26

Domain Decomposition with Local Fourier Bases applied to Frontal Polymerisation problems

M. Garbey , D. Tromeur-Dervout [1] [2]

Introduction

This paper describes a domain decomposition technique and high order accurate method that allows solving a nonlinear system of reaction diffusion equations coupled with Navier Stokes equations in a channel with local Fourier approximations. The present method is a generalization of the Israeli et al (1993) method which applies domain decomposition with local Fourier bases to the Helmoltz's problem. In the present work several new difficulties occur. First the problem is an unsteady and non linear problem which makes the periodic extension delicate to construct in terms of stability and accuracy. Secondly we use a stream function biharmonic formulation of the incompressible Navier Stokes equation in two space dimensions: the application of domain decomposition with local Fourier bases to a fourth order operator is more difficult to achieve than for a second order operator. This present work is a generalization of our previous work on combustion by the authors [GTD97] and the use of Fourier bases to solve non periodic problems [GTD98a][Gar97].

Our previous 2D numerical simulation of quasi-planar frontal polymerization (FP) used an adaptative domain decomposition method based on a piecewise C^1 Chebyshev polynomial approximation in the direction of propagation of the front and a Fourier approximation in a direction parallel to the front for periodic as well

1 This work was supported by the Région Rhône Alpes
2 CDCSP - University Lyon 1,69622 Villeurbanne France
{garbey,dtromeur}@cdcsp.univ-lyon1.fr, http://cdcsp.univ-lyon1.fr
Eleventh International Conference on Domain Decomposition Methods
Editors Choi-Hong Lai, Petter E. Bjørstad, Mark Cross and Olof B. Widlund

as nonperiodic boundary conditions. This adaptative method is very efficient and accurate but it requires a priori knowledge of the structure of the solution. We refer to [GTD97, GTD98b] for the parallel efficiency and the numerical accuracy of this method.

In this paper we use the same problem as a test case, but the methodology that we develop is based on a totally different philosophy. We do not use adaptivity and compute the solution with (local) pseudo-spectral Fourier approximations in such way that we can use a fast direct solver per subdomain and a large total number of discretization points efficiently on a parallel computer. In particular, by introducing a non-overlapping domain decomposition, we avoid the problem of global transposition of matrices induced by a two dimensional fast Fourier transform on a parallel computer. In addition, we use no a priori knowledge on the structure of the solution since there is no space adaptativity but possibly filtering to remove spurious oscillations. We expect the question of adaptativity to be resolved as a separate issue, using appropriate numerical generation of a 2D mapping applied to a regular grid structure [J.A98] [LMS98].

The outline of this article is as follows. Section 26 describes the governing equations of the model problem. Section 26 (Section 26) describes the algorithms for the stream function solution without domain decomposition (with domain decomposition). Section 26 deals with the numerical accuracy and stability of the method. Section 26 presents a numerical result obtained with the new methodology for a non-planar flame front. Our conclusions are given in Section 26.

Governing Equations

We consider a model that couples a reaction-diffusion system for the temperature T and the reactant concentration C, describing a one-step exothermic chemical process, with the incompressible Navier-Stokes equation using the Boussinesq approximation; it is worth noticing that we use the biharmonic formulation of Navier-Stokes with Ψ as the stream function. We refer to [GTD98b] for the precise statement of the problem and parameters, and we simply recall the basic equations

$$\partial T/\partial t + (\partial\Psi/\partial z)(\partial T/\partial x) - (\partial\Psi/\partial x)(\partial T/\partial z) = \Delta T + C\exp\frac{ZT}{1+\delta(1-T)},$$

$$\partial C/\partial t + (\partial\Psi/\partial z)(\partial C/\partial x) - (\partial\Psi/\partial x)(\partial C/\partial z) = \epsilon\Delta C - C\exp\frac{ZT}{1+\delta(1-T)},$$

