Some Computational Results for Dual-Primal FETI Methods for Elliptic Problems in 3D

Axel Klawonn[1], Oliver Rheinbach[1], and Olof B. Widlund[2]

[1] Universität Duisburg-Essen, Campus Essen, Fachbereich Mathematik (http://www.uni-essen.de/ingmath/Axel.Klawonn,www.uni-essen.de/ingmath/people/rheinbach.html)
[2] Courant Institute of Mathematical Sciences, New York University (http://cs.nyu.edu/cs/faculty/widlund/)

Summary. Iterative substructuring methods with Lagrange multipliers for elliptic problems are considered. The algorithms belong to the family of dual-primal FETI methods which were introduced for linear elasticity problems in the plane by Farhat et al. [2001] and were later extended to three dimensional elasticity problems by Farhat et al. [2000]. Recently, the family of algorithms for scalar diffusion problems was extended to three dimensions and successfully analyzed by Klawonn et al. [2002a,b]. It was shown that the condition number of these dual-primal FETI algorithms can be bounded polylogarithmically as a function of the dimension of the individual subregion problems and that the bounds are otherwise independent of the number of subdomains, the mesh size, and jumps in the diffusion coefficients. In this article, numerical results for some of these algorithms are presented and their relation to the theoretical bounds is studied. The algorithms have been implemented in PETSc, see Balay et al. [2001], and their parallel scalability is analyzed.

1 Elliptic model problem, finite elements, and geometry

Let $\Omega \subset \mathbb{R}^3$, be a bounded, polyhedral region, let $\partial\Omega_D \subset \partial\Omega$ be a closed set of positive measure, and let $\partial\Omega_N := \partial\Omega \setminus \partial\Omega_D$ be its complement. We impose homogeneous Dirichlet and general Neumann boundary conditions, respectively, on these two subsets and introduce the Sobolev space $H^1_0(\Omega, \partial\Omega_D) := \{v \in H^1(\Omega) : v = 0 \text{ on } \partial\Omega_D\}$.

We decompose Ω into non-overlapping subdomains $\Omega_i, i = 1, \ldots, N$, where each is the union of shape-regular elements with the finite element nodes on the boundaries of neighboring subdomains matching across the interface Γ. The interface Γ is the union of subdomain faces, which are shared by two subregions, edges which are shared by more than two subregions and vertices which form the endpoints of edges. All of them are regarded as open sets.

For simplicity, we will only consider a piecewise trilinear, conforming finite element approximation of the following scalar, second order model problem:

find $u \in H_0^1(\Omega, \partial\Omega_D)$, such that

$$a(u,v) = f(v) \quad \forall v \in H_0^1(\Omega, \partial\Omega_D), \tag{1}$$

where

$$a(u,v) = \sum_{i=1}^{N} \rho_i \int_{\Omega_i} \nabla u \cdot \nabla v dx, \quad f(v) = \sum_{i=1}^{N} \Big(\int_{\Omega_i} f v dx + \int_{\partial\Omega_i \cap \partial\Omega_N} g_N v ds \Big), \tag{2}$$

where g_N is the Neumann boundary data defined on $\partial\Omega_N$. We further assume that the diffusion coefficient ρ_i is a positive constant on each subregion Ω_i.

For the theoretical analysis, we also make a number of further technical assumptions; see Klawonn et al. [2002a,b] for details.

2 The FETI-DP Method

For each subdomain $\Omega_i, i = 1, \ldots, N$, we assemble local stiffness matrices $K^{(i)}$ and local load vectors $f^{(i)}$. We denote by $u^{(i)}$ the local solution vectors of nodal values. The local stiffness matrices $K^{(i)}$ can be partitioned according to vertex and remaining degrees of freedom, denoted by subscript c and r, respectively.

$$K^{(i)} = \begin{bmatrix} K_{rr}^{(i)} & K_{rc}^{(i)} \\ {K_{rc}^{(i)}}^T & K_{cc}^{(i)} \end{bmatrix}, \quad u^{(i)} = \begin{bmatrix} u_r^{(i)} \\ u_c^{(i)} \end{bmatrix}, \quad f^{(i)} = \begin{bmatrix} f_r^{(i)} \\ f_c^{(i)} \end{bmatrix}, \qquad i = 1, \ldots, N.$$

