HYBRID DIRECT AND ITERATIVE SOLVER
FOR H ADAPTIVE MESHES WITH POINT SINGULARITIES

Maciej Paszyński
Department of Computer Science,
AGH University of Science and Technology, Kraków, Poland
e-mail: paszynsk@agh.edu.pl

Collaborators:

Victor Calo (KAUST, Thuwal, Saudi Arabia)
Lisandro Dalcin (KAUST, Saudi Arabia)
David Pardo (IKERBASQUE, Spain)
Keshav Pingali (ICES, Austin TX, USA)

PhD Students:

Piotr Gurgul, Krzysztof Kuźnik, Maciej Woźniak, Damain Goik, Konrad Jopek
(AGH, Krakow, Poland)
OUTLINE

1. Introduction
   a) Examplary 2D two finite element mesh (2 slides)
   b) Frontal solver (2 slides)
   c) Linear cost of point singularity (4 slides)
   d) Multi-frontal solver (3 slides)
   e) Dealing with other singularities (2 slides)
   f) ILUPCG (2 slides)

2. Hybrid solver algorithm (2 slides)
   a) Elimination of local singularities + ILUPCG
   b) Static condensation + ILUPCG

3. Numerical examples
   a) 2D heat transfer with different materials (7 slides)
   b) 3D heat transfer with point sources (11 slides)
   c) 3D Fichera problem with point+edge singularities (6 slides)

4. Conclusions
INTRODUCTION
We seek the solution $u$ of some weak form of PDE as a linear combination of shape functions $e_{hp}^i$ spread over finite element mesh.
The coefficients $u^i_{hp}$ (called “degrees of freedom” d.o.f.) are obtained by solving system of linear equations – finite element discretization of a weak (variational) form of PDE

$$\sum_{i=1}^{15} u^i_{hp} b(e^i_{hp}, e^j_{hp}) = l(e^j_{hp}) \quad j = 1,\ldots,15$$

where $b(e^i_{hp}, e^j_{hp})$ and $l(e^j_{hp})$ are some integrals of shape functions $e^i_{hp}, e^j_{hp}$
FRONTAL SOLVER
SOLUTION BASED ON LINEAR ORDER OF ELEMENTS

Generates frontal matrix for the first element, eliminates fully assembled degrees of freedom
Generates frontal matrix for the second element, merges with the current frontal matrix, eliminates fully assembled degrees of freedom.
DIRECT SOLVER FOR RADICAL MESH

Interface size: green nodes

Constant matrix size $\rightarrow$ linear cost of the direct solver
Interface size: green nodes

Constant matrix size $\rightarrow$ linear cost of the direct solver
DIRECT SOLVER FOR RADICAL MESH

Interface size: green nodes

Constant matrix size $\rightarrow$ linear cost of the direct solver
DIRECT SOLVER FOR RADICAL MESH

Interface size: green nodes
Constant matrix size $\rightarrow$ linear cost of the direct solver
MULTI-FRONTAL SOLVER
SOLUTION BASED ON THE ELIMINATION TREE

Partial forward elimination
MULTI-FRONTAL SOLVER
SOLUTION BASED ON THE ELIMINATION TREE

Partial forward elimination
MULTI-FRONTAL SOLVER
SOLUTION BASED ON THE ELIMINATION TREE

Full forward elimination of the interface problem matrix
COMPARISON OF NUMERICAL AND THEORETICAL
SCALABILITY EXPONENT FACTORS
FOR REFINEMENTS TOWARDS A SINGLE ENTITY

<table>
<thead>
<tr>
<th></th>
<th>Uniform</th>
<th>Point</th>
<th>Edge Isotropic</th>
<th>Edge Anisotropic</th>
<th>Face Isotropic</th>
<th>Face Anisotropic</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>1.86</td>
<td>1.09</td>
<td>1.10</td>
<td>1.07</td>
<td>1.47</td>
<td>1.01</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>1.86</td>
<td>1.17</td>
<td>1.18</td>
<td>1.07</td>
<td>1.47</td>
<td>1.12</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>1.83</td>
<td>1.08</td>
<td>1.21</td>
<td>1.09</td>
<td>1.54</td>
<td>1.08</td>
</tr>
<tr>
<td>Theoretical</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1.5</td>
<td>1</td>
</tr>
</tbody>
</table>
NUMERICAL SCALABILITY EXPONENT FACTORS FOR REFINEMENTS TOWARDS MULTIPLE SINGULARITIES

