Spectral Multidomain Technique with Local Fourier Basis II: Decomposition into Cells *

appeared in Journal of Scientific Computing, Vol. 9, No. 3, pp. 311-326, 1994

A. Averbuch[‡],L. Vozovoi[†], M. Israeli[†]

[†] Faculty of Computer Science, Technion, Haifa 32000, Israel

[‡] School of Mathematical Sciences, Tel Aviv University, Tel Aviv 69978, Israel

Abstract

The spectral multidomain method for the solution of 2-D elliptic and parabolic PDE's is developed. The computational region is decomposed into rectangular cells. A Local Fourier Basis technique is implemented for the discretization in space. Such a technique enables the global (typically $\sim 10^4 - 10^5$) matching relations for the interface unknowns to be decoupled into a set of relations for only few interface points at a time.

1 Introduction

Domain decomposition methods for solving differential equations were intensively developed during the last several years because they are suitable for parallel computers of the MIMD (message-passing or shared-memory) type. The basic technique consists of splitting the computational region into smaller subdomains, distributed to different processors. The computations are carried out concurrently by all the processors, after which the particular solutions in the subdomains are "patched" to ensure the smoothness of the global solution.

In [1] a spectral domain decomposition method was developed with the Local Fourier Basis, that was suggested previously in [2] for other purposes. This basis preserves the usual advantages of the ordinary trigonometric basis for the spectral differentiation, spectral rate of convergence and accuracy. At the same time, it enables one to fulfil an overwhelming part of the computational work in each processor independently, because the basis functions are localized on a size of one subdomain. The required continuity of unknowns on the interfaces is achieved by a direct (not iterative) matching procedure, which exploits an exact form of the corresponding homogeneous solution.

This approach was implemented [1] for the solution of elliptic and time-dependent parabolic problems in one and two-dimensions. Two-dimensional problems were treated by domain decomposition in the form of parallel strips. After the FFT is applied along the strips, the

*This research is supported partly by a grant from the French-Israeli Binational Foundation for 1992-1993

problem is reduced to a set of uncoupled one-dimensional problems for the Fourier coefficients.

In this paper we generalize the previous method to apply to domain decomposition in the form of rectangular cells. Along with a general approach, a specific multidomain technique, pertinent to the 4×4- cells domain decomposition, is discussed.

In Section 2 we describe a domain decomposition technique in one-dimensional case with a global matching procedure, developed in [1]. In Section 3 we modify this algorithm to compute two-dimensional elliptic problems in rectangular regions divided into cells. In Section 4 we apply this method to the solution of time-dependent multidomain problems.

2 Domain Decomposition Technique in One-Dimension

We illustrate the method on an elliptic equation of the Helmholtz type

$$\frac{d^2U}{dx^2} - \lambda^2 U = F(x), \ x \in \mathcal{L} = [-\frac{L}{2}, \frac{L}{2}] \tag{2.1}$$

which is widely used in CFD as it is an outcome of semi-implicit time discretization procedure for time dependent parabolic problems. We consider the case of periodic boundary conditions

$$U(-L/2) = U(L/2), \ \frac{dU}{dx}(-L/2) = \frac{dU}{dx}(L/2) \tag{2.2}$$

(boundary conditions of Dirichlet and Neumann types are discussed in[1]).

The whole interval $\mathcal{L} = [-\frac{L}{2}, \frac{L}{2}]$ is decomposed into pieces (subdomains) $[\bar{x}_{n-1}, \bar{x}_n], \quad n = 1, ...P$ of the length $l = L/P$.

Before we treat the general approach, we consider one particular case.

2.1 Particular Case $P = 4$

It was shown in [2] that the solution of 2-domain problems with the Dirichlet boundary conditions (correspondingly 2×2-domain in two-dimensions) can be reduced to a set of one-domain problems. The procedure exploits the continuity conditions on the interfaces and appears as a folding of computational region across the interfaces.

We will show, that in the case of periodic boundary conditions (2.2) this approach can be extended to 4-domain problems (4×4- domain in two-dimensions- see Sec.3) by two successive foldings.

On the first step we divide the interval $\mathcal{L} = [-\frac{L}{2}, \frac{L}{2}]$ into two equal parts $\mathcal{L}_1 = [-\frac{L}{2}, 0], \mathcal{L}_2 = [0, \frac{L}{2}]$ and denote the corresponding pieces of functions U, F as $U^{(1)}, F^{(1)}$ in $\mathcal{L}_1$ and $U^{(2)}, F^{(2)}$ in $\mathcal{L}_2$. The continuity conditions at the interior point $x = 0$ can be written down as

$$x = 0, \quad U^{(1)} = U^{(2)}, \ \frac{dU^{(1)}}{dx} = \frac{dU^{(2)}}{dx} \tag{2.3}$$

Now, let us introduce functions H_1, H_2, G_1, G_2 in $\mathcal{L}_1$ such that

$$H_1(x) = U^{(1)}(x) + U^{(2)}(-x), \quad H_2(x) = U^{(1)}(x) - U^{(2)}(-x),$$

$$G_1(x) = F^{(1)}(x) + F^{(2)}(-x), \quad G_2(x) = F^{(1)}(x) - F^{(2)}(-x),$$
$$x \in [-L/4, 0] \tag{2.4}$$

The new dependent variables H_1, H_2 are subject to the boundary conditions

$$x = -L/2, 0 \qquad \frac{dH_1}{dx} = H_2 = 0, \tag{2.5}$$

that follows from (2.2) and (2.3).

