FreeFem++ workshop 2016, 08 Dec.

# Direct solver and domain decomposition preconditioner for indefinite finite element matrices

Atsushi Suzuki[1]

[1] Cybermedia Center, Osaka University
atsushi.suzuki@cas.cmc.osaka-u.ac.jp
http://www.ljll.math.upmc.fr/∼suzukia/

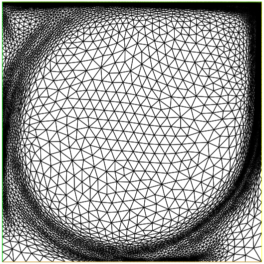
## outlines

- examples of indefinite finite element stiffness matrix
  sparse symmetric/unsymmetric, singular
- overview of sparse direct solver
- pivoting strategy to solve indefinite and/or singular matrix
- parallel efficiency of matrix from 3D N-S eqs. on super-scalar and vector CPUs
- coarse space to accelerate convergence of iterative solver
- conclusion

## indefinite and singular matrix in solving PDE

2D stationary cavity driven flow problem in $(0,1)\times(0,1)$

$\mu$ : viscosity coefficient



$$-2\mu\nabla\cdot D(u)+u\cdot\nabla u+\nabla p=0 \text{ in } \Omega,$$
$$\nabla\cdot u=0 \text{ in } \Omega,$$
$$u=g \text{ on } \partial\Omega.$$

$g=[4x(1-x)\ \ 0]^T$ on the top,

$g=0$ elsewhere.

$(u,p)$ : sol. $\Rightarrow$ $(u,p+1)$ : sol.

Newton iteration to solve nonlinear system

$$K\begin{bmatrix}\vec{u}^n\\ \vec{p}^n\end{bmatrix}=\begin{bmatrix}A(\vec{u}^{n-1}) & B^T\\ B & 0\end{bmatrix}\begin{bmatrix}\vec{u}^n\\ \vec{p}^n\end{bmatrix}=\begin{bmatrix}\vec{f}^n\\ \vec{0}\end{bmatrix}\qquad \text{Ker}K=\begin{bmatrix}\vec{0}\\ \vec{1}\end{bmatrix}$$

find $u\in\{H^1(\Omega); u=g \text{ on } \partial\Omega\}$, $p\in L^2_0(\Omega)$ is not implemented.

$$L^2_0(\Omega)=\{p\in L^2(\Omega)\,;\,\int_\Omega p=0\}$$

- fixing one point for pressure
- penalization for pressure
- matrix solver detects the kernel

## indefinite matrix from electro-magnetic problem

constraints on the external force: $\nabla \cdot f = 0$ in $\Omega \subset \mathbb{R}^3$.

$$\begin{aligned} \nabla \times (\nabla \times u) &= f && \text{in } \Omega, \\ \nabla \cdot u &= 0 && \text{in } \Omega, \\ u \times n &= 0 && \text{on } \partial\Omega. \end{aligned}$$

$H_0(\mathrm{curl}\,;\,\Omega) = \{u \in L^2(\Omega)^3\,;\,\nabla \times u \in L^2(\Omega)^3\,;\,u \times n = 0\}$

find $(u, p) \in H_0(\mathrm{curl}\,;\,\Omega) \times H_0^1(\Omega)$

$$\begin{aligned} (\nabla \times u, \nabla \times v) + (v, \nabla p) &= (f, v) && \forall v \in H_0(\mathrm{curl}\,;\,\Omega) \\ (u, \nabla q) &= 0 && \forall q \in H_0^1(\Omega) \end{aligned}$$

has a unique solution.

- $(\nabla \times \cdot, \nabla \times \cdot)$ : coercive on $W$ ,
  $W = H_0(\mathrm{curl}\,;\,\Omega) \cap \{u \in H(\mathrm{div};\Omega)\,;\,\mathrm{div}\; u = 0\}$.
- stiffness matrix is symmetric but indefinite.
- $H_0(\mathrm{curl}\,;\,\Omega) = \mathrm{grad} H_0^1(\Omega) \oplus W$.

## indefinte and singular from electro-magnetic problem

constraints on the external force: $\nabla \cdot f = 0$ in $\Omega$.

$$
\begin{aligned}
\nabla \times (\nabla \times u) &= f && \text{in } \Omega, \\
\nabla \cdot u &= 0 && \text{in } \Omega, \\
(\nabla \times u) \times n &= 0 && \text{on } \partial\Omega, \\
u \cdot n &= 0 && \text{on } \partial\Omega.
\end{aligned}
$$

$L^2_0(\Omega) = \{p \in L^2(\Omega) \,;\, (p, 1) = 0\}$

find $(u, p) \in H(\mathrm{curl}\,;\,\Omega) \times \{H^1(\Omega) \cap L^2_0(\Omega)\}$

$$
\begin{aligned}
(\nabla \times u, \nabla \times v) + (v, \nabla p) &= (f, v) && \forall v \in H(\mathrm{curl}\,;\,\Omega) \\
(u, \nabla q) &= 0 && \forall q \in H^1(\Omega) \cap L^2_0(\Omega)
\end{aligned}
$$

finite element approximation : Nédélec element of degree 0 and P1

$$N_0(K) = (P_0(K))^3 \oplus [x \times (P_0(K))^3], \qquad P_1(K)$$

$$\ker \begin{bmatrix} A & B^T \\ B & 0 \end{bmatrix} = \begin{bmatrix} \vec{0} \\ \vec{1} \end{bmatrix}, \quad \exists A^{-1} \text{ on } \ker B$$

not easy problem for usual direct solvers

## semi-conductor problem with Drift-Diffusion model : 1/3

hole concentration $p$ : unknown
potential $\varphi$ : given $\log(n_i/n_d)$ in N-region, $\log(n_a/n_i)$ in P.

