前処理付固有値解法の誤差評価

西　田　　　晃[†1]

大規模疎行列の固有値を数値的に求める場合, Krylov 部分空間法は最も有力な手法として知られている．Krylov 部分空間法には複数の手法があり, 一部の解法は適切な前処理手法と組み合わせることによってより高速に固有値を計算することができるが, 前処理の効果, 計算精度に与える影響などについては未解明な点が多い. 本稿では, 前処理付固有値解法の計算誤差について，解析解による検証を行い，その性能を評価する.

Error Analysis of Preconditioned Eigensolvers

Akira Nishida [†1]

When we need to compute some eigenvalues of a large sparse matrix numerically, the Krylov subspace method is known to be the optimal choice. Combining appropriate preconditioners with the some Krylov subspace method, you can compute the eigensolutions more efficiently, although there are many unsolved problems on the preconditioners. This study verifies the numerical error of preconditioned eigensolvers by examining real variation problems, and evaluate their performance.

1. 背　　景

大規模疎行列の固有値を数値的に求める場合, いくつかの解法を考えることができ, 冪乗法, 逆反復法, Lanczos 法, Davidson 法, 共役勾配法等の Krylov 部分空間法が提案されている．多くの問題では, 特定の範囲にある数個の固有値 (及び固有ベクトル) を求めればよいため, 疎行列性を保存するこれらの解法は, 微分方程式の離散化に伴う大規模計算などにおいて広く利用されている.

本研究では, 平成 14-19 年度科学技術振興機構 CREST 事業の一環として, 反復解法ライブラリ Lis [⋆1] を開発, 配布し, 様々な並列計算機上で大規模な線型方程式を解くための環境を提供している．また平成 20 年度には九州大学情報基盤研究開発センターにおいて固有値解法の実装を行い, 同年 11 月より疎行列固有値解法に対応した新版を公開している．本稿では，前処理付固有値解法の計算誤差について，解析解による検証を行い，その性能を評価する.

2. Krylov 部分空間法

本稿では，標準固有値問題

$$Ax = \lambda x \tag{1}$$

を Krylov 部分空間法を用いて解くことを考える.

Krylov 部分空間法のうち最も単純なものとして, 冪乗法, 及びその拡張である部分空間反復法，逆反復法等などを挙げることができる．冪乗法では, サイズ n の大規模行列 A の固有値を絶対値最大のものから[⋆2] 求めるため, これを次元 $\nu \ll n$ のベクトル部分空間 G_l 上への直交射影に関する固有値問題に近似する．射影を表す行列を π_l とすれば, 問題は G_l において

$$\pi_l(Ax_l - \lambda_l x_l) = 0 \tag{2}$$

を満たす近似固有対

$$\lambda_l \in C, \quad 0 \neq x_l \in G_l \tag{3}$$

の計算に帰着される．部分空間 S を生成する r 個の独立なベクトルを

$$U = [u_1, ..., u_r], \quad 1 \leq r < n \tag{4}$$

とすると, G_l 内の正規直交基底の選び方によって,

$$S_l = A^l S, \quad l = 1, 2, ... \tag{5}$$

とすれば $r = 1$ の場合は冪乗法, また $r > 1$ ならば部分空間反復法となる．同様にして, 逆反復法は A ではなく, A^{-1} をベクトルに適用することにより, A の固有値を絶対値最小のものから求める.

†1 九州大学情報基盤研究開発センター
Research Institute for Information Technology, Kyushu University

⋆1 http://www.ssisc.org/lis/
⋆2 以下単に最大固有値と書く.

一方，実対称行列 A に関する固有値問題

$$Ax = \lambda x \tag{6}$$

の最小固有値, またはこれと同値な問題

$$x = \mu Ax, \quad \mu = 1/\lambda \tag{7}$$

の最大固有値の計算を

Rayleigh 商

$$\mu(x) = \frac{x^T x}{x^T Ax} \tag{8}$$

の極値問題に帰着すると, 最急勾配方向が

$$\nabla\mu(x) \equiv g(x) = \frac{2(x - \mu Ax)}{x^T Ax} \tag{9}$$

であることから, 適当な係数 α_i, 修正方向

$$p_i = -g_i + \beta_{i-1}p_{i-1}, \ \beta_{i-1} = \frac{g_i^T g_i}{g_{i-1}^T g_{i-1}} \tag{10}$$

などを用いることにより, 非線型共役勾配法を導くことができる[1)–3)].

