疎で悪条件な非対称行列のコレクションサイト

Sparse Matrix Collection for Ill-conditioned Non-symmetric Matrices

南畑 淳史 *
Atsushi MINAMIHATA

抄　録

数値線形代数のアルゴリズムにおいて理論的な挙動の解析および実際の行列での挙動の解析は重要である。また，実アプリケーションに近い問題での挙動が重要視されてきており，スーパーコンピュータの分野でもHPCGと呼ばれる疎行列を係数行列とする連立一次方程式のベンチマークが導入された。本論文では疎で悪条件な非対称行列の作成法とその行列のコレクションサイトを提案する。コレクションサイトはHTML，CSSおよびJavaScriptで構成されている。また，行列はMATファイルで公開されているために，MATLABやPythonでの数値実験で簡単に使うことができる。MATLABとPythonを用いた数値実験を通じて，疎で悪条件な非対称行列の応用先について紹介する。

1.　はじめに

数値線形代数のアルゴリズムにおいて理論的な挙動の解析および実際の行列での挙動の解析は重要である。例えば，連立一次方程式の代表的な解法であるCG法において，理論的は行列サイズと同じ回数の反復で真の解にたどり着くことが示されている。しかし，実際の行列では真の解にたどり着くわけではなく，丸め誤差の影響で理論的な挙動から予想される結果とは異なる。また，丸め誤差も行列サイズ，疎行列のパターンや条件数に依存する。従って，実際の行列での挙動の解析をする場合，行列サイズと疎行列のパターンを指定し，いくつかの条件数でアルゴリズムの挙動を解析することが望ましい。

実際の行列を公開しているサイトとして SuiteSparse Matrix Collection[1)]や ELSES matrix library[2)]などがある。特に，SuiteSparse Matrix Collectionは実際のアプリケーションから現れる行列を収集し，行列とその性質をWebサイトに提示しており，疎行列向けの数値線形代数のアルゴリズムのテスト行列として広く使われている。

行列サイズと疎行列のパターンを指定して行列を得る方法として MATLAB の sprandsym 関数がある。sprandsym関数は対称行列に限って，条件数と疎行列のパターンを指定できる。しかしながら，

＊　中央大学理工学部~~情報工学科~~　助教

非対称行列には対応していない。また，MATLAB には sprand 関数があるが，こちらは条件数を指定することはできるが，疎行列のパターンは指定できず，非ゼロ要素の割合を指定する。また，密行列の条件数を指定した行列の作成法として，Higham[3]による行列の作成法が広く用いられている。この方法は MATLAB の gallery 関数に実装されているが，スパース行列には対応していない。より悪条件な行列の作成法として，Rump[4]による行列の作成法が知られているが，こちらもスパース行列には対応していない。本論文では，行列サイズ，疎行列のパターンと条件数を指定して，非対称行列を作成する方法を提案する。

2. 疎行列のパターンと条件数を指定した非対称行列の作成法

この章では疎行列のパターンと条件数を指定した非対称行列の作成法を示す。次に作成した行列がどのような特徴を持つのかを述べる。提案する非対称行列の作成法には LU 分解を用いる。行列Aを$A \in \mathbb{R}^{n \times n}$として，LU 分解は行列$A = LU$と単位下三角行列と上三角行列に分解する方法である。Uの(n, n)成分を 0 と置いた LU 分解は行列Aを摂動させ，ランクを n-1 とした行列と考えることができる。また，Uの(n, n)成分を非常に小さい数εとすれば，悪条件な行列になることが期待できる。従って，εをうまく制御すれば，所望の条件数に近づけることができる。ただし，Aは疎行列としているためにAの(n, n)成分が 0 であった場合には，この操作により値が入ってしまうために疎行列のパターンが崩れてしまう。そのため，行列Aと同じ疎行列のパターンを持つランダムな行列を用いて，LU 分解のピボッティングにてUの摂動に影響がない行列を探し出す。例えば，Algorithm 1.に示すように失敗した場合は乱数行列を作り直し，εを条件数の推定により調整するアルゴリズムであれば，行列依存ではあるが，悪条件行列を作ることができる。