$$-\text{div}(\nabla p + p\nabla\varphi) = 0 \text{ in } \Omega$$
$$p = g \text{ on } \Gamma_D$$
$$\partial_\nu p = 0 \text{ on } \Gamma_N$$

following Maxwell-Boltzman statistics : $p = n_i \text{exp}(\frac{\varphi_p - \varphi}{V_{th}})$

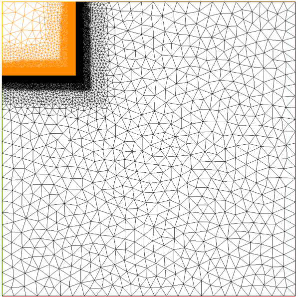
- $\varphi_p$ : quasi-Fermi level
- $n_i$ : intrinsic concentration of the semiconductor
- $V_{\text{th}} = K_B T/q$ : thermal voltage
- $K_B$ : Boltzmann constant
- $q$ : positive electron charge
- $T$ : lattice temperature

## semi-conductor problem with Drift-Diffusion model : 1/3

hole concentration $p$ : unknown
potential $\varphi$ : given $\log(n_i/n_d)$ in N-region, $\log(n_a/n_i)$ in P.

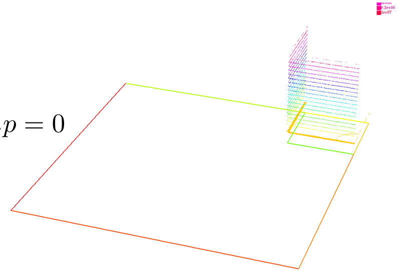
$$-\text{div}(\nabla p + p\nabla\varphi) = 0 \text{ in } \Omega$$
$$p = g \text{ on } \Gamma_D$$
$$\partial_\nu p = 0 \text{ on } \Gamma_N$$

$p = n_i/n_d$ $\quad$ $\partial_\nu p = 0$



$\partial_\nu p = 0$

$p = n_a/n_i$



## semi-conductor problem with Drift-Diffusion model : 2/3

Slotboom variable $\xi$ : $p = \xi e^{-\varphi}$ $\quad -J_p = \nabla p + p\nabla\varphi = \nabla \xi e^{-\varphi}$

$$-\mathrm{div}(-J_p) = 0 \text{ in } \Omega$$
$$-J_p e^{\varphi} = \nabla \xi \text{ in } \Omega$$

function space :

$$H(\mathrm{div}) = \{\tau \in L^2(\Omega)^2\,;\, \mathrm{div}\,\tau \in L^2(\Omega)\},$$
$$\Sigma = \{\tau \in H(\mathrm{div})\,;\, \tau\cdot\nu = 0 \text{ on } \Gamma_N\}$$

integration by parts leads to

$$-\int_\Omega e^{\varphi} J_p \cdot \tau = \int_\Omega \nabla\xi\cdot\tau = -\int_\Omega \xi\nabla\cdot\tau + \int_{\partial\Omega} \xi\tau\cdot\nu$$

F. Brezzi, L. D. Marini, S. Micheletti, P. Pietra, R. Sacco, S. Wang.
Discretization of semiconductor device problems (I) F. Brezzi et al.,
Handbook of Numerical Anasysis vol XIII, Elsevier 2005

hybridization of mixed formulation + mass lumping $\Rightarrow$ FVM

mixed formulation + higher order approximation $\Rightarrow$ indefinite matrix

## semi-conductor problem with Drift-Diffusion model : 2/3

mixed-type weak formulation

find $(J_p, \xi) \in \Sigma \times L^2(\Omega)$

$$\int_\Omega e^\varphi J_p \cdot \tau - \int_\Omega \xi \nabla \cdot \tau = -\int_{\Gamma_D} g e^\varphi \tau \cdot \nu \qquad \forall \tau \in \Sigma$$

$$\int_\Omega \nabla \cdot J_p v = 0 \qquad \forall v \in L^2(\Omega)$$

symmetric indefinte

replacing $\xi = e^\varphi p$ again,

find $(J_p, p) \in \Sigma \times L^2(\Omega)$

$$\int_\Omega e^\varphi J_p \cdot \tau - \int_\Omega e^\varphi p \nabla \cdot \tau = -\int_{\Gamma_D} g e^\varphi \tau \cdot \nu \qquad \forall \tau \in \Sigma$$

$$\int_\Omega \nabla \cdot J_p v = 0 \qquad \forall v \in L^2(\Omega)$$

unsymmetic indefintie

cf. exponential fitting with FVM

Ravier-Thomas element for $H(\mathrm{div})$ $\quad RT1(K) = (P1(K))^2 + \vec{x}P1(K)$,
picewsie linear element for $L^2(\Omega)$ $\quad P1(K)$.

## abstract framework

$V$: Hilbert space with inner product $(\cdot,\cdot)$ and norm $||\cdot||$.
bilinear form $a(\cdot,\cdot) : V \times V \to \mathbb{R}$

- continuous : $\exists \gamma > 0 \quad |a(u,v)| \leq \gamma ||u|| \, ||v|| \; \forall u,\, v \in V$.
- $\exists \alpha_1 > 0 \quad \sup_{v \in V, v \neq 0} \dfrac{a(u,v)}{||v||} \geq \alpha_1 ||u|| \; \forall u \in V$.
- $\exists \alpha_2 > 0 \quad \sup_{u \in V, u \neq 0} \dfrac{a(u,v)}{||u||} \geq \alpha_2 ||v|| \; \forall v \in V$.

find $u \in V$ s.t. $a(u,v) = F(v) \quad \forall v \in V$ has a unique solution.

