Journal of Computational Physics **173,** 575–599 (2001)
doi:10.1006/jcph.2001.6895, available online at http://www.idealibrary.com on IDEAL®

Parallel Algorithms with Local Fourier Basis[1]

Marc Garbey and Damien Tromeur-Dervout

Center for the Development of Scientific Parallel Computing, CDCSP/ISTIL Bd Latarjet, Université Lyon 1, 69622 Villeurbanne cedex, France
E-mail: garbey@cdcsp.univ-lyon1.fr; dtromeur@cdcsp.univ-lyon1.fr.

Received July 26, 2000; revised July 18, 2001

We present a nonoverlapping domain decomposition method with local Fourier basis applied to a model problem in liquid flames. The introduction of domain decomposition techniques in this paper is for numerical and parallel efficiency purposes when one requires a large number of grid points to catch complex structures. We obtain then a high-order accurate domain decomposition method that allows us to generalize our previous work on the use of local Fourier basis to solve combustion problems with nonperiodic boundary conditions (M. Garbey and D. Tromeur-Dervout, *J. Comput. Phys.* **145**, 316 (1998)). Local Fourier basis methodology fully uses the superposition principle to split the searched solution in a numerically computed part and an analytically computed part. Our present methodology generalizes the Israeli *et al.* (1993, *J. Sci. Comput.* 8, 135) method, which applies domain decomposition with local Fourier basis to the Helmholtz's problem. In the present work, several new difficulties occur. First, the problem is unsteady and nonlinear, which makes the periodic extension delicate to construct in terms of stability and accuracy. Second, we use a streamfunction biharmonic formulation of the incompressible Navier–Stokes equation in two space dimensions: The application of domain decomposition with local Fourier basis to a fourth-order operator is more difficult to achieve than for a second-order operator. A systematic investigation of the influence of the method's parameters on the accuracy is done. A detail parallel MIMD implementation is given. We give an a priori estimate that allows the relaxation of the communication between processors for the interface problem treatment. Results on nonquasi-planar complex frontal polymerization illustrate the capability of the method.

Key Words: domain decomposition, Fourier expansions, combustion, convective instability, parallelism.

[1] This work was supported by Région Rhônes Alpes.

0021-9991/01 $35.00

1. INTRODUCTION

This paper is devoted to a numerical methodology that uses fully the superposition principle and domain decomposition techniques with local Fourier basis. A simulation of a frontal polymerization (FP) with complex dynamics illustrates the capabilities of the method.

The main restriction that led us to develop this methodology is that our previous 2D numerical simulation was limited to quasi-planar frontal polymerizations (FP). In [7, 8] we used an adaptive domain decomposition method based on a piecewise C^1 Chebyshev polynomial approximation in the direction of propagation of the front (z-direction) and on a Fourier approximation in the direction parallel to the front (x-direction) for periodic as well as nonperiodic boundary conditions. Some 1D mapping of the Chebychev collocation points in the z-direction on each subdomain allows us to concentrate the points near the front location. Because the front was assumed quasi-planar, the mapping was independent of the x-variable, and the code simplicity follows. If the hypothesis of the quasi-planar structure of the front is not valid, we have to increase the number of subdomains in the z-direction in order to have enough points and subdomains to compute accurately the front along the x-direction, and we need to increase considerably the number of points in the x-direction where no domain decomposition was introduced. The disadvantages of this option are (1) the use of a great number of points at some part of the computation where they are not really needed, and (2) the loss of the parallelism efficiency of the methodology for large number of strip subdomains [7].

In this paper, we use the same FP problem as a test case, but the methodology that we develop follows a totally different philosophy in keeping the simplicity of the operators and the algorithm. We use no adaptivity, and we compute the solution with (local) pseudo-spectral Fourier approximations in such way that we can use efficiently on a parallel computer a fast direct solver per subdomain and a large total number of discretization points. Therefore, no a priori knowledge of the structure of the solution is required. We expect that the question of adaptivity for complex combustion front problem could be resolved as a separate issue using an appropriate numerical generation of a 2D mapping applied to a regular grid structure [16, 18], or with fictitious domain decomposition techniques as in [11]. Some examples of solutions of simple PDE problems computed with local Fourier basis on complex 2D geometries can be found in [6].

The new methodology can be summarized as follows. Let us consider a model boundary value problem written symbolically,

$$\frac{\partial U}{\partial t}(x,t) + A[U](x,t) = F(U,x,t), \quad \forall x \in \Omega \subset (O,2\pi)^q, \quad B[U(x,t)] = 0 \text{ on } \partial\Omega,$$

where q is a positive integer, A and B are linear differential operators, and $\tilde{F}$ a nonlinear operator. We consider a semi-implicit multistep method for time integration as

$$U^{n+1} + A[U^{n+1}] = \tilde{F}(U^n \cdots U^{n-p})$$

in domain Ω.

Our solution procedure to obtain U^{n+1} consists of (1) splitting the unknown U^{n+1} into a periodic solution, denoted $U^{[F],n+1}$, of the inhomogeneous extended problem

$$U^{[F],n+1} + A\left[U^{[F],n+1}\right] = \bar{F}(U^n \cdots U^{n-p}), \quad \forall x \in (O,2\pi)^q,$$

where $\bar{F}$ is a smooth periodic extension of $\tilde{F}$ that must be defined properly, and (2) a so-called corrector, denoted $U^{[C],n+1}$, solution of the following homogeneous problem:

$$U^{[C],n+1} + A\left[U^{[C],n+1}\right] = 0, \quad \forall x \in \Omega, \quad B\left[U^{[C],n+1}\right] = -B\left[U^{[F],n+1}\right] \text{ on } \partial\Omega.$$

This solution procedure is efficient if one can use fast direct solvers for $U^{[F],n+1}$ and an analytical or easy-to-compute approximation of the corrector $U^{[C],n+1}$ [3, 4, 6, 8, 19].

We further develop a nonoverlapping domain decomposition based on the same concept. Each subdomain is solved with a similar splitting, but the correctors have to solve the artificial boundary condition at the interfaces. We refer to [13–15] for pioneer work on domain decomposition with local Fourier basis applied to the Helmholtz's problem. However, new difficulties occur in the present application because the problem is unsteady and nonlinear, which makes the periodic extension delicate to construct in terms of stability and accuracy. In addition, we use a streamfunction biharmonic formulation of the incompressible Navier–Stokes equation in two space dimensions: The application of domain decomposition with local Fourier basis to a fourth-order operator is more difficult to achieve than that for a second-order operator. The main advantage of the domain decomposition with local Fourier basis, from parallel implementation point of view, is to avoid the global transposition of matrices that appears in parallel fast Fourier transform [5], because of the localization of the data of the Fourier transform on each processor. Moreover, we show how to decouple adaptively, with respect to the Fourier mode values, the dependencies between subdomains of the domain decomposition.

The plan of this paper is as follows. Section 2 describes the governing equations of the frontal polymerization model problem with arbitrary gravity direction. Section 3 describes the superposition principle used and the algorithms that follows for the reaction diffusion system and the Navier–Stokes equation. In Section 4, domain decompositions with local Fourier basis are introduced to extend the methodology for parallel efficiency purpose. The accuracy of the methodology and the influence of its parameters, such as the time steps, the number of subdomains, and the length of the extension, are studied. Section 5 details parallel MIMD algorithm implementation on a Digital Tru Cluster and discusses the parallel efficiency and scalability of the method. Section 6 presents numerical results obtained with the new methodology for FP with a no-quasi-planar front structure. Conclusions on the advantages and limitations of the methodology are given in Section 7.

2. GOVERNING EQUATIONS

We consider the propagation of a reaction front in a liquid phase [20] with a simple chemical reaction mechanism: The reactant A is converted to the final product B. This model includes equations for the temperature and the concentration for the one-step chemical reaction, and the Navier–Stokes (NS) equations written in the Boussinesq approximation. The conservation laws lead to the following equations:

$$\frac{\partial T}{\partial t} + V \cdot \nabla T = \kappa \Delta T + w(T, C) \tag{1}$$

$$\frac{\partial C}{\partial t} + V \cdot \nabla C = \mu \Delta C - w(T, C) \tag{2}$$

$$\frac{\partial V}{\partial t} + V \cdot \nabla V = -\frac{1}{\rho}\nabla p + \nu \Delta V + g\beta(T - T_0)\gamma \tag{3}$$

$$\nabla \cdot V = 0. \tag{4}$$

Here T is the temperature, C the concentration of the reactant A, V the velocity of the medium, p the pressure, κ the coefficient of thermal diffusion, q the adiabatic heat release, ρ the density, ν the viscosity, μ the mass diffusion, g the acceleration of gravity, β the coefficient of thermal expansion, T_0 the average value of temperature, γ the unit vector in the vertical direction, and $w(T, C)$ is the reaction rate. Usually w is considered of the form

$$w(T, C) = ke^{-E/R_0 T}\phi(C), \quad \phi(C) = C^n,$$

where k is the preexponential factor, E is the activation energy, R_0 the gas constant, and n the order of the reaction.

