

計算物質科学における時空間
アップスケーリングと数理手法
平成28年11月28日, 電気通信大学

疎行列用の線形計算アルゴリズムの 概観と展望

曾我部 知広

名古屋大学大学院工学研究科
計算理工学専攻



NAGOYA
UNIVERSITY

内容

1. 準備

- ・ クリロフ部分空間
- ・ Arnoldi 原理と Lanczos 原理

2. 線形方程式（疎行列）の数値解法

- ・ クリロフ部分空間法
- ・ 計算物質科学への応用（シフト線形方程式）

3. 固有値問題（疎行列）の数値解法

- ・ Lanczos法とArnoldi法
- ・ Jacobi-Davidson法
- ・ LOBPCG法
- ・ Sakurai-Sugiura法

展望（計算物質科学と数理の融合にむけて）

クリロフ部分空間 (Krylov subspace)

$$K_n(A, \mathbf{v}) := \text{span}\{\mathbf{v}, A\mathbf{v}, \dots, A^{n-1}\mathbf{v}\} \quad (A \in \mathbf{C}^{N \times N}, \mathbf{v} \in \mathbf{C}^N)$$
$$= \{c_0\mathbf{v} + c_1A\mathbf{v} + \dots + c_{n-1}A^{n-1}\mathbf{v} \mid c_0, c_1, \dots, c_{n-1} \in \mathbf{C}\}$$



Aleksey Nikolaevich Krylov (1863-1945)

ロシアの（応用）数学者

1892年ベテルスブルグ海洋アカデミー教授
船の復元力の基礎研究で知られる。

娘Annaの夫はPyotr Kapitsa（ロシアの物理学者）

Pyotr Kapitsa (1894-1984)

- 1937年：ヘリウム4の超流動現象の発見
- 1978年：ノーベル物理学賞
(1927年にAnnaとKapitsaは結婚)

[https://en.wikipedia.org
/wiki/Aleksey_Krylov](https://en.wikipedia.org/wiki/Aleksey_Krylov)

Arnoldi 原理 [W. E. Arnoldi, 1951] (米国の技術者)

クリロフ部分空間の正規直交基底の生成法

入力 : $A \in \mathbf{C}^{N \times N}$, $\mathbf{v} \in \mathbf{C}^N$, $n \in N$

出力 : $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

ただし $\left\{ \begin{array}{l} \textcircled{1} \text{span}\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\} = K_n(A, \mathbf{v}) \\ \textcircled{2} \underbrace{(\mathbf{v}_i, \mathbf{v}_j)}_{\text{エルミート内積}} = \delta_{i,j} = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases} \end{array} \right.$

エルミート内積

$$(\mathbf{v}_i, \mathbf{v}_j) := \mathbf{v}_i^H \mathbf{v}_j (= \bar{\mathbf{v}}_i^T \mathbf{v}_j)$$

Arnoldi 原理 (Arnoldi process)

$$\mathbf{v}_1 = \mathbf{v} / \|\mathbf{v}\|_2 \quad \left[\|\mathbf{v}\|_2 := \sqrt{(\mathbf{v}, \mathbf{v})} \right]$$

for $k = 1, 2, \dots, n-1$

$$h_{i,k} = (\mathbf{v}_i, A\mathbf{v}_k) \quad (i = 1, 2, \dots, k)$$

$$\tilde{\mathbf{v}}_{k+1} = A\mathbf{v}_k - \sum_{i=1}^k h_{i,k} \mathbf{v}_i$$

$$h_{k+1,k} = \|\tilde{\mathbf{v}}_{k+1}\|_2$$

$$\text{end} \quad \mathbf{v}_{k+1} = \tilde{\mathbf{v}}_{k+1} / h_{k+1,k}$$

【ノート】 このアルゴリズムは Classical Gram-Schmidt 型です .

- (その他) • Modified Gram-Schmidt 型
- Householder 型 [H.F. Walker, 1988]

Arnoldi 原理の k 反復後の行列表示

$$A[\underbrace{v_1, v_2, \dots, v_k}_{V_k}] = [\underbrace{v_1, v_2, \dots, v_k, v_{k+1}}_{V_{k+1}}] \underbrace{\begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,k-1} & h_{1,k} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,k-1} & h_{2,k} \\ & \ddots & \ddots & \vdots & \vdots \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & h_{k,k} \\ & & & & h_{k+1,k} \end{bmatrix}}_{H_{k+1,k}}$$

— 行列表示 —

$$AV_k = V_{k+1}H_{k+1,k}$$

$H_{k+1,k}$
(Hessenberg行列型)

Lanczos 原理 [C. Lanczos, 1950] (ハンガリーの数学者)

(A. Einsteinの助手を務めたこともある。)

クリロフ部分空間の正規直交基底の生成法

入力 : $\underline{A} \in \mathbf{C}^{N \times N}$, $\mathbf{v} \in \mathbf{C}^N$, $n \in N$
エルミート行列 ($A = A^H$)

出力 : $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n$

ただし $\left\{ \begin{array}{l} \textcircled{1} \text{span}\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\} = K_n(A, \mathbf{v}) \\ \textcircled{2} \underline{(\mathbf{v}_i, \mathbf{v}_j)} = \delta_{i,j} = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases} \end{array} \right.$

エルミート内積

$$(\mathbf{v}_i, \mathbf{v}_j) := \mathbf{v}_i^H \mathbf{v}_j (= \bar{\mathbf{v}}_i^T \mathbf{v}_j)$$

Lanczos 原理 (Lanczos process)

$$\underline{\mathbf{v}_0 = \mathbf{0}}, \quad \mathbf{v}_1 = \mathbf{v} / \|\mathbf{v}\|_2$$

for $k = 1, 2, \dots, n-1$

$$h_{i,k} = (\mathbf{v}_i, A\mathbf{v}_k) \quad (\underline{i = k-1, k}) \quad (\text{ ~~$i = 1, 2, \dots, k$~~ })$$

$$\tilde{\mathbf{v}}_{k+1} = A\mathbf{v}_k - \underline{h_{k,k}\mathbf{v}_k - h_{k-1,k}\mathbf{v}_{k-1}} \quad = \sum_{i=1}^k \text{ ~~$h_{i,k}\mathbf{v}_i$~~ }$$

$$h_{k+1,k} = \|\tilde{\mathbf{v}}_{k+1}\|_2$$

$$\mathbf{v}_{k+1} = \tilde{\mathbf{v}}_{k+1} / h_{k+1,k}$$

end

- 行列 A のエルミート性から $h_{k-1,k} = h_{k,k-1}$ が成り立つ.
- $\alpha_k := h_{k,k}$, $\beta_{k+1} := h_{k+1,k}$ と書き換える.

Lanczos 原理 (Lanczos process)

$$\mathbf{v}_0 = \mathbf{0}, \beta_1 = 0, \mathbf{v}_1 = \mathbf{v} / \|\mathbf{v}\|_2$$

for $k = 1, 2, \dots, n-1$

$$\alpha_k = (\mathbf{v}_k, A\mathbf{v}_k)$$

$$\tilde{\mathbf{v}}_{k+1} = A\mathbf{v}_k - \alpha_k \mathbf{v}_k - \beta_k \mathbf{v}_{k-1}$$

$$\beta_{k+1} = \|\tilde{\mathbf{v}}_{k+1}\|_2$$

$$\mathbf{v}_{k+1} = \tilde{\mathbf{v}}_{k+1} / \beta_{k+1}$$

end

【ノート】

Lanczosの論文(1950)では、非エルミート行列に対してクリロフ部分空間の双直交基底を生成する一般化Lanczos原理も書かれています。

Lanczos 原理の k 反復後の行列表示

$$A[\underbrace{v_1, v_2, \dots, v_k}_{V_k}] = [\underbrace{v_1, v_2, \dots, v_k, v_{k+1}}_{V_{k+1}}]$$

行列表示

$$AV_k = V_{k+1}T_{k+1,k}$$

$$\begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_k \\ & & & \beta_k & \alpha_k \\ & & & & \beta_{k+1} \end{bmatrix}$$

$T_{k+1,k}$

(三重対角行列型)

クリロフ部分空間法

Krylov subspace methods

$$Ax = b$$

20世紀の**Top 10** アルゴリズム

[SIAM News, Vol. 33, No. 4, 2000]

- Fast Fourier Transform
- Fast multipole method
- Fortran compiler
- Householder matrix decomposition
- Integer relation detection algorithm
- **Krylov subspace methods**
- Monte Carlo method
- QR method
- Quick sort
- Simplex methods

- クリロフ部分空間

$$K_n(A, \mathbf{v}) := \text{span}\{\mathbf{v}, A\mathbf{v}, \dots, A^{n-1}\mathbf{v}\}$$
$$(A \in \mathbf{C}^{N \times N}, \mathbf{v} \in \mathbf{C}^N)$$

- クリロフ部分空間法

クリロフ部分空間の中から (効率よく)
 $A\mathbf{x} = \mathbf{b}$ の近似解 $\mathbf{x}_n$ を探す解法の総称

$\mathbf{x}_n$: n 反復目の近似解ベクトル

$\mathbf{r}_n := \mathbf{b} - A\mathbf{x}_n$, n 反復目の残差ベクトル

空間条件

$$\mathbf{x}_n \in \mathbf{x}_0 + K_n(A, \mathbf{r}_0)$$

直交条件

$$\mathbf{r}_n \perp W_n (n\text{次元部分空間})$$

クリロフ部分空間法 (直交性の観点)

$$(A = A^H)$$

CG

$$\mathbf{r}_n \perp K_n(A, \mathbf{r}_0)$$

(ランチョス-プロセス)

$$(A = A^T \neq A^H)$$

COCG

$$\mathbf{r}_n \perp \overline{K_n(A, \mathbf{r}_0)}$$

(複素対称ランチョス-プロセス)

$$(A \neq A^H)$$

BiCG

$$\mathbf{r}_n \perp K_n(A^H, \mathbf{r}_0^*)$$

(双ランチョス-プロセス)

GMRES

$$\mathbf{r}_n \perp AK_n(A, \mathbf{r}_0)$$

(アーノルディ-プロセス)

クリロフ部分空間法 (注意：一部の解法のみ紹介)

$$(A = A^H)$$

CG (共役勾配法) (h.p.d)

[Hestenes & Stiefel, 1952]

CR (共役残差法)

[Stiefel, 1955]

$$(A = A^T \neq A^H)$$

COCG

[van der Vorst & Melissen, 1990]

COCR

[S. & Zhang, 2007]

$$(A \neq A^H)$$

BiCG

(双共役勾配法)

[Fletcher, 1976]

積型

[Zhang's framework, 1997]

CGS

[Sonneveld, 1989]

GPBiCG

[Zhang, 1997]

BiCGStab

[van der Vorst, 1992]

BiCGStab(l)

[Sleijpen & Fokkema, 1993]

FOM

[Saad, 1981]

GMRES

[Saad & Schultz, 1986]

IDR(s)

(Sonneveld部分空間法)

[Sonneveld & Gijzen, 2008]

IDRStab

[Sleijpen & Gijzen, 2010]

GBiCGStab(s,L)

[Tanio & Sugihara, 2010]

The Conjugate Orthogonal CG method

$$(A = A^T \neq A^H)$$

COCG

$$\mathbf{r}_n \perp \overline{K_n(A, \mathbf{r}_0)}$$

(複素対称ランチョス-プロセス)

Conjugate Orthogonal property

(共役直交性)

$$\mathbf{r}_i \perp \bar{\mathbf{r}}_j \quad (i \neq j)$$

内積 $(\mathbf{x}, \mathbf{y}) := \mathbf{x}^H \mathbf{y}$

$$\mathbf{p}_n = \mathbf{r}_n + \beta_{n-1} \mathbf{p}_{n-1}$$

$$\alpha_n = \frac{(\bar{\mathbf{r}}_n, \mathbf{r}_n)}{(\bar{\mathbf{p}}_n, A\mathbf{p}_n)}$$

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \alpha_n \mathbf{p}_n$$

$$\mathbf{r}_{n+1} = \mathbf{r}_n - \alpha_n A\mathbf{p}_n$$

$$\beta_n = \frac{(\bar{\mathbf{r}}_{n+1}, \mathbf{r}_{n+1})}{(\bar{\mathbf{r}}_n, \mathbf{r}_n)}$$

—計算物質科学への応用—

シフト線形方程式のための クリロフ部分空間法

KS methods for shifted linear systems

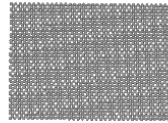
$$(A + \sigma_i I) \mathbf{x}^{(i)} = \mathbf{b} \quad \text{for } i = 1, 2, \dots, m.$$

曾我部知広・張紹良,
大規模シフト線形方程式の数値解法—クリロフ部分空間の性質に着目して—,
応用数理, 19(3), 2009, pp. 27-42.

