Int. J. Appl. Math. Comput. Sci., 2006, Vol. 16, No. 4, 419–429

OPERATOR–SPLITTING AND LAGRANGE MULTIPLIER DOMAIN DECOMPOSITION METHODS FOR NUMERICAL SIMULATION OF TWO COUPLED NAVIER–STOKES FLUIDS

DIDIER BRESCH*, JONAS KOKO**

* IMAG-LMC CNRS UMR 5523, BP 53
F–38041 Grenoble cedex, France

** LIMOS, Université Blaise Pascal – CNRS UMR 6158
ISIMA, Campus des Cézeaux – BP 10125,
63173 Aubière cedex, France
e-mail: koko@isima.fr

We present a numerical simulation of two coupled Navier-Stokes flows, using operator-splitting and optimization-based non-overlapping domain decomposition methods. The model problem consists of two Navier-Stokes fluids coupled, through a common interface, by a nonlinear transmission condition. Numerical experiments are carried out with two coupled fluids; one with an initial linear profile and the other in rest. As expected, the transmission condition generates a recirculation within the fluid in rest.

Keywords: domain decomposition, duality, conjugate gradient, Navier-Stokes flows

1. Introduction

This paper deals with a numerical simulation of coupled Navier-Stokes fluids by operator-splitting and domain decomposition methods. The fluids are coupled through a common interface by a nonlinear transmission condition. Models of two coupled flows are motivated by the study of geophysical flows like ocean and atmosphere or two layers of a stratified fluid, e.g., see (Lewandowski, 1997, Chapters 1 and 3) or (Lions *et al.*, 1993).

Coupled systems with a transmission condition are used to reduce the high computational cost associated with the discretization of the interface. The interface transmission condition is generally obtained by asymptotic analysis as a limit model when the thickness of the mixed layer goes to zero. The transmission condition can express various physical phenomena: adhesion (e.g., bonded structures (Bresch and Koko, 2004; Suquet, 1988)), friction (e.g., air/water (Bernardi *et al.*, 2002; 2003; 2004; Lions *et al.*, 1993)), the coupling of free-surface and ground water flows (Miglio *et al.*, 2003), etc. In the context of two coupled systems with an interface, domain decomposition methods apply in a natural way, since subdomains are already defined.

Using an operator-splitting method, at each step, we solve one coupled degenerated Stokes problem (first substep) and one uncoupled linear advection-diffusion problem (second substep). The transmission condition admitting a primitive, the problem of two coupled degenerated Stokes fluids can be reformulated as a constrained minimization problem with a strictly convex objective functional and a linear constraint (the divergence free condition). After introducing a fictitious unknown, we use saddle-point theory to derive three domain decomposition algorithms of the Uzawa/conjugate gradient type:

- DDM/S, at each iteration we solve two uncoupled degenerated Stokes problems,
- DDM/P1, at each iteration we solve two uncoupled vector Poisson problems (i.e., four uncoupled scalar Poisson equations),
- DDM/P2, the preconditioned version of DDM/P1.

In Algorithms DDM/P1 and DDM/P2, the pressure is computed iteratively as a Lagrange multiplier associated with the divergence free condition.

We refer to (Quarteroni and Valli, 1999) and the references therein for domain decomposition methods in general and to (Bresch and Koko, 2004; Du, 2001; Du and Gunzburger, 2000; Gunzburger and Lee, 2000; Gunzburger and Peterson, 1999; Koko, 2002) for optimization-based domain decomposition methods. For operator splitting methods, we refer interested readers, e.g., to (Glowinski, 2003; Glowinski and Le Tallec, 1989; Marchuk, 1990).

The paper is organized as follows: In Section 2 we present the model of two coupled Navier-Stokes flows. The finite element discretization is outlined in Section 3, followed by the operator-splitting scheme in Section 4. Solution methods for each substep are presented in Sections 5 and 6. In Section 7 we present some numerical experiments.

2. A Model of Coupled Navier-Stokes Fluids

Let Ω_1 and Ω_2 be adjacent bounded domains in $\mathbb{R}^2$ with a common boundary $S = \partial\Omega_1 \cap \partial\Omega_2$, as illustrated in Fig. 1. We set $\Gamma_i = \partial\Omega_i \setminus S$, for $i = 1, 2$. The interface S is assumed to be fixed and the *rigid lid hypothesis* is assumed: S is a mean interface and the values of $u_{i|S}$ and $p_{i|S}$ are mean values of the velocity and the pressure, respectively.

