

# 直接解法による大規模疎行列に 対する連立1次方程式ソルバー

— IBM WSMP の紹介 —

渡部 善隆

watanabe@cc.kyushu-u.ac.jp .

九州大学情報基盤センター

# 連立1次方程式

$n \times n$  行列  $A = [a_{ij}]$  とベクトル  $\mathbf{b} = [b_1, \dots, b_n]^T$  に対し

$$A\mathbf{x} = \mathbf{b}$$

をみたすベクトル  $\mathbf{x} = [x_1, \dots, x_n]^T$  を求める問題．

- 理学・工学・農学・社会科学・人文科学のあらゆる分野に現れる．
- 実際の科学技術分野における数値計算の計算時間の大半は連立1次方程式を解くことに費やされているといわれている．

# 次数

方程式の次数 ( $n \times n$  の “ $n$ ”) は問題によって異なる .

- 2次元 Poisson 方程式の数値解法の例題 ..... 121
- 定常 Navier-Stokes 方程式の精度保証 ..... 11,400
- 代数的マルチグリッド法による電磁界解析 ..... 160,464
- Stokes 方程式の有限要素近似解 ..... 1,410,479
- 原子炉圧力モデルの弾性構造解析 ..... 10,328,853

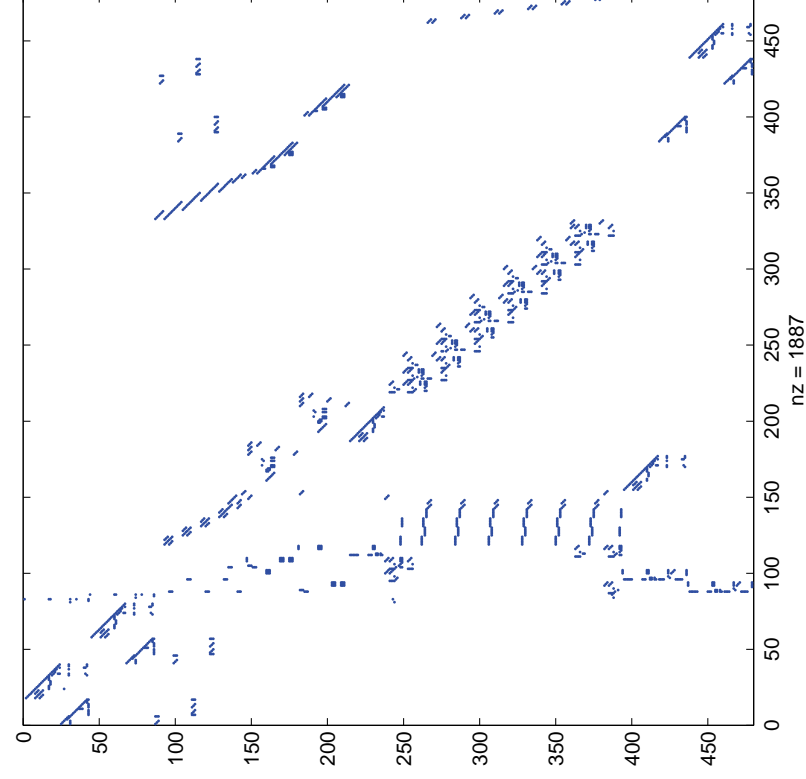
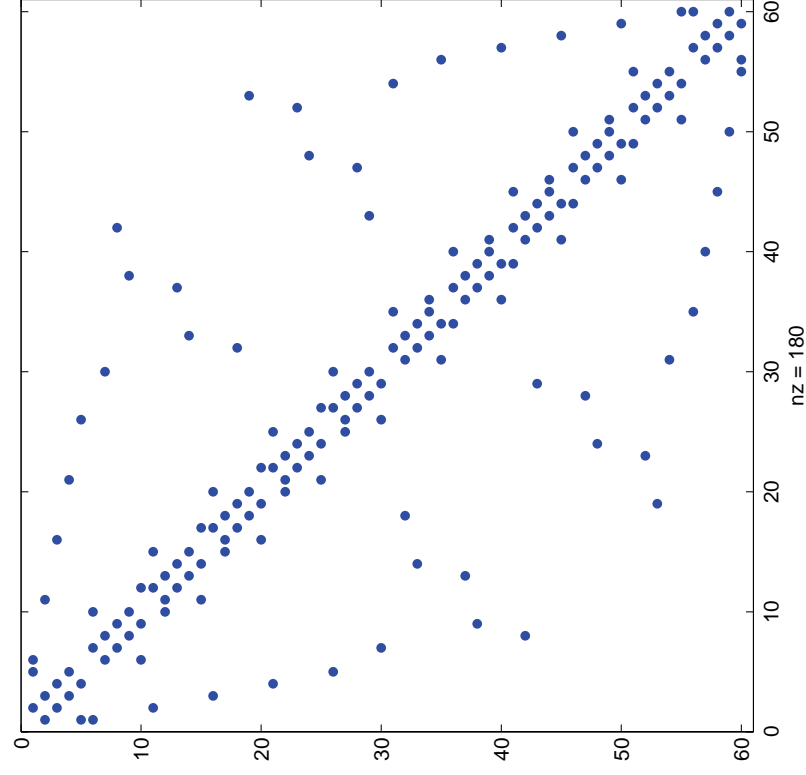
$n$  が大きくなると, 全要素をメモリに格納することは困難 .

《例》  $10,000,000 \times 10,000,000$  の行列を IEEE754 倍精度実数  
(8バイト) で宣言

$$8 \times 10,000,000 \times 10,000,000 \approx 800,000 \text{ GByte}$$

# 疎行列 (sparse matrix)

たとえば有限要素法・差分法で関数方程式（連続問題）を離散化した場合，得られる  $A$  の多くは疎行列となる．



# 疎行列の格納形式

行列の非零要素（と限られた数の零要素）を記憶領域に格納する方法．

$$A = \begin{bmatrix} 0 & 0 & 0 & 5 \\ 0 & 2 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 \end{bmatrix} \Rightarrow S = \begin{array}{ll} (3, 1) & 1 \\ (2, 2) & 2 \\ (4, 3) & 4 \\ (1, 4) & 5 \end{array}$$

- 圧縮行格納法 (compressed row storage)
- 圧縮列格納法 (compressed column storage)
- 圧縮対角格納法 (compressed diagonal storage)
- スカイライン格納法 (skyline storage)
- etc.

# 連立1次方程式の数値解法の分類

- 直接法
  - Gaussの消去法, Cholesky分解, etc.
- 反復法
  - 定常反復法
    - SOR法, Gauss-Seidel法, etc.
  - 非定常反復法
    - CG法, BiCG法, GMRES法, etc.
- 直接法と反復法の組み合わせ
  - 解の反復改良, 不動点定理に基づく解の精度保証, etc.