$$\partial\Delta\Psi/\partial t = (\partial\Psi/\partial x)(\partial\Delta\Psi/\partial z) - (\partial\Psi/\partial z)(\partial\Delta\Psi/\partial x) + \Delta\Delta\Psi - RP\vec{g}.\vec{\nabla}T, \quad (1)$$

$$T\to 0,\ C\to 1,\ \Psi\to 0,\ as\ z\to-\infty,\ \ T\to 1,\ C\to 0,\ \Psi\to 0,\ as\ z\to+\infty$$

$$\Psi(0,z)=0,\ \partial\Psi/\partial x(0,z)=0,\ \Psi(L,z)=0, \partial\Psi/\partial x(L,z)=0, \forall z \quad (2)$$

$$\partial T/\partial x(0,z)=0,\ \partial T/\partial x(L,z)=0,\ \partial C/\partial x(0,z)=0,\ \partial C/\partial x(L,z)=0,\ \forall z$$

Many different possible nonlinear regimes of the solution exist. In addition, the systems have a well-known one dimensional traveling wave solution $(T_0(z), C_0(z), \Psi\equiv 0)$. The existence and stability of these solutions depend on the specific value of the bifurcation or control parameters, such as the Zeldovich number Z or the Rayleigh number R.

In addition Z is large, thus making the problem difficult to compute. We are mainly interested in using this problem as a nontrivial test case to demonstrate and validate the feasibility of our approach with local Fourier bases.

A First Algorithm without Domain Decomposition to Solve the NS Equation

We consider the following semi-implicit Euler scheme for the time discretisation of equation (1) as in [GTD98b]:

$$\left\{\begin{array}{l} \Delta\Psi^{n+1} - dt\Delta\Delta\Psi^{n+1} = dt\ ((\partial\Psi^n/\partial x)(\partial\Delta\Psi^n/\partial z) - (\partial\Psi^n/\partial z)(\partial\Delta\Psi^n/\partial x) \\ \qquad\qquad +\Delta\Psi^n - dt\ f(x,z,t_n), \quad (x,z)\in(0,L)\times(-H,H), \\ \Psi^{n+1}(0,z) = 0, \quad (\partial\Psi^{n+1}/\partial x)(0,z) = 0, \quad z\in(-H,H), \\ \Psi^{n+1}(L,z) = 0, \quad (\partial\Psi^{n+1}/\partial x)(L,z) = 0, \quad z\in(-H,H). \end{array}\right. \tag{3}$$

We look for a z-periodic solution of period H with H large. As a matter of fact Ψ vanishes exponentially when $|z|$ goes to infinity and if H is large enough for the boundary conditions to have no influence on the dynamic of the combustion wave. For simplicity of notation, we will assume in the following that $H = \pi$. f and Ψ at time t_n are approximated by the following discrete Fourier expansion:

$$f(x,z,t_n) \ = \ \Sigma_{k=\frac{-N_z}{2},\frac{N_z}{2}-1}\hat{f}_k^n(x)\ e^{iky},\ \Psi^n(x,z,t_n) \ = \ \Sigma_{k=\frac{-N_z}{2},\frac{N_z}{2}-1}\hat{\Psi}_k^n(x)\ e^{iky},$$

with $y = \frac{(z+H)\,\pi}{H}$. Let $F^n(x,z)$ be the right hand side of equation (3). From the approximation,

$$\begin{array}{c} (\partial\Psi^n/\partial x)(\partial\Delta\Psi^n/\partial y) - (\partial\Psi^n/\partial y)(\partial\Delta\Psi^n\partial x) \\ \approx i\Sigma_{k=\frac{-N_z}{2},\frac{N_z}{2}-1}[e^{iky}\Sigma_{k_1+k_2=k}(\hat{\Psi^n}'_{k_1}\ k_2(\hat{\Psi^n}''_{k_2} - k_2^2\hat{\Psi^n}_{k_2}) - \hat{\Psi^n}_{k_1}\ k_1(\hat{\Psi^n}'''_{k_2} - k_2^2\hat{\Psi^n}'_{k_2}))] \quad , \end{array}$$