3. 固有値解法における前処理

固有値 λ が既知であると仮定すると, これに対応する固有ベクトルは

$$(A - \lambda)x = 0, \quad x \neq 0 \tag{11}$$

を解くことにより求める．すなわち, 固有値解法における理想的な前処理行列 T は $A - \lambda$ の逆行列であり, 実際には未知の λ を Ritz 値で置き換えた前処理行列を考えることもできる．一般にこのように前処理行列を取ると不定値となるため, T が対称正定値でなければならない場合には

$$T \approx A^{-1} \tag{12}$$

と取る．特に連立一次方程式 $Ax = f$ の解法が与えられている場合には, 前処理の計算も容易である．このように定めた T を固有値解法中の反復ベクトル r に対して

$$w^{(i)} = Tr \tag{13}$$

すなわち

$$T^{-1}w^{(i)} = r \tag{14}$$

を解くことにより，収束を加速するのが, 固有値解法における前処理の位置付けである[4)–6)].

前処理付固有値解法のアルゴリズムは, 前処理行列 $T \approx A^{-1}$, TA に関する m_k 次多項式 $P_{m_k}(TA)$ を用いて以下のように書くことができる.

(1) 初期ベクトル $x^{(0)}$ を選択する

(2) m_k 回の反復により $x^{(k)} = P_{m_k}(TA)x^{(0)}$ を計算する

(3) $\mu^{(k)} = (x^{(k)}, x^{(k)})/(x^{(k)}, Ax^{(k)})$ を計算する

前処理を共役勾配法に適用する場合，その反復は, 適当な初期ベクトル $x^{(0)}$ と対応する修正ベクトル $p^{(0)} = 0$ を用いて,

$$\mu^{(i)} = (x^{(i)}, x^{(i)})/(x^{(i)}, Ax^{(i)}) \tag{15}$$

$$r = x^{(i)} - \mu^{(i)}Ax^{(i)} \tag{16}$$

$$w^{(i)} = Tr \tag{17}$$

$$x^{(i+1)} = w^{(i)} + \tau^{(i)}x^{(i)} + \gamma^{(i)}p^{(i)} \tag{18}$$

$$p^{(i+1)} = w^{(i)} + \gamma^{(i)}p^{(i)} \tag{19}$$

と書くことができる．ここでは行列束 $x^{(i)} - \mu^{(i)}Ax^{(i)}$ に関する $\mathrm{span}\{w, x^{(i)}, p^{(i)}\}$ 上の Ritz 値, Ritz ベクトルを Rayleigh-Ritz 法を用いて計算し, 最大 Ritz 値に対応する Ritz ベクトルを $x^{(i+1)}$ とする．すなわち, 係数 $\tau^{(i)}$, $\gamma^{(i)}$ の値は, $\mathrm{span}\{w, x^{(i)}, p^{(i)}\}$ 上での局所的な最適解をもとに決められる．これによって, ベクトル間の直交性をもとに各係数を明に計算する必要のある従来の方法と比較して, 少ない計算量で更新値を求めることができる.

この方法で用いている前処理は, $T \approx A^{-1}$ をベクトルに適用している点で, 実際には逆反復法に近い意味を持っている．ただし, 近似逆行列を用いていることから, 精度に関しては問題が生じる可能性がある．本稿ではこの点について調べる.

4. Lis を用いた実装

既存の線型計算ライブラリはいくつかあるが, 本研究では, 以前より開発を進めている Lis に, 非線型共役勾配法の実装を行った.

海外においても，Valencia 工科大学による SLEPc (Argonne 米国立研究所の並列反復

解法ライブラリ PETSc ⋆1 をベースとして開発されている) ⋆2 や, Colorado 大学による BLOPEX (Lawrence Berkeley 米国立研究所による並列反復解法ライブラリ Hypre ⋆3 をベースとしている) ⋆4 などのライブラリに, 疎行列向けの固有値解法が実装されている[7].