大規模シフト線形方程式

物性物理学

大規模電子構造計算

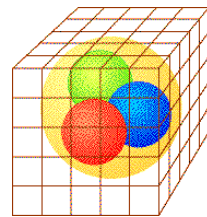


シフト線形方程式

$$(A + \sigma_i I) x^{(i)} = b$$

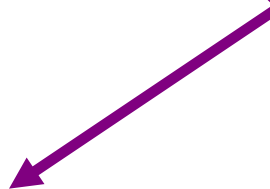
素粒子物理学

格子量子色力学



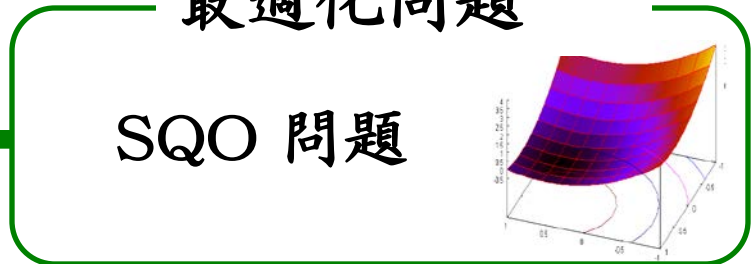
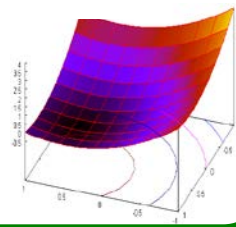
逆問題

Tikhonov正則化法



最適化問題

SQO 問題



標準固有値問題

Sakurai-Sugiura法



シフト線形方程式に対するKS法

シンプルな方法

$$(A + \sigma_1 I) \mathbf{x}^{(1)} = \mathbf{b} \longleftarrow \text{KS 法}$$

$\vdots$

$\longleftarrow \text{KS 法}$

$$(A + \sigma_m I) \mathbf{x}^{(m)} = \mathbf{b} \longleftarrow \text{KS 法}$$

効率よく解くために

シード方程式

~~クリロフ部分空間のシフト不変性~~

$$K_n(A + \sigma_i I, \mathbf{b}) = K_n(A + \sigma_j I, \mathbf{b})$$

$$(A + \sigma_1 I) \mathbf{x}^{(1)} = \mathbf{b}$$

$\vdots$

$$(A + \sigma_m I) \mathbf{x}^{(m)} = \mathbf{b}$$

クリロフ部分空間の生成

基底の再利用
行列ベクトル積や
内積計算が不要

シフト線形方程式のための**KS**法 (一部のみ)

$$(A = A^H)$$

(Shifted) CG

[Young, 1988]

$$(A = A^T \neq A^H)$$

Shifted COCG

[Takayama, Hoshi,
S., Zhang, Fujiwara 2006]

$$(A \neq A^H)$$

Shifted BiCG

[Frommer *et al.*, 1995]

Shifted BiCGStab(ℓ)

[Frommer, 2003]

Shifted IDR(s)

[Du, S., Zhang, 2015]

(Shifted) GMRES(m)

[Datta & Saad, 1991]

(Restarted) Shifted GMRES(m)

[Frommer & Grassner, 1998]

Shifted FOM

[Simoncini, 2003]

Shifted COCG 法

$$Ax = b$$

COCG法

近似解ベクトル

$$x_n \in \underline{K_n(A, b)}$$

$$(A + \sigma_i I)x^{(i)} = b$$

COCG法

近似解ベクトル

$$x_n^{(i)} \in \underline{K_n(A + \sigma_i I, b)}$$

$$K_n(A, b) = K_n(A + \sigma_i I, b)$$

クリロフ部分空間のシフト不変性

Shifted COCG 法

$$Ax = b$$

COCG

$\alpha, \beta, \mathbf{r}$

$$(A + \sigma_i I) \mathbf{x}^{(i)} = \mathbf{b}$$

$$\pi_{n+1}^{(i)} = R_{n+1}(-\sigma_i)$$

$$\alpha_n^{(i)} = (\pi_n^{(i)} / \pi_{n+1}^{(i)}) \alpha_n$$

$$\beta_{n-1}^{(i)} = (\pi_{n-1}^{(i)} / \pi_n^{(i)})^2 \beta_{n-1}$$

$$\mathbf{p}_n^{(i)} = \left(1 / \pi_n^{(i)}\right) \mathbf{r}_n + \beta_{n-1}^{(i)} \mathbf{p}_{n-1}^{(i)}$$

$$\mathbf{x}_{n+1}^{(i)} = \mathbf{x}_n^{(i)} + \alpha_n^{(i)} \mathbf{p}_n^{(i)}$$

行列・ベクトル積や
内積計算がない！

$$\pi_{n+1}^{(i)} = R_{n+1}(-\sigma_i) = (1 + \alpha_n \sigma_i) \pi_n^{(i)} + \frac{\beta_{n-1}}{\alpha_{n-1}} \alpha_n (\pi_n^{(i)} - \pi_{n-1}^{(i)})$$

—計算物質科学への応用—

一般化シフト線形方程式のための クリロフ部分空間法

KS methods for generalized shifted linear systems

$$(A + \sigma_i B) \mathbf{x}^{(i)} = \mathbf{b} \quad \text{for } i = 1, 2, \dots, m.$$

T. Sogabe, T. Hoshi, S.-L. Zhang, and T. Fujiwara,
Solution of generalized shifted linear systems with complex symmetric matrices
J. Comput. Phys., 231(2012), pp. 5669-5684.

問題点

シフト線形方程式

$$(A + \sigma_i I) \mathbf{x}^{(i)} = \mathbf{b} \quad \text{for } i = 1, 2, \dots, m.$$

シフト不変性 ○

$$K_n(A + \sigma_i I, \mathbf{b}) = K_n(A + \sigma_j I, \mathbf{b})$$

一般化シフト線形方程式

$$(A + \sigma_i B) \mathbf{x}^{(i)} = \mathbf{b} \quad \text{for } i = 1, 2, \dots, m.$$

シフト不変性 ✕

$$K_n(A + \sigma_i B, \mathbf{b}) \neq K_n(A + \sigma_j B, \mathbf{b})$$

我々のアプローチ

1. シフト線形方程式への変換

$$(A + \sigma_i B) \mathbf{x}^{(i)} = \mathbf{b} \quad \text{for } i = 1, 2, \dots, m.$$



B : 正則行列

$$(B^{-1}A + \sigma_i I) \mathbf{x}^{(i)} = B^{-1}\mathbf{b} \quad \text{for } i = 1, 2, \dots, m.$$

2. 双線形形式の利用

$$\text{双線形形式 } (\mathbf{x}, \mathbf{y})_B := \mathbf{x}^T B \mathbf{y}$$

$$(\mathbf{x}, B^{-1}A\mathbf{y})_B = (\mathbf{y}, B^{-1}A\mathbf{x})_B \rightarrow \text{Shifted COCG法の拡張}$$

Generalized Shifted COCG法

$$(B^{-1}A + \sigma_s I) \mathbf{x}^{(s)} = B^{-1} \mathbf{b}$$

$$(B^{-1}A + \sigma_i I) \mathbf{x}^{(i)} = B^{-1} \mathbf{b}$$

$$\begin{aligned} \mathbf{p}_n &= \mathbf{r}'_n + \beta_{n-1} \mathbf{p}_{n-1} \\ \alpha_n &= \frac{(\bar{\mathbf{r}}'_n, \mathbf{r}'_n)_B}{(\mathbf{p}_n, (B^{-1}A + \sigma_s I) \mathbf{p}_n)_B} \end{aligned}$$

$\alpha, \beta, \mathbf{r}$

$$\begin{aligned} \mathbf{x}_{n+1} &= \mathbf{x}_n + \alpha_n \mathbf{p}_n \\ \mathbf{r}'_{n+1} &= \mathbf{r}'_n - \alpha_n (B^{-1}A + \sigma_s I) \mathbf{p}_n \\ \beta_n &= \frac{(\mathbf{r}'_{n+1}, \mathbf{r}'_{n+1})_B}{(\mathbf{r}'_n, \mathbf{r}'_n)_B} \end{aligned}$$

$$\begin{aligned} \pi_{n+1}^\sigma &= R_{n+1}(\sigma_i - \sigma_s) \\ \alpha_n^{(i)} &= (\pi_n^{(i)} / \pi_{n+1}^{(i)}) \alpha_n \\ \beta_{n-1}^{(i)} &= (\pi_{n-1}^{(i)} / \pi_n^{(i)})^2 \beta_{n-1} \\ \mathbf{p}_n^{(i)} &= (1 / \pi_n^{(i)}) \mathbf{r}'_n + \beta_{n-1}^{(i)} \mathbf{p}_{n-1}^{(i)} \\ \mathbf{x}_{n+1}^{(i)} &= \mathbf{x}_n^{(i)} + \alpha_n^{(i)} \mathbf{p}_n^{(i)} \end{aligned}$$

$\mathbf{r}' := B^{-1} \mathbf{r}_n = B^{-1}(\mathbf{b} - (A + \sigma_s I) \mathbf{x}_n)$ の代入

Generalized Shifted COCG法

$$(B^{-1}A + \sigma_s I) \mathbf{x}^{(s)} = B^{-1} \mathbf{b}$$

$$(B^{-1}A + \sigma_i I) \mathbf{x}^{(i)} = B^{-1} \mathbf{b}$$

$$\mathbf{p}_n = B^{-1} \mathbf{r}_n + \beta_{n-1} \mathbf{p}_{n-1}$$

$$\alpha_n = \frac{(\bar{\mathbf{r}}_n, B^{-1} \mathbf{r}_n)}{(\bar{\mathbf{p}}_n, (A + \sigma_s B) \mathbf{p}_n)}$$

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \alpha_n \mathbf{p}_n$$

$$\mathbf{r}_{n+1} = \mathbf{r}_n - \alpha_n (A + \sigma_s B) \mathbf{p}_n$$

$$\beta_n = \frac{(\bar{\mathbf{r}}_n, B^{-1} \mathbf{r}_n)}{(\bar{\mathbf{r}}_{n+1}, B^{-1} \mathbf{r}_{n+1})}$$

$\alpha, \beta, \mathbf{r}$

$$\pi_{n+1}^\sigma = R_{n+1}(\sigma_i - \sigma_s)$$

$$\alpha_n^{(i)} = (\pi_n^{(i)} / \pi_{n+1}^{(i)}) \alpha_n$$

$$\beta_{n-1}^{(i)} = (\pi_{n-1}^{(i)} / \pi_n^{(i)})^2 \beta_{n-1}$$

$$\mathbf{p}_n^{(i)} = \left(1 / \pi_n^{(i)}\right) B^{-1} \mathbf{r}_n + \beta_{n-1}^{(i)} \mathbf{p}_{n-1}^{(i)}$$

$$\mathbf{x}_{n+1}^{(i)} = \mathbf{x}_n^{(i)} + \alpha_n^{(i)} \mathbf{p}_n^{(i)}$$

Generalized Shifted COCG法

$$(B^{-1}A + \sigma_s I) \mathbf{x}^{(s)} = B^{-1} \mathbf{b}$$

$$\mathbf{p}_n = B^{-1} \mathbf{r}_n + \beta_{n-1} \mathbf{p}_{n-1}$$

$$\alpha_n = \frac{(\bar{\mathbf{r}}_n, B^{-1} \mathbf{r}_n)}{(\bar{\mathbf{p}}_n, (A + \sigma_s B) \mathbf{p}_n)}$$

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \alpha_n \mathbf{p}_n$$

$$\mathbf{r}_{n+1} = \mathbf{r}_n - \alpha_n (A + \sigma_s B) \mathbf{p}_n$$

$$\beta_n = \frac{(\bar{\mathbf{r}}_{n+1}, B^{-1} \mathbf{r}_{n+1})}{(\bar{\mathbf{r}}_n, B^{-1} \mathbf{r}_n)}$$

内部反復

Solve $B\mathbf{v} = \mathbf{r}_n$
(本研究ではCG法)

数值実験

- **GSCOCG**法 : Generalized Shifted COCG法

VT64 Workstation 2100	CPU: 2.4GHz (AMD Opteron 252) Memory: 8GB
Code	Fortran 77 (double precision arithmetic)
Initial guess	all zeros
Stopping criteria	Outer iteration Rel.Res. $\leq 10^{-12}$ Inner iteration Rel.Res. $\leq 10^{-12}$

数値例 1

Au 256 atoms (FCC)

$$(A + \sigma_i B) x^{(i)} = b, \quad N: 2,304, \text{Nnz}: 1,059,584$$

$$\sigma_1 = 0.400 + 0.001i$$

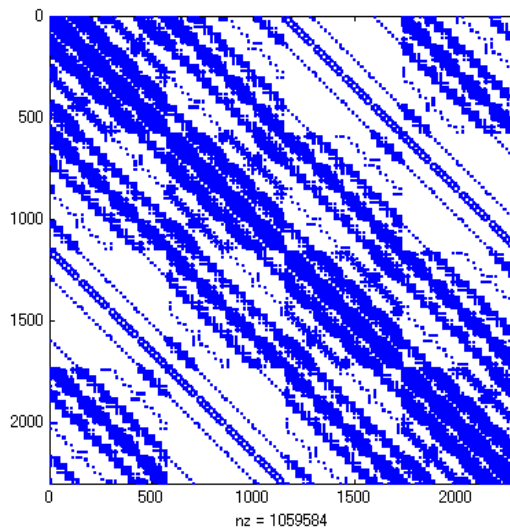
$$\sigma_2 = 0.401 + 0.001i$$

$\vdots$

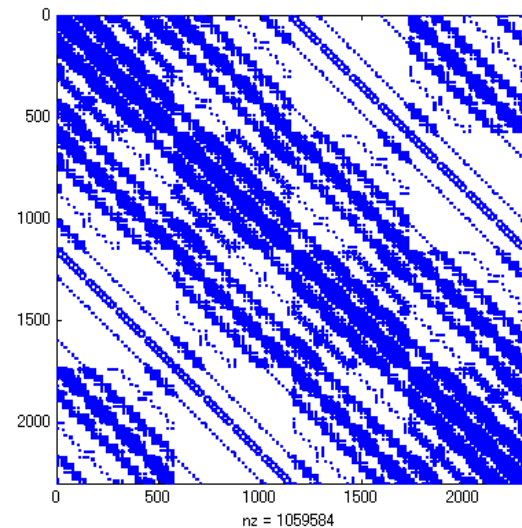
$$\sigma_{1001} = 1.400 + 0.001i$$

1001 個の方程式

A:

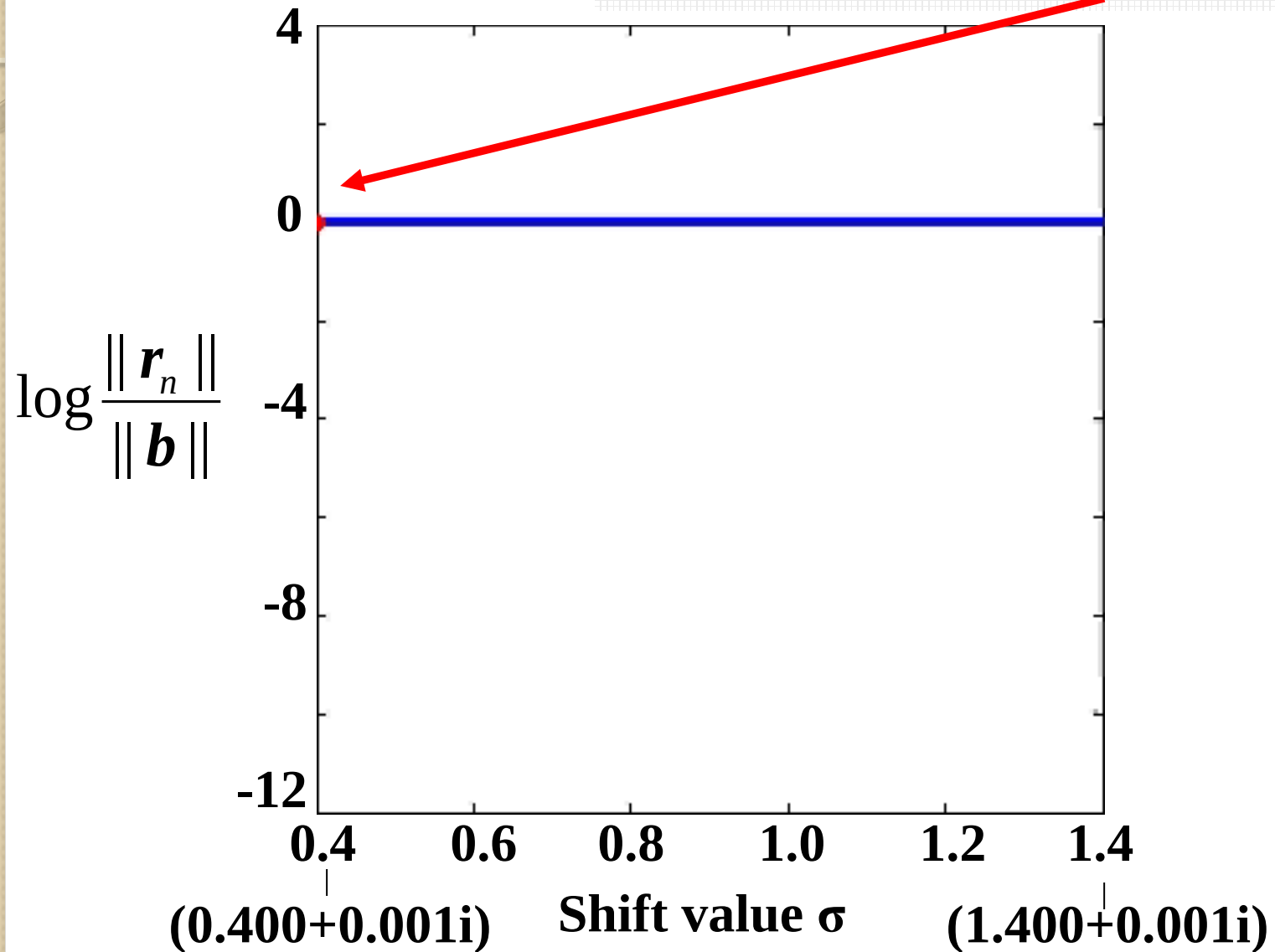


B:



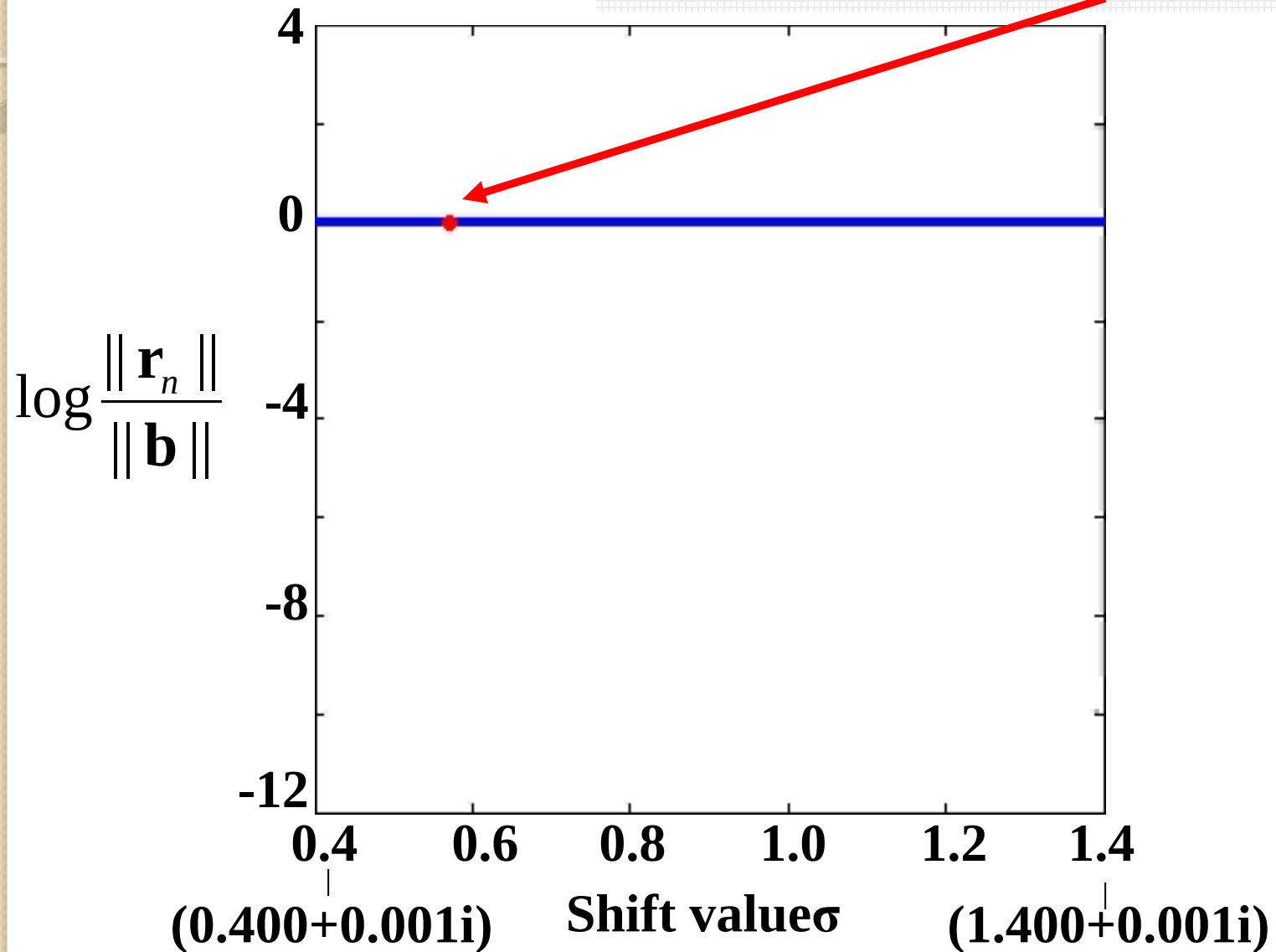
数値例 1

初期シード : $\sigma = 0.400 + 0.001i$



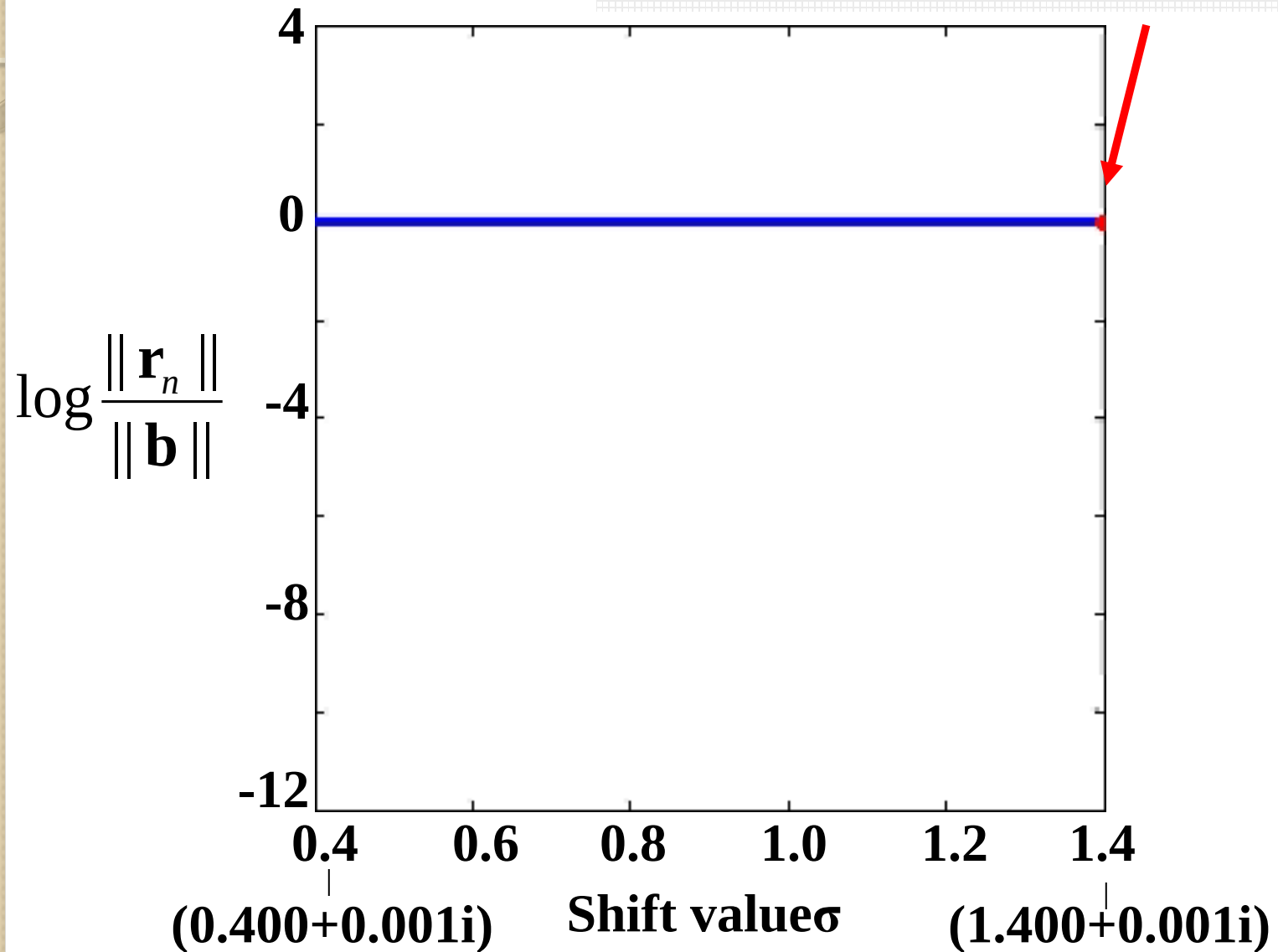
数値例 1

初期シード : $\sigma = 0.573 + 0.001i$



数値例 1

初期シード : $\sigma = 1.400 + 0.001i$



計算時間

COCG法: 1217.3 [sec.]

GSCOCG法: **33.7** [sec.]

計算時間の比 (COCG法/GSCOCG)

約**36倍**高速

数値例 2

Au 864 atoms

$$(A + \sigma_i B) x^{(i)} = b, \quad N: 7,776, Nnz: 3,619,104$$

$$\sigma_1 = 0.400 + 0.001i$$

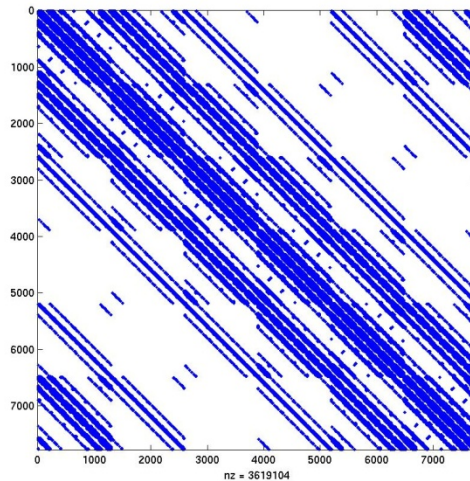
$$\sigma_2 = 0.401 + 0.001i$$

$\vdots$

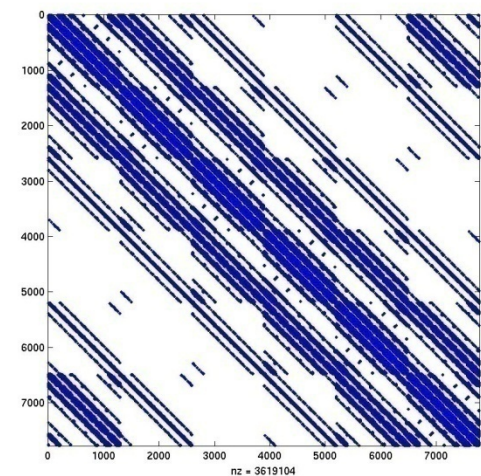
$$\sigma_{1001} = 1.400 + 0.001i$$

1001 個の方程式

A:



B:



計算時間

COCG法: 16867.8 [sec.]

4.7時間

GSCOCG法: **534.6** [sec.]

