On a Fast Direct Elliptic Solver by a Modified Fourier Method

A. Averbuch, M. Israeli, L. Vozovoi

appeared in Numerical Algorithms, Vol. 15 (3,4) pp. 287-313, 1997.

Abstract

A fast elliptic solver of high order accuracy is constructed for the solution of the Poisson and the Helmholtz equations in rectangular domains. The algorithm is based on the Fourier method in combination with a subtraction procedure. The singularities at the corner points, arising due to non-smoothness of the boundaries, are treated explicitly using properly constructed singular corner functions. The present algorithm is a generalization of the Fast Poisson Solver developed in our previous paper [12].

Key words: Helmholtz equation, Fourier method, fast solver, corner singularities.

1 Introduction

This paper presents further developments of a fast direct Poisson solver constructed previously in [12]. The present approach is applied, with only slight modifications, to a wider class of second order elliptic problems in rectangular regions: Poisson, the oscillatory (indefinite) Helmholtz (*OH*) and the monotonic Helmholtz (*MH*) equations. The *MH* equation frequently appears in Computational Fluid Dynamics as a result of time discretization of the Navier-Stokes equations.

Our approach is based on the Fourier method in combination with the subtraction technique in order to accelerate the convergence of the Fourier series. It is known that the finite Fourier series, associated with a non-periodic function, converges very slowly, like $O(N^{-1})$, where N is the number of Fourier harmonics (or collocation points in the pseudo-spectral formulation). Therefore, the accuracy of the Fourier method for elliptic problems with non-periodic boundary functions is not better than for a first order finite-difference scheme.

However, the accuracy of the Fourier method can be improved considerably if an approximated function, f, is smoothed. This can be achieved by subtracting a known function, F, which has the same values on the boundaries. The resultant function, $g = f - F$, will be of class C^0 so that the associated Fourier series converges like $O(N^{-2})$. The rate of convergence can be further improved if several first derivatives of the subtraction function F are equal to corresponding derivatives of f on the boundaries.

In two dimensions an auxiliary subtraction function has to be constructed such that it matches (along with its several derivatives) the boundary function at the corners of a computational domain. Another requirement for this function is that it satisfy the homogeneous elliptic equation under consideration. Otherwise, the subtraction procedure will introduce a non-periodic source term which is difficult to treat by the Fourier method.

An important feature of the present method is that it treats explicitly the singular points in order to preserve fast convergence. Typically, solutions to elliptic problems in rectangular regions have singularities at the corner points, which may appear even if the source function is smooth inside the region and the boundary functions are continuous and continuously differentiable on each side of the rectangle. These singularities reflect the non-smoothness of the boundary.

The presence of singularities in the solution tends to significantly affect the rates of the convergence for any numerical algorithm, especially for high order methods. A commonly used approach for accurate treatment of singularities, associated with elliptic boundary value problems, consists of grid refinement with an appropriate adaptive strategy [1].

When using finite element approximations one can improve convergence by including so called "singular elements" in the basis functions [3]. These singular elements are designed to match the leading terms in the local power series expansions of solutions near the point of singularity.

In [4] the Helmholtz equation is solved in a region with smooth boundary using the boundary integral formulation. The integral equation is solved by the Fourier spectral method. Since the integral kernel for the Helmholtz equation has a logarithmic singularity, the corresponding Fourier series converges very slowly. A proper treatment of singularity in the integral kernels yields accurate solutions with significantly fewer datum points.

In our previous work, [12] we proposed to remove corner singularities using properly constructed singular subtraction functions. These functions necessarily satisfy a homogeneous boundary value problem. Such a technique is consistent with the basic approach, which makes use of subtraction functions in order to remove jumps in the periodic extension of the boundary functions. Singular corner functions resemble, in a sense, singular elements in the finite element method. In the case of the Poisson equation, these functions are constructed analytically in [12].

The construction of a singular homogeneous solution to the Helmholtz equation is a more difficult task. In this paper we propose a numerical algorithm to compute singular subtraction functions for both the oscillatory and the monotonic Helmholtz equations. The same algorithm can be employed for the Poisson equation as well. Incorporating these functions in the subtraction procedure enables the algorithm to preserve high order accuracy even in the presence of corner singularities.

The paper is organized as follows. In section 2, we describe a numerical algorithm in the case of periodic boundary functions. In section 3, we extend this technique to treat elliptic problems with not periodic boundary distributions. In section 4, we construct subtraction functions in a regular (not singular) case. Singular solutions are considered in section 5. Finally, in section 6 we extend the previous technique for the solution of non-homogeneous elliptic problems.

2 Homogeneous Problems with Periodic Boundary Functions

We are interested in the solution of the two-dimensional monotonic Helmholtz (MH) equation

$$\Delta u - \lambda^2 u = 0, \qquad \text{in } \Omega \tag{2.1a}$$

and the oscillatory Helmholtz (OH) equation

$$\Delta u + \kappa^2 u = 0, \qquad \text{in } \Omega \tag{2.1b}$$

in the rectangular domain $\Omega = [0,1]^2$. Here λ and κ are real numbers. In a particular case, $\kappa = 0$, we have the Laplace equation

$$\Delta u = 0, \qquad \text{in } \Omega. \tag{2.1c}$$

The boundary conditions of Dirichlet or Neumann types are imposed:

$$u = \Phi(x, y) \qquad \text{on} \quad \partial\Omega \tag{2.2a}$$

or

$$\frac{\partial u}{\partial n} = \Phi(x, y) \qquad \text{on} \quad \partial\Omega \tag{2.2b}$$

where n is the internal normal to $\partial\Omega$. The mixed Dirichlet/Neumann type boundary conditions can be considered as well. The boundary functions

$$\phi_1(x) = \Phi(x, 0), \qquad \phi_3(x) = \Phi(x, 1),$$

$$\phi_2(y) = \Phi(0, y), \qquad \phi_4(x) = \Phi(1, y) \tag{2.3}$$

are assumed to be smooth on each side. However, they may have jumps at the corner points or may be incompatible with the differential equation at these points.

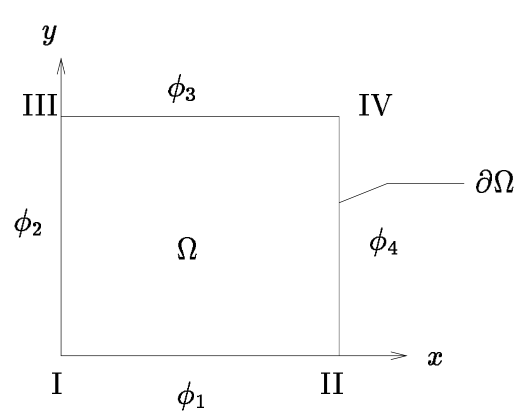

Figure 1: Computational region and boundary distribution.

The basis for the suggested approach is the use of rapidly convergent series expansions

$$u(x,y) = \sum_{n=0}^{M} c_n h_n(x,y) \tag{2.4}$$

where $h_n(x,y)$ satisfy the same homogeneous equation (2.1) as the solution $u(x,y)$. The coefficients, c_n, have to be found such that u satisfies the boundary conditions (2.2).

Assume for a moment that the boundary functions ϕ_i have smooth (C^∞- continuous) periodic extensions beyond the corner points. (Note that this setting is different from the case of periodic boundary conditions, which implies $u(x+1,y) = u(x,y)$, $u(x,y+1) = u(y)$ without specification of the boundary functions). Also assume that the values at the corner points are zero. In such a case one can seek the expansion (2.4) in the form of trigonometric/hyperbolic series. For example, in the case of Dirichlet boundary conditions (2.2a), the basis functions h_n can be constructed in the form:

MH equation:

$$h_n^I(x,y) = \sin\ \lambda_n x\ \frac{\sinh\ \mu_n y}{\sinh\ \mu_n}, \qquad \lambda_n = 2\pi n,\ \ \mu_n^2 = \lambda_n^2 + \lambda^2, \tag{2.5a}$$

OH equation:

$$h_n^{II}(x,y) = \sin\ \lambda_n x\ \frac{\sin\ \mu_n y}{\sin\ \mu_n}, \qquad \lambda_n = 2\pi n,\ \ \mu_n^2 = \kappa^2 - \lambda_n^2 \quad (\kappa - \lambda_n > 0) \tag{2.5b}$$

or

$$h_n^I(x,y) = \sin\ \lambda_n x\ \frac{\sinh\ \mu_n y}{\sinh\ \mu_n}, \qquad \lambda_n = 2\pi n,\ \ \mu_n^2 = \lambda_n^2 - \kappa^2 \quad (\kappa - \lambda_n < 0). \tag{2.5c}$$

For the Laplace equation one can use the basis functions (2.5c) with $\mu_n^2 = \lambda_n^2$.

For the *MH* equation the values λ may be chosen arbitrarily, while in the case of the *OH* equation we have to exclude from consideration the resonance values $\kappa = \kappa_*$ for which $\sin \mu_n = 0$. Using the relations in (2.5b), one can rewrite non-resonance conditions as

$$\kappa \neq \kappa_*, \qquad \kappa_*^2 = \pi^2(4n^2 + k^2), \qquad n,\ k \in \mathrm{Z}. \tag{2.6}$$

For $\kappa = \kappa_*$, the boundary value problem (2.1b), (2.5b) has an infinite number of solutions that vanish on the boundaries. Notice that the condition (2.6) is true for the box of unit size, but it can be easily rescaled to reflect a box of arbitrary size.

