Embedded boundary methods and domain decomposition

Santiago Badia [1] and Francesc Verdugo [1]

SIAM CSE17, Atlanta, March 3rd, 2017

[1] Universitat Politècnica de Catalunya & LSSC-CIMNE (Spain)

Table of Contents

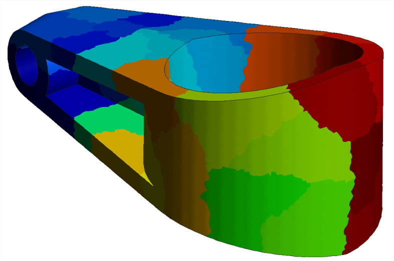
Motivation

Simulation bottlenecks

Mesh generation

Mesh partition

Linear solver



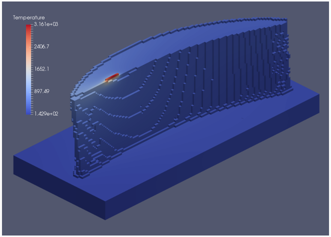
Background

- Scalable linear solvers are making possible to solve larger problems efficiently on HPC platforms
- Simulation bottleneck for industrial applications is shifting towards body-fitted mesh generation and graph partitioning of unstructured grids

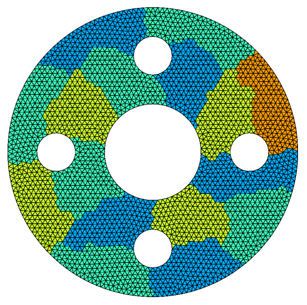
Motivation

Need for additive manufacturing simulation (EU projects: CAxMan, eMusic), body-fitted meshes are not suitable

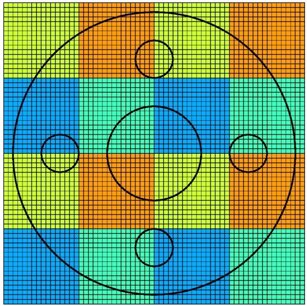
- Our code can read AM process data (CLI) and perform process simulations
- Geometry depends in time $\Omega(t)$...
- Conforming meshes $\mathcal{T}_h(t)$ at all times not possible



Motivation



Body-fitted unstructured grid



Unfitted Cartesian grid

Our goals

1) To use (adaptive) Cartesian meshes for scalable mesh generation and partitioning; (Adapted) octree meshes can be generated/partitioning fast/scalable (e.g., `p4est` [Burstedde et al'11, Issac et al'15])

   Showstopper: Very ill-conditioned problems

2) To extend an optimal/scalable linear solver (BDDC) to unfitted meshes

Table of Contents

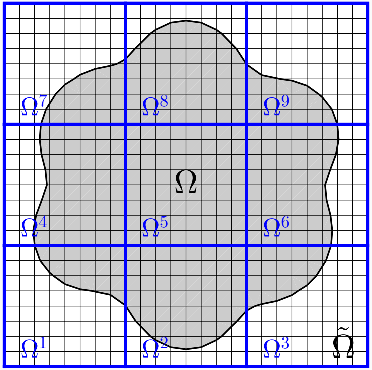
Problem setup

Model problem (Poisson equation)

$$\begin{cases} -\Delta u = f & \text{in } \Omega, \\ u = g^{\mathrm{D}} & \text{on } \Gamma^{\mathrm{D}}, \\ \nabla u \cdot n = g^{\mathrm{N}} & \text{on } \Gamma^{\mathrm{N}}. \end{cases}$$

Discretization with an (adaptive) Cartesian grid

+ Easy to generate
+ Easy to partition into sub-domains
- Difficult to impose Dirichlet BC
- Difficult to integrate the weak form
- Difficult for iterative linear solvers



Notation:

