Proceedings of ALGORITMY 2005
pp. 222–229

A FAST SOLVER FOR ELLIPTIC EQUATIONS WITH HARMONIC COEFFICIENT APPROXIMATIONS*

ELENA BRAVERMAN†, MOSHE ISRAELI‡, AND ALEXANDER SHERMAN§

Abstract. Based on a fast subtractional spectral algorithm for the solution of the Poisson equation, we develop a solver for elliptic equations with coefficient which are squares of harmonic functions. A transformation suggested by P. Concus and G. H. Golub in SIAM J. Numer. Anal., Vol. 10 (Dec. 1973),1103-1120, results in a constant coefficient equation. If the square root of the coefficient is not harmonic, we approximate it by a harmonic function. Several correction steps are then applied to achieve the required accuracy. The procedure is efficient if the harmonic function provides a good approximation for the coefficient function. As the quality of the approximation depends inversely on the size of the domain, a hierarchical domain decomposition procedure is suggested which improves the accuracy of the approximation.

Key words. fast spectral direct solver, the Poisson equation, nonseparable elliptic equations, correction steps

AMS subject classifications. 65N35, 65T20

1. Introduction. Variable coefficient elliptic equations are ubiquitous in many branches of scientific and engineering applications, the most important case being that of the self-adjoint operator appearing in diffusion problems in nonuniform media. Such problems also arise in the process of solution of variable coefficient and nonlinear time dependent problems by implicit marching methods. One of traditional methods to solve elliptic equations with nonconstant coefficients was to apply some iterative procedure. A solver for the Poisson equation was usually employed as a preconditioner. Thus a fast and accurate solver for a problem with constant coefficients became an essential part of the numerical method.

Application of high-order (pseudo) spectral methods, which are based on global expansions into orthogonal polynomials (Chebyshev or Legendre polynomials), to the solution of elliptic equations, results in full (dense) matrix problems. The cost of inverting a full $N \times N$ matrix without using special properties is $O(N^3)$ operations [2]. The Fourier method for the solution of the Poisson equation in principle has an exponential convergence but faces the Gibbs phenomenon for non-periodic boundary conditions. Among recent development in the resolution of the Gibbs phenomenon let us mention papers [4, 5, 8] and references therein and [1, 7] for recently developed fast solvers based on the Fourier method and Fast Fourier Transform.

The methods to resolve the Gibbs phenomenon are described in [4] (see also references to this review article). They can be classified as Fourier space filters and methods concerned with an adjustment in a physical space. For the solution of the Poisson equation with Fourier series we have to restore a solution rather than the

*The first author was supported by the University Research Grant of the University of Calgary; the second author was supported by the VPR fund for promotion of research at the Technion.

†Dept. of Mathematics and Statistics, University of Calgary, 2500 University Drive N.W., Calgary, Alberta T2N 1N4, Canada (maelena@math.ucalgary.ca).

‡Technion-Israel Institute of Technology, Computer Science Dept.,Haifa 32000, Israel (israeli@cs.technion.ac.il).

§Technion-Israel Institute of Technology, Applied Mathematics Dept., Haifa 32000, Israel (asherman@techunix.technion.ac.il).

original right hand side (RHS) which is presented in the Fourier space. Since the accuracy of the solution degrades due to the Gibbs phenomenon in the RHS representations, then the algorithm can benefit if the RHS is presented as a sum of a smooth periodic function and another function which can be integrated analytically. Rather simple for 1D problems, the implementation of this idea becomes more complicated for higher dimensions. This procedure is called sometimes the subtraction technique (a function which is later integrated analytically is subtracted from the RHS).

To the best of our knowledge the application of the subtraction technique in the resolution of the Gibbs phenomenon for the Fourier series solution goes back to Sköllermo [9], where a modification of the Fourier method was developed for the Poisson equation

$$\Delta u = f \tag{1}$$

in the rectangle $[0,1]\times[0,1]$ with periodic boundary conditions. It is to be noted that the subtraction algorithm in [9] was developed for some specific boundary conditions only.