For the direct computation of (1–4) we consider the reaction rate to be first order, i.e., $n = 1$ in the formula for the kinetic function $\phi(C)$. Moreover, we denote θ the angle between the gravity vector and the z-axis. We use the biharmonic formulation of Navier–Stokes with Ψ as the stream function:

$$\text{(I)} \begin{cases} \partial T/\partial t + (\partial \Psi/\partial z)(\partial T/\partial x) - (\partial \Psi/\partial x)(\partial T/\partial z) = \Delta T + W \\ \partial C/\partial t + (\partial \Psi/\partial z)(\partial C/\partial x) - (\partial \Psi/\partial x)(\partial C/\partial z) = \epsilon \Delta C - W \\ T \to 0, \quad C \to 1, \quad \text{as } z \to -\infty, \\ T \to 1, \quad C \to 0, \quad \text{as } z \to +\infty, \\ \partial T/\partial x(0, z) = 0, \quad \partial T/\partial x(L, z) = 0, \quad \forall z \\ \partial C/\partial x(0, z) = 0, \quad \partial C/\partial x(L, z) = 0, \quad \forall z \end{cases} \tag{5}$$

$$\text{(II)} \begin{cases} \partial \Delta\Psi/\partial t = (\partial \Psi/\partial x)(\partial \Delta\Psi/\partial z) - (\partial \Psi/\partial z)(\partial \Delta\Psi/\partial x) \\ \qquad\qquad + \Delta^2 \Psi - RP\cos(\theta)\partial T/\partial x - RP\sin(\theta)\partial T/\partial z. \\ \Psi \to 0, \quad \text{as } z \to -\infty, \\ \Psi \to 0, \quad \text{as } z \to +\infty, \\ \Psi(0, z) = 0, \quad (\partial \Psi/\partial x)(0, z) = 0, \quad \forall z, \\ \Psi(L, z) = 0, \quad (\partial \Psi/\partial x)(L, z) = 0, \quad \forall z. \end{cases} \tag{6}$$

Here W denotes the source term given by the Arrhenius law,

$$W = C \exp \frac{ZT}{1 + \delta(1 - T)}.$$

The parameters are the Zeldovich number $Z = \frac{qE}{R_0 T_b^2}$, the Prandtl number $P = \frac{\nu}{\kappa}$, the Rayleigh number R, a dimensionless mass diffusion ϵ, and the number $\delta = q/T_b$, where $T_b = T_i + q$ is the adiabatic temperature, and T_i is the temperature of the cold product A.

The boundary conditions are nonperiodic in the x-direction. The boundary conditions on the other walls are defined according to the asymptotic behavior of the unknowns when $z \to \infty$. Also, the length $2H$ of the discretized computational domain $[-H, H]$ in the z-direction has to be large enough in order to have no influence on the dynamics of the front in the numerical computation.

The coupling between the two models comes from the Boussinesq term, which involves derivatives of the temperature in the x-direction and eventually stiff derivative in the z-direction, if the symmetry axis in the z-direction of the computational domain makes a nonzero angle θ with the gravity vector.

The system has a well-known one-dimensional traveling wave solution ($T_0(z), C_0(z)$, $\Psi \equiv 0, \theta = 0$). In addition, many different possible nonlinear regimes of the solution exist [17]. They all depend on the specific value of the bifurcation or control parameters, such as the Zeldovich number or the Rayleigh number. This test case is relevant in demonstrating and validating the feasibility of our approach with local Fourier basis. Low-order methods fail on this test case except if one uses drastic mesh refinement to capture pattern formations and bifurcation values. These numerical difficulties usually lead to use of the spectral method as in [2]. But the global transposition of data on a network of processors needed in the spectral method makes them difficult to parallelize [5].

3. SUPERPOSITION PRINCIPLE TO SOLVE (I) AND (II)

The superposition principle is a concept widely used in physics. This section derives the methodology to solve the coupled systems of PDEs (I) and (II). In both of the two considered systems, the solutions and/or their derivatives exhibit strong variation. The main idea of the methodology developed to solve (I) is to introduce a shift on the unknowns T and C, which can be a numerical analogue of the traveling wave of the system and which makes the computation "amenable" to Fourier approximation.

3.1. Superposition Principle for the Reaction–Diffusion PDEs (I)

A convenient way to compute the unknowns T and C is to use the splitting

$$T(x, z) = T_0(z) + T_1(x, z) \tag{7}$$

$$C(x, z) = C_0(z) + C_1(x, z), \tag{8}$$

where (T_0, C_0) satisfies the required asymptotic behavior

$$T_0(-\infty) = 0, \quad T_0(\infty) = 1, \quad C_0(-\infty) = 1, \quad C_0(\infty) = 0.$$

Typically, we take

$$T_0(z) = \frac{1}{2}(1 + \tanh(z)), \quad C_0(z) = 1 - T_0(z).$$

The computation of (T_1, C_1) is then amenable to Fourier approximation in the z-direction on the finite but large interval $(-H, H)$. In fact, if H is large enough, T_1, C_1 should vanish exponentially in the neighborhood of $\pm H$, and T_1, C_1 can be interpreted as smooth periodic functions of period $2H$. Because of the one-sided flame front propagation, the solution must be shifted in space occasionally to keep the position of the flame front roughly in the neighborhood of the central line $z = 0$. A possible way of getting an adaptive computation with a regular discretization is to let T_0, C_0 be a traveling wave solution of the system [10] instead of an a priori hyperbolic tangent profile. Depending on the value of the bifurcation

parameters, T_1, C_1 can be seen then as a possibly small perturbation of the so-called basic solution.

For simplicity of notation in the following, we will denote (T_1, C_1) as (T, C). We use a second-order semi-implicit backward Euler scheme for time marching that is explicit in the nonlinear source terms:

$$T^{n+1} - 2/3\Delta t \Delta T^{n+1} = 2T^n - 1/2T^{n-1} - 2/3\Delta t \Bigg(\Delta T_0 + W(T^n + T_0, C^n + C_0) + \frac{\partial \Psi^n}{\partial x}\left(\frac{\partial T_0}{\partial z} + \frac{\partial T^n}{\partial z}\right) - \frac{\partial \Psi^n}{\partial z}\left(\frac{\partial T_0}{\partial x} + \frac{\partial T^n}{\partial x}\right)\Bigg) \tag{9}$$

$$C^{n+1} - 2/3\Delta t \epsilon \Delta C^{n+1} = 2C^n - 1/2C^{n-1} - 2/3\Delta t \Bigg(\epsilon \Delta C_0 - W(T^n + T_0, C^n + C_0) + \frac{\partial \Psi^n}{\partial x}\left(\frac{\partial C_0}{\partial z} + \frac{\partial C^n}{\partial z}\right) - \frac{\partial \Psi^n}{\partial z}\left(\frac{\partial C_0}{\partial x} + \frac{\partial C^n}{\partial x}\right)\Bigg). \tag{10}$$

We look for discrete Fourier expansion of the unknowns:

$$T^{n+1}(x, z) = \sum_{k=-M_z}^{M_z} \hat{T}_k^{n+1}(x) \exp(ikz), \tag{11}$$

$$C^{n+1}(x, z) = \sum_{k=-M_z}^{M_z} \hat{C}_k^{n+1}(x) \exp(ikz). \tag{12}$$

The system of ODE equations for the mode k is written as

$$(3/(2\Delta t) + k^2)\hat{T}_k^{n+1}(x) - \partial^2 \hat{T}_k^{n+1}(x)/\partial x^2 = \hat{F}_k \tag{13}$$

$$(3/(2\Delta t) + \epsilon k^2)\hat{C}_k^{n+1}(x) - \epsilon \partial^2 \hat{C}_k^{n+1}(x)/\partial x^2 = \hat{G}_k, \tag{14}$$

with homogeneous Neumann boundary conditions in the x-direction.

In order to compute for each Fourier k-mode the coefficients $(\hat{T}_k^{n+1}(x), \hat{C}_k^{n+1}(x))$, with Fourier expansion in the x-direction, we use the technique of [8] (see also [13–15]) based on the superposition principle and the construction of a smooth periodic extension of the right-hand sides. Let us restrict ourselves to the temperature equation, since the algorithm is the same for the concentration equation. We split again the unknowns as follows:

$$\hat{T}_k^{n+1}(x) = T_k^{[F]}(x) + T_k^{[C]}(x), \quad x \in [0, L]. \tag{15}$$

Let $d > 0, d \in \mathbf{R}$, 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 search for the $L + d$ periodic solution of the inhomogeneous problem

$$(3/(2\Delta t) + k^2)T_k^{[F]}(x) - \partial^2 T_k^{[F]}(x)/\partial x^2 = \bar{F}_k, \quad T_k^{[F]}\ L + d \text{ periodic}. \tag{16}$$

Then we retrieve the homogeneous Neumann boundary condition satisfied by $\hat{T}_k^{n+1}$ by computing the so-called corrector terms $T_k^{[C]}$. These corrector terms satisfy the ODEs

$$\begin{aligned} &(2/(3\Delta t) + k^2)T_k^{[C]}(x) - \partial^2 T_k^{[C]}(x)/\partial x^2 = 0, \\ &\partial T_k^{[C]}/\partial x(0) = -\partial T_k^{[F]}/\partial x(0), \\ &\partial T_k^{[C]}/\partial x(L) = -\partial T_k^{[F]}/\partial x(L). \end{aligned} \tag{17}$$

The solution is written as

$$T_k^{[C]}(x) = \alpha_k v_{T,k}(x)) + \beta_k\ w_{T,k}(x)), \tag{18}$$

with

$$v_{T,k}(x) = \exp(-\sqrt{k^2 + 2/(3\Delta t)}(x - 0)), \tag{19}$$

$$w_{T,k}(x) = \exp(-\sqrt{k^2 + 2/(3\Delta t)}(L - x)). \tag{20}$$

The two-dimensional vector basis of this corrector space for each k mode can be computed analytically once and for all. The coefficients (α_k, β_k) are then solutions of two-by-two linear systems that are solved at each time step. We notice that the boundary corrections on the left and on the right are numerically decoupled when the time step Δt is small enough and/or the wave number k is large enough.

