Multiscale Inversion of Elliptic Operators

Amir Averbuch, Gregory Beylkin,
Ronald Coifman, and Moshe Israeli

Abstract. A fast adaptive algorithm for the solution of elliptic partial differential equations is presented. It is applied here to the Poisson equation with periodic boundary conditions. The extension to more complicated equations and boundary conditions is outlined.

The purpose is to develop algorithms requiring a number of operations proportional to the number of significant coefficients in the representation of the r.h.s. of the equation. This number is related to the specified accuracy, but independent of the resolution. The wavelet decomposition and the conjugate gradient iteration serve as the basic elements of the present approach.

The main difficulty in solving such equations stems from the inherently large condition number of the matrix representing the linear system that result from the discretization. However, it is known

velop a framework for solving problems with general boundary conditions. Let us consider the partial differential equation

$$\mathcal{L}u = f \quad x \in \mathbf{D} \subset \mathbf{R}^{\mathrm{d}}, \tag{1.1}$$

with the boundary condition

$$\mathcal{B}u|_{\partial \mathbf{D}} = g, \tag{1.2}$$

where $\mathcal{L}$ is an elliptic operator,

$$\mathcal{L}u = - \sum_{i,j=1,\ldots,d} (a_{ij}(x)\; u_{x_i})_{x_j} + b(x)\; u, \tag{1.3}$$

and $\mathcal{B}$ is the boundary operator,

$$\mathcal{B}u = \alpha u + \beta \frac{\partial u}{\partial N}. \tag{1.4}$$

We assume that the boundary $\partial\mathbf{D}$ is "complicated." As a practical matter

valid in higher dimensions as well.

We adopt a classical approach to this problem which, until now, was not practical from the numerical point of view. We consider the following steps for solving the problem in (1.1) and (1.2):

In order to realize the preceding steps we need to develop:

1. An algorithm for extending the function f outside the domain $\mathbf{D}$.
2. An efficient method for solving (1.5).
3. An efficient method for solving the boundary integral equation derived from (1.7) and (1.8).
4. An effective algorithm for generating a representation of the solution v of (1.7) and (1.8) once the boundary integral equation is solved

where $\mathbf{V}_j$ is a subspace of an MRA spanned by translations of the scaling function,

$$\phi_{j,\mathbf{k}}(x) = 2^{-jd/2}\phi(2^{-j}x_1 - k_1)\,\phi(2^{-j}x_2 - k_2)\ldots\phi(2^{-j}x_{\mathrm{d}} - k_{\mathrm{d}}), \quad (2.3)$$

where $x = (x_1, \ldots, x_{\mathrm{d}})$ and $k = (k_1, \ldots, k_{\mathrm{d}}) \in \mathbf{Z}^{\mathrm{d}}$. The function ϕ is the scaling function of MRA of $\mathbf{L}^2(\mathbf{R})$.

Let us define the subspaces $\mathbf{W}_j$ as orthogonal complements of $\mathbf{V}_j$ in $\mathbf{V}_{j-1}$,

$$\mathbf{V}_{j-1} = \mathbf{V}_j \oplus \mathbf{W}_j, \quad (2.4)$$

and represent the space $\mathbf{L}^2(\mathbf{R}^{\mathrm{d}})$ as a direct sum

$$\mathbf{L}^2(\mathbf{R}^{\mathrm{d}}) = \mathbf{V}_n \bigoplus_{j\le n} \mathbf{W}_j \quad (2.5)$$

where

$$\langle f, g \rangle = \int_{-\infty}^{\infty} f(x)\, g(x) dx. \tag{2.12}$$

The sum over σ is finite and the number of terms is $2^{\mathrm{d}} - 1$ for each $\mathbf{k}$.

Instead of estimating $M_{\lambda,\mu}$ directly, we may use an iterative approach. For example, solving directly on $M^{\epsilon}_{\text{r.h.s}}$ produces a solution $\tilde{u}$ with accuracy $\tilde{\epsilon} > \epsilon$. Applying the Laplacian to $\tilde{u}$, we generate $\tilde{f}$. Estimating the

In [5] the s-form is used to solve the two-point boundary-value problem. Alternatively, we may use the ns-form. Some care is required at this point since the preconditioned ns-form is dense unlike the s-form, which remains sparse. Thus, in the process of solving the linear system, it is necessary to apply the preconditioner and the ns-form sequentially in order to maintain sparsity. The ns-form is preferable in multiple dimensions since, for example, differential operators require $O(1)$ elements for representation on all scales (see *e.g.* [4]).