# 反復法 $Ax = b$

真の解に収束していく近似解の列を逐次作成していく手法．

$$\boldsymbol{x}^0 \rightarrow \boldsymbol{x}^1 \rightarrow \boldsymbol{x}^2 \rightarrow \dots \rightarrow \boldsymbol{x}$$

- 定常反復法

一定（定常）の処理を繰り返すことによって近似解を作成する方法．

$$\boldsymbol{x}^k = \boldsymbol{x}^{k-1} + R(\boldsymbol{b} - A\boldsymbol{x}^{k-1}), \quad k \geq 1, \quad R: \text{given}$$

- 非定常反復法

各反復ごとに変化する情報を取り込みながら計算を進める．  
多くは反復によって直交するベクトル列を作り出す．

$$\boldsymbol{x}^k \in \text{span}\{\boldsymbol{r}_0, A\boldsymbol{r}_0, A^2\boldsymbol{r}_0, \dots, A^{k-1}\boldsymbol{r}_0\} \quad \boldsymbol{r}_0: \text{given}$$

# 反復法の特徴

- 行列を分解する必要がないため，疎行列の構造を保つことができる。
- 基本の計算は「行列×ベクトル」と内積計算のため，収束が速ければ計算量が

$$\text{非零要素数} \times n$$

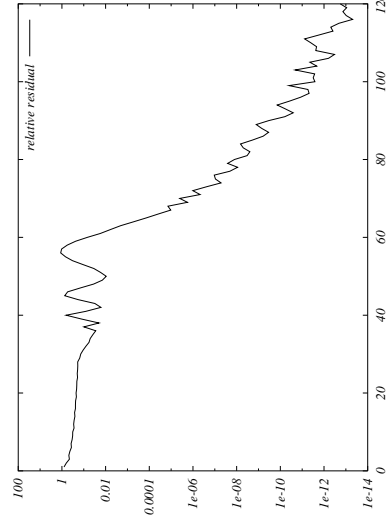
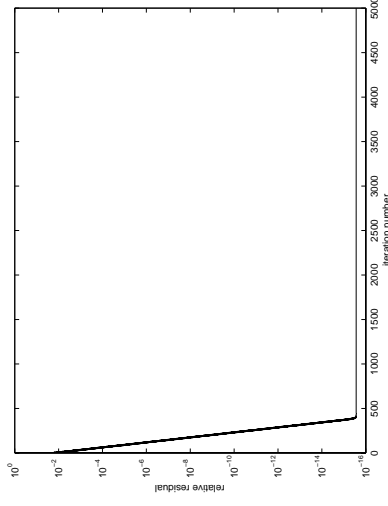
のオーダーになることが期待される。

- 収束を加速させるための前処理技法が数多く提案されている。



# 反復法の問題点

- 行列の性質（固有値・特異値など）に大きく依存する．
- 初期値  $x^0$  によって収束特性が変わることがある．
- 右辺  $b$  によって収束特性が変わることがある．
- 非定常反復法は丸め誤差の影響を受けやすい．



- 主要な非対称行列に対する非定常反復法については「最適な手法はない」という研究結果がある．

“How fast are nonsymmetric matrix iterations?”

SIAM Journal on Matrix Analysis and Applications, Vol.13, 1992, 778-795.

# 直接法 $Ax = b$

ほとんどが行列の分解に基づく手法．

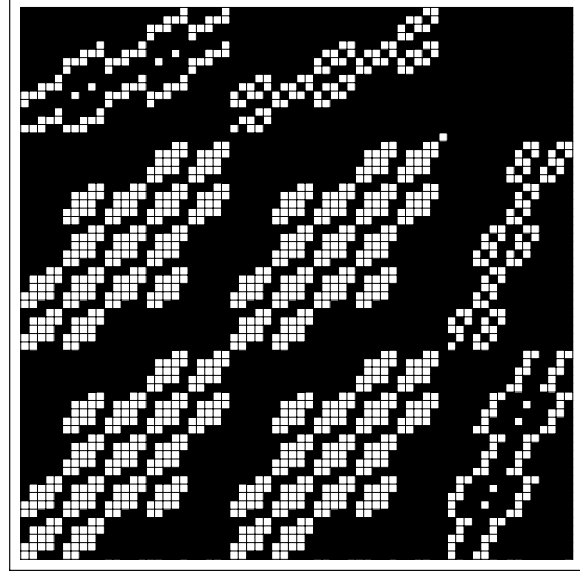
- $LU$  分解:

$$A = LU, \quad L: \text{下三角行列}, \quad U: \text{上三角行列}.$$

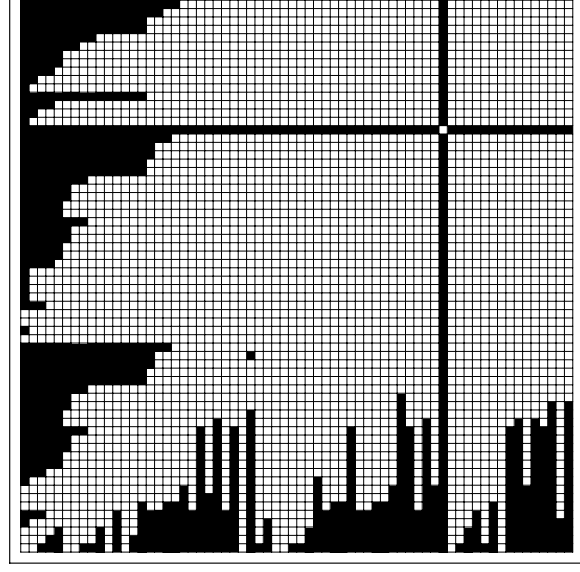
- $A$  が正則  $\Leftrightarrow A$  が適当な列変換により  $LU$  分解可能．
- 行列に特異性がない限り（経験的に）安定に解ける．
- 一度行列を分解しておけば，異なる  $b$  に対して（行列の分解に比べて）高速に計算可能．
- 精度保証付き数値計算への応用が可能  
（連立 1 次方程式，固有値問題，特異値問題）

# (通常の) 直接法の問題点

1.  $LU$  分解の計算量は  $O(n^3)$  ( $n$  は次数) .  
次数が大きくなると計算時間が膨大になる .
2. 分解を行なうと行列の疎な構造が崩れる .  
より多くのメモリ空間が必要 .



$LU$  分解前



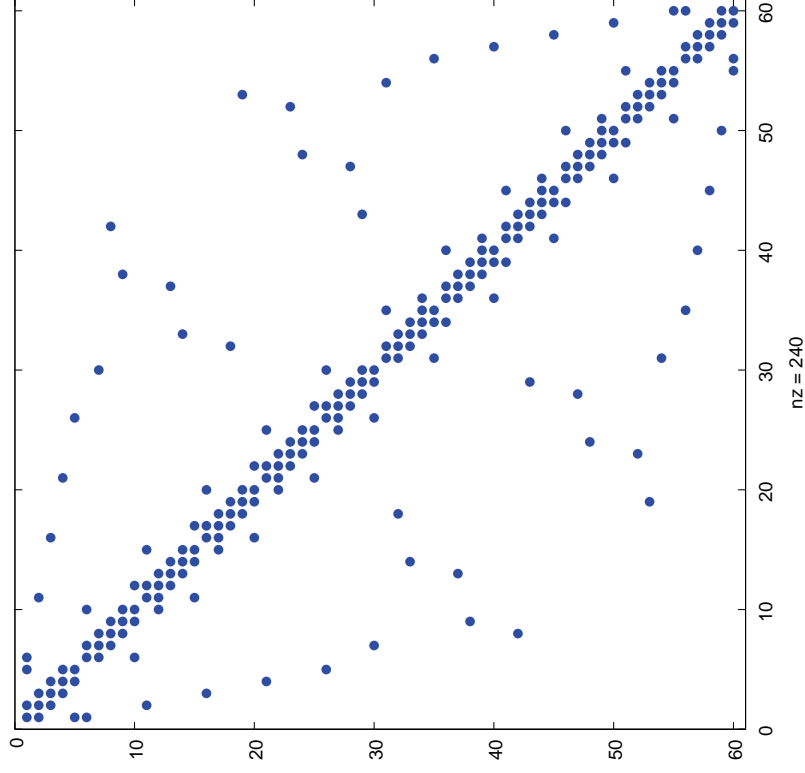
$LU$  分解後

分解によるゼロ要素の損失を **fill-in** と呼ぶ .