Our approach differs from that of Israeli *et al.* [13, 14] 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 $q + 2$ at most [12]. Let us notice that the order of the method is $q + 2$ at the artificial interfaces and increases away from the discontinuity points. Further, the accuracy of the method is relatively insensitive to the size of the extension d if the number of modes is large enough [6].

When the right-hand side is a given analytical function that can be defined on the interval $[0,\ L + d]$, one could use (as in Israeli *et al.* [13, 14]) a so-called bell function B 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 $L + d$. Results of Averbuch *et al.* [1] show evidence of the accuracy of their method for the Laplace equation or the Helmholtz problem. However, in our computation the right-hand side is given numerically at each time step only inside 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 $\bar{F}_k$ until order q at $x = 0$ and $x = L$. The classical Hermite interpolation allows us to define a polynomial function P on $[L,\ L + d]$ of degree $2q + 1$ that interpolates the function $\bar{F}_k$ with the following conditions:

$$\begin{aligned} &P^{(0)}(L) = \hat{F}_k^{(0)}(L) \quad P^{(1)}(L) = \hat{F}_k^{(1)}(L), \ldots \quad P^{(q)}(L) = \hat{F}_k^{(q)}(L), \\ &P^{(0)}(L + d) = \hat{F}_k^{(0)}(0) \quad P^{(1)}(L + d) = \hat{F}_k^{(1)}(0), \ldots \quad P^{(q)}(L + d) = \hat{F}_k^{(q)}(0). \end{aligned} \tag{21}$$

The extended right-hand side is then

$$\bar{F}_k = \begin{cases} \hat{F}_k, & \forall x \in [0, L] \\ P(x), & \forall x \in]L, L + d\,[. \end{cases} \tag{22}$$

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

To summarize, the unknowns (T, C) at each time step can be computed with two-dimensional FFTs complemented by some lower order cost numerical procedures, such as

Hermite interpolation, to get the right-hand side of the equation defined properly and some a posteriori combination of given basis functions for the corrector terms. Note that these corrections are exponentially negligible outside some neighborhood of the wall $x = 0,\ x = L$ for a small time step.

3.2. Superposition Principle for the NS Equation (II)

We consider the following semi-implicit Euler scheme for the time discretization of Eq. (6) as in [8]:

$$\begin{aligned}\Delta\Psi^{n+1} - \Delta t\Delta^2\Psi^{n+1} &= \Delta t((\partial\Psi^n/\partial x)(\partial\Delta\Psi^n/\partial z) - (\partial\Psi^n/\partial z)(\partial\Delta\Psi^n/\partial x)\\ &\quad + \Delta\Psi^n - \Delta t f(x,z,t_n), \quad (x,z)\in(0,L)\times(-H,H),\\ \Psi^{n+1}(0,z) &= 0, \quad (\partial\Psi^{n+1}/\partial x)(0,z) = 0, \quad z\in(-H,H),\\ \Psi^{n+1}(L,z) &= 0, \quad (\partial\Psi^{n+1}/\partial x)(L,z) = 0, \quad z\in(-H,H).\end{aligned} \tag{23}$$

We look for a z-periodic solution of period $2H$ with H large. As a matter of fact, Ψ vanishes exponentially when $|z|$ goes to infinity. H is taken large enough in order that the boundary conditions in the z-direction have no influence on the dynamic of the combustion wave.

The terms f and Ψ at time t_n are approximated by the discrete Fourier expansion,

$$f(x,z,t_n) = \sum_{k=\frac{-N_z}{2}}^{\frac{N_z}{2}-1} \hat{f}_k^n(x)\, e^{iky}, \tag{24}$$

$$\Psi^n(x,z,t_n) = \sum_{k=\frac{-N_z}{2}}^{\frac{N_z}{2}-1} \hat{\Psi}_k^n(x)\, e^{iky}, \tag{25}$$

with $y = \frac{(z+H)\,\pi}{H}$. For simplicity of notation we will assume in the following that $H = \pi$. Let $F^n(x,z)$ be the right-hand side of Eq. (23). From the approximation,

$$\begin{aligned}&(\partial\Psi^n/\partial x)(\partial\Delta\Psi^n/\partial y) - (\partial\Psi^n/\partial y)(\partial\Delta\Psi^n\partial x)\\ &\approx i\sum_{k=\frac{-N_z}{2}}^{\frac{N_z}{2}-1}\left[e^{iky}\sum_{k_1+k_2=k}\left(\hat{\Psi}_{k_1}^{n'}k_2\left(\hat{\Psi}_{k_2}^{n''} - k_2^2\hat{\Psi}_{k_2}^n\right) - \hat{\Psi}_{k_1}^n k_1\left(\hat{\Psi}_{k_2}^{n'''} - k_2^2\hat{\Psi}_{k_2}^{n'}\right)\right)\right],\end{aligned} \tag{26}$$

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

$$\begin{aligned}L_0\left[\hat{\Psi}_0^{n+1}\right] &= \left(\hat{\Psi}_0^{n+1}\right)'' - \Delta t\left(\hat{\Psi}_0^{n+1}\right)'''' = \hat{F}_0^n, \quad x\in(0,L),\\ \hat{\Psi}_0^{n+1}(0) &= \left(\hat{\Psi}_0^{n+1}\right)'(0) = \hat{\Psi}_0^{n+1}(L) = \left(\hat{\Psi}_0^{n+1}\right)'(L) = 0,\end{aligned} \tag{27}$$

and for $k \neq 0$:

$$\begin{aligned}L_k\left[\hat{\Psi}_k^{n+1}\right] &= \left(\hat{\Psi}_k^{n+1}\right)'' - k^2\hat{\Psi}_k^{n+1} - \Delta t\left(\left(\hat{\Psi}_k^{n+1}\right)'''' \right.\\ &\quad \left. - 2k^2\left(\hat{\Psi}_k^{n+1}\right)'' + k^4\hat{\Psi}_k^{n+1}\right) = \hat{F}_k^n, \quad x\in(0,L),\\ \hat{\Psi}_k^{n+1}(0) &= \left(\hat{\Psi}_k^{n+1}\right)'(0) = \hat{\Psi}_k^{n+1}(L) = \left(\hat{\Psi}_k^{n+1}\right)'(L) = 0.\end{aligned} \tag{28}$$

In order to compute $\hat{\Psi}_k^{n+1}(x)$ with Fourier in the x-direction as well as in the z-direction, we use the technique of [8] (see also [13–15]) based on the superposition principle and the construction of a smooth periodic extension of the right-hand sides. We split the unknowns as follows:

$$\hat{\Psi}_k^{n+1}(x) = \hat{\Psi}_k^{[F],n+1}(x) + \hat{\Psi}_k^{[C],n+1}(x), \quad x \in [0, L]. \tag{29}$$

This splitting is efficient because one can use a fast Fourier transform to compute $\hat{\Psi}_k^{[F],n+1}$ and an 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 $d > 0, d \in \mathbf{R}$, 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

$$L_k\big[\hat{\Psi}^{[F]}\big] = \bar{F}, \quad x \in (0, L+d), \quad \hat{\Psi}^{[F]}\ L+d \text{ periodic}, \quad \hat{\Psi}^{[F]} \in C^{q+4}(\mathbf{R}). \tag{30}$$

Then we retrieve the homogeneous Neumann and Dirichlet boundary conditions satisfied by $\hat{\Psi}_k^{n+1}$ by computing the so-called corrector term $\Psi^{[C]}$. These corrector terms satisfy the ODE problems

$$\begin{aligned} L_k\big[\hat{\Psi}^{[C]}\big] &= 0, && x \in (0, L), \\ \hat{\Psi}^{[C]}(0) &= -\hat{\Psi}^{[F]}(0), && \big(\hat{\Psi}^{[C]}\big)'(0) = -\big(\hat{\Psi}^{[F]}\big)'(0), \\ \hat{\Psi}^{[C]}(L) &= -\hat{\Psi}^{[F]}(L), && \big(\hat{\Psi}^{[C]}\big)'(L) = -\big(\hat{\Psi}^{[F]}\big)'(L). \end{aligned} \tag{31}$$

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

$$v_0(x) = \exp\left(-\frac{1}{\sqrt{\Delta t}}(x-0)\right), \quad w_0(x) = \exp\left(-\left(\frac{1}{\sqrt{\Delta t}}(L-x)\right),$$
$$r_0(x) = x, \quad s_0(x) = (L-x),$$

and for $k \neq 0$,

$$v_k(x) = \exp\left(-\sqrt{k^2 + \frac{1}{\Delta t}}\ (x-0)\right), \quad w_k(x) = \exp\left(-\sqrt{k^2 + \frac{1}{\Delta t}}\ (L-x)\right),$$
$$r_k(x) = \exp(-|k|\ (x-0)), \quad s_k(x) = \exp(-|k|\ (L-x)).$$

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 $\hat{\Psi}_k^{[C]}$ are readily computed from the previous formula. The derivatives of $\hat{\Psi}_k^{[F]}$ follow from its discrete Fourier expansion. We assemble the right-hand side of Eq. (23) at each time step using splitting (29) on the derivatives as well.

Notice that the smoothness of the 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.

However, the parallel efficiency of this method with no domain decomposition poses the classical problem that one has to do a global transpose of the unknown fields on a network of processors. The introduction of the domain decomposition procedure in the next section solves this problem.