The second step is implemented in a similar way. We divide once again the interval $\mathcal{L}_1$ into equal subintervals $\mathcal{L}_{11}, \mathcal{L}_{12}$, and denote the corresponding pieces of functions H_i, G_i as $H_{i,1}, G_{i,1}$ in $\mathcal{L}_{11}$ and $H_{i,2}, G_{i,2}$ in $\mathcal{L}_{12}$, $i = 1, 2$. Then, we introduce the coordinate $\xi = x + L_1/4$ and define new functions $S_{i,k}$, $R_{i,k}$ in $\mathcal{L}_{11}$:

$$S_{i,1} = H_{i,1}(\xi) + H_{i,2}(-\xi), \qquad S_{i,2} = H_{i,1}(\xi) - H_{i,2}(-\xi),$$
$$R_{i,1} = G_{i,1}(\xi) + G_{i,2}(-\xi), \qquad R_{i,2} = G_{i,1}(\xi) - G_{i,2}(-\xi),$$
$$\xi \in [-L/4, 0], \qquad i = 1, 2 \tag{2.6}$$

Finally, the original periodic boundary problem (2.1)-(2.2) on the interval $\mathcal{L}$ becomes to be a set of four problems on the subdomain $\mathcal{L}_{11}$ with the boundary conditions with the Dirichlet-Neumann types. From from (2.5) and (2.6) we have:

$$\frac{d^2 S_{ik}}{dx^2} - \lambda^2 S_{ik} = R_{ik}$$
$$\xi = -L/4 \ (x = -L/2) \qquad \frac{dS_{11}}{dx} = \frac{dS_{12}}{dx} = S_{21} = S_{22} = 0,$$
$$\xi = 0 \ (x = -L/4) \qquad \frac{dS_{11}}{dx} = \frac{dS_{21}}{dx} = S_{12} = S_{22} = 0,$$
$$i, k = 1, ...4, \tag{2.7}$$

2.2 General Case (arbitrary P)

In the case of an arbitrary number of subdomains P the computational algorithm consists of two stages. At first we construct particular solutions $U_p^{(n)}(x)$, $n = 1, 2, ..., P$ in each subdomain, using one of the available space discretizations, such as (pseudo) spectral or finite differences, with any arbitrary interface values $U_p(\bar{x}_n)$. A spectral multidomain technique with a Local Trigonometric Basis is described in [1]. The construction and implementation of this basis for the solution of differential equations is discussed briefly in the Appendix.

In the second stage we have to satisfy the continuity conditions on the interfaces

$$x = \bar{x}_n = nl, \qquad U^{(n)} = U^{(n+1)}, \quad \frac{dU^{(n)}}{dx} = \frac{dU^{(n+1)}}{dx}, \quad n = 1, 2, ..., P-1. \tag{2.8}$$

This is achieved by using the homogeneous solution of (2.1) in the form

$$U_h^{(n)} = A_n e^{\lambda(x-l)} + B_n e^{-\lambda x}, \quad x \in [0, l] \tag{2.9}$$

The complete solution to (2.1), (2.2) is represented as a superposition

$$U = \bigcup_{n=1}^{P} U^{(n)}, \quad U^{(n)} = U_p^{(n)} + U_n^{(n)}. \tag{2.10}$$

The interface problem (2.8)-(2.10) leads to the set of relations

$$A_n = A_{n+1}e^{-\lambda l} - \alpha_n, \quad B_{n+1} = B_n e^{-\lambda l} - \beta_n$$

$$\alpha_n = \frac{1}{2}(\frac{\delta_n'}{\lambda} + \delta_n), \quad \beta_n = \frac{1}{2}(\frac{\delta_n'}{\lambda} - \delta_n),$$

$$\delta_n = U_p^{(n)}(l) - U_p^{(n+1)}(0), \quad \delta_n' = \frac{dU_p^{(n)}(l)}{dx} - \frac{dU_p^{(n+1)}(0)}{dx}, \quad n = 1, 2, ..., P-1, \tag{2.11.1}$$

(δ_n and δ_n' are the jumps of the function and its first derivative on the interfaces $\bar{x}_n$, respectively).

The periodic boundary conditions require certain relations between the two extreme processors $n = 1$ and $n = P$:

$$A_P = A_1 e^{-\lambda l} - \alpha_P, \quad B_1 = B_P e^{-\lambda l} - \beta_P. \tag{2.11.2}$$

where α_P and β_P are given by (2.6) where the superscript $n = P+1$ is replaced by $n = 1$.

Once the system (2.11.1)-(2.11.2) is resolved with respect to one unknown coefficient, say A_P:

$$A_P = -\frac{1}{1 - e^{-\lambda P l}} \sum_{n=0}^{P-1} \lambda_n e^{-n\lambda l}$$

then, the other coefficients A_n, B_n are determined by using the reccurence relations (2.6). Thus, all the coefficients are represented explicitly as functions of the jumps δ_n and δ_n' on the interfaces.

