

DOMAIN DECOMPOSITION METHODS FOR FLUID DYNAMICS

F. Nataf¹, F. Rogier² and E. de Sturler³

¹CMAP, CNRS URA756, Ecole Polytechnique
91128 Palaiseau Cedex, France

²Division Calcul parallèle, ONERA, 29 Av. de la Division Leclerc
92322 Châtillon, France

³CSCS, ETH, Via Cantonale
6928 Manno, Switzerland

1. INTRODUCTION

In order to solve the Navier-Stokes equations, it is essential to have at one's disposal efficient numerical algorithms of solution of the linear convection-diffusion equations. Indeed, despite its nonlinear nature, the solving of the Navier-Stokes equations can be reduced to the solving of linear equations. Consider the incompressible Navier-Stokes equations in vorticity-stream function (ω, ψ) formulation in 2-D:

$$\begin{aligned}\frac{\partial \omega}{\partial t} + u \frac{\partial \omega}{\partial x} + v \frac{\partial \omega}{\partial y} - \nu \Delta \omega &= 0 \\ \Delta \psi &= -\omega \\ \begin{pmatrix} u \\ v \end{pmatrix} &= \text{rot } \psi\end{aligned}$$

In order to solve this system, a classical and widely used algorithm is the following. Let $(\omega^n, \psi^n, u^n, v^n)$ denote the approximation of (ω, ψ, u, v) at time $n\Delta t$. $(\omega^{n+1}, \psi^{n+1}, u^{n+1}, v^{n+1})$ is obtained by the solution of two linear boundary value problems and by the computation of a curl:

$$\frac{\omega^{n+1} - \omega^n}{\Delta t} + u \frac{\partial \omega^{n+1}}{\partial x} + v \frac{\partial \omega^{n+1}}{\partial y} - \nu \Delta \omega^{n+1} = 0$$

then,

$$\Delta \psi^{n+1} = -\omega^{n+1}$$

and

$$\begin{pmatrix} u^{n+1} \\ v^{n+1} \end{pmatrix} = \text{rot } \psi^{n+1}$$

The solution of the Navier-Stokes equations is thus reduced to the successive solutions of symmetric (Δ) and nonsymmetric (convection-diffusion) linear systems. More generally, the convection-diffusion equation

$$\mathcal{L}(C) = \frac{\partial C}{\partial t} + a \frac{\partial C}{\partial x} + b \frac{\partial C}{\partial y} - \nu \Delta C = f$$

plays an important role in numerous physical phenomena. It models the transport of a quantity C along a vector field (a, b) and its diffusion, proportional to a viscosity coefficient ν . In fluid dynamics, for instance, C may be the concentration of a dye or of a pollutant in water (river, sea, estuary, ...) or air (plume of smoke, ...).

We consider here the numerical solution on parallel computers (MIMD, Multiple Instruction Multiple Data) of the linear systems arising from the discretization of the convection-diffusion equation. These computers are new and present many interesting features. They are made up of many processors called nodes linked together by a communication network. Each node is a computer by itself in the sense where it has its own direct access memory. Each node can host its own program possibly distinct from the ones of the other nodes. The nodes can work concurrently. They communicate with each other thanks to messages they can send and receive. These messages can be used to synchronize the nodes and to send and receive data. The programming language is Fortran 77 or C except for the orders corresponding to the sending and receiving of messages (eg. csend, crecv). These machines are potentially very powerful. The benefits in terms of memory is obvious. In order to have an actual speed of computation close to the optimum (speed of a processor \times nbr of processors) it is necessary to have algorithms fitted to parallel machines.

Domain decomposition methods seem a natural and promising approach. Many works have been devoted to symmetric systems and have led to efficient methods. The discretization of the convection-diffusion equation leads to non symmetric linear systems of equations. For a large viscosity, the algorithms designed for symmetric systems can be used and perform well. Nevertheless, for convection dominated flows (small viscosity), these methods perform poorly and there are few theoretical results. In this work, we present a method based on the use of artificial boundary conditions. Indeed, the rate of convergence of domain decomposition methods is very sensitive to the choice of the interface conditions. The original Schwarz method is based on the use of Dirichlet boundary conditions. In order to increase the efficiency of the algorithm, it has been proposed to replace the Dirichlet boundary conditions with more general boundary conditions, see Lions (1989). In the usual Schur method, Dirichlet and Neumann boundary conditions are used. In Hagstrom, Tewarson and Jazcilevich (1988), they are replaced with artificial boundary conditions. More generally, it has been remarked that absorbing (or artificial) boundary conditions are a good choice (see, Hagstrom, Tewarson and Jazcilevich, 1988; Despres, 1991; Nataf, 1992; Givois, 1992, where such boundary conditions are used). In this report, we partially clarify this question.