In the case of the Neumann boundary conditions, (2.2b), we assume that the normal derivative vanishes at the corner points and seek the solution in the form of (2.4) with the basis functions

$$h_n(x,y) = \cos\ \lambda_n x\ \frac{\cosh\ \mu_n y}{\mu_n\ \sinh\ \mu_n} \tag{2.7}$$

where $\lambda_n = 2\pi n$, and $\mu_n^2 = \lambda_n^2 - \lambda^2$ for the *MH* equation or $\mu_n^2 = \kappa^2 + \lambda_n^2$ for the *OH* equation.

It is convenient to split the solution into four sums as follows:

$$u(x,y)=\sum_{n=1}^{N}c_n^{(1)}h_n(x,1-y)+\sum_{n=1}^{N}c_n^{(2)}h_n(y,1-x)+\sum_{n=1}^{N}c_n^{(3)}h_n(x,y)+\sum_{n=1}^{N}c_n^{(4)}h_n(y,x). \quad (2.8)$$

Each sum in (2.8) corresponds to a part of the solution which satisfies the prescribed boundary conditions on one side of the rectangle in Fig.1 and vanishes (or has vanishing normal derivatives) on the other three sides. Thus, the first sum represents a function which matches a given distribution on the boundary $y=0$, the second- on the boundary $x=0$, and so on.

Technically, we expand a given boundary function, ϕ_i, of one variable into trigonometric series. These series converge exponentially fast due to the previous assumption on smoothness of ϕ_i. Then we apply the Fourier transform to compute the expansion coefficients $c_n^{(i)}$. After that, we multiply $c_n^{(i)}$ by a corresponding "complementary" hyperbolic or trigonometric function of the second variable to extend the solution "from" this boundary inside the domain. This procedure is repeated for each boundary.

To summarize, the numerical algorithm, in the case of periodic boundary functions, consists of the following steps.

Algorithm I (**the case of periodic boundary functions**)

1. **Discretization.** Approximate Ω by the uniform grid $\Omega_g=\{(x_i,y_j)|\ x_i=(i-1)\Delta\ ,\ y_j=(j-1)\Delta,\ 1\le i,j\le N+1\}$, where $\Delta=1/N$ is the grid spacing with N a given power of two.

2. **Construction of the first sum in** (2.8):

   2.1 Apply the Discrete Sine Transform (DST) to the function $\phi_1(x_i)$ on the boundary $y=0$ to obtain the expansion coefficients $c_n^{(1)},\ \ n=1,...,N$.

   2.2 For each $y_j,\ j=1,...N$ do:

   - multiply $c_n^{(1)}$ by the corresponding complementary (hyperbolic or trigonometric) function of y according to (2.5). For example, in the case of the *MH* equation and Dirichlet b.c. we have

   $$\tilde{c}_n^{(1)}=c_n^{(1)}\cdot\frac{\sinh\ \mu_n y_j}{\sinh\ \mu_n},\qquad \mu_n^2=(2\pi n)^2+\lambda^2.$$

   - transform $\tilde{c}_n^{(1)}$ back to the physical space by applying the inverse DST.

   The resulting solution satisfies the prescribed boundary conditions at $y=0$ and vanishes, according to the construction, on the other three boundaries.

3. **Construction of the second, third and fourth sums in** (2.8).
   Perform the operations of step 2 successively for the other three boundaries.

□

Notice that each sum in (2.8) is constructed *independently* using the expansion coefficients of the boundary function on the corresponding side.

Step 2.1 requires $5N log_2 N$ operations. Step 2.2 requires $O(N^2)$ multiplications and N times application of the DST routine of the size N, a total of $5N^2 log_2 N$ arithmetic operations. Therefore, the computational cost of this algorithm amounts to

$$20N^2 log_2 N + O(N^2) \tag{2.9}$$

arithmetic operations.

Similarly, in the case of Neumann b.c. one can use the Discrete Cosine Transform to obtain the expansion coefficients $c_n^{(i)}$, $i = 1, .., 4$ of the boundary functions on each side of the rectangular. As in the case of Dirichlet b.c., the construction is performed independently for each side since the normal derivative vanishe on the other three sides due to the choice of the basis functions.

A similar construction can be used for the case of mixed Dirichlet/Neumann boundary conditions. For example, in the case $u(x,0) = \phi_1(x)$, $\frac{\partial u}{\partial y}(0,y) = \phi_2(y)$, $u(x,1) = \phi_3(x)$, $\frac{\partial u}{\partial y}(1,y) = \phi_4(y)$ and $u = 0$ at the corner points, the solution can be constructed in the form:

$$\begin{aligned} u(x,y) = & \sum_{n=1}^{N} c_n^{(1)} \cos\ \lambda_n x\ \frac{\sinh\ \mu_n(1-y)}{\sinh\ \mu_n} + \sum_{n=1}^{N} c_n^{(2)} \sin\ \lambda_n y\ \frac{\cosh\ \mu_n(1-x)}{\mu_n \sinh\ \mu_n} \\ & + \sum_{n=1}^{N} c_n^{(3)} \cos\ \lambda_n x\ \frac{\sinh\ \mu_n y}{\sinh\ \mu_n} + \sum_{n=1}^{N} c_n^{(4)} \sin\ \lambda_n y\ \frac{\cosh\ \mu_n x}{\mu_n \sinh\ \mu_n}, \end{aligned} \tag{2.10}$$

where the relation between μ_n and λ_n is described by Eq.(2.5). As before, each sum in this expression can be computed independently (using the expansion coefficients $c_n^{(i)}$ of the corresponding boundary function ϕ_i) without affecting distributions on the other sides.

3 Non-periodic boundary functions

In the case of non-periodic boundary distribution, the finite trigonometric series exhibits $O(1)$ spurious oscillations near the boundaries which decay slowly into the region (the Gibbs phenomenon). Therefore, the approximation (2.4), using trigonometric/hyperbolic series (2.5) or (2.7), converges poorly in Ω.

The rate of convergence can be accelerated considerably by removing, from the given function f, a function F such that it has the same discontinuities on the boundaries as f. As a result, the Fourier series of a smooth component, $g = f - F$, converges much faster than that associated with the original discontinuous function f.

In one dimension this procedure can be implemented as follows.

Algorithm II (**smoothing procedure in one dimension**)

1. Project $f(x),\ 0 \le x \le 1$ onto a uniform collocation grid $x_i = (i-1)\Delta,\ 1 \le i \le N+1$.

2. Compute several initial *even* derivatives $f^{(2)}, f^{(4)}, ..., f^{(2r)}$ at the end points $x = 0$ and $x = 1$ (here $f^{(s)}$ denotes the sth derivative of f). The numerical procedure is described in Appendix A.

3. Subtract from $f(x)$ a known function $F(x)$, $g = f - F$, such that

$$\text{at} \quad x = 0, 1 \qquad F = f,\ \frac{d^2F}{dx^2} = f^{(2)},\ \frac{d^4F}{dx^4} = f^{(4)}, ..., \frac{d^{(2r)}F}{dx^{(2r)}} = f^{(2r)}. \tag{3.1}$$

4. Extend g anti-symmetrically over a half-period as follows:

$$\tilde{g}(x) = \begin{cases} g(x) & 0 < x < 1, \\ -g(2-x) & 1 < x < 2 \end{cases} \tag{3.2}$$

□

This algorithm results in a smooth periodic function, $\tilde{g} \in C^{2r+1}$, which can be represented by a rapidly convergent, like $O(N^{-2(r+1)})$, trigonometric series. This approach is known as the method of A.N. Krylov ([6]). A particular version of this technique was implemented in [13] for the solution of stiff differential equations with high accuracy.

In this paper, we propose a two-dimensional generalization of the above technique. The basic computational step consists of constructing the subtraction function $U(x, y)$ such that it itself satisfies the homogeneous equation (2.1). Therefore, Algorithm II can be applied efficiently and with high accuracy to compute the smooth part of the solution, $v = u - U$. After that the original solution u can be easily reconstructed by adding the known subtraction function U. The following Algorithms III, IV constitute the Modified Fourier (MF) method.

Algorithm III (**smoothing procedure in two dimensions**)

1. Compute several initial even derivatives of the boundary functions ϕ_i at the corner points (for example, using the procedure in Appendix A):

$$\begin{aligned} &\text{at} \quad x = 0,\ 1 \qquad \phi_1^{(2)}, \phi_1^{(4)}, ..., \phi_1^{(2r)}; \qquad \phi_3^{(2)}, \phi_3^{(4)}, ..., \phi_3^{(2r)} \\ &\text{at} \quad y = 0,\ 1 \qquad \phi_2^{(2)}, \phi_2^{(4)}, ..., \phi_2^{(2r)}; \qquad \phi_4^{(2)}, \phi_4^{(4)}, ..., \phi_4^{(2r)}. \end{aligned} \tag{3.3}$$

2. Construct an auxiliary function $U^{(r)}(x, y)$ which satisfies Eq.(2.1) and the following conditions

$$\begin{aligned} \text{at} \quad x = 0,\ y = 0 \qquad U^{(r)} = \phi_1, \quad &\frac{\partial^2 U^{(r)}}{\partial x^2} = \phi_1^{(2)}, \quad ... \quad \frac{\partial^{2r} U^{(r)}(x,0)}{\partial x^{2r}} = \phi_1^{(2r)}, \\ &\frac{\partial^2 U^{(r)}}{\partial y^2} = \phi_2^{(2)}, \quad ... \quad \frac{\partial^{2r} U^{(r)}}{\partial y^{2r}} = \phi_2^{(2r)} \end{aligned} \tag{3.4}$$

and similarly at the other three corners. Thus, the values of the function $U^{(r)}(x, y)$ and its r even derivatives at the corner points match those of the boundary function $\Phi(x, y)$. We denote the boundary distribution $U^{(r)}|_{\partial\Omega} = \Psi^{(r)}(x, y)$.

