論　考

BiCG-STAB 法による 連立一次方程式の計算関数の作成

Making of the calculation function for the coalition linear equation by the BiCG-STAB method

Author　: 志多　友史　(Yuji Shida)
Date　: 2018/6/16
Keywords : BiCG-STAB 法(Biconjugate Gradient Stabilized Method),
連立一次方程式(coalition linear equation), C 言語(C programming language)

Abstract:===

本稿では、BiCG-STAB 法（前処理無）を用いた連立一次方程式の計算関数を作成する。有限差分法・有限要素法等の数値解析では、ほとんどの問題が巨大な連立一次方程式を解く事に帰着する。従って、備忘録として一度、連立一次方程式を解くための準備をここで行う。

In this report, I made the calculation function for the coalition linear equation using the BiCG-STAB method (no preprocessing). Most problems of the numerical analysis such as the Finite Difference Methods (FDM) or Finite Element Method (FEM) result in solving a huge coalition linear equation. Therefore, I performed preparation to solve a coalition linear equation as a memorandum for the future in this report.

==

１．序論(Introduction)

計算機の性能の向上に伴って、実験・理論に次ぐ第三の研究分野である計算機科学が発展した。超高速・超並列の大型計算機により、人間には到底成しえない莫大な量の計算が短時間で行えるようになり、複雑な問題に対して、これまで知りえなかった知見が発見されている。
本稿では、計算機で行われる多種多様な演算の内、比較的実行頻度が高いと思われる連立一次方程式の演算に関して BiCG-STAB 法（前処理無）を用いた計算関数を作成する。

２．理論(Theory)

本稿では前処理を行わない BiCG-STAB 法のアルゴリズムの概要について記す。ここでは詳細な理論的背景については省略する。解くべき連立一次方程式を $A\boldsymbol{x}=\boldsymbol{b}$ として、以下に計算の流れを示す。なお"ε"は反復演算停止条件の閾値である。

（１）初期ベクトル $\boldsymbol{x}_0$ を適当に設定する。
（２）ベクトル $\boldsymbol{r}_0\left(=\boldsymbol{b}-A\boldsymbol{x}_0\right)$ を計算する。
（３）ベクトル $\boldsymbol{r}_0$ との内積が"0"とならないベクトル $\hat{\boldsymbol{r}}_0$ を設定する。
※ $\hat{\boldsymbol{r}}_0\cdot\boldsymbol{r}_0\neq 0$ を満たすのであれば $\hat{\boldsymbol{r}}_0=\boldsymbol{r}_0$ でも良い。
（４）初期パラメータ $\rho_0=\alpha=\omega_0=1,\ \boldsymbol{v}_0=\boldsymbol{p}_0=\boldsymbol{0}$ を設定する。
（５）以下の反復演算を行う。
（５．１） $\rho_i=\hat{\boldsymbol{r}}_0\cdot\boldsymbol{r}_{i-1}$
（５．２） $\beta=\frac{\rho_i}{\rho_{i-1}}\frac{\alpha}{\omega_{i-1}}$
（５．３） $\boldsymbol{p}_i=\boldsymbol{r}_{i-1}+\beta\left(\boldsymbol{p}_{i-1}-\omega_{i-1}\boldsymbol{v}_{i-1}\right)$
（５．４） $\boldsymbol{v}_i=A\boldsymbol{p}_i$
（５．５） $\alpha=\frac{\rho_i}{\hat{\boldsymbol{r}}_0\cdot\boldsymbol{v}_i}$
（５．６） $\boldsymbol{s}=\boldsymbol{r}_{i-1}-\alpha\boldsymbol{v}_i$
（５．７） $\boldsymbol{t}=A\boldsymbol{s}$
（５．８） $\omega_i=\frac{\boldsymbol{t}\cdot\boldsymbol{s}}{\boldsymbol{t}\cdot\boldsymbol{t}}$
（５．９） $\boldsymbol{x}_i=\boldsymbol{x}_{i-1}+\alpha\boldsymbol{p}_i+\omega_i\boldsymbol{s}$
（５．１０） $\boldsymbol{r}_i=\boldsymbol{s}-\omega_i\boldsymbol{t}$
（５．１１） $\|\boldsymbol{r}_i\|<\epsilon$ が満たされれば反復演算終了。

３．方法(Method)