$\forall U \subset V$ subspace

$$\text{find } u \in U \text{ s.t. } a(u,v) = F(v) \quad \forall v \in U$$

in general, inf-sup condition in subspace $U$ is unclear.
in discretized problem : $V_h \subset V$ ?
in linear solver (subspace of $V_h$) ?

## State of the art : software for sparse direct solver

| Software | parallel env. | elimination strategy | data manag. | pivoting | kernel detection |
|---|---|---|---|---|---|
| UMFPACK | —- | multi-frontal | static | yes | no |
| SuperLU_MT | shared | super-nodal | dynamic | yes | no |
| Pardiso | shared | super-nodal | dynamic | yes + $\sqrt{\varepsilon}$-p. | no |
| SuperLU_DIST | distributed | super-nodal | static | no, $\sqrt{\varepsilon}$-p. | no |
| MUMPS | distributed | multi-frontal | dynamic | yes | yes |
| Dissection | shared | multi-frontal | static | yes | yes |

T. A. Davis, I. S. Duff. A combined unifrontal/multifrontal method for unsymmetric sparse matrices,
*ACM Trans. Math. Software*, 25 (1999), 1–20.

J. W. Demmel, S. C. Eisenstat, J. R. Gilbert, X. S. Li, J. W. H. Liu.
A supernodal approach to sparse partial pivoting,
*SIAM J. Matrix Anal. Appl.*, 20 (1999), 720–755.

O. Schenk, K. Gärtner. Solving unsymmetric sparse systems of liner equations with PARDISO,
*Future Generation of Computer Systems*, 20 (2004), 475–487.

X. S. Li, J. W. Demmel. SuperLU_DIST : A scalable distributed-memory sparse direct solver for unsymmetric linear systems,
*ACM Trans. Math. Software*, 29 (2003), 110–140.

P. R. Amestoy, I. S. Duff, J.-Y. L'Execellent. Mutlifrontal parallel distributed symmetric and unsymmetric solvers,
*Comput. Methods Appl. Mech. and Engrg*, 184 (2000) 501–520.

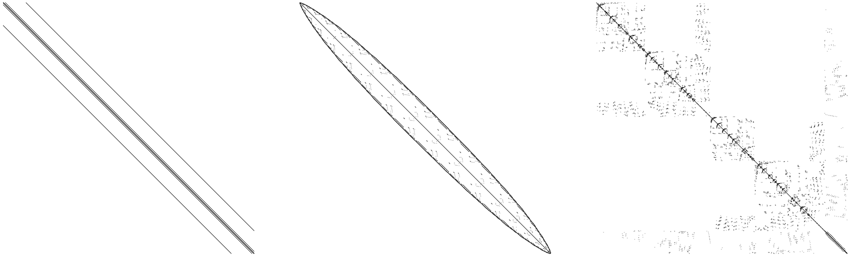
A. Suzuki, F.-X. Roux, A dissection solver with kernel detection for symmetric finite element matrices on shared memory computers,
*Int. J. Numer. Meth. in Engng*, 100 (2014) 136–164.

## ordering of sparse matrix

sparse matrix needs to be re-ordered

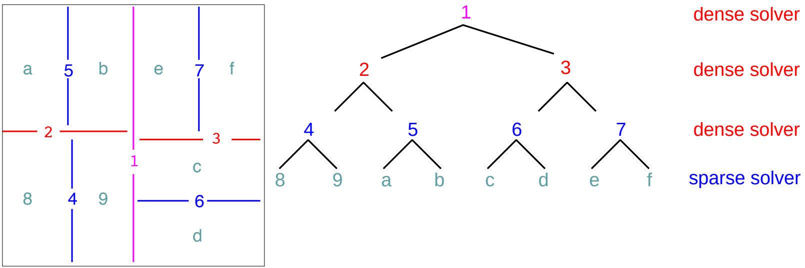
- to reduce fill-in
- to increase parallelization of factorization $\rightarrow$ multi-front
- to increase size of block structure $\rightarrow$ supernode

example:
7 stencil of Poisson equation, $11^3$ nodes.



original matrix    reverse Cuthill-McKee    nested-dissection (3 layers)

## nested-dissection by graph decomposition




A. George. Numerical experiments using dissection methods to solve n by n grid problems. SIAM J. Num. Anal. 14 (1977),161–179.

software package:

METIS : V. Kumar, G. Karypis, A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM J. Sci. Comput. 20 (1998) 359–392.

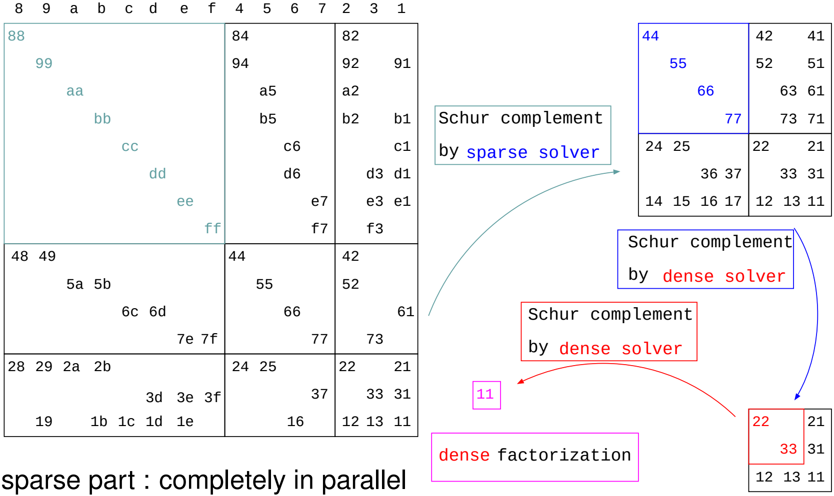
SCOTCH : F. Pellegrini J. Roman J, P. Amestoy, Hybridizing nested dissection and halo approximate minimum degree for efficient sparse matrix ordering. Concurrency: Pract. Exper. 12 (2000) 69–84.

