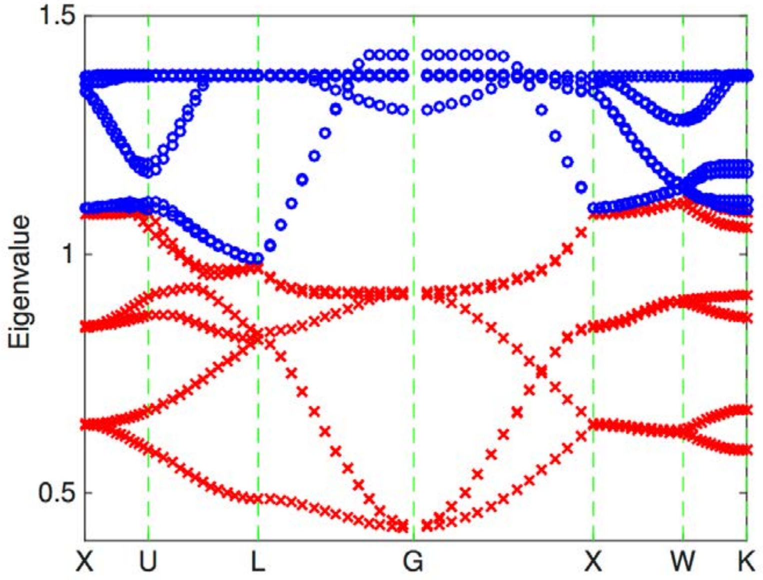
$$\sum_{k=1}^{m}\#\,\mathrm{JD}(K(\omega_k^{(i)}))$$

$$\varepsilon(\omega)=\varepsilon_\infty-\frac{\omega_p^2}{\omega^2+\imath\Gamma_p\omega}+\sum_{j=1}^{2}\Omega_j A_j\left(\frac{e^{\imath\phi_j}}{\Omega_j-\omega-\imath\Gamma_j}+\frac{e^{-\imath\phi_j}}{\Omega_j+\omega+\imath\Gamma_j}\right)$$