Ω: Physical domain

$\tilde{\Omega}$: Extended domain

Imposing Dirichlet BC with Nitsche's method

Variatonal problem (Nitsche-XFEM)

$$\begin{cases} \text{Find } u_h \in V_h \text{ such that} \\ a(v_h, u_h) = l(v_h) \quad \forall v_h \in V_h, \end{cases}$$

with

$$\begin{aligned} a(v,u) &:= \int_\Omega \nabla v \cdot \nabla u \; \mathrm{d}V + \int_{\Gamma_\mathrm{D}} \left(\beta u v - v\,(n \cdot \nabla u) - u\,(n \cdot \nabla v)\right) \; \mathrm{d}V \\ l(v) &:= \int_\Omega v f \; \mathrm{d}V + \int_{\Gamma_\mathrm{N}} v g^N \; \mathrm{d}S + \int_{\Gamma_\mathrm{D}} \left(\beta g^D v - g^D\,(n \cdot \nabla v)\right) \; \mathrm{d}S, \end{aligned}$$

and $\beta > 0$ is a stability parameter that must be large "enough" to ensure coercivity (Nitsche's method)

Properties

\+ Coercivity (stability)

\+ Consistent (optimal convergence order for high order FEs)

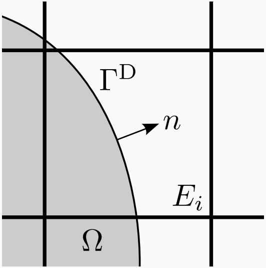
– β can be arbitrary large for cut elements (problems for the linear solver)

Minimum admissible value [de Prenter]

$$\beta_i \geq \sup_{v \in V_h} \frac{b_i(v,v)}{a_i^1(v,v)}$$

with

$$\begin{aligned} b_i(v,u) &:= \int_{\overline{E}_i \cap \Gamma^{\mathrm{D}}} (n \cdot \nabla v)(n \cdot \nabla u) \; \mathrm{d}S \\ a_i^1(v,u) &:= \int_{E_i \cap \Omega} \nabla v \cdot \nabla u \, \mathrm{d}V \end{aligned}$$



Element-wise generalized eigenvalue problem

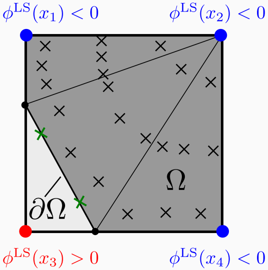
$$B_i x_i = \lambda A_i^1 x_i \Rightarrow \beta_i \geq \lambda_{\max}$$

[de Prenter] F. de Prenter, C.V. Verhoosel, G.J. van Zwieten, E.H. van Brummelen, Condition number analysis and preconditioning of the finite cell method. In: "Comput. Methods Appl. Mech. Engrg.". In press.

Numerical integration in cut elements

Gauss quadrature in sub-triangulation of cut elements

+ Simple and robust approach



We adopt a level-set based boundary representation

$$\partial\Omega := \{x \in \mathbb{R}^d : \phi^{\mathrm{LS}}(x) = 0\}$$

+ Easy to compute intersections
+ Easy to compute sub-triangulation (reduced number of cases)
– Difficult to reconstruct high order surfaces (talk by Reusken)
– Difficult to reconstruct sharp corners

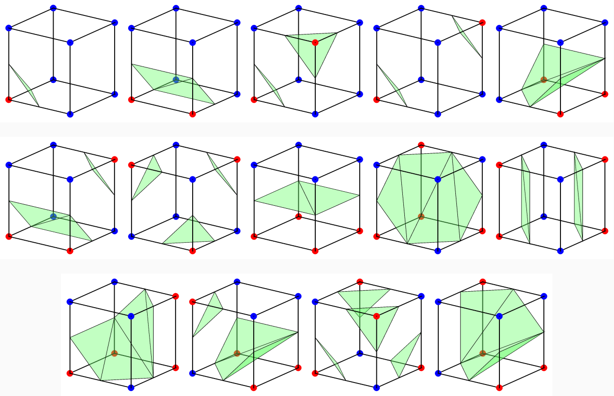
Remark

Other more sophisticated integration and geometry representation can be adopted without changing the preconditioner presented later

Total of 2^8 intersection cases for an hexahedron (only 14 unique intersection cases).

Sub-triangulations can be precomputed and reused !