- each leaf can be computed in parallel ⇐ multi-front
- load unbalance? because of different size of subdomains
- parallel computation of higher levels?
  \# cores > # subdomains

# recursive generation of Schur complement

$$\begin{bmatrix} A_{11} & A_{21} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & 0 \\ A_{21}A_{11}^{-1} & S_{22} \end{bmatrix} \begin{bmatrix} I_1 & A_{11}^{-1}A_{12} \\ 0 & I_2 \end{bmatrix}$$

$S_{22} = A_{22} - A_{21}A_{11}^{-1}A_{12} = A_{22} - (A_{21}U_{11}^{-1})D_{11}^{-1}L_{11}^{-1}A_{12}$ : recursively computed



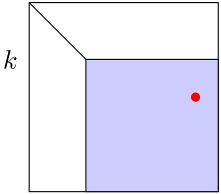

sparse part : completely in parallel

dense part : better use of BLAS 3; dgemm, dtrsm

## pivoting strategy

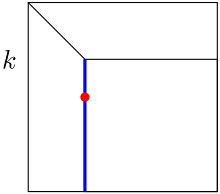
full pivoting : $A = \Pi_L^T L U \Pi_R$

find $\max_{k<i,j\leq n} |A(i,j)|$



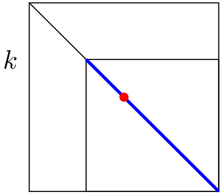

partial pivoting : $A = \Pi L U$

find $\max_{k<i\leq n} |A(i,k)|$



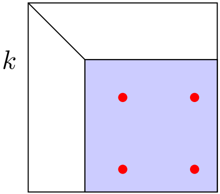

symmetric pivoting : $A = \Pi^T L D U \Pi$

find $\max_{k<i\leq n} |A(i,i)|$




$2 \times 2$ pivoting : $A = \Pi^T \, L \, \tilde{D} \, U \Pi$

find $\max_{k<i,j\leq n} \det \begin{vmatrix} A(i,i) & A(i,j) \\ A(j,i) & A(j,j) \end{vmatrix}$




sym. pivoting is mathematically not always possible $\rightarrow$ $2 \times 2$ pivoting

## understanding pivoting strategy by solution in subspaces

$A = \Pi^T LDU\Pi$: symmetric pivoting
$D$: diagonal, $L$: lower triangle, $L(i,i) = 1$, $U$: upper tri., $U(i,i) = 1$.

- index set $\{i_1, i_2, \cdots, i_m\}$
- $V_m =$ span$[\vec{e}_{i_1}, \vec{e}_{i_2}, \cdots, \vec{e}_{i_m}] \subset \mathbb{R}^N$
- $P_m : \mathbb{R}^N \to V_m$ orthogonal projection.

find $\vec{u} \in V_m \quad (A\vec{u} - \vec{f}, \vec{v}) = 0 \quad \forall v \in V_m$.

$\exists \Pi : A = \Pi^T L\, D\, U \Pi$
$\Rightarrow \exists\{i_1, i_2, \cdots, i_N\}$ s.t. $\quad P_m A\, P_m^T$ : invertible on $V_m \quad 1 \leq \forall m \leq N$.

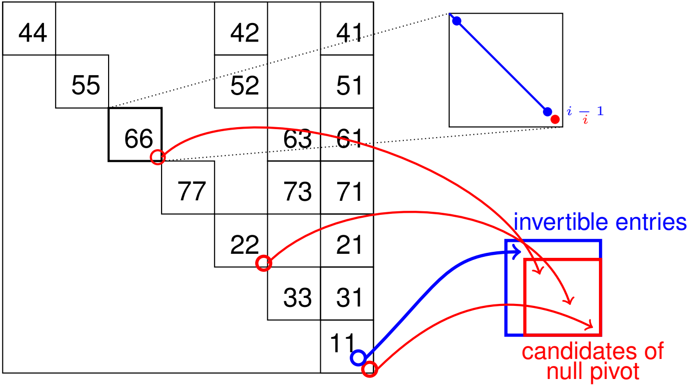
$2 \times 2$ pivoting: $\quad V_{m-1}, \quad V_m, \quad V_{m+1}, V_{m+2}, V_{m+3},$ by skipping $V_{m+1}$.

J. R. Bunch, L. Kaufman. Some stable methods for calculating inertia and solving symmetric linear systems,
*Math. Comput*, 31 (1977) 163–179.
R. Bank, T.-F. Chan. An analysis of the composite step biconjugate gradient method.
*Numer. Math*, 66 (1993) 295–320.

## symmetric pivoting with postponing for block strategy

- nested-dissection decomposition may produce singular sub-matrix for indefinite matrix

$\tau$ : given threshold for null pivot
$|A(i,i)|/|A(i-1,i-1)| < \tau \;\Rightarrow |A(i,i)|$ is null pivot.