3. Define a new function

$$v^{(r)} = u - U^{(r)} \tag{3.5}$$

whose boundary conditions are:

$$\phi_1^{(r)}(x) = \Phi(x,0) - \Psi^{(r)}(x,0), \qquad \phi_3^{(r)}(x) = \Phi(x,1) - \Psi^{(r)}(x,1)$$
$$\phi_2^{(r)}(y) = \Phi(0,y) - \Psi^{(r)}(0,y), \qquad \phi_4^{(r)}(x) = \Phi(1,y) - \Psi^{(r)}(1,y). \tag{3.6}$$

4. Extend the boundary functions anti-symmetrically over a half-period as in (3.2): $\phi_i^{(r)} \longrightarrow \tilde{\phi}_i^{(r)}$.

□

Since the functions $\tilde{\phi}_i^{(r)}$ are in C^{2r+1}, they can be represented by rapidly convergent trigonometric series. This fact is utilized by the next algorithm.

Algorithm IV (**solution in the case of non-periodic boundary functions**)

1. Discretize the problem (2.1), (2.2) on a uniform grid $N_x \times N_y$.

2. For a given order r, construct the subtraction function $U^{(r)}(x,y)$ and compute new boundary functions $\tilde{\phi}_k$ using Algorithm III.

3. Solve Eq.(2.1) with the boundary functions $\tilde{\phi}_k$ using Algorithm I to compute the smooth part of the solution, $v^{(r)} = u - U^{(r)}$.

4. Reconstruct the sought solution $u = v^{(r)} + U^{(r)}$.

□

Constructing the subtraction function $U^{(r)}$ at all N^2 interior points requires $O(N^2)$ operations (see section 4). Therefore, the most time consuming step of this algorithm is the computation of the smooth solution $v^{(r)}$, at the cost of $O(N^2 \log N)$ operations, using Algorithm I.

4 Construction of $U^{(r)}$ (not singular case)

In this section we construct the auxiliary subtraction functions $U^{(r)}$ which satisfies the homogeneous equation (2.1) and match a given boundary distribution $\Phi(x,y)$ at the corner points. The construction procedure is basically the same for the Helmholtz equation (of both monotonic and oscillatory types) and the Laplace equation.

We assume that the boundary functions ϕ_i are continuous at the corner points and consistent with Eq.(2.1). For example, in the case of the *MH* equation we assume that

$$x = 0,\ y = 0 \qquad \phi_1 = \phi_2, \qquad \frac{d^2\phi_1}{dx^2} + \frac{d^2\phi_2}{dy^2} - \lambda^2\phi_1 = 0 \tag{3.7}$$

and similarly at the other corners.

Corollary 1. If the subtraction function matches at some corner point the derivatives of the boundary function along one side of the rectangle Ω, it automatically matches its derivatives along the other side sharing the same corner point. $\square$

This follows from the consistency condition (3.7). A more general case when the boundary functions are not consistent with the equation is considered in section 5.

4.1 Monotonic Helmholtz (*MH*) Equation

In the case of the *MH* equation the subtraction function $U^{(r)}$ can be constructed in the form

$$U^{(r)}(x,y) = \sum_{p,q=0}^{1} U^{(r)}_{p,q}(x,y), \qquad U^{(r)}_{p,q}(x,y) = \sum_{n} A^{(r,n)}_{p,q} H^{n}_{p,q}(x,y), \tag{4.1}$$

$$H^{n}_{p,q}(x,y) = \frac{\sinh\ \lambda_n (x_p - x)}{\sinh\ \lambda_n} \frac{\sinh \mu_n (y_q - y)}{\sinh \mu_n}; \qquad \mu_n^2 = \lambda^2 - \lambda_n^2, \tag{4.2}$$

where $x_0 = y_0 = 0$, $x_1 = y_1 = 1$, and λ is the parameter of Eq.(2.1a). The parameter λ_n may be different for each corner, that is $\lambda_n = \lambda_n(p,q)$.

Each function $H^{n}_{p,q}$ takes its maximum at one corner point and vanishes at three other corners, see Fig.4. Moreover, all *even derivatives* of this function vanish at the same points. This allows us to match the function and its even derivatives at each corner point *independently* of the other three corners.

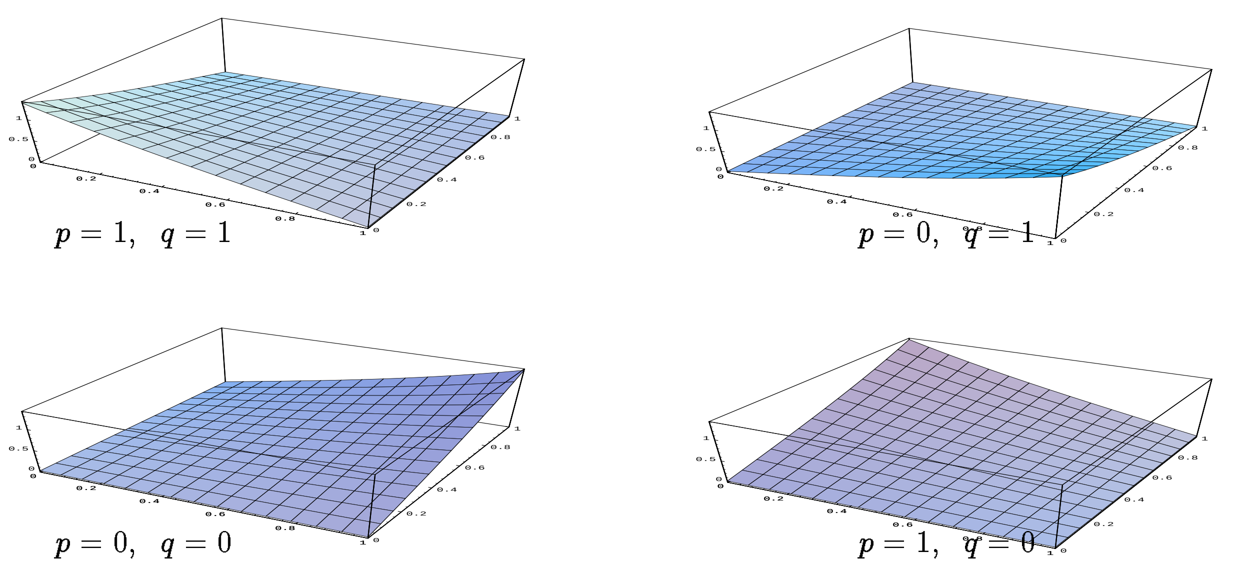

Figure 4: Plot of the "corner" functions $H^{1}_{p,q}$.

In the first step ($r = 0$), one can match a given boundary distribution at the corner points using the four properly weighted "corner functions" $H^{1}_{p,q}$. As a result, the residual function

$v^{(0)} = u - U_s^{(0)}$ vanishes at the corner points. The parameter λ_1 can be chosen arbitrarily at this stage.

The second matching step ($r = 1$) intends to make zero both the function and their second derivatives at the corners. This can be achieved by using the following four corner functions

$$\text{at } x = 1, y = 1, \qquad U_{0,0}^{(1)} = A\,\frac{\sinh\ \lambda_1 x}{\sinh\ \lambda_1}\,\frac{\sinh \mu_1 y}{\sinh\ \mu_1},$$

$$A = \phi_3(1), \ \ \lambda_1 = \frac{d^2\phi_3}{dx^2}(1), \ \ \mu_1^2 = \lambda^2 - \lambda_1^2 \tag{4.3a}$$

and similarly at the other corners.

Suppose that the boundary function is zero at the corner points but its second derivative is discontinuous at the point $(1, 1)$. In such a case an appropriate subtraction function can be constructed as a combination of eight (four pairs) corner functions:

$$\text{at } x = 1, y = 1, \quad U_{0,0}^{(1)} = A\left(\frac{\sinh\ \lambda_1 x}{\sinh\ \lambda_1}\,\frac{\sinh \mu_1 y}{\sinh\ \mu_1} - \frac{\sinh\ \lambda_2 x}{\sinh\ \lambda_2}\,\frac{\sinh \mu_2 y}{\sinh\ \mu_2}\right),$$

$$A = \frac{1}{\lambda_1^2 - \lambda_2^2}\,\frac{d^2\phi_3}{dx^2}(1), \qquad \lambda_2 \neq \lambda_1, \ \ \mu_{1,2}^2 = \lambda^2 - \lambda_{1,2}^2, \tag{4.3b}$$

and similarly for the other corners. We can proceed constructing the higher order corner functions to obtain a smoother distribution on the boundaries. Each subtraction step results in a more rapid convergence of the trigonometric series when computing a smooth part $v^{(r)}$ of the solution.