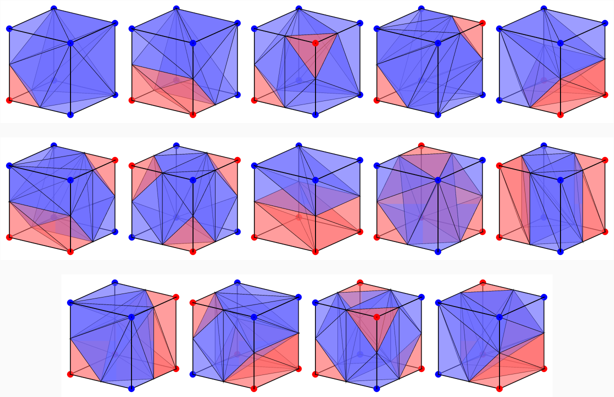
Surface sub-cells



Total of 2^8 intersection cases for an hexahedron (only 14 unique intersection cases).

Sub-triangulations can be precomputed and reused !

Volume sub-cells

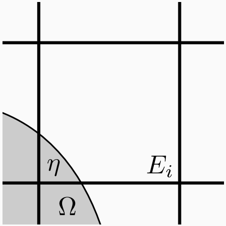


Condition number estimate

– The condition number of the discrete problem (for a fixed grid) scales as [de Prenter]

$$k_2(A) \sim |\eta|^{-(2p+1-2/d)}$$

with η the smallest intersection.

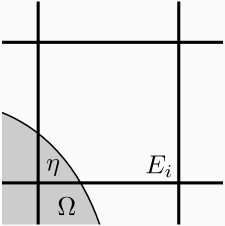


Condition number estimate

– The condition number of the discrete problem (for a fixed grid) scales as [de Prenter]

$$k_2(A) \sim |\eta|^{-(2p+1-2/d)}$$

with η the smallest intersection.



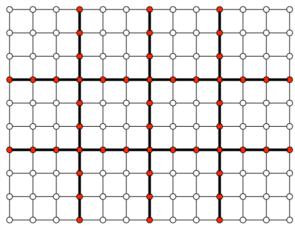
State-of-the-art (solvers for XFEM) : Menk and Bordas'11, Berger-Vergiat et al'12, Hiriyur et al'12, Lang et al'14 [...]

AMG for internal nodes + external nodes send to a coarse solver... limited parallel efficiency

Table of Contents

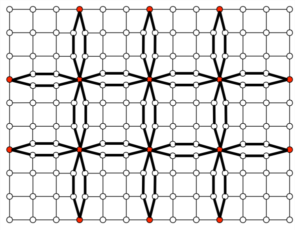
BDDC preconditioning

V_0



BDDC preconditioner [Dohrmann'03, ...]

- Replace V_0 by $\bar{V}_0$ (reduced continuity)



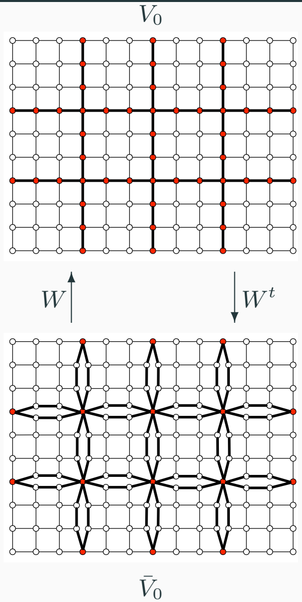
$\bar{V}_0$

[Dohrmann '03] C. R. Dohrmann. A Preconditioner for Substructuring Based on Constrained Energy Minimization. In: SIAM Journal on Scientific Computing 25.1 (2003), pp. 246–258.

BDDC preconditioning

BDDC preconditioner [Dohrmann'03, ...]

- Replace V_0 by $\bar{V}_0$ (reduced continuity)
- Define the injection $W : \bar{V}_0 \longrightarrow V_0$ (weight, comm and add)



[Dohrmann '03] C. R. Dohrmann. A Preconditioner for Substructuring Based on Constrained Energy Minimization. In: SIAM Journal on Scientific Computing 25.1 (2003), pp. 246–258.