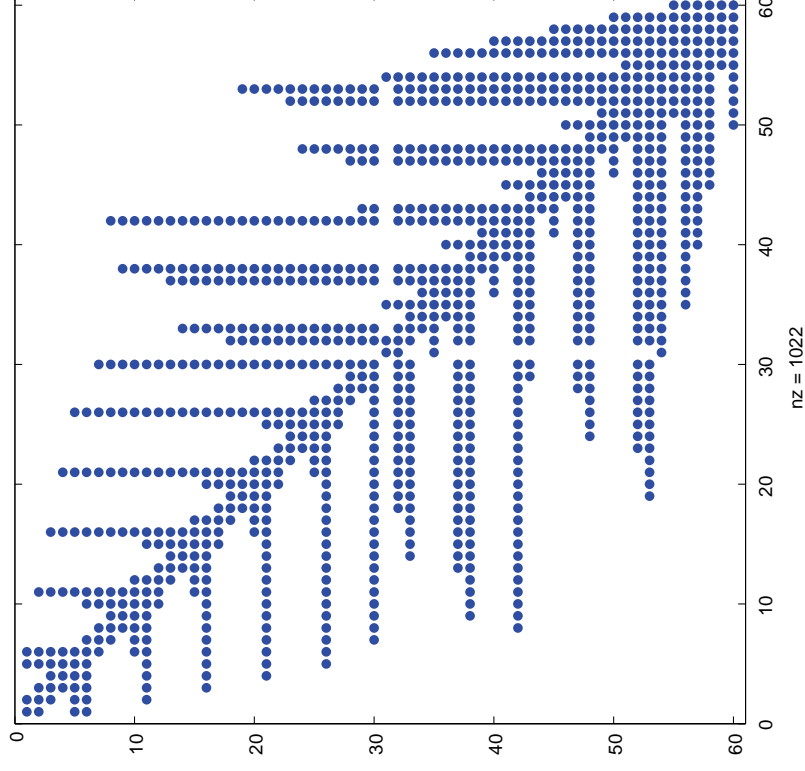
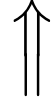
# 疎行列に対する直接法の研究

- fill-in の数を《出来るだけ》減らす手法．  
メモリ量と計算量の削減．
- 疎行列構造を生かした  $LU$  分解手法．
  - $LU$  分解後の行列構造をあらかじめ求め、不要な計算を省く．  
計算量の削減．

# Fill reducing ordering



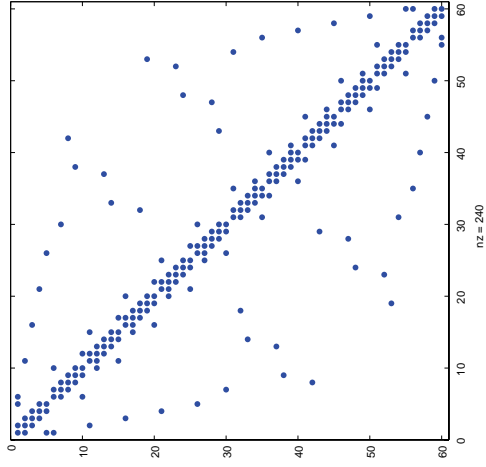
$A$



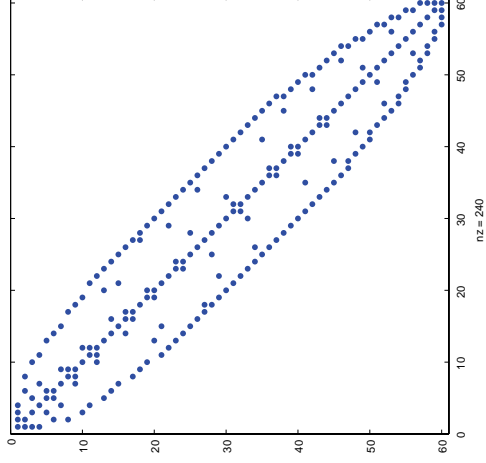
$LU$

$P \times A \times Q,$      $P, Q:$  permutation matrix

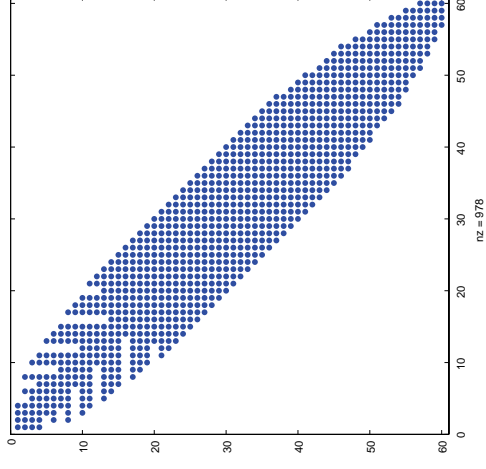
# Reverse Cuthill-McKee ordering



$A$

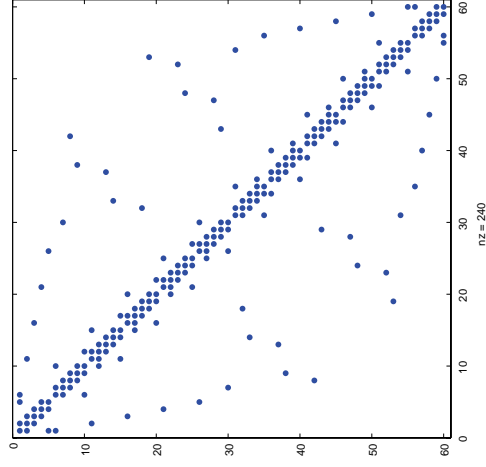


$PAQ$

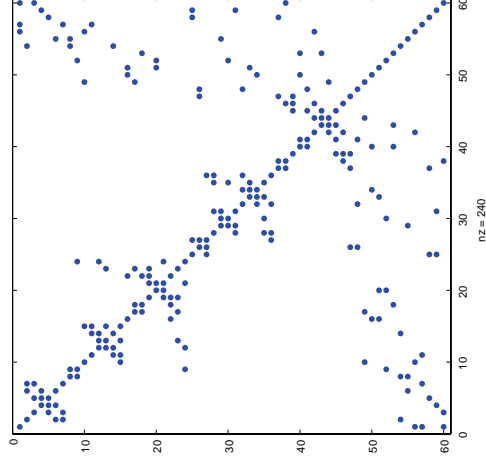


$LU$

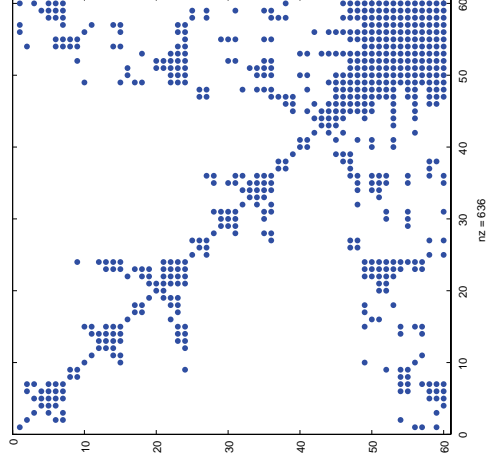
# Minimum degree ordering



$A$



$PAQ$



$LU$

# Ordering algorithms

**MD** minimum degree (Tinney and Walker, 1967)

**MMD** multiple minimum degree (Liu, 1985)

**AMD** approximate minimum degree (Amestoy, et al., 1996)

**ND** nested dissection (George, 1973)

**METIS** multilevel nested dissection routine  
(Karypis and Kumar, 1998)

**MS** multisection (Ashcraft and Liu 1998)

**MF** local minimum fill (Tinney and Walker 1967)



# Factorization algorithm for $A = A^T$

$$A = \begin{bmatrix} B & V^T \\ V & C \end{bmatrix}, \quad B = L_B L_B^T \in \mathbf{R}^{j-1 \times j-1}$$