3. 疎で悪条件な非対称行列のコレクションサイト

数値線形代数のアルゴリズムの数値実験において結果の再現は重要である。結果の再現にはアルゴリズムの実装だけでなく，使用した行列も必要となる。従って，行列の作成法は行列を再現できる機構が組み込まれているか，行列が公開されている必要がある。本手法は行列を再現できる機構が組み込まれていないために，行列~~が~~を Web サイト(https://ill-conditioned.sakura.ne.jp/)にて公開をする。

```
function X=gIllconditionedMatrix(A)
    ind_A = find(A);
    n=length(A);
    ind_flag=0;
    u_v = 2^-30;
    cond_v=Inf;

    while (~(cond_v > 10^13 && cond_v <10^15 && ind_flag==1))
        X=sparse(n,n);X(ind_A)=randn(length(ind_A),1);%乱数行列を作る
        [L,U,p,q]=lu(X,'vector');%LU分解
        U(end,end)=max(abs(diag(U)))*u_v;%epsilonの設定
        X(p(end),q(end))=L(end,:)*U(:,end);
        ind_X =find(X);

        if isequal(ind_A,ind_X)
            ind_flag=1;
            cond_v = condest(X)
            if(cond_v >10^15)
                u_v = u_v*2^5;
            elseif(cond_v <10^13)
                u_v = u_v*2^-5;
            end
        else
            ind_flag=0;
        end

    end
end
```