We apply the Poisson solver [1] in a rectangular domain with an equispaced grid. Then the subtraction technique (in the physical space) should be used for the resolution of the Gibbs phenomenon rather than other methods due to the following reasons.

a) After subtraction, Fast Fourier Transform can be applied to the remaining part of RHS with high convergence.
b) The algorithm keeps the diagonal representation of the Laplace operator, so, unlike Chebyshev and Legendre expansions, it is not necessary to find an inverse of a full matrix.
c) Generally, the computation of the subtraction functions is even less time consuming than FFT implementation.

The solver developed in [1] is fast ($O(N^2 \log N)$, where N is a number of points in each direction). It is also applied to solve an elliptic equation with nonconstant coefficients

$$\mathcal{L}u = \nabla \cdot a(x,y)\ (\nabla u(x,y)) = f \tag{2}$$

with the preconditioned iterations

$$\mathcal{L}_0 u^{n+1} = f - (\mathcal{L} - \mathcal{L}_0)u^n, \tag{3}$$

where $\mathcal{L}_0 u = f$ is the Poisson equation with a constant coefficient which is equal to the average of the maximal and the minimal values of function $a(x,y)$ in the domain. However for convergence to the accuracy provided by our basic solver sometimes a significant number of iterations (above 30) is necessary. In the framework of the present paper:

1. We develop a fast direct algorithm for the solution of Eq.(2) for any function $a(x,y)$, such that $\sqrt{a(x,y)}$ is harmonic. It is based on the fast direct solver developed in [1] and a transformation described in [3]. This already involves a wide class of equations with nonconstant coefficients.
2. If $\sqrt{a(x,y)}$ is not harmonic, we approximate it by a harmonic function. The numerical scheme incorporates the basic algorithm with some correction steps are required (the procedure is described in Section 2 and tested numerically in Section 3).
3. An adaptive domain decomposition approach is suggested in order to improve the approximation for any function $a(x,y)$.

2. Outline of the algorithm. We solve an elliptic equation

$$\nabla \cdot (a(x,y)\ \nabla u(x,y)) = f(x,y), \quad (x,y) \in \mathbf{D}, \tag{4}$$

where $\mathbf{D}$ is a rectangular subdomain, with the Dirichlet boundary conditions

$$u(x,y) = g(x,y), \quad (x,y) \in \partial\mathbf{D}. \tag{5}$$

We assume $a(x,y) > 0$ for any $(x,y) \in \mathbf{D}$.

Following [3] we make the change of variable

$$w(x,y) = \sqrt{a(x,y)}u(x,y), \tag{6}$$

then Eq. (4) takes the form

$$\Delta w - p(x,y)w = q(x,y), \tag{7}$$

where

$$p(x,y) = \frac{\Delta(\sqrt{a(x,y)}\ )}{\sqrt{a(x,y)}}, q(x,y) = \frac{f(x,y)}{\sqrt{a(x,y)}}. \tag{8}$$

In case $\sqrt{a(x,y)}$ is a harmonic function, Eq.(7) becomes the Poisson equation in w:

$$\Delta w = q(x,y) \tag{9}$$

This leads to the fast direct algorithm for the numerical solution of Eq.(4), where $\sqrt{a(x,y)}$ is a harmonic function.

Algorithm A

1. Using the modified spectral subtractional algorithm which was described in the introduction, we solve Eq.(9) with the boundary conditions

$$\tilde{g}(x,y) = \sqrt{a(x,y)}g(x,y).$$

2. The solution of Eq.(4) is $u(x,y) = \dfrac{w(x,y)}{\sqrt{a(x,y)}}$.

This algorithm enables the solution of

$$\nabla \cdot (\widetilde{a}(x,y)\ \nabla u) = f(x,y), \quad \text{where} \quad \Delta\left(\sqrt{\widetilde{a}(x,y)}\right) = 0 \ \text{ for any } \ (x,y) \in \mathbf{D}, \tag{10}$$

as a constant coefficient problem with the boundary conditions (5).

