

Additive Schwarz preconditioners for a fully-discrete and symmetric boundary element method

Thanh Tran*

(Received 7 August 2000)

Abstract

We discuss various additive Schwarz preconditioners for a fully-discrete and symmetric boundary element method when used to solve a Dirichlet problem in the plane. These preconditioners work in the

*Centre for Mathematics and its Applications, School of Mathematical Sciences, Australian National University, Canberra ACT 0200, AUSTRALIA.

<mailto:Thanh.Tran@anu.edu.au>

⁰See <http://anziamj.austms.org.au/V42/CTAC99/Tran> for this article and ancillary services, © Austral. Mathematical Soc. 2000. Published 27 Nov 2000.

same way as when they are used for the Galerkin boundary element method: the condition numbers of the preconditioned stiffness matrices grow at most logarithmically with the degree of freedom. Several numerical results are presented to support the theory.

Contents

1	Introduction	C1422
2	The boundary-integral equation	C1423
3	The fully-discrete and symmetric method	C1425
4	Additive Schwarz preconditioners	C1428
4.1	General framework	C1428
4.2	Two-level methods	C1431
4.2.1	Non-overlapping method	C1432
4.2.2	Overlapping method	C1433
4.3	Multilevel methods	C1435
5	Implementation	C1436
6	Numerical results	C1437
7	Conclusion	C1440

1 Introduction

We consider in this paper a boundary-value problem for the Laplace equation in a bounded domain in \mathbb{R}^2 with smooth boundary. Via the standard fundamental solution, we reformulate the problem as a boundary-integral equation of the first kind with a symmetric kernel. A fully-discrete and symmetric method for this equation was suggested in [5], which results in solving a linear system the coefficient matrix \mathcal{A} of which is *dense*. If N is the size of \mathcal{A} , then the Gauss solver requires $O(N^3)$ operations for computation of the coefficients giving the solution of the fully-discrete method. Hence when N is large one resorts to iterative methods, among which the conjugate gradient method is the most practical and efficient one. Since \mathcal{A} is ill-conditioned, in the sense that its condition number increases with N , the convergence rate of the iterative method deteriorates, which leads to a large number of iterations needed. Preconditioners are necessary to reduce the cost of computation.

In this paper we shall discuss the use of additive Schwarz methods as preconditioners for the above linear system. Non-overlapping and overlapping two-level methods, and a multilevel method are discussed. We shall prove that the condition numbers of the preconditioned systems grow at most logarithmically with the degrees of freedom.

For a fairly complete list of references to works on preconditioners for boundary-integral equations, the reader is referred to [8]. Multigrid analysis of finite element methods with numerical integration for partial differential equations has been discussed in [1, 3].

We should like to note that for the problem considered in this paper, due to the smoothness of the solution, a direct method may be sufficient in solving the linear system. However, this study is a step along the way to developing preconditioners for more demanding problems, where they are truly needed. Furthermore, the results in this paper apply not only to the logarithmic boundary integral equation but also to any equation of the form (6) in which the bilinear form $l(\cdot, \cdot)$ is equivalent to the $H^{-1/2}$ norm.

2 The boundary-integral equation

We consider the problem

$$\begin{aligned} \Delta U &= 0 && \text{in } \Omega \\ U &= f && \text{on } \Gamma, \end{aligned} \tag{1}$$

where $\Omega \subset \mathbb{R}^2$ is a bounded domain with smooth boundary Γ . Let $\nu_{\mathbf{y}}$ denote the outward unit normal at $\mathbf{y} \in \Gamma$. It is known that U can be represented as

$$U(\mathbf{x}) := \frac{1}{2\pi} \int_{\Gamma} \phi(\mathbf{y}) \log \frac{\alpha}{|\mathbf{x} - \mathbf{y}|} d\sigma_{\mathbf{y}} - \frac{1}{2\pi} \int_{\Gamma} f(\mathbf{y}) \frac{\partial}{\partial \nu_{\mathbf{y}}} \log \frac{1}{|\mathbf{x} - \mathbf{y}|} d\sigma_{\mathbf{y}},$$

for $\mathbf{x} \in \Omega$, where ϕ is the normal derivative of U with respect to ν , and α is a positive parameter to be specified later. Taking the trace on Γ and using the jump relations of the single- and double-layer potentials, we deduce that ϕ satisfies