we obtain the discrete Fourier expansion of F^n. We then have to solve the uncoupled fourth order N_z ODE's at each time step. For $k = 0$, we have:

$$\begin{array}{r} L_0[\hat{\Psi}_0^{n+1}] = (\hat{\Psi}_0^{n+1})'' \ - \ dt\ (\hat{\Psi}_0^{n+1})'''' = \hat{F}_0^n,\ x\in(0,L), \\ \hat{\Psi}_0^{n+1}(0) = (\hat{\Psi}_0^{n+1})'(0) = \hat{\Psi}_0^{n+1}(L) = (\hat{\Psi}_0^{n+1})'(L) = 0, \end{array} \tag{4}$$

and for $k \neq 0$:

$$\begin{array}{r} L_k[\hat{\Psi}_k^{n+1}] = (\hat{\Psi}_k^{n+1})'' - k^2\hat{\Psi}_k^{n+1} - dt\ ((\hat{\Psi}_k^{n+1})'''' \\ -2k^2(\hat{\Psi}_k^{n+1})'' + k^4\hat{\Psi}_k^{n+1}) = \hat{F}_k^n,\ x\in(0,L), \\ \hat{\Psi}_k^{n+1}(0) = (\hat{\Psi}_k^{n+1})'(0) = \hat{\Psi}_k^{n+1}(L) = (\hat{\Psi}_k^{n+1})'(L) = 0. \end{array} \tag{5}$$

In order to compute $\hat{\Psi}_k^{n+1}(x)$ with a Fourier transform in the x direction as well as in the z direction, we use the technique as in [GTD98b] (see also [IVA93] [IVA94a] [IVA94b]) based on the superposition principle and the construction of a smooth periodic extension of the right hand sides. We split the unknowns as follows:

$$\hat{\Psi_k}^{n+1}(x) = \hat{\Psi_k}_F^{n+1}(x) + \hat{\Psi_k}_C^{n+1}(x), x\in[0,L] \tag{6}$$

This splitting is efficient because one can use a fast Fourier transform to compute $\hat{\Psi}_{k\,F}^{\,n+1}$ and analytical formulae for $\hat{\Psi}_{k\,C}^{\,n+1}$. In what follows, we will omit subscript k and $n+1$ to simplify the notations. Let $d > 0, d \in \mathbb{R}$, and let $[0, L+d]$ be an extension of the domain $(0, L)$. Let $\bar{F}_k(x)$ be a smooth periodic extension of $\hat{F}_k(x)$ on the interval $[0, L+d]$. We look for the $L+d$ periodic solution of the nonhomogeneous problem

$$L_k[\hat{\Psi}_F] = \bar{F},\ x \in (0, L+d),\ \hat{\Psi}_F\ L+d\ periodic,\ \hat{\Psi}_F\ \in\ C^{n+4}(\mathbb{R}) \tag{7}$$

Then we retrieve the homogeneous Neuman and Dirichlet boundary conditions satisfied by $\hat{\Psi}_k^{n+1}$ by computing the corrector term Ψ_C. These corrector terms satisfy the ODE problems

$$\begin{cases} L_k[\hat{\Psi}_C] = 0,\ x \in (0, L), \\ \hat{\Psi}_C(0) = -\hat{\Psi}_F(0),\ (\hat{\Psi}_C)'(0) = -(\hat{\Psi}_F)'(0),\ \hat{\Psi}_C(L) = -\hat{\Psi}_F(L),\ (\hat{\Psi}_C)'(L) = -(\hat{\Psi}_F)'(L) \end{cases}$$