4. DOMAIN DECOMPOSITION WITH LOCAL FOURIER BASIS

We extend the previous methodology by introducing domain decomposition in the x-direction for each k-mode equation (13 and 14). We split $[0, L]$ into nd nonoverlapping subdomains of not necessarily equal sizes. In particular, the size of the subdomains near the boundary conditions of the problem can be smaller, leading to a smaller space step. To simplify the presentation here, we take subdomains of equal size and we denote them generically as $(0, l)$. We first detail the domain decomposition for each system (I) and (II) of the model problem. We secondly investigate numerically the influence on accuracy of each parameter of the domain decomposition method.

4.1. The Domain Decomposition Algorithm

For each subdomain, we apply the same splitting on the Fourier modes of the unknowns as described above for the problem (I), and we impose in addition the C^1 continuity of 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 strictly what we had for the single domain case, but it is applied for each subdomain in parallel. Then we compute the corrector term on each subdomain in order to retrieve the C^1 continuity of the solution at the artificial interfaces. Let us denote (v_k^j, w_k^j) as the set of basis functions for the corrector in each subdomain j and (α_k^j, β_k^j) as the corresponding coefficients; in the local coordinate system of the subdomain, the basis functions are identical to (19 and 20). With four subdomains, for example, the matrix of the interface problem for the unknown coefficient vector $\vec{T} = (\alpha_k^1, \beta_k^1, \alpha_k^2, \beta_k^2, \alpha_k^3, \beta_k^3, \alpha_k^4, \beta_k^4)^T$ is

$$\begin{pmatrix}
v'^1_k(0) & w'^1_k(0) & 0 & 0 & 0 & 0 & 0 & 0 \\
v^1_k(l) & w^1_k(l) & -v^2_k(0) & -w^2_k(0) & 0 & 0 & 0 & 0 \\
v'^1_k(l) & w'^1_k(l) & -v'^2_k(0) & -w'^2_k(0) & 0 & 0 & 0 & 0 \\
0 & 0 & v^2_k(l) & w^2_k(l) & -v^3_k(0) & -w^3_k(0) & 0 & 0 \\
0 & 0 & v'^2_k(l) & w'^2_k(l) & -v'^3_k(0) & -w'^3_k(0) & 0 & 0 \\
0 & 0 & 0 & 0 & v^3_k(l) & w^3_k(l) & -v^4_k(0) & -w^4_k(0) \\
0 & 0 & 0 & 0 & v'^3_k(l) & w'^3_k(l) & -v'^4_k(0) & -w'^4_k(0) \\
0 & 0 & 0 & 0 & 0 & 0 & v'^4_k(l) & w'^4_k(l)
\end{pmatrix}.$$

Let us remark that the matrix is time independent.

To solve the Navier–Stokes problem (II), on each subdomain we apply the same splitting of the unknowns as described above for the problem (I), and in addition we impose the C^3 continuity of 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.

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 subtracts this shift and the superposition principle gives the unique solution of (27).

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 $(v_k^j, w_k^j, r_k^j, s_k^j)$ as the set of basis functions for the corrector in each subdomain j, and $(\alpha_k^j, \beta_k^j, \gamma_k^j, \delta_k^j)$ as 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 $\vec{T} = (\alpha_k^1, \beta_k^1, \gamma_k^1, \delta_k^1, \alpha_k^2, \beta_k^2, \gamma_k^2, \delta_k^2)^T$ writes

$$\begin{pmatrix}
v_k^1(0) & w_k^1(0) & r_k^1(0) & s_k^1(0) & 0 & 0 & 0 & 0 \\
v'^1_k(0) & w'^1_k(0) & r'^1_k(0) & s'^1_k(0) & 0 & 0 & 0 & 0 \\
v_k^1(l) & w_k^1(l) & r_k^1(l) & s_k^1(l) & -v_k^2(0) & -w_k^2(0) & -r_k^2(0) & -s_k^2(0) \\
v'^1_k(l) & w'^1_k(l) & r'^1_k(l) & s'^1_k(l) & -v'^2_k(0) & -w'^2_k(0) & -r'^2_k(0) & -s'^2_k(0) \\
v''^1_k(l) & w''^1_k(l) & r''^1_k(l) & s''^1_k(l) & -v''^2_k(0) & -w''^2_k(0) & -r''^2_k(0) & -s''^2_k(0) \\
v'''^1_k(l) & w'''^1_k(l) & r'''^1_k(l) & s'''^1_k(l) & -v'''^2_k(0) & -w'''^2_k(0) & -r'''^2_k(0) & -s'''^2_k(0) \\
0 & 0 & 0 & 0 & v_k^2(l) & w_k^2(l) & r_k^2(l) & s_k^2(l) \\
0 & 0 & 0 & 0 & v'^2_k(l) & w'^2_k(l) & r'^2_k(l) & s'^2_k(l)
\end{pmatrix}.$$

4.2. The Decoupling of the Interface Problem with Respect to Mode

For small time steps or large wave number k, the local interface problems become approximately decoupled because of the exponential decay of the basis function. Then one can replace, for modes k larger than k_0, the interface problem operator A_k by an approximate interface operator B_k where the coupling terms between far subdomains are neglected. This property is very interesting in term of parallelism because the data dependencies of the interface problem on each subdomain only depend on the two neighbor subdomains. These interface problem operators, A_k and B_k for the temperature solution and the concentration for $nd = 4$ subdomains of local size L, can be written with $\xi = \sqrt{k^2 + 2./(3 \cdot \Delta t)}$, and $\eta = e^{-\xi L}$ as

$$A_k = \begin{bmatrix}
-\xi & \xi\eta & 0 & 0 & 0 & 0 & 0 & 0 \\
\eta & 1 & -1 & -\eta & 0 & 0 & 0 & 0 \\
-\xi\eta & \xi & \xi & -\xi\eta & 0 & 0 & 0 & 0 \\
0 & 0 & \eta & 1 & -1 & -\eta & 0 & 0 \\
0 & 0 & -\xi\eta & \xi & \xi & -\xi\eta & 0 & 0 \\
0 & 0 & 0 & 0 & \eta & 1 & -1 & -\eta \\
0 & 0 & 0 & 0 & -\xi\eta & \xi & \xi & -\xi\eta \\
0 & 0 & 0 & 0 & 0 & 0 & \xi\eta & -\xi
\end{bmatrix} \tag{32}$$

$$B_k = \begin{bmatrix} -\xi & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \xi & \xi & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \xi & \xi & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \xi & \xi & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\xi \end{bmatrix}. \tag{33}$$

Similar matrix operators can be written for C and Ψ. The cut level k_0 separates the Fourier mode set in two subsets. The first subset includes the modes less than k_0 for which the interface problem involves communications between all the subdomains. The second subset includes the modes greater or equal to k_0 for which the interface problem involves communications only between the two adjacent subdomains. The value of k_0 is determined as follows.

Let $A_k T_k^{[C]} = RHS^{\text{interface}}(T_k^{[F]})$ be the exact interface problem. We want to know when it is possible to replace A_k by B_k.

Let $B_k(T_k^{[C]} + \delta) = RHS^{\text{interface}}(T_k^{[F]})$ be the approximate problem. A direct computation leads to the following error bound on the perturbation δ of the corrector term $T_k^{[C]}$:

$$\|\delta\|_\infty \le \left\|A_k^{-1}\right\|_\infty \|A_k - B_k\|_\infty \left\|B_k^{-1}\right\|_\infty \left\|RHS^{\text{interface}}\left(T_k^{[F]}\right)\right\|_\infty. \tag{34}$$

For a given set of the numerical method's parameters $(L, k, \Delta t)$, one obtains a priori the constant $\|A_k^{-1}\|_\infty \|A_k - B_k\|_\infty \|B_k^{-1}\|_\infty$. The time-dependent term $\|RHS^{\text{interface}}(T_k^{[F]})\|_\infty$ must be obtained in the core of the numerical simulation itself.

Figure 1 (respect. 2) gives the value of $\log 10(\|A_k^{-1}\|_\infty \|A_k - B_k\|_\infty \|B_k^{-1}\|_\infty)$ for the interface problem of temperature Eq. (9) (respect. streamfunction Eq. (23)) with respect to

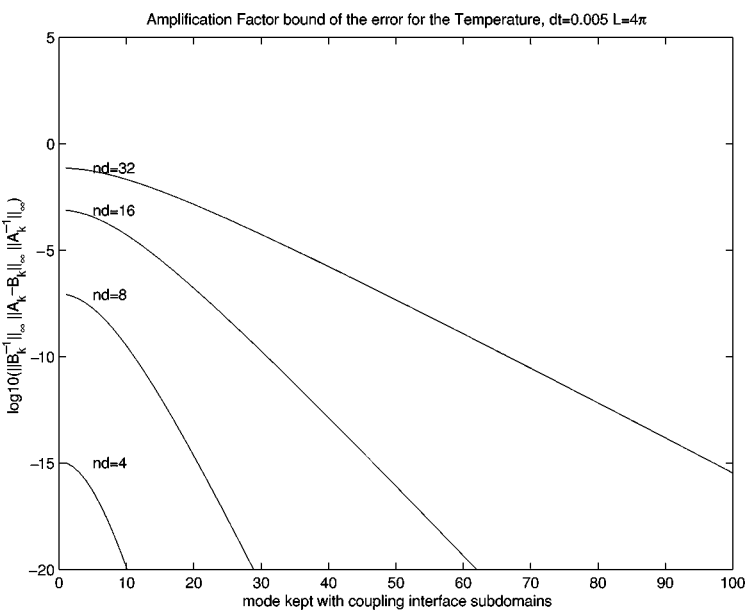

FIG. 1. Factor of the bound Error estimation for $T_k^{[C]}$ for different values of nd subdomains.