The generic point in $\mathbb{R}^2$ is denoted by $\boldsymbol{x} = (x, z)$. For simplicity, we assume that the interface coincides with $z = 0$. Therefore, Ω_1 and Ω_2 are contained respectively in the half-spaces $z > 0$ and $z < 0$. In the real physical context, the horizontal dimensions are much larger than the vertical one.

We assume that each subdomain Ω_i is filled with an incompressible viscous fluid with the viscosity parameter $\nu_i > 0$. In what follows, the vector field u_i stands for the velocity of the fluid in Ω_i and p_i signifies its pressure. We set $\boldsymbol{u} = (u_1, u_2)$ and $\boldsymbol{p} = (p_1, p_2)$ as the velocity and the pressure in the coupled system, respectively. We consider the following model of two Navier-Stokes fluids coupled by a nonlinear transmission condition on the interface S:

$$\frac{\partial u_i}{\partial t} - \nu_i \Delta u_i + (u_i \cdot \nabla) u_i + \nabla p_i = f_i, \quad \text{in } \Omega_i \times (0,T), \quad i = 1,2, \tag{1}$$

$$\nabla \cdot u_i = 0, \quad \text{in } \Omega_i \times (0,T), \quad i = 1,2, \tag{2}$$

$$\nu_i \frac{\partial u_i}{\partial n_i} - p_i n_i = (-1)^i \kappa \big|[\boldsymbol{u}]\big|[\boldsymbol{u}], \quad \text{on } S \times (0,T), \quad i = 1,2, \tag{3}$$

$$u_i = g_i, \quad \text{on } \Gamma_i \times (0,T), \quad i = 1,2, \tag{4}$$

$$u_i(x,0) = u_i^0, \quad \text{in } \Omega_i, \quad i = 1,2, \tag{5}$$

where $[\boldsymbol{u}] = (u_1 - u_2)|_S$, n_i is the unit outward normal to Ω_i, $\kappa > 0$ is the (positive) traction coefficient and $|\cdot|$ is the usual Euclidean norm. The *rigid lid hypothesis* is assumed: S is a mean interface and the values of u_i and p_i are mean values of the velocity and the pressure. The mixed layer of the two Navier-Stokes flows is modelled by the nonlinear equation (3) (see, e.g., Lewandowski, 1997).

The model (1)–(5) is an extension to Navier-Stokes fluids of the model of two coupled Stokes fluids by Koko (2006). The model studied in Koko (2006) is derived from the model by Bernardi *et al.* (2002; 2003; 2004) for two turbulent fluids separated by a fixed interface.

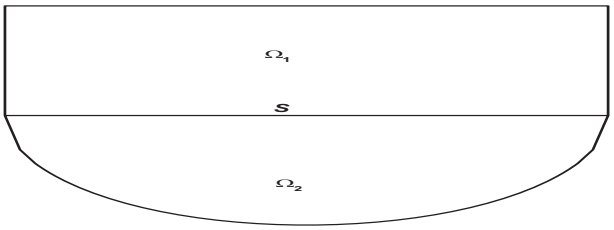

Fig. 1. System of two coupled – Navier-Stokes flows: Ω_1, Ω_2 Navier-Stokes fluids; S – fixed interface.

Remark 1. With the assumption that S coincides with $z = 0$, the transmission condition (3) can be replaced by

$$\nu_i \partial_z u_{iH} = (-1)^i \kappa \big|u_{1H} - u_{2H}\big|(u_{1H} - u_{2H}), \quad u_{iV} = 0 \quad \text{on } S, \tag{6}$$

where u_{iH} and u_{iV} represent the horizontal and vertical components of the velocity fields, respectively.

Remark 2. The transmission condition (3) is a special case of the general transmission condition

$$\nu_i \frac{\partial u_i}{\partial n_i} - p_i n_i = (-1)^i \kappa \big|[\boldsymbol{u}]\big|^{\beta-2}[\boldsymbol{u}], \quad \beta \geq 2,$$

studied in (Bresch and Koko, 2004; Suquet, 1988) for bounded structures.