Let us now consider the case when $\sqrt{a(x,y)}$ is not exactly harmonic but can be well approximated by a harmonic function $\sqrt{\widetilde{a}(x,y)}$. This means that the difference

$$\varepsilon(x,y) = a(x,y) - \widetilde{a}(x,y) \tag{11}$$

is small. Denote by $u_0(x,y)$ the solution of (10) with boundary conditions (5) and introduce $\widetilde{u}(x,y) = u(x,y) - u_0(x,y)$, where $u(x,y)$ is an exact solution of Eq.(4). Then (4) can be rewritten as

$$\nabla \cdot [(\widetilde{a}(x,y) + \varepsilon(x,y))\ \nabla(u_0 + \widetilde{u})] = f(x,y). \tag{12}$$

Taking into account (10), we obtain

$$\nabla \cdot (\widetilde{a}(x,y)\ \nabla \widetilde{u}) = -\nabla \cdot [\varepsilon(x,y)\ \nabla(u_0 + \widetilde{u})]\,, \tag{13}$$

where $\widetilde{u}(x,y)$ satisfies the zero boundary conditions as the difference of two functions $u(x,y)$ and $u_0(x,y)$, which both satisfy (5). Since $\widetilde{u}$ is unknown, the following correction procedure is suggested:

$$\nabla \cdot (\widetilde{a}(x,y)\ \nabla u_1) = -\nabla \cdot [\varepsilon(x,y)\ \nabla u_0] \tag{14}$$

$$\nabla \cdot (\widetilde{a}(x,y)\ \nabla u_{n+1}) = -\nabla \cdot [\varepsilon(x,y)\ \nabla(u_0 + u_n)]\,, \quad n \geq 1. \tag{15}$$

Subtracting (13) from (15) we have

$$\nabla \cdot [\widetilde{a}(x,y)\ \nabla(u_{n+1} - \widetilde{u})] = -\nabla \cdot [\varepsilon(x,y)\ \nabla(u_n - \widetilde{u})] \tag{16}$$

For example, if $\|\varepsilon\| \leq s\|a\|$ in some norm, where s is small, then $\|u_{n+1} - \widetilde{u}\| \leq s\|u_n - \widetilde{u}\|$. The corrected solution u^n after n correction steps is $u^n = u_0 + u_n$. Since the exact solution is $u = u_0 + \widetilde{u}$, then the error decreases according to:

$$\|u^{n+1} - u\| \leq s\|u^n - u\| \tag{17}$$

Thus the algorithm for the solution of (4) can be described as follows.

Algorithm B

1. The coefficient $a(x,y)$ in (4) is approximated by $\widetilde{a}(x,y)$ such that $\sqrt{\widetilde{a}(x,y)}$ is a harmonic function in the domain **D**. Equation (10) is solved using Algorithm A.

2. Some correction steps are made using (15) until the desired accuracy is attained.

Here we leave aside the problem how $\sqrt{a(x,y)}$ is best approximated by a harmonic function. The simplest approach considers a function $b(x,y)$ in the square $[0,1]\times[0,1]$ it can be approximated by the bilinear function

$$\widetilde{b}(x,y) = c_{11} + c_{12}x + c_{21}y + c_{22}xy, \tag{18}$$

which takes on the corner values of $b(x,y)$ i.e.:

$$\begin{aligned} c_{11} &= b(0,0), \qquad c_{12} = b(1,0) - b(0,0), \\ c_{22} &= b(0,1) - b(0,0), c_{22} = b(1,1) + b(0,0) - b(1,0) - b(0,1). \end{aligned} \tag{19}$$

This approximation can be improved by matching more points on the boundary of the square $[0,1] \times [0,1]$ by the addition of functions of the type

$$\varphi(x,y) = d_k \sin(\pi k x)\sinh(\pi k y) \tag{20}$$

which do not influence corner points. Dividing the domain to smaller squares improves the error according to the square of the size of the subdomain.