In § 2.1, we set the problem to be solved and reformulate it as a problem whose unknowns are functions from the boundaries of the subdomains to \mathbb{R} . In § 2.2, we take as interface conditions exact artificial boundary conditions (ABC). We show that GMRES and BIGSTAB algorithms converge in a number of steps equal to the number of subdomains minus one. These exact ABC are difficult to use and we outline in § 2.3

how to approximate them by partial differential operators. In § 3, we show numerical results.

2. DOMAIN DECOMPOSITION METHOD FOR THE CONVECTION-DIFFUSION EQUATION

2.1. Reformulation of the problem

For sake of simplicity, we consider the continuous convection-diffusion equation set on the whole plane \mathbb{R}^2 :

$$\mathcal{L}(u) = f \text{ in } \mathbb{R}^2 \quad (2.1)$$

$$(2.2)$$

where f is a given function. The plane is decomposed into N vertical strips with or without overlap. We have $\mathbb{R}^2 = \cup_{i=1}^N \Omega_i$ where $\Omega_i =]t_i, L_i[\times \mathbb{R}$. In this §, we write an equivalent form of problem (2.1) where the unknowns are functions defined on the boundaries of the subdomains. Let $B_{i,r}, B_{i,l}$ be operators so that the following BVP are well-posed:

$$\begin{aligned} \mathcal{L}(v) &= f_i \text{ in } \Omega_i \\ B_{i,l}(v) &= g_{i,l} \text{ on } \{t_i\} \times \mathbb{R} \\ B_{i,r}(v) &= g_{i,r} \text{ on } \{L_i\} \times \mathbb{R} \end{aligned} \quad (2.3)$$

for any functions $f_i, g_{i,l}$ and $g_{i,r}$. We define the operator S_i by

$$v = S_i(g_{i,l}, g_{i,r}, f_i).$$

It is clear that the knowledge of $B_{i,l}(u), B_{i,r}(u), 1 \leq i \leq N$ enables to recover the value of u by solving the BVPs (2.3). Let us denote H the $2(N-1)$ -tuple $(B_{2,l}(u), \dots, B_{N,l}(u), B_{1,r}(u), \dots, B_{N-1,r}(u))$. It is possible to write a linear system for H . Indeed, we have for $B_{i,l}(u)$:

$$\begin{aligned} B_{i,l}(u) &= B_{i,l}(S_{i-1}(B_{i-1,l}(u), B_{i-1,r}(u), f)) \\ &= B_{i,l}(S_{i-1}(B_{i-1,l}(u), B_{i-1,r}(u), 0)) + B_{i,l}(S_{i-1}(0, 0, f)) \end{aligned}$$

and for $B_{i,r}(u)$ a similar relation. It is thus possible to define a linear operator T so that the previous linear system writes in a compact form:

$$(Id - T)(H) = G \quad (2.4)$$

where G can be easily computed and depends on f . When there is no overlap, system (2.4) is well-posed and when there is no overlap, one needs the extra condition that $B_{i,r} - B_{i+1,l}$ is invertible.

2.2. Choice of the interface conditions

The principle of substructuring methods (also called Schur methods) is to solve (2.4) with conjugate gradient like methods as GMRES or BIGSTAB. The speed of convergence will of course depend on T and thus on the choice of the interface conditions $B_{i,l}$ and $B_{i,r}$. We consider here the case where we take for $B_{i,l}$ and $B_{i,r}$ (resp. $B_{i,l}$) exact

artificial boundary conditions which are defined by $B_{l,r} = \partial_x - A_{l,r}$ (resp. $B_{r,l} = -\partial_x - A$) where $A_{l,r}$ is the Dirichlet to Neumann operator of the right (resp. left) half-plane $[L_{l,r}, \infty[\times \mathbb{R}$ (resp. $] - \infty, l[\times \mathbb{R}$). Then it can be seen that T is nilpotent operator of order $N - 1$, $T^{N-1} = 0$ (see Nataf, Rogier and de Sturler, 1994). We deduce from this that GMRES and BICGSTAB methods applied to (2.4) will converge in $N - 1$ steps while (2.4) is an infinite dimensional system. This shows that exact artificial boundary conditions are a very good choice as interface conditions.