By assembling the stiffness matrix contributions from the vertices, we obtain from the local submatrices $K_{cc}^{(i)}$ the global matrix $\widetilde{K}_{cc}$ and from the local matrices $K_{rc}^{(i)}$ the partially assembled matrices $\widetilde{K}_{rc}^{(i)}$. Here, we choose to assemble at all vertices. It is also possible to take only a sufficient number of them; for details, see Klawonn et al. [2002a]. We introduce the following notation $K_{rr} := diag_{i=1}^{N}(K_{rr}^{(i)})$ and $\widetilde{K}_{rc} := [\widetilde{K}_{rc}^{(1)T} \cdots \widetilde{K}_{rc}^{(N)T}]^T$. The global vectors $\widetilde{u}_c$ and $\widetilde{f}_c$ are defined accordingly. We note that the FETI-DP iterates will be continuous at all vertices throughout the iterations.

To guarantee continuity at the remaining interface nodes, i.e., those which are not vertices, we introduce the jump operator $B_r = [B_r^{(1)}, \ldots, B_r^{(N)}]$. The entries of this matrix are $0, 1, -1$ and it is constructed such that components of any vector u_r, which are associated with the same node on the interface Γ, coincide when $B_r u_r = \sum_{i=1}^{N} B_r^{(i)} u_r^{(i)} = 0$.

We can now reformulate the finite element discretization of (1) as

$$\begin{bmatrix} K_{rr} & \widetilde{K}_{rc} & B_r^T \\ \widetilde{K}_{rc}^T & \widetilde{K}_{cc} & 0 \\ B_r & 0 & 0 \end{bmatrix} \begin{bmatrix} u_r \\ \widetilde{u}_c \\ \lambda \end{bmatrix} = \begin{bmatrix} f_r \\ \widetilde{f}_c \\ 0 \end{bmatrix}. \tag{3}$$

Elimination of the primal variables u_r and $\widetilde{u}_c$ leads to a reduced linear system of the form

$$F_A \lambda = d_A,$$

where the matrix F_A and the right hand side d_A are formally obtained by block Gauss elimination. Let us note that the matrix F_A is never built explicitly but that in every iteration appropriate linear systems are solved; see Farhat et al. [2000] or Klawonn et al. [2002a] for more details.

To obtain better convergence properties for three dimensional problems, a larger coarse problem was suggested by introducing additional optional constraints of the form

$$Q_r u_r = 0. \tag{4}$$

Here, $Q_r := [Q_r^{(1)} \dots Q_r^{(N)}]$, $Q_r^{(i)} := [O \quad Q_\Delta B_\Delta^{(i)}]$, and Q_Δ is a rectangular matrix which has as many columns as there are remaining degrees of freedom which are on the interface; for the latter set, we will also use the subscript Δ. The number of rows is determined by the number of primal edges and faces. A primal edge is an edge where the average of u is the same across this edge whichever component of the product space is used in its computation. Analogously, we define a primal face. The matrix Q_Δ is constructed such that (4) guarantees that certain linear combinations of the rows of $B_\Delta u_\Delta$ are zero. These linear combinations are related to primal edges and faces. Then, (4) enforces that averages at primal edges and faces have common values across the interface.

Introducing additional optional Lagrange multipliers μ to enforce the extra constraints given in (4), we obtain from (3) the following linear system

$$\begin{bmatrix} K_{rr} & \widetilde{K}_{rc} & Q_r^T & B_r^T \\ \widetilde{K}_{rc}^T & \widetilde{K}_{cc} & 0 & 0 \\ Q_r & 0 & 0 & 0 \\ B_r & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_r \\ \widetilde{u}_c \\ \mu \\ \lambda \end{bmatrix} = \begin{bmatrix} f_r \\ \widetilde{f}_c \\ 0 \\ 0 \end{bmatrix}. \tag{5}$$

Elimination of $u_r, \widetilde{u}_c$, and μ leads again to a reduced linear system of the form

$$F\lambda = d, \tag{6}$$

where the matrix F and the right hand side d are again formally obtained by block Gauss elimination.