We illustrate the present Modified Fourier method by showing several examples.

Example 1.

Solve the homogeneous Helmholtz equation (2.1a) with the boundary conditions

$$\Phi(x, 0) = \cos\ \pi x, \qquad \Phi(x, 1) = 0,$$

$$\Phi(0, y) = \frac{\sinh \mu(1 - y)}{\sinh \mu}, \qquad \Phi(1, y) = -\frac{\sinh \mu(1 - y)}{\sinh \mu}, \tag{4.4}$$

where $\mu = \sqrt{\lambda^2 + \pi^2}$. The corresponding exact solution of this problem is

$$u_{ex} = \cos \pi x\,\frac{\sinh \mu(1 - y)}{\sinh \mu}.$$

The results are shown in Figs. 5 and 6 and in Table 1 for $\lambda^2 = 4$ and 100.

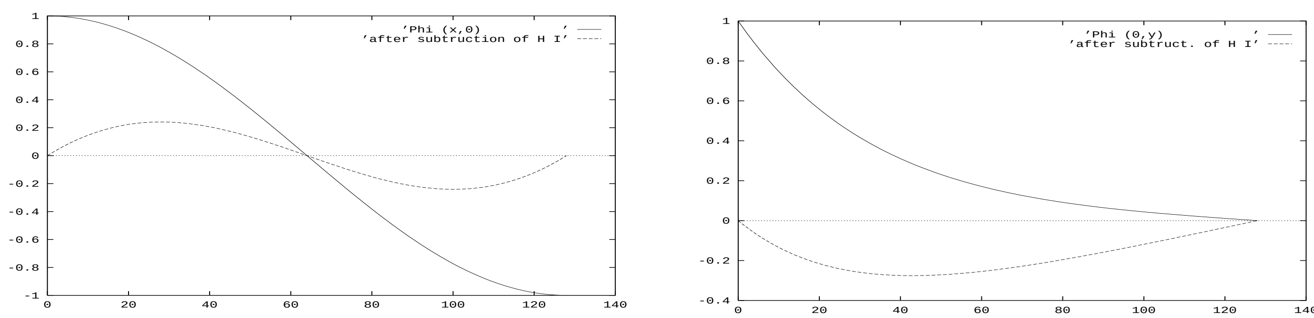

Figure 5: Original (rigid line) and smoothed (dashed line) boundary functions after subtraction of $U^{(1)}$ on horizontal (left) and vertical (right) sides.

		after I subtr. step		after II subtr. step		after III subtr. step	
$N_x \times N_y$	λ^2	ε_{max}	ε_{RMS}	ε_{max}	ε_{RMS}	ε_{max}	ε_{RMS}
32×32	4	3.1 (-6)	2.1 (-7)	2.3 (-8)	1.1 (-9)	3.8 (-12)	2.7 (-13)
	100	3.0 (-6)	2.0 (-7)	2.1 (-8)	1.0 (-9)	3.7 (-12)	2.6 (-13)
64×64	4	7.8 (-7)	2.6 (-8)	1.4 (-9)	3.4 (-11)	1.0 (-13)	1.8 (-14)
	100	7.7 (-7)	2.5 (-8)	1.4 (-9)	3.3 (-11)	1.5 (-13)	1.9 (-14)
128×128	4	1.9 (-7)	3.2 (-9)	8.8 (-11)	1.0 (-12)	1.2 (-13)	1.9 (-14)
	100	1.9 (-7)	3.2 (-9)	8.8 (-11)	1.0 (-12)	2.0 (-13)	2.1 (-14)

Table 1: Maximum and RMS errors in the computed solution (4.4) after one, two and three subtraction steps.

Naturally, the error decays as the number of subtraction steps r (i.e. the the order of smoothness) increases. After three subtraction steps high accuracy is achieved both for small and large λ.

Example 2.

The solution is generated by two point charges of unit intensity

$$u(r) = I_0(r - r_1) + K_0(r - r_2), \qquad r = \sqrt{x^2 + y^2}, \tag{4.5}$$

where $I_0(r)$ and $K_0(r)$ are the modified Bessel functions of zero order (the last one represents a singular charge with logarithmic singularity at $r = 0$). The accuracy is listed in Table 2 for $\lambda^2 = 4, r_1 = (0.2, 0.5), r_2 = (1.2, 0.5)$ and several resolutions $N_x \times N_y$.

$N_x \times N_y$	after I subtr. step		after II subtr. step		after III subtr. step	
	ε_{max}	ε_{RMS}	ε_{max}	ε_{RMS}	ε_{max}	ε_{RMS}
32×32	2.2 (-7)	1.3 (-8)	8.4 (-8)	5.2 (-9)	8.4 (-8)	5.2 (-9)
64×64	5.8 (-8)	1.3 (-9)	3.2 (-10)	6.2 (-12)	2.7 (-12)	1.0 (-13)
128×128	1.2 (-8)	2.3 (-10)	2.1 (-11)	1.4 (-13)	1.9 (-14)	2.6 (-15)

Table 2: Maximum and RMS errors in the computed solution (4.5) after one, two and three subtraction steps.

4.2 Oscillatory Helmholtz *(OH)* Equation

The subtraction functions in this case can be constructed in the form of Eq.(4.1) with the basis functions:

$$H^n_{p,q}(x,y) = \frac{\sin\ \lambda_n(x_p - x)}{\sin\ \lambda_n}\ \frac{\sinh \mu_n(y_q - y)}{\sinh \mu_n}, \qquad \lambda_n > \kappa \tag{4.6a}$$

or

$$H^n_{p,q}(x,y) = \frac{\sin\ \lambda_n(x_p - x)}{\sin\ \lambda_n}\ \frac{\sin \mu_n(y_q - y)}{\sin \mu_n}, \qquad \lambda_n < \kappa \tag{4.6b}$$

where

$$\mu_n = |\kappa^2 - \lambda_n^2|, \qquad x_0 = y_0 = 0,\ x_1 = y_1 = 1 \tag{4.7}$$

and κ is the parameter of Eq.(2.1b). The choice of λ_n is restricted by the condition $\sin \lambda_n \neq 0$. In the case of Eq.(4.6b) we also require that $\sin \mu_n \neq 0$. Altogether, we have

$$\lambda_n \neq \lambda_n^*, \quad \lambda_n^* = 0, \pi, 2\pi, ..., \kappa, \sqrt{\kappa^2 - \pi^2}, ... \tag{4.8}$$

In practice, we can choose λ_n in a close vicinity of λ_n^*, $|\lambda_n - \lambda_*| < 10^{-2} - 10^{-3}$, without affecting the accuracy. Notice that subtraction functions in the form of (4.6a) can be also employed for the Laplace equation ($\kappa = 0$) as an alternative to the polynomial harmonic functions constructed in [12].

Example 3.

Solve the *OH* equations in the rectangle $\Omega = \{0 \leq x \leq \pi, 0 \leq x \leq \pi\}$ with the boundary conditions corresponding to the exact solution

$$u_{ex} = \sin \nu x\ \frac{\sin \sqrt{\kappa^2 - \nu^2}(y - \pi/2)}{\sin \frac{\pi}{2}\sqrt{\kappa^2 - \nu^2}}, \qquad \lambda > \nu. \tag{4.9}$$

Table 3 shows the results for $\nu = 1.4$ and $\kappa^2 = 4, 42, 102$ and 163.

The remarkable fact here is that the error depends only slightly on κ. In other words, a rapidly oscillating ("rough") solution with large κ can be obtained with the *same* accuracy and computational efforts as a slowly oscillating ("smooth") solution with small κ. The explanation is the following. In this example we have a solution that changes slowly in one

$N_x \times N_y$	κ^2	after I subtr. step		after II subtr. step		after III subtr. step	
		ε_{max}	ε_{RMS}	ε_{max}	ε_{RMS}	ε_{max}	ε_{RMS}
64×64	4	1.2 (-6)	3.0 (-8)	5.3 (-10)	9.7 (-12)	2.4 (-14)	8.2 (-16)
	42	1.3 (-6)	3.5 (-8)	5.5 (-10)	2.4 (-11)	2.6 (-14)	2.8 (-15)
	102	1.3 (-6)	2.9 (-8)	6.0 (-10)	4.7 (-11)	2.6 (-13)	6.7 (-14)
	163	1.5 (-6)	8.0 (-8)	6.9 (-10)	1.2 (-10)	3.2 (-13)	5.4 (-15)

Table 3: Maximum and RMS errors in the computed solution (4.9) for $\nu = 1.4$

direction (small ν) and rapidly in another (large κ). According to Corollary 1 (see beginning of this section), it is sufficient to perform all necessary operations (e.g. to compute derivatives of a boundary function) in a "smooth" direction with the corresponding high accuracy. Then the extension in the second ("rough") direction is carried out analytically according to the expressions in (2.5). In this way we increase the accuracy of computation by taking advantage of the analytical form of the solution.