Since the operators L_k are fourth order linear operators with constant coefficients, one can compute the basis functions for the four dimensional vector space of the solutions once and for all. The solution is written as $\hat{\Psi}_{kC} = \alpha_k v_k(x) + \beta_k w_k(x) + \gamma_k r_k(x) + \delta_k s_k(x)$, and the bases functions are explicitely given with formulae, for $k = 0$:

$$v_0(x) = exp(-\frac{1}{\sqrt{dt}}(x-0)),\ w_0(x) = exp(-(\frac{1}{\sqrt{dt}}(L-x)),\ r_0(x) = x,\ s_0(x) = (L-x), \tag{8}$$

and for $k \neq 0$:

$$\begin{aligned} v_k(x) &= exp(-\sqrt{k^2 + \frac{1}{dt}}\,(x-0)) \qquad & w_k(x) &= exp(-\sqrt{k^2 + \frac{1}{dt}}\,(L-x)) \\ r_k(x) &= exp(-|k|\,(x-0)) \qquad & s_k(x) &= exp(-|k|\,(L-x)) \end{aligned}$$

The coefficients $(\alpha_k, \beta_k, \gamma_k, \delta_k)$ are the solution of four-by-four linear systems that are solved at each time step. The boundary conditions on the left and on the right are numerically decoupled when the wave number k is large enough. The derivatives of $\hat{\Psi}_{kC}$ are readily computed from the previous formula. The derivatives of $\hat{\Psi}_F^k$ follow from its discrete Fourier expansion. We assemble the right hand side of equation (3) at each time step using the splitting (6) for the derivatives as well.

Our approach differs from that of Israeli et al in the way we compute a sufficiently regular periodic extension of the right hand sides. Let us recall that the smoothness of this extension is the essential limitation on the spectral accuracy of the method. More precisely, if the right hand side has regularity C^q, the numerical scheme is of order $q+4$ at most.

When the right hand-side is a given analytical function that can be defined for the interval $[0, L+d]$, one could use a so-called bell function B (as in Israeli et al) that is equal to one in [O,L] and zero in the vicinity of $L + d/2$. B times the right-hand side is then a smooth periodic function of period $L+d$. [IVA94b] [IVA94a] show evidence of the accuracy of this method for the Laplace equation or the Helmoltz problem. However, in our computation the right hand side is given *numerically* at each time step *only within* the physical domain of computation. We therefore use a numerical procedure to derive a smooth extension of this function. We proceed as follows.

We consider the exact or computed values of the derivatives of $\bar{F}_k$ up to order q at $x = 0$ and $x = L$. The classic Hermite interpolation allows us to define a polynomial function P on $[L, L+d]$ of degree $2q+1$ that interpolates function $\bar{F}_k$ with the following conditions:

$$\text{for } j = 0, ..., q,\ P^{(j)}(L) = \hat{F_k}^{(j)}(L),\ P^{(j)}(L+d) = \hat{F_k}^{(j)}(0).$$

The extended right hand side is then

$$\bar{F}_k = \begin{cases} \hat{F}_k,\ \forall x \in [0, L] \\ P(x),\ \forall x \in]L, L+d[. \end{cases} \tag{9}$$

In practice we require a C^2 continuity condition at the end points $x = 0$ and $x = L+d$ and the derivatives are computed by using sixth order one sided finite differences.

Domain Decomposition Algorithm

The next step is to introduce a domain decomposition in the x direction for each k-mode equation (1). We split $[0, L]$ into nd nonoverlapping subdomains of equal sizes, which we denote generically as $(0, l)$.

For each subdomain we apply the same splitting of the unknowns as described above, and in addition we impose C^3 continuity on the solution at the artificial interfaces. We compute the extension of the right hand sides for each subdomain and its corresponding periodic solution with a local Fourier discrete approximation. This part of the algorithm is exactly what we had for the single domain case, but it is applied for each subdomain in parallel.

Note that for mode 0, the operator with a periodic boundary condition is singular. The solution is then defined up to a shift. However the corrector term substracts this shift and the superposition principle gives the unique solution of (4).