$$\frac{1}{2\pi} \int_{\Gamma} \phi(\mathbf{y}) \log \frac{\alpha}{|\mathbf{x} - \mathbf{y}|} d\sigma_{\mathbf{y}} = F(\mathbf{x}), \quad \mathbf{x} \in \Gamma, \quad (2)$$

where

$$F(\mathbf{x}) := \frac{1}{2}f(\mathbf{x}) + \frac{1}{2\pi} \int_{\Gamma} f(\mathbf{y}) \frac{\partial}{\partial \nu_{\mathbf{y}}} \log \frac{1}{|\mathbf{x} - \mathbf{y}|} d\sigma_{\mathbf{y}}, \quad \mathbf{x} \in \Gamma.$$

Let γ be a 1-periodic parametric representation of Γ such that $\gamma \in C^{\infty}$ and $|\gamma'(x)| \neq 0$ for $x \in \mathbb{R}$ and let

$$u(x) := \frac{1}{2\pi} \phi[\gamma(x)] |\gamma'(x)| \quad \text{and} \quad g(x) := F[\gamma(x)].$$

Then we can recast (2) as

$$\int_0^1 K(x, y) u(y) dy = g(x), \quad 0 \leq x \leq 1, \quad (3)$$

where

$$K(x, y) := \log \frac{\alpha}{|\gamma(x) - \gamma(y)|} \quad \forall x, y \in [0, 1]. \quad (4)$$

Let

$$l(v, w) := \int_0^1 \int_0^1 K(x, y) v(x) w(y) dx dy. \quad (5)$$

We solve (3) in the weak form: find $u \in H^{-1/2}$ such that

$$l(u, v) = \langle g, v \rangle \quad \forall v \in H^{-1/2}. \quad (6)$$

Here, for all $s \in \mathbb{R}$, H^s denotes the usual 1-periodic Sobolev space of order s with norm denoted by $\|\cdot\|_s$.

The parameter α in the definition (4) of K is chosen to satisfy

$$\alpha \geq \text{logarithmic capacity of } \Gamma, \quad (7)$$

so that

$$l(v, v) \geq \lambda \|v\|_{-1/2}^2 \quad \forall v \in H^{-1/2}. \quad (8)$$

For an explanation of the logarithmic capacity, see e.g. [4]. In practice, a simple way of guaranteeing that (7) holds is to choose α to be greater than the diameter of Γ .

3 The fully-discrete and symmetric method

In the standard Galerkin method, we seek a numerical solution $u_h^* \approx u$ by choosing a finite dimensional space S_h and requiring that $u_h^* \in S_h$ satisfy

$$l(u_h^*, v) = \langle g, v \rangle \quad \forall v \in S_h. \quad (9)$$

Galerkin's method is a standard discretisation of (6), but in the system of linear equations arising from (9) the coefficients and right-hand sides involve

complicated integrals that usually cannot be evaluated analytically. Hence, we are led to consider a fully-discrete Galerkin method in which the numerical solution $u_h \in S_h$ satisfies

$$l_h(u_h, v) = \langle g, v \rangle_h \quad \forall v \in S_h. \quad (10)$$

Here l_h and $\langle \cdot, \cdot \rangle_h$ are certain discrete approximations to l and $\langle \cdot, \cdot \rangle$. The forms l_h and $\langle \cdot, \cdot \rangle_h$ are constructed in such a way that u_h achieves the same rates of convergence as u_h^* in the appropriate Sobolev norms. In [5] l_h is designed to preserve the symmetry of the exact problem so that

$$l_h(v, w) = l_h(w, v) \quad \forall v, w \in S_h.$$

In the following we choose S_h to be the space of piecewise-constant functions on a 1-periodic uniform mesh with N subintervals between 0 and 1 defined by $t_j := jh$ for $j = 0, \dots, N$, where $h := 1/N$.