3 Domain Decomposition Technique in Two-Dimensions

In the two-dimensional case the Helmholtz equation becomes

$$\Delta U_n - \lambda^2 U_n = F(x, y) \quad in\ \Omega = (-\frac{L_1}{2}, \frac{L_1}{2}) \times (-\frac{L_2}{2}, \frac{L_2}{2}) \tag{3.1}$$

The source function $F(x, y)$ is assumed to be periodic in both directions x and y . We seek a periodic solution of (3.1)

$$U(-L_1/2, y) = U(L_1/2, y), \quad \frac{\partial U(-L_1/2, y)}{\partial x} = \frac{\partial U(L_1/2, y)}{\partial x},$$

$$U(x, -L_2/2) = U(x, L_2/2), \quad \frac{\partial U(x, -L_2/2)}{\partial y} = \frac{\partial U(x, L_2/2)}{\partial y}. \tag{3.2}$$

The domain Ω is partitioned into $P_1 \times P_2$ rectangular cells of the size $l_1 \times l_2$; $l_1 = L_1/P_1$, $l_2 = L_2/P_2$.

3.1 Particular Case 4×4- Subdomains

The specific multidomain technique, described in Section 2.1, can be generalized to two-dimensions of 4×4- subdomains. This technique enables one to reduce the multidomain problem in Ω to a set of problems in one subdomain ω_{11} (see Fig. 1).

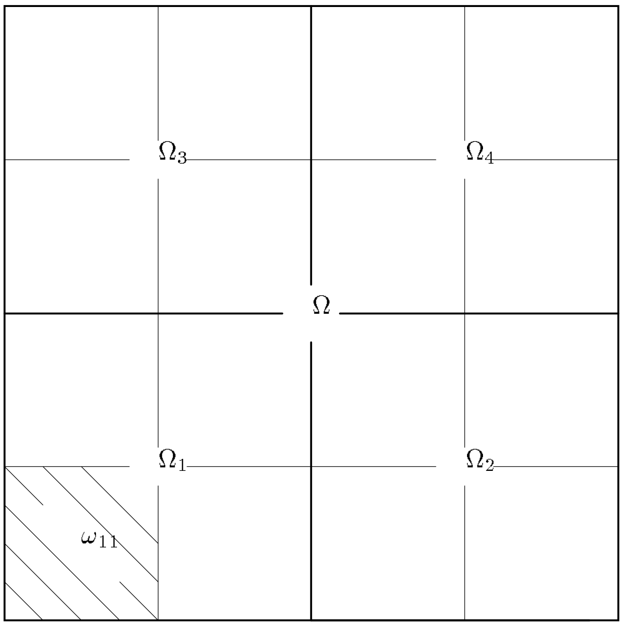

Figure 1: Domain decomposition into 4×4 subdomains

In the first step we decompose the whole region Ω into four subdomains Ω_i. The corresponding restrictions of u and f to Ω_i are denoted by u_i and f_i. The continuity conditions on the interfaces require:

$$
\begin{aligned}
&x = 0, y \in \left[-\frac{L_2}{2}, 0\right] && u_1 = u_2,\ \frac{\partial u_1}{\partial x} = \frac{\partial u_2}{\partial x} \\
&x = 0, y \in \left[0,\ \frac{L_2}{2}\right] && u_3 = u_4,\ \frac{\partial u_3}{\partial x} = \frac{\partial u_4}{\partial x} \\
&y = 0, x \in \left[-\frac{L_1}{2}, 0\right] && u_1 = u_3,\ \frac{\partial u_1}{\partial y} = \frac{\partial u_3}{\partial y} \\
&y = 0, x \in \left[0,\ \frac{L_1}{2}\right] && u_2 = u_4,\ \frac{\partial u_2}{\partial y} = \frac{\partial u_4}{\partial y}
\end{aligned}
\tag{3.3}
$$

Let us introduce the functions H_i in $\Omega_1 = (-L_1/2, 0) \times (-L_2/2, 0)$ as follows:

$$H_1(x, y) = U_1(x, y) + U_2(-x, y) + U_3(x, -y) + U_4(-x, -y)$$

$$H_2(x, y) = U_1(x, y) + U_2(-x, y) - U_3(x, -y) - U_4(-x, -y)$$

$$H_3(x, y) = U_1(x, y) - U_2(-x, y) + U_3(x, -y) - U_4(-x, -y)$$

$$H_4(x,y) = U_1(x,y) - U_2(-x,-y) - U_3(x,-y) + U_4(-x,-y) \tag{3.4}$$

and similarly, the functions $G_i(x,y)$ as the corresponding combinations of $F_i(\pm x, \pm y)$, $i = 1,..,4$. The boundary conditions for H_i follow from (3.2)-(3.4):

$$\begin{aligned} x = -\frac{L_1}{2}, 0 \qquad & \frac{\partial H_1}{\partial x} = \frac{\partial H_2}{\partial x} = H_3 = H_4 = 0 \\ y = -\frac{L_2}{2}, 0 \qquad & \frac{\partial H_1}{\partial y} = H_2 = \frac{\partial H_3}{\partial y} = H_4 = 0. \end{aligned} \tag{3.5}$$