Schur complement matrix from suspicious (postponed) null pivots and additional nodes $\Rightarrow$ kernel detection algorithm

## kernel detection (rank deficient problem)

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & 0 \\ A_{21} & S_{22} \end{bmatrix} \begin{bmatrix} I_1 & A_{11}^{-1} A_{12} \\ 0 & I_2 \end{bmatrix} \quad S_{22} = 0 \Rightarrow \mathrm{Ker} A = \begin{bmatrix} A_{11}^{-1} A_{12} \\ -I_2 \end{bmatrix}$$

symmetric semi-positive definite, $m + k = 4 + 6 = 10$

by Householder-QR factorization:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 4.60e-02 | -1.20e-02 | 2.91e-03 | 1.16e-02 | 2.24e-02 | -9.33e-05 | -3.60e-02 | 8.22e-03 | -7.77e-03 | -2.90e-02 |
| 0.0 | 3.84e-02 | 4.84e-03 | -2.21e-02 | 1.87e-02 | 1.30e-03 | -9.14e-03 | -2.74e-02 | 1.48e-02 | -1.91e-02 |
| 0.0 | 0.0 | 2.96e-02 | 1.68e-03 | -2.55e-02 | 1.11e-04 | 1.28e-02 | -1.04e-04 | -1.12e-03 | -1.20e-02 |
| 0.0 | 0.0 | 0.0 | 1.28e-02 | -1.66e-03 | 8.48e-04 | -4.29e-05 | -7.90e-04 | -8.56e-03 | 2.51e-03 |
| 0.0 | 0.0 | 0.0 | 0.0 | 1.23e-11 | -5.49e-13 | -8.30e-12 | 1.67e-13 | 2.10e-14 | -7.08e-12 |
| 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 6.70e-13 | -1.02e-13 | -5.33e-13 | -1.62e-13 | 1.18e-13 |
| 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 3.33e-13 | -2.48e-14 | -6.61e-14 | -3.18e-13 |
| 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 1.22e-13 | 4.46e-15 | -4.34e-14 |
| 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 3.05e-14 | 8.16e-15 |
| 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | 0.0 | -1.09e-14 |

how to set threshold to distinguish between
non-zero (1.28e-02) and zero (1.23e-11) values ?

| | |
|---|---|
| Pardiso | no capability of kernel detection. |
| MUMPS | user has to choose this value. |
| Dissection | + an algorithm by measuring dimension of residual of matrix with a projection onto the image space. |

## kernel detection algorithm based on $LDU$ : 3/4

$A : N \times N$ unsymmetric, $\mathrm{dimKer} A = k \geq 1$, $\mathrm{dimIm} A \geq m$.
two parameters: $l$, $n$, which define size of factorization,

$$\begin{array}{c} N-n \updownarrow \\ n \updownarrow \end{array} \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & 0 \\ A_{21} & S_{22} \end{bmatrix} \begin{bmatrix} I_1 & A_{11}^{-1} A_{12} \\ 0 & I_2 \end{bmatrix} \quad \widetilde{\mathrm{Im}}_n = \mathrm{span} \begin{bmatrix} \widetilde{A_{11}^{-1}} A_{12} \\ -I_2 \end{bmatrix}^{\perp}.$$

- projection : $P_n^{\perp} : \mathbb{R}^N \rightarrow \widetilde{\mathrm{Im}}_n$
- solution in subspace, $\widetilde{A}_{N-l}^{\dagger} b = \begin{bmatrix} \widetilde{A_{11}^{-1}} b_1 \\ 0 \end{bmatrix}$, $b = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \quad \begin{array}{l} \updownarrow N-l \\ \updownarrow l \end{array}$

Theoretically $\neg A_{N-k+1}^{-1}$
perturbed solution with machine epsilon of double precision $\varepsilon_0$

$$\widetilde{A_{11}}^{-1} b_1 = U_{11}^{-1} D_{11}^{-1} L_{11}^{-1} b_1 + \varepsilon_0 e_m$$

$$\mathrm{err}_l^{(n)} := \max \left\{ \max_{x=[0\ x_l]\neq 0} \frac{||P_n^{\perp}(\widetilde{A}_{N-l}^{\dagger} A\, x - x)||}{||x||}, \max_{x=[x_{N-l}\ 0]\neq 0} \frac{||\widetilde{A}_{N-l}^{\dagger} A\, x - x||}{||x||} \right\}$$

$$\begin{array}{lllllll} n = k+1 & \Leftrightarrow & \mathrm{err}_k^{(k+1)} \approx 0 & \wedge & \mathrm{err}_{k+1}^{(k+1)} \approx 0 & \wedge & \mathrm{err}_{k+2}^{(k+1)} \sim 1 \\ n = k & \Leftrightarrow & \mathrm{err}_{k-1}^{(k)} \gg 0 & \wedge & \mathrm{err}_k^{(k)} \approx 0 & \wedge & \mathrm{err}_{k+1}^{(k)} \sim 1 \\ n = k-1 & \Leftrightarrow & \mathrm{err}_{k-2}^{(k-1)} \gg 0 & \wedge & \mathrm{err}_{k-1}^{(k-1)} \gg 0 & \wedge & \mathrm{err}_k^{(k-1)} \sim 1 \end{array}$$

## Exchange of $1\times 1$ and $2\times 2$ pivot entries

$$B=\begin{bmatrix}1&&\\l_2&1&\\l_3&0&1\end{bmatrix}\begin{bmatrix}d_1&&\\&d_2&d_0\\&d_0&d_3\end{bmatrix}\begin{bmatrix}1&u_2&u_3\\&1&0\\&&1\end{bmatrix}=\begin{bmatrix}d_1&d_1u_2&d_1u_3\\d_1l_2&d_2+d_1l_2u_2&d_0+d_1l_2u_3\\d_1l_3&d_0+d_1l_3u_2&d_3+d_1l_3u_3\end{bmatrix}$$

find $(i,\ j)$ : $|b_{ii}\cdot b_{jj}-b_{ji}b_{ij}|\geq|b_{kk}\cdot b_{mm}-b_{mk}b_{km}|$ for $2\times 2$ block
$(i,\ j,\ h),(k,\ m\ ,n)\in\{(1,\ 2,\ 3),\ (2,\ 3,\ 1),\ (3,\ 1,\ 2)\}$.