Notice that in the most current numerical algorithms for solving the (oscillatory) Helmholtz equation the computational complexity increases considerably as κ becomes larger. For example, using the algorithm of [2] based on a preconditioned iteration method, the solution of the same problem (with $\nu = 0.5$, resolution 61×61 and $\varepsilon_{RMS} < 10^{-6}$) requires 304 iterations for $\kappa = 2.77$ and 342 iterations for $\kappa = 4.16$. For higher resolutions this tendency becomes even more pronounced.

Consider now the same problem in the case of equal wave numbers, $\nu = \sqrt{\kappa^2 - \nu^2}$. In such a case there is no preferable direction in which one could perform computations with higher accuracy. Table 4 gives the numerical errors after three subtraction steps for the resolution 128×128 and several values of ν. The error increases as ν becomes larger. The rate of convergence can be improved, however, by using higher order subtraction functions.

$N_x \times N_y$	ν	after three subtraction steps	
		ε_{max}	ε_{RMS}
128×128	1.4	4.5 (-15)	1.6 (-15)
	3.4	6.8 (-12)	5.0 (-12)
	5.4	2.1 (-11)	5.4 (-12)
	7.4	2.1 (-10)	1.5 (-10)

Table 4: Maximum and RMS errors in the computed solution (4.9) with equal wave numbers in both directions.

5 Corner Singularities

In this section we consider a more general setting when the boundary functions (2.3) have jumps at one or several corner points or when they are not compatible with the equation (2.1). This causes singularities in the solution. Consider, for example, the case of the *MH* equation when the following inequalities take place at the point $x = 0$, $y = 0$ (corner I):

$$\phi_2 - \phi_1 = \delta_I^{(0)} \neq 0, \tag{5.1}$$

$$\frac{d^2\phi_1}{dx^2} + \frac{d^2\phi_2}{dy^2} - \lambda^2\phi_1 = \delta_I^{(2)} \neq 0, \tag{5.2}$$

$$\frac{d^4\phi_1}{dx^4} - \frac{d^4\phi_2}{dy^4} - \lambda^2\left(\frac{d^2\phi_1}{dx^2} - \frac{d^2\phi_2}{dy^2}\right) = \delta_I^{(4)} \neq 0 \tag{5.3}$$

and similarly at the other corners.

When $\delta_{\mathcal{P}}^{(0)}$ ($\delta_{\mathcal{P}}^{(2)}$, $\delta_{\mathcal{P}}^{(4)}$ are not equal to zero, the solution (correspondingly its second, fourth and higher derivatives) have singularities at the corner $\mathcal{P}$. Such singularities deteriorate the convergence of any numerical method. For example, for a second order finite difference method the $O(N^{-2})$ convergence is reduced to a $O(N^{-1/2})$ convergence, see [8]). High order methods, in particular spectral methods, are even more sensitive to the presence of singularities in the solution.

In [12] we proposed an approach which consists of removing singularities by subtracting the singular part of the solution. This is in the spirit of the present method which is aimed at removing jumps in the periodic prolongation of boundary functions using the subtraction technique.

More specifically, the Modified Fourier algorithm in the presence of corner singularities (MFS method) includes the following steps.

Algorithm V **(solution in the presence of corner singularities)**

1. Seek the solution as a superposition of the smooth U and the singular $\sigma^{(r)}$ components

$$u = v^{(r)} + \sigma^{(r)} \tag{5.4}$$

where $\sigma^{(r)}(x,y)$ is a *known* function which satisfies an original homogeneous elliptic equation. Besides, it is subject to the same discontinuity conditions (5.1)-(5.3) at the corner points, as the boundary function. We shall refer to $\sigma^{(r)}$ as the r- order singular corner function (SCF).

2. Construct the smooth part of the solution, $v^{(r)}$, which does not have singularities at the corners. It satisfies the same homogeneous equation as the original solution u, with smooth boundary conditions $v^{(r)}|_{\partial\Omega} = (u - \sigma^{(r)})|_{\partial\Omega}$. Thus, it can be computed using Algorithm IV.

3. Reconstruct u according to (5.4)

□

5.1 Construction of $\sigma^{(r)}$

It was shown in [12] that in the case of the Poisson equation singularities at the corners can be removed, up to a required degree, by using the SCF in the form:

$$\sigma = \mathcal{R}e\left\{\sum_{k=0}^{K} c_k z^{2k} \ln z\right\}, \qquad z = x + iy, \quad c_k = a_k + ib_k. \tag{5.5}$$

The complex coefficients c_k are expressed via derivatives of the boundary functions at the corner points. Such a form is in accordance with Lehman's theorem [7]. This theorem designates the type of singularity in the solution of the second order elliptic equation near the corner subtending an interior angle π/β. It states that the lowest order term in the asymptotic expansion near the corner is proportional to z^β if β is irrational, and to $z^q \log z$ if $\beta = q/p$ is a rational number. The present case of rectangular domains corresponds to $q = 2$. Therefore, the leading order singular term is $z^2 \log z$. The function $\sigma(z)$ in (5.5) is a harmonic function satisfying the homogeneous Laplace equation.

However in the case of the Helmholtz equation the use of (5.5) as a SCF would introduce a singular non-uniform source term which is difficult to deal with by the Fourier method. We propose the following numerical algorithm to construct a singular solution to the homogeneous Helmholtz equation.

The procedure is independent of a particular boundary distribution. It can be also applied, with only slight modifications, to all types of second order elliptic equations (2.1a) through (2.1c). Thus, the SCF can be precomputed and stored (for a given resolution) to be used for every boundary distribution.

To illustrate our approach, we construct the SCF $\sigma_I^{(2k)}$, related to the corner I, in the case of the *MH* equation. The subscript k indicates that the jump appears in the boundary function ($k = 0$), or in its second ($k = 1$), fourth ($k = 2$) or higher derivative. Functions related to other corners can be obtained from $\sigma_I^{(2k)}$ by merely reflecting across the lines $x = 0.5$ and $y = 0.5$. For example, $\sigma_{II}^{(2k)}(x, y) = \sigma_I^{(2k)}(1 - x, y)$.

Jump in the function ($k = 0$)

In this case the boundary function has a jump at the corner point $x = 0, y = 0$, that is condition (5.1) holds. To remove this jump one can use a SCF in the form (below we suppress subscript "I" in $\sigma_I^{(0)}$):

$$\sigma^{(0)}(r) = \frac{2\theta}{\pi} I_0(\lambda r) \tag{5.6}$$

where $r = \sqrt{x^2 + y^2}$, $\theta = \arctan(y/x)$, and $I_0(r)$ is the modified Bessel function of order zero. It is easily seen that $\sigma^{(0)}$ is the solution to Eq.(2.1a). At the corner I we have:

$$x \to 0, \ \ y = 0 \qquad\qquad \sigma^{(0)} = 0,$$

$$x = 0, \ \ y \to 0 \qquad\qquad \sigma^{(0)} \to I_0(0) = 1.$$

Therefore, the residual function $v^{(0)} = u - \delta^{(0)}\, \sigma^{(0)}$ satisfies the same *MH* equation as the original solution u, but it has no jump at the corner I.

Jump in the second derivative ($k=1$)

Suppose that the condition (5.2) holds, that is the second derivative is discontinuous at the corner I. Assume that the boundary functions $\phi_1, ..., \phi_4$ vanish at the corner points. If it is not the case, we can apply first the subtraction procedure using the corner functions (4.2), in order to make zero values at the corners.

We can seek a SCF, related to the corner I, as the solution of the MH equation

$$\Delta\sigma^{(2)} - \lambda^2\sigma^{(2)} = 0 \tag{5.7}$$

with the following boundary conditions

$$y=0, \qquad \sigma^{(2)} = s(x;\alpha_1,\alpha_2) \equiv \frac{\sinh\ \alpha_2(1-x)}{\sinh\ \alpha_2} - \frac{\sinh\ \alpha_1(1-x)}{\sinh\ \alpha_1},$$

$$x=0,\ x=1,\ y=1, \qquad \sigma_2 = 0. \tag{5.8}$$

where $\alpha_2^2 - \alpha_1^2 = 1$ (e.g. $\alpha_1 = 1,\ \alpha_2 = \sqrt{2}$). The boundary function s has the following properties:

$$\text{at } x=0, \qquad s=0,\ s''=1,$$

$$\text{at } x=1, \qquad s=s''=s''''=...=0. \tag{5.9}$$

Thus, the function $v^{(2)} = u - \delta^{(2)}\sigma^{(2)}$ satisfies Eq.(2.1a) and has a continuous second derivative at the corner I.

We construct the solution to (5.7), (5.8) numerically in the discretized region $\Omega_g = \{(x_i,y_j)|\ x_i=(i-1)\Delta,\ y_j=(j-1)\Delta, 1\le i,j\le N+1\}$, with the grid spacing $\Delta = 1/N$.

The algorithm consists of the following steps.

Algorithm VI (construction of $\sigma^{(2)}$)

1. Introduce an auxiliary "fine" grid $\mathcal{L} = \{\xi_i, |\ \xi_i = (i-1)/N_f,\ \ 1\le i\le N_f+1\}$ where N_f is a given power of two. Project the function $s(x)$ on the grid $\mathcal{L}$: $s_i = s(\xi_i)$.