The second step is similar to the previous one. We divide the subdomain Ω_1 in half for both directions and define the corresponding pieces of functions the H_i and G_i in subdomains ω_{1k} as H_{ik} and G_{ik}. Then, we introduce the new coordinates $\xi = x + L_1/4, \quad \eta = y + L_2/4$ and define the functions $S_{ik}(\xi,\eta)$ in ω_{11} :

$$\begin{aligned} S_{i,1}^{(\xi,\eta)} &= H_{i,1}(\xi,\eta) + H_{i,2}(-\xi,\eta) + H_{i,3}(\xi,-\eta) + H_{i,4}(-\xi,-\eta) \\ S_{i,2}^{(\xi,\eta)} &= H_{i,1}(\xi,\eta) + H_{i,2}(-\xi,\eta) - H_{i,3}(\xi,-\eta) - H_{i,4}(-\xi,-\eta) \\ S_{i,3}^{(\xi,\eta)} &= H_{i,1}(\xi,\eta) - H_{i,2}(-\xi,\eta) + H_{i,3}(\xi,-\eta) - H_{i,4}(-\xi,-\eta) \\ S_{i,4}^{(\xi,\eta)} &= H_{i,1}(\xi,\eta) - H_{i,2}(-\xi,\eta) - H_{i,3}(\xi,-\eta) + H_{i,4}(-\xi,-\eta) \\ i = 1,...4; \qquad & \xi \in \left[-\frac{L_1}{4}, 0\right], \eta \in \left[-\frac{L_2}{4}, 0\right] \end{aligned} \tag{3.6}$$

In a similar way we define $R_{i,k}(\xi,\eta)$ as the corresponding combinations of $G_{i,u}(\pm\xi, \pm\eta)$.

It is easily verified that the original problem (3.1), (3.2) with periodic boundaries on $\Omega = \bigcup_{i,k=1}^{4} \omega_{ik}$ is reduced to the set of 16 Dirichlet-Neumann boundary problems in the subdomain ω_{11}:

$$\begin{aligned} & \Delta S_{i,k} - \lambda^2 S_{i,k} = R_{i,k} \\ \xi = -L_1/4 \qquad & \frac{\partial S_{1,k}}{\partial \xi} = \frac{\partial S_{2,k}}{\partial \xi} = S_{3,k} = S_{4,k} = 0, \\ \xi = 0 \qquad & \frac{\partial S_{k,1}}{\partial \xi} = \frac{\partial S_{k,2}}{\partial \xi} = S_{k,3} = S_{k,4} = 0, \\ \eta = -L_2/4 \qquad & \frac{\partial S_{1,k}}{\partial \eta} = S_{2,k} = \frac{\partial S_{3,k}}{\partial \eta} = S_{4,k} = 0, \\ \eta = 0 \qquad & \frac{\partial S_{k,1}}{\partial \eta} = S_{k,2} = \frac{\partial S_{k,3}}{\partial \eta} = S_{k,4} = 0, \\ & i,k = 1,...4 \end{aligned}$$

where the boundary conditions for $S_{i,k}$ are merely the consequences of the formulas (3.5) and (3.6) and the continuity of the functions $H_{ik}(\xi,\eta)$ and their normal derivatives on the interfaces $\xi = \eta = 0$.

3.2 General Approach

In this section we develop general algorithm, which can be applied to an arbitrary number of subdomains $P_1 \times P_2$. The direct pointwise matching by using the corresponding two-dimensional homogeneous solutions of the form $e^{(\lambda_1 x + \lambda_2 y)}, \lambda_1^2 + \lambda_2^2 = \lambda^2$ would involve all the interfaces points and results in $2(N_1 P_1 + N_2 P_2)$ algebraic matching conditions, where $N1$ and $N2$ are the numbers of collocation points in the x and y directions. This approach was implemented for the 1×2 and 2×2 domain decomposition problems in [4], while the application to a larger number of subdomains seems to be cumbersome and expensive.

The implementation of the LFB technique enables us to reduce the overall coupling of the interface points to a sequence of simple local matching relations. The idea is to apply this algorithm alternatively in the x and y directions. Returning to the problems (3.1) and (3.2), we divide the domain Ω into rectangular cells $\Omega = \bigcup_{m=1}^{P_1} \bigcup_{k=1}^{P_2} \omega_n^m$ of size $l_1 \times l_2$; $l_1 = L_1/P_1, l_2 = L_2/P_2$ and denote the restrictions of U and F to ω_{mn} by $u^{(m,n)}$, $f^{(m,n)}$. The algorithm consists of several steps:

First step - we take an horizontal row of cells $\Omega_\nu = \bigcup_{m=1}^{P_1} \omega_\nu^m, \nu = 1, ..., P_2$ and transform the source function F into the Local Fourier Basis in the y-direction. The procedure includes an ϵ-extension of a function (using data from contiguous subdomains), smoothing across the boundaries $\bar{y} = (\nu - 1)l_2 - \epsilon$ and $\bar{y} = \nu l_2 + \epsilon$, antisymmetric reflection and applying the FFT on a local period $\Delta = 2(l_2 + 2\epsilon)$ (see Appendix).

