Multilevel spectral coarse space methods in `FreeFem++` on parallel architectures

Pierre Jolivet

Laboratoire Jacques-Louis Lions
Laboratoire Jean Kuntzmann

DD 21, Rennes.

June 29, 2012

In collaboration with F. Nataf, N. Spillane, V. Dolean,
F. Hecht, C. Prud'homme.

Outline

Context

Solve large systems arising from the finite element method.

Context

Solve large systems arising from the finite element method.

What are the different alternatives ?

- parallel direct solvers (`MUMPS`, `SuperLU`, `PaStiX` ..),
- parallel iterative solvers (`HIPS`, `Hypre` ..),
- hybrid solvers (`BoomerAMG`, `ML`, `PARDISO` ..),
- ...

Context

Solve large systems arising from the finite element method.

What are the different alternatives ?

- parallel direct solvers (`MUMPS`, `SuperLU`, `PaStiX` ..),
- parallel iterative solvers (`HIPS`, `Hypre` ..),
- hybrid solvers (`BoomerAMG`, `ML`, `PARDISO` ..),
- ...

$\Longrightarrow$ high-performance algorithms on massively parallel distributed memory multiprocessor architectures.

Constraints:
- robust in size and heterogeneities,
- easy to maintain.

Boundary value problems

Given a FE space V and $f \in V^\star$, find $u \in V$ such that:

$$a(u, v) = \langle f, v \rangle \quad \forall v \in V \implies Au = F$$

where

$$\left.\begin{array}{l} a_\Omega(u, v) = \int_\Omega \kappa \nabla u \cdot \nabla v \\ a_\Omega(u, v) = \int_\Omega \mathcal{C} : \varepsilon(u) : \varepsilon(v) \end{array}\right. \implies A \text{ is assumed to be SPD}$$

Notation

Ω is split into N overlapping subdomains $\{\Omega_i\}_{i=1}^N \rightsquigarrow \{V_i\}_{i=1}^N$.

Let:

- R_i be the restriction from $V^\star$ to $V_i^\star$ (thus R_i^T is the extension by 0 from V_i to V),
- $A_{ij} := R_i A R_j^T$,
- $\mathcal{O}_i$ the set of neighboring subdomains (and $\overline{\mathcal{O}_i} = \mathcal{O}_i \cup \{i\}$),
- $\{\chi_i\}_{i=1}^N$ be a partition of unity:

$$\mathrm{supp}(\chi_i) \subset \Omega_i \quad \text{and} \quad \sum_{i=1}^N \chi_i = 1,$$

- $\{D_i\}_{i=1}^N$ be the discretization of this partition:

$$\sum_{i=1}^N R_i^T D_i R_i = I.$$

One-level Schwarz method

Two classical preconditioners are:

$$M_{\text{RAS}}^{-1} = \sum_{i=1}^{N} R_i^T D_i A_{ii}^{-1} R_i$$

$$M_{\text{ASM}}^{-1} = \sum_{i=1}^{N} R_i^T A_{ii}^{-1} R_i$$

Some properties

- obviously parallel,
- not scalable.

Outline

Principle

Let Z be a rectangular matrix so that the "bad eigenvectors" of $M^{-1}A$ belong to the space spanned by its columns. Define:

$$E := Z^T A Z \qquad P := I - AQ \qquad Q := Z E^{-1} Z^T$$

$\Longrightarrow$ the number of columns of Z is $O(N)$.

Principle

Let Z be a rectangular matrix so that the "bad eigenvectors" of $M^{-1}A$ belong to the space spanned by its columns. Define:

$$E := Z^T A Z \qquad P := I - AQ \qquad Q := ZE^{-1}Z^T$$

$\implies$ the number of columns of Z is $O(N)$.

Two-level preconditioners

$$\begin{aligned}
\mathcal{P}_{\text{BNN}} &:= (I - AQ)^T M^{-1}(I - AQ) + Q \\
\mathcal{P}_{\text{A-DEF1}} &:= M^{-1}(I - AQ) + Q \\
\mathcal{P}_{\text{A-DEF2}} &:= (I - AQ)^T M^{-1} + Q
\end{aligned}$$

Differences and similarities studied in (Tang et al. 2009).