Indeed, let H^0 be the initial approximation to the solution to (2.4). Let $r_0 = G - (Id - T)(H^0)$ be the initial residual. We seek for \hat{H} such that $H = H^0 + \hat{H}$ i.e. \hat{H} satisfies:

$$(Id - T)(\hat{H}) = r_0$$

The GMRES method minimizes the residual norm over the Krylov space $K^n(Id - T), r_0) \equiv \text{span}\{r_0, (Id - T)r_0, \dots, (Id - T)^{n-1}r_0\}$. Clearly, $\hat{H} \in K^{N-1}(Id - T), r_0)$ so that $N - 1$ iterations are necessary for the solution of (2.4). Thus, we have just proved

Proposition 2.1. *The GMRES algorithm applied to (2.4) converges in at most $N - 1$ steps.*

Let us now consider the convergence of Bi-CGSTAB Van der Vorst (1992) for the solution of the linear system (2.4). We shall see that

Proposition 2.2. *If there is no breakdown of Bi-CGSTAB, we have convergence of Bi-CGSTAB applied to (2.4) in at most $N - 1$ steps.*

Because Bi-CGSTAB is based on BiCG (Fletcher, 1976) we will first discuss the convergence of BiCG. We choose some $\tilde{r}_0 \neq 0$, (for example $\tilde{r}_0 = r_0$). Now the BiCG algorithm generates two sequences of polynomials, the residuals $r_i = P_i(Id - T)r_0$:

$$r_0, r_1, r_2, \dots$$

and $\tilde{r}_i = \tilde{P}_i((Id - T)^T)\tilde{r}_0$:

$$\tilde{r}_0, \tilde{r}_1, \tilde{r}_2, \dots$$

where P_i indicates a polynomial of degree i . These sequences satisfy the following relations (Fletcher, 1976):

$$r_i^T \tilde{r}_j = 0 \quad i \neq j \quad (2.5)$$

$$r_i^T \tilde{r}_i \neq 0 \quad (2.6)$$

If $r_i^T \tilde{r}_i = 0$ then BiCG would break down, but we will not discuss this problem here. For the residuals we have $r_i = P_i(Id - T)r_0 \in \text{span}\{r_0, (Id - T)r_0, (Id - T)^2r_0, \dots, (Id - T)^i r_0\} = K^{i+1}(Id - T, r_0)$, and furthermore we have $K^{i+1}((Id - T), r_0) = K^{i+1}(T, r_0)$. Together this gives

$$r_i \in K^{i+1}(T, r_0) \quad (2.7)$$

Proposition 2.3. *Let $\{r_0, r_1, \dots, r_{k-1}\}$ be independent and $r_k \in \text{span}\{r_0, r_1, \dots, r_{k-1}\}$, then $r_k = 0$ and BiCG converges in k steps.*

Although being similar, this property differs from the finite termination properties for BiCG (Fletcher, 1976), for CG if the operator is a low rank perturbation of the identity, which leads, as in this case, to convergence in a number of steps equal to the rank of the perturbation (Golub and Van Loan, 1989), and for GMRES Saad and

Schultz (1986). In these other cases, the residual is necessarily zero because it is both an element of and orthogonal to the same space, whereas the present property is derived from the residual being an element of one space and orthogonal to another, in principle completely different space.

Proof: For r_k we have the following two relations:

$$r_k \in \text{span}\{r_0, r_1, \dots, r_{k-1}\} \quad (2.8)$$

$$r_k \perp \text{span}\{\tilde{r}_0, \tilde{r}_1, \dots, \tilde{r}_{k-1}\} \quad (2.9)$$

So (2.8) implies $r_k = \sum_{i=0}^{k-1} \alpha_i r_i$, and then (2.9) gives

$$\forall j : 0 \leq j \leq k-1 : \tilde{r}_j^T \left(\sum_{i=0}^{k-1} \alpha_i r_i \right) = 0 \Leftrightarrow \sum_{i=0}^{k-1} \alpha_i \tilde{r}_j^T r_i = 0$$

Together with (2.5) this leads to

$$\forall j : 0 \leq j \leq k-1 : \alpha_j \tilde{r}_j^T r_j = 0,$$

which means, using (2.6), that $\alpha_j = 0$, $0 \leq j \leq k-1$. Therefore we have

$$r_k = 0,$$

and hence BiCG has converged.