<table>
<thead>
<tr>
<th></th>
<th>Point + Edge</th>
<th>Point + Face</th>
<th>Edge + Face</th>
<th>Point + Edge + Face</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>1.33</td>
<td>1.46</td>
<td>1.24</td>
<td>1.57</td>
</tr>
<tr>
<td>$p = 4$</td>
<td>1.45</td>
<td>1.60</td>
<td>1.35</td>
<td>1.75</td>
</tr>
<tr>
<td>$p = 6$</td>
<td>1.39</td>
<td>1.56</td>
<td>1.23</td>
<td>1.65</td>
</tr>
</tbody>
</table>
ILUPCG = ILU PRECONDITIONER + CONJUGATED GRADIENTS

Conjugated gradients algorithm (Y. Saad „Iterative methods for sparse linear systems“)

\[
\begin{align*}
\text{Compute } r_0 & := b - Ax_0, p_0 := r_0. \\
\text{For } j = 0, 1, \ldots, \text{ until convergence Do:} & \\
\quad & \alpha_j := (r_j, r_j) / (Ap_j, p_j) \\
\quad & x_{j+1} := x_j + \alpha_j p_j \\
\quad & r_{j+1} := r_j - \alpha_j Ap_j \\
\quad & \beta_j := (r_{j+1}, r_{j+1}) / (r_j, r_j) \\
\quad & p_{j+1} := r_{j+1} + \beta_j p_j \\
\text{EndDo}
\end{align*}
\]
ILUPCG = ILU PRECONDITIONER + CONJUGATED GRADIENTS

Gaussian Elimination (Y. Saad „Iterative methods for sparse linear systems“)

for i = 2 ... n
    for k = 1 ... i-1
        a(i,k) = a(i,k)/a(k,k)
    for j = k+1 ... n
        a(i,j) = a(i,j) - a(i,k)*a(k,j)

ILU(0) preconditioner (Gaussian Elimination skipping zeros)

for i = 2 ... n
    for k = 1 ... i-1 and for (i,k) ∈ NZ(A)
        a(i,k) = a(i,k)/a(k,k)
    for j = k+1 ... n and for (i,j) ∈ NZ(A)
        a(i,j) = a(i,j) - a(i,k)*a(k,j)

Applying the preconditioner
Ax=b
A=LU (by incomplete LU) \rightarrow L^{-1}AU^{-1}Ux = L^{-1}b

Solve \quad A^* \quad x^* = b^* \quad by \ Conjugated \ Gradients


HYBRID SOLVER
HYBRID SOLVER FOR 2D GRIDS WITH POINT SINGULARITIES

Several levels of h refinements

O(logN)
GPU preprocessor

O(N)
iterative solver from SLATEC library

ILUPCG

NVIDIA CUDA GPU solver for 2D grids (by Piotr Gurgul and Krzysztof Kuźnik)

ILUPCG from SLATEC library
http://people.sc.fsu.edu/~jburkardt/f_src/slatec/slatec.html

Comparison to MUMPS direct solver
http://mumps.enseeiht.fr/
ILUPCG AFTER STATIC CONDENSATION

Several levels of h refinements

Static condensation

O(N) iterative solver from SLATEC library

ILUPCG

ILUPCG from SLATEC library
http://people.sc.fsu.edu/~jburkardt/f_src/slatec/slatec.html

Static condensation implemented „by hand”
8x8 initial elements mesh
Heat transfer
3 materials $K_{ii}=1, 1000, 1000000$
4x4 point singularities

$$\nabla \circ (K \nabla u) = 0 \quad \text{on } \Omega$$

$$\nabla u \circ n = 1 \quad \text{on } \Gamma_N$$

$$u = 0 \quad \text{on } \Gamma_D$$
HYBRID SOLVER FOR 2D GRIDS WITH POINT SINGULARITIES

8x8 initial elements mesh
3 materials $K_{ii}=1, 1000, 1000000$
Heat transfer
4x4 point singularities