We develop a constrained (see below) preconditioned CG algorithm for solving (1.5) in an adaptive manner. Both the s-form and the ns-form may be used for this purpose but it appears that using the ns-form is more efficient, especially if compactly supported wavelets are used and high accuracy is required.

Let us consider (1.5) in the wavelet system of coordinates

$$L_{ns} u_w = f_w, \tag{3.2}$$

where f_w and u_w are representations of f and u in the wavelet system of coordinates. This equation should be understood to include the rules for applying the ns-form (see [6]).

Let us rewrite (3.2) using the preconditioner $\mathcal{P}$ as

$$\mathcal{P} L_{ns} \mathcal{P} v = \mathcal{P} f_w, \tag{3.3}$$

where $\mathcal{P} v = u$. For example, for the second derivative the preconditioner $\mathcal{P}$ is as follows:

$$\mathcal{P}_{il} = \delta_{il} 2^j \tag{3.4}$$

where $1 \leq j \leq n$ is chosen depending on i, l so that $n - n/2^{j-1} + 1 \leq i, l \leq n - n/2^j$, and $\mathcal{P}_{nn} = 2^n$.

The periodized operator Δ has the null space of dimension one which contains constants. If we use the full decomposition (over all n scales) in the construction of the ns-form, then the null space coincides with the

§4 Preconditioner for the operator $-\Delta + \text{Const}$

An "efficient" preconditioner is an essential element in the present approach. In a more restricted sense, "efficient" means insensitive to the size of the problem.

Let us demonstrate how to construct a diagonal preconditioner for the sum of operators $-\Delta + \text{Const}$ in the wavelet bases. We observe that if A and B are diagonal operators with diagonal entries a_i and b_i, then the diagonal operator with entries $1/(a_i + b_i)$ (provided $a_i + b_i \neq 0$) is an ideal preconditioner.

In our case, the operator $-\Delta$ is not diagonal but we know a good diagonal preconditioner for it in wavelet bases (3.4). Let us use this preconditioner instead of $-\Delta$ for the purpose of constructing a preconditioner for $-\Delta + \text{Const}$, where $\text{Const} > 0$. We note that in wavelet bases th [illegible]

identity operator remains unchanged. We restrict $\text{Const} \cdot I$, where I is the identity operator, to the subspace

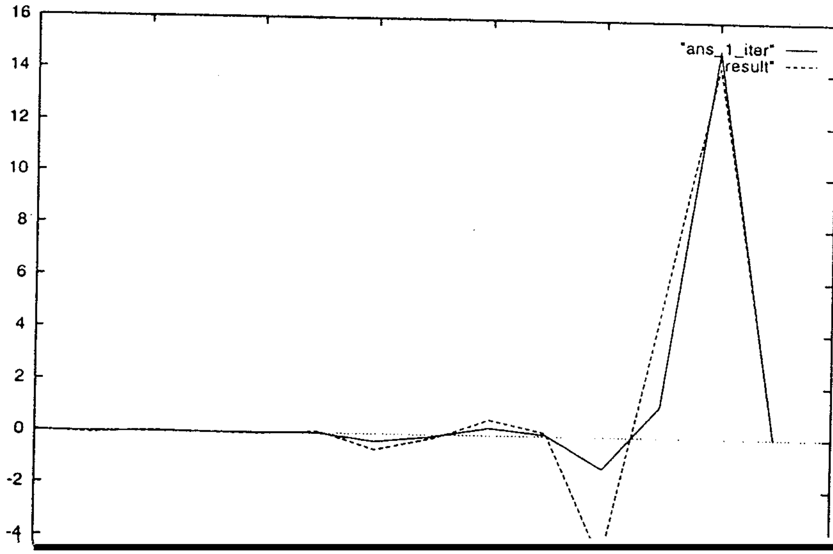
$$\bigoplus_{1 \leq j \leq n} \mathbf{W}_j \tag{4.1}$$