permutation $\Pi(\{1,2,3\})=\{i,j,h\}$,

$$\Pi\,B\,\Pi^T=\begin{bmatrix}1&&\\0&1&\\l'_1&l'_2&1\end{bmatrix}\begin{bmatrix}d'_1&d'_0&\\d'_0&d'_2&\\&&d'_3\end{bmatrix}\begin{bmatrix}1&0&l'_1\\&1&l'_2\\&&1\end{bmatrix}.$$

$d_2\neq 0\Rightarrow$

$$\begin{vmatrix}d_1&d_1u_2\\d_1l_2&d_2+d_1l_2u_2\end{vmatrix}=d_1\cdot(d_2+d_1l_2u_2)-(d_1l_2)(d_2l_2)=d_1d_2\neq 0\,.$$

$d_2=0\wedge d_3=0\Rightarrow d_0\neq 0,$

$$\begin{vmatrix}d_1l_2u_2&d_0+d_1l_2u_3\\d_0+d_1l_3u_2&d_1l_3u_3\end{vmatrix}=d_1l_2u_3\cdot d_1l_3u_2-(d_0+d_1l_2u_3)(d_0+d_1l_3u_2)\neq 0\,.$$

$$d_{-4}d_{-3}d_{-2}(d_{-1}d_0)(d_1d_2)\to d_{-4}d_{-3}(d'_{-2}d'_{-1})d'_0(d_1d_2)\to(d'_{-4}d''_{-3})(d''''_{-2}d''''_{-1})d''''_0d''_1d'_2\,.$$

Kernel detection algorithm assumes dimIm$A\geq m\geq 4$.

## example of kernel detection algorithm

stationary Navier-Stokes equations, $Re = 12,800$, $N = 43,998$,
$\tau = 10^{-2}$, $(m = 4) + 1 + 1$ : kernel $\sim$ pressure ambiguity
$6 \times 6$ matrix by Householder QR factorization

| | | | | | |
|---|---|---|---|---|---|
| 4.221911e-2 | 5.065337e-3 | 5.137137e-3 | 1.493815e-3 | 3.874611e-2 | 1.218166e-2 |
| | 4.060156e-2 | 3.228548e-2 | 5.466190e-3 | 2.174984e-3 | 6.749120e-4 |
| | | 1.389616e-2 | 6.729308e-3 | 8.980537e-3 | 1.813681e-3 |
| | | | 1.708745e-3 | 1.640027e-15 | 6.871814e-1 |
| | | | | 1.203270e-15 | 1.788546e-13 |
| | | | | | 1.674888e-16 |

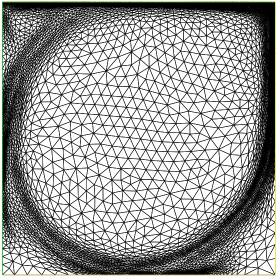
computed residuals with orthogonal projection:

| $k$ | $\text{err}_{k-1}^{(k)}$ | $\text{err}_{k}^{(k)}$ | $\text{err}_{k+1}^{(k)}$ |
|---|---|---|---|
| 2 | $1.57098143 \cdot 10^{-3}$ | $2.69712366 \cdot 10^{-16}$ | $8.11624415 \cdot 10^{-1}$ |

| | |
|---|---|
| $\beta_1$ | $2.22044604 \cdot 10^{-16}$ |
| $\beta_4$ | $8.88178419 \cdot 10^{-16}$ |
| $\beta_6$ | $1.50415172 \cdot 10^{-3}$ |
| $\gamma_0$ | $1.15583524 \cdot 10^{-9}$ |

residual of kernel vectors:

| dim. of kernel = 1 | dim. of kernel = 2 |
|---|---|
| $2.23779349 \cdot 10^{-15}$ | $1.92578081 \cdot 10^{-3}$ |
| | $1.84833445 \cdot 10^{-3}$ |



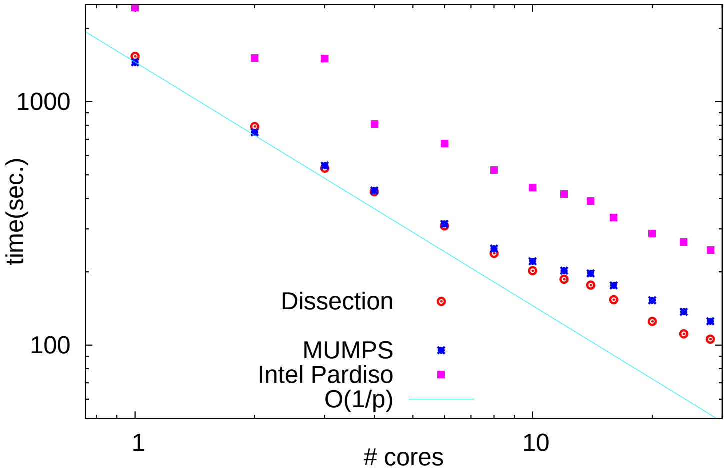
## stiffness matrix of electro-magnetic equations by FreeFem++