We compute the corrector term for each subdomain in order to retrieve the C^3 continuity of the solution at the artificial interfaces. Let us denote by $(v_k^j, w_k^j, r_k^j, s_k^j)$ the set of basis functions for the corrector in each subdomain j and by $(\alpha_k^j, \beta_k^j, \gamma_k^j, \delta_k^j)$ the corresponding coefficients; in the local coordinate system of the subdomain, the basis functions are identical to the monodomain basis function. With two subdomains for example, the matrix of the interface problem for the unknown coefficient vector $\vec{T} = \left(\alpha_k^1, \beta_k^1, \gamma_k^1, \delta_k^1, \alpha_k^2, \beta_k^2, \gamma_k^2, \delta_k^2\right)^T$ writes

$$\begin{pmatrix}
v\,{}^1_k(0) & w\,{}^1_k(0) & r\,{}^1_k(0) & s\,{}^1_k(0) & 0 & 0 & 0 & 0 \\
v'\,{}^1_k(0) & w'\,{}^1_k(0) & r'\,{}^1_k(0) & s'\,{}^1_k(0) & 0 & 0 & 0 & 0 \\
v\,{}^1_k(l) & w\,{}^1_k(l) & r\,{}^1_k(l) & s\,{}^1_k(l) & -v\,{}^2_k(0) & -w\,{}^2_k(0) & -r\,{}^2_k(0) & -s\,{}^2_k(0) \\
v'\,{}^1_k(l) & w'\,{}^1_k(l) & r'\,{}^1_k(l) & s'\,{}^1_k(l) & -v'\,{}^2_k(0) & -w'\,{}^2_k(0) & -r'\,{}^2_k(0) & -s'\,{}^2_k(0) \\
v''\,{}^1_k(l) & w''\,{}^1_k(l) & r''\,{}^1_k(l) & s''\,{}^1_k(l) & -v''\,{}^2_k(0) & -w''\,{}^2_k(0) & -r''\,{}^2_k(0) & -s''\,{}^2_k(0) \\
v'''\,{}^1_k(l) & w'''\,{}^1_k(l) & r'''\,{}^1_k(l) & s'''\,{}^1_k(l) & -v'''\,{}^2_k(0) & -w'''\,{}^2_k(0) & -r'''\,{}^2_k(0) & -s'''\,{}^2_k(0) \\
0 & 0 & 0 & 0 & v\,{}^2_k(l) & w\,{}^2_k(l) & r\,{}^2_k(l) & s\,{}^2_k(l) \\
0 & 0 & 0 & 0 & v'\,{}^2_k(l) & w'\,{}^2_k(l) & r'\,{}^2_k(l) & s'\,{}^2_k(l)
\end{pmatrix}$$

Note that for large wave number k, the local interface problems are fully decoupled because of the exponential decay of the basis function. In such a case, the domain decomposition algorithm only requires local communication between node processing adjacent domains.

Numerical Accuracy of the Domain Decomposition Method

We now consider the accuracy of the domain decomposition with respect to the number of Fourier modes and the number of subdomains. It is important to notice that problems (4, 5) are singular perturbation problems when the time step dt is small and/or the wave number k is large. The sensitiviy of the method with respect to dt and k will also be tested. In order to test the numerical accuracy of our domain decomposition method, we consider each fourth order wave equation analogous to (5) with k replaced by $k\frac{\pi}{H}$ separately and choose the right hand side $\hat{F}_k^n$ such that the fourth order polynomial $x^2 * (x - 4\pi)^2/(16 * \pi^4)$ is the exact steady solution. The size of the domain of computation is given by $H = 25$ and $L = 2\pi$. In the following, we use the time marching scheme starting from the trivial initial condition until convergence to the steady solution is reached. We measure the difference between the converged numerical solution and the exact steady solution in maximum norm.

Table 1 gives the error in a maximum norm for the computed solution corresponding to the wave number $k = 0$ and $k = 1$ with a space step of order 0.05 and several numbers of subdomains. The total number of discretisation points in a physical domain is $2N_x$. The number of discretization points used for the extension is $2N_r$ for each subdomain. The number of Fourier modes per subdomain depends on the number of domains n_d and it is $N_x/n_d + N_r$. The time step is $dt = 0.1$

We see in all cases that $N_r = 16$ gives better results than $N_r = 8$. More precisely the accuracy depends strongly on the size of the extension only for large number of subdomains. This means that if the number of Fourier modes per subdomain is large enough the accuracy is not so sensitive to the size of the extension. The results of this table for mode $k = 1$ are always better than the corresponding results for the zero mode no matter what the shift is.