Proposition 2.4. *For the linear system defined in (2.4) BiCG will converge in at most $N - 1$ iterations if there is no breakdown.*

Proof: From $T^{N-1} = 0$, we can derive that $K^N(T, r_0) = K^{N-1}(T, r_0)$. Together with (2.7) this leads to $r_{N-1} \in K^{N-1}(T, r_0)$, so that $r_{N-1} \in \text{span}\{r_0, r_1, \dots, r_{N-2}\}$. Proposition 2.3. then proves that $r_{N-1} = 0$, and therefore BiCG has converged. \square

Note that if the set $\{r_0, r_1, \dots, r_k\}$ becomes dependent before $k = N - 1$ BiCG will have converged as well.

It is not difficult to see that if the BiCG-residual $r_{N-1} = 0$, then also the Bi-CGSTAB-residual $r_{N-1}^{stab} = 0$. Bi-CGSTAB constructs its residual r_i^{stab} such as to be a polynomial of the form $r_i^{stab} = Q_i(Id - M)P_i(Id - M)r_0$, where $P_i(Id - M)r_0$ is still the BiCG-residual (Van der Vorst, 1992). So that, if the BiCG-residual $r_i = P_i(Id - T)r_0 = 0$, then also $r_i^{stab} = 0$, and Bi-CGSTAB will have converged as well.

Assuming that the norm of $T^{N-1}G$ is sufficiently large, the equality $H = \sum_{i=0}^{N-2} T^i G$ also indicates that GMRES cannot solve the set of equations (2.4) in less iterations than BiCG (however with half the number of matrix vector products).

2.3. Approximate artificial boundary conditions and DDM

We have seen that exact artificial boundary conditions lead to very interesting convergence properties. Unfortunately, they are difficult to use in a code. Indeed, operators $A_{l,r}$ or τ are not partial differential operators. Moreover, in general, we do not have an explicit form of these operators. Nevertheless, it is usually possible to approximate them by partial differential operators as it is done for approximating exact artificial boundary conditions (see e.g. Engquist and Majda, 1977, 1979). In this section, we explain how these exact artificial boundary conditions are approximated by local operators (i.e. partial differential operators). This enables us to write a Schur type formulation for an

arbitrary decomposition of the domain and to remove the restriction of a decomposition into vertical strips. In § 3, this strategy is applied to the convection-diffusion operator and numerical results are shown.

Our goal is to approximate at some point x_0 of the boundary of a subdomain the operators $A_{i,r}$ or L_i by partial differential operators. In order to be able to follow the strategy developed in Engquist and Majda (1977), we assume that the coefficients of the operator L vary slowly so that they can be approximated by their values at x_0 . By making use of the Fourier transform with respect to the tangential variable, we obtain an approximation of $A_{i,r}$ or L_i in the form of a convolution operator. This operator is itself approximated by a partial differential operator by approximating its symbol by a polynomial (for more details, see Nataf and Rogier, 1995; Nataf, 1993). In some cases, it is possible to make less restrictive assumptions (see e.g. in the context of absorbing boundary conditions or of paraxial approximations Bamberger, Engquist, Halpern and Joly, 1988a, 1988b; Lohéac, 1991; Nataf, Rogier and de Sturler, 1994; Halpern and Rauch, 1993).

We want to write a system analogous to system (2.4) but based on the approximate ABC. Since these operators are local, we are not restricted any more to decompositions into vertical strips. We will thus obtain a substructuring formulation which can be solved by conjugate gradient like methods. The resulting algorithm is what we call a Schur type algorithm (or substructuring methods).