Let us now define the Dirichlet preconditioner. We need a scaled jump operator $B_{D,r}$. It is obtained from $B_r = [O \quad B_\Delta]$ by scaling B_Δ subdomainwise with appropriate diagonal scaling matrices $D^{(i)}$ and setting $B_{D,\Delta} := [D^{(1)} B_\Delta^{(1)} \dots D^{(N)} B_\Delta^{(N)}]$. The scaling matrices $D^{(i)}$ are defined using the diffusion coefficients ρ_i; for details, see Klawonn et al. [2002a]. Finally, we add a zero column to $B_{D,r}$ for each vertex node. From the local stiffness matrices $K^{(i)}$, we obtain local Schur complements $S^{(i)}$, by eliminating the interior

variables, which operate on the degrees of freedom belonging to the interface nodes. Let us define the block diagonal matrix $S := diag_{i=1}^{N}(S^{(i)})$. The Dirichlet preconditioner is then defined as

$$M^{-1} := B_{D,r} S B_{D,r}^T.$$

The FETI-DP algorithms are preconditioned conjugate gradient methods for solving the preconditioned linear system

$$M^{-1} F \lambda = M^{-1} d.$$

Following the notation in Klawonn et al. [2002a,b], we denote the algorithm using just vertex constraints by Algorithm A. For those methods which additionally use optional constraints, we denote the method choosing all edges and faces as primal by Algorithm B, the one using all edges by Algorithm C, and finally the algorithm which uses just faces by Algorithm E. We denote the corresponding matrix F in (6) by F_B, F_C, and F_E.

3 Theoretical Estimates

For Algorithms A, B, C, and E, we have the following estimates; cf. Klawonn et al. [2002a,b].

Theorem 1. *The condition numbers satisfy*

1. $\kappa(M^{-1}F_A) \leq C(H/h)(1+\log(H/h))^2$,
2. $\kappa(M^{-1}F_B) \leq C(1+\log(H/h))^2$,
3. $\kappa(M^{-1}F_C) \leq C(1+\log(H/h))^2$,
4. $\kappa(M^{-1}F_E) \leq C\max((1+\log(H/h))^2, TOL*(1+\log(H/h)))$,

where $C > 0$ is a constant which is independent of H, h, TOL, and the values of the coefficients ρ_i.

We note that the condition number estimate for Algorithm E is only valid if, for all pairs of substructures Ω_i, Ω_k, which have an edge $\mathcal{E}^{ik}$ in common, we have an *acceptable face path*. An acceptable face path is a path from Ω_i to Ω_k, possibly via several other substructures Ω_j, which do not necessarily touch the edge in question, such that the associated coefficients ρ_j, ρ_i, and ρ_k satisfy $TOL * \rho_j \geq \min(\rho_i, \rho_k)$ for some chosen tolerance TOL.

4 Computational results

We have applied the FETI-DP algorithms A, B, C, and E to the model problem (1), where $\Omega := [0,1]^3$ is the unit cube. We decompose the unit cube into $N \times N \times N$ cubic subdomains with sidelength $H := 1/N$. The diffusion

coefficients ρ_i alternate between 1 and 10^4 and are distributed in a threedimensional checkerboard pattern; cf. Figure 1. On the front, left, and bottom part, homogeneous Dirichlet boundary conditions are applied. On all the remaining parts of the boundary, we imposed homogeneous Neumann boundary conditions. The coefficients are constant on each subdomain and (1) is discretized by conforming trilinear elements with finite element diameter h. All algorithms are implemented in PETSc, see Balay et al. [2001]. We use the preconditioned conjugate gradient method with a zero initial guess. The stopping criterion is the relative reduction of the initial residual by 10^{-7} in the Euclidean norm. In order to analyze the numerical scalability of our algo-

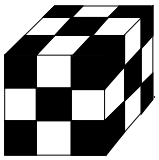

Fig. 1. Model domain decomposed into cubes with discontinuous diffusion coefficients $\rho_i = 1$ and $\rho_i = 10^4$.

rithms, we have carried out two different types of experiments. In our first set of runs, we kept the subdomain size H/h fixed and increased the number of subdomains and thus the overall problem size; cf. Tables 1,2,3,4. Our second series of experiments is carried out with a fixed number of subdomains and an increasing subdomain size H/h resulting in an increased $1/h$; cf. Tables 5 and 6 and Figure 2. From both set of runs, we see that our computational results support the theoretical condition number estimates. However, for Algorithm E, we cannot decide if the growth of the condition number is polylogarithmic. From the range of H/h used in the experiments, it rather looks linear than polylogarithmic. We note that for this problem, the bound of Theorem 1 is basically meaningless since $TOL = 10^4$. Experiments for an isotropic material, i.e., with no jumps in the coefficients show the same polylogarithmic growth as Algorithms B and C. This is an interesting point which needs some further analysis. In a third set of experiments, we have tested our algorithms for parallel scalability. We considered a decomposition into 216 subdomains with 13824 degrees of freedom for each subdomain which yields an overall problem size of 2 685 619 degrees of freedom; cf. Table 7.