We follow [5, Section 5] to define l_h as follows. Let

$$a = 1/2, \quad b = 0.1262814358793956, \quad c = 0.6841827226000933, \quad d = 1/6,$$

or

$$a = 1/2, \quad b = 0.4992036265530720, \quad c = 0.0398933386986113, \quad d = 1/6,$$

and let $s_1 = s_6 = a$, $s_2 = s_5 = 0$, $s_3 = s_4 = -a$, $\sigma_1 = \sigma_6 = b$, $\sigma_2 = \sigma_5 = c$, $\sigma_3 = \sigma_4 = b$, $w_1 = w_6 = d$, $w_2 = w_5 = 1/2 - 2d$, and $w_3 = w_4 = d$. With the

quadrature weights being $\{w_1, \dots, w_6\}$ and the quadrature points being

$$(\xi_p, \eta_p) := \frac{1}{2}(1 + s_p - \sigma_p, 1 + s_p + \sigma_p), \quad p = 1, \dots, 6,$$

we define

$$l_h(v, w) := h^2 \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \sum_{p=1}^6 w_p K(x_{jp}, y_{kp}) v(y_{kp}) w(x_{jp}),$$

where $x_{jp} := t_j + \xi_p h$ and $y_{kp} := t_k + \eta_p h$. We note that the quadrature points (ξ_p, η_p) , $p = 1, \dots, 6$, are symmetric about both diagonals of the original unit square.

For the discrete inner product $\langle \cdot, \cdot \rangle_h$ we also choose the quadrature proposed in [5, Section 5] which is a 2-point Gauss–Legendre rule of degree of precision 2 with quadrature points $\xi'_1 = 0.5(1 - 1/\sqrt{3})$, $\xi'_2 = 0.5(1 + 1/\sqrt{3})$, and weights $w'_1 = w'_2 = 0.5$. We then define

$$\langle v, w \rangle_h := h \sum_{j=0}^{N-1} \sum_{p=1}^2 w'_p v(x'_{jp}) w(x'_{jp}),$$

where $x'_{jp} := t_j + \xi'_p h$.

With l_h and $\langle \cdot, \cdot \rangle_h$ defined as above, it follows from [5, Theorem 5.1] that for h sufficiently small there exists a unique solution $u_h \in S_h$ of (10) satisfying

$$\|u_h - u\|_s \leq c h^{t-s} \|u\|_{t+\max(-1-s, 0)},$$

for $s < 1/2$, $-1/2 < t$, and $-1 \leq s \leq t \leq 1$. Moreover, the discrete Galerkin matrix is positive definite.

We note that the method designed as above is of translation-invariant order 3 and that the orders of precision of the left-hand side and right-hand side are both at least 3; see [5, Definition 4.2 and page 337].

4 Additive Schwarz preconditioners

4.1 General framework

In this subsection, we present the additive Schwarz method in an abstract form. We consider a general problem on a finite-dimensional space \mathcal{S}

$$\text{Find } u \in \mathcal{S} \text{ such that } L(u, v) = G(v) \quad \forall v \in \mathcal{S}, \quad (11)$$

where $L(\cdot, \cdot) : \mathcal{S} \times \mathcal{S} \rightarrow \mathbb{R}$ is a symmetric, bounded, and coercive bilinear form, and $G : \mathcal{S} \rightarrow \mathbb{R}$ is a bounded linear functional.

To define an additive Schwarz method for (11), we represent \mathcal{S} as

$$\mathcal{S} = \mathcal{S}_0 + \cdots + \mathcal{S}_J, \quad (12)$$

where \mathcal{S}_i , $i = 0, \dots, J$, are subspaces of \mathcal{S} . Let $P_i : \mathcal{S} \rightarrow \mathcal{S}_i$ be the projection defined by

$$L(P_i v, w) = L(v, w) \quad \forall v \in \mathcal{S}, w \in \mathcal{S}_i. \quad (13)$$

Denoting

$$P := \sum_{i=0}^J P_i, \quad (14)$$

we solve the variational problem (11) by solving

$$Pu = \tilde{G}, \quad (15)$$

where $\tilde{G} := \sum_{i=0}^J G_i$ with $G_i \in \mathcal{S}_i$ being a solution of

$$L(G_i, w) = G(w) \quad \forall w \in \mathcal{S}_i.$$

The l_2 condition number of P is given by $\kappa(P) = \lambda_{\max}(P)/\lambda_{\min}(P)$ where the extremum eigenvalues are related to the decomposition (12) in the following manner, see e.g. [10],

$$\lambda_{\min}(P) = \min_{v \in \mathcal{S}} \frac{L(v, v)}{\min_{\sum v_i = v} \sum_i L(v_i, v_i)} \quad \text{and} \quad \lambda_{\max}(P) = \max_{v \in \mathcal{S}} \frac{L(v, v)}{\min_{\sum v_i = v} \sum_i L(v_i, v_i)}. \quad (16)$$