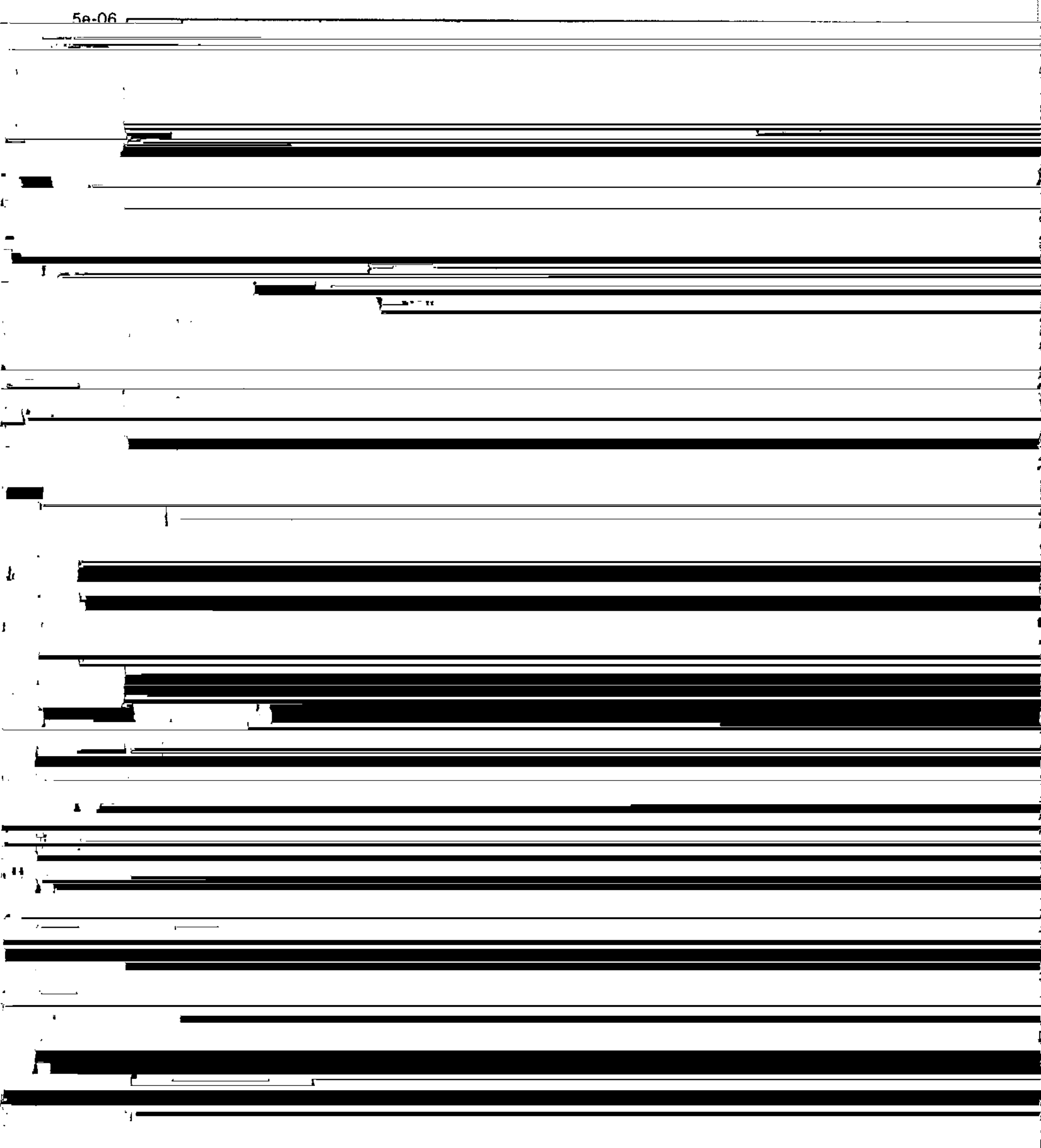
and construct a preconditioner on this subspace.

We obtain

$$\mathcal{P}_{il} = \frac{\delta_{il}}{[illegible]} \tag{4.2}$$

Const	κ	κ_p
$7.1 \cdot 10^{-0}$	$2.4 \cdot 10^{0}$	2.1
$7.1 \cdot 10^{-1}$	$1.5 \cdot 10^{1}$	6.3
$7.1 \cdot 10^{-2}$	$1.4 \cdot 10^{2}$	9.4
$7.1 \cdot 10^{-3}$	$1.3 \cdot 10^{3}$	9.5


16
14
12
10
8
6
4
2
0
-2
-4
"ans_1_iter"
"result"