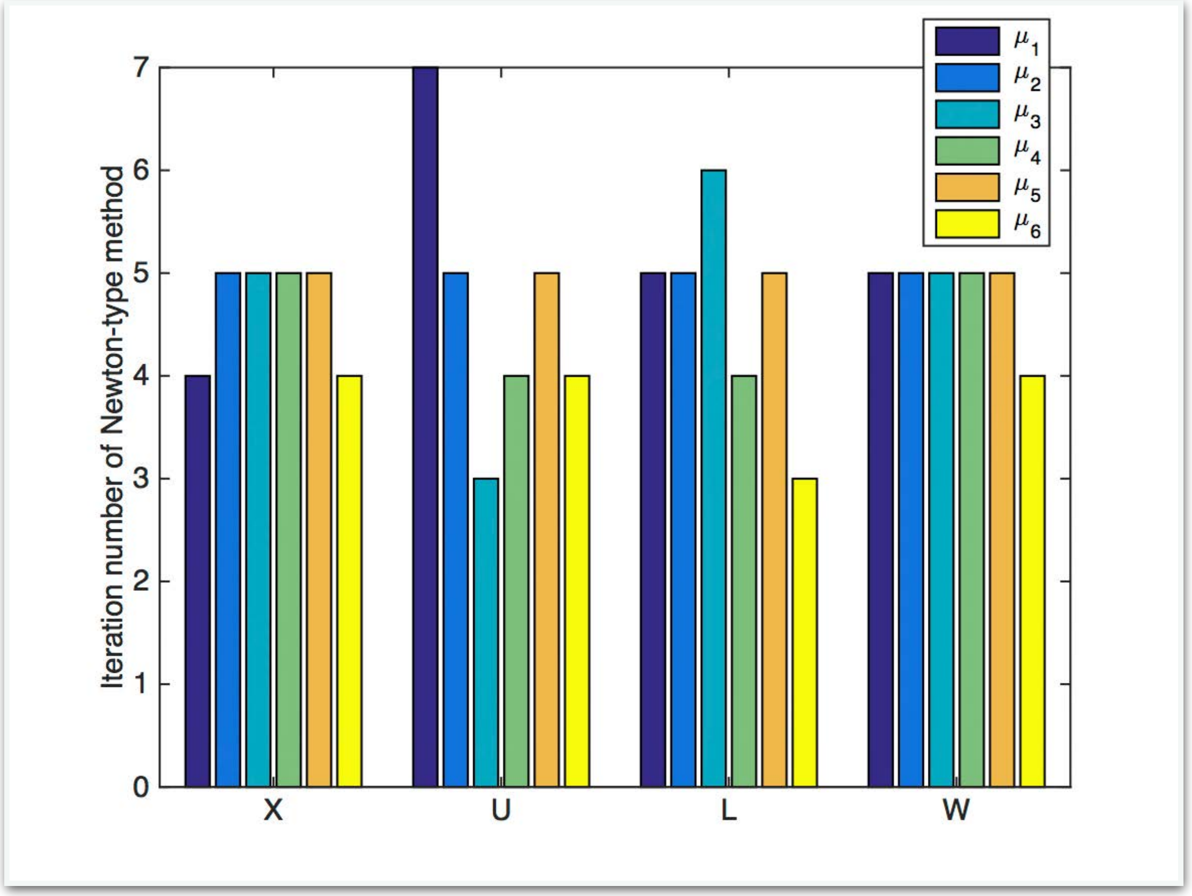
Convergence of Newton-type method



The six smallest real part nonzero eigenvalues are denoted by (red) x

$$\varepsilon(\omega)=1-\frac{\omega_p^2}{\omega^2+\imath\Gamma_p\omega}$$

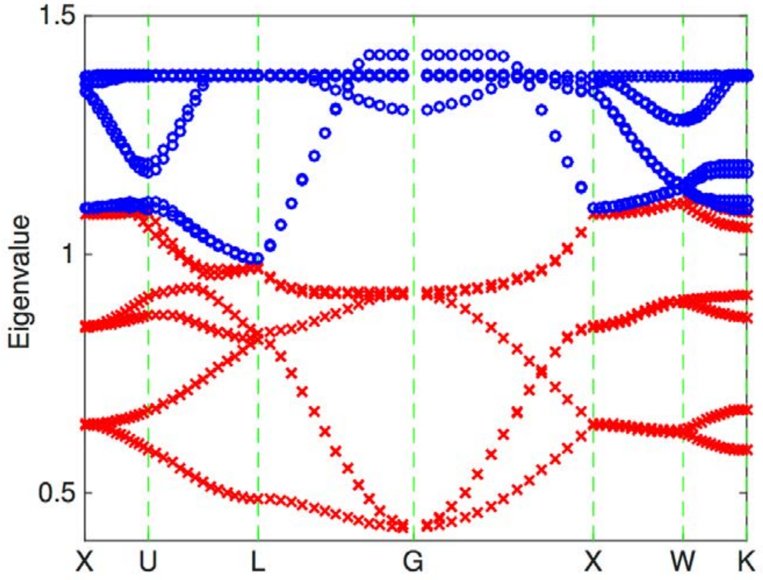
$$K(\omega_k^{(d)})\mathbf{u}=\lambda\mathbf{u},\ \text{ for } k=1,\ldots,m$$



	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6
X	4	5	5	5	5	4
U	7	5	3	4	5	4
L	5	5	6	4	5	3
W	5	5	5	5	5	4