Here $v = \sum_i v_i$ is a decomposition of v with $v_i \in \mathcal{S}_i$. This decomposition may not be unique. It results from (16) that, see e.g. [10],

$$\kappa(P) \leq \frac{\Lambda_2}{\Lambda_1} \quad (17)$$

if the following two conditions are satisfied:

(C1) For any $v \in \mathcal{S}$ and for any decomposition $v = \sum_i v_i$, $v_i \in \mathcal{S}_i$, there holds

$$L(v, v) \leq \Lambda_2 \sum_{i=0}^J L(v_i, v_i);$$

(C2) For any $v \in \mathcal{S}$ there exists a decomposition $v = \sum_i v_i$, $v_i \in \mathcal{S}_i$, such that

$$\Lambda_1 \sum_{i=0}^J L(v_i, v_i) \leq L(v, v).$$

Our aim is to design additive Schwarz methods so that Λ_2/Λ_1 is bounded independently of (or grows at most logarithmically with) the degree of freedom. For the Galerkin approximation (9), it has been noted that the use of some extra preconditioner is necessary before applying additive Schwarz or multigrid algorithms. In [2], a difference quotient was used, whereas the use of Haar bases was mentioned in [6, 7, 8, 9]. In this paper, Haar bases will be used in the design of additive Schwarz methods for the fully-discrete approximation (10).

Before moving to the description of the methods, we prove the following lemma which yields the equivalence of the two bilinear forms l and l_h .

Lemma 1 *There exists positive constants C_1 and C_2 such that*

$$C_1 l(v, v) \leq l_h(v, v) \leq C_2 l(v, v) \quad \forall v \in S_h.$$

Proof: The boundedness of the form $l(\cdot, \cdot)$ and Theorem 2.1 (i) and Corollary 3.9 in [5] immediately give the left-hand inequality. To prove the right-hand inequality we first note that Lemmas 4.4 and 3.5 in [5] yield, respectively (noting Theorem 2.1 (v) in the same paper),

$$|a_h(v, v)| \leq |a(v, v)| + c \|v\|_{-1/2}^2 \leq c \|v\|_{-1/2}^2 \quad \forall v \in S_h,$$

and

$$|b_h(v, v)| \leq |b(v, v)| + c \|v\|_{-1/2}^2 \quad \forall v \in S_h,$$

where $a_h(\cdot, \cdot)$ and $b_h(\cdot, \cdot)$ are defined as in [5] so that

$$l_h(v, w) = a_h(v, w) + b_h(v, w) \quad \forall v, w \in S_h.$$

(Here and in the sequel c denotes a generic positive constant which may take different values at different occurrences.) Therefore by using (8) we obtain

$$\begin{aligned} l_h(v, v) &\leq |a_h(v, v)| + |b_h(v, v)| \leq c \|v\|_{-1/2}^2 + |b(v, v)| \\ &\leq c \|v\|_{-1/2}^2 + l(v, v) \leq cl(v, v) \quad \forall v \in S_h. \end{aligned}$$

The lemma is proved. 

4.2 Two-level methods

The convergence result for the fully-discrete method (10) discussed in [5] being established under the assumption that the meshes are uniform, we assume in the following uniformity of a two-level mesh to simplify our presentation, even though quasi-uniformity is allowed for the Galerkin method.

The coarse mesh: We first divide I into disjoint subdomains I_i with length H , $i = 1, \dots, J$, so that $\bar{I} = \cup_{i=1}^J \bar{I}_i$.

The fine mesh: Each I_i is further divided into disjoint subintervals I_{ij} with length h , $j = 1, \dots, N$, so that $\bar{I}_i = \cup_{j=1}^N \bar{I}_{ij}$.

We recall that the finite-dimensional space $\mathcal{S} = \mathcal{S}_h$ is defined as the space of 1-periodic piecewise-constant functions on the fine mesh. For both non-overlapping and overlapping methods, we define \mathcal{S}_0 to be the space of 1-periodic piecewise-constant functions on the coarse mesh.