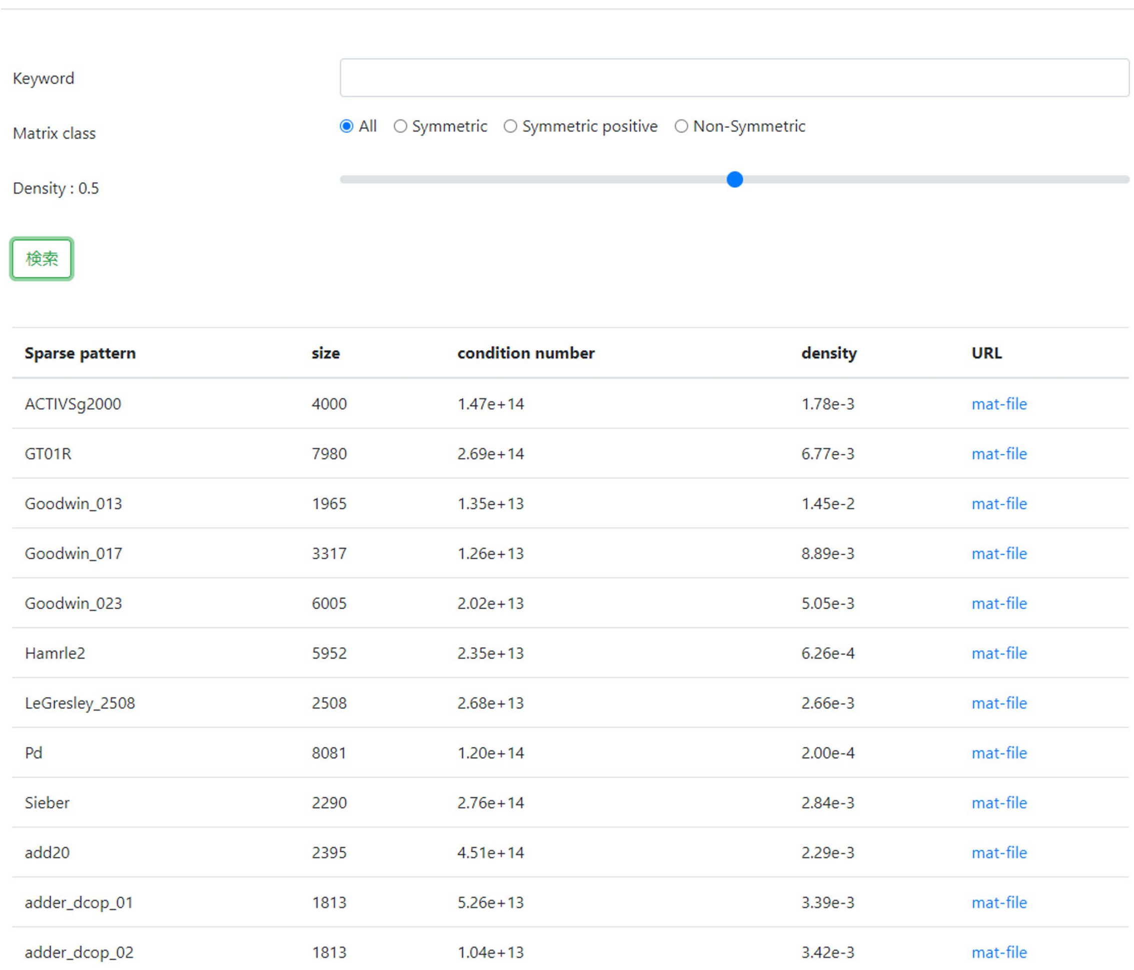
Algorithm 1 LU 分解を用いた悪条件行列の作成アルゴリズム

本 Web サイトは HTML, CSS および JavaScript で構成されている。具体的には CSS のフレームワークには Boos~~t~~trap を用い, JavaScript のフレームワークには Vue.js を用いた。また各悪条件行列は Table 1. のように疎行列のパターン, 行列サイズ, 条件数, density および URL を持ち, JSON~~json~~ファイルによってこれらを管理している。

行列ファイルのフォーマットは MATLAB の MAT ファイルを用いている。また，本 Web サイトのインターフェースを Fig.　1. に示す。本 Web サイトは検索機能があり，Keyword 検索，Matrix　class による検索，density による検索を組み合わせることで悪条件行列を探すことができる。

Table 1 公開している悪条件行列の一例

Sparse pattern	size	condition number	density	URL
ACTIVSg2000	4000	2.58e+13	1.78e-3	/mat/ACTIVSg2000.mat
Hamrle1	32	6.02e+13	9.57e-2	/mat/Hamrle1.mat
LeGresley_2508	2508	5.24e+13	2.66e-3	/mat/LeGresley_2508.mat
Sieber	2290	1.87e+14	2.84e-3	/mat/Sieber.mat
add20	2395	4.18e+13	2.29e-3	/mat/add20.mat



Sparse pattern	size	condition number	density	URL
ACTIVSg2000	4000	1.47e+14	1.78e-3	mat-file
GT01R	7980	2.69e+14	6.77e-3	mat-file
Goodwin_013	1965	1.35e+13	1.45e-2	mat-file
Goodwin_017	3317	1.26e+13	8.89e-3	mat-file
Goodwin_023	6005	2.02e+13	5.05e-3	mat-file
Hamrle2	5952	2.35e+13	6.26e-4	mat-file
LeGresley_2508	2508	2.68e+13	2.66e-3	mat-file
Pd	8081	1.20e+14	2.00e-4	mat-file
Sieber	2290	2.76e+14	2.84e-3	mat-file
add20	2395	4.51e+14	2.29e-3	mat-file
adder_dcop_01	1813	5.26e+13	3.39e-3	mat-file
adder_dcop_02	1813	1.04e+13	3.42e-3	mat-file

Fig． 1 疎で悪条件な非対称行列のコレクションサイトのインターフェース

4.　悪条件行列の作成結果とその応用

この章では提案手法である疎行列のパターンと条件数を指定した非対称行列の作成法の成功率を紹介する。SuiteSparse Matrix Collection の行列サイズが 1 万以下の非対称の疎行列のパターンを用意し，提案手法で行列を作成した。その作成した総数と作成された悪条件の総数を Table 2. に示す。