Principle

Let Z be a rectangular matrix so that the "bad eigenvectors" of $M^{-1}A$ belong to the space spanned by its columns. Define:

$$E := Z^T AZ \qquad P := I - AQ \qquad Q := ZE^{-1}Z^T$$

$\implies$ the number of columns of Z is $O(N)$.

Two-level preconditioners

$$\mathcal{P}_{\text{BNN}} := (I - AQ)^T M^{-1}(I - AQ) + Q$$
$$\mathcal{P}_{\text{A-DEF1}} := M^{-1}(I - AQ) + Q$$
$$\mathcal{P}_{\text{A-DEF2}} := (I - AQ)^T M^{-1} + Q$$

Differences and similarities studied in (Tang et al. 2009).

The real problem is: how to build Z ?

A priori construction

Based on an analysis of the underlying PDE, (Nataf et al. 2011; Spillane et al. 2011) — M13P1.

Generalized eigenvalue problem on the overlap

On each Ω_i, find the eigenpairs $(\Lambda_{i_k}, \lambda_{i_k})_k$ such that:

$$a_{\Omega_i}(\Lambda_{i_k}, v_i) = \lambda_{i_k} a_{\Omega_i^\circ}(\chi_i \Lambda_{i_k}, \chi_i v_i) \qquad \forall v_i \in V_i$$

which after discretization yields:

$$A_i \Lambda_{i_k} = \lambda_{i_k} D_i A_i^\circ D_i \Lambda_{i_k}$$

A threshold criterion selects only ν_i eigenvectors associated to low frequency eigenvalues:

$$Z = \begin{bmatrix} W_1 & 0 & \cdots & 0 \\ 0 & W_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & W_N \end{bmatrix}$$

where

$$W_i = \begin{bmatrix} D_i \Lambda_{i_1} & D_i \Lambda_{i_2} & \cdots & D_i \Lambda_{i_{\nu_i}} \end{bmatrix}$$

Outline

Domain Specific Language

`FreeFem++`, with the help of:

- Metis (Karypis and Kumar 1998)
- SCOTCH (Chevalier and Pellegrini 2008)
- UMFPACK (Davis 2004)
- ARPACK (Lehoucq et al. 1998)
- SLEPc (Hernandez et al. 2005)
- BLOPEX (Knyazev et al. 2007)
- MPI (Snir et al. 1995)
- ...

Domain Specific Language

FreeFem++, with the help of:

- Metis (Karypis and Kumar 1998)
- SCOTCH (Chevalier and Pellegrini 2008)
- UMFPACK (Davis 2004)
- ARPACK (Lehoucq et al. 1998)
- SLEPc (Hernandez et al. 2005)
- BLOPEX (Knyazev et al. 2007)
- MPI (Snir et al. 1995)
- ...

Why use a DS(E)L instead of C/C++/Fortran/. . . ?

- performances close to low-level language implementation,
- hard to beat something as simple as:

$$\textbf{varf } \mathrm{a(u, v)} = \textbf{int3d}(\mathrm{mesh})([\textbf{dx}(\mathrm{u}), \textbf{dy}(\mathrm{u}), \textbf{dz}(\mathrm{u})]' * [\textbf{dx}(\mathrm{v}), \textbf{dy}(\mathrm{v}), \textbf{dz}(\mathrm{v})]) + \textbf{int3d}(\mathrm{mesh})(\mathrm{f} * \mathrm{v}) + \textbf{on}(\mathrm{boundary\_mesh})(\mathrm{u} = 0)$$

Reformulation

Parallel FE computations implies parallel matrix assembly:

$$A_{jk} := R_j A R_k^T \quad \forall k \in \mathcal{O}_j$$

Cannot be done with `FreeFem++`.

Reformulation

Parallel FE computations implies parallel matrix assembly:

$$A_{jk} := R_j A R_k^T \quad \forall k \in \mathcal{O}_j$$

Cannot be done with `FreeFem++`.

A simple transformation on the overlap

$$A_{jk} D_k R_k u = R_j R_k^T A_{kk} D_k R_k u$$

Reformulation

Parallel FE computations implies parallel matrix assembly:

$$A_{jk} := R_j A R_k^T \quad \forall k \in \mathcal{O}_j$$

Cannot be done with `FreeFem++`.