Unknown Fourier coefficients $\hat{u}_\alpha^{(\nu,m)}(x)$, $\alpha = 0, 1, .., 2J_2 - 1$, $J_2 = N_2/P_2$ satisfy ordinary differential equations

$$\frac{d^2 \hat{u}_\alpha^{(\nu,m)}}{dx^2} - (k_\alpha^2 + \lambda^2)\hat{u}_\alpha^{(\nu,m)} = \hat{f}_\alpha^{\nu,m}(x), \; m = 1, ...P_1 \tag{3.7}$$

where $k_\alpha = \frac{2\pi}{\Delta}\alpha$ are the local wave numbers.

Second step - we solve the system of one-dimensional problems (3.7) as described in Section 2.2. This step is completed by applying the inverse Fourier transform in the y direction. The resulting particular solution $U^{(I)}(x, y)$ will be smooth in each horizontal band Ω_{nu} except for the interfaces $\bar{y}_n = nl_2, \; n = 1, ..., P_1$.

Third step - To remove discontinuities on the horizontal interfaces we apply again our one-dimensional matching routine. We take the vertical column $\Omega^\mu = \bigcup_{n=1}^{P_2} \omega_n^\mu, \; \mu = 1, ..., P_1$, transform U^I into the LFB (similar to the source function in the first step) and implement the matching on the lines $\bar{y}_n$ for each Fourier coefficient. After applying the inverse transform we get a vertically smooth solution $U^{(II)}$ in the physical space.

Smoothing in the y- direction induces discontinuities on the vertical lines $\bar{x}_n = nl_1, \; n = 1, ..., P_1$, because each Fourier coefficient is matched independently. These jumps can be removed, however, by another matching (in the x- direction). Each alternate "ironing" will result in a smoother solution.

Figs. 2 and 3 show the logarithm of the maximum numerical error as function of S (number of smoothing steps in both directions) for the solution $U = \cos k_x x \cos k_y y$ ($P = 4 \times 4, N = 128 \times 128$). The algorithm converges very fast and after several smoothing steps the maximum precision is achieved . The rate of convergence increases as the parameter λ in (3.1) and the wave number $k = \sqrt{k_1^2 + k_2^2}$ get larger.

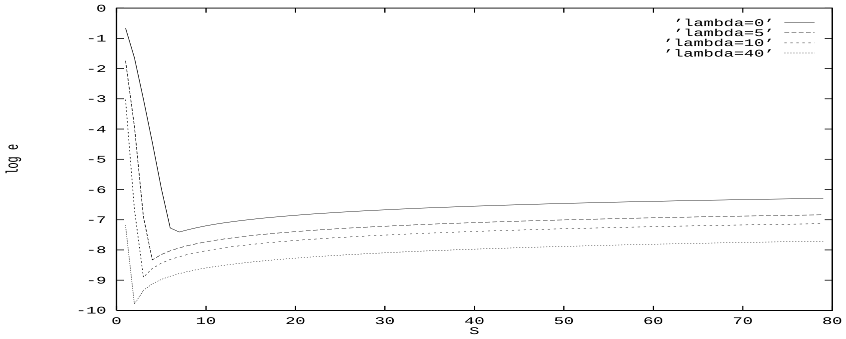

Figure 2: Convergence of alternate smoothing procedure for the solution $U_{ex} = \cos k_x x \cos k_y y$ at $k_1 = k_2 = 1$

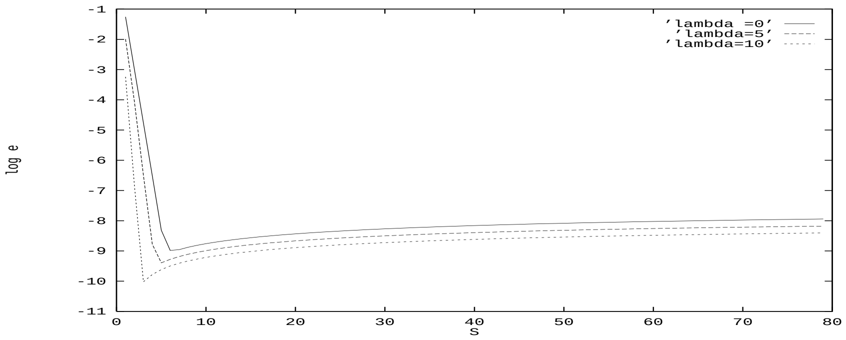

Figure 3: Convergence of alternate smoothing procedure for the solution $U_{ex} = \cos k_x x \cos k_y y \quad k_1 = k_2 = 4$

Figs. 4-7 show a succession of isolines for the first three smoothing steps ($\lambda = 0,\ k_1 = k_2 = 1$). Fig. 4 corresponds to a particular solution without any matching on the interfaces $(S = 0)$, the last figure (7) $(S = 2)$ is already indistinguishable from the exact one.