Table 2 SuiteSparse Matrix Collection の行列サイズが 1 万以下の疎行列の数と作成できた悪条件行列の総数

1 万以下の疎行列のパターンの総数	526
作成できた悪条件行列の総数	294

次に，提案手法で作成された 294 個の行列中ですべての対角成分に値を持つ 86 個の行列が対角スケーリングでどのように条件数が変化をするかを示す。スケーリングが悪い行列は対角スケーリングで条件数が大幅に下がることがある。MATLAB の sprandsym 関数を用いて悪条件を作成した場合はスケーリングの悪い行列が作成されることがある。例えば，以下のように MATLAB で以下のように行列 A を与え，sprandsym 関数を用いて行列 S を作成した場合，スケーリングの悪い行列が作成された。

```
>> A
A =
    1.0000         0         0   -0.0418   -0.0925
         0    1.5184         0   -1.7054    0.2721
         0         0    1.0000         0    1.1737
   -0.0418   -1.7054         0    1.0000   -0.7876
   -0.0925    0.2721    1.1737   -0.7876    1.0000
>> S=full(sprandsym(A,[],10^-13,3))
S =
    0.0001         0         0    0.0000    0.0001
         0    0.0000         0    0.0000    0.0000
         0         0    0.0000         0    0.0000
    0.0000    0.0000         0    0.4600    0.0000
    0.0001    0.0000    0.0000    0.0000    0.0002
```

数値計算においてスケーリングの悪い行列は対角スケーリングを行えば条件数が大きく改善され

ることもあるためにテスト問題としては好ましくない。上記の例でも対角スケーリングにより大幅に条件数が変わってしまう。

```
>> cond(S)
ans =
   3.4927e+10
>> cond(diag(1./diag(S))*S)
ans =
   6.1896e+05
```

提案手法を用いて作成した悪条件行列のうちすべての対角成分に値を持つ 86 個のスケーリング後の条件数と元の条件数の比率を Table 3. に示す。最小値が 0.05 であるから最悪の場合でも対角スケーリングで条件数が 2 乗しか落ちないことがわかる。また，中央値も 7.35 であり，対角スケーリングでは条件数が変わっていないことが分かる。

Table 3 対角スケーリング後の条件数と元の条件数の比率

対角スケーリング後の条件数と元の条件数の比率の中央値	7.35
対角スケーリング後の条件数と元の条件数の比率の最小値	0.05

また，数値実験を行う上では解が分かる問題であれば詳しく検討ができる。ここでは，Ozaki-Ogita[5)]による連立一次方程式の解が分かる行列の生成のアルゴリズムを用いて解が分かる行列を作成する。このとき，提案手法で作成した行列を用いることで悪条件かつ解が分かる行列を作成できることを示す。ただし，そのような行列を作れるかどうかは行列依存である。今回は ACTIVSg2000 のパターンを持つ行列(https://ill-conditioned.sakura.ne.jp/mat/ACTIVSg2000.mat)を用いて，Ozaki-Ogita の論文 [5)]の generate_ones 関数を用いて解の分かる行列を作成した。ただし，密行列ではなく疎行列を扱うので行列サイズnの比例ではなく，非ゼロ要素の値に比例するように変更した。また，出力された行列が元の行列と疎行列のパターンが同じになるかどうかは調べる必要がある。generate_ones 関数は解がすべて 1 となるように行列を修正する関数である。ACTIVSg2000 のパターンを持つ行列をBと表記し，Ozaki-Ogita の generate_ones 関数を用いて連立一次方程式の解が分かる行列とベクトルをそれぞれCとbと表記し，作成された行列Cの性質を Table 4. に示す。

Table 4 ACTIVSg2000 のパターンを持つ行列から作成された行列$\boldsymbol{C}$の性質

$\|\|C-B\|\|_\infty$	4.90×10^{-13}
Cの条件数	1.47×10^{14}

次にPython3.7とSciP~~p~~yを用いてACTIVSg2000のパターンを持つ行列をBを元にgenerate ones関数を用いて作られた連立一次方程式$Cx = b$を解いた場合の精度についての実験を紹介する。連立一次方程式の解を求める関数はSciP~~p~~yのspsolve関数を用いた。spsolve関数で求めた近似解と真の解(すべてが 1)の差の絶対値をerrとし，Fig 2.に真の解xのインデックス毎の精度を示す。Fig 2.より精度の良いindexとそうではないindexがあるのが分かる。また，spsolve関数で求めた近似解は今回の問題では多くの場合で8桁程度は正しいことがわかる。このような実験を連立一次方程式のアルゴリズム毎に評価すればアルゴリズムの精度比較を定量的に行うことができる。また，反復解法を用いる場合は反復毎に解がどのような精度になっているのか細かく検討することができる。