2. Reflect $s(\xi)$ anti-symmetrically across the point $\xi = 1$:

$$\tilde{s}(\xi) = \begin{cases} s(\xi) & 0<\xi<1, \\ -s(2-\xi) & 1<\xi<2. \end{cases}$$

The function $\tilde{s}(\xi)$ is continuous and continuously differentiable, but it has a discontinuous second derivative at $\xi = 0$, that is $\tilde{s}\in C^1$.

3. Expand $\tilde{s}(\xi_i)$ into sine series (using the DST): $\tilde{s}_i \longrightarrow \hat{s}_k,\ \ k=0,...,2N_f$.

4. For each discrete value of $y = y_i \in \Omega_g$ corresponding to the main ("coarse") grid, do:
- multiply the Fourier coefficients $\hat{s}_k$ by the complementary function of y according to (2.5):

$$\hat{\sigma}_k = \hat{s}_k\,\frac{\sinh\ \mu_k(1-y_j)}{\sinh\ \mu_k}, \qquad \mu_k^2 = (2\pi k)^2 + \lambda^2.$$

- transform back to the physical space by using the inverse DST: $\hat{\sigma}_k \longrightarrow \sigma^{(2)}(\xi_i)$.

6. Project onto the coarse grid: $\sigma(\xi_i) \to \sigma(x_i)$.

□

The above algorithm is similar to Algorithm I in section 2 except that the DST is applied on a "fine" grid with $2N_f + 1$ collocation points. This is necessary in order to compute the SCF with no less accuracy than the smooth part. Since the boundary function $\tilde{s}$ is in C^1, the associated Fourier series converges rather slowly, like $O(N^{-3})$. Therefore, we would like to discretize $\tilde{s}$ on a finer grid such that the maximum approximation error at the distance $\Delta = 1/N$ from the point of discontinuity be less than the prescribed accuracy. Since the Gibbs oscillations decay inside the region, the maximum approximation error on the interval $\Delta < x < 1 - \Delta$ remains within this accuracy.

Table 5 shows maximum errors in the computed function $\sigma^{(2)}$ for several resolutions $N \times N$ while using various auxiliary fine grids N_f. The solution of reference, σ_*, is computed with $N_f = 8192$.

	$\mid \sigma_{N_f} - \sigma_* \mid$			
$N \times N$	$N_f = 512$	$N_f = 1024$	$N_f = 2048$	$\mathcal{N} = 4096$
16×16	7.4 (-11)	4.6 (-12)	2.9 (-13)	1.6 (-14)
32×32	3.1 (-10)	1.9 (-11)	1.2 (-12)	7.1 (-14)
64×64	1.3 (-9)	7.9 (-11)	4.9 (-12)	2.9 (-13)

Table 5:

In the case of the OH equation, the construction is similar except that in step 4 of the above algorithm one should multiply the Fourier coefficients $\hat{s}_k$ by a different complementary (hyperbolic or trigonometric) function according to (2.5). In the case of the Laplace equation, the SCF can be also constructed numerically using Algorithm VI. However in this case it is preferable to use the exact analytic form (5.5).

Jump in the fourth derivative ($k = 2$)

Consider the case when jumps exist in the second and the fourth derivatives (both inequalities (5.2) and (5.3) take place). To remove these jumps we construct a fourth order SCF, $\sigma_4(x, y; \delta_2,\ \delta_4)$. This function satisfies the homogeneous equation (5.7) with the following boundary conditions:

$$\begin{aligned} y = 0, &\quad \sigma^{(4)} = As(x; \alpha_1, \alpha_2) + Bs(x; \alpha_3, \alpha_4), \\ x = 0,\ x = 1,\ y = 1, &\quad \sigma_4 = 0 \end{aligned} \tag{5.10}$$

where the function $s(x; \alpha_i, \alpha_j)$ is defined in (5.8). The parameters A, B and $\alpha_i \neq 0$ have to satisfy the following relations

$$A\ (\alpha_2^2 - \alpha_1^2) + B\ (\alpha_4^2 - \alpha_3^2) = \delta^{(2)},$$

$$A\ (\alpha_2^4 - \alpha_1^4) + B\ (\alpha_4^4 - \alpha_3^4) - A\ (\alpha_2^2 - \alpha_1^2) + B\ (\alpha_4^2 - \alpha_3^2)\ \lambda^2 = \delta^{(4)}.$$

For example, one can choose

$$\alpha_1^2 = 1/2, \quad \alpha_2^2 = \alpha_3^2 = 3/2, \quad \alpha_4^2 = 5/2,$$

$$A = -\frac{1}{2}\ \left(\delta^{(2)}(\lambda^2 - 4) + \delta^{(4)}\right), \quad B = \frac{1}{2}\ \left(\delta^{(2)}(\lambda^2 - 2) + \delta^{(4)}\right).$$

Thus, the jumps in the second and fourth derivatives of $\sigma^{(4)}$ match the corresponding jumps in the boundary function, Eqs.(5.2)-(5.3). Note that only amplitudes A and B depend on the jumps $\delta^{(2)}$, $\delta^{(4)}$ while α_i are constant numbers. It allows us to construct the fourth order SCF $\sigma_4(x, y; \delta_2,\ \delta_4)$ as a linear combination of two second-order functions $\sigma^{(2)}(\alpha_1, \alpha_2)$ and $\sigma^{(2)}(\alpha_3, \alpha_4)$, defined by Eqs.(5.7)-(5.9). These functions can be precomputed and stored to be used for various boundary conditions and arbitrary jumps at the corners. Similar constructions can be carried out for the OH and the Laplace equations.

We would like to point out that the SCF constructed above are able to cope with singularities in the solution resulting from not smoothness of the boundary. The boundary functions on each side of the rectangle are assumed to be continuous and continuously differentiable.

Example 4.

Consider the Poisson equation with the constant forcing term and zero boundary conditions

$$\Delta u = -1 \qquad \text{in } \ \Omega = [0,1]^2, \qquad u|_{\partial\Omega} = 0. \tag{5.11}$$

The solution of (5.11) has singularities at the corner points where the sum $u_{xx} + u_{yy}$ must be zero due to the boundary conditions, whereas it is equal to -1 everywhere inside the region due to the equation.

We can seek the solution of Eq.(5.11) as a superposition of the particular and the homogeneous parts, $u = u_p + u_h$. For this simple example we can easily construct the particular solution, for example $u_p = x\ (1 - x)$ (a general case of non-homogeneous problems with arbitrary source functions is discussed in section 7). The homogeneous component satisfies the Laplace equation with the boundary conditions $u_h|_{\partial\Omega} = (u - u_p)|_{\partial\Omega}$. In our case

$$\Delta u_h = 0 \qquad \text{in } \ \Omega, \qquad u_h|_{\partial\Omega} = -x(1 - x).$$

The maximum error in the numerical solution, obtained by several methods, is listed in Table 6. MF denotes the present modified Fourier method based on Algorithms III, IV. MFS corresponds to the same method incorporating special treatment of corner singularities (Algorithms V, VI). CT is the Chebyshev-tau method. CG is the Chebyshev-Galerkin method of [9]. The results in the first column are obtained using a second order finite-difference scheme.

The CT and the MF methods have approximately the same accuracy, which is somewhat less than that of the LG method. A remarkable improvement of accuracy is achieved by preliminary removal of corner singularities, as implemented in the FMS method.

Example 5. Consider the MH equation

$$\Delta u - \lambda^2 u = 0, \qquad \text{in } \ \Omega = [0,1]^2$$

$N \times N$	$f-d$	CT	LG	MF	$MFS(r=2)$
8×8	1.7 (-3)			5.6 (-5)	3.4 (-9)
16×16	4.5 (-4)	3.5 (-5)	1.4 (-6)	1.4 (-5)	1.3 (-11)
32×32	1.1 (-4)	2.2 (-6)	7.4 (-8)	3.4 (-6)	5.1 (-14)
64×64	2.8 (-5)			8.6 (-7)	3.6 (-16)

Table 6: The accuracy of several numerical methods for Example 4

with the following boundary conditions:

$$y = 0, \qquad u = ax(1-x) + bx(1-x)^2,$$

$$x = 0,\ x = 1,\ y = 1, \qquad u = 0.$$

The results are shown in Table 7 for two particular cases $a = 1, b = 0$ ($\delta^{(2)} = 2,\ \delta^{(4)} = 0$) and $a = 0, b = 1$ ($\delta^{(2)} = 0, \delta^{(4)} = 24$). The comparison of the MFS and the MF methods allows one to see vividly the effect of special treatment of the corner singularities. The results in this table are obtained for $\lambda^2 = 40$. However, the dependence on λ is not significant.

	Jump in 2-d derivative ($a = 1, b = 0$)		Jump in 4-th derivative ($a = 0, b = 1$)	
$N \times N$	MF	$MFS(r=2)$	MF	$MFS(r=2)$
16×16	1.3 (-5)	3.0 (-12)	3.1 (-8)	2.8 (-11)
32×32	3.4 (-6)	5.0 (-14)	2.0 (-9)	4.6 (-13)
64×64	8.2 (-7)	9.3 (-16)	1.3 (-10)	8.0 (-15)

Table 7: Maximum error in the solution computed by the MF and the MFS methods.