A simple transformation on the overlap

$$A_{jk} D_k R_k u = R_j R_k^T A_{kk} D_k R_k u$$

Reformulation

Parallel FE computations implies parallel matrix assembly:

$$A_{jk} := R_j A R_k^T \quad \forall k \in \mathcal{O}_j$$

Cannot be done with `FreeFem++`.

A simple transformation on the overlap

$$A_{jk} D_k R_k u = R_j R_k^T A_{kk} D_k R_k u$$

What does it mean ?

$$Au = \text{black box plugged with a Krylov method}$$

Reformulation

Parallel FE computations implies parallel matrix assembly:

$$A_{jk} := R_j A R_k^T \quad \forall k \in \mathcal{O}_j$$

Cannot be done with `FreeFem++`.

A simple transformation on the overlap

$$A_{jk} D_k R_k u = R_j R_k^T A_{kk} D_k R_k u$$

What does it mean ?

$$Au = \sum_{j=1}^{N} \quad A R_j^T D_j R_j u$$

Reformulation

Parallel FE computations implies parallel matrix assembly:

$$A_{jk} := R_j A R_k^T \quad \forall k \in \mathcal{O}_j$$

Cannot be done with `FreeFem++`.

A simple transformation on the overlap

$$A_{jk} D_k R_k u = R_j R_k^T A_{kk} D_k R_k u$$

What does it mean ?

$$R_i A u = \sum_{j=1}^{N} R_i A R_j^T D_j R_j u$$

Reformulation

Parallel FE computations implies parallel matrix assembly:

$$A_{jk} := R_j A R_k^T \quad \forall k \in \mathcal{O}_j$$

Cannot be done with `FreeFem++`.

A simple transformation on the overlap

$$A_{jk} D_k R_k u = R_j R_k^T A_{kk} D_k R_k u$$

What does it mean ?

$$R_i A u = \sum_{j=1}^{N} R_i A R_j^T D_j R_j u = \sum_{j \in \overline{\mathcal{O}_i}} A_{ij} D_j R_j u$$

$$= \sum_{j \in \overline{\mathcal{O}_i}} R_i R_j^T A_{jj} D_j R_j u$$

Reformulation

Parallel FE computations implies parallel matrix assembly:

$$A_{jk} := R_j A R_k^T \quad \forall k \in \mathcal{O}_j$$

Cannot be done with `FreeFem++`.

A simple transformation on the overlap

$$A_{jk} D_k R_k u = R_j R_k^T A_{kk} D_k R_k u$$

What does it mean ?

$$R_i A u = \sum_{j=1}^{N} R_i A R_j^T D_j R_j u = \sum_{j \in \overline{\mathcal{O}_i}} A_{ij} D_j R_j u$$

$$= \sum_{j \in \overline{\mathcal{O}_i}} R_i R_j^T A_{jj} D_j R_j u$$

local unknowns on V_j

Coarse space construction

The same goes for the construction of $E := Z^T A Z$.

A double block matrix multiplication

$$E_{ij} = \Lambda^T_{\mathcal{R}(i)_{\varphi(i)}} D_{\mathcal{R}(i)} A_{\mathcal{R}(i)\mathcal{R}(j)} D_{\mathcal{R}(j)} \Lambda_{\mathcal{R}(j)_{\varphi(j)}}$$

where

$$\mathcal{R} : j \mapsto \max \left\{ i : \sum_{k=1}^{i} \nu_k < j \right\} \qquad \varphi : j \mapsto j - \sum_{k=1}^{\mathcal{R}(j)} \nu_k$$

A similar reformulation leads to:

$$E_{ij} = \Lambda^T_{\mathcal{R}(i)_{\varphi(i)}} D_{\mathcal{R}(i)} R_i R_j^T A_{\mathcal{R}(j)\mathcal{R}(j)} D_{\mathcal{R}(j)} \Lambda_{\mathcal{R}(j)_{\varphi(j)}} \neq 0 \iff j \in \mathcal{O}_i$$

Heterogeneous architectures

	N° cores	Memory	Peak performance	Compilers
titane@CEA	12192*	37 To	140 TFLOP/s	`Intel`
babel@IDRIS	40960	20 To	139 TFLOP/s	`IBM + GNU`
curie@CEA	80640	322 To	2 PFLOP/s	`Intel`

* + 46080 CUDA cores

ANR Preconditioning scientific applications on pETascALe Heterogeneous machines

PRACE High Performance Computing-PDE

`http://www-hpc.cea.fr`, Bruyères-le-Châtel, France.
`http://www.idris.fr`, Orsay, France.