The subspaces \mathcal{S}_i , $i = 1, \dots, J$, will be defined as spaces of functions which are derivatives of piecewise-linear functions. We denote by \mathcal{T} the space of piecewise-linear functions on the fine mesh.

4.2.1 Non-overlapping method

Let

$$\mathcal{T}_i := \{v \in \mathcal{T} \mid \text{supp } v \subset \bar{I}_i\}, \quad i = 1, \dots, J,$$

and let

$$\mathcal{S}_i := \{w \mid \exists v \in \mathcal{T}_i : w = v'\}, \quad i = 1, \dots, J. \quad (18)$$

Bases for \mathcal{S}_i can be formed by the Haar functions; see [9].

The decomposition of \mathcal{S} by (12) with \mathcal{S}_i defined as above allows us to use (13) and (14) to define additive Schwarz operators P_G and P_F when $L = l$ (Galerkin approximation, see (11)) and when $L = l_h$ (fully-discrete method), respectively.

Theorem 2 *The condition number of the non-overlapping additive Schwarz operator P_F is bounded as*

$$\kappa(P_F) = O\left(1 + \log \frac{H}{h}\right). \quad (19)$$

Proof: Lemmas 3.4 and 3.7 in [6] assure us that conditions (C1) and (C2) hold for $L = l$ with

$$\Lambda_1 = c \left(1 + \log \frac{H}{h}\right)^{-1},$$

and Λ_2 being a constant independent of H and h . In view of Lemma 1, the same results hold for $L = l_h$ (with different constants), implying (19); see (17). 

4.2.2 Overlapping method

We now extend each subdomain I_i on each side by a fixed number of subintervals so that the length of the overlap between two extended subdomains

I'_i and I'_{i+1} is 2δ for some $\delta \in (0, H]$ which is a multiple of h . This implies

$$|I'_i| \sim H. \quad (20)$$

The subspaces \mathcal{S}_i , $i = 1, \dots, J$, are now defined as in (18) with \mathcal{T}_i now defined by

$$\mathcal{T}_i := \left\{ v \in \mathcal{T} \mid \text{supp } v \subset \bar{T}'_i \right\}, \quad i = 1, \dots, J.$$

This overlapping decomposition completely defines the additive Schwarz operators P_G and P_F .

Theorem 3 *The condition number of the overlapping additive Schwarz operator P_F is bounded as*

$$\kappa(P_F) = O\left(1 + \log^2 \frac{H}{\delta}\right). \quad (21)$$

Proof: Following the argument in the proof of Lemma 3.4 in [6] and using Lemma 5.2 of [8] in lieu of [6, Eqn (3.17)] we obtain (C2) for $L = l$ with

$$\Lambda_1 = c \left(1 + \log^2 \frac{H}{\delta}\right)^{-1}.$$

The constant Λ_2 in (C1) can also be obtained from Lemma 5.1 in [8], using the same approach. Thus Lemma 1 yields the desired result. 

4.3 Multilevel methods

The multilevel method is defined as follows. Starting with a coarse mesh

$$\mathcal{N}^1 : 0 = x_0^1 < x_1^1 = 1/2 < x_2^1 = 1,$$

we divide each subinterval into two equal intervals. Hence, if h_l is the meshstep of \mathcal{N}^l , $l = 1, \dots, L-1$, then $h_l = 2h_{l+1}$. For $l = 1, \dots, L$ and $i = 0, \dots, N_l := 2^l$, let ϕ_i^l be the hat function which takes the value 1 at x_i^l and zero at the other mesh points. We then define $\mathcal{S}_i^l := \text{span} \{\psi_i^l\}$, where the Haar basis function ψ_i^l is the derivative of ϕ_i^l . The space \mathcal{S} is now decomposed by

$$\mathcal{S} = \mathcal{S}_0 + \sum_{l=2}^L \sum_{i=1}^{N_l} \mathcal{S}_i^l.$$

It was proved in [9] that the condition number of the multilevel additive Schwarz operator P_G is bounded independently of the number of level L and the number of mesh points. Using Lemma 1 again we deduce

Theorem 4 *The condition number of the multilevel additive Schwarz operator P_F is bounded independently of the number of level L and the number of mesh points.*

5 Implementation

We briefly discuss in this section the implementation aspect of the two-level methods. For the implementation of the multilevel method the reader is referred to e.g. [9, 10].