We could try to find the best approximation of $\sqrt{a(x,y)}$ by least squares in a class of harmonic functions. However the simplest approximation with bilinear functions (18),(19) has the following advantage: if we use domain decomposition(see Section 4), then the collection of $\widetilde{a}(x,y)$ approximated in subdomains is not smooth but is a continuous function, this simplifies very much the inter domain matching process. This is true also for (20). All other approximations involving interior values do not enjoy this beneficial property.

TABLE 1
MAX, MSQ and $\mathcal{L}^2$ *errors for the exact solution* $u(x,y) = (x+1)^2 + (y+0.5)^2$ *and* $a(x,y) = (x+1)^2(y+0.5)^2$ *in the domain* $[0,1] \times [0,1]$.

$N_x \times N_y$	ε_{MAX}	ε_{MSQ}	$\varepsilon_{\mathcal{L}^2}$
8×8	3.2e-5	2.0e-5	5.3e-6
16×16	2.0e-6	1.3e-6	2.7e-8
32×32	1.3e-7	2.7e-8	2.4e-8
64×64	7.9e-9	5.6e-9	1.6e-9
128×128	5.0e-10	3.5e-10	9.8e-11

TABLE 2
MAX, MSQ and $\mathcal{L}^2$ *errors for the exact solution* $u(x,y) = (x+1)^2 + (y+0.5)^2$ *and* $a(x,y) = (x+1+0.1\sin x)^2(y+0.5)^2$ *in the domain* $[0,1] \times [0,1]$.

$N_x \times N_y$	ε_{MAX}	ε_{MSQ}	$\varepsilon_{\mathcal{L}^2}$
8×8	3.4e-5	2.0e-5	5.5e-6
16×16	2.2e-6	1.4e-6	3.8e-8
32×32	1.4e-7	8.9e-8	2.5e-8
64×64	8.6e-9	5.7e-9	1.6e-9
128×128	4.7e-10	3.6e-10	1.0e-10

3. Numerical results. First let us demonstrate the rate of convergence of the algorithm with the growth of the number of grid points in the case where the coefficient $a(x,y)$ is a square of a harmonic function.

Assume that u is the exact solution of Eq.(4) and u' is the computed solution. We will use the following measures to estimate the errors:

$$\begin{aligned}
\varepsilon_{MAX} &= \max |u'_i - u_i| \\
\varepsilon_{MSQ} &= \sqrt{\frac{\sum_{i=1}^{N}(u'_i - u_i)^2}{N}} \\
\varepsilon_{\mathcal{L}^2} &= \sqrt{\frac{\sum_{i=1}^{N}(u'_i - u_i)^2}{\sum_{i=1}^{N} u_i^2}}
\end{aligned}$$

Example 1. Consider the equation with $a(x,y) = (x+1)^2(y+0.5)^2$, the right hand side and the boundary conditions correspond to the exact solution $u(x,y) = (x+1)^2 + (y+0.5)^2$. The results are presented in Table 1.

Now let us proceed to an example, where $\sqrt{a}$ is not harmonic.

Example 2. Consider the equation with $a(x,y) = (x+1+r\sin x)^2(y+0.5)^2$, the right hand side and the boundary conditions correspond to the same exact solution as in Example 2. Here we need to apply some correction steps in order to get desired accuracy. We used (18),(19) for the approximation of $\sqrt{a}$ by a harmonic function. The results for $\alpha = 0.1$ are presented in Table 2.

If we insist to get the same (excessive) accuracy as in the previous example, it is necessary to apply from 2 correction steps for 32×32 points to 6 steps for 128×128 points. It is expected that with the growth of r more corrections steps are required. Let us consider $r = 0.5$. Fig. 1 describes the convergence of the maximal error for $r = 0.5$ with the growth of the number of correction steps. Here 32×32 grid points were used.