Darcy pressure equation I

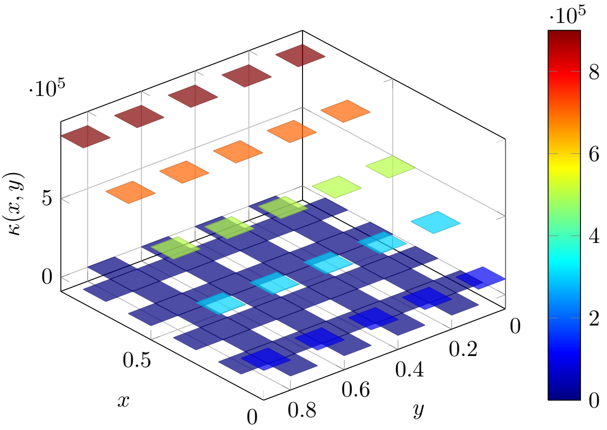

Fig: Two dimensional diffusivity κ

Darcy pressure equation II

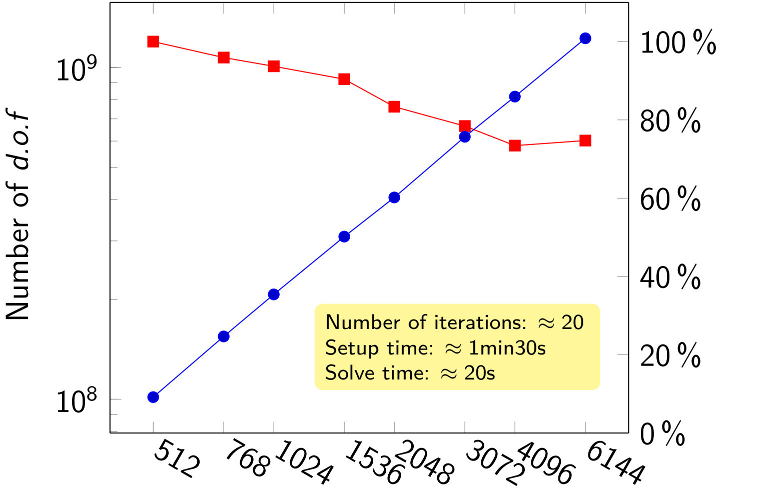

Fig: 2D efficiency (weak scaling) — $\mathbb{P}_3$ FE, stopping criterion $\varepsilon = 10^{-8}$

System of linear elasticity I (60M unkowns)

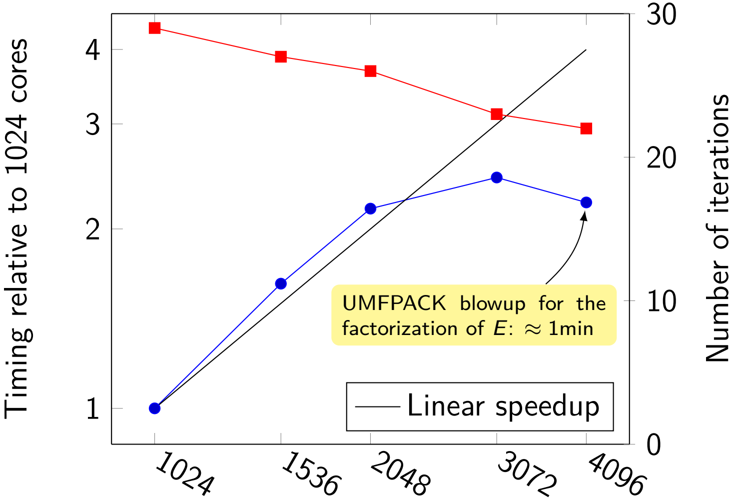

Fig: 3D strong scaling — $\mathbb{P}_2$ FE, stopping criterion $\varepsilon = 10^{-10}$

System of linear elasticity II (375M unkowns)