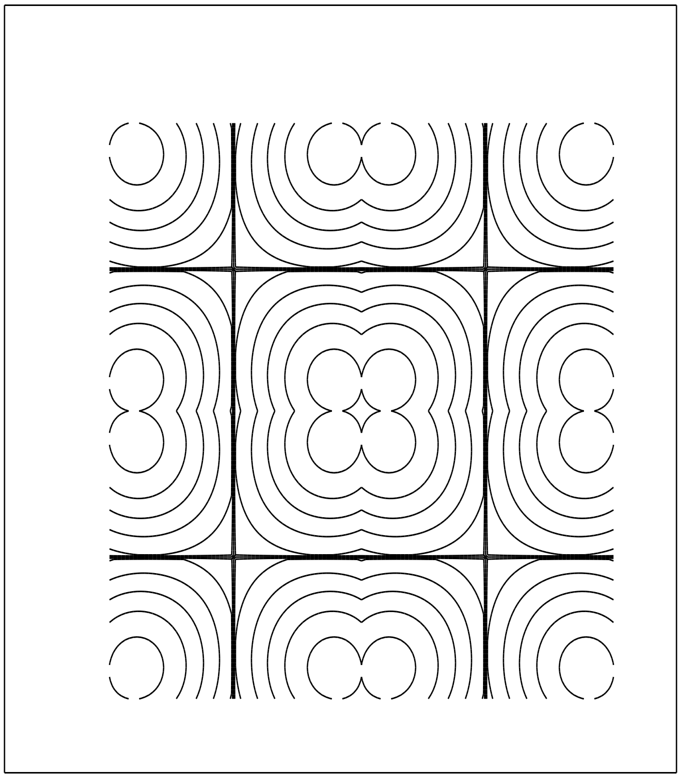

Figure 4: Isoline patterns of particular subsolutions in 4×4 domain region without matching ($S = 0$)

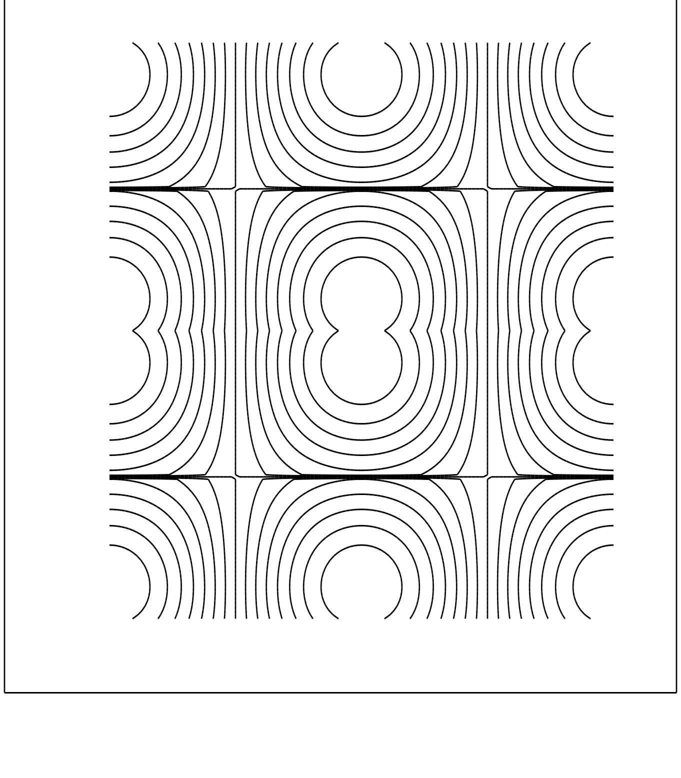

Figure 5: After one matching step in the x direction ($S = 1$)

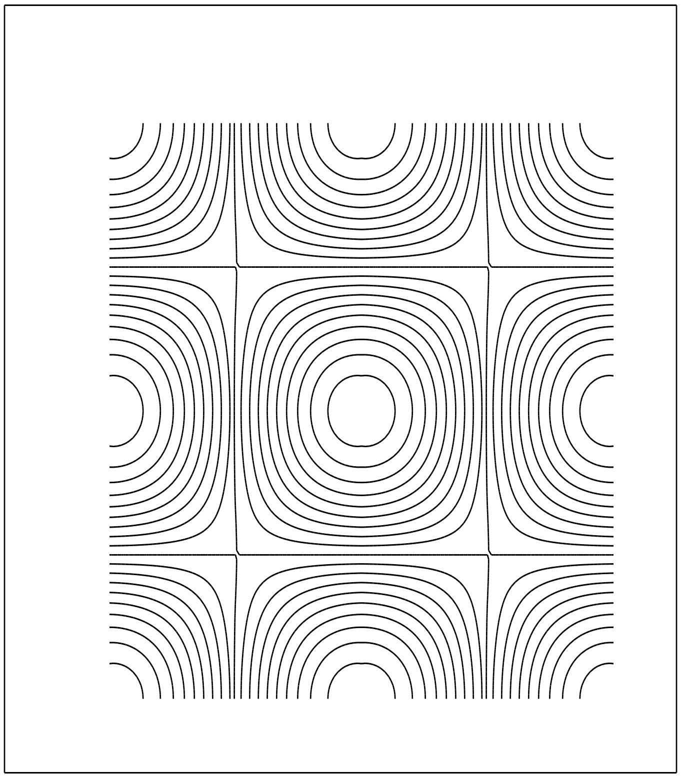

Figure 6: after two matching steps ($S = 2$)