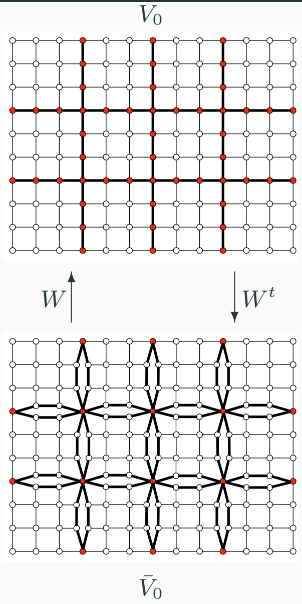
BDDC preconditioning

BDDC preconditioner [Dohrmann'03, ...]

- Replace V_0 by $\bar{V}_0$ (reduced continuity)
- Define the injection $W : \bar{V}_0 \longrightarrow V_0$ (weight, comm and add)
- Find $\bar{u}_0 \in \bar{V}_0$ such that:

$$\bar{u}_0 \in \bar{V}_0 \; : a(\bar{u}_0, \bar{v}_0) = (f, \bar{v}_0) \quad \forall \bar{v}_0 \in \bar{V}_0$$

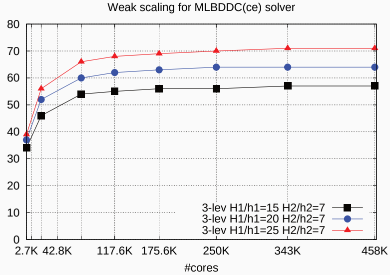
and obtain $u = M_{BDDC} r = \mathcal{E} W \bar{u}_0$, where $\mathcal{E}$ is the harmonic extension operator (correct in the interior of subdomains)



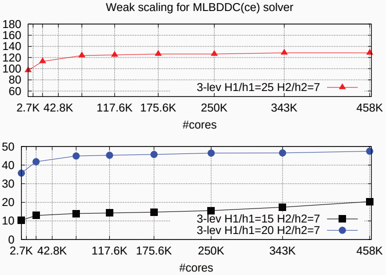
[Dohrmann '03] C. R. Dohrmann. A Preconditioner for Substructuring Based on Constrained Energy Minimization. In: SIAM Journal on Scientific Computing 25.1 (2003), pp. 246–258.

Weak scaling 3-lev BDDC(ce) solver

3D Linear Elasticity problem on IBM BG/Q (JUQUEEN@JSC) w/ FEMPAR
https://gitlab.com/fempar/fempar (A. Martín's talk, 11:45h)



#PCG iterations



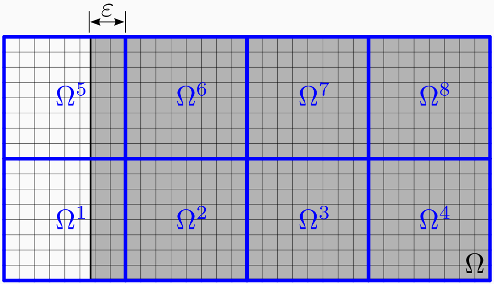
Total time (secs.)

Experiment set-up								
Lev.	# MPI tasks							FEs/core
1st	42.8K	74.1K	117.6K	175.6K	250K	343K	456.5K	**15^3/20^3/25^3**
2nd	125	216	343	512	729	1000	1331	7^3
3rd	1	1	1	1	1	1	1	n/a

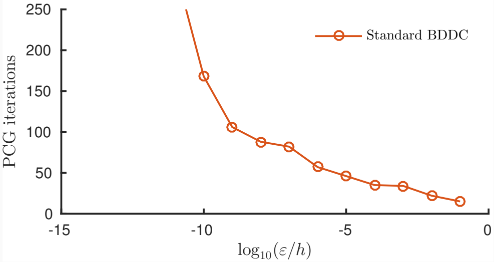
Table of Contents

Problematic example

Poisson equation
Dirichlet BC on $\partial\Omega$
Arbitrary small ε



(!) Standard BDDC cannot be robust with respect to the position of the cut (example)