6 Non-homogeneous Elliptic Problems

In this section we discuss an algorithm for the solution of non-homogeneous elliptic problems

$$\mathcal{L}u = f, \qquad \text{in } \Omega = [0,1]^2, \qquad u|_{\partial\Omega} = \Phi \tag{6.1}$$

where $\mathcal{L}$ is one of the differential operators in (2.1). We represent the solution of (6.1) as a superposition of the particular and the homogeneous parts, $u = u_p + u_h$. The particular solution is constructed using an appropriate numerical method. Then, we solve a b.v.p. for the homogeneous part u_h using the above techniques.

The algorithm incorporates the following steps.

Algorithm VII **(solution of non-homogeneous problems)**

1. Construct a particular solution u_p of Eq.(6.1) with some convenient (e.g. periodic or zero) boundary conditions.

2. Compute the deficiency on the boundaries

$$\tilde{\Phi} = \Phi - u_p \qquad\qquad \text{on } \partial\Omega. \tag{6.2}$$

3. Solve the homogeneous b.v.p.

$$\mathcal{L}u = 0, \qquad \text{in } \Omega = [0,1]^2, \qquad u|_{\partial\Omega} = \tilde{\Phi}. \tag{6.3}$$

using Algorithm V with the concomitant techniques of the previous sections.

4. Reconstruct the solution to Eq.(6.1) as $u = u_p + u_h$.

□

We compute the particular solution using the Local Fourier Basis (LFB) method of [5], [11] It consists of the following steps:

- extend the source function f beyond the boundaries over the distance 2ϵ;

- perform a smooth projection ("folding" across the new boundaries situated at the distance ϵ outside the original ones);

- apply the Fourier transform to a smoothed function on the extended domain $\Omega_e = \Omega \bigcup \omega$, where ω is an extra region of the width ϵ surrounding Ω.

A more detailed discription of the LFB method is given in Appendix B.

The advantage of this approach is that both the particular and the homogeneous solutions are computed on the same (uniform) grid of collocation points. The computational complexity of the LFB method scales like

$$20N_e^2 log_2 N_e + O(N_e^2), \qquad N_e = N + 2n. \tag{6.4}$$

Here N^2 is the number of grid points in the computational domain Ω, N_e^2 is the number of points in an extended domain Ω_e, and $n = O(1)$ is the number of points across ω. Typically $n \ll N$, and the relative "extra" work (for applying the FFT on an extended region) scales like $O(n/N)$, see [5]. Combining (6.4) with (2.9), we can estimate the computational complexity of Algorithm VII as

$$20N^2 log_2 N + 20N_e^2 log_2 N_e + O(N^2).$$

In Example 4 we consider a particular case of the Poisson equation with constant forcing. Some other examples of the Poisson equation have been discussed in our earlier paper [12]. Following new examples related to the Helmholtz equation.

Example 6.

Consider the OH equation with the sinusoidal forcing term

$$\Delta u + \lambda^2 u = (\lambda^2 - 2\omega^2\pi^2)\cos(\omega\pi x)\ \cos(\omega\pi y), \qquad \text{in } \Omega = [0,1]^2 \tag{6.5}$$

ω	$N=16$	$N=32$	$N=64$
1	3.1 (-8)	9.0 (-13)	1.0 (-13)
3	6.8 (-7)	2.4 (-11)	1.5 (-12)
5	5.0 (-6)	2.5 (-10)	1.6 (-11)

Table 8: Maximum pointwise errors in the solution of Eq.(6.5) for $\lambda^2 = 40$.

whose exact solution is $u = \cos(\omega\pi x)\cos(\omega\pi y)$. The results are shown in Table 8 for $\lambda^2 = 40$ and several resolutions $N = 16, 32$ and 64. The odd wave numbers ω correspond to discontinuous boundary functions.

The errors decay while N increases. However the rate of convergence is rather slowly. It can be improved, however, by implementing this method with a higher order subtraction procedure.

Example 7.

Consider the *MH* equation with the Gaussian source function

$$\Delta u - \lambda^2 u = (4\omega^2 r_0^2 - 4\omega - \lambda^2)\, e^{-\omega r_0^2}, \qquad \text{in } \Omega = [0,1]^2 \tag{6.6}$$

where $r_0 = (x - x_0)^2 + (y - y_0)^2$. The corresponding exact solution is $u = e^{-\omega r_0^2}$. The maximum pointwise errors are listed in Table 9.

ω	$N=32$	$N=64)$	$N=128)$
1	3.1 (-12)	1.9 (-13)	6.2 (-14)
10	6.7 (-10	1.3 (-12)	1.3 (-14)
40	6.7 (-7)	1.4 (-10)	5.6 (-13)
100	3.5 (-6)	1.2 (-9)	2.8 (-12)

Table 9: Maximum pointwise errors in the solution of Eq.(6.6) for $x_0 = 0.3, y_0 = 0.5$, $\lambda^2 = 40$.

For large ω, a sharp Gaussian profile is located inside the region around the point (x_0, y_0), while near the boundaries the distribution is flat. Therefore, for large ω, the rate of convergence is practically exponential.

7 Conclusions and Generalizations

A direct and fast algorithm is developed for the solution of the Poisson and the Helmholtz equations (both of monotonic and oscillatory types) in rectangular domains. The method is shown to be of high (in fact arbitrary) order accuracy.

The solution of boundary value problems is sought in the form of series expansion into trigonometric/hyperbolic bases. Several functions in this basis represent a discontinuous component of the solution while the rest of the basis functions approximate a smooth component. The expansion coefficients are computed by the Fast Fourier Transform. Evaluating the solution at a given interior point requires an order of $N \log N$ arithmetic operations, where N is the number of grid points in each direction. Constructing the solution at all $N \times N$ interior points requires $O(N^2 \log N)$ operations.

The present algorithm also permits an explicit treatment of singularities at the corner points. It is shown that singular solutions can be computed with no less accuracy than the smooth ones. The constructed singular corner functions can be employed by other methods, for example, by finite-differences or finite elements methods, to improve the convergence in the presence of corner singularities.

The present algorithm can be extended to the three-dimensional case. On the other hand, it is applied only in rectangular regions. Problems with variable coefficients can be treated by a preconditioned iteration method using the present elliptic solver as a preconditioner. The results will be reported elsewhere.

Appendix A

In sections 4, 5 we constructed subtractitition functions whose parameters are expressed in terms of derivatives of a non-periodic function on the boundaries. In this paper we compute derivatives using the following algorithm. Given the vector of the grid values $f_j = f(x_j),\ x_j = x_0 + j\Delta,\ \ \Delta = 1/N,\ j = 1, ..., N$, find the first p derivatives of the function f at the boundary point $x = x_0$.

Algorithm A **(numerical evaluation of boundary derivatives)**

1 Choose $s \geq p$ grid points, $x_j,\ \ j = 0, ..., s-1$, adjacent to the boundary x_0.

2 Solve the linear system for s unknown values D_j

$$\sum_{k=1}^{s} D_j \, \frac{(j\Delta)^k}{k!} = f_j - f_0, \qquad j = 1, ..., s.$$

3 Approximate $f^{(j)} \approx D_j$.

□

In fact we approximate the derivatives by the corresponding divided differences. This algorithm is found to be sufficiently accurate to be consistent with the present method up to order $6-8$.

We illustrate the accuracy of this procedure by computing several derivatives of the function $f(x) = e^{-\alpha x},\ \ x \in [0,1]$ at the boundary $x = 0$. Table 10 gives relative errors $|f^{(p)} - f_{ex}^{(p)}|/|f^{(p)}|_{max}$ in the first six derivatives. The mesh size $\Delta = 1/128$, the size of the stencil is $s = 9$.

The errors in the computed derivatives must affect the precision of the constructed subtraction functions and thus, the overall accuracy of the present MF method. We illustrate this effect by computing the error in the Fourier approximation of the function

$$f(x) = \frac{\sinh\ \alpha_1(1-x)}{\sinh\ \alpha_1} - \frac{\sinh\ \alpha_2(1-x)}{\sinh\ \alpha_2}, \qquad x \in [0,1] \qquad (A.1)$$

α	$p=1$	$p=2$	$p=3$	$p=4$	$p=5$	$p=6$
5	4.4 (-14)	6.3 (-12)	5.4 (-10)	3.3 (-8)	1.4 (-6)	4.7 (-5)
10	7.8 (-12)	5.7 (-10)	2.5 (-8)	8.0 (-7)	1.9 (-5)	3.5 (-4)
15	2.6 (-10)	1.2 (-8)	3.7 (-7)	7.8 (-6)	1.2 (-4)	1.5 (-3)
20	3.0 (-9)	1.1 (-7)	2.4 (-6)	3.8 (-5)	4.6 (-4)	4.2 (-3)

Table 10: Relative errors in the derivatives of the function $f = e^{-\alpha x}$ at $x = 0$; $\quad s = 9,\ \Delta x = 1/128$.

whose periodic extension is of class C^0. First, we apply Algorithm II of section 3 to obtain a smoother function $g^{(2r)} = f - F^{(2r)}$, where $F^{(2r)}$ is a $(2r+1)-$ degree polynomial (its parameters are expressed via the first $2r$ derivatives of $f(x)$ on the boundaries $x = 0, 1$). Then we compute the Fourier coefficients $\hat{g}_k$ of a smoothed function $g^{(2r)}$ by applying the Discrete Fast Fourier Transform (DFFT) to the grid values at the sampling points $x_j = (j-1)\Delta,\quad \Delta = 1/N,\ j = 1, .., N$. Next, we reconstruct $g^{(2r)}$ in the physical space at the shifted points $\xi_j = x_j + \frac{\Delta}{2}$ (this can be done by applying the inverse DFFT routine to the "shifted" Fourier coefficients $\hat{g}_k e^{ik\tilde{\Delta}}$, where $\tilde{\Delta} = \frac{2\pi}{l}\ \frac{\Delta}{2}$, and l is the period). Finally, we reconstruct the original function $f = g^{(2r)} + F$ at the points $\{\xi_j\}$.