前章で示した計算方法をＣ言語のプログラムで具体的に作り上げる。なお、当プログラムでは行列とベクトルの積を行う関数を別途作成し、この関数を BiCG-STAB 法の演算の中で逐次呼び出す形を採る。

３．１．行列とベクトルの積

行列とベクトルとの積を求める関数を以下に示す。なお、引数の"bn"は未知数の数（正方行列の一辺の要素数）、"bx[]"は行列が掛けられるベクトルの成分を格納した一次元配列、"by[]"は結果が格納される一次元配列である。

行列"mat[][]"とベクトルの積

```
void M_V_product(int bn,double bx[],double by[])
{
    int i,j;
    double buf0;

    for(i=0;i<bn;i++){
        buf0=0.0e0;
        for(j=0;j<bn;j++) buf0+=mat[i][j]*bx[j];
        by[i]=buf0;
    }

    return;
}
```

３．２．BiCG-STAB 法による連立一次方程式の演算

BiCG-STAB 法による演算を行う関数を具体的に記述すると次のようになる。なお、引数の"bn"は未知数の数（正方行列の一辺の要素数）、"bCOE"は反復演算停止条件(Condition of End)の閾値、"bav[]"は右辺のベクトル"**b**"の成分を格納した一次元配列、"bbv[]"は解ベクトルの成分が格納される一次元配列である。

BiCG-STAB 法による解ベクトルの演算

```
int BiCG_STAB_solver(int bn,double bCOE,double bav[],double bbv[])
{
    int i,j;
    double balp,bbta,brho0,brho1,bomg;
    double bro[NN],bri[NN],bvp[NN],bvs[NN],bvt[NN],bvv[NN],bvx[NN];
    double buf0,buf1;

    for(i=0;i<bn;i++) bvx[i]=1.0e0;                                   //(1)
    M_V_product(bn,bvx,bro);
    for(i=0;i<bn;i++){
        bri[i]=bav[i]-bro[i];                                         //(2)
        bro[i]=bri[i];                                                //(3)
    }
    buf0=0.0e0;
    for(i=0;i<bn;i++) buf0+=bri[i]*bro[i];
    if(buf0==0.0e0) for(i=0;i<bn;i+=2) bro[i]+=1.0e0;                 //(3)'

    balp=1.0e0; bomg=1.0e0; brho0=1.0e0; brho1=1.0e0;                 //(4)
    for(i=0;i<bn;i++){
        bvp[i]=0.0e0;                                                 //(4)
        bvv[i]=0.0e0;                                                 //(4)
    }
```

BiCG-STAB 法による解ベクトルの演算（続き）

```
    i=0;
    while(true){
        i++;
        brho1=0.0e0;
        for(j=0;j<bn;j++) brho1+=bro[j]*bri[j];                          //(5.1)
        bbta=(brho1*balp)/(brho0*bomg);                                  //(5.2)
        for(j=0;j<bn;j++) bvp[j]=bri[j]+bbta*(bvp[j]-bomg*bvv[j]);       //(5.3)
        M_V_product(bn,bvp,bvv);                                         //(5.4)
        buf0=0.0e0;
        for(j=0;j<bn;j++) buf0+=bro[j]*bvv[j];
        balp=brho1/buf0;                                                 //(5.5)
        for(j=0;j<bn;j++) bvs[j]=bri[j]-balp*bvv[j];                     //(5.6)
        M_V_product(bn,bvs,bvt);                                         //(5.7)
        buf0=0.0e0;
        for(j=0;j<bn;j++) buf0+=bvt[j]*bvs[j];
        buf1=0.0e0;
        for(j=0;j<bn;j++) buf1+=bvt[j]*bvt[j];
        bomg=buf0/buf1;                                                  //(5.8)
        for(j=0;j<bn;j++) bvx[j]=bvx[j]+balp*bvp[j]+bomg*bvs[j];         //(5.9)
        for(j=0;j<bn;j++) bri[j]=bvs[j]-bomg*bvt[j];                     //(5.10)
        buf0=0.0e0;
        for(j=0;j<bn;j++) buf0+=bri[j]*bri[j];
        buf0=sqrt(buf0);
        if(buf0<bCOE){                                                   //(5.11)
            for(j=0;j<bn;j++) bbv[j]=bvx[j];
            printf("\nBiCG-STAB repetitions=%5d, difference=%12.4e\n",i,buf0);
            fprintf(fp,"\nBiCG-STAB repetitions=%5d, difference=%12.4e\n",i,buf0);
            return i;
        }
        if(i>10*bn){                                                     ※強制打切
            for(i=0;i<bn;i++) bbv[i]=bvx[i];
            return -1;
        }
        brho0=brho1;
    }

    return -1;
}
```