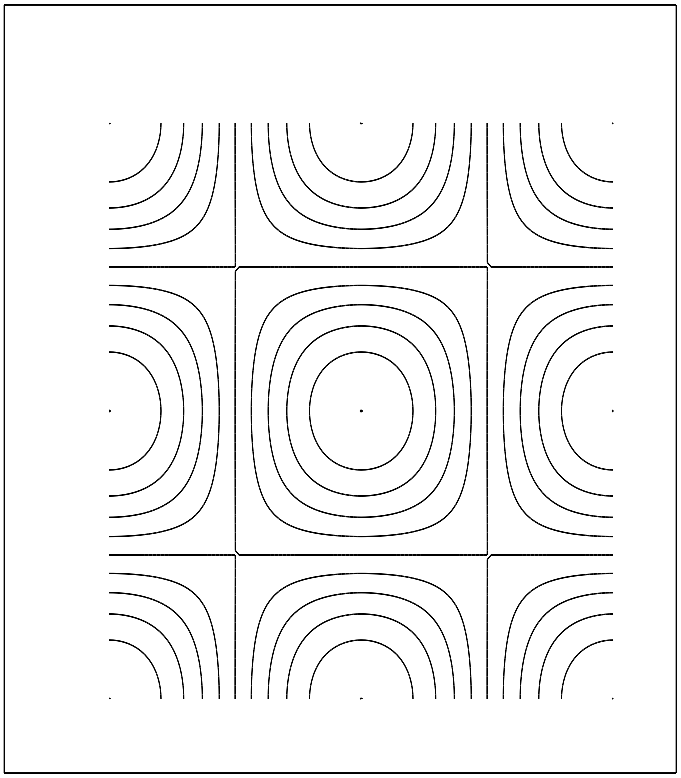

Figure 7: after three matching steps ($S = 3$)

3.3 Time-Dependent Problems

We consider time-dependent problem with diffusion type linear operator

$$\frac{\partial U}{\partial t} = \Delta U + B(U), \qquad in\ \Omega = (0, L_1) \times (0, L_2) \tag{4.1}$$

where $B(u)$ contains all non-linear and non-uniform terms. The implicit (or semi-implicit) time discretization procedure for this problem leads to the elliptic equation (2.1) at each

τ	$\lvert U^{II} - U_{ex}\rvert_{max}$	$\lvert U^{III} - U_{ex}\rvert_{max}$
0.1	8.60 (-2)	8.27 (-7)
0.05	3.70 (-2)	1.36 (-8)
0.01	8.63 (-4)	2.40 (-9)

Table 1: numerical error on the time asymptotic stage for several τ

time step. Then, we have a two-level scheme

$$\frac{U_{n+1} - U_n}{\tau} = \theta \Delta U_{n+1} + (1-\theta)\Delta U_n + B_n \tag{4.2}$$

where n refers to time $t_n = n\tau$ and $\theta > 0$ (the particular cases $\theta = 1/2$ and $\theta = 1$ correspond to the Crank-Nikolson and the Euler-forward schemes, respectively).

Setting $\lambda^2 = 1/\tau\theta$ we obtain

$$\Delta U_{n+1} - \lambda^2 U_{n+1} = \frac{\theta - 1}{\theta}\Delta U_n - \lambda^2 U_n - \frac{1}{\theta} B_n \tag{4.3}$$

Equation (4.3) can be simplified by defining

$$V^* = \theta U_{n+1} + (1-\theta) U_n \tag{4.4}$$

and we get

$$\Delta V^* - \lambda^2 V^* = -\lambda^2 U_n - \frac{1}{\theta} B_n \tag{4.5}$$

This equation is solved for V^* for the knowns U_n and $B_n(u_n)$, then, U_{n+1}, at a new time step, is determined by using (4.4).

As an example, we consider the two-dimensional heat equation

$$\frac{\partial U}{\partial t} = \Delta U + F(x, y) \tag{4.6}$$

with the sinusoidal forcing term. The domain Ω is decomposed into 4×4 square cells, number of collocation points $N_1 = N_2 = 128$, and the parameter of time discretization scheme is $\theta = 1/2$. The sufficient small time step τ corresponds to a larger parameter λ in (2.1), and according to conclusion in Section 3.2, only few smoothing iterations are required to gain maximum precision.

Table 1 gives the numerical error for the time asymptotic stage for several τ; the number S of smoothing iterations for each time step is either 2- first column or 3- second column (larger S provides no better accuracy.)

Conclusions

In this paper we propose an efficient parallel spectral algorithm for the solution of stationary (elliptic) time-dependent (parabolic) problems in multidomain regions. The domain is partitioned into rectangular cells. The time discretization is performed by using a semi-implicit time-stepping scheme (linear term is treated implicitly). The discretization in space is carried out by a pseudo-spectral method with the Local Fourier Basis (LFB).