The number of correction steps which is necessary to achieve the prescribed accuracy grows as the difference between the harmonic approximation of $\sqrt{a}$ and its real

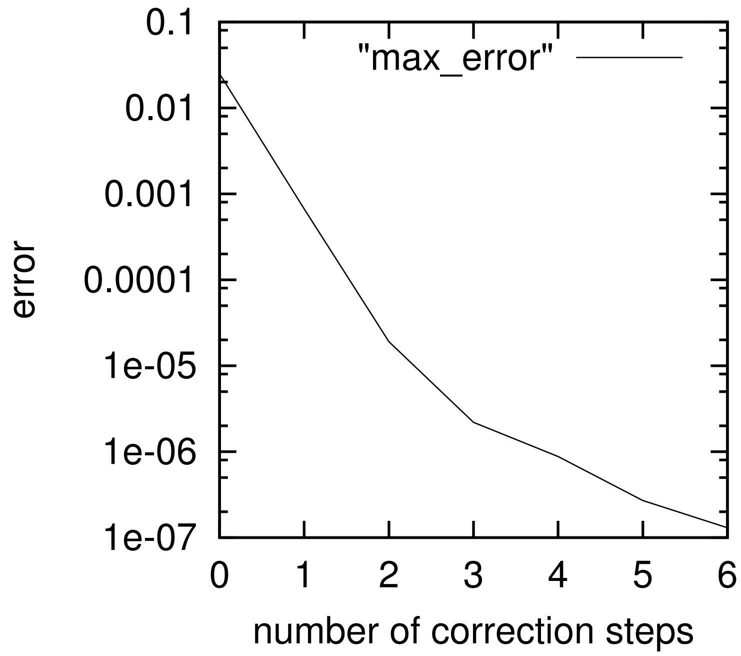

FIG. 1. *The maximal error for various number of correction steps for the coefficient* $a(x, y) = (x + 1 + 0.5 \sin x)^2(y + 0.5)$ *and* 32×32 *grid points*

value increases. Fig. 2 presents the dependency of the number of correction steps which are necessary to get the accuracy of 3e-7, on the parameter r in the coefficient $a(x, y) = (x + 1 + r \sin x)^2(y + 0.5)$.

4. Summary and discussion. The present algorithm incorporates the following novel elements:

1. It extends our previous fast Poisson solvers [1, 7] as it provides an essentially direct solution for equations (4) where $\sqrt{a(x, y)}$ is an arbitrary harmonic function, in particular, a bilinear function

$$\sqrt{a} = c_{11} + c_{12}x + c_{21}y + c_{22}xy.$$

2. In the case where $\sqrt{a(x, y)}$ is not harmonic, we approximate it by $\sqrt{\widetilde{a}(x, y)}$ and apply some correction steps to improve the accuracy. If $\widetilde{a}(x, y)$ is chosen to be a constant we obtain an iteration procedure with a constant coefficient solver as a preconditioner (for details, see [1]).

However high accuracy for the solution of (4) requires an accurate approximation of $\sqrt{a}$ by a harmonic function. Such an approximation is not always easy to derive in the global domain, however it can be achieved in smaller subdomains. In this case we suggest the following Domain Decomposition algorithm.

1. The domain is decomposed into smaller rectangular subdomains. Where the boundary of the subdomains coincides with full domain boundary we take on the original boundary conditions. For other interfaces we introduce some initial boundary conditions which do not contradict the equation at the corners, where the left hand side of (4) can be computed. The function a is approx-