これらのライブラリでは，本研究の実装と同様に, いずれもオブジェクト指向に基づいた設計を行っており, 並列化はライブラリのレベルで実現されている．また, すべてのデータはAPI を用いて処理されている．すなわち, 行列, ベクトルデータ等の生成・廃棄, 及びこれらのオブジェクトに対する操作は, それぞれの操作を記述する API を呼び出すことにより処理される．各前処理手法はそれぞれ線型解法として実装され, 必要に応じて前処理として利用することができるようになっており, これらを組み合わせて評価することができる．

本研究では, Lis 上に，共役勾配法を含む Krylov 部分空間法を実装し, 評価を行った．なお，逆反復法等の内部処理においても, A^{-1} を反復ベクトルに適用する際に線型方程式を解く必要があるため, 多様な解法を実装した反復解法ライブラリの利用は有効である．表1, 2, 3 に Lis が対応する固有値解法, 線型方程式解法, 前処理手法を示す．Lis ではこれらを自由に組み合わせて新たに固有値解法を構成することができる．

表 1　Lis で利用可能な固有値解法

Power Iteration
Inverse Iteration
Approximate Inverse Iteration
Conjugate Gradient
Lanczos Iteration
Subspace Iteration
Conjugate Residual

実装した解法は, Lis の固有値計算アルゴリズムとして収録され, 要素演算と線型方程式解法, 前処理に関するライブラリを必要に応じて使用している．反復解法部は行列等のデータに関する要素演算として定義された Lis の API を用いて記述されており, MPI を用いた並列化はこのレベルで行われている．

⋆1 http://www-unix.mcs.anl.gov/petsc/
⋆2 http://www.grycap.upv.es/slepc/
⋆3 http://computation.llnl.gov/casc/hypre/
⋆4 http://www-math.cudenver.edu/ aknyazev/software/BLOPEX/

表 2　Lis で利用可能な線型方程式解法

CG	CR
BiCG	BiCR
CGS	CRS
BiCGSTAB	BiCRSTAB
GPBiCG	GPBiCR
BiCGSafe	BiCRSafe
BiCGSTAB(l)	BiCRSTAB(l)
Jacobi	Gauss-Seidel
SOR	Orthomin(m)
TFQMR	MINRES
GMRES(m)	FGMRES(m)
IDR(s)	

表 3　Lis で利用可能な前処理

Jacobi	SSOR
ILU(k)	ILUT
Crout ILU	I+S
SA-AMG	Hybrid
SAINV	Additive Schwarz
ユーザ定義前処理	

Lis に実装されている前処理手法について, 主なものを簡単に述べる．

- 対角スケーリング・Jacobi 前処理
  これらは前処理としては単純な方法であるが, スケーラビリティに関して良好な性能を示す．
- SSOR・Hybrid 前処理
- 不完全 LU 分解前処理
  ILU 前処理のスケーラブルな実装．問題サイズがプロセッサ数に比例する場合に，ほぼ一定の計算時間で処理することができる．ILU(k), ILUT, Crout ILU が実装されている．
- SA-AMG[8] 前処理
  代数的マルチグリッド法の逐次・並列実装．Lis では前処理の一つとして実装され，クリロフ部分空間法の前処理として用いることができる．
- 近似逆行列前処理

- Additive Schwarz 前処理

5. 性能評価

Lis 上に実装した固有値解法の特性について調べるため, 九州大学情報基盤研究開発センターに設置された Xeon 5570 サーバ (2.93 GHz クアッドコアプロセッサ ×2) を用いて評価を行った．なお，2009年9月に公開した Lis 1.2.15 以降のバージョンでは，前処理行列生成部分を線型方程式解法部分から分離し，逆反復法のように内部で繰り返し線型方程式を解く場合には，最初に一度だけ前処理行列を生成するよう仕様を変更している．このため，前処理に時間を要する AMG 等で，以前より評価値が改善している．

今回は, Lis 上に実装した疎行列ベクトル積に関するベンチマークプログラム `spmvtest` での評価結果をもとに，行列格納形式として CRS を選択し，また，AMG 前処理の並列化手法が MPI にのみ対応していることから，MPI 並列版を用いて評価を行った．また,[9] において行った同様の評価結果に関しても，7-9 に修正点を反映した．

ここでは，矩形領域 $[0, l] \times [0, m]$ の長方形膜の自由振動を表す2次元 Laplace 作用素を5点中心差分により離散化した行列の最小固有対 (最小固有値と対応する固有ベクトル) を Lis 上に実装した固有値解法により計算した．この場合，固有値の解析解は

$$\pi^2 \left(\frac{\sigma^2}{l^2} + \frac{\tau^2}{m^2} \right), \ \sigma, \tau \in N \tag{20}$$