Let Ω be a bounded open set of \mathbb{R}^2 . Let $\Omega_i, 1 \leq i \leq N$ be a finite sequence of sets embedded in Ω such that $\Omega = \cup_{i=1}^N \Omega_i$. Let $\Gamma = \partial\Omega$, $\Gamma_i = \partial\Omega_i - \Gamma$. The outward normal from Ω_i is \vec{n}_i and $\vec{\tau}_i$ is a tangential unit vector. Let us denote by $B_{i,r}, 1 \leq i \leq N$ the approximations to the exact ABC. Since the operators B_i are local, the subscript r or l is meaningless and will not be used here. We assume the operators $B_{i,r}, 1 \leq i \leq N$ to lead to well posed boundary value problems (see below BVP (2.10)). We assign to each subdomain i an operator S_i : Let f be a function from Ω_i to \mathbb{R} and h a function from Γ_i to \mathbb{R} , $S_i(h, f, g)$ is the solution v of the following boundary value problem:

$$\begin{aligned} \mathcal{L}(v) &= f(x), & x \in \Omega_i \\ B_i(v) &= h(x), & x \in \Gamma_i \\ \mathcal{L}(v) &= 0, & x \in \partial\Omega_i \cap \Gamma \end{aligned} \quad (2.10)$$

In order to take multiple overlaps into account, we introduce a sequence $(\eta_i^j), 1 \leq i \leq N, 1 \leq j \leq N, i \neq j$ of functions defined on the boundaries of the subdomains which satisfy:

- i) $\eta_i^j : \partial\Omega_i \rightarrow [0, 1]$
- ii) $\eta_i^j = 0$ on $\partial\Omega_i - \Omega_j$
- iii) $\sum_{j \neq i} \eta_i^j(x) = 1, \quad x \in \partial\Omega_i$

Remark 2.5. η_i^j is zero if $\partial\Omega_i \cap \Omega_j = \emptyset$.

It is now possible to write a substructuring formulation. Let u be the solution to (2.1) and $u_i = u|_{\Omega_i}$. We write a system for $B_i(u_i)$:

$$\begin{aligned} B_i(u_i) &= \sum_{j \neq i} \eta_i^j B_i(u_i) = \sum_{j \neq i} \eta_i^j B_i(u_j) \\ &= \sum_{j \neq i} \eta_i^j B_i(S_j(B_j(u_j), f_{\Omega_j}, g)) \\ &= \sum_{j \neq i} \eta_i^j B_i(S_j(0, f_{\Omega_j}, g)) + \sum_{j \neq i} \eta_i^j B_i(S_j(B_j(u_j), 0, 0)) \end{aligned}$$

Thus, $(B_i(u_i))_{1 \leq i \leq N}$ solves the following linear system:

$$B_i(u_i) - \sum_{j \neq i} \eta_i^j B_i(S_j(B_j(u_j), 0, 0)) = \sum_{j \neq i} \eta_i^j B_i(S_j(0, f_{\Omega_j}, g)), \quad 1 \leq i \leq N \quad (2.11)$$

Let $H = (H_i)_{1 \leq i \leq N}$ and $G = (G_i)_{1 \leq i \leq N}$ be the vectors

$$H = \begin{bmatrix} B_1(u_1) \\ \vdots \\ B_N(u_N) \end{bmatrix} \quad \text{and} \quad G = \begin{bmatrix} \sum_{j \neq 1} \eta_1^j B_1(S_j(0, f_{\Omega_j}, g)) \\ \vdots \\ \sum_{j \neq N} \eta_N^j B_N(S_j(0, f_{\Omega_j}, g)) \end{bmatrix}$$

and \mathcal{T} be the linear operator defined by

$$\mathcal{T}(H) = \begin{bmatrix} \sum_{j \neq 1} \eta_1^j B_1(S_j(B_j(u_j), 0, 0)) \\ \vdots \\ \sum_{j \neq N} \eta_N^j B_N(S_j(B_j(u_j), 0, 0)) \end{bmatrix}$$

System (2.11) may now be written in the following compact form:

$$(Id - \mathcal{T})(H) = G \quad (2.12)$$

We consider three algorithms for the solution of (2.12), GMRES, BiCGSTAB and Jacobi:

$$H^{n+1} = \mathcal{T}(H^n) + G$$

The last algorithm corresponds to the additive Schwarz method. Since the operator \mathcal{T} is no longer nilpotent, the Schwarz method should not converge in a finite number of steps. GMRES and BiCGSTAB (except if breakdown occurs) always converge in a finite number of steps (ignoring round-off errors) for a finite dimensional problem.