$$= \begin{bmatrix} L_B & 0 \\ V L_B^{-T} & I \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & C - V B^{-1} V^T \end{bmatrix} \begin{bmatrix} L_B & 0 \\ V L_B^{-T} & I \end{bmatrix}^T.$$













$$V B^{-1} V^T = (V L_B^{-T}) (V L_B^{-T})^T$$

$$= \sum_{k=1}^{j-1} \begin{bmatrix} l_{j,k} \\ \vdots \\ l_{n,k} \end{bmatrix} [l_{j,k} \cdots l_{n,k}].$$

# Sparse direct solvers

code	fill reducing ordering	factorization algorithm
BCSLIB-EXT	MMD, METIS	multifrontal
MA57	MD, AMD, METIS	multifrontal
MUMPS	MD, AMD, METIS	multifrontal
Obljo	MMD, METIS	left/right-looking, multifrontal
PARDISO	MMD, METIS	left/right-looking
SPOOLES	MMD, ND, MS	left-looking
SPRSBLKLLT	MMD	left-looking
SuperLU	MMD	left-looking
Super Matrix Solver-MF	ND	multifrontal
TAUCS	MD, AMD, MMD, METIS	left-looking, multifrontal
UMFPACK	AMD	unsymmetric multifrontal
WSMP	ND, MF	multifrontal

# Sparse direct solvers

code	fill reducing ordering	factorization algorithm
BCSLIB-EXT	MMD, METIS	multifrontal 
MA57	MD, AMD, METIS	multifrontal 
MUMPS	MD, AMD, METIS	multifrontal 
Obljo	MMD, METIS	left/right-looking, multifrontal 
PARDISO	MMD, METIS	left/right-looking 
SPOOLES	MMD, ND, MS	left-looking 
SPRSBLKLLT	MMD	left-looking 
SuperLU	MMD	left-looking 
Super Matrix Solver-MF	ND	multifrontal 
TAUCS	MD, AMD, MMD, METIS	left-looking, multifrontal 
UMFPACK	AMD	unsymmetric multifrontal 
WSMP	ND, MF	multifrontal 

# applications

- **Mathematica**  
LinearSolve は UMFPACK を利用
- **Soil Plus**  
地盤・耐震統合 FEM 解析システム
- **APSYS**  
半導体デバイス解析ソフト
- **FUJITSU FEM5**  
有限要素法による構造解析プログラム
- **MSC.SuperForm**  
成形加工専用プログラム
- **Mr.SOIL3D**  
2次元/3次元地盤 FEM 解析システム

# WSMPの概要

- Watson Sparse Matrix Package
- 作者: Anshul Gupta and M.Joshi(IBM T.J. Watson Research Center)
- 逐次版は無料 (AIX, Linux, SunOS, Tru64, HP-UX) .
- ソースコード非公開 .
- 実対称正定値 , 実対称不定値 , 実一般 , 複素 Hermite , 複素対称 , 複素数一般
- Fortran, C から利用可能
- 倍精度
- 圧縮格納形式: CSR, CSC, CSR-UT, CSC-LT, MSR, MSC,
- スレッド並列版 , MPI 並列版 , 内部でチューニングされた Level3 BLAS を使用 .

# 数値計算環境



計算機名	IBM eServer p5 モデル 595
プロセッサ	POWER5 1.90GHz(Turbo) 64way
主記憶容量	256GB
L2 キャッシュ	1.9MB/2way
L3 キャッシュ	36MB/2way
OS	AIX 5L V5.3
コンパイラ	IBM XL Fortran Enterprise Edition V9.1

16way で並列実行，最大使用可能メモリ 48GB，  
IEEE754 倍精度（64 ビット）実数．

# 利用方法

```
! real symmetric
```

```
WSSMP (N, IA, JA, AVALS, DIAG, PERM, INVP, B, LDB, NRHS,  
        AUX, NAUX, MRP, IPARM, DPARM)
```

```
! complex Hermite/symmetric
```

```
ZSSMP (N, IA, JA, AVALS, DIAG, PERM, INVP, B, LDB, NRHS,  
        AUX, NAUX, MRP, IPARM, DPARM)
```

```
! real general
```

```
WGSMP (N, IA, JA, AVALS, B, LDB, NRHS, RMISC,  
        IPARM, DPARM)
```

```
! complex general
```

```
ZGSMP (N, IA, JA, AVALS, B, LDB, NRHS, RMISC,  
        IPARM, DPARM)
```

# Phases in WSMP

- 0. Initializing**  
(parameter check, allocation, etc.)
- 1. Ordering**  
(Permute rows and columns of  $A$ )
- 2. Symbolic Factorization**  
(Compute elimination graph and structure of  $L$ ,  $U$ ,  
where  $A = LU$ )
- 3. Numerical Factorization**  
(Compute  $A = LU$ , or  $A = LL^T$ , or  $A = LDL^T$ )
- 4. Forward and Backward Triangle Solves**  
(Solve triangular systems  $Ly = b$  and  $Ux = y$ )
- 5. Iterative Refinement**  
(Compute  $r := b - Ax^n$  and  $x^{n+1} := x^n + (LU)^{-1}r$ )



# Typical Complexities in WSMP

## 1. Ordering

$$O(n) - O(n \log(n))$$

## 2. Symbolic Factorization

$$O(\text{NNZ}(A)) - O(\text{NNZ}(L + U))$$

## 3. Numerical Factorization

$$O(n^{3/2}) - O(n^2)$$

## 4. Forward and Backward Triangle Solves

$$O(\text{NNZ}(L + U)) = O(n \log(n)) - O(n^{4/3})$$

## 5. Iterative Refinement

$$O(n \log(n)) - O(n^{4/3})$$

# 各ステップの実測比（対称行列）

行列名	並列度	step0	step1	step2	step3	step4	step5
FINAN512	1	3.13	74.38	3.87	13.46	2.62	2.53
	2	0.28	72.40	8.94	12.99	2.75	2.62
	3	0.39	63.19	12.01	19.92	2.13	2.33
	4	0.42	63.00	12.03	20.05	2.10	2.38
TUER	1	0.01	61.98	3.53	30.16	2.05	2.27
	2	0.01	59.93	7.23	28.67	1.91	2.25
	3	0.01	62.04	10.78	23.62	1.52	2.02
	4	0.01	62.03	10.75	23.67	1.52	2.02
BODY4	1	0.79	74.35	5.33	13.89	2.74	2.82
	2	1.48	64.98	10.81	15.92	3.56	3.13
	3	1.99	57.25	13.21	22.18	2.58	2.65
	4	2.05	56.55	13.10	22.68	2.60	2.88