$p=2$

Red plot = Schur complement up to initial mesh
Blue plot = entire mesh submitted to ILUPCG after static condensation

number of iterations of ILU(P)CG

number of refinement levels
HYBRID SOLVER FOR 2D GRIDS WITH POINT SINGULARITIES

number of iterations of ILUBPCG

number of refinement levels

p=3

Red plot = Schur complement up to initial mesh

Blue plot = entire mesh submitted to ILUPCG after static condensation

8x8 initial elements mesh
3 materials $K_{ii}=1, 1000, 1000000$
Heat transfer
4x4 point singularities
16 executions of NVIDIA CUDA GPU solver for 4x4 singularities for p=2 followed by ILUPCG call for top problem
Comparison with MUMPS multi-frontal direct solver
GeForce GTX 260 graphic card with 24 x 8 =192 cores, 896MB of memory
NUMERICAL RESULTS FOR 2D GRIDS

16 executions of NVIDIA CUDA GPU solver for 4x4 singularities for p=3 followed by ILUPCG call for top problem
Comparison with MUMPS multi-frontal direct solver
GeForce GTX 260 graphic card with 24 x 8 =192 cores, 896MB of memory
NUMERICAL RESULTS FOR 2D GRIDS

16 executions of NVIDIA CUDA GPU solver for 8x8 singularities for p=2 followed by ILUPCG call for top problem
Comparison with MUMPS multi-frontal direct solver
GeForce GTX 260 graphic card with 24 x 8 =192 cores, 896MB of memory
NUMERICAL RESULTS FOR 2D GRIDS

16 executions of NVIDIA CUDA GPU solver for 8x8 singularities for p=3 followed by ILUPCG call for top problem
Comparison with MUMPS multi-frontal direct solver
GeForce GTX 260 graphic card with 24 x 8 =192 cores, 896MB of memory
HYBRID SOLVER FOR 3D GRIDS WITH POINT SINGULARITIES

Several levels of $h$ refinements

$O(\log N)$
GALOIS preprocessor

$O(N)$
iterative solver from SLATEC library

$\rightarrow$ ILUPCG

Graph grammar based solver in GALOIS for 3D grids
Damian Goik, Konrad Jopek, Maciej Paszynski, Andrew Lenharth, Donald Nguyen and Keshav Pingali

*Graph grammar based multi-thread multi-frontal direct solver with Galois scheduler* submitted to *International Conference on Computational Science, ICCS 2014*, June 10-12, Cairns, Australia

ILUPCG from SLATEC library
[http://people.sc.fsu.edu/~jburkardt/f_src/slatec/slatec.html](http://people.sc.fsu.edu/~jburkardt/f_src/slatec/slatec.html)
ILUPCG AFTER STATIC CONDENSATION

Several levels of h refinements → Static condensation → O(N) iterative solver from SLATEC library

ILUPCG from SLATEC library
http://people.sc.fsu.edu/~jburkardt/f_src/slatec/slatec.html

Static condensation implemented „by hand”
HYBRID SOLVER FOR 3D GRIDS WITH POINT SINGULARITIES

\[ \Delta u = f \quad \text{on } \Omega \]
\[ \nabla u \cdot n = 1 \quad \text{on } \Gamma_N \]
\[ u = 0 \quad \text{on } \Gamma_D \]

8x8x8 initial elements mesh
Heat transfer
Point sources
4x4x4 point singularities
HYBRID SOLVER FOR 3D GRIDS WITH POINT SINGULARITIES

number of iterations of ILUBPCG

Red plot = Schur complement up to initial mesh
Blue plot = entire mesh submitted to ILUPCG after static condensation

\( p=2 \)  

number of refinement levels

8x8x8 initial elements mesh
Heat transfer
Point sources
4x4x4 point singularities
HYBRID SOLVER FOR 3D GRIDS WITH POINT SINGULARITIES

number of iterations of ILUBPCG

$p=3$

Red plot = Schur complement up to initial mesh

Blue plot = entire mesh submitted to ILUPCG after static condensation

8x8x8 initial elements mesh
Heat transfer
Point sources
4x4x4 point singularities
HYBRID SOLVER FOR 3D GRIDS WITH POINT SINGULARITIES

Number of iterations

Submission of the entire mesh into ILUPCG solver after static condensation

8x8x8 initial elements mesh
Heat transfer
Point sources
4x4x4 point singularities
HYBRID SOLVER FOR 3D GRIDS WITH POINT SINGULARITIES