- Only 3 to 7 iterations are needed for computing each eigenvalue

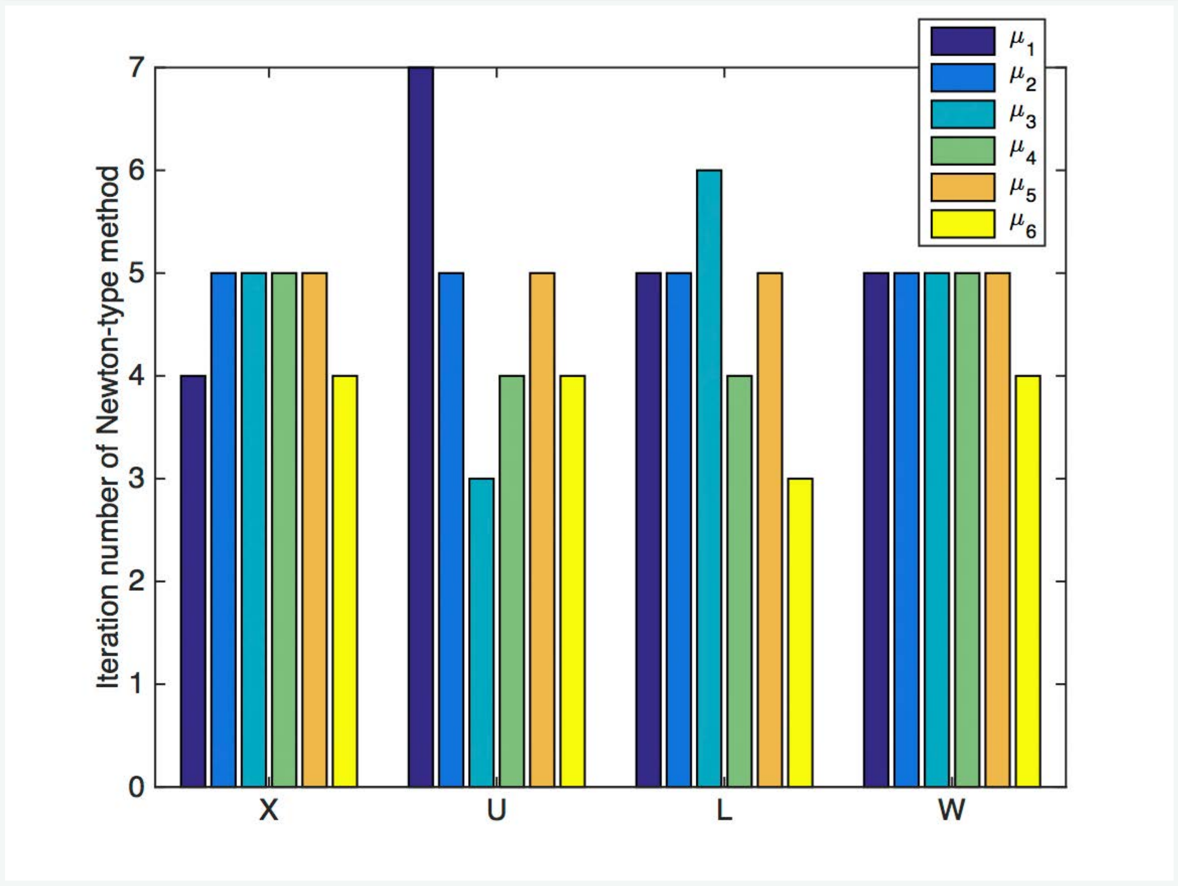
Convergence of Newton-type method

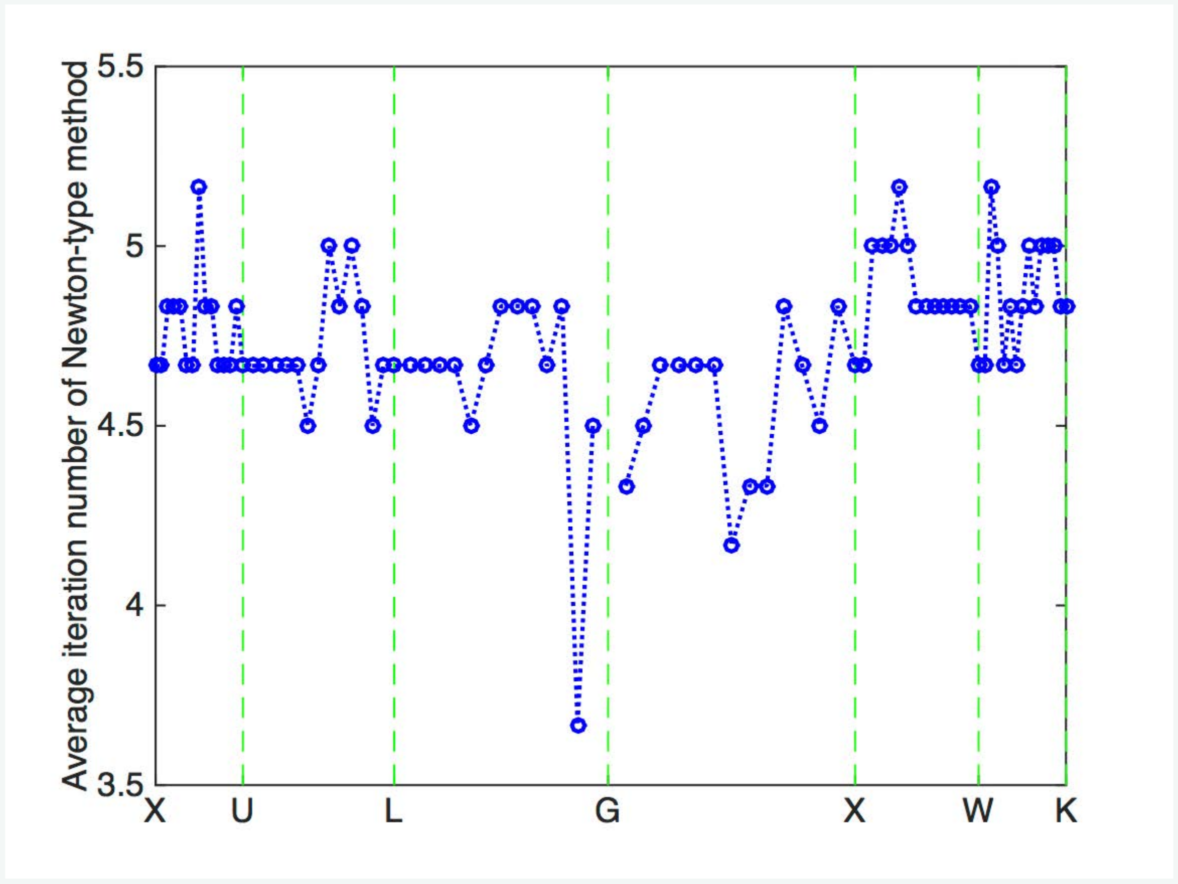


The six smallest real part nonzero eigenvalues are denoted by (red) x

$$\varepsilon(\omega) = 1 - \frac{\omega_p^2}{\omega^2 + \imath\Gamma_p\omega}$$

$$K(\omega_k^{(d)})\mathbf{u} = \lambda\mathbf{u}, \text{ for } k = 1, \ldots, m$$