8.9分

計算時間の比 (COCG法/GSCOCG)

約**31.5倍**高速

固有値問題 (疎行列)

Sparse Eigenvalue Problems

$$A x = \lambda x$$

- Lanczos法とArnoldi法
- Jacobi-Davidson (JD) 法 [Sleijpen & van der Vorst, 1996]
- LOBPCG法 [Knyazev, 2001]
- Sakurai-Sugiura (SS) 法 [Sakurai, Sugiura, 2003]

固有値問題 (疎行列)

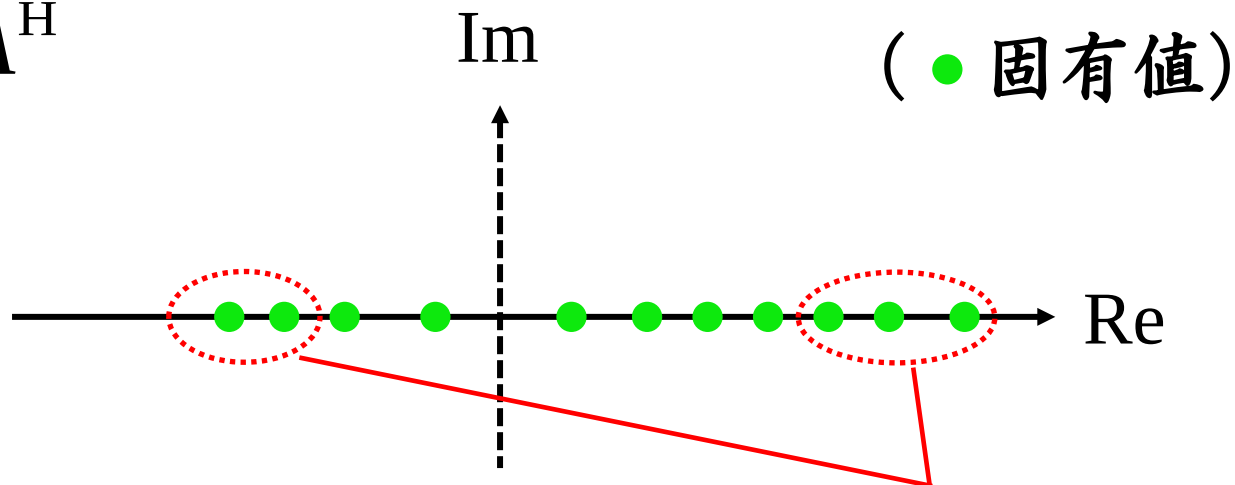
Sparse Eigenvalue Problems

$$A x = \lambda x$$

- **Lanczos法とArnoldi法**
- **Jacobi-Davidson法** [Sleijpen & van der Vorst, 1996]
- **LOBPCG法** [Knyazev, 2001]
- **Sakurai-Sugiura法** [Sakurai, Sugiura, 2003]

Lanczos法

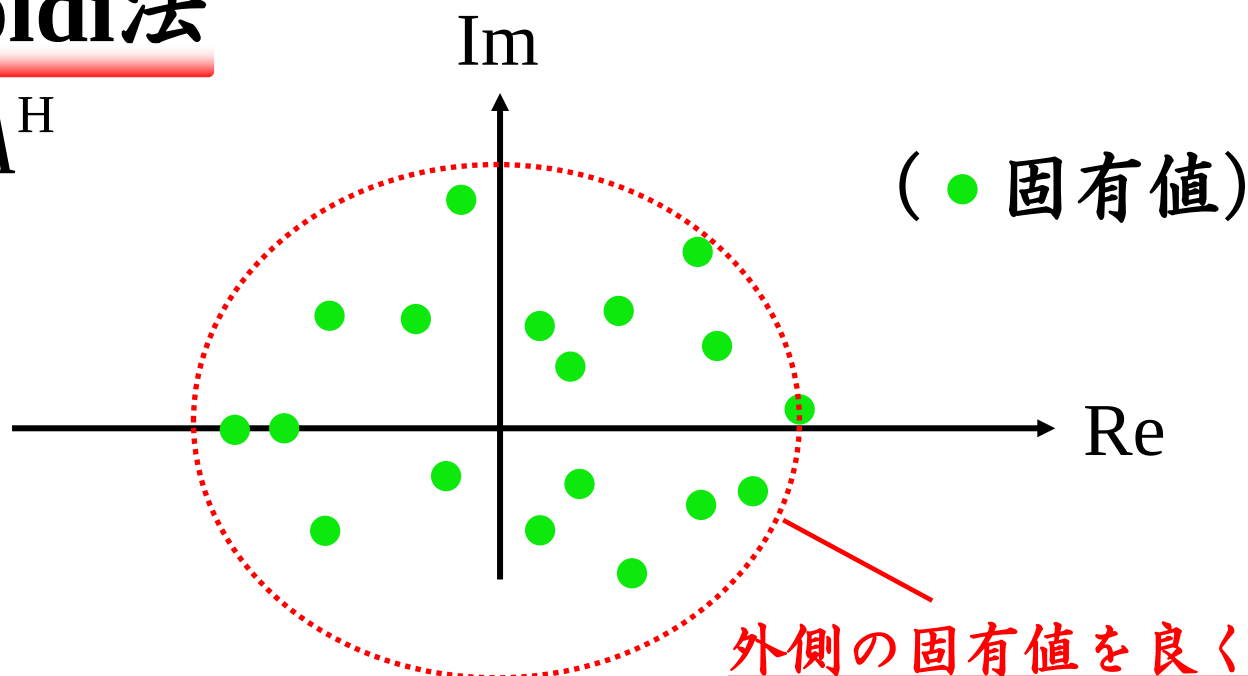
$$A = A^H$$



外側の固有値を良く近似

Arnoldi法

$$A \neq A^H$$



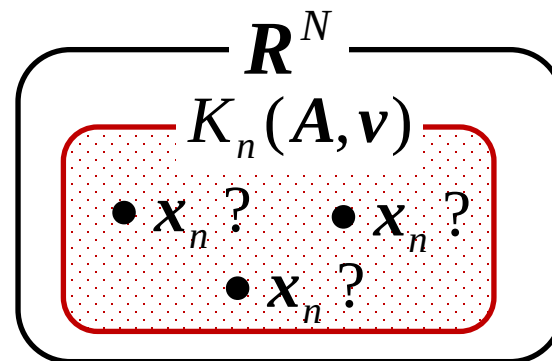
外側の固有値を良く近似

Lanczos法とArnoldi法

空間条件

$$\mathbf{x}_n \in K_n(A, \mathbf{v})$$

近似固有ベクトル $\mathbf{x}_n$ を探す
範囲に関する条件



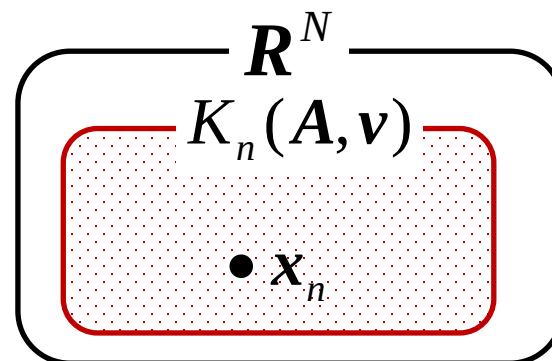
直交条件

$$\mathbf{r}_n := A\mathbf{x}_n - \mathcal{G}_n \mathbf{x}_n \perp K_n(A, \mathbf{v})$$

探索範囲の中から具体的に

- ・ 近似固有ベクトル $\mathbf{x}_n$
- ・ 近似固有値 $\mathcal{G}_n$

を求めるための条件



Lanczos法とArnoldi法

空間条件 (実現方法)

$$\mathbf{x}_n = V_n \mathbf{y}_n, \quad V_n := [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n]$$

↑ ↑ ↑
Arnoldi (Lanczos) 原理で得られる
正規直交基底を並べた行列

$\mathbf{y}_n$ をどう決めるか? (以下の直交条件で決める.)

直交条件 (実現方法)

$$\mathbf{r}_n = AV_n \mathbf{y}_n - \mathcal{G}_n V_n \mathbf{y}_n \perp K_n(A, \mathbf{v})$$

$$\iff V_n^H \mathbf{r}_n = \mathbf{0} \iff \underline{V_n^H AV_n} \mathbf{y}_n = \mathcal{G}_n \mathbf{y}_n \quad (\because V_n^H V_n = I_n)$$

$$V_n^H AV_n = \begin{cases} V_n^H V_{n+1} H_{n+1,n} =: H_n & \text{(Arnoldi法)} \\ V_n^H V_{n+1} T_{n+1,n} =: T_n & \text{(Lanczos法)} \end{cases}$$

H_n や T_n を得るための計算は不要

Lanczos法とArnoldi法

H_n や T_n を得るための計算は不要

$$H_{n+1,n} = \begin{bmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,n-1} & h_{1,n} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,n-1} & h_{2,n} \\ & \ddots & \ddots & \ddots & \vdots \\ & & \ddots & \ddots & \vdots \\ & & & h_{n-1,n} & h_{n,n} \\ & & & & h_{n+1,n} \end{bmatrix}$$

H_n

$$T_{n+1,n} = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \beta_n \\ & & & \beta_n & \alpha_n \\ & & & & \beta_{n+1} \end{bmatrix}$$

T_n

Lanczos法

(1) 行列 A に対してLanczos原理 (n 反復)

(2) $T_n \mathbf{y}_n = \mathcal{G}_n \mathbf{y}_n$ の固有値・固有ベクトルを計算※1

(3) $\mathbf{x}_n = V_n \mathbf{y}_n$

出力: $\mathcal{G}_n, \mathbf{x}_n$ (行列 A の外側の近似固有値と近似固有ベクトル)

Arnoldi法

(1) 行列 A に対してArnoldi原理 (n 反復)

(2) $H_n \mathbf{y}_n = \mathcal{G}_n \mathbf{y}_n$ の固有値・固有ベクトルを計算※2

(3) $\mathbf{x}_n = V_n \mathbf{y}_n$ (近似固有ベクトル)

出力: $\mathcal{G}_n, \mathbf{x}_n$ (行列 A の外側の近似固有値と近似固有ベクトル)

※1: QR法, 2分法+逆反復法, 分割統治法, MR³法

※2: QR法

スペクトル変換を用いたLanczos法・Arnoldi法

固有値は変わるが固有ベクトルは不変

$$A\mathbf{x} = \lambda \mathbf{x} \iff \underbrace{(A - \sigma I)^{-1}}_{A'} \mathbf{x} = \underbrace{(\lambda - \sigma)^{-1}}_{\lambda'} \mathbf{x} \quad (\sigma \neq \lambda)$$

$\lambda = \sigma + \lambda'^{-1}$: 変換した行列 A' の固有値から
元の行列 A の固有値が計算可能

➡ A' に対してLanczos法・Arnoldi法を適用

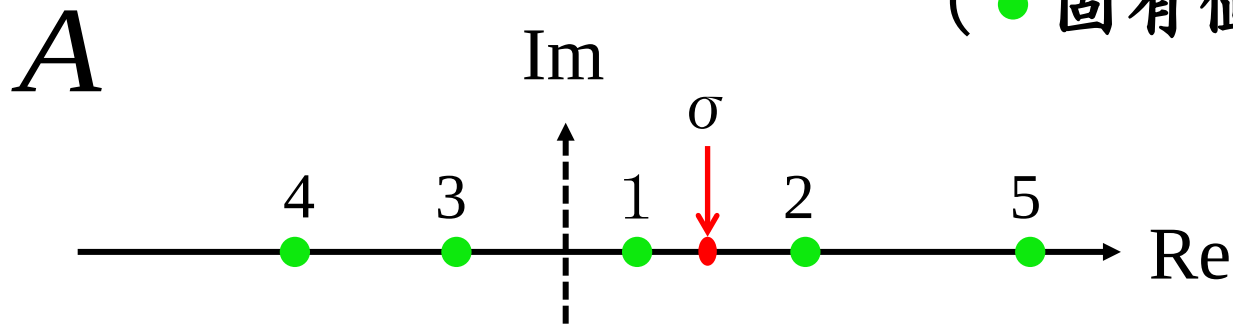
出力 : σ に近い固有値・固有ベクトル

狙った値

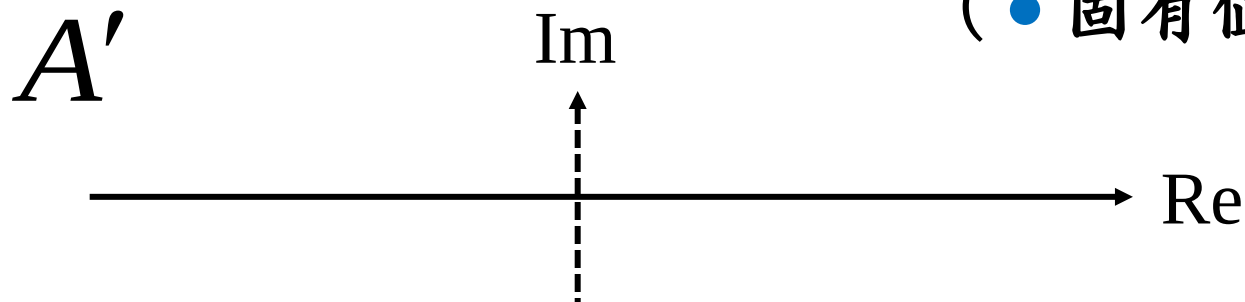
スペクトル変換を用いたLanczos法・Arnoldi法

$$A\mathbf{x} = \lambda \mathbf{x} \iff \underbrace{(A - \sigma I)^{-1}}_{A'} \mathbf{x} = \underbrace{(\lambda - \sigma)^{-1}}_{\lambda'} \mathbf{x} \quad (\sigma \neq \lambda)$$

(● 固有値 λ)