Remark

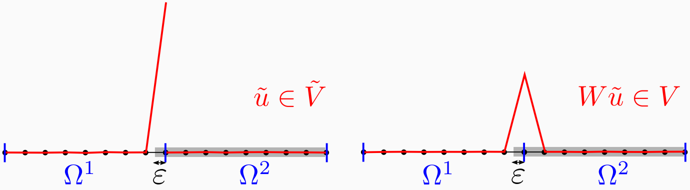
The condition number is computed as

$$k_2(M^{\text{bddc}}A) = \frac{\lambda_{\max}}{\lambda_{\min}} = \frac{\lambda_{\max}}{1} = \sup_{\tilde{u}\in\tilde{V}} \frac{||W\tilde{u}||_a}{||\tilde{u}||_a}$$

which for body-fitted meshes can be bounded as

$$k_2(M^{\text{bddc}}A) \leq C\left(1 + \log^2\left(\frac{H}{h}\right)\right).$$

(!) For cut-elements we can have arbitrarily large condition numbers. Example:

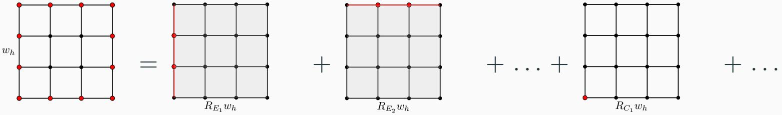


Before weighting   After weighting

$||\tilde{u}||_a \to 0$ for $\varepsilon \to 0$. However $||W\tilde{u}||_a > c$. That is, $\lambda_{\max} \to \infty$.

(Some) basic DD anaysis ingredients (see, e.g., [Toselli & Widlund'05]):

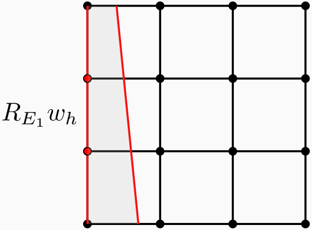
1. Stable decomposition of harmonic functions (corners/edges/faces):
   $w_h = \sum_{\lambda \in \{C,E,F\}} R_\lambda w_h$,
   $$|R_\lambda w_h|_a \leq \beta |w_h|_{a(\Omega_i)}, \qquad \beta = c\left(1 + \log^2(H/h)\right) \text{ for } w_h \in \bar{V}_0$$
2. $R_\lambda w_h = 0$ on $\partial\Omega_i \setminus \lambda$ for all objects
3. Trace theorem (+ harmonic function):
   $$c_- |R_\lambda w_h|_{a(\Omega_i)} \leq |R_\lambda w_h|_{s(\lambda)} \leq c_+ |R_\lambda w_h|_{a(\Omega_i)}$$



DD analysis

(Some) basic DD anaysis ingredients (see, e.g., [Toselli & Widlund'05]):

1. Stable decomposition of harmonic functions (corners/edges/faces): $w_h = \sum_{\lambda \in \{C,E,F\}} R_\lambda w_h$,
$$|R_\lambda w_h|_a \leq \beta |w_h|_{a(\Omega_i)}, \qquad \beta = c\left(1 + \log^2(H/h)\right) \text{ for } w_h \in \bar{V}_0$$
2. $R_\lambda w_h = 0$ on $\partial\Omega_i \setminus \lambda$ for all objects
3. Trace theorem (+ harmonic function):
$$c_- |R_\lambda w_h|_{a(\Omega_i)} \leq |R_\lambda w_h|_{s(\lambda)} \leq c_+ |R_\lambda w_h|_{a(\Omega_i)}$$



The second property is lost in EBM in general

Enlarged coarse spaces

Property (2) only lost when there are nodes in λ that belong to cut elements
"Solution": Consider all these nodes as corner constraints

- All the theory of BDDC methods readily apply (robustness with respect to cuts)
- It can be extremely expensive and induce load balance loss (interface subdomains)