The problem is solved concurrently in all subdomains, after that a direct matching procedure is employed to match all particular subsolutions on the interfaces. The implementation of the LFB technique enables us to split a 2-D problem, with the overall coupling of the interface unknowns, into a set of uncoupled 1-D differential equations with simple matching relations. The continuity of a global solution is achieved after several alternative matching steps in both directions. Thus, the numerical algorithm is reduced to repeated implementation of 1-D routine, which is very similar to the usual spectral Fourier method, but includes also some additional operations to smooth the source function and match the particular subsolutions.

The implementation of the LFB algorithm requires the extension of the function out of the subdomain in a continuous way, so that the current method can be applied only to problems with periodic boundary conditions. Nevertheless, this technique can be employed also for the solution of non-periodic problems by combination of the LFB method with other techniques to treat the extreme subdomains. This approach is reported in [5].

A Implementation of the Local Fourier Basis for Multidomain Problems

The Local Fourier Basis (LFB) was introduced in [2] and was modified in [1] to solve differential equations in multidomain regions. Here we briefly repeat on the basic LFB technique. The algorithm consists of smoothing a non-periodic function in each subdomain and application of the FFT on the local period. We illustrate the idea of this method on the 1-D equation

$$U'' - \lambda^2 U = f(x,t), \qquad x \in (0,L). \tag{A.1}$$

The interval (0,L) is partitioned into equal sub-intervals of the length $l = L/P$.

To perform a smooth projection on the range of a subdomain $x \in (a,b)$, we introduce a bell function $B(x)$, supported on an extended interval $x \in (a_1, b_1),\ a_1 < a < b < b_1$:

$$\begin{array}{ll} B^2(x) + B^2(2\bar{a} - x) = 1 & x \in (a_1, a) \\ B(x) = 1 & x \in (a,b) \\ B^2(x) + B^2(2\bar{b} - x) = 1 & x \in (b, b_1) \\ B(x) = 0 & x < a_1,\ x > b_1 \end{array} \tag{A.2}$$

where $\bar{a} = (a + a_1)/2,\ \bar{b} = (b + b_1)/2$.This function is equal to $B = 1$ inside the subdomain and decays smoothly outwards over a distance $2\epsilon = b_1 - b = a - a_1$. Some specific forms of $B(x)$ were tested in [1]

The projection $\tilde{f}$ of the function f appears as a "folding" across the lines $\bar{a}$ and $\bar{b}$:

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

(the "folded" function $\tilde{f}$ is defined on $[\bar{a}, \bar{b}]$; the second term is "switched on" only on the interval $x \in [a_1, a]$ the third term is "switched on" on the interval $x \in [b, b_1]$). Extra pieces of f on the intervals $[a_1, a]$ and $[b, b_1]$ are required in order to localize the function in a smooth way. They are provided by overlapping of neighboring subdomains over the range of 4ϵ. On the interval $x \in [a, b]$ we have $\tilde{f} = f$.

The projection procedure keeps the function $\tilde{f}$ highly continuous at $x = a, b$; it is also seen from (A.3), that in the vicinity of the points $x = \bar{a}$ and $x = \bar{b}$, the function $\tilde{f}(x)$ is odd and thus all its even derivatives $\tilde{f}^{(2s)}(\bar{a}) = \tilde{f}^{(2s)}(\bar{b}) = 0,\ s = 0, 1, ...$ After an antisymmetric reflection across the point $x = \bar{b}$ (or $x = \bar{a}$) we get a smooth periodic function $\tilde{F}$, which can be represented by a rapidly converging Fourier series. Now, we can apply the pseudo-spectral Fourier method to the problem (2.1) with the "smoothed" source function $\tilde{F}$ instead of f. The corresponding solution U_p will coincide with the solution of the original problem on the interval $x \in [a, b]$, where the projection procedure does not distort $f(x)$.

The accuracy of the projection procedure (2.3) depends on the number of collocation points N_ϵ on the interval of smoothing 2ϵ. If the density of the points is sufficiently large, a higher resolution can be obtained for a relatively small ratio ϵ/l. Thus, the relative amount of overhead due to overlapping of subdomains, needed for the smoothing, become insignificant as number of collocation points increases.

References

[1] M. Israeli, L. Vozovoi A. Averbuch, Spectral Multidomain Technique with Local Fourier Basis, *Journal of Scientific Computing*, Vol. 8, No. 2, pp. 135-149, June 1993.

[2] R. Coifman, Y. Meyer, Remarques sur l'analyse de Fourier à fenêtre, série, *C.R.Acad.Sci.* Paris **312** (1991), p. 259-261.

[3] D.Gottlieb, R.S. Hirsh, Parallel Pseudospectral Domain Decomposition Techniques, *Journal of Scientific Computing* **4**, No. 4, (1989), p. 309-325

[4] M.G.Macaraeg, C.L. Street, Improvements in spectral collocation discretization through a multidomain technique, *Appl. Numeric. Mathem.* **2**, (1986), p. 95-108

[5] L. Vozovoi, M. Israeli, A. Averbuch, Spectral Fourier Algorithm for Navier-Stokes Problems in Multidomain Regions, to appear.