で与えられるので，各問題サイズでの最小固有値の近似値は，表4に示す値を取る．ここでは，相対残差ノルムの閾値を 10^{-5} とし, 非零要素数がプロセッサ数に比例するよう問題サイズを取って各解法の性能を測定した．結果を表4-5 に示す. 表から分かるように，逆反復法では解析解に対応する最小固有対が得られているのに対し，共役勾配法では最小固有値以外の固有対が計算されている．また，AMG 前処理を用いた場合の解が最も解析解に近いのに対して，ILU(0) では十分な精度が得られず，フィルインを増やした場合に，精度が改善していることが分かる．

共役勾配法での前処理と精度の関係について調べるため，逆反復法を用いて，問題行列の代わりに前処理行列を作用させた場合 (以降近似逆反復法と呼ぶ) の解についても計算した．結果を表6に示す．これから，近似逆反復法の結果は共役勾配法とほぼ一致することが分かる．

表4 局所 ILU(0), AMG, Jacobi 前処理付共役勾配法を線型方程式解法に使用した逆反復法による最小固有対の計算結果

Problem Size	40,000	80,000	160,000	320,000
# cores	1	2	4	8
time (s) (ILU(0)CG)	2.59	9.23	16.3	62.3
# iter (ILU(0)CG)	8	12	8	12
eigenvalue (ILU(0)CG)	4.89e-4	3.06e-4	1.23e-4	7.68e-5
time (s) (AMGCG)	1.23	1.59	1.46	2.64
# iter (AMGCG)	8	12	8	12
eigenvalue (AMGCG)	4.89e-4	3.06e-4	1.23e-4	7.68e-4
time (s) (JCG)	1.39	3.37	4.39	20.8
# iter (JCG)	8	12	8	12
eigenvalue (JCG)	4.89e-4	3.06e-4	1.23e-4	7.68e-5
eigenvalue (analytic)	4.934e-4	3.084e-4	1.234e-4	7.711e-5

表5 局所 ILU(0), 局所 ILU(64), AMG 前処理付共役勾配法による最小固有対の計算結果

Problem Size	40,000	80,000	160,000	320,000
# cores	1	2	4	8
time (s) (ILU(0)CG)	6.12	12.3	17.3	21.0
# iter (ILU(0)CG)	1926	3487	2554	2002
eigenvalue (ILU(0)CG)	5.87e-1	5.87e-1	5.87e-1	5.87e-1
time (s) (ILU(64)CG)	5.96	9.50	16.3	21.8
# iter (ILU(64)CG)	11	14	27	26
eigenvalue (ILU(64)CG)	5.86e-4	5.86e-4	1.12e-3	1.12e-3
time (s) (AMGCG)	0.558	0.632	0.722	1.16
# iter (AMGCG)	11	18	12	27
eigenvalue (AMGCG)	8.97e-4	5.64e-4	2.70e-4	2.08e-4

表 6　局所 ILU(0), 局所 ILU(64), AMG 前処理行列を用いた近似逆反復法による最小固有対の計算結果

Problem Size	40,000	80,000	160,000	320,000
# cores	1	2	4	8
time (s) (ILU(0)II)	2.56	5.25	9.80	13.2
# iter (ILU(0)II)	2098	3799	2782	2181
eigenvalue (ILU(0)II)	5.86e-1	5.86e-1	5.86e-1	5.86e-1
time (s) (ILU(64)II)	5.68	9.66	11.7	14.5
# iter (ILU(64)II)	10	28	26	25
eigenvalue (ILU(64)II)	5.86e-4	1.12e-3	1.12e-3	1.12e-3
time (s) (AMGII)	0.41	0.47	0.54	0.75
# iter (AMGII)	10	18	11	26
eigenvalue (AMGII)	8.97e-4	5.64e-4	2.70e-4	2.08e-4

6. ま と め

解析解を持つ問題に対して, 逆反復法, 共役勾配法で得られる最小固有値は一致しなかった．実際には, 近似逆反復法 (Approximate Inverse Iteration, すなわち逆反復法において, A^{-1} の代わりに $T \approx A^{-1}$ を適用する手法) での結果と一致する．これは, 共役勾配法における前処理の効果が, 逆反復法に類似したものであることによることを示している．共役勾配法は逆反復法と比較して短時間で解が求まっているものの，精度に関しては注意が必要である．

本稿では, 固有値解法への前処理の適用について, 本研究でこれまでに得られている知見を報告した．本解法の収束特性等に関しては, 未解明な部分も残っている．今後大規模問題を対象として, 各解法の特性を明らかにしていきたい．