４．結果(Results)

前章で作成したプログラムの実行結果を示す。

４．１．連立一次方程式の問題の準備

問題は以下に示す 5 つの連立一次方程式で、適当に係数行列と右辺のベクトルを設定したものである。

Ｑ１．

$$\begin{pmatrix} 9 & 8 & 0 & 0 & 2 & 1 \\ 3 & 5 & 1 & 6 & 3 & 0 \\ 5 & 6 & 2 & 3 & 3 & 2 \\ 6 & 4 & 5 & 0 & 0 & 3 \\ 0 & 7 & 9 & 4 & 4 & 3 \\ 9 & 1 & 7 & 5 & 5 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 3 \\ 3 \\ 1 \\ 7 \end{pmatrix}$$

Ｑ２．

$$\begin{pmatrix} 3 & 1 & 1 & 0 & 4 & 0 & 9 \\ 0 & 1 & 1 & 0 & 2 & 8 & 7 \\ 2 & 1 & -1 & 4 & 7 & 0 & 1 \\ 4 & 5 & 1 & 0 & 7 & 6 & 5 \\ 8 & 1 & 7 & 9 & 8 & 4 & 0 \\ 7 & 0 & 5 & 3 & 5 & 0 & 1 \\ 0 & 8 & 2 & 2 & 4 & 2 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix} = \begin{pmatrix} 0 \\ 4 \\ -4 \\ 2 \\ 7 \\ 8 \\ 9 \end{pmatrix}$$

Ｑ３．

$$\begin{pmatrix} 9 & 8 & 0 & 4 & 2 & 0 & 1 & 9 \\ 4 & 5 & 1 & 6 & 3 & 0 & 2 & 8 \\ 6 & 6 & 2 & 8 & 3 & 2 & 5 & 9 \\ 3 & 4 & 5 & 0 & 0 & 3 & 6 & 6 \\ 2 & 7 & 9 & 2 & 4 & 6 & 3 & 3 \\ 0 & 1 & 7 & 5 & 5 & 4 & 0 & 2 \\ 0 & 1 & 5 & 6 & 1 & 1 & 0 & 1 \\ 3 & 0 & 3 & 7 & 8 & 7 & 1 & 4 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{pmatrix} = \begin{pmatrix} 1 \\ 5 \\ 6 \\ 3 \\ 3 \\ 7 \\ 3 \\ 0 \end{pmatrix}$$

Ｑ４．

$$\begin{pmatrix} 0 & 2 & 0 & 4 & 0 & 0 & 1 & 7 & 8 \\ 6 & 9 & 1 & 6 & 5 & 0 & 5 & 8 & 5 \\ 4 & 5 & 2 & 8 & 3 & 2 & 9 & 9 & 3 \\ 8 & 4 & 5 & 0 & 0 & 3 & 6 & 5 & 1 \\ 5 & 7 & 9 & 2 & 3 & 6 & 3 & 3 & 0 \\ 0 & 6 & 7 & 5 & 5 & 4 & 0 & 2 & 2 \\ 2 & 0 & 5 & 6 & 8 & 1 & 0 & 6 & 0 \\ 8 & 4 & 3 & 7 & 8 & 7 & 7 & 4 & 8 \\ 9 & 5 & 5 & 7 & 0 & 5 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \end{pmatrix} = \begin{pmatrix} 8 \\ 5 \\ 4 \\ 6 \\ 0 \\ 1 \\ 3 \\ 0 \\ 7 \end{pmatrix}$$