5e-06

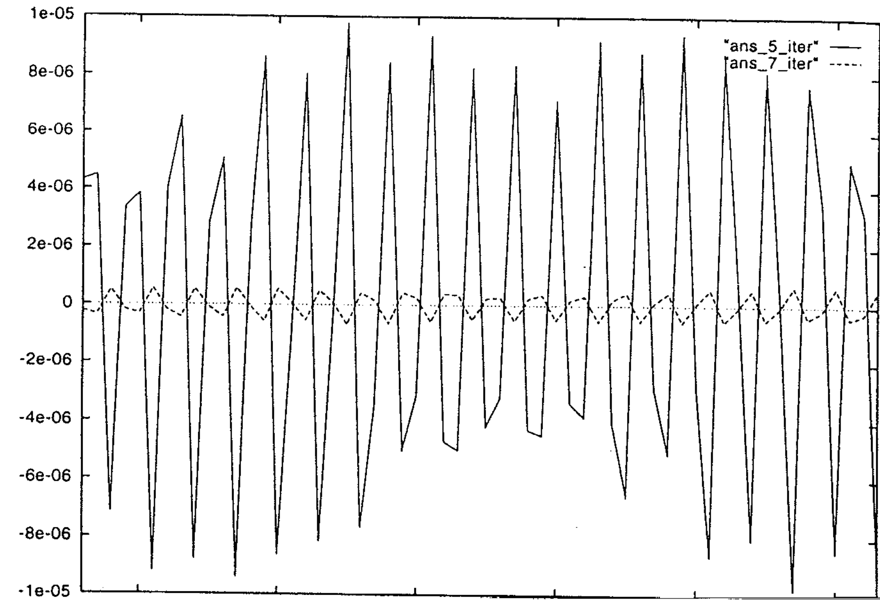


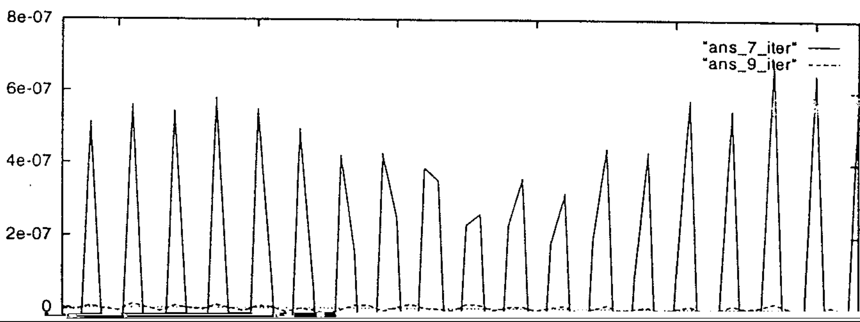
Figure 4. Compare the results after 5 and 7 iterations for $\omega = 0.8$, 1024 points, Daubechies 20, no skip.

[redacted]

[redacted]

[redacted]

[redacted]



[redacted]

[redacted]

# scales to skip	# iterations	$\| \ \|_\infty$	$\| \ \|_2$
0	36	10^{-13}	10^{-14}
1	36	0.45×10^{-9}	0.32×10^{-9}
2	34	0.96×10^{-8}	0.52×10^{-8}
3	32	0.17×10^{-6}	0.83×10^{-7}
4	27	0.30×10^{-5}	0.13×10^{-5}
5	17	0.48×10^{-4}	0.21×10^{-4}
6	9	0.80×10^{-3}	0.34×10^{-3}

[illegible]

§6 Further numerical experiments

In the second set of one-dimensional experiments we verify that using the "constrained CG" method we can maintain the sparsity of the conjugate [illegible]

# scales			

Acknowledgment. The research of A. Averbuch, R. Coifman, and M. Israeli was supported by U.S-Israel Binational Science Foundation Grant # 92-00269/1. The research of M. Israeli was supported also by the Fund for the Promotion of Research at the Technion. The research of G. Beylkin was partially supported by ARPA grant F49620-93-1-0474 and ONR grant N00014-91-J4037.

References

[1] Averbuch, A., G. Beylkin, R. Coifman, and M. Israeli, Fast adaptive algorithm for elliptic equations with periodic boundary conditions, to appear.

[2] Averbuch, A., M. Israeli, and L. Vozovoi, Spectral multidomain technique with local Fourier basis, *J. Scientific Computing* **8**(2) (1993), 135–149.

[3] Aharoni, G., A. Averbuch, R. Coifman, and M. Israeli, Local cosine [illegible]

Amir Averbuch
School of Mathematical Sciences
Tel Aviv University
Tel Aviv 69978, Israel
amir@math.tau.ac.il

Gregory Beylkin
Program in Applied Mathematics
University of Colorado at Boulder
Boulder, CO 80309-0526
beylkin@julia.colorado.edu

Ronald Coifman
Department of Mathematics
P.O. Box 208283
Yale University
New Haven, CT 06520-8283
coifman@jules.math.yale.edu

Moshe Israeli
Faculty of Computer Science
Technion-Israel Institute of Technology
Haifa 32000, Israel
israeli@cs.technion.ac.il