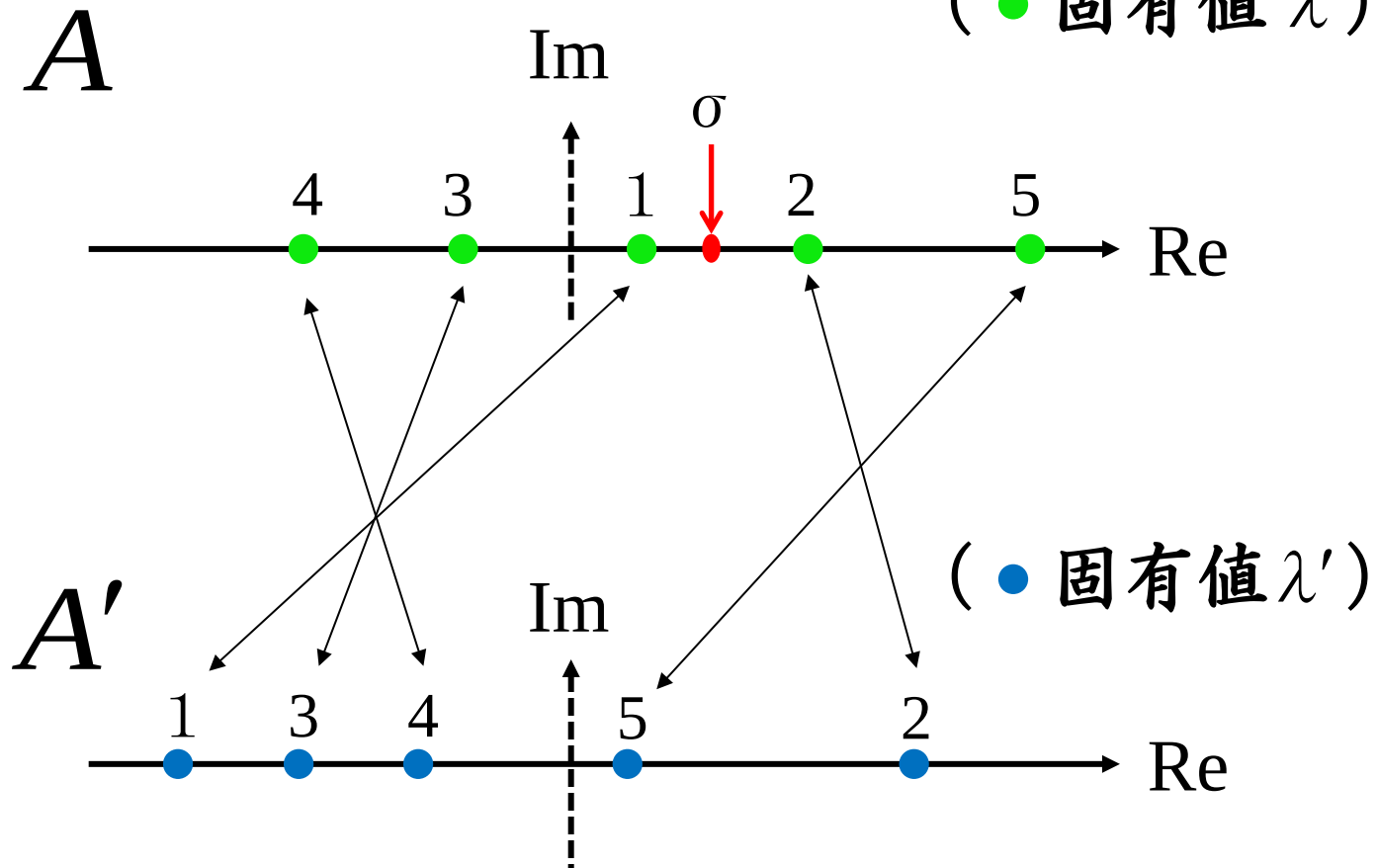
(● 固有値 λ')



スペクトル変換を用いたLanczos法・Arnoldi法

$$A\mathbf{x} = \lambda \mathbf{x} \iff \underbrace{(A - \sigma I)^{-1}}_{A'} \mathbf{x} = \underbrace{(\lambda - \sigma)^{-1}}_{\lambda'} \mathbf{x} \quad (\sigma \neq \lambda)$$

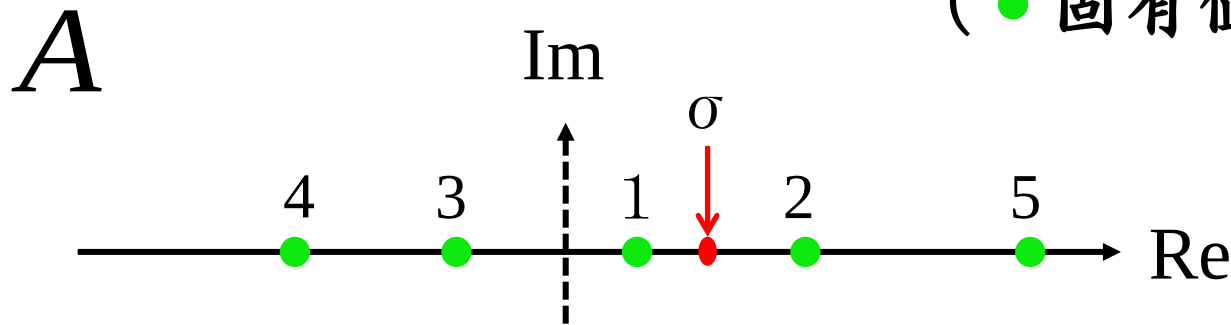
(● 固有値 λ)



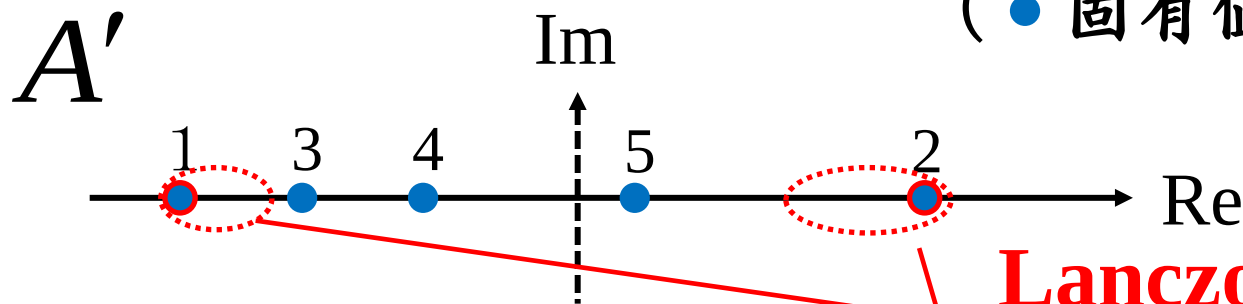
スペクトル変換を用いたLanczos法・Arnoldi法

$$A\mathbf{x} = \lambda \mathbf{x} \iff \underbrace{(A - \sigma I)^{-1}}_{A'} \mathbf{x} = \underbrace{(\lambda - \sigma)^{-1}}_{\lambda'} \mathbf{x} \quad (\sigma \neq \lambda)$$

(● 固有値 λ)



(● 固有値 λ')



Lanczos法

外側の固有値を良く近似

Shift-and-invert Lanczos法

- (1) 行列 $(A - \sigma I)^{-1}$ に対してLanczos原理 (n 反復)
 - ・ 各反復で $w_k := (A - \sigma I)^{-1} v_k$ の計算が必要
→ 線形方程式 $(A - \sigma I)w_k = v_k$ を解けばよい
- (2) $T_n y_n = \mathcal{G}'_n y_n$ の固有値・固有ベクトルを計算※1
 $\mathcal{G}'_n$ は通常 n 個あるが, n 個の中で最も外側の値を選ぶ.
- (3) $x_n = V_n y_n$
- (4) $\theta_n = \sigma + \mathcal{G}'_n^{-1}$

出力: $\mathcal{G}_n, x_n$ (σ に近い A の近似固有値と近似固有ベクトル)

Shift-and-invert Arnoldi法も同様

※1: QR法, 2分法+逆反復法, 分割統治法, MR³法

その他

・ 再直交化

Lanczos原理・Arnoldi原理

→ クリロフ部分空間の正規直交基底の生成

数値計算では丸め誤差等の影響により直交性の崩れ

→ 偽固有値

対応： $\left\{ \begin{array}{l} (1) \text{ 再直交化： } \mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \text{ を直交化しなおす。} \\ (2) \text{ 部分再直交化： 直交性が低いベクトルの組} \\ \quad \text{だけを直交化しなおす。} \end{array} \right.$

・ リスタート

反復回数の増加 → メモリ量・演算量の増加，丸め誤差蓄積

反復回数に上限値を定め，上限値に達すると，
対応： 得られた1本の近似固有ベクトルを初期ベクトル
として再計算する。

→ Thick Restart Lanczos 法・Implicitly restarted Arnoldi法

関連研究（宣伝）

Arnoldi(M,W,G)法

山下達也・宮田考史・曾我部知広・星健夫・藤原毅夫・張紹良
一般化固有値問題に対するArnoldi(M,W,G)法,
日本応用数理学会論文誌, Vol. 21, No. 3, 2011, pp. 241-254.

固有値問題 (疎行列)

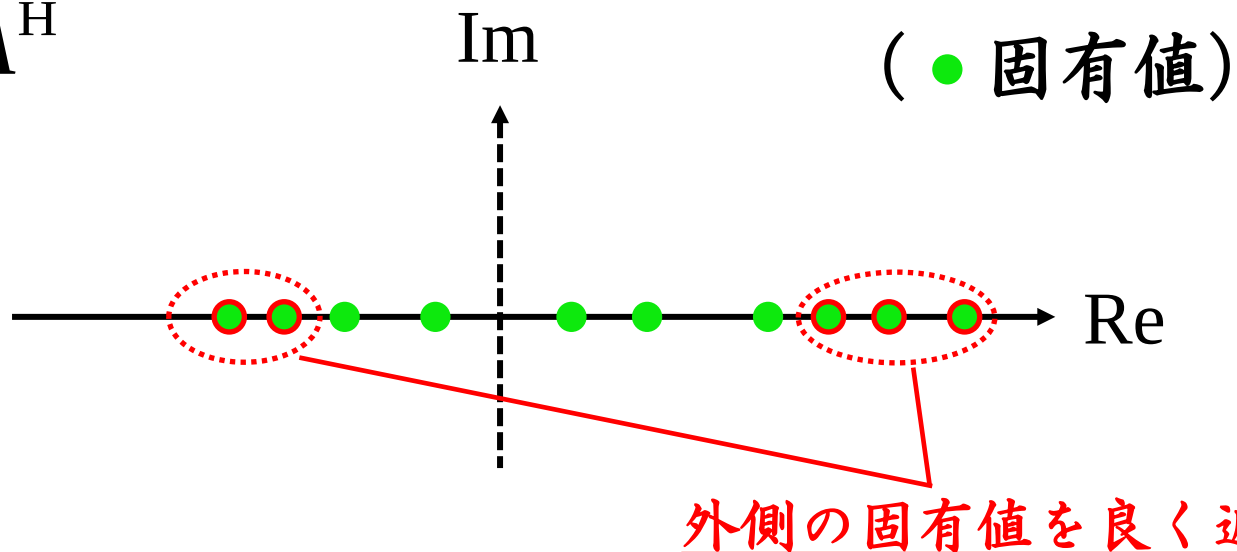
Sparse Eigenvalue Problems

$$A x = \lambda x$$

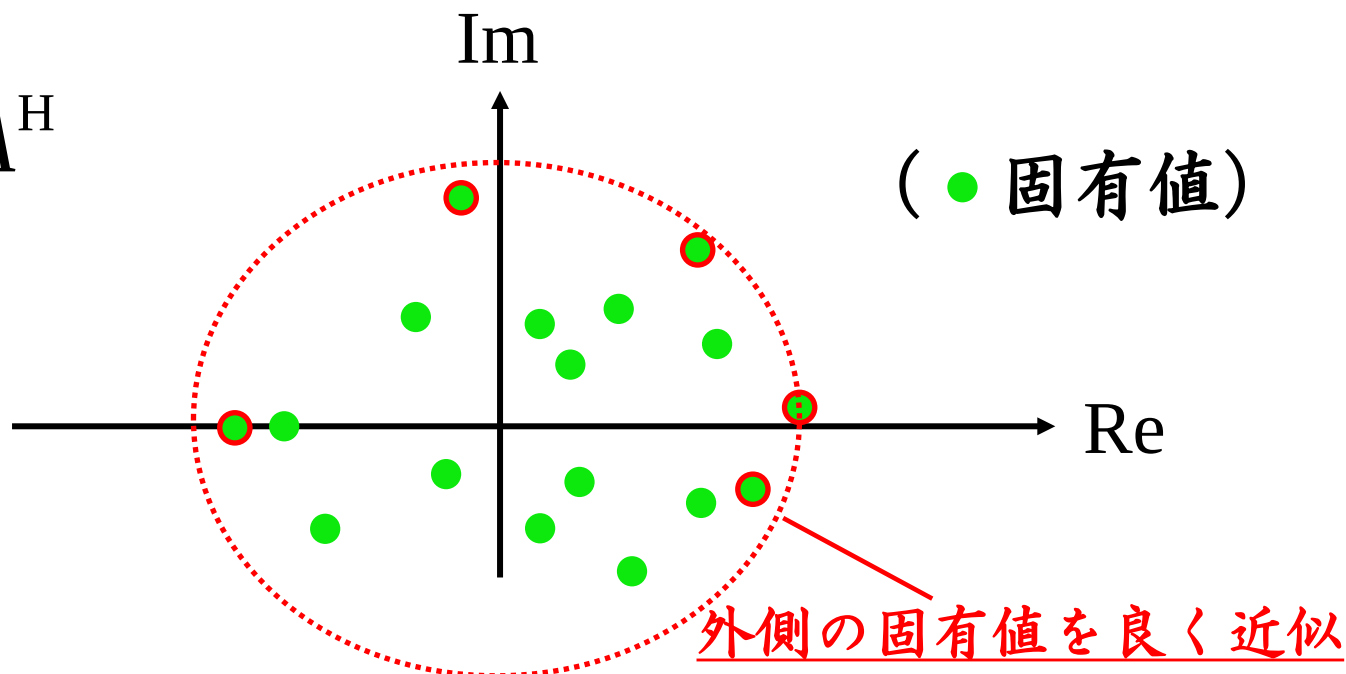
- Lanczos法とArnoldi法
- **Jacobi-Davidson法** [Sleijpen & van der Vorst, 1996]
- LOBPCG法 [Knyazev, 2001]
- Sakurai-Sugiura法 [Sakurai, Sugiura, 2003]