Enlarged coarse spaces

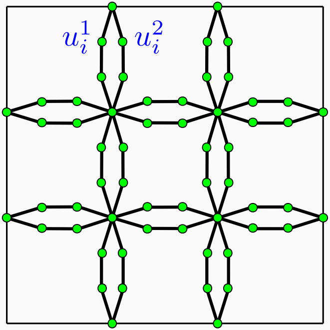
Property (2) only lost when there are nodes in λ that belong to cut elements
"Solution": Consider all these nodes as corner constraints

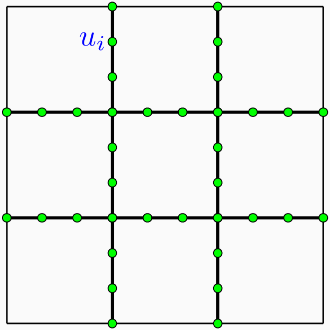
- All the theory of BDDC methods readily apply (robustness with respect to cuts)
- It can be extremely expensive and induce load balance loss (interface subdomains)

In any case, the coarse space can be easily reduced:

- Reducing cut cells touching the interface (attaching cut cells to full cells)
- Neumann bc's easily handled wo/ additional corners (analysis possible)
- Still, costly when many interface cut cells touching Γ^{D}

Alternative weighting operator





Standard weighting

$$u_i = \frac{1}{2}u_i^1 + \frac{1}{2}u_i^2$$

i.e. the mean value.

Stiffness weighting (e.g., in [Dohrmann '03])

$$u_i = \frac{k_{ii}^1}{k_{ii}^1 + k_{ii}^2}u_i^1 + \frac{k_{ii}^2}{k_{ii}^1 + k_{ii}^2}u_i^2$$

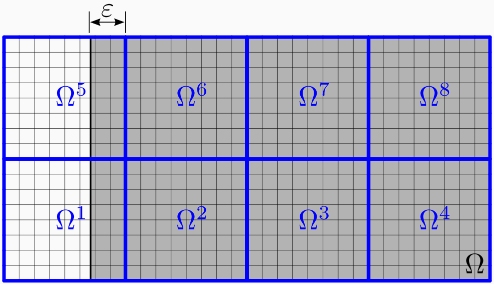
i.e. weighted average using the diagonal entries of the stiffness matrix

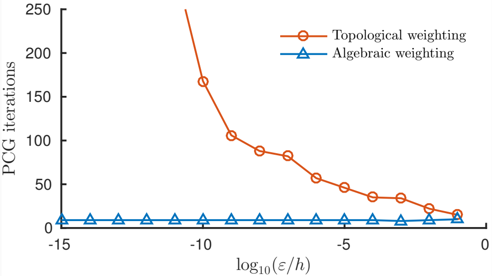
Problematic example (fixed)

Poisson equation
Dirichlet BC on $\partial\Omega$
Arbitrary small ε

Algebraic weighting

+ Very robust method with respect to the position of the interface
– Non-constant weighting within objects: loss of mathematical properties

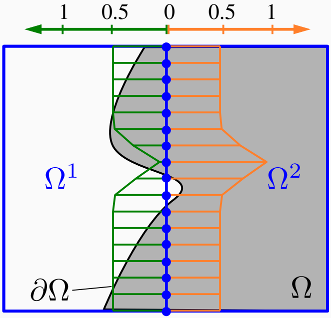




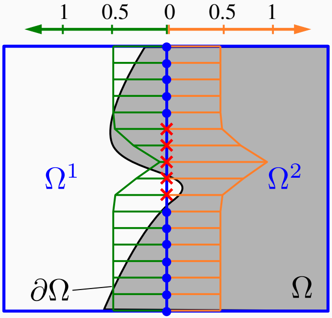
Alternative definition of edges

Motivation

- We require constant weighting coefficient within the objects in mathematical analysis
- Split only edges into new objects with (nearly) constant weighting