Ｑ５．

$$\begin{pmatrix} 1 & 5 & 3 & 9 & 8 & 8 & 1 & 3 & 4 & 0 \\ 1 & 5 & 3 & 6 & 0 & 0 & 8 & 2 & 9 & 9 \\ 0 & 2 & 1 & 2 & 9 & 9 & 7 & 6 & 3 & 9 \\ 7 & 4 & 6 & 2 & 0 & 8 & 4 & 7 & 0 & 5 \\ 8 & 5 & 0 & 0 & 2 & 7 & 6 & 0 & 2 & 8 \\ 5 & 4 & 8 & 9 & 7 & 6 & 5 & 8 & 1 & 5 \\ 1 & 9 & 6 & 6 & 4 & 2 & 5 & 8 & 8 & 0 \\ 2 & 8 & 7 & 4 & 6 & 3 & 9 & 7 & 2 & 6 \\ 0 & 2 & 8 & 0 & 3 & 5 & 3 & 1 & 2 & 0 \\ 9 & 2 & 8 & 3 & 8 & 4 & 0 & 1 & 5 & 7 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{pmatrix} = \begin{pmatrix} 2 \\ 6 \\ 3 \\ 5 \\ 9 \\ 0 \\ 6 \\ 8 \\ 6 \\ 1 \end{pmatrix}$$

４．２．実行結果

実行時のログファイルを以下に示す。幸いにも全ての係数行列は正則だったようで、全ての問題に対してほぼ正確に解ベクトルが得られた。なお、いずれも残差の大きさ（各成分の差の二乗和の平方根）が 1.0×10^{-10} 未満となったら演算を終了するようにしている。また、収束しなかった場合の対策として、演算回数が未知数の数の 10 倍を超えたら演算を打ち切るようにしている。

Ｑ１．6×6 の場合

```
SOM=6

9.0e+000, 8.0e+000, 0.0e+000, 0.0e+000, 2.0e+000, 1.0e+000,
3.0e+000, 5.0e+000, 1.0e+000, 6.0e+000, 3.0e+000, 0.0e+000,
5.0e+000, 6.0e+000, 2.0e+000, 3.0e+000, 3.0e+000, 2.0e+000,
6.0e+000, 4.0e+000, 5.0e+000, 0.0e+000, 0.0e+000, 3.0e+000,
0.0e+000, 7.0e+000, 9.0e+000, 4.0e+000, 4.0e+000, 3.0e+000,
9.0e+000, 1.0e+000, 7.0e+000, 5.0e+000, 5.0e+000, 0.0e+000,

0.0e+000, 0.0e+000, 3.0e+000, 3.0e+000, 1.0e+000, 7.0e+000,

BiCG-STAB repetitions=    9, difference= 1.4598e-011

result
 4.4221e-001, -9.9329e-001, -2.6398e-001,  1.2901e-001,  1.0433e+000,  1.8799e+000,

check
-8.7e-012, 4.2e-012, 3.0e+000, 3.0e+000, 1.0e+000, 7.0e+000,
```

Ｑ２．7×7 の場合

```
SOM=7

3.0e+000, 1.0e+000, 1.0e+000, 0.0e+000, 4.0e+000, 0.0e+000, 9.0e+000,
0.0e+000, 1.0e+000, 1.0e+000, 0.0e+000, 2.0e+000, 8.0e+000, 7.0e+000,
2.0e+000, 1.0e+000, -1.0e+000, 4.0e+000, 7.0e+000, 0.0e+000, 1.0e+000,
4.0e+000, 5.0e+000, 1.0e+000, 0.0e+000, 7.0e+000, 6.0e+000, 5.0e+000,
8.0e+000, 1.0e+000, 7.0e+000, 9.0e+000, 8.0e+000, 4.0e+000, 0.0e+000,
7.0e+000, 0.0e+000, 5.0e+000, 3.0e+000, 5.0e+000, 0.0e+000, 1.0e+000,
0.0e+000, 8.0e+000, 2.0e+000, 2.0e+000, 4.0e+000, 2.0e+000, 0.0e+000,

0.0e+000, 4.0e+000, -4.0e+000, 2.0e+000, 7.0e+000, 8.0e+000, 9.0e+000,

BiCG-STAB repetitions=   12, difference= 8.2531e-012

result
-3.6235e+000, -9.8079e-001,  5.4045e+000, -2.5875e+000,  2.9389e+000, -2.7156e-001, -5.8985e-001,

check
3.0e-012, 4.0e+000, -4.0e+000, 2.0e+000, 7.0e+000, 8.0e+000, 9.0e+000,
```