In particular, the error decreases with the number of subdomains for mode one. We observe the inverse for the zero mode except when $N_x = 256$ and $N_r = 16$, which is when the number of Fourier modes per subdomain is large.

Let us now consider four subdomains and study the sensitivity of the method to the time step, wave number k and the size of the extension per subdomain. Table 2 gives the error in maximum norm depending on the percentage of the extension i.e $100 \times d/(L + d)$ and the number of Fourier modes N_y for different couples (k, dt) of time steps and wave numbers. N_y is the number of Fourier modes per subdomain *including* the extension.

First we observe that the accuracy of the method deteriorates when the time step goes to zero: this is no surprise since the problem becomes increasingly singular as dt goes to zero. Eventually the time dependant schemes diverge for the zero mode equation if the space step is larger than the boundary layer thickness $\sqrt{dt}$ or if the extension is not large enough. The comparison between results for $k = 1$ and $k = 10$ or 30 with the same time step shows that accuracy deteriorates when k grows. The thickness of the boundary layer of the wave equation is $\sqrt{\frac{1}{k^2} + dt}$ and this phenomenon can be interpreted as above. However the amplitude of the solution Ψ_k should decrease with the wave number k in two dimensional problems and this phenomenon should not significantly affect the accuracy of the overall solution.

In all cases it is also significantly better to take a large extension of the subdomain

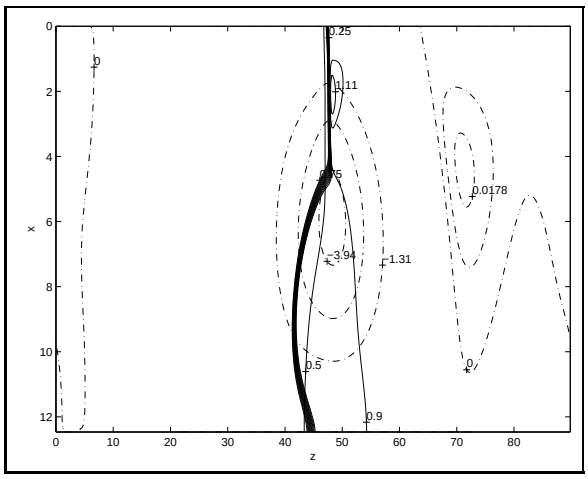

Figure 1 Superposition of the stream function contour lines to the hot spot formation; localization of the front is given by the thick line

rather than a small one. We speculate that small extensions lead to stability problems of the time marching scheme.

The method is at least of order 5 for mode 0 and for $d/(L+d) = 50\%$ or 25%. The order of the method for the higher modes 0 is less dependant on the value $d/(L+d)$, but differs with the mode value: (order 3.2 for k=30, order 4.3 for k=10). Nevertheless the decrease of the order of convergence is compensated for by the fact that the module of the Fourier coefficients decreases steeply with increasing mode value.

Results on FP process in Liquid (i.e R=0.5)

In order to solve our model problem in Frontal polymerization given in Section 26, we have combined the domain decomposition algorithm given above for the Navier Stokes equation with the analogous algorithm for a second order system of reaction diffusions describing the exothermic one-step chemical process [GTD98a]. We refer to that paper for preliminary results on the efficiency of our parallel algorithm. A detailed description of the implementation and parallel efficiency of the domain decomposition will be reported elsewhere. We have applied this methodology to compute non-planar flame front.