The experiments in Tables 1,2,3,4 were carried out on two dual Athlon MP 2200+ PCs with 2 GByte memory each. The experiments in Tables 5,6 and 7 were computed on the 350 node Linux cluster Jazz at the Argonne National Laboratory. Each node is a 2.4 GHz Pentium Xeon where half of the nodes has 2 GByte memory and the other half has 1 GByte.

The experiments show that all algorithms have a good parallel scalability for our model problem. For this problem and the number of degrees of freedom considered, the CPU times are not significantly different, although Algorithm

C is always slightly faster. To decide which method is the best, more extensive testing with different model problems and geometries is needed. This is currently ongoing research and will be published elsewhere.

Table 1. Algorithm A - Constant H/h

Subdomains	Dof/Subdom.	Dof	Iterations	$\lambda_{\min}$	$\lambda_{\max}$
8	1000	6,859	**9**	1.00035	11.5539
27	1000	21,952	**14**	1.00051	28.8335
64	1000	50,653	**19**	1.00361	25.0130
125	1000	97,336	**22**	1.00283	28.8335
216	1000	166,375	**24**	1.00231	25.0127
343	1000	262,144	**26**	1.00188	28.8335
512	1000	389,017	**25**	1.00161	25.0127
729	1000	551,368	**26**	1.00138	28.8335
1000	1000	753,571	**24**	1.00125	25.0127

Table 2. Algorithm B - Constant H/h

Subdomains	Dof/Subdom.	Dof	Iterations	$\lambda_{\min}$	$\lambda_{\max}$
8	1000	6,859	**7**	1.00085	1.47091
27	1000	21,952	**8**	1.00049	1.55036
64	1000	50,653	**8**	1.00025	1.47011
125	1000	97,336	**8**	1.00022	1.55036
216	1000	166,375	**8**	1.00013	1.46995
343	1000	262,144	**8**	1.00013	1.55036
512	1000	389,017	**8**	1.00009	1.46989
729	1000	551,368	**8**	1.00010	1.55036
1000	1000	753,571	**7**	1.00014	1.46985

Acknowledgement. The authors gratefully acknowledge the use of "Jazz", a 350 node computing cluster operated by the Mathematics and Computer Science Division at Argonne National Laboratory as part of its Laboratory Computing Resource Center.

References

Satish Balay, Kris Buschelman, William D. Gropp, Dinesh Kaushik, Matt Knepley, Lois Curfman McInnes, Barry F. Smith, and Hong Zhang. PETSc home page. URL `http://www.mcs.anl.gov/petsc`. 2001.

Table 3. Algorithm C - Constant H/h

Subdomains	Dof/Subdom.	Dof	Iterations	$\lambda_{\min}$	$\lambda_{\max}$
8	1000	6,859	**8**	1.00030	1.61492
27	1000	21,952	**9**	1.00040	2.06800
64	1000	50,653	**9**	1.00020	1.93210
125	1000	97,336	**10**	1.00012	2.06875
216	1000	166,375	**9**	1.00009	1.93192
343	1000	262,144	**10**	1.00008	2.06875
512	1000	389,017	**9**	1.00006	1.93210
729	1000	551,368	**10**	1.00005	2.06875
1000	1000	753,571	**9**	1.00005	1.93210

Table 4. Algorithm E - Constant H/h

Subdomains	Dof/Subdom.	Dof	Iterations	$\lambda_{\min}$	$\lambda_{\max}$
8	1000	6,859	**8**	1.00102	11.4671
27	1000	21,952	**10**	1.00185	16.2107
64	1000	50,653	**14**	1.00129	16.2191
125	1000	97,336	**16**	1.00113	16.2246
216	1000	166,375	**19**	1.00089	16.2281
343	1000	262,144	**19**	1.00079	16.2304
512	1000	389,017	**20**	1.00067	16.2319
729	1000	551,368	**20**	1.00060	16.2329
1000	1000	753,571	**20**	1.00054	16.2335