Ｑ３． 8×8 の場合

```
SOM=8

9.0e+000, 8.0e+000, 0.0e+000, 4.0e+000, 2.0e+000, 0.0e+000, 1.0e+000, 9.0e+000,
4.0e+000, 5.0e+000, 1.0e+000, 6.0e+000, 3.0e+000, 0.0e+000, 2.0e+000, 8.0e+000,
6.0e+000, 6.0e+000, 2.0e+000, 8.0e+000, 3.0e+000, 2.0e+000, 5.0e+000, 9.0e+000,
3.0e+000, 4.0e+000, 5.0e+000, 0.0e+000, 0.0e+000, 3.0e+000, 6.0e+000, 6.0e+000,
2.0e+000, 7.0e+000, 9.0e+000, 2.0e+000, 4.0e+000, 6.0e+000, 3.0e+000, 3.0e+000,
0.0e+000, 1.0e+000, 7.0e+000, 5.0e+000, 5.0e+000, 4.0e+000, 0.0e+000, 2.0e+000,
0.0e+000, 1.0e+000, 5.0e+000, 6.0e+000, 1.0e+000, 1.0e+000, 0.0e+000, 1.0e+000,
3.0e+000, 0.0e+000, 3.0e+000, 7.0e+000, 8.0e+000, 7.0e+000, 1.0e+000, 4.0e+000,

1.0e+000, 5.0e+000, 6.0e+000, 3.0e+000, 3.0e+000, 7.0e+000, 3.0e+000, 0.0e+000,

BiCG-STAB repetitions=   11, difference= 3.7171e-012

result
  2.2462e+000, -1.0529e+000,  2.2326e+000, -5.8441e-001,  4.8903e+000, -6.0585e+000,  3.6830e+000,
-2.4355e+000,

check
1.0e+000, 5.0e+000, 6.0e+000, 3.0e+000, 3.0e+000, 7.0e+000, 3.0e+000, -2.0e-012,
```

Ｑ４． 9×9 の場合

```
SOM=9

0.0e+000, 2.0e+000, 0.0e+000, 4.0e+000, 0.0e+000, 0.0e+000, 1.0e+000, 7.0e+000, 8.0e+000,
6.0e+000, 9.0e+000, 1.0e+000, 6.0e+000, 5.0e+000, 0.0e+000, 5.0e+000, 8.0e+000, 5.0e+000,
4.0e+000, 5.0e+000, 2.0e+000, 8.0e+000, 3.0e+000, 2.0e+000, 9.0e+000, 9.0e+000, 3.0e+000,
8.0e+000, 4.0e+000, 5.0e+000, 0.0e+000, 0.0e+000, 3.0e+000, 6.0e+000, 5.0e+000, 1.0e+000,
5.0e+000, 7.0e+000, 9.0e+000, 2.0e+000, 3.0e+000, 6.0e+000, 3.0e+000, 3.0e+000, 0.0e+000,
0.0e+000, 6.0e+000, 7.0e+000, 5.0e+000, 5.0e+000, 4.0e+000, 0.0e+000, 2.0e+000, 2.0e+000,
2.0e+000, 0.0e+000, 5.0e+000, 6.0e+000, 8.0e+000, 1.0e+000, 0.0e+000, 6.0e+000, 0.0e+000,
8.0e+000, 4.0e+000, 3.0e+000, 7.0e+000, 8.0e+000, 7.0e+000, 7.0e+000, 4.0e+000, 8.0e+000,
9.0e+000, 5.0e+000, 5.0e+000, 7.0e+000, 0.0e+000, 5.0e+000, 0.0e+000, 1.0e+000, 0.0e+000,

8.0e+000, 5.0e+000, 4.0e+000, 6.0e+000, 0.0e+000, 1.0e+000, 3.0e+000, 0.0e+000, 7.0e+000,

BiCG-STAB repetitions=   13, difference= 2.3873e-011

result
  8.3626e-001, -5.7959e-001,  2.0406e+000,  9.3447e-001, -9.3005e-001, -2.7308e+000,  3.2408e-001,
-7.1850e-001,  1.2658e+000,

check
8.0e+000, 5.0e+000, 4.0e+000, 6.0e+000, -1.1e-011, 1.0e+000, 3.0e+000, 1.6e-012, 7.0e+000,
```