単位: %

# 使用メモリ量

行列名	次元数	非零要素数	Aに必要な量	実使用量
dawson5.rsa	51537	531157	4MB	73MB
HERM2D03.rsa	392257	1567096	11MB	320MB
cfd2.RSA	123440	1605669	12MB	450MB
M_T1.rsa	97578	4925574	37MB	530MB
bmwcra_1.rsa	148770	5396386	41MB	930MB
inline_1.rsa	503712	18660027	142MB	2,500MB
ldoor.rsa	952203	23737339	181MB	2,500MB

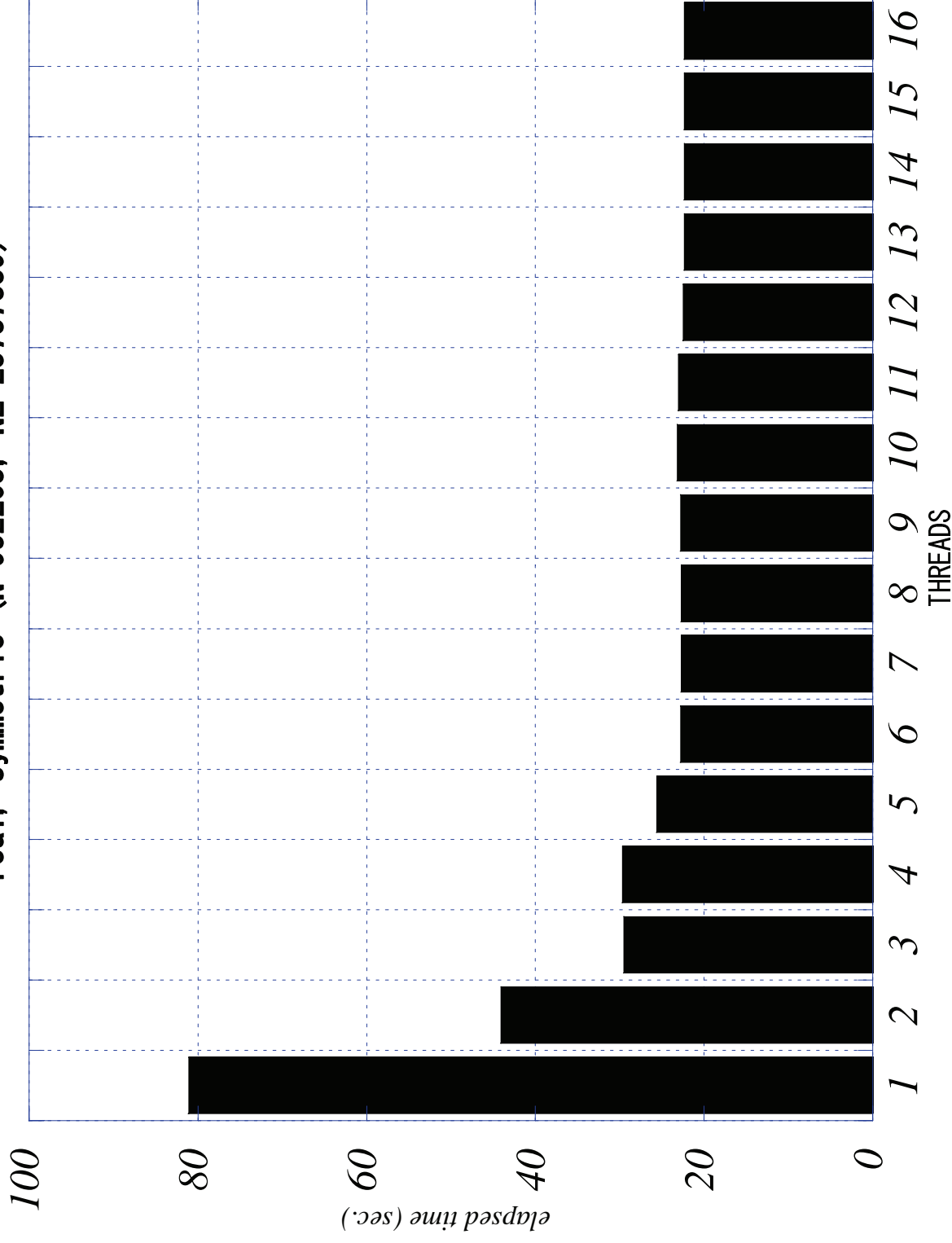
Compaq DS20 Alpha server (EV6), WSMP Version 1.9.8

N. I. M. Gould, Y. Hu and J. A. Scott.:

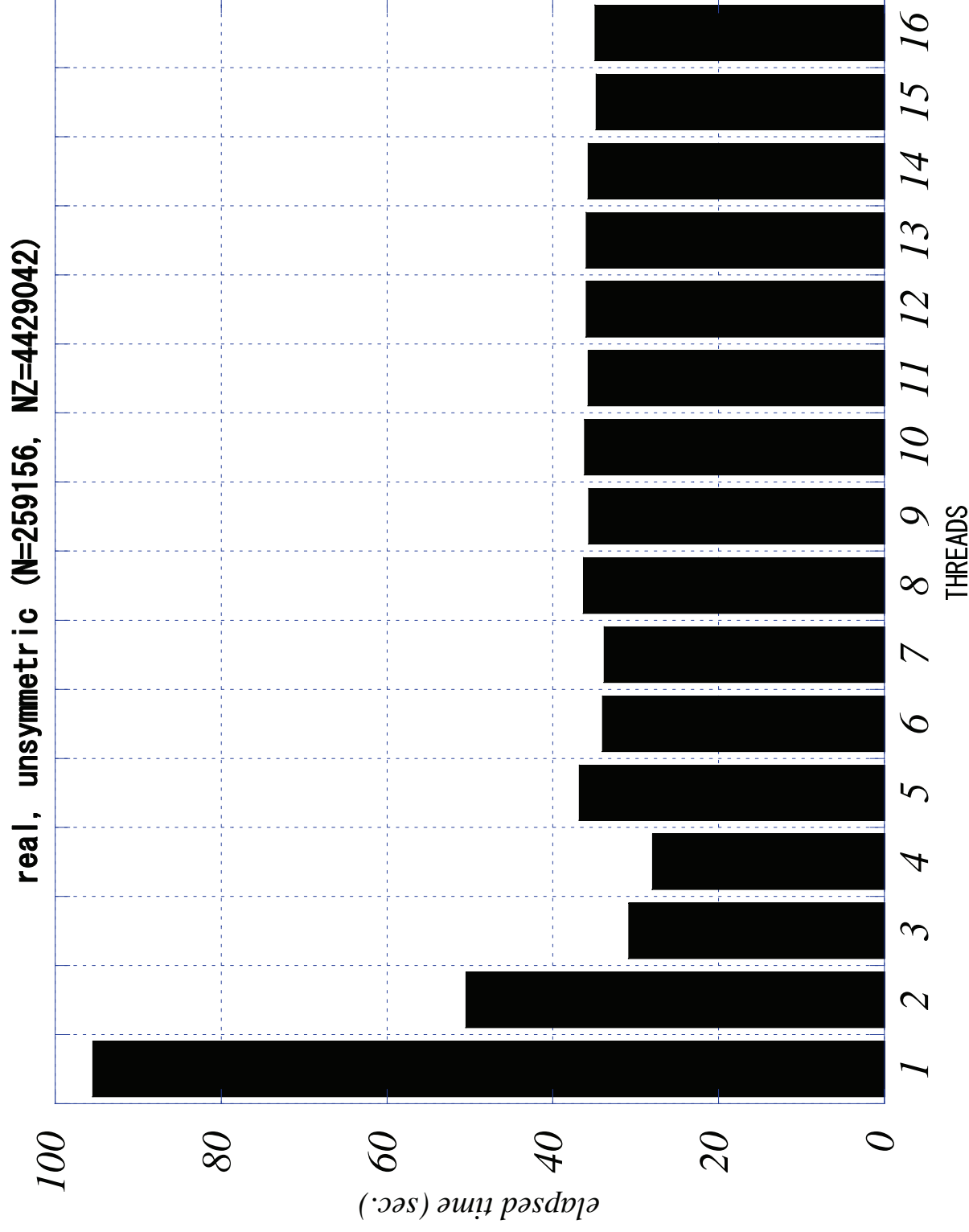
“Complete results from a numerical evaluation of sparse direct solvers for the solution of large, sparse, symmetric linear systems of equations”  
Numerical Analysis Group Internal Reports, 2005.