FreeFem++ script

```
load "msh3"
load "Dissection"
defaulttoDissection;
mesh3 Th=cube(20,20,20);
fespace VQh(Th, [Edge03d, P1]); // Nedelec element
VQh [u1, u2, u3, p], [v1, v2, v3, q];
varf aa([u1, u2, u3, p], [v1, v2, v3, q]) =
  int3d(Th)((dy(u3)-dz(u2)) * (dy(v3) - dz(v2)) +
            (dz(u1)-dx(u3)) * (dz(v1) - dx(v3)) +
            (dx(u2)-dy(u1)) * (dx(v2) - dy(v1)) +
           dx(p) * v1 + dy(p) * v2 + dz(p) * v3 +
           dx(q) * u1 + dy(q) * u2 + dz(q) * u3);
matrix A = aa(VQh, VQh, solver=sparsesolver,
              tolpivot=1.0e-2,strategy=102);
```

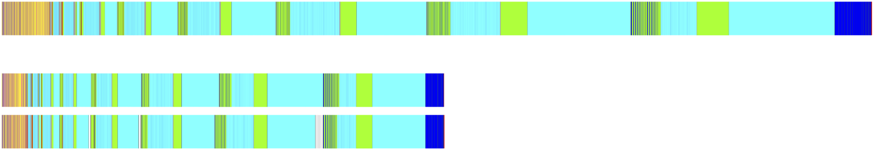
| solver | elapsed time (sec). | algebraic error |
|---|---|---|
| UMFPACK | 32.348 | $3.55790$ |
| Intel Pardiso | 9.698 | $4.07409 \times 10^{-7}$ |
| Dissection | 10.534 | $5.89406 \times 10^{-15}$ |

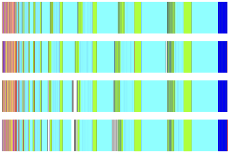
## parallel performance on Xeon E5-2695v3@2.3GHz 14cores $\times 2$




unsymmetric matrix $N = 1,032,183$, $nnz = 97,961,089$, dim ker = 1.
from 3D Navier-Stokes eqs., P2/P1, $h = 1/35$, Re=$300$. 57GB mem.

## parallel performance on Xeon v3





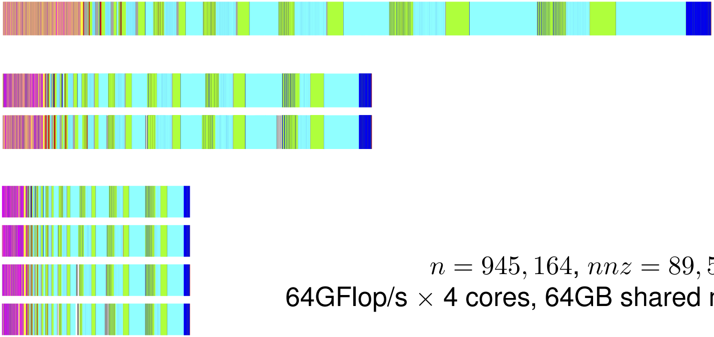
$n = 945,164$, $nnz = 89,588,848$.

36.8GFlop/s $\times$ 28 cores, 64GB shared memory

| purple | yellow | light green | light blue | dark blue |
|---|---|---|---|---|
| sparse LDU | spase Schur | `DTRSM` | `DGEMM` | dense LDU |

| # of cores | CPU time (sec.) | elapsed (sec.) | GFlop/s of `DGEMM` |
|---|---|---|---|
| 1 | 1,268.0 | 1,268.9 | 36.35 |
| 2 | 1,108.3 | 659.39 | 36.27 |
| 4 | 1,178.5 | 356.22 | 34.06 |
| 8 | 1,469.2 | 201.24 | 31.23 |
| 16 | 1,813.2 | 129.63 | 25.25 |
| 28 | 2,002.0 | 94.43 | 22.90 |

## parallel performance on NEC SX-ACE



$n = 945,164$, $nnz = 89,588,848$.
64GFlop/s $\times$ 4 cores, 64GB shared memory

| purple | yellow | light green | light blue | dark blue |
|---|---|---|---|---|
| sparse LDU | spase Schur | `DTRSM` | `DGEMM` | dense LDU |

| # of cores | CPU time (sec.) | elapsed (sec.) | GFlop/s of `DGEMM` |
|---|---|---|---|
| 1 | 1,080.4 | 1,081.9 | 44.85 |
| 2 | 1,108.3 | 590.96 | 43.76 |
| 4 | 1,178.5 | 345.84 | 41.31 |

## Additive Schwarz preconditioner for 3D computation : 1/4

$R_p$ : overlapping decomposition, $D_p$ : a partition of unity (discrete)

$$\sum_{p=1}^{M} R_p^T D_p R_p = I_N,$$

coarse space by Nicolaides
$\{\vec{z}_p\} \subset \mathbb{R}^N$ : basis of coarse space, $Z = [\vec{z}_1, \cdots, \vec{z}_M]$, $R_0 = Z^T$.

$$\vec{z}_p = R_p^T D_p R_p \vec{1},$$

2-level ASM preconditioner

$$Q_{\mathrm{ASM},2}^{-1} = R_0^T (R_0 A R_0^T)^{-1} R_0 + \sum_{p=1}^{M} R_p^T (R_p A R_p^T)^{-1} R_p$$

hybrid version of 2-level ASM preconditioner

$$Q_0 = R_0^T (R_0 A R_0^T)^{-1} R_0, P_0 = I - Q_0 A$$

$$Q_{\mathrm{ASM,hybrid}}^{-1} = Q_0 + P_0^T \sum_{p=1}^{M} R_p^T (R_p A R_p^T)^{-1} R_p P_0$$

cf. V. Dolean, P Jolivet, F. Nataf, An Introduction to Domain Decomposition Methods – Algorithms, Theory, and Parallel Implementation, SIAM, 2015