Let $H^1(\mathcal{D})$ denote the standard Hilbert space, with respect to a domain $\mathcal{D}$. We define the function spaces (for $i = 1, 2$):

$$V_0^i = \left\{ v \in \left(H^1(\Omega_i)\right)^2;\ v = 0 \text{ on } \Gamma_i \right\},\ \mathbf{V}_0 = V_0^1 \times V_0^2,$$

$$V_g^i = \left\{ v \in \left(H^1(\Omega_i)\right)^2;\ v = g_i \text{ on } \Gamma_i \right\},\ \mathbf{V}_g = V_g^1 \times V_g^2$$

and forms (for $i = 1, 2$):

$$a_i(u,v) = \nu_i \int_{\Omega_i} \nabla u \cdot \nabla v \,\mathrm{d}x, \quad u, v \in V_i,$$

$$b_i(u,p) = \int_{\Omega_i} p \nabla \cdot u \,\mathrm{d}x, \quad (u,p) \in V_i \times L^2(\Omega_i),$$

$$c_i(w,u,v) = \int_{\Omega_i} (w \nabla \cdot) u v \,\mathrm{d}x, \quad u, v, w \in V_i,$$

$$(f,v)_{\Omega_i} = \int_{\Omega_i} f v \,\mathrm{d}x, \quad (v,f) \in V_i \times L^2(\Omega_i).$$

We set

$$\mathbf{H^1}(\boldsymbol{\Omega}) = H^1(\Omega) \times H^1(\Omega),$$

$$\mathbf{L^2}(\boldsymbol{\Omega}) = L^2(\Omega_1) \times L^2(\Omega_2),$$

$$\boldsymbol{f} = (f_1, f_2)$$

and, for $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}$ in $\mathbf{H^1}(\boldsymbol{\Omega})$, we define

$$(\boldsymbol{f}, \boldsymbol{v}) = \sum_{i=1}^{2} (f_i, v_i)_{\Omega_i}, \tag{7}$$

$$\boldsymbol{a}(\boldsymbol{u}, \boldsymbol{v}) = \sum_{i=1}^{2} a_i(u_i, v_i), \tag{8}$$

$$\boldsymbol{b}(\boldsymbol{u}, \boldsymbol{p}) = \sum_{i=1}^{2} b_i(u_i, p_i), \tag{9}$$

$$\boldsymbol{c}(\boldsymbol{w}, \boldsymbol{u}, \boldsymbol{v}) = \sum_{i=1}^{2} c_i(w_i, u_i, v_i). \tag{10}$$

The scalar product on S is denoted by

$$(\lambda, \mu)_S = \int_S \lambda \mu \, \mathrm{d}s,$$

with λ and μ belonging to suitable vector spaces.

With the notation above, the problem (1)–(5) can be rewritten as the following system of coupled variational equations:

Find $(u_i, p_i) \in V_g^i \times L^2(\Omega_i)$, a.e. on $(0, T)$, such that

$$\begin{aligned}(\partial_t u_i, v) + a_i(u_i, v) + c_i(u_i, u_i, v) - b_i(v, p_i) \\ +(-1)^{i+1}\kappa \left(\big|[\boldsymbol{u}]\big|[\boldsymbol{u}], v\right)_S = (f_i, v), \ \ \forall v \in V_0^i, \\ -b_i(u_i, q) = 0, \quad \forall q \in L^2(\Omega_i),\end{aligned}$$

or equivalently, using the notation (7)–(10):

Find $(\boldsymbol{u}, \boldsymbol{p}) \in \mathbf{V}_g \times \mathbf{L^2}(\boldsymbol{\Omega})$, a.e. on $(0, T)$, such that

$$\begin{aligned}(\partial_t \boldsymbol{u}, \boldsymbol{v}) + \boldsymbol{a}(\boldsymbol{u}, \boldsymbol{v}) + \boldsymbol{c}(\boldsymbol{u}, \boldsymbol{u}, \boldsymbol{v}) - \boldsymbol{b}(\boldsymbol{v}, \boldsymbol{p}) \\ +\kappa \left(\big|[\boldsymbol{u}]\big|[\boldsymbol{u}], [\boldsymbol{v}]\right)_S = (\boldsymbol{f}, \boldsymbol{v}), \ \ \forall \boldsymbol{v} \in \mathbf{V}_0, \qquad (11) \\ -\boldsymbol{b}(\boldsymbol{u}, \boldsymbol{q}) = 0, \qquad \forall \boldsymbol{q} \in \mathbf{L^2}(\boldsymbol{\Omega}). \quad (12)\end{aligned}$$

3. Finite Element Approximation

We assume that each Ω_i is polygonal and therefore can be exactly triangulated. We consider $\mathcal{T}_h^i$, a finite element triangulation of Ω_i, and $\mathcal{T}_{2h}^i$, a triangulation twice coarser, with the following assumptions:

(a) The meshes constitute a regular family of triangulations, in the usual sense of the finite element method, cf. (Ciarlet, 1979).