参 考 文 献

1) Fletcher, R. and Reeves, C.M.: Function minimization by conjugate gradients, *Comp. J.*, Vol.7, pp.149–154 (1964).
2) Bradbury, W.W. and Fletcher, R.: New Iterative Method for Solution of the Eigenproblem, *Numer. Math.*, Vol.9, pp.259–267 (1966).
3) Hestenes, M.R. and Karush, W.: A method of gradients for the caluculation of the characteristic roots and vectors of a real symmetric matrix, *J. Res. Natl. Inst. Stand. Technol.*, Vol.47, pp.45–61 (1951).
4) Knyazev, A.V.: Preconditioned eigensolvers—an oxymoron?, *Electron. Trans. Nu-*

表 7　Xeon 5570 サーバ上での疎行列ベクトル積 (MPI, スレッド並列) の性能

Problem Size	40,000	80,000	160,000	320,000
# cores	1	2	4	8
CRS (MFLOPS)	1110	2190	4140	4440
DIA (MFLOPS)	1630	3190	5010	4370
ELL (MFLOPS)	1210	2400	4270	3770
JDS (MFLOPS)	1030	2030	3370	3040
Problem Size	40,000	80,000	160,000	320,000
# cores	1	2	4	8
CRS (MFLOPS)	1190	2380	2160	3780
DIA (MFLOPS)	1640	3130	6040	4940
ELL (MFLOPS)	1250	2440	4350	3560
JDS (MFLOPS)	1020	2030	3070	2860

表 8　SR16000 上での疎行列ベクトル積 (MPI, スレッド並列) の性能

Problem Size	40,000	80,000	160,000	320,000	640,000	1,280,000
# cores	1	2	4	8	16	32
CRS (MFLOPS)	580	851	1680	3270	6430	13400
DIA (MFLOPS)	2130	2370	4000	7730	14900	30600
ELL (MFLOPS)	1790	1750	2440	4860	9490	19100
JDS (MFLOPS)	1390	1330	2020	4020	7960	16200
# cores	1	2	4	8	16	32
CRS (MFLOPS)	580	1090	2210	4390	8740	15100
DIA (MFLOPS)	2130	3930	7500	15000	28700	44100
ELL (MFLOPS)	1790	2930	5900	11500	23000	33800
JDS (MFLOPS)	1390	2250	4550	9060	17600	27300

表 9　SX-9 上での疎行列ベクトル積 (MPI, スレッド並列) の性能

Problem Size	40,000	80,000	160,000	320,000	640,000
# cores	1	2	4	8	16
CRS (MFLOPS)	13.2	26.5	52.7	106	-
DIA (MFLOPS)	10400	12100	16000	29600	-
ELL (MFLOPS)	1310	2410	4450	8720	-
JDS (MFLOPS)	1070	1890	3460	6340	-
# cores	1	2	4	8	16
CRS (MFLOPS)	13.4	27.0	53.6	-	212
DIA (MFLOPS)	11700	15500	28300	-	100000
ELL (MFLOPS)	1310	2530	4900	-	19500
JDS (MFLOPS)	1080	2050	4090	-	16000

mer. Anal., Vol.7, pp.104–123 (electronic) (1998). Large scale eigenvalue problems (Argonne, IL, 1997).

5) Knyazev, A.V.: Toward the Optimal Preconditioned Eigensolver: Locally Optimal Block Preconditioned Conjugate Gradient Method, *SIAM J. Sci. Comput.*, Vol.23, No.2, pp.517–541 (2001).

6) Arbenz, P. and Lehoucq, R.: A comparison of algorithms for modal analysis in the absense of a sparse direct method, Technical Report SAND2003-1028J, Sandia National Laboratories (2003).

7) Dongarra, J.: Freely Available Software for the Solution of Linear Algebra Problems, Technical Report http://www.netlib.org/utk/people/JackDongarra/la-sw.html, Innovative Computing Laboratory, University of Tennessee (2009).

8) Fujii, A., Nishida, A. and Oyanagi, Y.: Evaluation of Parallel Aggregate Creation Orders : Smoothed Aggregation Algebraic Multigrid Method, *High Performance Computational Science A*, pp.99–122 (2005).

9) 西田晃：並列非線型共役勾配法アルゴリズムとその性能評価，情処研報，Vol.2009-HPC-121, No.36, pp.1–6 (2009).