# 並列化効果（実数・正定値対称）

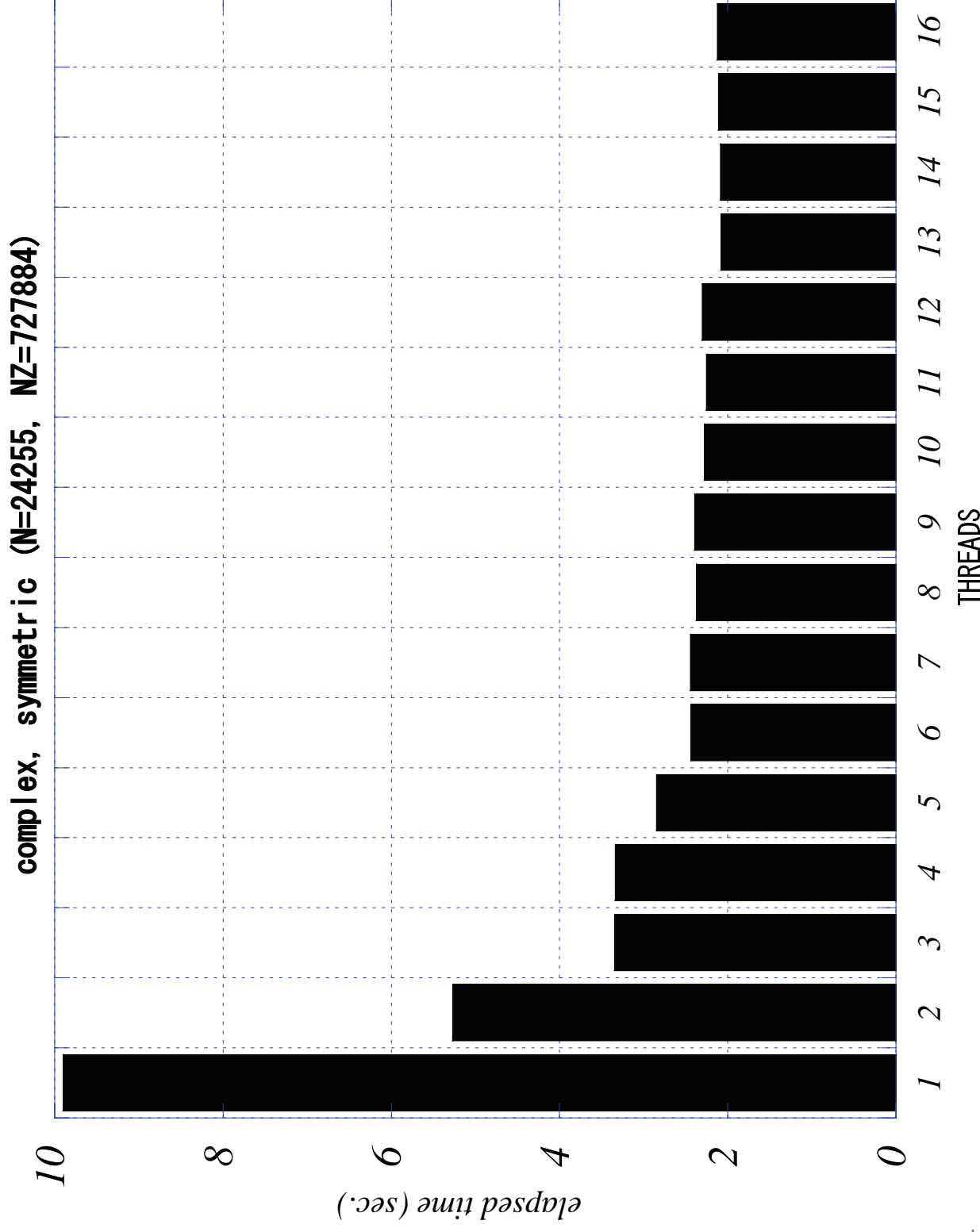
real, symmetric (N=952203, NZ=23737339)



# 並列化効果 (実数・非対称)



# 並列化効果 (複素数・対称)



# Test Matrices

- **Matrix Market**

<http://math.nist.gov/MatrixMarket/>

- **University of Florida Sparse Matrix Collection**

<http://www.cise.ufl.edu/research/sparse/matrices/>

- **Numerical Analysis Group of CCLRC**

<ftp://ftp.numerical.rl.ac.uk/pub/matrices/symmetric/>

# 性能比較・正定値対称行列



# テスト行列

名称	次元数	非零要素数	分野
bodyy6.RSA	19366	77057	NASA matrix
bcsstk36.RSA	23052	583096	automobile shock absorber
msc23052.RSA	23052	588933	MSC.NASTRAN
THREAD.rsa	29736	2249892	Threaded connector
GRIDGENA.rsa	48962	329485	Grid generation optimization
nasasrb.RSA	54870	1366097	Shuttle rocket booster
crankseg_2.rsa	63838	7106348	Linear static analysis
OILPAN.rsa	73752	1835470	Car oilpan
finan512.RSA	74752	335872	Portfolio optimization
s3dkq4m2.rsa	90449	2455670	Cylindrical Shell
M_T1.rsa	97578	4925574	Tubular joint
X104.rsa	108384	5138004	Beam joint
cfd2.RSA	123440	1605669	CFD pressure matrix
bmwcra_1.rsa	148770	5396386	Automotive crankshaft model
SHIPSEC5.rsa	179860	5146478	Ship section
hood.rsa	220542	5494489	Car hood
pwtk.RSA	217918	5926171	pressurized wind tunnel
inline_1.rsa	503712	18660027	Inline skater
audikw_1.rsa	943695	39297771	Automotive crankshaft model
ldoor.rsa	952203	23737339	Large door

# 解法と測定条件

- WSMP 5.3.15

- IBM ESSL V4.2

- DSRIS<sup>-10</sup>

不完全 Cholesky 分解付き CG 法，疎行列用反復法  
 $x^0 = 0$ ，反復停止基準:  $\|b - Ax^n\|_2 / \|x^n\|_2 < 10^{-10}$  .

- DSRIS<sup>-6</sup>

不完全 Cholesky 分解付き CG 法，疎行列用反復法  
 $x^0 = 0$ ，反復停止基準:  $\|b - Ax^n\|_2 / \|x^n\|_2 < 10^{-6}$  .

- DGESV

部分ピボット選択付き LU 分解，一般行列用直接法 .

- 疎行列データは圧縮列格納 .

- 真の解が  $x = 1$  となるように  $A$  の各行和を右辺  $b$  に設定 . 加算により発生する丸め誤差は無視 . 精度を最大値ノルムで計算 .

- サブルーチンの前後に時間計測関数を挿入し，経過時間を測定 .