Table 5. Algorithms A and C - Constant H

Subdomains	H/h	Dof	**Algorithm A**			**Algorithm C**		
			Iter	$\lambda_{\min}$	$\lambda_{\max}$	Iter	$\lambda_{\min}$	$\lambda_{\max}$
216	4	6,859	14	1.00018	4.20279	6	1.00001	1.28960
216	8	79,507	22	1.00147	16.7662	8	1.00029	1.75693
216	12	300,763	27	1.00306	34.0512	10	1.00010	2.08459
216	16	753,571	31	1.00371	53.9590	11	1.00017	2.34317
216	20	1,520,875	32	1.00519	75.7574	11	1.00024	2.55999
216	24	2,685,619	34	1.00651	99.0372	12	1.00029	2.74869
216	28	4,330,747	36	1.00660	123.530	12	1.00035	2.91716
216	32	6,539,203	36	1.00677	149.054	13	1.00034	3.07033

Charbel Farhat, Michel Lesoinne, Patrick LeTallec, Kendall Pierson, and Daniel Rixen. FETI-DP: A dual-primal unified FETI method - part i: A faster alternative to the two-level FETI method. *Int. J. Numer. Meth. Engrg.*, 50:1523–1544, 2001.

Charbel Farhat, Michel Lesoinne, and Kendall Pierson. A scalable dual-primal domain decomposition method. *Numer. Lin. Alg. Appl.*, 7:687–714, 2000.

Table 6. Algorithms B and E - Constant H

Subdomains	H/h	Dof	**Algorithm B**			**Algorithm E**		
			Iter	$\lambda_{\min}$	$\lambda_{\max}$	Iter	$\lambda_{\min}$	$\lambda_{\max}$
216	4	6,859	5	1.01252	1.06768	13	1.00006	4.19816
216	8	79,507	7	1.00052	1.31862	19	1.00044	12.1453
216	12	300,763	8	1.00021	1.62065	22	1.00058	20.3391
216	16	753,571	10	1.00021	1.90164	23	1.00054	28.5889
216	20	1,520,875	10	1.00033	2.14742	23	1.00066	36.8711
216	24	2,685,619	11	1.00040	2.36688	25	1.00062	45.1044
216	28	4,330,747	12	1.00040	2.61352	24	1.00081	53.3703
216	32	6,539,203	12	1.00046	2.80160	24	1.00097	61.5779

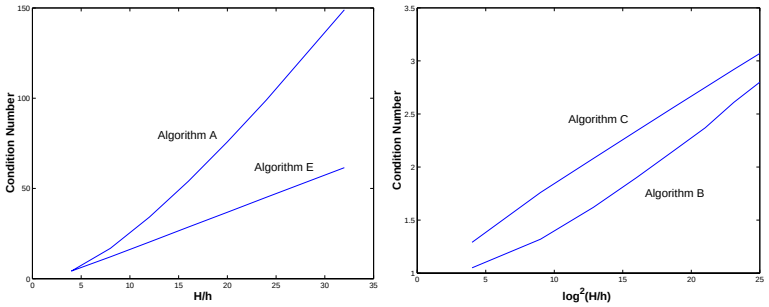

Fig. 2. Condition number growth for varying H/h for Algorithms A and E (left) and Algorithms B and C (right).

Table 7. Parallel Scalability - Algorithms A, B, C and E with 216 subdomains, 13824 dof for each subdomain (2,685,619 dof).

Processors	**Algorithm**			
	A	**B**	**C**	**E**
27	223s	207s	205s	216s
54	113s	106s	106s	110s
108	57.0s	54.2s	53.8s	55.4s
216	29.1s	28.9s	27.2s	29.1s

Axel Klawonn, Olof B. Widlund, and Maksymilian Dryja. Dual-Primal FETI methods for three-dimensional elliptic problems with heterogeneous coefficients. *SIAM J.Numer.Anal.*, 40, 159-179 2002a.

Axel Klawonn, Olof B. Widlund, and Maksymilian Dryja. Dual-Primal FETI methods with face constraints. In Luca F. Pavarino and Andrea Toselli, editors, *Recent developments in domain decomposition methods*, pages 27–40. Springer-Verlag, Lecture Notes in Computational Science and Engineering, Volume 23, 2002b.