Figure 1 shows the effect of hydrodynamics on the structure of the flame front when the channel is horizontal. Thin solid lines represent the temperature isovalues while dash lines represent the stream function isovalues. The thick solid line represents the location of the front, centered on the level set $C = 0.5$. The computation was done with a total of $N_x = 112$ by $M_z = 256$ modes on a physical domain of size $[0, 4\pi] \times [0, 90]$. The Zeldovich number is equal to 6., the Rayleigh number is equal to 1.5, the Prandlt number is $P = 1$, the mass diffusion is given by $\epsilon = 0.02$, and the time step is set to $dt = 0.01$. This solution is a travelling wave moving toward the left with a hot spot

close to the top wall of the horizontal channel. Gravity is vertical. The location of the hot spot and the front curvature of the concentration profile are closely related to the circular motion of the flow.

Conclusions

We have developed and implemented a domain decomposition methodology based on local Fourier approximations and the superposition principle to solve incompressible Navier Stokes equation in two space dimensions.

This approach has been generalized to solve a frontal polymerisation model. It allows us to tackle complex two D nonlinear regimes without any a priori hypothesis on the structure of the flame front.

REFERENCES

[Gar97] Garbey M. (1997) On some applications of the superposition principle with (local) fourier basis. *Preprint CDCSP-97-02* to appear in SIAM J. of Scientific Computing.

[GTD97] Garbey M. and Tromeur-Dervout D. (1997) Massively parallel computation of stiff propagating fronts. *Combustion Theory and Modelling* 1(3): 271–294.

[GTD98a] Garbey M. and Tromeur-Dervout D. (1998) Application of local fourier basis to domain decomposition in combustion problems. In Ecer A. and al (eds) *Proc. Parallel CFD98*. North-Holland, Taiwan. to appear.

[GTD98b] Garbey M. and Tromeur-Dervout D. (1998) A new parallel solver for non periodic incompressible navier-stokes equations with fourier basis: Application to frontal polymerisation. *J. Comp. Phys.* 145(1): 316–331.

[IVA93] Israeli M., Vozovoi L., and Averbuch A. (1993) Spectral multidomain technique with local fourier basis. *J. of Scientific Computing* 8(2): 135–149.

[IVA94a] Israeli M., Vozovoi L., and Averbuch A. (1994) Domain decomposition methods with local fourier basis for parabolic problems. *Comtemporary Mathematics* 157.

[IVA94b] Israeli M., Vozovoi L., and Averbuch A. (1994) Spectral multidomain technique with local fourier basis. *J. of Scientific Computing* 9(3): 311–325.

[J.A98] J.A.Mackenzie (1998) The efficient generation of simple two-dimensionnal adaptive grids. *SIAM J. Scientific Compting* 19(4): 1340–1365.

[LMS98] L.S. Mulholland W. H. and Sloan D. (1998) Pseudospectral solution of near-singular problems using numerical coordinate transformations based on adaptivity. *SIAM J. Scientific Compting* 19(4): 1261–1289.

Mode 0	number of subdomains				Mode 1	number of subdomains			
N_x=256	2	4	8	16	N_x=256	2	4	8	16
N_r=16	4.6e-5	1.9e-5	1.3e-5	9.3e-6	N_r=16	4.2e-7	2.6e-6	8.8e-7	4.2e-7
N_r=8	4.6e-5	4.6e-4	7.7e-4	1.4e-3	N_r=8	1.2e-5	5.5e-6	2.6e-6	1.3e-6
N_x=128	2	4	8	16	N_x=128	2	4	8	16
N_r=16	7.7e-5	8.8e-5	1.1e-4	1.3e-4	N_r=16	1.8e-5	1.0e-5	4.5e-6	2.4e-6
N_r=8	1.6e-4	6.3e-4	1.2e-3	2.0e-3	N_r=8	7.9e-5	3.9e-5	1.8e-5	9.9e-6