# 正定值对称行列 (速度・秒)

名称	次元数	非零要素数	WSMP	DSRIS <sup>-10</sup>	DSRIS <sup>-6</sup>	DGESV
body6.RSA	19366	77057	0.159	0.205	0.130	63.162
bcsstk36.RSA	23052	583096	0.510	—	66.587	97.816
msc23052.RSA	23052	588933	0.514	—	89.271	97.329
THREAD.rsa	29736	2249892	3.113	—	—	204.202
GRIDGENA.rsa	48962	329485	—	—	—	951.616
nasasrb.RSA	54870	1366097	1.201	109.20	42.414	1370.299
crankseg_2.rsa	63838	7106348	6.215	43.241	31.185	2064.227
OILPAN.rsa	73752	1835470	1.398	80.049	47.560	3044.869
finan512.RSA	74752	335872	0.675	0.164	0.129	3456.823
s3dkq4m2.rsa	90449	2455670	2.005	172.90	147.08	—
M_T1.rsa	97578	4925574	4.061	4997.9	1956.1	—
X104.rsa	108384	5138004	5.039	9495.7	2443.5	—
cfd2.RSA	123440	1605669	2.817	22.767	14.233	—
bmwcra_1.rsa	148770	5396386	5.892	202.61	174.55	—
SHIPSEC5.rsa	179860	5146478	5.366	140.18	89.481	—
hood.rsa	220542	5494489	4.243	60.943	31.676	—
pwtk.RSA	217918	5926171	4.917	2496.6	582.79	—
inline_1.rsa	503712	18660027	18.556	4365.5	1316.5	—
audikw_1.rsa	943695	39297771	116.59	2658.5	1980.4	—
ldoor.rsa	952203	23737339	22.438	625.83	370.67	—

# 正定值对称行列 (精度)

名称	次元数	非零要素数	WSMP	DSRIS <sup>-10</sup>	DSRIS <sup>-6</sup>	DGESV
bodyy6.RSA	19366	77057	0.975E-12	0.66E-08	0.16E-03	0.182E-11
bcsstk36.RSA	23052	583096	0.124E-06	—	0.47E+00	0.848E-03
msc23052.RSA	23052	588933	0.508E-07	—	0.54E+00	0.239E-03
THREAD.rsa	29736	2249892	0.227E-07	—	—	0.595E-07
GRIDGENA.rsa	48962	329485	—	—	—	0.406E-11
nasasrb.RSA	54870	1366097	0.624E-09	0.60E-06	0.26E-01	0.175E-08
crankseg_2.rsa	63838	7106348	0.312E-11	0.12E-07	0.20E-03	0.675E-10
OILPAN.rsa	73752	1835470	0.550E-09	0.10E-06	0.11E+00	0.199E-08
finan512.RSA	74752	335872	0.133E-14	0.71E-09	0.32E-04	0.877E-14
s3dkq4m2.rsa	90449	2455670	0.301E-06	0.45E-06	0.25E-03	—
M_T1.rsa	97578	4925574	0.188E-06	0.13E-04	0.81E-01	—
X104.rsa	108384	5138004	0.270E-05	0.52E-05	0.17E+00	—
cfd2.RSA	123440	1605669	0.136E-11	0.30E-06	0.35E-02	—
bmwcra_1.rsa	148770	5396386	0.702E-10	0.34E-06	0.33E+00	—
SHIPSEC5.rsa	179860	5146478	0.178E-08	0.47E-06	0.15E-02	—
pwtk.RSA	217918	5926171	0.100E-06	0.71E-03	0.26E+01	—
hood.rsa	220542	5494489	0.136E-10	0.28E-06	0.76E-02	—
inline_1.rsa	503712	18660027	0.243E-09	0.37E-06	0.13E+01	—
audikw_1.rsa	943695	39297771	0.205E-10	0.18E-06	0.15E-01	—
ldoor.rsa	952203	23737339	0.195E-10	0.45E-05	0.97E-02	—

# 性能比較・不定値対称行列

# テスト行列

名称	次元数	非零要素数	分野
SPMSRTLS.rsa	29995	129971	Sparse matrix square root
HELM3D01.rsa	32226	230335	Helmholtz problem
bcstk39.RSA	46772	1068033	shuttle rocket booster
SPARSINE.rsa	50000	799494	Structural optimization
copter2.rsa	55476	407714	Helicopter rota blade
DIXMAANL.rsa	60000	179999	Dixon-Maany optimization
c-71.RSA	76638	468096	Optimization model
A0NSDSIL.rsa	80016	200021	Linear Complementarity problem
c-72.RSA	84064	395811	Optimization model
bmw3_2.rsa	227362	5757996	Linear static analysis
HELM2D03.rsa	392257	1567096	Helmholtz problem

# 解法と測定条件

- WSMP 5.3.15
- IBM ESSL V4.2
  - DSRIS  
対角スケーリング付きTFQMR法，疎行列用反復法  
 $x^0 = 0$ ，反復停止基準： $\|b - Ax^n\|_2 / \|x^n\|_2 < 10^{-10}$ 。
  - DGESV  
部分ピボット選択付きLU分解，一般行列用直接法
- 疎行列データは圧縮列格納。
- 真の解が  $x = 1$  となるように  $A$  の各列和を右辺  $b$  に設定．加算により発生する丸め誤差は無視．精度を最大値ノルムで計算．
- サブルーチンの前後に時間計測関数を挿入し，経過時間を測定．

# 不定值对称行列（速度・秒）

名称	次元数	非零要素数	WSMP	DSRIS	DGESV
SPMSRTLS.rsa	29995	129971	0.211	—	233.297
HELM3D01.rsa	32226	230335	0.683	—	285.757
bcsstk39.RSA	46772	1068033	0.872	—	891.506
SPARSINE.rsa	50000	799494	—	—	1003.262
copter2.rsa	55476	407714	0.926	—	1501.076
DIXMAANL.rsa	60000	179999	—	0.046	1760.508
c-71.RSA	76638	468096	4.839	4.085	3924.913
A0NSDSL.rsa	80016	200021	—	—	4183.087
c-72.RSA	84064	395811	7.870	7.510	—
bmw3_2.rsa	227362	5757996	5.375	—	—
HELM2D03.rsa	392257	1567096	3.670	—	—



# 不定值对称行列（精度）

名称	次元数	非零要素数	WSMP	DSRIS	DGESV
SPMSRTLS.rsa	29995	129971	0.103E-11	—	0.104E-11
HELM3D01.rsa	32226	230335	0.193E-11	—	0.255E-10
bcsstk39.RSA	46772	1068033	0.238E-10	—	0.273E-09
SPARSINE.rsa	50000	799494	—	—	0.124E-07
copter2.rsa	55476	407714	0.254E-12	—	0.121E-09
DIXMAANL.rsa	60000	179999	—	0.13E-07	0.128E-10
c-71.RSA	76638	468096	0.143E-11	0.25E-05	0.119E-09
A0NSDSLIL.rsa	80016	200021	—	—	0.194E-09
c-72.RSA	84064	395811	0.159E-11	0.28E-05	—
bmw3_2.rsa	227362	5757996	0.440E-06	—	—
HELM2D03.rsa	392257	1567096	0.285E-12	—	—