Edge object with non-constant weighting

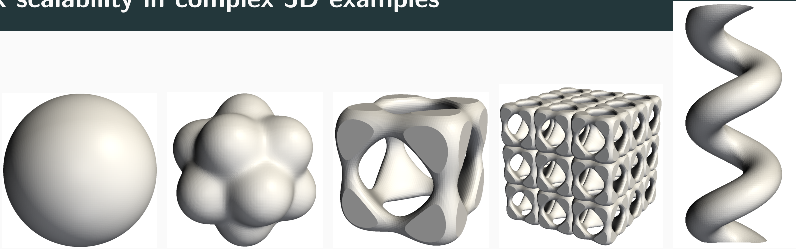


Splitting into new objects with constant weighting

+ It works for Dirichlet BC/Neumann BC
− Larger coarse space

Table of Contents

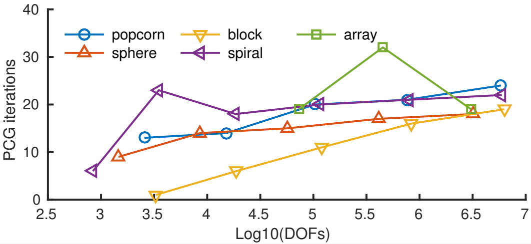
Weak scalability in complex 3D examples



Sphere. Popcorn flake. Block. Array of blocks. Spiral.

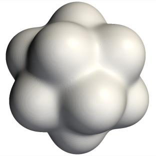
Alternative weighting. No extra corners added.

- Poisson equation
- Fixed ratio $H/h = 8$
- Solver tolerance 10^{-9}

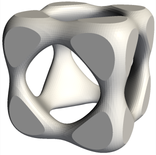


Weak scalability in complex 3D examples

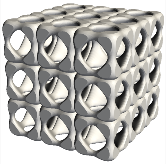
Sphere.



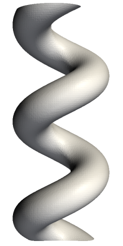
Popcorn flake.



Block.



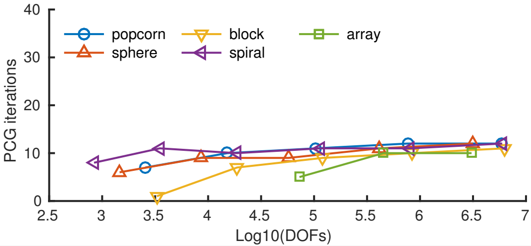
Array of blocks.



Spiral.

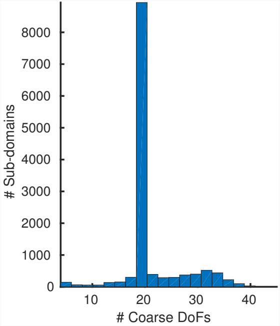
Alternative weighting. Splitting edges (version 1). Dirichlet BC.

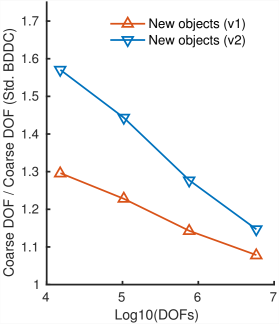
- Poisson equation
- Fixed ratio $H/h = 8$
- Solver tolerance 10^{-9}

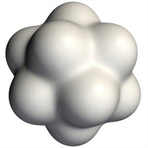


Coarse space size

- – Splitting the edges results in a larger coarse space.
- \+ The increment tends to standard coarse space as more subd's
- Worst case... about twice more expensive than full elements (scalable/robust)







Conclusions and future work

Conclusions:

- Substructuring DD theory cannot be applied to EBM
- Unless expensive coarse spaces being considered
- For Neumann problems, it can be handled (not explained, provably robust)
- Stiffness-based weighting very robust (+ constant weighting on edge objects)
- Heuristic approach, no theory
- Robust + scalable solvers for unfitted methods

Ongoing work

- Mathematical analysis for Nitsche's bc's
- Preconditioners for ghost penalty stabilization strategies [Burman'10]
- Extension to other problems (Navier-Stokes...)
- Multilevel extension (MLBDDC in `FEMPAR`)
- Adaptive Cartesian grids and space filling curves (using `p4est`+`FEMPAR`)