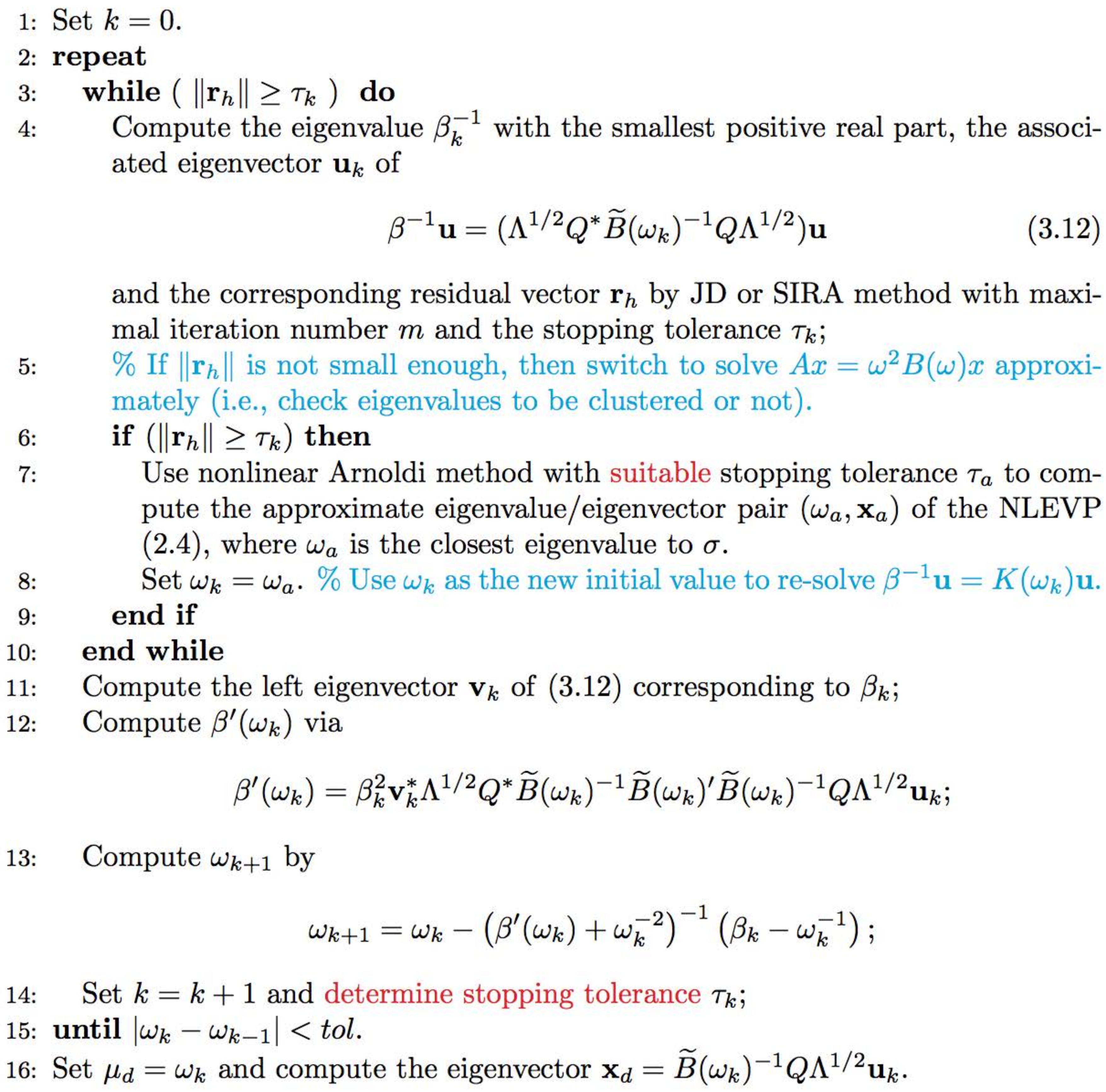




- Only 3 to 7 iterations are needed for computing each eigenvalue
- The average ranges from 3.6 to 5.2 for all benchmark problems.
- Quadratic convergence of Newton-type method

Nonlinear Arnoldi method

Alternative Newton-type method



```
Set $k = 0$.
repeat
  while ( $\|\mathbf{r}_h\| \geq \tau_k$ ) do
    Compute the eigenvalue $\beta_k^{-1}$ with the smallest positive real part, the associated eigenvector $\mathbf{u}_k$ of
        $$\beta^{-1}\mathbf{u} = (\Lambda^{1/2}Q^*\widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2})\mathbf{u}$$   (3.12)
    and the corresponding residual vector $\mathbf{r}_h$ by JD or SIRA method with maximal iteration number $m$ and the stopping tolerance $\tau_k$;
    % If $\|\mathbf{r}_h\|$ is not small enough, then switch to solve $Ax = \omega^2 B(\omega)x$ approximately (i.e., check eigenvalues to be clustered or not).
    if ($\|\mathbf{r}_h\| \geq \tau_k$) then
      Use nonlinear Arnoldi method with suitable stopping tolerance $\tau_a$ to compute the approximate eigenvalue/eigenvector pair $(\omega_a, \mathbf{x}_a)$ of the NLEVP (2.4), where $\omega_a$ is the closest eigenvalue to $\sigma$.
      Set $\omega_k = \omega_a$. % Use $\omega_k$ as the new initial value to re-solve $\beta^{-1}\mathbf{u} = K(\omega_k)\mathbf{u}$.
    end if
  end while
  Compute the left eigenvector $\mathbf{v}_k$ of (3.12) corresponding to $\beta_k$;
  Compute $\beta'(\omega_k)$ via
        $$\beta'(\omega_k) = \beta_k^2\mathbf{v}_k^*\Lambda^{1/2}Q^*\widetilde{B}(\omega_k)^{-1}\widetilde{B}(\omega_k)'\widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}\mathbf{u}_k;$$
  Compute $\omega_{k+1}$ by
        $$\omega_{k+1} = \omega_k - \left(\beta'(\omega_k) + \omega_k^{-2}\right)^{-1}\left(\beta_k - \omega_k^{-1}\right);$$
  Set $k = k + 1$ and determine stopping tolerance $\tau_k$;
until $|\omega_k - \omega_{k-1}| < tol$.
Set $\mu_d = \omega_k$ and compute the eigenvector $\mathbf{x}_d = \widetilde{B}(\omega_k)^{-1}Q\Lambda^{1/2}\mathbf{u}_k$.
```