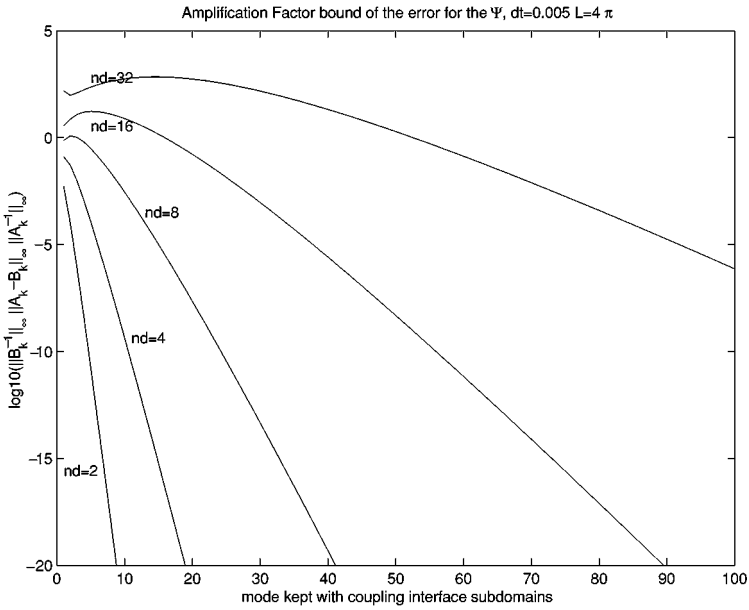

FIG. 2. Factor of the bound Error estimation for $\Psi_k^{[C]}$ for different values of nd subdomains.

the mode k, and for different numbers nd of subdomains. The time step is $\Delta t = 0.005$, and the width of the subdomains is $L = 4 \cdot \pi / nd$.

These results clearly show that the value of k_0 increases with the number of subdomains. It is a consequence of the fact that for a given global domain, the exponential decay of the corrector basis function in variable $y = x/l$ is slower when the number of subdomains increases. In practice, a factor error bounds of 10^{-5} for $nd = 16$ is reached with keeping few modes ($k_0 = 40$ for Ψ and $k_0 = 15$ for T).

Similarly the rate of decay proportional to $e^{-|k|x}$ of the corrector basis function for T and C is slower than the rate of decay of the corrector basis proportional to $e^{-\sqrt{k^2+2./(3\Delta t)}x}$ for the streamfunction. Consequently, k_0 is larger for Ψ than for T and C.

As we want $\|\delta\|_\infty$ to be less than a set tolerance value ϵ, we can determine adaptively the value of k_0 at each time step with respect to the value of the Fourier solution-dependent term $\|RHS^{\text{interface}}(T_k^{[F]})\|_\infty$.

To anticipate the results of Section 6, we show in Figs. 3 and 4 an example of the computed values of the $\|RHS^{\text{interface}}(\Psi_k^{[F]})\|_\infty$ and $|||RHS^{\text{interface}}(T_k^{[F]})||_\infty$ for four subdomains with 128 Fourier modes for the complete domain in the x-direction and 256 modes for the z-direction. The parameters of the numerical simulation are $Z = 7.8$, $R = 1.5$, $\theta = 90^o$, $\Delta t = 0.005$.

In conclusion, combining the results of Figs. 1 and 2 with Fig. 3 and the estimate (34), we see that k_0 is 5 for T for the 8 subdomains case, and k_0 is 13 for Ψ with $\delta < 10^{-9}$ for the 4 subdomains case.

4.3. Numerical Accuracy of the Domain Decomposition Method

We are going to test the accuracy first of the domain decomposition with respect to the number of Fourier modes and the number of subdomains. We consider fourth-order wave

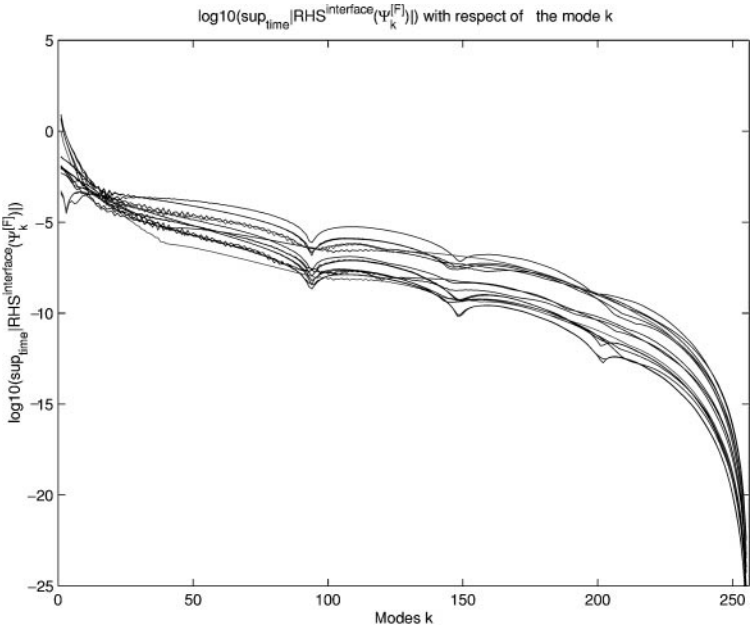

FIG. 3. Example of the maximum computed values of $RHS^{\text{interface}}(\Psi_k^{[F]})$ on the interfaces involving in the interface problem.

equation analogous to (28) with H-periodical function instead of 2π-periodical function. We force the right-hand side $\hat{F}_k^n$ such that the fourth-order polynomial $x^2 * (x - 4\pi)^2/(16 * \pi^4)$ is an exact steady solution. We use the time marching scheme starting from the trivial initial condition until convergence to the steady solution is reached. We measure the

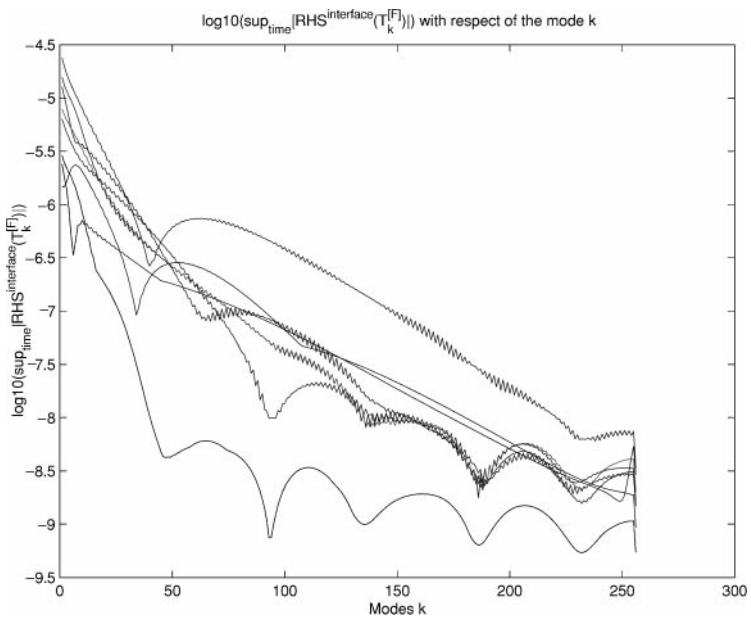

FIG. 4. Example of the maximum computed values of $RHS^{\text{interface}}(T_k^{[F]})$ on the interfaces involving in the interface problem.

TABLE I
Accuracy with Respect to the Number of Subdomains

Mode 1	Number of subdomains			
$N_x = 256$	2	4	8	16
$N_r = 16$	4.2e-7	2.6e-6	8.8e-7	4.2e-7
$N_r = 8$	1.2e-5	5.5e-6	2.6e-6	1.3e-6
$N_x = 128$	2	4	8	16
$N_r = 16$	1.8e-5	1.0e-5	4.5e-6	2.4e-6
$N_r = 8$	7.9e-5	3.9e-5	1.8e-5	9.9e-6

difference between the converged numerical solution and the exact steady solution in maximum norm.

Table I gives the error in a maximum norm for the computed solution corresponding to the wave number $k = 0$ and $k = 1$ and several different numbers of subdomains. The total number of discretization points in the 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 $\Delta t = 0.1$. The size of the domain of computation is given by $H = 25$ and $L = 2\pi$.

We see in all cases that $N_r = 16$ gives better results than $N_r = 8$, and that the influence of the number of extra mode N_r gets significant when the number of modes per subdomain is not large enough.

The results of this table for mode $k = 1$ are always better than the corresponding results for the zero mode no matter the size of the extension. In particular, the error decreases with the number of subdomains for mode one. We observe the opposite behavior for the zero mode result except for the large number of Fourier modes per subdomain case ($N_x = 256$ and $N_r = 16$).

It is important to notice that as the problems (27) and (28) are singular perturbation problems when the time step Δt is small and/or the wave number k is large, the domain decomposition method can be sensitive to these parameters. Let us now study the sensitivity of the method to the time step, wave number k, and the size of the extension per subdomain, with the same set of k dependent boundary value problems.

Table II 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_x for different couples $(k, \Delta t)$ of time steps and wave numbers and four subdomains. N_y is the number of Fourier modes per subdomain including the extension.