Table 1 accuracy with respect to the number of subdomains

$k=0$, $dt=0.1$					$k=1$, $dt=0.1$			
N_x	50%	25%	12.5%	6.25%	50%	25%	12.5%	6.25%
16	7.55e-3	4.33e-1	div	5.39e-1	1.65e-4	8.59e-4	5.80e-3	9.70e-3
32	8.42e-5	7.62e-3	4.05e-1	div	6.73e-6	3.47e-5	3.02e-4	2.46e-3
64	2.52e-6	1.56e-5	7.04e-3	3.94e-1	2.44e-7	1.02e-6	1.07e-5	1.20e-4
128	6.00e-7	1.24e-6	4.77e-6	6.74e-3	7.75e-9	2.77e-8	3.41e-7	4.46e-6
$k=0$, $dt=0.01$					$k=1$, $dt=0.01$			
N_x	50%	25%	12.5%	6.25%	50%	25%	12.5%	6.25%
16	3.75e-2	1.22e-0	div	div	3.30e-4	1.73e-3	1.13e-2	1.98e-2
32	7.90e-4	7.49e-2	8.37e-1	div	1.48e-5	7.57e-5	6.51e-4	5.31e-3
64	2.45e-5	1.54e-4	7.17e-2	8.46e-1	5.53e-7	2.27e-6	2.38e-5	2.66e-4
128	6.05e-7	4.77e-6	5.38e-5	6.90e-2	1.73e-8	6.14e-8	7.58e-7	9.91e-6
$k=0$, $dt=0.001$					$k=1$, $dt=0.001$			
N_x	50%	25%	12.5%	6.25%	50%	25%	12.5%	6.25%
16	div	div	div	div	div	7.27e-3	div	div
32	3.34e-3	6.61e-1	div	div	7.10e-5	4.11e-4	3.57e-3	2.87e-2
64	1.67e-4	1.42e-3	div	div	3.67e-6	1.39e-5	1.48e-4	1.65e-3
128	5.26e-6	4.48e-5	5.30e-4	div	1.14e-7	3.86e-7	4.85e-6	6.38e-5
$k=10$, $dt=0.1$					$k=30$, $dt=0.1$			
N_x	50%	25%	12.5%	6.25%	50%	25%	12.5%	6.25%
16	2.92e-4	1.72e-3	8.37e-3	2.09e-2	2.89e-4	1.66e-3	7.06e-3	2.20e-2
32	3.74e-5	2.29e-4	1.46e-3	8.74e-3	3.99e-5	2.54e-4	1.55e-3	7.52e-3
64	3.79e-6	2.05e-5	1.38e-4	1.04e-3	5.02e-6	3.23e-5	2.18e-4	1.45e-3
128	2.56e-7	1.05e-6	7.02e-6	6.06e-5	5.57e-7	3.26e-6	2.19e-5	1.62e-4
$k=10$, $dt=0.01$					$k=30$, $dt=0.01$			
N_x	50%	25%	12.5%	6.25%	50%	25%	12.5%	6.25%
16	2.96e-4	1.74e-3	8.45e-3	2.10e-2	2.90e-4	1.66e-3	7.07e-3	2.20e-2
32	3.79e-5	2.32e-4	1.48e-3	8.82e-3	3.99e-5	2.55e-4	1.55e-3	7.53e-3
64	3.84e-6	2.07e-5	1.40e-4	1.05e-3	5.02e-6	3.23e-5	2.18e-4	1.45e-3
128	2.59e-7	1.07e-6	7.12e-6	6.15e-5	5.57e-7	3.26e-6	2.20e-5	1.63e-4
$k=10$, $dt=0.001$					$k=30$, $dt=0.001$			
N_x	50%	25%	12.5%	6.25%	50%	25%	12.5%	6.25%
16	3.32e-4	1.94e-3	9.22e-3	2.23e-2	2.94e-4	1.69e-3	7.15e-3	2.22e-2
32	4.23e-5	2.58e-4	1.64e-3	9.60e-3	4.05e-5	2.58e-4	1.57e-3	7.61e-3
64	4.29e-6	2.33e-5	1.57e-4	1.18e-3	5.09e-6	3.28e-5	2.21e-4	1.47e-3
128	2.93e-7	1.22e-6	8.12e-6	6.97e-5	5.65e-7	3.31e-6	2.23e-5	1.65e-4

Table 2 Sensitivity of the methods to parameters k and dt