The results are shown in Table 11 for $N = 128$, $r = 0$ through 3, $\alpha_2 = 1$ and several values of α_1. One can conclude that the accuracy of the Fourier approximation is practically insensitive to the errors in the boundary derivatives incurred when using the above numerical procedure. See Table 10.

α_1	$r=0$	$r=1$	$r=2$	$r=3$	
	I, II	I, II	I, II	I	II
5	5.0 (-5)	6.0 (-9)	7.1 (-13)	5.9 (-15)	7.3 (-15)
10	2.0 (-4)	9.7 (-8)	4.5 (-11)	6.3 (-13)	5.6 (-13)
15	4.7 (-4)	4.9 (-7)	5.2 (-10)	6.7 (-12)	6.5 (-12)
20	8.3 (-4)	1.5 (-6)	2.9 (-9)	4.1 (-11)	3.6 (-11)

Table 11: Maximum relative errors in the Fourier approximation of a function Eq.(A.1); the subtraction function $g^{(2r)}$ is constructed using numerical (I) or exact (II) derivatives on the boundaries.

However, this simple procedure does not allow the computation of higher derivatives ($p > 6$) with suitable accuracy. Using a more accurate procedure would enable the MF method to construct higher order subtraction functions in order to achieve faster convergence.

Appendix B

For the convenience of the reader, we give here a brief description of the Local Fourier Basis (LFB) method for the solution of differential equations. The detailed analysis is presented in [5], [11].

We illustrate the LFB approach by solving a simple 1-D differential equation

$$\frac{d^2u}{dx^2} = f(x), \qquad x \in [a, b]. \tag{B.1}$$

Assume that the source function $f(x)$ is known on an extended interval $x \in [a - 2\epsilon, b + 2\epsilon]$.

The computational algorithm consists of two main steps.

1. Project the source function $f(x)$ in a smooth way on the interval $x \in [\bar{a}, \bar{b}]$, where $\bar{a} = a - \epsilon$, $\bar{b} = b - \epsilon$. This results in a smooth function $\tilde{f}(x)$, sup$\tilde{f} = [\bar{a}, \bar{b}]$, as in Fig.9.

2. Apply a pseudo-spectral Fourier method to Eq.(A.1) with the source function $\tilde{f}$ in the RHS.

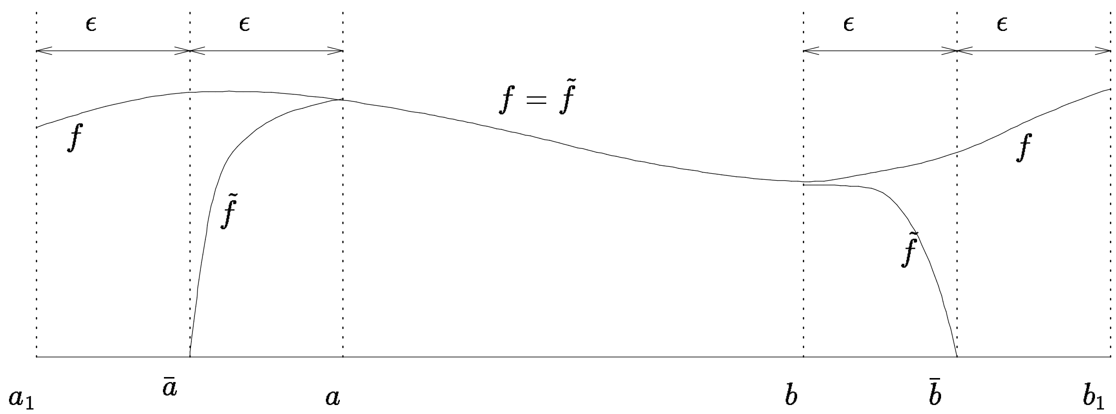

Figure 9. The projection operation

The projection procedure $\mathcal{P}$ consists of two "folding" operations across the extended boundaries $\bar{a}$ and $\bar{b}$:

$$\tilde{f} = \mathcal{P}f = \mathcal{F}_{\bar{a}}\mathcal{F}_{\bar{b}}f(x) \tag{B.2}$$

where the "folding" across the line $\bar{a}$ is defined as follows

$$\mathcal{F}_{\bar{a}} = B(x)f(x) - B(2\bar{a} - x)f(2\bar{a} - x) \tag{B.3}$$

and similarly across the line $\bar{b}$. Here $B(x)$ is the window function which satisfies the following properties:

$$\begin{array}{ll} \multicolumn{2}{c}{\text{sup}B(x) = [a - 2\epsilon,\ b + 2\epsilon],} \\ B^2(x) + B^2(2\bar{a} - x) = 1 & x \in [a - 2\epsilon, a) \\ B(x) = 1 & x \in [a, b] \\ B^2(x) + B^2(2\bar{b} - x) = 1 & x \in (b, b + 2\epsilon]. \end{array} \tag{B.4}$$

Inside the computational interval $[a, b]$, this function is equal to $B = 1$ and it smoothly decays outwards over a distance 2ϵ. Some specific forms of $B(x)$ and the influence of the size of the extra interval ϵ on the accuracy are tested in [5].

The smoothed function $\tilde{f}$ can be then represented by a rapidly convergent Fourier series

$$\tilde{f}(x) = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}} \hat{f}_k \sin(\pi k x/\bar{l}), \qquad \bar{l} = \bar{b} - \bar{a}.$$

This allows us to accurately apply the Fourier method to the solution of Eq.(A.1) on the interval $x \in [\bar{a}, \bar{b}]$ with the smooth source function $\tilde{f}$. The expansion coefficients can be obtained by using the Fast Sine Transform when the function $\tilde{f}(x)$ is discretized on a uniform grid.

According to the definitions (B.2) and (B.3), and the properties of the window function (B.4), we have $\tilde{f} = f$ on $x \in [a, b]$. Therefore, the solution of this auxiliary problem coincides with the solution of interest $u(x)$ in the main region $x \in [a, b]$.

References

[1] I. Babushka, W. Rheinboldt, Error estimates for adaptive finite element computations, *SIAM J. Numer. Anal.*, **15**, p. 736-754, (1978).

[2] A. Bayliss, C. Goldstein, E. Turkel, An iterative method for the Helmholtz equation, *J.Comput.Phys.*, **49**, No. 3, p.443-457, (1983).

[3] G. Birkhoff, R.E. Lynch, Numerical solution of elliptic problems, *SIAM Publications*, 1984.

[4] Fang Q. Hu A fast numerical solution of scattering by a cylinder: spectral method for the boundary integral equation. ICASE report No.94-36, May 1994.

[5] M. Israeli, L. Vozovoi, A. Averbuch, Spectral multi-domain technique with Local Fourier Basis, *J. Scientific Computing*, **8**, N 2, p. 135-149, (1993).

[6] L.V. Kantorovich, V.I. Krylov, Approximate methods of higher analysis, Groningen, the Netherlands, 1958.

[7] R.S. Lehman, Developments at analytic corner of solutions of elliptic partial differential equations, *J.Math.Mech.*, **8**, p. 727-760, (1959).

[8] W. Liniger, F. Odeh, On the numerical treatment of singularities in the solutions of Laplace's equation, Research report RC 18202 (79906), IBM T.J. Watson Research Center, NY, May 1992.

[9] J. Shen, Efficient spectral-Galerkin method II: direct solvers of second- and fourth-order equations using polynomials, *SIAM J.Sci.Comput.*, **15**, No.1, p.74-87, (1995).

[10] L. Vozovoi, M. Israeli, A. Averbuch, Spectral multi-domain technique with Local Fourier Basis II: decomposition into cells. *J. Scientific. Computing*, **9**, No. 3, p. 311-326, (1994).

[11] L. Vozovoi, M. Israeli, A. Averbuch, Spectral multi-domain technique with Local Fourier Basis II: decomposition into cells. *J. Scientific. Computing*, **9**, No. 3, p. 311-326, (1994).

[12] Vozovoi L., Israeli M, Averbuch A., A Fast Poisson Solver of Arbitrary order accuracy in Rectangular Regions, submitted to *J.Sci.Comput.*

[13] L. Vozovoi, M. Israeli, A. Averbuch, Analysis and application of Fourier-Gegenbauer method to stiff differential equations, to appear in *SIAM, J.Numer.Anal.*, **35**, No. 5, (1996)