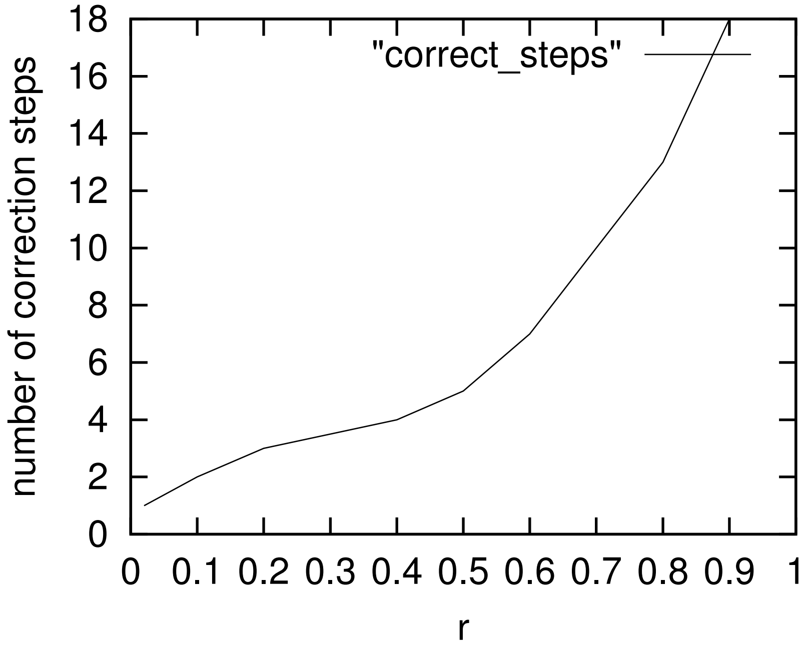

FIG. 2. *The number of correction steps necessary to attain the accuracy of 3e-7 for the coefficient* $a(x, y) = (x + 1 + r \sin x)^2 (y + 0.5)$ *and* 32×32 *grid points*

imated by $\tilde{a}$ in each subdomain such that $\sqrt{\tilde{a}}$ is harmonic. An auxiliary equation (10) is solved in each subdomain.

2. The collection of solutions obtained at Step 1 is continuous but has noncontinuous derivatives at domain interfaces. To further match subdomains, a hierarchical procedure can be applied similar to the one described in [6]. For example, if we have four subdomains 1,2,3 and 4, then 1 can be matched with 2, 3 with 4, while at the final step the merged domain 1,2 is matched with 3,4.

REFERENCES

[1] E. Braverman, B. Epstein, Boris, M. Israeli and A. Averbuch, *A fast spectral subtractional solver for elliptic equations*, J. Sci. Comput., 21 (2004), pp. 91–128.
[2] C. Canuto, M.Y. Hussaini, A. Quarteroni, T. A. Zang, *Spectral Methods in Fluid Dynamics*, Springer-Verlag, 1989.
[3] P. Concus and G. H. Golub, *Use of fast direct methods for the efficient numerical solution of nonseparable elliptic equations*, SIAM J. Numer. Anal., 10 (1973), pp. 1103–1120.
[4] D. Gottlieb and C. W. Shu, *On the Gibbs phenomenon and its resolution,* SIAM Review, 39 (1997), pp. 644–668.
[5] M. S. Min and D. Gottlieb, *On the convergence of the Fourier approximation for eigenvalues and eigenfunctions of discontinuous problems*, SIAM J. Numer. Anal., 40 (2002), pp. 2254-2269.
[6] M. Israeli, E. Braverman and A. Averbuch, *A hierarchical domain decomposition method with low communication overhead*, Domain decomposition methods in science and engineering (Lyon, 2000), pp. 395–403, Theory Eng. Appl. Comput. Methods, Internat. Center Numer. Methods Eng. (CIMNE), Barcelona, 2002.
[7] O. F. Næss and K.S. Eckhoff, *A modified Fourier-Galerkin method for the Poisson and*

Helmholtz equations, J. Sci. Comput., 17 (2002), pp. 529–539.
[8] B. D. Shizgal and J. H. Jung, *Towards the resolution of the Gibbs phenomena*, J. Comput. Appl. Math., 161 (2003), pp. 41–65.
[9] G. Skölermo, *A Fourier method for numerical solution of Poisson's equation*, Math. Comput. (U.S.A.) 29 (1975), pp. 697–711.