# 性能比較・非対称行列

# テスト行列

名称	次元数	非零要素数	分野
west2021	2021	7353	Chemical engineering
fidap011	16614	1096948	Fluid dynamics
e40r0100	17281	553562	Fluid dynamics
memplus	17758	126150	Electronic circuit design
raefsky3	21200	1488768	Fluid dynamics
af23560	23560	484256	Fluid dynamics
wang3	26064	177168	Fluid dynamics
lhr34c	35152	764014	Chemical engineering
onetone1	36057	341088	Nonlinear circuits
onetone2	36057	227628	Nonlinear circuits
bbmat	38744	1771722	Fluid dynamics
av41092	41092	1683902	Finite Element Analysis
bayer01	57735	277774	Chemistry
venkat50	62424	1717792	Fluid dynamics
epb3	84617	463625	Thermodynamics
twotone	120750	1224224	Nonlinear circuits
torso3	259156	4429042	Fluid dynamics
languag	399130	1216334	Natural-language processing
pre2	659033	5959282	Nonlinear circuits
hamrle3	1447360	5514242	Electronic circuit design

# 解法と測定条件

- WSMP 5.3.15
- IBM ESSL V4.2
  - DGSF(S)  
修正 Markowitz 法による  $LU$  分解，疎行列用直接法
  - DSRIS  
不完全  $LU$  分解付き BiCGStab 法，疎行列用反復法， $x^0 = 0$ ，反復停止基準:  $\|b - Ax^n\|_2 / \|x^n\|_2 < 10^{-10}$ .
  - DGESV  
部分ピボット選択付き  $LU$  分解，一般行列用直接法.
- 疎行列データは圧縮列格納， $A^T x = b$  を解く.
- 真の解が  $x = 1$  となるように  $A^T$  の各行の列和を右辺  $b$  に設定．加算により発生する丸め誤差は無視．精度を最大値ノルムで計算．
- サブルーチンの前後に時間計測関数を挿入し，経過時間を測定．

# 非对称行列 (速度・秒)

名称	次元数	非零要素数	WSMP	DGSF(S)	DSRIS	DGESV
west2021	2021	7353	0.03	0.003	—	0.12
fidap011	16614	1096948	1.36	625.88	—	37.71
e40r0100	17281	5535562	0.57	51.55	—	39.30
memplus	17758	126150	0.38	0.05	1.25	44.75
raefsky3	21200	1488768	1.38	69.33	5.38	70.34
af23560	23560	484256	1.20	3044	—	102.27
wang3	26064	177168	1.41	603.93	0.31	129.75
lhr34c	35152	764014	1.51	17.32	—	317.53
onetone1	36057	341088	1.86	2.88	—	381.07
onetone2	36057	227628	0.94	0.78	—	386.70
bbmat	38744	1771722	4.62	—	—	441.22
av41092	41092	1683902	14.09	10.48	—	561.83
bayer01	57735	277774	1.15	0.18	—	1583.33
venkat50	62424	1717792	1.70	270.69	8.56	1859.18
epb3	84617	463625	1.79	20.70	1.68	—
twotone	120750	1224224	4.42	12.84	—	—
torso3	259156	4429042	37.07	344.66	4.09	—
languag	399130	1216334	—	0.68	0.77	—
pre2	659033	5959282	34.47	—	—	—
hamrie3	1447360	5514242	1313	—	—	—

# 非对称行列 (精度)

名称	次元数	非零要素数	WSMP	DGSF(S)	DSRIS	DGESV
west2021	2021	7353	0.206E-08	0.265E-07	—	0.448E-07
fidap011	16614	1096948	0.192E-05	0.279E-01	—	0.571E-04
e40r0100	17281	553562	0.478E-11	0.895E-06	—	0.338E-08
memplus	17758	126150	0.233E-11	0.103E-07	0.771E-05	0.207E-10
raefsky3	21200	1488768	0.249E-12	0.286E-00	0.542E-08	0.316E-08
af23560	23560	484256	0.712E-13	0.161E-04	—	0.119E-12
wang3	26064	177168	0.448E-13	0.416E-05	0.905E-08	0.253E-12
lhr34c	35152	764014	0.176E-01	0.289E+03	—	0.105E+01
onetone1	36057	341088	0.314E-10	0.749E-09	—	0.751E-09
onetone2	36057	227628	0.991E-11	0.980E-10	—	0.137E-09
bbmat	38744	1771722	0.100E-10	—	—	0.379E-08
av41092	41092	1683902	0.840E-13	0.717E-07	—	0.864E-12
bayer01	57735	277774	0.790E-06	0.211E+03	—	0.757E-06
venkat50	62424	1717792	0.321E-12	0.624E-05	0.248E-07	0.204E-11
epb3	84617	463625	0.281E-13	0.640E-08	0.381E-07	—
twotone	120750	1224224	0.932E-10	0.414E-08	—	—
torso3	259156	4429042	0.444E-14	0.561E-06	0.240E-07	—
languag	399130	1216334	—	0.596E-11	0.179E-07	—
pre2	659033	5959282	0.224E-04	—	—	—
hamrle3	1447360	5514242	0.206E-00	—	—	—

# 知見

## 「ほぼ頑強 (nearly robust)」

- ESSLの（ピーク性能比70%を達成する）直接解法と比較して、速度・精度の面で勝る。
  - Orderingによる fill-in の抑止
  - 疎行列構造を生かした行列分解アルゴリズム
- 他の手法と比較して、精度の面で勝る。
  - 直接法の性質と計算量削減によって丸め誤差の影響が  
少ないため？
- 行列の構造が同じだったり、多くの右辺を持つ場合には特に有効。
- 比較の解法としても利用価値大（速度比較、解き難さの確認, etc.）。

# WSMPの問題点

「頑強 (robust)」とはまだ言えない

- 行列のサイズが大きくなると4バイト整数の表現可能な値 (-2147483648 ~ 2147483647) を超え内部エラーとなる。
  - 8バイト版の提供待ち。
- 4並列以上の並列化性能が芳しくない。
- 対称行列はヒポット選択を行なわない。
  - 対角成分に0がある行列に対しては利用者が事前に置換を行なう必要がある。
- 分解された行列の格納方法が不明。
- マニュアルが読みにくい。



# 問題募集！

- 100 万元以上歓迎
- 圧縮行格納法 (compressed row storage)
- 圧縮列格納法 (compressed column storage)

watanabe@cc.kyushu-u.ac.jp までお気軽に .

# 誤差評価 1

定理:  $Ax = b$  の近似解  $\tilde{x}$  と  $A$  の逆行列  $R$  が求められたとき,  $\|RA - I\|_\infty < 1$  ならば  $A$  は正則であり,

$$\|x - \tilde{x}\|_\infty \leq \frac{\|R(b - A\tilde{x})\|_\infty}{1 - \|RA - I\|_\infty}.$$

ただし  $\|\cdot\|_\infty$  は最大値ノルム.

$$R = (LU)^{-1}$$

# 誤差評価 2

特異値分解:

$$A = U\Sigma V^T$$

$$UU^T = VV^T = I, \Sigma = \text{diag}(\sigma_1, \dots, \sigma_n), U = (\mathbf{u}_1, \dots, \mathbf{u}_n)$$

$$\|\mathbf{x} - \tilde{\mathbf{x}}\|_2^2 = \sum_{i=1}^n \frac{1}{\sigma_i^2} (\mathbf{u}_i, \mathbf{b} - A\tilde{\mathbf{x}})^2$$

$$\Rightarrow \|\mathbf{x} - \tilde{\mathbf{x}}\|_2 \leq \frac{1}{\sigma_{\min}} \|\mathbf{b} - A\tilde{\mathbf{x}}\|_2$$

$A$  の (特に最小) 特異値を計算する必要がある。通常, 特異値計算は  $A\mathbf{x} = \mathbf{b}$  を解く以上の手間が必要。