We observe that the accuracy of the method deteriorates when the time step goes to zero: This is no surprise since the problem becomes more and more singular as Δt goes to zero. Eventually, the time-dependent schemes diverge for the zero mode equation if the space step is larger than the boundary layer thickness $\sqrt{\Delta t}$ 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 $\sqrt{\frac{1}{k^2} + \Delta t}$, and this phenomenon can be interpreted as above. However,

TABLE II
Influence of the Parameters on Accuracy

	50%	25%	12.5%	6.25%
		$k = 30$, $\Delta t = 0.1$		
16	2.89e-4	1.66e-3	7.06e-3	2.20e-2
32	3.99e-5	2.54e-4	1.55e-3	7.52e-3
64	5.02e-6	3.23e-5	2.18e-4	1.45e-3
128	5.57e-7	3.26e-6	2.19e-5	1.62e-4
		$k = 30$, $\Delta t = 0.01$		
16	2.90e-4	1.66e-3	7.07e-3	2.20e-2
32	3.99e-5	2.55e-4	1.55e-3	7.53e-3
64	5.02e-6	3.23e-5	2.18e-4	1.45e-3
128	5.57e-7	3.26e-6	2.20e-5	1.63e-4
		$k = 30$, $\Delta t = 0.005$		
16	2.90e-4	1.67e-3	7.08e-3	2.21e-2
32	4.00e-5	2.55e-4	1.55e-3	7.54e-3
64	5.03e-6	3.24e-5	2.18e-4	1.45e-3
128	5.58e-7	3.27e-6	2.20e-5	1.63e-4
		$k = 30$, $\Delta t = 0.001$		
16	2.94e-4	1.69e-3	7.15e-3	2.22e-2
32	4.05e-5	2.58e-4	1.57e-3	7.61e-3
64	5.09e-6	3.28e-5	2.21e-4	1.47e-3
128	5.65e-7	3.31e-6	2.23e-5	1.65e-4

the amplitude of solution $\hat{\Psi}_k$ should decrease with respect to the wave number k dependency on the regularity of Ψ. As reported in Section 6, this phenomenon does 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 rather than a small one. We speculate that small extensions lead to stability problems of the time marching scheme.

Finally, we have checked that the method is at least of order 5 for mode 0 and $d/(L+d) =$ 50% or 25% and $\Delta t \leq 0.01$. The order of the method was computed by involving the largest N_x values. The order of the method for the modes greater than 0 is less dependent on the value $d/(L+d)$, but differs following the mode value: (order 3.2 for k = 30, order 4.3 for k = 10). Nevertheless, the decrease of the order of this convergence is compensated by the fact that the module of the Fourier coefficients can decrease steeply with the increase of the mode value.

5. PARALLELISM OF THE METHODOLOGY

This section is devoted to the parallel implementation of the methodology on a multiple instruction multiple data (MIMD) computer architecture. We show how to overcome the loss of efficiency that the gathering of the interface problem constitutes. Then we study the efficiency of the implementation using the message passing interface (MPI) library on a distributed memory multiprocessor architecture.

5.1. MIMD Implementation

If we consider the parallelization of the local Fourier basis on one domain, we have to face the parallelization of the Fourier transform. Usually this parallelization consists of splitting the set of Fourier modes between several processors. The Fourier mode equations are solved then in parallel, but all the solution Fourier modes have to be gathered on each processor to build the solution in the physical space. This transposition of the Fourier modes leads to a large amount of data communication from all processors to all processors, and deteriorates the parallelism efficiency of the method. Therefore, the main advantage of the domain decomposition with local Fourier basis is to avoid this global transposition, because of the localization of the data of the Fourier transform on each processor. Several steps of the domain decomposition with local Fourier basis algorithm—as the computation of the Fourier part of the solution on each subdomain, the computation of the derivatives, and the computation of the correction once the coefficients in the vector corrector basis are computed—can be done independently on each subdomain. Thus, the methodology gets a high potential of parallelism.

The break of parallelism arises from the resolution of the interface problem, corresponding to the continuity of the solution and solution derivatives at the interface. The assembly of the right-hand side of the linear system and the computation of the corrector coefficients require data that belong to two or all subdomains. These non-*in situ* informations need to be sent (respect. received) through the communication network to (respect. from) the others processors. Each communication between processors has a cost that depends on the bandwidth of the communication network and on the start up representing the incompressible time needed to establish the communication. Consequently, a parallel efficient implementation needs to minimize the time spent in communication processes.

A straightforward MIMD implementation can be summarized in the two following steps. The first step builds the interface problem RHS, which involves the contribution from each subdomain of trace derivatives of the Fourier term. This can be done by sending from all processors to all processors each local contribution, and then adding each received contribution (with the minus sign) at the right subscript position of the interface vector. At the end of this step, each processor gets the global interface RHS vector. The second step solves redundantly the global interface problem on each processor. Then each processor gets the components of the corrector of the subdomain that it has in charge.

One can take advantage of the no-blocking messages facility that hides communication cost by computation. A no-blocking send or receive starts the communication process and then gives the control to the next instruction with no “rendezvous” for the termination acknowledgment. The termination communication process must be checked with a waiting instruction. Before this checking point, the send or receive of the data is not guaranteed. Table III gives the algorithm of this straightforward implementation with local Fourier basis:

Further, according to Section 4.2, we can take advantage adaptively of the possible decoupling of the local interfaces problems within truncation error accuracy according to the wave numbers, the time step, and estimate (34). This allows us to minimize the number of the global communication and to reduce the size of the linear system per modes that we have to solve redundantly. The time cost of the nonparallel part of the algorithm is then going down in value.

TABLE III
Algorithm I of the Straightforward Implementation of the Domain Decomposition Method with Fourier Local Basis

1. Start the no blocking receive from all processors of the trace of Fourier Term.
2. Compute Fourier solution and Fourier solution derivatives traces at boundaries and artificial boundaries
3. Start the no blocking send to all processors for all modes of the traces of Fourier solution and Fourier solution derivatives
4. wait for the receive completion of step (1)
5. Assemble the interface RHS for all modes
6. Solve redundantly on all processors the interface problem of size 2nd $\times$ 2nd for T and C (respect. 4nd $\times$ 4nd for Ψ)

Table IV summarizes the algorithm II strategy of the domain decomposition with local Fourier basis with adaptive decoupling of the interface problem. In particular, it shows how to overlap the communication from all processors to all processors with the computation of the decoupled interface problems.

Algorithm II is a more sophisticated implementation of the local Fourier method than algorithm I, and we present in the next section our experiments on a parallel computer.

5.2. On the Performance of the Parallel Algorithm

The target parallel computer is a True cluster from Compaq. We use our system with four hypernodes linked by a memory channel hardware. Each hypernode has four alpha processors ev5 cadenced at 400 MHz with 4 MB of cache each and between 512 MB and 852 MB of shared memory. The parallelism measurement is based on the elapsed time to perform 100 time steps.

TABLE IV
Algorithm II Implementation of the Domain Decomposition Method with Local Fourier Basis with Adaptive Decoupling Interface Problem for the Low and the High Modes

1. For j $=$ 1:nd, no-blocking receive of the traces at the artificial boundaries of Fourier solution and Fourier solution derivatives of mode $> k_0$, End
2. For $k \leq k_0$, no-blocking receive of the traces at the artificial boundaries of the Fourier solution and solution derivatives for the kth (high) mode, End
3. Compute Fourier solution and Fourier solution derivatives at the artificial interfaces and boundaries.
4. For j $=$ 1:nd, no blocking send of the traces of Fourier solution and Fourier solution derivatives for modes $> k_0$, End.
5. For $k \leq k_0$, no-blocking send of the traces of Fourier solution and Fourier solution derivatives for the high modes, End.
6. For $k \leq k_0$, Wait for the no-blocking receive of step 2)
   (i) Assemble the kth mode of the interface RHS.
   (ii) Solve the interface problem of size 4 $\times$ 4 for T and C (respect. 8 $\times$ 8 for Ψ) End.
7. For j $=$ 1:nd, Wait the no-blocking receive of Low Mode from subdomain j, End.
8. Assemble the Interface RHS for low mode.
9. Solve redundantly the interface problem of size 2nd $\times$ 2nd for T and C (respect. 4nd $\times$ 4nd for Ψ) for $k \leq k_0$

TABLE V
Scalability of the T and C Solver

MZ = 64/Processors	2	4	8	16
Nx = 32, N1 = 28	100.0%	95.16%	91.48%	81.74%
Nx = 64, N1 = 56	100.0%	93.19%	93.19%	86.45%
Nx = 128, N1 = 112	100.0%	94.57%	94.84%	89.44%
MZ = 128/Processors	2	4	8	16
Nx = 32, N1 = 28	100.0%	96.49%	95.65%	89.84%
Nx = 64, N1 = 56	100.0%	98.09%	96.12%	91.59%
Nx = 128, N1 = 112	100.0%	97.06%	95.66%	91.90%

• First, because of memory constraints on the memory channel, we restrict ourselves to test the scalability and efficiency of the parallel algorithm I for the reaction–diffusion system ((I) and (II)) up to 16 processors. Let M_z be the number of Fourier modes used in the direction of propagation of the front (z-direction). We set M_z to be 64 or 128, (i.e., 128 or 256 discretization points) for the performance evaluation, but M_z in production runs can easily go up to 256 and further. Let $2N_1$ (respect. $2N_r$) be the number of regular discretization points used in $[0, L[$ (respect. in the extension $[L, L + d[$). The total number of modes in x-direction is then $N_x = N_1 + N_r$. We restrict ourselves to have the same number of processors as subdomains.

Table V gives the scalability of the method. The size of the global domain increases as the number of processors used increases. A method has a perfect scalability if the elapsed time to solve a problem of size N on 1 processor is the same as the time to solve a problem of size $P \times N$ on P processors. We define the scalability of the method as the ratio of these two elapsed times. The ratio defining the scalability usually decreases with the number of processors because of the time spent in the communication process. The present method is a domain decomposition method, so we took the two processors elapsed times as the referenced 100% scalability number.