3. NUMERICAL RESULTS FOR THE CONVECTION-DIFFUSION EQUATION

We apply the strategy explained above to the convection-diffusion equation. Let

$$\mathcal{L} = \frac{1}{\Delta t} + \vec{a}(x, y) \cdot \frac{\partial}{\partial x} + b(x, y) \frac{\partial}{\partial y} - \nu \Delta \quad (3.1)$$

where $\vec{a} = (a, b)$ is the velocity field, ν is the viscosity, Δt is a constant which could correspond for instance to a time step for a backward-Euler scheme for the time dependent convection-diffusion equation.

For a subdomain Ω_i , the approximations to the exact ABC obtained using the method outlined in § 2.3 read as follows (\vec{a} is the velocity field (a, b) , \vec{n}_i is the outward normal from Ω_i and $\vec{\tau}_i$ is a tangential unit vector on $\partial\Omega_i$):

$$B_i^0 = \frac{\partial}{\partial \vec{n}_i} - \frac{\vec{a} \cdot \vec{n}_i - \sqrt{(\vec{a} \cdot \vec{n}_i)^2 + \frac{4\nu}{\Delta t}}}{2\nu} \quad (3.2)$$

or

$$B_i^0 = \frac{\partial}{\partial \vec{n}_i} - \frac{\vec{a} \cdot \vec{n}_i - \sqrt{(\vec{a} \cdot \vec{n}_i)^2 + \frac{4\nu}{\Delta t}}}{2\nu} + \frac{\vec{a} \cdot \vec{\tau}_i}{\sqrt{(\vec{a} \cdot \vec{n}_i)^2 + \frac{4\nu}{\Delta t}}} \frac{\partial}{\partial \vec{\tau}_i}$$

$$\frac{\nu}{\sqrt{(\bar{d}_i \bar{n}_i)^2 + \frac{\Delta t^2}{4}}} \left(1 + \frac{(\bar{d}_i \bar{n}_i)^2}{\sqrt{(\bar{d}_i \bar{n}_i)^2 + \frac{\Delta t^2}{4}}} \right) \frac{\partial^2}{\partial \bar{t}_i^2}$$

where the superscript denotes the order of the approximation, for more details see Nataf and Rogier (1995), Nataf (1993). The boundary conditions \mathcal{B}_1^0 or \mathcal{B}_2^0 are far field boundary conditions (also called Outflow B.C., Absorbing B.C., Artificial B.C., Radiation B.C., ..., see Engquist and Majda, 1977; Halpern, 1986).

We use a two-dimensional test problem to illustrate the validity of the method. We solve the following problem:

$$\begin{cases} \mathcal{L}(u) = \frac{\partial}{\partial t} + a(x, y) \frac{\partial}{\partial x} + b(x, y) \frac{\partial}{\partial y} - \nu \Delta u = 0, & 0 \leq x \leq 1, 0 \leq y \leq 1 \\ u(0, y) = 1, & 0 < y < 1 \\ \frac{\partial u}{\partial x}(x, 1) = 0, & 0 < x < 1 \\ \frac{\partial u}{\partial x}(1, y) = 0, & 0 < y < 1 \\ u(x, 0) = 0, & 0 < x < 1 \end{cases}$$

The operator \mathcal{L} is discretized by a standard upwind finite difference scheme of order 1 (see Fletcher (1991)) and $\mathcal{B}_i, 1 \leq i \leq N$ by a finite difference approximation. We used a rectangular finite difference grid. The mesh size is denoted by h . The unit square is decomposed into overlapping rectangles. The resulting discretization of system (2.12) is denoted by:

$$(Id - \mathcal{T}_h)(H_h) = G_h \quad (3.3)$$

The test problem has been implemented at ONERA on an IPSC860.

Remark 3.6. Any other discretization could be used as well.

From the definition of \mathcal{T}_h , we see that the computation of \mathcal{T}_h applied to some vector H_h amounts to the solution of N independent boundary value subproblems (one subproblem in each subdomain) which can be solved in parallel. We have considered three algorithms in order to solve (3.3): GMRES(∞), Bi-CGSTAB and a Jacobi algorithm:

$$H_h^{k+1} = \mathcal{T}_h(H_h^k) + G_h \quad (3.4)$$

which corresponds to an additive Schwarz method (ASM) whose convergence in the continuous case has been studied in Nataf and Rogier (1995) for outflow boundary conditions.