Jacobi-Davidson法

$$A = A^H$$



$$A \neq A^H$$

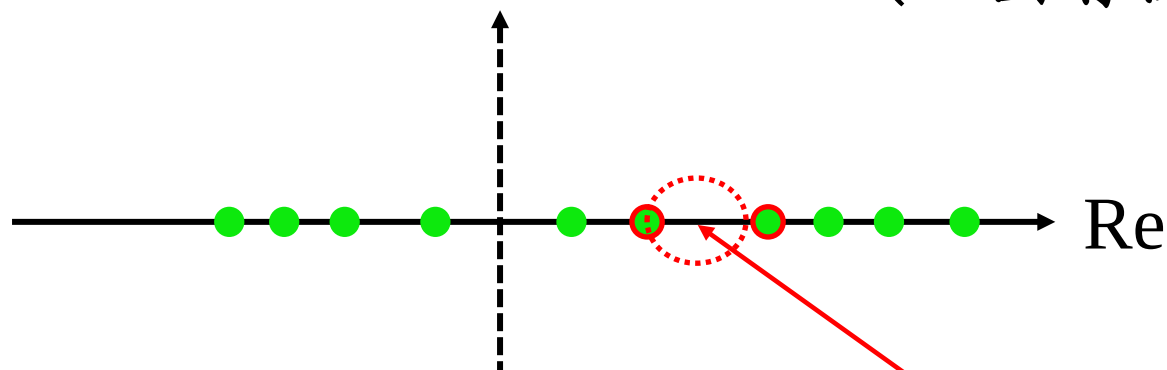


Jacobi-Davidson法

$$A = A^H$$

Im

(● 固有値)

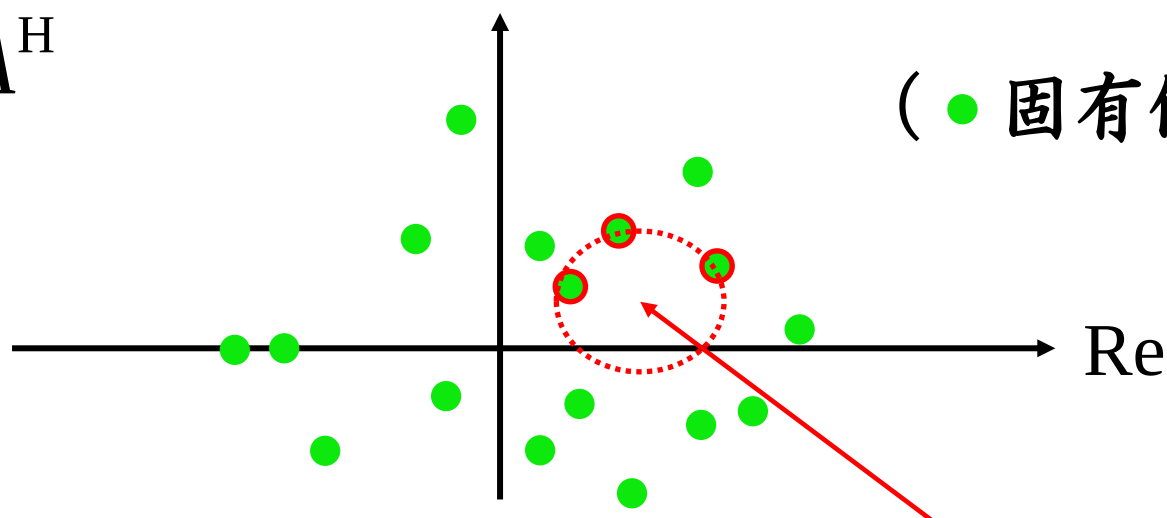


狙った点の近くの固有値を良く近似

$$A \neq A^H$$

Im

(● 固有値)



狙った点の近くの固有値を良く近似

Jacobi-Davidson法

対象とする問題
 $Ax = \lambda x$

u : 規格化された近似固有ベクトル

理想

$A(u+t) = \lambda(u+t)$ となる修正ベクトル t が欲しい.

$t \perp u$ だと無駄がない (u の直交補空間から探す)

条件 $(I - uu^H)t = t$ ($t \perp u$ のとき成立)

$$A(u+t) = \lambda(u+t)$$

$$\Rightarrow (I - uu^H)A(u+t) = \lambda(I - uu^H)(u+t)$$

(u の直交補空間上での方程式)



条件 $(I - uu^H)t = t$ を用いて式を変形

修正ベクトル t の満たすべき方程式

$$(I - uu^H)(A - \lambda I)(I - uu^H)t = -r$$

ここで $r := Au - \theta u$, $\theta := (u, Au)$

Jacobi-Davidson法

対象とする問題
 $Ax = \lambda x$

修正ベクトル t の満たすべき方程式

$$(I - uu^H)(A - \lambda I)(I - uu^H)t = -r$$

ここで $r := Au - \theta u$, $\theta := (u, Au)$

問題点：上式には固有値 λ (未知量) がある。

対応： λ を θ (既知量) で代用する。

修正方程式

$$(I - uu^H)(A - \underline{\theta} I)(I - uu^H)t = -r$$

ここで $r := Au - \theta u$, $\theta := (u, Au)$

修正方程式は線形方程式になっていることに注意

→ クリロフ部分空間法 (例えばGMRES法) で解く。

GMRES法は数反復程度、つまりとても粗く解いてもよいことが (半経験的・半理論的) に知られている。

Jacobi-Davidson法

対象とする問題
 $Ax = \lambda x$

v_1 : 初期ベクトルの設定 ($\|v_1\|_2 = 1$)

for $k = 1, 2, \dots$ (停止条件を満たすまで)

- $V_k := [v_1, v_2, \dots, v_k]$
- $(V_k^H A V_k) y_k = \theta_k y_k$ ($k \times k$ の小規模固有値問題を解く)
→ (例えば絶対値最大の) 固有値・固有ベクトルを計算
- $u_k := V_k y_k$ (近似固有ベクトル)
- 修正方程式を解き, 修正ベクトル t を求める.
→ 実際は修正方程式と粗く解くため,
得られた近似解を t_{k+1} とおく.
- $\tilde{v}_{k+1} := (I - V_k V_k^H) t_{k+1}$
 $v_{k+1} := \tilde{v}_{k+1} / \|\tilde{v}_{k+1}\|_2$ } t_{k+1} を $v_1, v_2, \dots, v_k$ に対して
直交化後に規格化

end

次の反復において, 良い方向で1次元拡大された部分空間から近似固有値・近似固有ベクトルを求めることができる.

その他

- ・ JDQR法：複数の固有値と固有ベクトルが計算可能
- ・ Jacobi-Davidson法系統では，調和Ritz値と呼ばれる考え方をを用いて，狙った場所付近の固有値と固有ベクトルを計算可能にする手法がある。
 - 一方，shift-and-invert Arnoldi法では，反復毎に線形方程式（修正方程式のことではない）をきちんと解く必要がある。
- ・ Jacobi-Davidson法は，一般化固有値問題（JDQZ法），や特異値問題（JDSVD法）などにも拡張されている。

関連研究（宣伝）

宮田考史・曾我部知広・張紹良，
Jacobi-Davidson 法における修正方程式の解法－射影空間における Krylov 部分空間のシフト不変性に基づいて－
日本応用数理学会論文誌，20(2)，2010，pp. 115-129.

固有値問題（疎行列）

Sparse Eigenvalue Problems

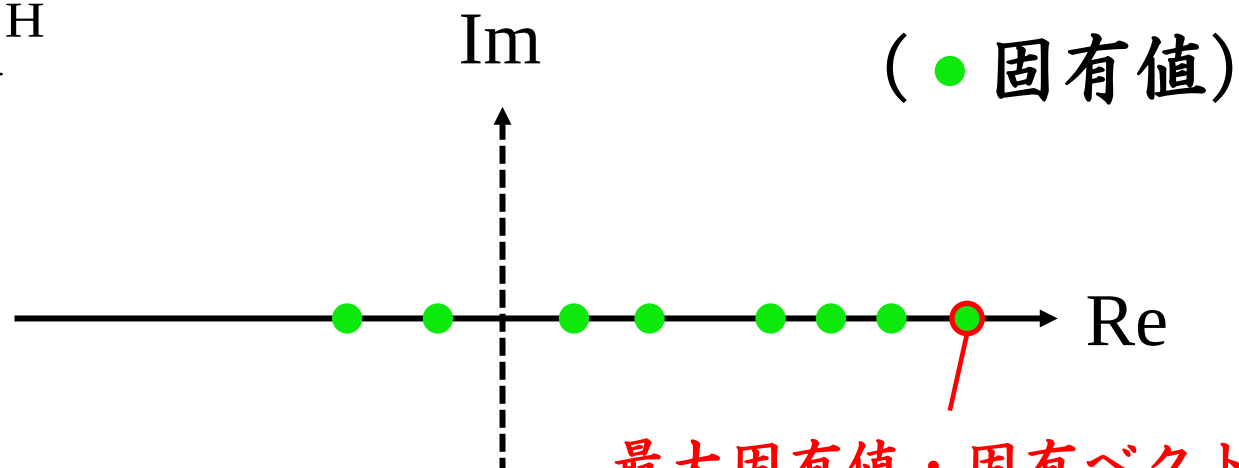
$$A x = \lambda x$$

- Lanczos法とArnoldi法
- Jacobi-Davidson法 [Sleijpen & van der Vorst, 1996]
- **LOBPCG**法[Knyazev, 2001]
- Sakurai-Sugiura法 [Sakurai, Sugiura, 2003]

Locally **O**ptimal **B**lock **P**reconditioned **C**onjugate **G**radient

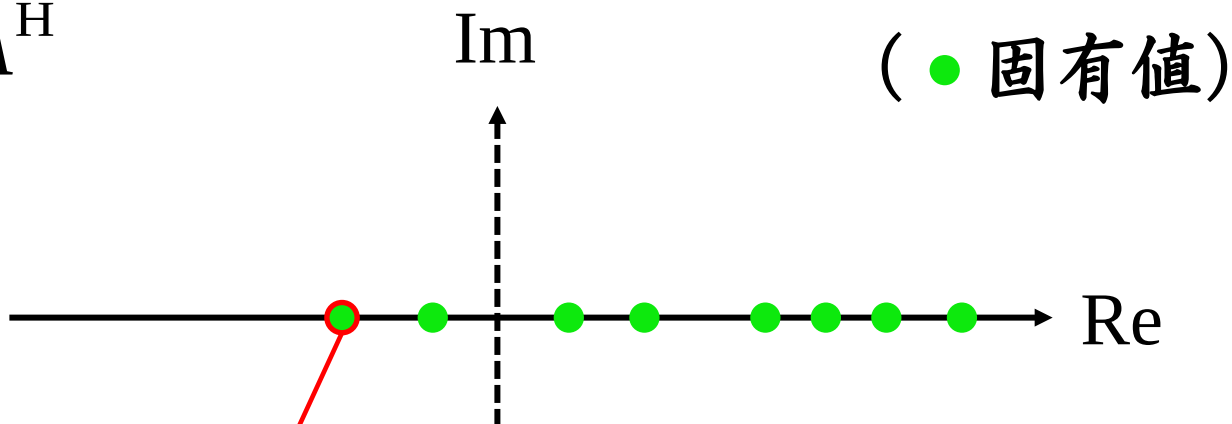
LOBPCG法

$$A = A^H$$



最大固有値・固有ベクトル
効率よく計算する.

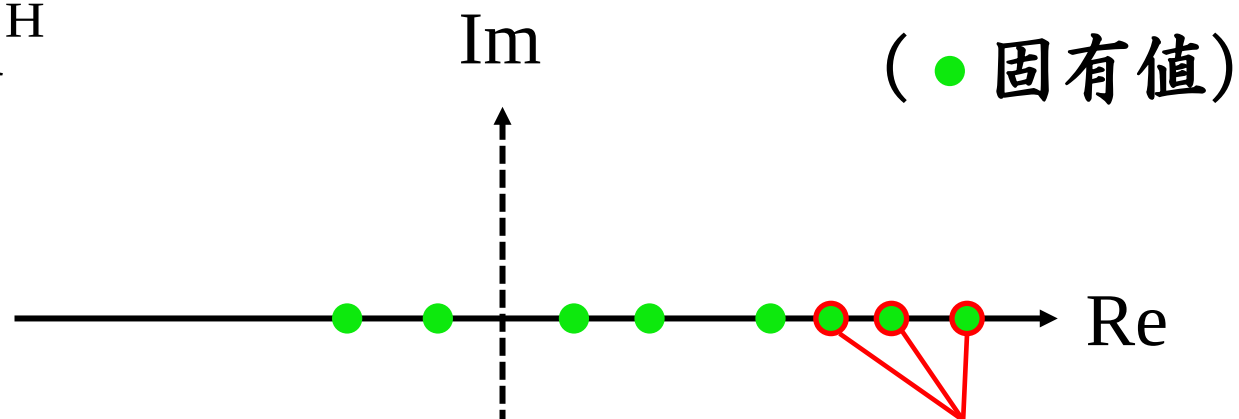
$$A = A^H$$



最小固有値・固有ベクトル
効率よく計算する.