Table V shows that the scalability decreases with the number of subdomains but still gives quite reasonable results for 16 processors. As predicted by the Gustafson's law, the scalability increases when the size of the global problem increases, except when the problem is too large ($M_z = 128$, $M_x = 128$) because of the memory swap.

Table VI gives the efficiency of the method. In this test, the global size of useful modes $N1$ is fixed once and for all. Thus, the size of the problem $N1/P + N_r$ on one processor decreases when the number of processor P increases. In the table, we set N_r to be always 6.

TABLE VI
Parallel Efficiency of the T and C Solver

MZ = 64/Processors	2	4	8	16
N1 = 128, $N_r = 6$	100.0%	98.85%	90.47%	61.17%
N1 = 256, $N_r = 6$	100.0%	117.24%	122.88%	100.18%
N1 = 512, $N_r = 6$	100.0%	139.47%	173.40%	167.79%
MZ = 128/Processors	2	4	8	16
N1 = 128, $N_r = 6$	100.0%	98.43%	89.54%	62.77%
N1 = 256, $N_r = 6$	100.0%	115.56%	111.88%	99.85%

TABLE VII
Influence of k_0 and the Number of Subdomain on Efficiency of the Total Code

	8	16	32	64	
4	118.7%	117.6%	115.4%	114.5%	
8	138.4%	136.4%	129.9%	121.9%	
	8	16	32	64	128
4	100.0%	98.43%	100.2%	99.42%	98.62%
8	116.4%	115.7%	115.0%	112.0%	100.5%

A method is a perfectly efficient if the time to solve a problem of size N on P processors is P times less than the time to solve the same problem on one processor. The efficiency is given by the ratio of the elapsed time on one processor over P times the elapsed time on P processors. However, as for the scalability table, we took the two processors elapsed time as the referenced 100% efficiency number.

Table VI points out that the efficiency is superlinear for $N1 = 256$ and $N1 = 512$. This superlinear efficiency comes mainly from the quadratic complexity of the matrix multiply approach used in the x-direction combined with the efficiency of the cache memory when the number of points in computational subdomain is small enough. However, for 16 processors, the efficiency may drop because the global communication used to gather the interface problem data becomes time consuming.

• Second, we test the efficiency of up to eight processors of the complete combustion code, i.e., with T, C, Ψ solvers with algorithm II and with respect to the cut off parameter k_0. The total number mode N_x is 160 and the number of mode in z-direction is $M_z = 64$ or $M_z = 128$. We set the number of time steps to be 5. Once again, because the method is a domain decomposition method, we took the run on two processors as time reference for the case $Mz = 64$. For the case $Mz = 128$, some memory constraints make us take the time on four processors as the reference time.

Table VII summarizes the efficiencies obtained for this test. As expected, it exhibits that for two subdomains the level k_0 has no influence on the efficiency. The advantage of the adaptive decoupling of algorithm II becomes clear for eight processors. The superlinear behavior of the algorithm can be explained as above. Unfortunately, some constraints on the resources of our memory channel do not allow us to test the efficiency on 16 processors.

6. NUMERICAL RESULTS

We give first some numerical results with the reaction diffusion model and then some results on the combustion model with Navier–Stokes equations.

6.1. Results on the Reaction Diffusion System

Figure 5 gives the concentration and temperature profiles as well as the intermediate unknowns T_1 and C_1 profiles in each space direction for the reaction diffusion system. The computing was done with $N_x = 64$ by $M_z = 128$ modes on a physical domain of size $[0, 4\pi] \times [-25, 25]$.

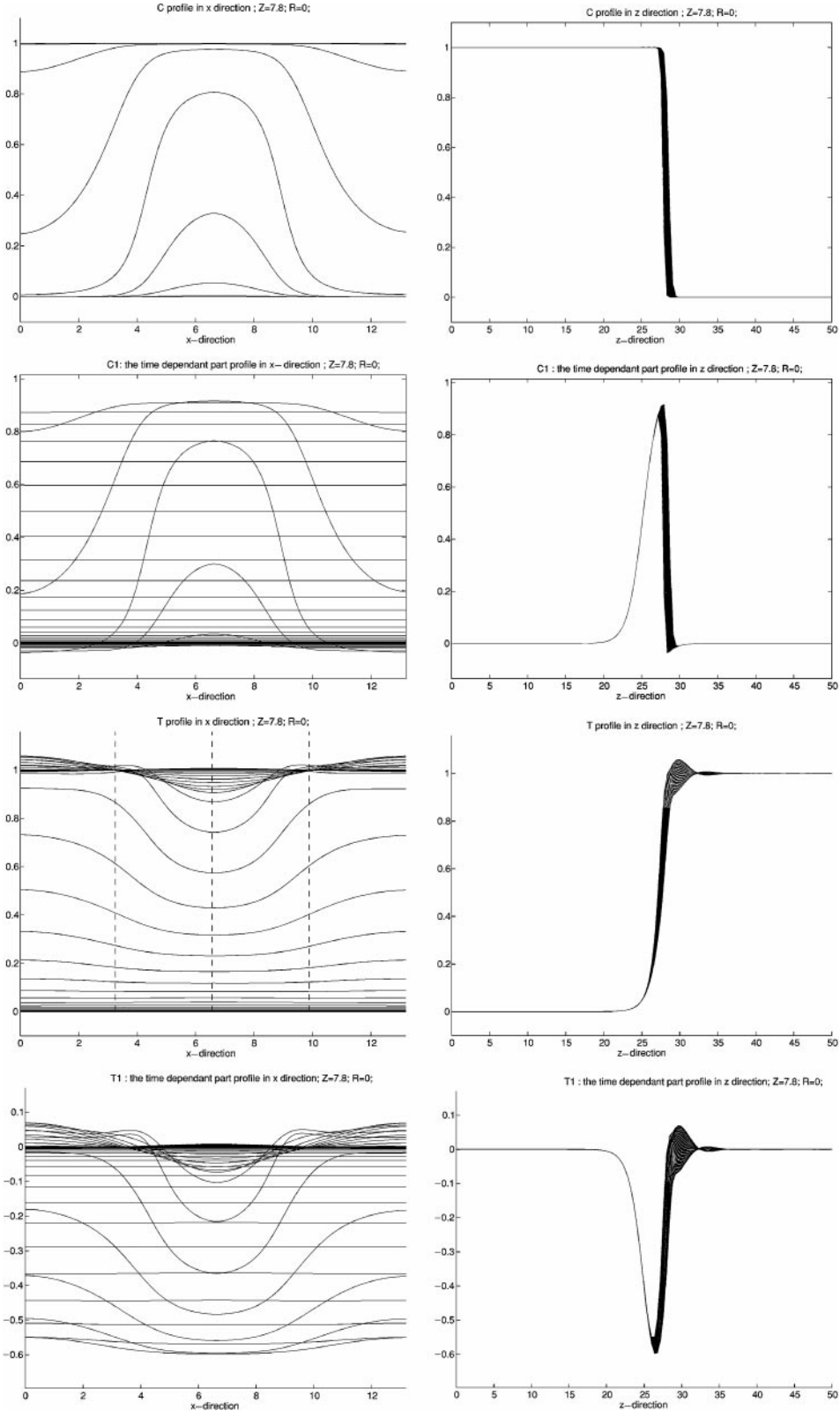

FIG. 5. T and C results and their time-dependent part of ($Z = 7.8$, $R = 0$, $\theta = 0^{\circ}$).

The Zeldovich number is equal to 7.8, the Rayleigh number is equal to 0 (no hydrodynamics), the Prandtl number is $P = 1$, the mass diffusion is $\epsilon = 0.025$. This stabilized solution pattern was obtained after 500 units of time with a time step of $\Delta t = 0.005$. The characteristic pattern of the solution is given by the periodic appearance and disappearance in time of one hot spot in the center of the domain or two symmetric hot spots along the wall. We refer to [8] and its references for a precise description of this pattern formation. The maximum temperature amplitude 1.2236 of the hot spots is 5.1% greater than those obtained by our previous computation with adaptive domain decomposition and piecewise Chebychev approximation [8], while the time periodicity of this instability is equal to 3.695, which is 0.95% slower than the time periodicity of previous computation. These results validate the splitting of the solutions in a time-dependent part and a determined profile.

6.2. FP Process in Liquid

To investigate the effect of the hydrodynamics on the previous thermal instabilities obtained, we have to consider nonzero values of the Rayleigh number. We refer to [8] for a detailed description of the competition's mechanism between the thermal instabilities and the hydrodynamical instabilities (Z = 7.8, R = 5) and the gravity parallel to the combustion propagation direction.

For completeness, we check that the methodology retrieves the same solution behavior as the methodology with domain decomposition with Chebychev piecewise approximation in the z-direction and local Fourier basis without domain decomposition in the x-direction [8].