In tables 1 and 2, we give the number of subproblems solved so that the maximum of the error is smaller than 10^{-6} . One iteration of GMRES(∞) or of ASM counts for computing the solution for each subdomain once and one iteration of BiCGSTAB counts for computing the solution for each subdomain twice. In the tables, Id corresponds to the use of Id as interface condition (Dirichlet problems). The tests include the case $\mathcal{B}_i = Id$ since it corresponds to the classical Schwarz method when the Jacobi algorithm is used.

The results in Table 1 were obtained using the following parameters: 8×1 subdomains, 21×120 points in each subdomain, overlap = $2h$, $\nu = 0.1$, $\Delta t = 10^{40}$, $a = y$, $b = 0$.

Table 1: Computational cost vs. interface conditions and solvers

Boundary Cond.	ASM	Bi-CGSTAB	GMRES
Id	844	88	61
\mathcal{B}_0	86	38	33
\mathcal{B}_2	46	28	24

The results in Table 2 were obtained using the following parameters: 4×4 subdomains, 35×35 points in each subdomain, overlap = $2h$, $\nu = 0.1$, $\Delta t = 1$, $a = y$, $b = 0$.

Table 2: Computational cost vs. interface conditions and solvers

Boundary Cond.	ASM	Bi-CGSTAB	GMRES
Id	479	64	50
\mathcal{B}_0	27	22	19
\mathcal{B}_2	18	16	16

The use of outflow boundary conditions leads to a significant improvement whatever iterative solver is used. Bi-CGSTAB and GMRES give similar results with an advantage to GMRES in terms of computational cost and to BiCGSTAB in terms of storage requirements, since only two directions have to be stored.

REFERENCES

- DESPRES, B. (1991), *Domain Decomposition Method and the Helmholtz Problem*, Mathematical and Numerical aspects of wave propagation phenomena (SIAM), p. 44-52.
- BAMBERGER, A., ENGQUIST, B., HALPERN, L. and P. JOLY (1988a), Parabolic wave equation approximations in heterogeneous media, *SIAM J. Appl. Math.* 48 No 1, p. 99-128.
- BAMBERGER, A., ENGQUIST, B., HALPERN, L. and P. JOLY (1988b), Higher order paraxial wave approximations in heterogeneous media, *SIAM J. Appl. Math.* 48 No 1, p. 129-154.
- ENGQUIST, B. and A. MAJDA (1977), Absorbing Boundary Conditions for the Numerical Simulation of Waves, *Math. Comp.* 31 (139), p. 629-651.
- ENGQUIST, B. and A. MAJDA (1979), Radiation Boundary Conditions for Acoustic and Elastic Wave Calculations, *Comm. on Pure and Appl. Math.* vol. XXXII, p. 313-357.
- FLETCHER, C.A.J. (1991), *Computational Techniques for Fluid Dynamics* (Springer Series in Computational Physics).
- FLETCHER, R. (1976), *Conjugate gradient methods for indefinite systems in: WATSON, G.A. (ed.), Numerical Analysis Dundee 1975, Lecture Notes in Mathematics* 506, p. 73-89, (Berlin, Heidelberg, New York: Springer-Verlag).
- GIVOIS, E. (1992), *Etude et implémentation de deux méthodes de décomposition de domaines. Une approche monodimensionnelle pour l'initiation de la détonique à l'échelle moléculaire.* (Thèse de l'Université Paris Dauphine).
- GOLUB, G.H. and C.F. VAN LOAN (1989), *Matrix Computations* (2nd edition, John Hopkins University Press).