Let ϕ_j be the brick function in the interval (t_{j-1}, t_j) and let \mathbf{A}_{kl} , $k, l = 1, \dots, J$, be sub-blocks of the stiffness matrix \mathbf{A} arising from (10) defined by $\mathbf{A}_{kl} := (a_{ij}^{kl})$ with $a_{ij}^{kl} = l_h(\phi_i, \phi_j)$. Here ϕ_i and ϕ_j are brick functions such that $\text{supp } \phi_i \subset \bar{I}_k$ and $\text{supp } \phi_j \subset \bar{I}_l$ for the non-overlapping method, and $\text{supp } \phi_i \subset \bar{I}'_k$ and $\text{supp } \phi_j \subset \bar{I}'_l$ for the overlapping method. Then \mathbf{A} has the form

$$\begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \cdots & \mathbf{A}_{1J} \\ & \mathbf{A}_{22} & \cdots & \mathbf{A}_{2J} \\ & & \ddots & \vdots \\ & & & \mathbf{A}_{JJ} \end{pmatrix}.$$

The matrix representation \mathbf{B} of the non-overlapping preconditioner will now have the form

$$\begin{pmatrix} \mathbf{A}_{11}^{-1} & \mathbf{O} & \cdots & \mathbf{O} \\ & \mathbf{A}_{22}^{-1} & \cdots & \mathbf{O} \\ & & \ddots & \vdots \\ & & & \mathbf{A}_{JJ}^{-1} \end{pmatrix} + \Pi \mathbf{A}_0^{-1} \Pi^t,$$

where \mathbf{A}_0 is the stiffness matrix on the coarse mesh level, Π is the interpolation operator from the coarse mesh level to the fine mesh level, and

Π^t is its adjoint operator. Let \mathbf{v} be the vector representation of a function $v = \sum_{i=0}^J v_i \in \mathcal{S}$, $v_i \in \mathcal{S}_i$. To compute $\mathbf{B}\mathbf{v}$, there is no need to know the explicit form of \mathbf{B} . It suffices to solve the local problems $\mathbf{A}_{ii}\mathbf{w}_i = \mathbf{v}_i$, $i = 1, \dots, J$, and the global problem $\mathbf{A}_0\mathbf{w}_0 = \Pi^t\mathbf{v}_0$, and obtain $\mathbf{B}\mathbf{v}$ as

$$\mathbf{B}\mathbf{v} = \Pi\mathbf{w}_0 + \sum_{i=1}^J \mathbf{w}_i.$$

However, before doing so one has to transform the representation of the function v in the usual basis to the Haar basis; see Section 4. After obtaining the resulting function from $\mathbf{B}\mathbf{v}$, a reverse transform to the usual basis is necessary.

6 Numerical results

In our numerical experiments we considered the problem (1) for Γ being the ellipse with semi-axes $a_1 = 4$ and $a_2 = 2$. The logarithmic capacity of Γ in this case is $(a_1 + a_2)/2 = 3$. In our numerical calculations we chose $\alpha = 3.5$. The boundary condition was chosen to be $f(\mathbf{x}) = \sqrt{|x_1 + x_2^2|}$, where $\mathbf{x} = (x_1, x_2)$.

We solved the problem using the fully-discrete method described in Section 3. The linear systems were solved via the conjugate gradient method, with and without preconditioning. The iterations were stopped when the

TABLE 1: (1): unpreconditioned, (2): preconditioned by non-overlapping method, (3): preconditioned by overlapping method, (4): preconditioned by multilevel method

DoF	Condition numbers				Iterations			
	(1)	(2)	(3)	(4)	(1)	(2)	(3)	(4)
4	0.366E+01	0.366E+01	0.366E+01	0.234E+01	2	2	2	2
8	0.474E+01	0.474E+01	0.474E+01	0.457E+01	4	4	4	4
16	0.116E+02	0.193E+02	0.474E+01	0.665E+01	7	8	8	7
32	0.242E+02	0.253E+02	0.315E+01	0.290E+01	15	12	11	10
64	0.489E+02	0.319E+02	0.281E+01	0.105E+02	24	14	10	13
128	0.977E+02	0.389E+02	0.105E+02	0.122E+02	34	15	12	13
256	0.195E+03	0.467E+02	0.115E+02	0.144E+02	47	16	13	14
512	0.390E+03	0.550E+02	0.128E+02	0.164E+02	64	17	13	14
1024	0.779E+03	0.641E+02	0.145E+02	0.183E+02	87	18	14	14