Execution time of ILUPCG

Submission of the entire mesh into ILUPCG solver after static condensation

8x8x8 initial elements mesh
Heat transfer
Point sources
4x4x4 point singularities
Damian Goik, Konrad Jopek, Maciej Paszynski, Andrew Lenharth, Donald Nguyen and Keshav Pingali  "Graph grammar based multi-thread multi-frontal direct solver with Galois scheduler" submitted to International Conference on Computational Science, ICCS 2014, June 10-12, Cairns, Australia
ELIMINATION OF 3D POINT SINGULARITIES WITH GALOIS
FROM GROUP OF KESHA V PINGALI
COMPARISON WITH MUMPS p=3

Damian Goik, Konrad Jopek, Maciej Paszynski, Andrew Lenharth, Donald Nguyen
and Keshav Pingali "Graph grammar based multi-thread multi-frontal direct solver with Galois scheduler" submitted to International Conference on Computational Science, ICCS 2014, June 10-12, Cairns, Australia
HYBRID SOLVER

Comparison of strategies for p=2:

**Blue line** = MUMPS

**Red line** = Static condensation
  (elimination of interiors of all elements)
  followed by submission to ILUPCG

**Green line** = Elimination of point singularities with GALOIS
  followed by ILUPCG for resulting Schur complements

8x8x8 initial elements mesh
Heat transfer
Point sources
4x4x4 point singularities
Comparison of strategies for $p=3$:

**Blue line** = MUMPS

**Red line** = Static condensation
   (elimination of interiors of all elements)
   followed by submission to ILUPCG

**Green line** = Elimination of point singularities with GALOIS
   followed by ILUPCG for resulting Schur complements

8x8x8 initial elements mesh
Heat transfer
Point sources
4x4x4 point singularities
3D FICHERA PROBLEM

\[
\begin{cases}
\Delta u = 0 & \text{in } \Omega \\
u = 0 & \text{on } \Gamma_D \\
\frac{\partial u}{\partial n} = g & \text{on } \Gamma_N
\end{cases}
\]

\[g(r, \theta) = r^3 \sin \frac{2}{3} \left( \theta + \frac{\Pi}{2} \right)\]

Laplace equation
HYBRID SOLVER FOR 3D FICHERA PROBLEM

Several levels of h refinements

Hypersolver preprocessor

O(N) iterative solver from SLATEC library

ILUPCG

Hypersolver by Maciej Paszyński

ILUPCG from SLATEC library
[http://people.sc.fsu.edu/~jburkardt/f_src/slatec/slatec.html](http://people.sc.fsu.edu/~jburkardt/f_src/slatec/slatec.html)
ILUPCG AFTER STATIC CONDENSATION

Several levels of h refinements

Static condensation

O(N) iterative solver from SLATEC library

ILUPCG from SLATEC library
http://people.sc.fsu.edu/~jburkardt/f_src/slatec/slatec.html

Static condensation implemented „by hand”
Number of iterations of ILUBPCG

Number of refinement levels

\( p = 1 \)

Fichera problem
Automatic h refinements

Red plot = Schur complement up to initial mesh

Blue plot = entire mesh submitted to ILUBPCG after static condensation
HYBRID SOLVER

number of iterations of ILUBPCG

number of refinement levels

\[ p=2 \]

Fichera problem
Automatic h refinements

Red plot = Schur complement up to initial mesh

Blue plot = entire mesh submitted to ILUBPCG after static condensation
HYBRID SOLVER

number of iterations of ILUBPCG

number of refinement levels

\( p=3 \)

Red plot = Schur complement up to initial mesh

Blue plot = entire mesh submitted to ILUBPCG after static condensation

Fichera problem
Automatic h refinements
CONCLUSIONS

1. Local point singularities can be eliminated with linear cost
2. Hybrid solver algorithm:
   a) Elimination of local singularities + ILUPCG
   b) Static condensation + ILUPCG
3. Significant reduction on number of iterations of ILUPCG
   after elimination of local singularities
4. Verification by numerical examples for 2D and 3D problems
   a) Two dimensional heat transfer with non uniform materials
   b) Three dimensional heat transfer with point sources
   c) Fichera problem
5. Future work
   a) Local elimination of other singularities with Galois
   b) Parallel implementation of ILUPCG and CG
   c) Implementation in PETSC