(b) The meshes are compatible on S, in the sense that

$$\mathcal{T}_h^1 \cap S = \mathcal{T}_h^2 \cap S, \quad \mathcal{T}_{2h}^1 \cap S = \mathcal{T}_{2h}^2 \cap S.$$

In practice, $\mathcal{T}_{2h}^i$ is constructed first and then $\mathcal{T}_h^i$ is formed by joining the midpoints of the edges of $\mathcal{T}_{2h}^i$. Each triangle of $\mathcal{T}_{2h}^i$ is therefore divided into four triangles of $\mathcal{T}_h^i$ as shown in Fig. 2.

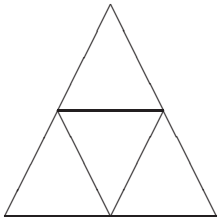

Fig. 2. Subdivision of a triangle of $\mathcal{T}_{2h}^i$.

For discrete velocity-pressure spaces, we use the P1-iso-P2/P1 element. These spaces are well known to satisfy the discrete Babuska-Brezzi inf-sup condition. Let $\mathcal{P}_1$ be the space of polynomials in two variables of degree ≤ 1. We define the following finite element spaces which approximate V_0^i, V_g^i and $L^2(\Omega_i)$, respectively:

$$\begin{aligned}V_{0h}^i = \Big\{ & v_h \in \mathcal{C}^0(\Omega_i)^2, \ v_h|_T \in \mathcal{P}_1, \ \forall T \in \mathcal{T}_h^i, \\ & v_h = 0 \text{ on } \Gamma_i \Big\},\end{aligned}$$

$$\begin{aligned}V_{gh}^i = \Big\{ & v_h \in \mathcal{C}^0(\Omega_i)^2, \ v_h|_T \in \mathcal{P}_1, \ \forall T \in \mathcal{T}_h^i, \\ & v_h = g_h \text{ on } \Gamma_i \Big\},\end{aligned}$$

$$P_h^i = \Big\{ p_h \in \mathcal{C}^0(\Omega_i), \ p_h|_T \in \mathcal{P}_1, \ \forall T \in \mathcal{T}_{2h}^i \Big\}.$$

For simplicity, we set

$$\mathbf{V}_{0h} = V_{0h}^1 \times V_{0h}^1, \ \ \mathbf{V}_{gh} = V_{gh}^1 \times V_{gh}^1, \ \ \mathbf{P}_h = P_h^1 \times P_h^2.$$

The compatibility condition (b) induces on S a decomposition into intervals. The function space $L^2(S)$ is therefore approximated by piece-wise linear polynomials in one variable:

$$\begin{aligned}\Lambda_h = \Big\{ & \lambda_h \in \mathcal{C}^0(S), \ \lambda_h|_I \in \mathcal{P}_1, \\ & \forall I \in S \cap \mathcal{T}_h^1 = S \cap \mathcal{T}_h^2 \Big\}.\end{aligned}$$

With the above results, the finite element approximation of (11)–(12) is

Find $(\boldsymbol{u}_h, \boldsymbol{p}_h) \in \mathbf{V}_{gh} \times \mathbf{P}_h$, a.e. on $(0, T)$, such that

$$\begin{aligned}(\partial_t \boldsymbol{u}_h, \boldsymbol{v}_h) + \boldsymbol{a}(\boldsymbol{u}_h, \boldsymbol{v}_h) + \boldsymbol{c}(\boldsymbol{u}_h, \boldsymbol{u}_h, \boldsymbol{v}_h) - \boldsymbol{b}(\boldsymbol{v}_h, \boldsymbol{p}_h) \\ +\kappa \left(\big|[\boldsymbol{u}_h]\big|[\boldsymbol{u}_h], \boldsymbol{v}_h\right)_S = (\boldsymbol{f}, \boldsymbol{v}_h), \quad \forall \boldsymbol{v}_h \in \mathbf{V}_{0h}, \ (13) \\ -\boldsymbol{b}(\boldsymbol{u}_h, \boldsymbol{q}_h) = 0, \qquad \forall \boldsymbol{q}_h \in \mathbf{P}_h. \quad (14)\end{aligned}$$