5. まとめと今後

疎行列のパターンと条件数を指定した非対称行列の作成法を提案した。SuiteSparse Matrix Collectionの行列を用いた結果，約56%の確率で悪条件の問題が作成できた。提案手法は1つの値を除いて，作成元となる行列の値と同じになる性質を持つ。作成元の行列を乱数で作れば，乱数に近い特定の疎行列のパターンを持つ悪条件行列が作成できる。また，Ozaki-Ogita による連立一次方程式の解が分かる行列の生成のための行列としても使うことができることがあり，連立一次方程式の解が分かる特定のスパースパターンを持つ悪条件行列の作成にも応用ができる。ただし，提案した非対称行列の作成法および連立一次方程式の解が分かる問題への応用も非正則な行列を除く任意のパターンで行列の作成できたわけではない。そのため，行列の作成率を向上させる研究や多くの疎行列のパターンに対する悪条件行列の作成とWebでの公開が必要となる。

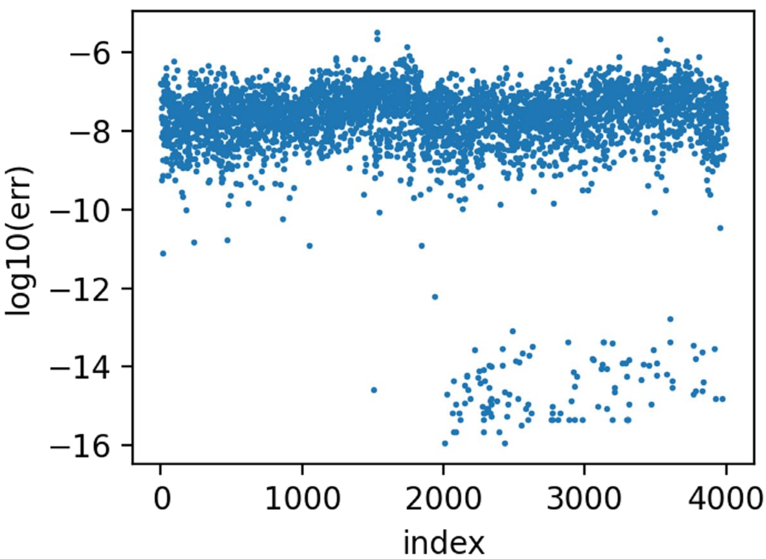


Fig . 2 近似解と真の解の絶対値の分布

参考文献

1) T. A. Davis and Y. Hu. "The University of Florida sparse matrix collection" *ACM Transactions on Mathematical Software (TOMS)* vol. 38(1), pp. 1:1-1:25, 2011.

2) T. Hoshi. "ELSES matrix library" http://www.elses.jp/matrix/,(参照 2020-11-26).

3) N. J. Higham. "Algorithm 694: A collection of test matrices in MATLAB." *ACM Transactions on Mathematical Software (TOMS)* 17.3, pp. 289-305, 1991.

4) S. M. Rump. "A Class of Arbitrarily Ill-conditioned Floating-Point Matrices" *SIAM J. Matrix Anal. Appl. (SIMAX)* 12(4), pp. 645-653, 1991.

5) K. Ozaki and T. Ogita. "Generation of Linear Systems with Specified Solutions for Numerical Experiments" *Reliable Computing* vol. 25, pp. 148-167, 2017.

本研究は若手研究 19K20281 の助成を受けたものである。

Abstract

A numerical behavior on the matrix which arises from a real application and a theoretical analysis are important in linear algebra. Recently a performance of a real application has attracted attention and HPCG benchmark which measures the time to solve a sparse linear system uses to rank the TOP500 super computing systems. In this paper, sparse matrix collection for ill-conditioned non-symmetric matrices is proposed. The matrix collection website is developed by HTML, CSS and JavaScript. The test matrices are provided as MAT-file. MATLAB and Python can use MAT-file easily. Numerical results of ill-conditioned non-symmetric matrices using MATLAB and Python are introduced.