## Additive Schwarz preconditioner for 3D computation : 2/4

the stiffness matrix restricted on the coarse space
$R_0 A R_0^T$ is invertible for indefinite problem ?
Stokes eqs : coarse space $\Leftarrow$ rigid body modes + pressure constant

penalty-type stabilized finite element method
$V_h \subset V$ : P1 finite element
$Q_h \subset Q$ : P1 finite element + $\int_\Omega p_h \, dx = 0$.
Find $(u_h, p_h) \in V_h(g) \times Q_h$ s.t.

$$a(u_h, v_h) + b(v_h, p_h) = (f, v_h) \quad \forall v_h \in V_h,$$
$$b(u_h, q_h) - \delta d(p_h, q_h) = 0 \quad \forall q_h \in Q_h.$$

$\delta > 0$ : stability parameter, $d(p_h, q_h) = \sum_{K \in \mathcal{T}} h_K^2 \int_K \nabla p_h \cdot \nabla q_h \, dx$.

$|p_h|_h^2 = d(p_h, p_h)$ : mesh dependent norm on $Q_h$.

- uniform weak inf-sup condition : Franca-Stenberg [1991]

  $$\exists \beta_0,\ \beta_1 > 0\ \forall h > 0\ \sup_{v_h \in V_h} \frac{b(v_h, q_h)}{||v_h||_1} \geq \beta_0 ||q_h||_0 - \beta_1 |q_h|_h\ \forall\, q_h \in Q_0.$$

## Additive Schwarz preconditioner for 3D computation : 3/4

matrix formulation of the stabilized FEM for the Stokes eqs.
$V=\mathbb{R}^{N_v}, \quad Q\subset\mathbb{R}^{N_p}, (\vec{q},\vec{1})=0$ for $\vec{q}\in Q$.
find $(\vec{u},\vec{p})\in V\times Q$ s.t.

$$\left(\begin{bmatrix} A & B^T \\ B & -\delta\, D\end{bmatrix}\begin{bmatrix}\vec{u}\\ \vec{p}\end{bmatrix}-\begin{bmatrix}\vec{f}\\ \vec{0}\end{bmatrix},\begin{bmatrix}\vec{v}\\ \vec{q}\end{bmatrix}\right)=0 \quad \forall(\vec{v},\vec{q})\in V\times Q$$

subspace : $U\times R$, $U\subset V$, $R\subset Q$

$\Rightarrow \quad \begin{bmatrix} A & B^T \\ B & -\delta\, D\end{bmatrix}$ is invertible on $U\times R$.

proof

$$\left(\begin{bmatrix} A & B^T \\ B & -\delta\, D\end{bmatrix}\begin{bmatrix}\vec{u}\\ \vec{p}\end{bmatrix},\begin{bmatrix}\vec{v}\\ \vec{q}\end{bmatrix}\right)=0 \quad \forall(\,\vec{v},\vec{q}\,)\in U\times R \quad \Rightarrow \vec{u}=\vec{0},\vec{p}=\vec{0}.$$

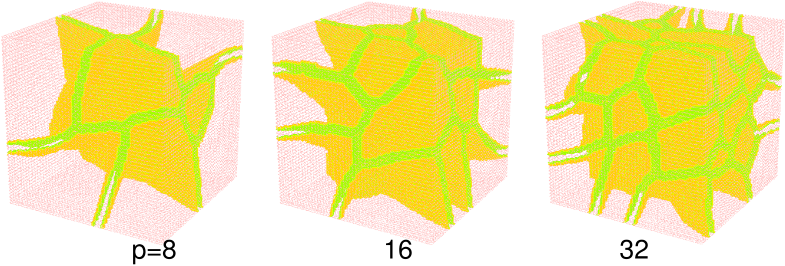
$$(\,\vec{u},\ \vec{p}\,)\in U\times R \;\Rightarrow\; (\,\vec{u},\ -\vec{p}\,)\in U\times R.$$

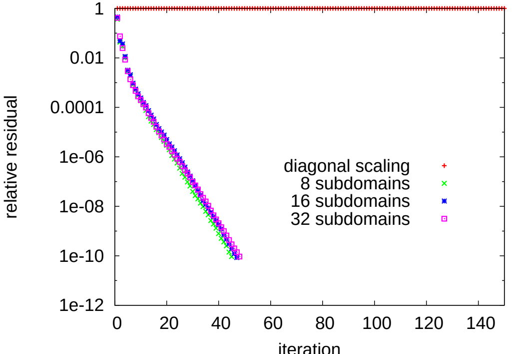
$$\left(\begin{bmatrix} A & B^T \\ B & -\delta\, D\end{bmatrix}\begin{bmatrix}\vec{u}\\ \vec{p}\end{bmatrix},\begin{bmatrix}\vec{u}\\ -\vec{p}\end{bmatrix}\right)=(\,A\vec{u},\vec{u}\,)+\delta(\,D\vec{p},\vec{p}\,)>0.$$

cf. S, Profeedings of ALGORITMY 2009

# Additive Schwarz preconditioner for 3D computation : 4/4

50×50×50 FEM nodes, overlap = 1




$R_0 A R_0^T$ : $7p \times 7p$

dimker$(R_0 A R_0^T) = 1$

is detected by Dissection.

## conclusion

- indefinite matrix is factorized by postponing strategy for suspicious null pivots
- combination of 1x1 and 2x2 pivoting can factorize finite element matrices without adding perturbation
- new kernel detection algorithm resolves rank deficient problem from FEM matrices
- 1M DOF is factorized with 57GB memory on a shared memory computer within 2 minutes
- direct solver is efficiently used as sub-domain solver
- stabilized term for the Stokes equations ensures solvability of the coarse problem