TABLE 2: (1): unpreconditioned, (2): preconditioned by non-overlapping method, (3): preconditioned by overlapping method, (4): preconditioned by multilevel method

	CPU times			
DoF	(1)	(2)	(3)	(4)
4	0.635E-01	0.704E-03	0.669E-03	0.648E-01
8	0.106E-02	0.922E-03	0.980E-03	0.468E-02
16	0.153E-02	0.201E-01	0.219E-01	0.106E-01
32	0.410E-02	0.343E-01	0.371E-01	0.206E-01
64	0.142E-01	0.615E-01	0.536E-01	0.396E-01
128	0.572E-01	0.149E+00	0.169E+00	0.695E-01
256	0.594E+00	0.608E+00	0.533E+00	0.264E+00
512	0.719E+01	0.359E+01	0.281E+01	0.165E+01
1024	0.520E+02	0.188E+02	0.150E+02	0.878E+01

l_2 norm of the residual was less than 10^{-9} , and all calculations were performed in double precision. The Lanczos algorithm was used to compute the condition numbers.

Table 1 shows the condition numbers and numbers of iterations for the unpreconditioned systems and the systems preconditioned by the non overlapping, overlapping, and multilevel methods designed in Section 4. Table 2 shows the cpu times for each of the methods.

7 Conclusion

We designed in this paper a non-overlapping and an overlapping two-level method, and a multilevel method for the linear system arising from a fully-discrete and symmetric boundary element method used to solve a Dirichlet problem. The preconditioners work in the same way as in the case of the Galerkin method: the condition number grows at most logarithmically with the degree of freedom.

References

- [1] J. H. Bramble, C. I. Goldstein, and J. E. Pasciak. Analysis of V-cycle multigrid algorithms for forms defined by numerical quadrature. *SIAM J. Sci. Comput.*, 15:566–576, 1994. [C1423](#)
- [2] J. H. Bramble, Z. Leyk, and J. E. Pasciak. The analysis of multigrid algorithms for pseudodifferential operators of order minus one. *Math. Comp.*, 63:461–478, 1994. [C1430](#)
- [3] C. I. Goldstein. Multigrid analysis of finite element methods with numerical integration. *Math. Comp.*, 56:409–436, 1991. [C1423](#)
- [4] E. Hille. *Analytic Function Theory*, volume II. Ginn, Boston, 1962. [C1425](#)
- [5] W. McLean and I. H. Sloan. A fully discrete and symmetric boundary element method. *IMA J. Numer. Anal.*, 14:311–345, 1994. [C1422](#), [C1426](#), [C1426](#), [C1427](#), [C1427](#), [C1428](#), [C1431](#), [C1431](#), [C1431](#), [C1431](#)
- [6] E.P. Stephan and T. Tran. Domain decomposition algorithms for indefinite weakly singular integral equations: the h and p versions. *IMA J. Numer. Anal.*, 20:1–24, 2000. [C1430](#), [C1433](#), [C1434](#), [C1434](#)
- [7] T. Tran. Additive Schwarz algorithms and the Galerkin boundary element method. In J. Noye, M. Teubner, and A. Gill, editors,

- Computational Techniques and Applications: CTAC97*, pages 703–710, Singapore, 1998. World Scientific. C1430
- [8] T. Tran. Overlapping additive Schwarz preconditioners for boundary element methods. *J. Integral Eqns Appl.*, to appear. C1423, C1430, C1434, C1434
- [9] T. Tran and E.P. Stephan. Additive Schwarz methods for the h version boundary element method. *Appl. Anal.*, 60:63–84, 1996. C1430, C1432, C1435, C1436
- [10] X. Zhang. Multilevel Schwarz methods. *Numer. Math.*, 63:521–539, 1992. C1429, C1429, C1436