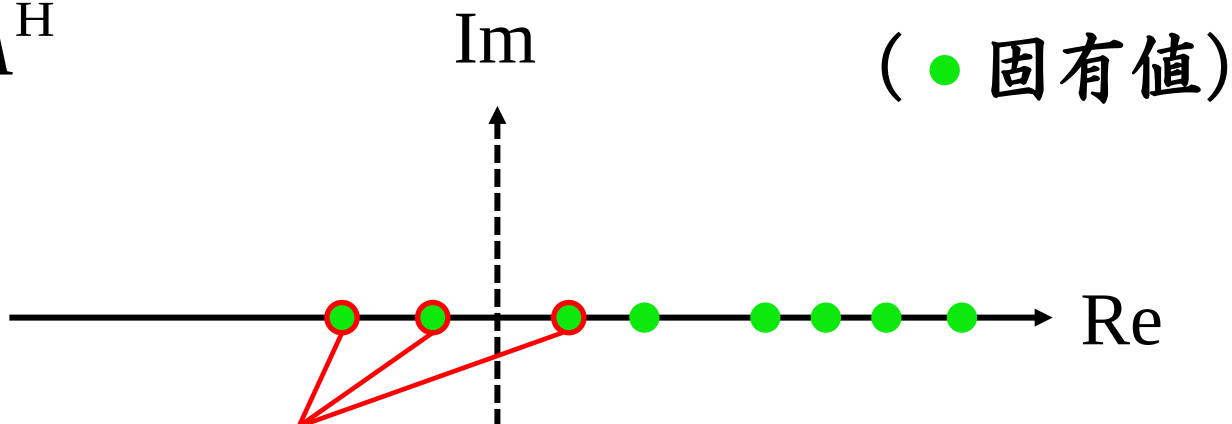
LOBPCG法

$$A = A^H$$



最大付近の固有値・固有ベクトル
を効率よく計算する。

$$A = A^H$$



最小付近の固有値・固有ベクトル
を効率よく計算する。

LOBPCG法

対象：一般化固有値問題 $Ax = \lambda Bx$

【注意】 Knyazevの論文では A, B を以下のように仮定している.

A : エルミートかつ正定値, B : エルミート

※線形計算や計算物質科学では違和感を感じるかもしれません.

線形計算の視点

標準固有値問題 $Ax = \lambda Ix$

一般化



$Ax = \lambda Bx$

エルミート行列
(正定値の制約なし)

単位行列なので
エルミートかつ正定値

計算物質科学の視点

$$H\varphi_k = \varepsilon_k S\varphi_k$$

H : ハミルトニアン行列, S : 重なり行列

LOBPCG法

対象：一般化固有値問題 $Ax = \lambda Bx$

【注意】 Knyazevの論文では A, B を以下のように仮定している.

A : エルミートかつ正定値, B : エルミート

※線形計算や計算物質科学では違和感を感じるかもしれません.

一方, 構造力学分野では,

$$Ax = \lambda Bx$$

剛性行列
エルミート行列かつ正定値

質量行列
エルミート行列かつ正定値

座屈の場合は, **正定値で
ない**ことがあるようです.

LOBPCG法

対象：一般化固有値問題 $Ax = \lambda Bx$

【注意】 Knyazevの論文では A, B を以下のように仮定している.

A : エルミートかつ正定値, B : エルミート

- ・ B が正則のとき, 全ての固有値 λ は有限の値
- ・ B が半正定値のとき, 全ての固有値 λ は正

LOBPCG法では, 固有値問題を最適化問題として捉える.

$$\min_{x \in \mathbb{C}^N} f(x), \quad f(x) := \frac{(x, Ax)}{(x, Bx)}$$

しかし, これでは分母に0除算が発生する可能性があるため

$$\mu Ax = Bx \quad (\mu := \lambda^{-1})$$

を考える.

LOBPCG法

LOBPCG法では，固有値問題を最適化問題として捉える．

$$\max_{x \in \mathbb{C}^N} \mu(x), \quad \mu(x) := \frac{(x, Bx)}{(x, Ax)} \neq 0$$

- $\max_{x \in \mathbb{C}^N} \mu(x)$ の逆数が最小固有値になる．
- そのときの x が最小固有値に対応する固有ベクトルになる．

理想

$$x_{\text{opt}}^{(k)} = \arg \max_{x \in K_k(TA, TB, x^{(0)})} \mu(x)$$

$K_k(TA, TB, x^{(0)})$: 一般化クリロフ部分空間 (T : 前処理行列)

$k=1$ のとき

$$\underline{K}_1 = \text{span}\{x^{(0)}, TA x^{(0)}, TB x^{(0)}\} \quad \times K_1 \text{ は } K_1(TA, TB, x^{(0)}) \text{ の略記}$$

$k=2$ のとき

$$\underline{K}_2 = \underline{K}_1 + \text{span}\{(TA)^2 x^{(0)}, (TA)(TB) x^{(0)}, (TB)(TA) x^{(0)}, (TB)^2 x^{(0)}\}$$

LOBPCG法

LOBPCG法では，固有値問題を最適化問題として捉える．

$$\max_{x \in \mathbb{C}^N} \mu(x), \quad \mu(x) := \frac{(x, Bx)}{(x, Ax)} \neq 0$$

- $\max_{x \in \mathbb{C}^N} \mu(x)$ の逆数が最小固有値になる．
- そのときの x が最小固有値に対応する固有ベクトルになる．

理想

$$x_{\text{opt}}^{(k)} = \arg \max_{x \in K_k(TA, TB, x^{(0)})} \mu(x)$$

理想の実現： k の増加→長い漸化式→計算効率の悪化の可能性
理想に近づけるために，CG法を参考に短い漸化式で構成する．

- $r^{(k)} := Bx^{(k)} - \mu^{(k)} Ax^{(k)}, \quad w^{(k)} := Tr^{(k)}$
- $x^{(k+1)} := \alpha^{(k)} w^{(k)} + \tau^{(k)} x^{(k)} + \gamma^{(k)} p^{(k)} \in K_k(TA, TB, x^{(0)})$
- $p^{(k+1)} := \alpha^{(k)} w^{(k)} + \gamma^{(k)} p^{(k)}$ (前処理付きCG法の形に似ている)

LOBPCG法 $\left[\mathbf{x}^{(0)} : \text{初期ベクトルの設定}, \mathbf{p}^{(0)} := \mathbf{0} \right]$

for $k = 0, 1, \dots$

- $\mu^{(k)} := (\mathbf{x}^{(k)}, B\mathbf{x}^{(k)}) / (\mathbf{x}^{(k)}, A\mathbf{x}^{(k)})$
- $\mathbf{r}^{(k)} := B\mathbf{x}^{(k)} - \mu^{(k)} A\mathbf{x}^{(k)}, \quad \mathbf{w}^{(k)} := T\mathbf{r}^{(k)}$
- $V := [\mathbf{w}^{(k)}, \mathbf{x}^{(k)}, \mathbf{p}^{(k)}]$
- $(V^H B V) \mathbf{y} = \mu (V^H A V) \mathbf{y} \quad (\leftarrow 3 \times 3 \text{ の一般化固有値問題})$
※ “最大” 固有値と固有値ベクトル $\mathbf{y} = [\alpha^{(k)}, \tau^{(k)}, \gamma^{(k)}]^T$ を求める。
- $\mathbf{x}^{(k+1)} := \alpha^{(k)} \mathbf{w}^{(k)} + \tau^{(k)} \mathbf{x}^{(k)} + \gamma^{(k)} \mathbf{p}^{(k)}$
- $\mathbf{p}^{(k+1)} := \alpha^{(k)} \mathbf{w}^{(k)} + \gamma^{(k)} \mathbf{p}^{(k)}$

end

※最小に変更すると計算物質科学で都合がよいと思います。

LOBPCG法 $\left[\mathbf{x}_j^{(0)} (j=1,2,\dots,m), \mathbf{p}_j^{(0)} := \mathbf{0} (j=1,2,\dots,m) \right]$

for $k = 0, 1, \dots$

- $\mu_j^{(k)} := (\mathbf{x}_j^{(k)}, B\mathbf{x}_j^{(k)}) / (\mathbf{x}_j^{(k)}, A\mathbf{x}_j^{(k)}) \dots\dots\dots (j=1,2,\dots,m)$
- $\mathbf{r}_j^{(k)} := B\mathbf{x}_j^{(k)} - \mu_j^{(k)} A\mathbf{x}_j^{(k)}, \quad \mathbf{w}_j^{(k)} := \text{Tr}_j^{(k)} \dots\dots\dots (j=1,2,\dots,m)$
- $V := [\mathbf{w}_1^{(k)}, \dots, \mathbf{w}_m^{(k)}, \mathbf{x}_1^{(k)}, \dots, \mathbf{x}_m^{(k)}, \mathbf{p}_1^{(k)}, \dots, \mathbf{p}_m^{(k)}]$
- $(V^H B V) \mathbf{y} = \mu (V^H A V) \mathbf{y} \quad (\leftarrow 3m \times 3m \text{ の一般化固有値問題})$
※ “最大” から m 個の固有値と固有値ベクトル $\mathbf{y}_j$ を求める。

$$\mathbf{y}_j = [\alpha_{1,j}^{(k)}, \dots, \alpha_{m,j}^{(k)}, \tau_{1,j}^{(k)}, \dots, \tau_{m,j}^{(k)}, \gamma_{1,j}^{(k)}, \dots, \gamma_{m,j}^{(k)}]^T$$

- $\mathbf{x}_j^{(k+1)} := \sum_{\ell=1}^m \alpha_{\ell,j}^{(k)} \mathbf{w}_\ell^{(k)} + \tau_{\ell,j}^{(k)} \mathbf{x}_\ell^{(k)} + \gamma_{\ell,j}^{(k)} \mathbf{p}_\ell^{(k)} (= V \mathbf{y}_j) \dots\dots\dots (j=1,2,\dots,m)$
- $\mathbf{p}_j^{(k+1)} := \sum_{\ell=1}^m \alpha_{\ell,j}^{(k)} \mathbf{w}_\ell^{(k)} + \gamma_{\ell,j}^{(k)} \mathbf{p}_\ell^{(k)} \dots\dots\dots (j=1,2,\dots,m)$

end

※最小に変更すると計算物質科学で都合がよいと思います。

固有値問題（疎行列）

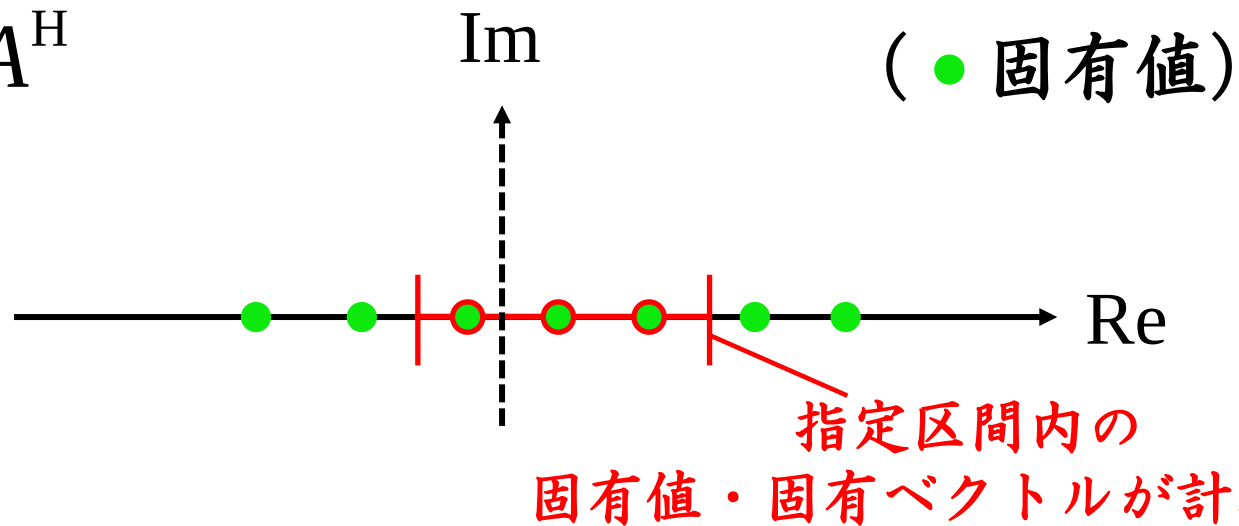
Sparse Eigenvalue Problems

$$A x = \lambda x$$

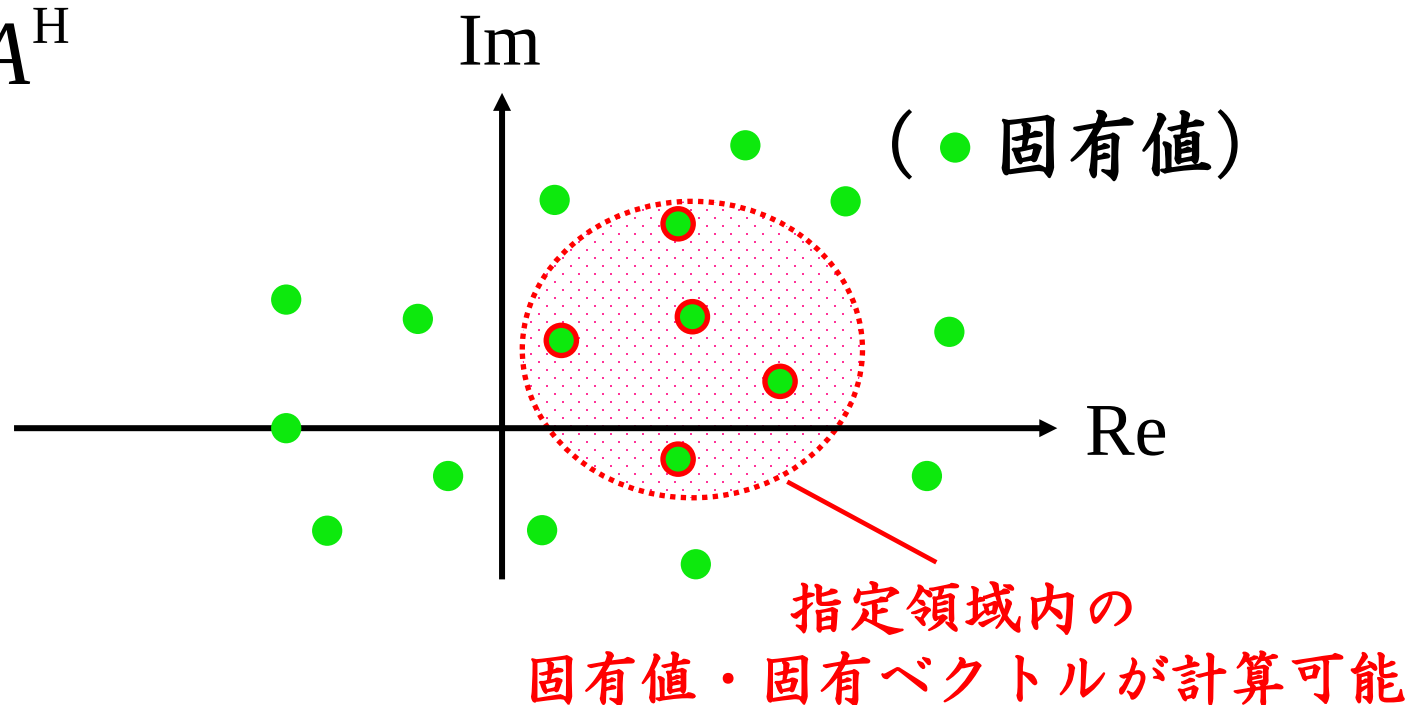
- Lanczos法とArnoldi法
- Jacobi-Davidson法 [Sleijpen & van der Vorst, 1996]
- LOBPCG法 [Knyazev, 2001]
- Sakurai-Sugiura法 [Sakurai, Sugiura, 2003]