Summary of JD and preconditioner

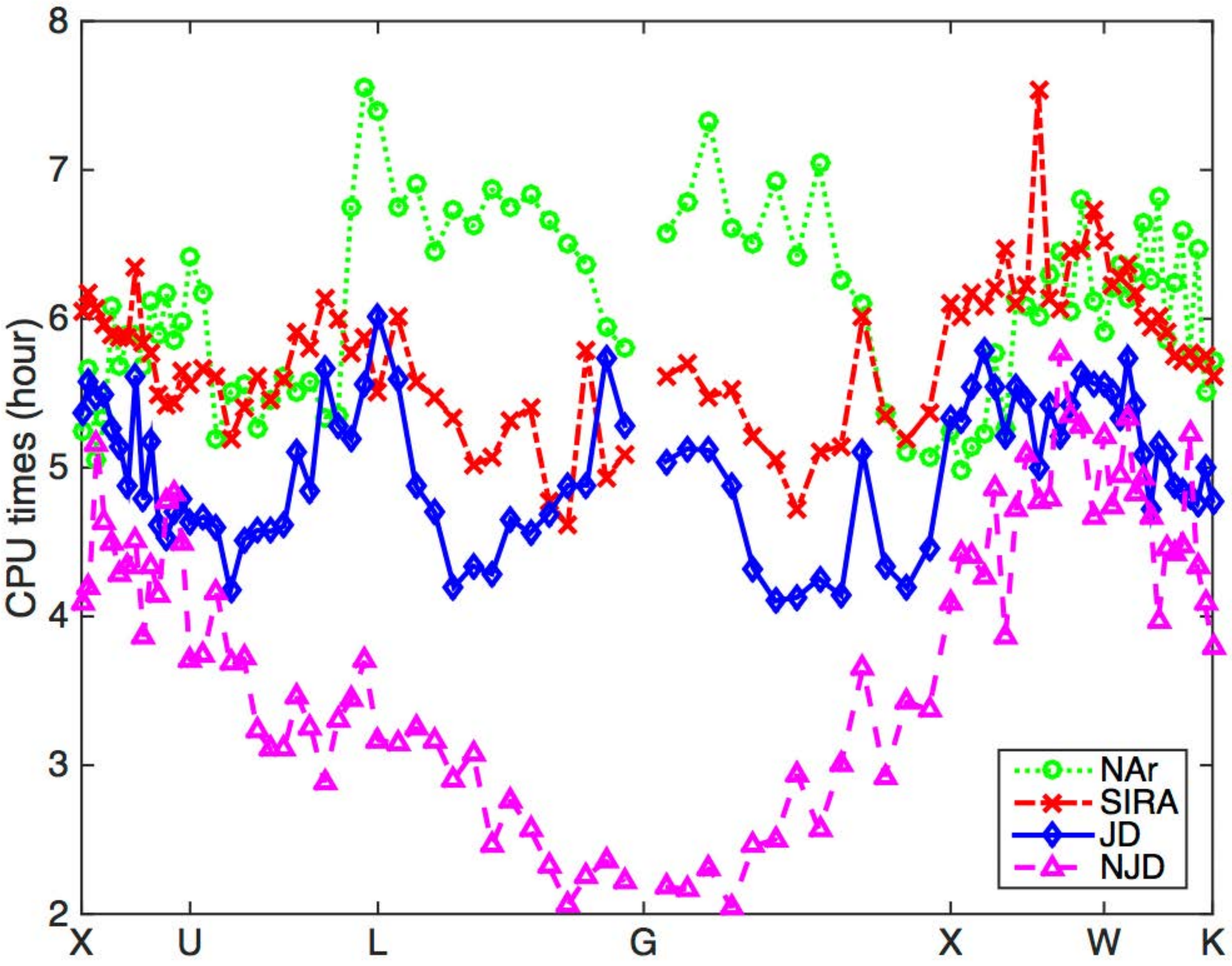
$$M_K^{-1}=\Omega_k^{-1}\left\{I+U(\omega_k)\left(\Psi(\omega_k)-V(\omega_k)^*\Omega_k^{-1}U(\omega_k)\right)^{-1}V(\omega_k)^*\Omega_k^{-1}\right\}$$

- M_K is an efficient preconditioner for solving the correction equation

$$\left(I-\mathbf{u}\mathbf{u}^*\right)\left(K(\omega_k^{(d)})-\theta I\right)\left(I-\mathbf{u}\mathbf{u}^*\right)\mathbf{t}=-\mathbf{r},\quad \mathbf{t}\perp\mathbf{u}$$

- Since the accuracy of solving correction Eq. can achieve to 1.0e-3, only few iterations of JD are needed to solve

$$K(\omega_k^{(d)})\mathbf{u}\equiv\left(\Lambda^{1/2}Q^*\tilde{B}(\omega_k^{(d)})^{-1}Q\Lambda^{1/2}\right)\mathbf{u}=\lambda\mathbf{u}$$


8
7
6
5
4
3
2
CPU times (hour)
X
U
L
G
X
W
K
NAr
SIRA
JD
NJD

Perodic Lattice