Figure 6 represents the effect of hydrodynamics on the previous computed solution with (Z = 7.8, R = 0.5) and the gravity parallel to the z-direction, the front combustion propagation direction. Solid lines represent the temperature isovalues, while dashed lines represent the streamfunction isovalues. The flow structure, generated by the hot spot, transports some heat to the fresh reactant. This preheating increases the combustion process, leading to a rise in the value of the hot spot at the center of the combustion front. When the hot spot at the center of the combustion front reaches the maximum value, it diffuses in the x-direction

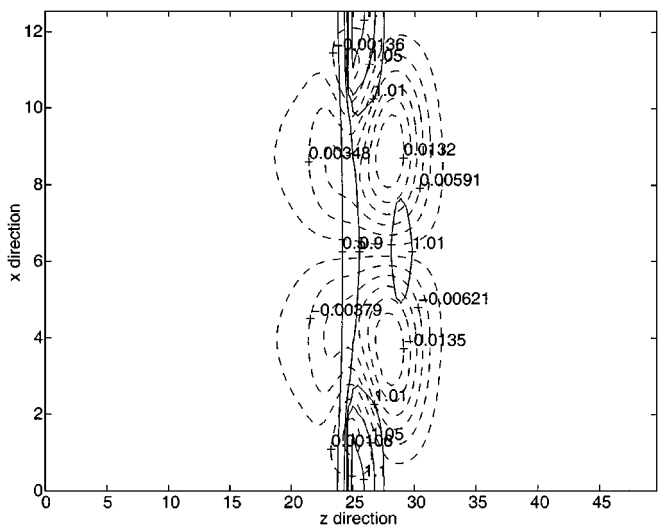

FIG. 6. Vertical case, (Z = 7.8, R = 0.5, $\theta = 0^\circ$), isovalues of T (solid lines) and Ψ (dash lines).

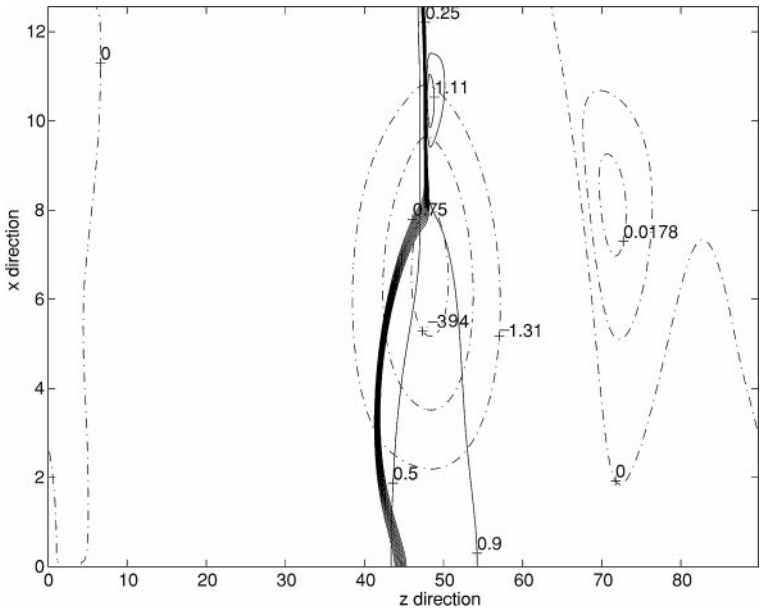

FIG. 7. Horizontal case, (Z = 7.8, R = 1.5, $\theta = 90^{\circ}$), isovalues of T (solid lines) and Ψ (dash lines).

and creates new flow structures with opposite spin as the previous one. This behavior of the solution agrees with those of the solution obtained in [8].

This solution exhibits a quasi-planar front of combustion. We develop the present methodology to dispense with the quasi-planar front hypothesis made in the methodology of [8]. Indeed, when the gravity made a nonzero angle with the front propagation, we do not have any guarantee that the front is still quasi-planar.

Figure 7 shows the effect of hydrodynamics on the structure of the flame front when the channel is horizontal and gravity vertical. Thin solid lines represent the temperature isovalues while dash lines represent the streamfunction 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 $N_z = 256$ Fourier modes on a physical domain of size $[0, 4\pi] \times [0, 90]$. The horizontal z-direction is the direction of propagation of the front. The Zeldovich number is equal to 6, the Rayleigh number is equal to 1, the Prandtl number is $P = 1$, the mass diffusion is given by $\epsilon = 0.02$, and the time step is set to $\Delta t = 0.01$. This solution is a traveling wave moving toward the left with a hot spot close to the top wall of the horizontal channel. The location of the hot spot and the front curvature of the concentration profile are closely related to the circular motion of the flow. This example shows that one can compute nonplanar flame front structure with local Fourier basis. We were not able to obtain this solution on reasonable elapse time with our previous method [8] because of the explicit character of the x dependency and the large number of subdomains needed in the z-direction.

7. CONCLUSIONS

We have developed and implemented a new methodology based on local Fourier approximation and the superposition principle to solve nonspace periodic time-dependent

PDE models. We illustrate the capabilities of the method on a reaction–diffusion system coupled to incompressible Navier–Stokes in two space dimensions that models frontal polymerization.

The accuracy of the local Fourier basis (LFB) is satisfactory compared to [8] for a quasi-planar front, but the method presented in this paper is more robust since it allows the computation of a nonplanar flame front.

The parallel efficiency of LFB follows from the introduction of a one-dimensional domain decomposition that avoids global transposition of matrices distributed on the network of processors. Artificial interface problems are defined adaptively depending on the wave number needed to minimize global communications.

The arithmetic complexity of the method is then dominated by subdomain's Fourier transforms. However, it should be noticed that LFB is sensitive to small time stepping because of the artificial boundary layer of square root of Δt thickness associated to so-called corrector terms. Postprocessing with smooth filters might be the appropriate way to overcome this difficulty [12]. In addition, the parallel scalability of the method is limited by the incompressible size of the extension of each subdomain used to build periodic extensions. In our opinion, LFB should be more attractive for large-scale computation in 3D with larger parallel systems than the one we have used. Finally, let us notice that this methodology is well designed to be used within $C(p, q, j)$ schemes [9]. $C(p, q, j)$ schemes are designed to solve systems of coupled PDEs, such as the target application presented here. They require the Fourier expansion of each PDE solution in order to adapt with respect to the wave number the delay in the exchange of the coupling terms between PDES. This allows us to relax the penalty on communication between distributed PDEs in a distributed memory parallel implementation. With the LFB methodology, the delay in the communication of the coupling terms can be set adaptively depending only on the PDE solution behavior on the subdomain.

REFERENCES

1. A. Averbuch, M. Israeli, and L. Vozovoi, A fast Poisson solver of arbitrary order accuracy in rectangular regions, *SIAM J. Sci. Comput.* **19**, 933 (1998).
2. A. Bayliss, D. Gottlieb, B. J. Matkowsky, and M. Minkoff, An adaptive pseudo-spectral method for reaction diffusion problems, *J. Comput. Phys.* **81**, 421 (1989).
3. M. Elghaoui and R. Pasquetti, Mixed spectral-boundary element embedding algorithms for the Navier-Stokes equations in the vorticity-stream function formulation, *J. Comput. Phys.* **153**, 82 (1996).
4. M. Elghaoui and R. Pasquetti, A spectral embedding method applied to the advection–diffusion equation, *J. Comput. Phys.* **125**, 464 (1996).
5. I. Foster and P. Worley, Parallel algorithms for the spectral transform method, *SIAM J. Sci. Comput.* **18**, 806 (1997).
6. M. Garbey, On some applications of the superposition principle with Fourier basis, *SIAM J. Sci. Comput.* **22**, 1087 (2000).
7. M. Garbey and D. Tromeur-Dervout, Massively parallel computation of stiff propagating fronts, *Combust. Theory Modelling* **1**, 271 (1997).
8. M. Garbey and D. Tromeur-Dervout, A new parallel solver for non periodic incompressible Navier–Stokes equations with Fourier basis: application to frontal polymerisation, *J. Comput. Phys.* **145**, 316 (1998).
9. M. Garbey and D. Tromeur-Dervout, A parallel adaptive coupling algorithm for systems of differential equations, *J. Comput. Phys.* **161**, 401 (2000).
10. M. Garbey and Hans G. Kaper, Asymptotic-numerical study of supersensitivity for generalized burgers equation, *SIAM J. Sci. Comput.* **22**, 368 (2000).

11. V. Girault, R. Glowinski, H. López, and J.-P. Vila, A boundary multiplier/fictitious domain method for the steady incompressible Navier–Stokes equations, *Numer. Math.* **88**, 75 (2001).
12. D. Gottlieb and Chi-Wang Shu, On the Gibbs phenomenon and its resolution, *SIAM Rev.* **39**, 644 (1997).
13. M. Israeli, L. Vozovoi, and A. Averbuch, Spectral multidomain technique with local Fourier basis. *J. Sci. Comput.* **8**, 135 (1993).
14. M. Israeli, L. Vozovoi, and A. Averbuch, Domain decomposition methods with local Fourier basis for parabolic problems, *Contemp. Math.* **157**, 223 (1994).
15. M. Israeli, L. Vozovoi, and A. Averbuch, Spectral multidomain technique with local Fourier basis. II. Decomposition into cells, *J. Sci. Comput.* **9**, 311 (1994).
16. J. A. Mackenzie, The efficient generation of simple two-dimensionnal adaptive grids, *SIAM J. Sci. Comput.* **19**, 1340 (1998).
17. S. B. Margolis, H. Kaper, G. Leaf, and B. Matkowsky, Bifurcation of pulsating and spinning reaction fronts in condensed two-phase combustion, *Combust. Sci. Technol.* **43**, 127 (1985).
18. L. S. Mulholland, W. Z. Huang, and D. M. Sloan, Pseudospectral solution of near-singular problems using numerical coordinate transformations based on adaptivity, *SIAM J. Sci. Comput.* **19**, 1261 (1998).
19. J. Nordström, N. Nordin, and D. Henningson, The fringe region technique and the Fourier method used in the direct numerical simulation of spatially evolving viscous flows, *SIAM J. Sci. Comput.* **20**, 1365 (1999).
20. I. P. Nagy, L. Sike, and J. A. Pojman, Thermochromic composite prepared via a propagating polymerization front, *J. Am. Chem. Soc.* (1995).