Sakurai-Sugiura法

$$A = A^H$$

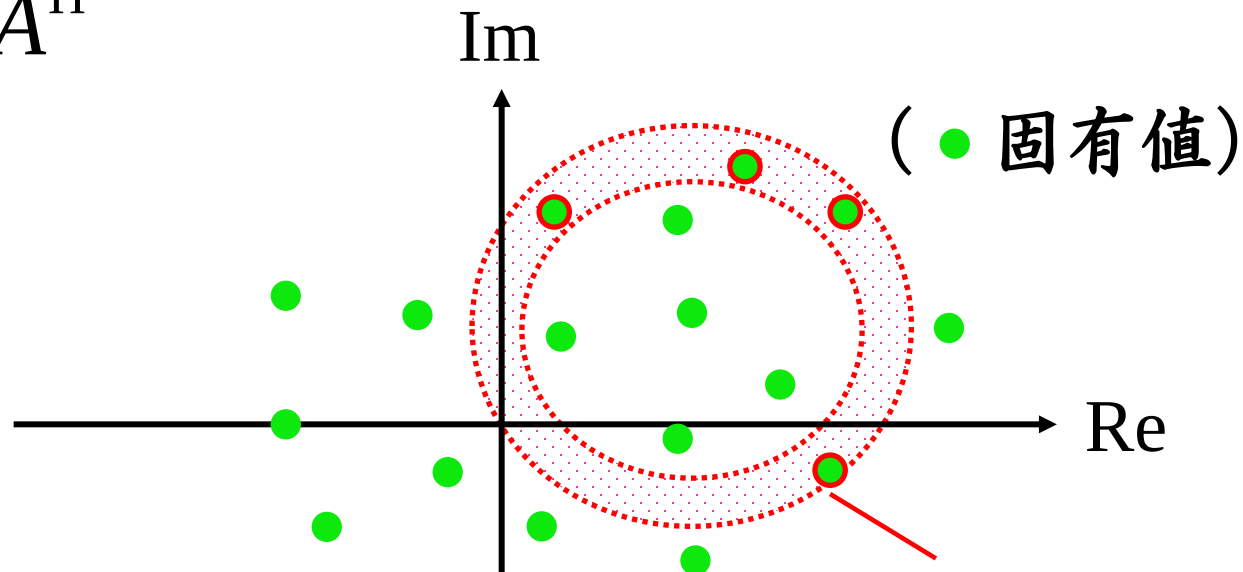


$$A \neq A^H$$



Sakurai-Sugiura法の拡張

$$A \neq A^H$$



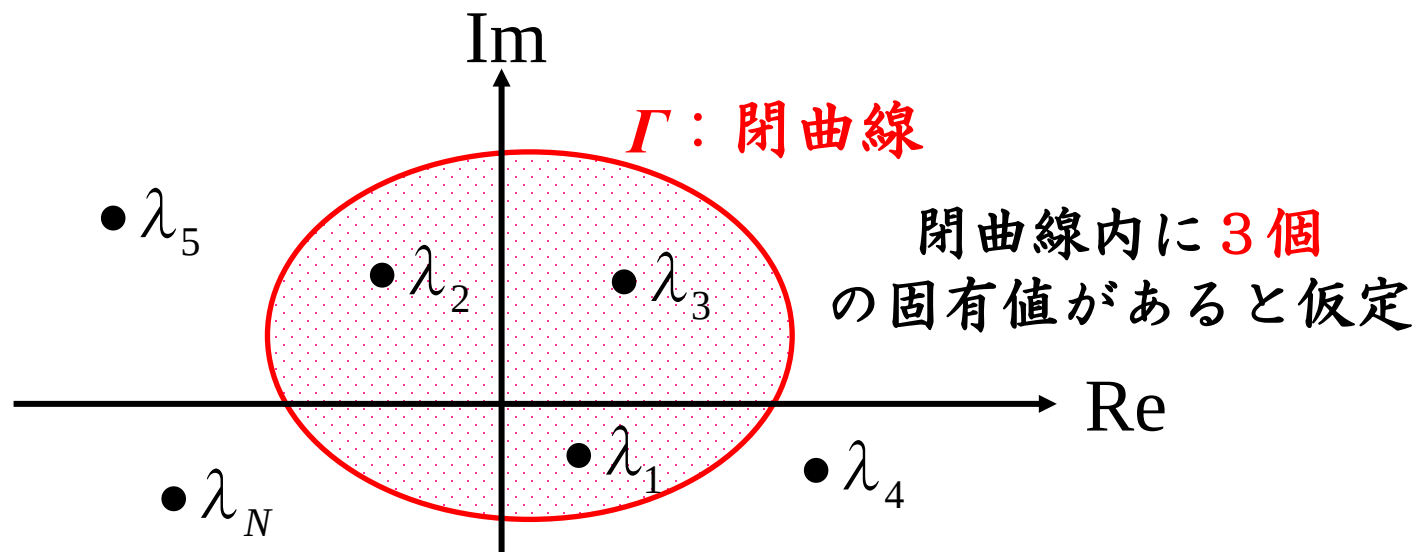
穴が開いている領域内の
固有値・固有ベクトルが計算可能

宮田考史・杜磊・曾我部知広・山本有作・張紹良,
多重連結領域の固有値問題に対する Sakurai-Sugiura 法の拡張,
日本応用数理学会論文誌, 19(4), 2009, pp. 537-550.

Sakurai-Sugiura法

一般論を避け，具体的な説明を試みます。

(→本来は大変汎用的で理論的・実用的にも発展しています。)



定義 $f(z) := \mathbf{u}^H (zI - A)^{-1} \mathbf{v} \quad (\mathbf{u}, \mathbf{v} \neq \mathbf{0})$

$$\oint_{\Gamma} f(z) dz$$

$$= \oint_{\Gamma} a_1 (z - \lambda_1)^{-1} + a_2 (z - \lambda_2)^{-1} + a_3 (z - \lambda_3)^{-1} + \cdots + a_N (z - \lambda_N)^{-1} dz$$

$$= \oint_{\Gamma} a_1 (z - \lambda_1)^{-1} + a_2 (z - \lambda_2)^{-1} + a_3 (z - \lambda_3)^{-1} dz \quad (\neq 0)$$

Sakurai-Sugiura法

理想 $\varphi_{\text{ideal}}(z) := (z - \lambda_1)(z - \lambda_2)(z - \lambda_3)$ が見つかると

$$\begin{aligned} & \oint_{\Gamma} \varphi_{\text{ideal}}(z) f(z) dz \\ &= \oint_{\Gamma} a_1(z - \lambda_2)(z - \lambda_3) + a_2(z - \lambda_1)(z - \lambda_3) + a_3(z - \lambda_1)(z - \lambda_2) dz \\ &= 0 \end{aligned}$$

アイデア

- $f(z)$ は領域 Γ 内で極を持つので $\oint_{\Gamma} f(z) dz \neq 0$
- $\oint_{\Gamma} \varphi(z) f(z) dz = 0$ となる3次多項式 $\varphi(z)$ が見つかると
 $\varphi(z)$ の根は A の固有値に対応する

固有値問題



$\varphi(z) f(z)$ が Γ 内で極をもたなくなるような多項式 $\varphi(z)$ とその根を求める問題に帰着させる.

Sakurai-Sugiura法

$\varphi(z) := z^3 + c_2 z^2 + c_1 z + c_0$ とおくと満たすべき方程式は

- $\oint_{\Gamma} \varphi(z) f(z) dz$

$$= \oint_{\Gamma} z^3 f(z) dz + c_2 \oint_{\Gamma} z^2 f(z) dz + c_1 \oint_{\Gamma} z f(z) dz + c_0 \oint_{\Gamma} f(z) dz$$

$$= \underline{\mu_3 + c_2 \mu_2 + c_1 \mu_1 + c_0 \mu_0} = 0$$

この式だけでは, c_0, c_1, c_2 が決まらないので以下の計算を行う.

- $\oint_{\Gamma} \underline{z} \varphi(z) f(z) dz$

$$= \oint_{\Gamma} z^4 f(z) dz + c_2 \oint_{\Gamma} z^3 f(z) dz + c_1 \oint_{\Gamma} z^2 f(z) dz + c_0 \oint_{\Gamma} z f(z) dz$$

$$= \underline{\mu_4 + c_2 \mu_3 + c_1 \mu_2 + c_0 \mu_1} = 0$$

- $\oint_{\Gamma} \underline{z^2} \varphi(z) f(z) dz$

$$= \oint_{\Gamma} z^5 f(z) dz + c_2 \oint_{\Gamma} z^4 f(z) dz + c_1 \oint_{\Gamma} z^3 f(z) dz + c_0 \oint_{\Gamma} z^2 f(z) dz$$

$$= \underline{\mu_5 + c_2 \mu_4 + c_1 \mu_3 + c_0 \mu_2} = 0$$

Sakurai-Sugiura法

- $\varphi(z) := z^3 + c_2 z^2 + c_1 z + c_0$ とおくと満たすべき方程式は

$$\underbrace{\begin{bmatrix} \mu_0 & \mu_1 & \mu_2 \\ \mu_1 & \mu_2 & \mu_3 \\ \mu_2 & \mu_3 & \mu_4 \end{bmatrix}}_{H_3} \begin{bmatrix} c_0 \\ c_1 \\ c_2 \end{bmatrix} = - \begin{bmatrix} \mu_3 \\ \mu_4 \\ \mu_5 \end{bmatrix} \quad (1)$$

H_3 (ハンケル行列)

- $\varphi(z)$ の根は次の行列の固有値に等しいことに着目

$$C_3 := \begin{bmatrix} 0 & 0 & -c_0 \\ 1 & 0 & -c_1 \\ 0 & 1 & -c_2 \end{bmatrix} \quad (\text{コンパニオン行列})$$

- $\det(C_3 - \lambda I) = 0 \Leftrightarrow \det(\underbrace{H_3 C_3}_{(1) \text{より}} - \lambda \underbrace{H_3}) = 0$

(1)より

$$\begin{bmatrix} \mu_1 & \mu_2 & \mu_3 \\ \mu_2 & \mu_3 & \mu_4 \\ \mu_3 & \mu_4 & \mu_5 \end{bmatrix} \mathbf{x} = \lambda \begin{bmatrix} \mu_0 & \mu_1 & \mu_2 \\ \mu_1 & \mu_2 & \mu_3 \\ \mu_2 & \mu_3 & \mu_4 \end{bmatrix} \mathbf{x}$$

の一般化固有値問題の固有値を求めればよい

Sakurai-Sugiura法

今回の例では

- u, v を与える (例: 乱数ベクトル)
- $\mu_k := \oint_{\Gamma} z^k u^H (zI - A)^{-1} v \, dz \quad (k=1,2,3,4,5)$
- 一般化固有値問題の固有値を求める.

$$\begin{bmatrix} \mu_1 & \mu_2 & \mu_3 \\ \mu_2 & \mu_3 & \mu_4 \\ \mu_3 & \mu_4 & \mu_5 \end{bmatrix} x = \lambda \begin{bmatrix} \mu_0 & \mu_1 & \mu_2 \\ \mu_1 & \mu_2 & \mu_3 \\ \mu_2 & \mu_3 & \mu_4 \end{bmatrix} x$$

固有ベクトルの求め方は文献[1]をご覧ください.

参考文献

[1] T. Sakurai and H. Sugiura,

A projection method for generalized eigenvalue problem using numerical integration, J. Comput. Appl. Math. Vol. 159, pp. 119-128 (2003).

[2] 櫻井鉄也, 大規模一般化固有値問題の並列解法,
応用数理, 13(4), 2003, pp. 28-37.

展望

線形計算（大規模疎行列）

- 共役勾配法（線形方程式）も Lanczos法・Arnoldi法（固有値問題）も提案から半世紀以上が経過しているが、現在でも揺るぎない重要な解法で宝石のような輝きを放つ→原石であったが、非常に磨きがかかった。
- 現代的視点で未来を見るのも重要だが過去を視るのも重要ではないか？ 原石の再発見・再評価（温故知新）

分野間融合に向けて

- 数学記号の使い方および互いの専門用語の共有が必要では？
例：エルミート転置，複素共役，内積とその具体的な定義など
- 物理側が数理側に歩み寄っていただいている現状を踏まえ、数理側も物理側に歩み寄る必要があるかもしれません。
（すでに物理側に歩み寄られている数理側の研究者はいると思います。）