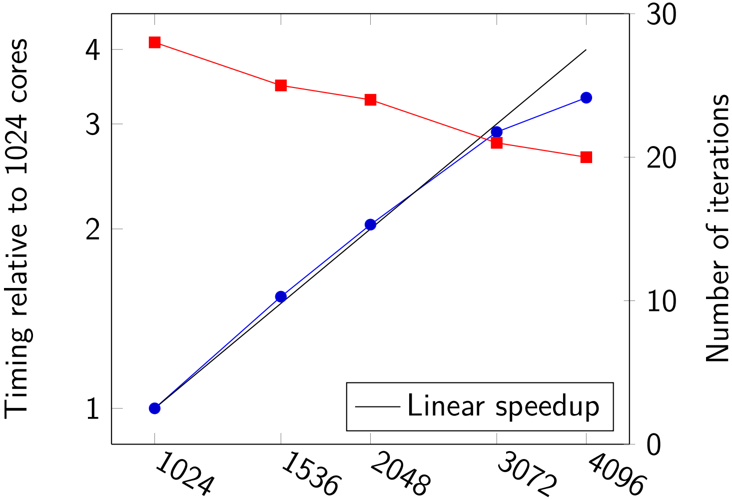

Fig: 2D strong scaling — $\mathbb{P}_3$ FE, stopping criterion $\varepsilon = 10^{-8}$

The limits reached

6144-way decomposition:

- in $\mathbb{R}^2$, 1.2G unkowns,
- in $\mathbb{R}^3$, 300M unkowns.

All systems are solved with:

- coarse spaces of size $[\![100; 120\,000]\!]$,
- less than 30 iterations.

Outline

Final words

What to remember:

FreeFem++ + Two-level DDM = easy framework to solve large systems.

New problems being tackled:

- nonlinear elasticity (Newton-Krylov-Schwarz methods),
- reuse of Krylov and deflation subspaces.

Final words

What to remember:

$$\begin{array}{c}\texttt{FreeFem++}\\ + \\ \text{Two-level DDM}\end{array} \quad = \quad \text{easy framework to solve large systems.}$$

New problems being tackled:

- nonlinear elasticity (Newton-Krylov-Schwarz methods),
- reuse of Krylov and deflation subspaces.

Thanks for your attention.

Chevalier, C. and F. Pellegrini (2008). "PT-Scotch: a tool for efficient parallel graph ordering." In: *Parallel Computing* 34.6, pp. 318–331.

Davis, T.A. (2004). "Algorithm 832: UMFPACK — an unsymmetric-pattern multifrontal method." In: *ACM Transactions on Mathematical Software* 30.2, pp. 196–199.

Hernandez, V., J.E Roman, and V. Vidal (2005). "SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems." In: *ACM Transactions on Mathematical Software* 31.3, pp. 351–362.

Karypis, G. and V. Kumar (1998). "A fast and high quality multilevel scheme for partitioning irregular graphs." In: *SIAM Journal on Scientific Computing* 20.1, pp. 359–392.

- Knyazev, A.V., M.E. Argentati, I. Lashuk, and E. Ovtchinnikov (2007). “Block Locally Optimal Preconditioned Eigenvalue Xolvers (BLOPEX) in Hypre and PETSc.” In: *SIAM Journal on Scientific Computing* 29.5, pp. 2224–2239.
- Lehoucq, R.B., D.C. Sorensen, and C. Yang (1998). *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods.* Vol. 6. Society for Industrial and Applied Mathematics.
- Nataf, F., H. Xiang, V. Dolean, and N. Spillane (2011). “A coarse space construction based on local Dirichlet to Neumann maps.” In: *SIAM Journal on Scientific Computing* 33.4, pp. 1623–1642.
- Snir, M., S.W. Otto, D.W. Walker, J. Dongarra, and S. Huss-Lederman (1995). *MPI: The complete reference.* The MIT Press.

Spillane, N., V. Dolean, P. Hauret, F. Nataf, C. Pechstein, and R. Scheichl (2011). "A robust two-level domain decomposition preconditioner for systems of PDEs." In: *Comptes Rendus Mathematique* 349.23, pp. 1255–1259.

Tang, J.M., R. Nabben, C. Vuik, and Y.A. Erlangga (2009). "Comparison of two-level preconditioners derived from deflation, domain decomposition and multigrid methods." In: *Journal of Scientific Computing* 39.3, pp. 340–370.