- HAGSTROM, T., TEWARSON, R.P. and A. JAZCLEVICH (1988), Numerical Experiments on a Domain Decomposition Algorithm for Nonlinear Elliptic Boundary Value Problems, *Appl. Math. Lett.* 1, No 3, p. 299-302.
- HALPERN, L. (1986), Artificial Boundary Conditions for the Advection-Diffusion Equations, *Math. Comp.* 174, p. 425-438.
- HALPERN, L. and J. RAUCH (1983), Absorbing Boundary Conditions for Diffusion Equations, *Prépublications Mathématiques de l'Université de Paris Nord* 93-05.
- LIONS, P.L. (1989), On the Schwarz Alternating Method III: A Variant for Nonoverlapping Subdomains, Third International Symposium on Domain Decomposition Methods for Partial Differential Equations, *SIAM*, p. 202-223.
- LOHÉAC, J.P. (1991), An Artificial Boundary Condition for an Advection-Diffusion Equation, *Math. Meth. in the Appl. Sci.* 14, p. 155-175.
- LOHÉAC, J.P., NATAF, F. and M. SCHATZMAN (1993), Parabolic Approximations of the Convection-Diffusion Equation, *Math. of Comp.* 60 (202), p. 515-530.
- NATAF, F. (1992), Méthodes de Schur généralisées pour l'équation d'advection-diffusion (Generalized Schur methods for the advection-diffusion equation), *C.R. Acad. Sci. Paris t. 314, Série I*, p. 419-422.
- NATAF, F. (1993), On the Use of Open Boundary Conditions in Block Gauss-Seidel Methods for the Convection-Diffusion Equations, *Rapport interne CMAP* 284.
- NATAF, F. and F. ROGIER (1995), Factorization of the Convection-Diffusion Operator and the Schwarz Algorithm, *M²AS* (to appear).
- NATAF, F., ROGIER, F. and E. DE STURLER (1994), Optimal interface conditions for domain decomposition methods, *Rapport interne n° 301*, CMAP (Ecole Polytechnique).
- SAAD, Y. and M. SCHULTZ (1986), GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, *SIAM J. Sci. Statist. Comput.* 7, p. 856-869.
- VAN DER VORST, H.A. (1992), A fast and smoothly converging variant of BI-CG for the solution of nonsymmetric linear systems, *SIAM J. Sci. Statist. Comput.* 13, p. 631-644.

A REGULARIZING PROPERTY OF ROTHE'S METHOD TO THE NAVIER-STOKES EQUATIONS

Reimund Rautmann

Universität Paderborn, Fachbereich Mathematik-Informatik,
Warburger Str. 100, D-33098 Paderborn, Germany

INTRODUCTION

The first order semi-discrete scheme introduced by Rothe (1930) approximates evolution problems by a sequence of boundary value problems for some resolvent equation, which have to be solved step by step. In case of the Navier-Stokes evolution equation the resolvent equation of some suitable linearization has the regularization properties of second order elliptic systems (of course, in dependence on the step length). Therefore on a bounded domain $\Omega \subset \mathbb{R}^3$, the convergence of Rothe's scheme in $L^2(\Omega)$ and its boundedness in $H^2(\Omega)$ implies its convergence even in $H^{2r}(\Omega)$ for any $r \in [2, \infty)$, as we will see below. Since we have established L^2 -convergence and H^2 -boundedness (or even H^2 -convergence) in Rautmann (1993b, 1993c) under suitable assumptions, our result in this note is boundedness and convergence of Rothe's scheme in $H^{2r}(\Omega)$ for all $r \in [2, \infty)$.

We will show elsewhere that our result also applies in the study of more refined schemes of the product formula type. Namely, convergence proofs for such schemes depend on $H^{1,\infty}$ -bounds for the flow velocity which follow from H^{2r} -bounds if $r > 3$ by Sobolev's imbedding theorem. Product formula approximations have firstly been proposed in Chorin (1973), Chorin, Hughes, Mc Cracken and Marsden (1978), Pironneau (1982), in order to overcome the destabilizing effect of the nonlinear transport term in the Navier-Stokes equation by the stabilizing influence of its elliptic part. Since then similar schemes have been studied by several authors, see the references in the recent publications Beale and Greengard (1992) and Rautmann and Masuda (1994), where explicit convergence rates for different splitting schemes of this type have been proved.

Having explained the notations in Section 1 we formulate our result in Theorem 1.2. For a linearized Rothe scheme introduced in Section 2, in Section 3 we establish L^r -bounds for its right hand side by means of a former result on H^2 -boundedness (which we have recalled in Theorem 1.1 of Section 1). From this in Section 4 we find the existence of Rothe approximations in H^{2r} . Then due to Solonnikov, Miyakawa, Giga