Ｑ５．10×10の場合

```
SOM=10

1.0e+000, 5.0e+000, 3.0e+000, 9.0e+000, 8.0e+000, 8.0e+000, 1.0e+000, 3.0e+000, 4.0e+000, 0.0e+000,
1.0e+000, 5.0e+000, 3.0e+000, 6.0e+000, 0.0e+000, 0.0e+000, 8.0e+000, 2.0e+000, 9.0e+000, 9.0e+000,
0.0e+000, 2.0e+000, 1.0e+000, 2.0e+000, 9.0e+000, 9.0e+000, 7.0e+000, 6.0e+000, 3.0e+000, 9.0e+000,
7.0e+000, 4.0e+000, 6.0e+000, 2.0e+000, 0.0e+000, 8.0e+000, 4.0e+000, 7.0e+000, 0.0e+000, 5.0e+000,
8.0e+000, 5.0e+000, 0.0e+000, 0.0e+000, 2.0e+000, 7.0e+000, 6.0e+000, 0.0e+000, 2.0e+000, 8.0e+000,
5.0e+000, 4.0e+000, 8.0e+000, 9.0e+000, 7.0e+000, 6.0e+000, 5.0e+000, 8.0e+000, 1.0e+000, 5.0e+000,
1.0e+000, 9.0e+000, 6.0e+000, 6.0e+000, 4.0e+000, 2.0e+000, 5.0e+000, 8.0e+000, 8.0e+000, 0.0e+000,
2.0e+000, 8.0e+000, 7.0e+000, 4.0e+000, 6.0e+000, 3.0e+000, 9.0e+000, 7.0e+000, 2.0e+000, 6.0e+000,
0.0e+000, 2.0e+000, 8.0e+000, 0.0e+000, 3.0e+000, 5.0e+000, 3.0e+000, 1.0e+000, 2.0e+000, 0.0e+000,
9.0e+000, 2.0e+000, 8.0e+000, 3.0e+000, 8.0e+000, 4.0e+000, 0.0e+000, 1.0e+000, 5.0e+000, 7.0e+000,

2.0e+000, 6.0e+000, 3.0e+000, 5.0e+000, 9.0e+000, 0.0e+000, 6.0e+000, 8.0e+000, 6.0e+000, 1.0e+000,

BiCG-STAB repetitions=   14, difference= 5.3957e-011

result
-1.8941e-001,  1.0913e+000,  2.5426e-001, -4.5816e-001, -2.4814e-001,  3.9616e-001,  3.3397e-001,
-5.1468e-001,  2.9827e-002,  8.9786e-002,

check
2.0e+000, 6.0e+000, 3.0e+000, 5.0e+000, 9.0e+000, -3.5e-012, 6.0e+000, 8.0e+000, 6.0e+000, 1.0e+000,
```

５．考察(Discussion)

今回は適当な問題を設定し、それをBiCG-STAB法を用いて解いた。収束条件をもう少し緩くすれば反復回数を少なくできるだろうが、係数行列の性質にもかなり依存するので個々の問題に対して適切な収束条件を考える必要がある。

６．結言(Summary)

連立一次方程式の解法には色々なものがあり、今回はその内のBiCG-STAB法のみを取り上げた。これは共役勾配法(Conjugate Gradient method)が発展・派生したものの1つで、現在もさらなる改良が続けられているアルゴリズムである。しかし、あくまで連立一次方程式が実用的な範疇で解ければ、何でも構わないというような人（工学系研究者・技術者等）にとっては、その後ろ盾となる理論にはあまり興味が無く、便利な道具として自分が利用できれば良いというスタンスがほとんどであると思われる。もちろん自分もその立場で、今後色々な問題を考えていく上で、自分自身が本稿の内容を再利用できればいいという考えである。

７．文献(References)

［１］：https://en.wikipedia.org/wiki/Biconjugate_gradient_stabilized_method
［２］：http://www.jicfus.jp/wiki/index.php?Bi-CGSTAB%20%E6%B3%95
［３］：工学基礎　数値解析とその応用　久保田光一著　数理工学社
［４］：C言語による数値計算入門　皆本晃弥著　サイエンス社

８．著者(Author)

氏名：志多　友史（工学修士）
略歴：
　2011年：下位国立大学　工学部電気系学科卒業
　2013年：同大学大学院　工学研究科修了
　2013年：研究開発機関へ就職
興味：物理・数学・コンピュータ・電気電子工作

９．備考(Notes)

特になし。