4. Operator-Splitting Scheme

This section is concerned with time discretization of (13)–(14). It is quite natural to discretize this initial value problem by a method taking advantage of the separability of incompressibility, diffusion, advection and subdomains coupling. Such a goal can be achieved by an operator-splitting scheme. We use an L^2-projection method with an operator-splitting scheme à la Marchuk-Yanenko (see, e.g., Glowinski, 2003; Marchuk, 1990), where incompressibility and subdomains coupling is treated by an L^2-projection method, and advection and diffusion are coupled together. Operator-splitting à la Marchuk-Yanenko has been applied successfully to numerical simulation of flows around moving rigid bodies (see, e.g., Glowinski, 2003; Glowinski *et al.*, 1998; 2000; Pan *et al.*, 2005) or particulate flows (see, e.g., Glowinski, 2003; Glowinski *et al.*, 2000).

The time interval $(0,T)$ is divided into N subintervals of equal length $\delta t = T/N$, and we set $\boldsymbol{u}_h^n = \boldsymbol{u}_h(t_n)$ and $\boldsymbol{p}_h^n = \boldsymbol{p}_h(t_n)$, where $t_n = n\delta t$ for $n = 0,\ldots,N$. Given $(\boldsymbol{u}_h^n, \boldsymbol{p}_h^n)$, we compute $(\boldsymbol{u}_h^{n+1}, \boldsymbol{p}_h^{n+1})$ via the following two substep scheme:

Substep 1. Compute $(\boldsymbol{u}_h^{n+1/2}, \boldsymbol{p}_h^{n+1}) \in \mathbf{V}_{gh} \times \mathbf{P}_h$ via

$$\frac{1}{\delta t}(\boldsymbol{u}_h^{n+1/2}, \boldsymbol{v}_h) - \boldsymbol{b}(\boldsymbol{v}_h, \boldsymbol{p}_h^{n+1}) + \kappa\left(\left|[\boldsymbol{u}_h^n]\right|[\boldsymbol{u}_h^{n+1/2}], [\boldsymbol{v}_h]\right)_S = \frac{1}{\delta t}(\boldsymbol{u}_h^n, \boldsymbol{v}_h), \quad \forall \boldsymbol{v}_h \in \mathbf{V}_{0h}, \tag{15}$$

$$-\boldsymbol{b}(\boldsymbol{u}_h, \boldsymbol{q}_h) = 0, \quad \forall \boldsymbol{q}_h \in \mathbf{P}_h. \tag{16}$$

Substep 2. Compute $\boldsymbol{u}_h^{n+1} \in \mathbf{V}_{gh}$ via

$$\frac{1}{\delta t}(\boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h) + \boldsymbol{a}(\boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h) + \boldsymbol{c}(\boldsymbol{u}_h^{n+1/2}, \boldsymbol{u}_h^{n+1}, \boldsymbol{v}_h) = (\boldsymbol{f}^{n+1}, \boldsymbol{v}_h) + \frac{1}{\delta t}(\boldsymbol{u}_h^{n+1/2}, \boldsymbol{v}_h), \ \forall \boldsymbol{v}_h \in \mathbf{V}_{0h}. \tag{17}$$

The problem (15)–(16) makes no sense, in general, since the boundary condition $\boldsymbol{u}_h^{n+1/2} = \boldsymbol{g}_h^{n+1}$ is "too strong" for a solution which does not have H^1-regularity. But numerical experiments (Glowinski, 2003, Ch. 7) show that the solutions of (15)–(16) obtained with the boundary condition $\boldsymbol{u}_h^{n+1/2} = \boldsymbol{g}_h^{n+1}$ are more accurate than those obtained with $\boldsymbol{u}_h^{n+1/2} \cdot \boldsymbol{n} = \boldsymbol{g}_h^{n+1} \cdot \boldsymbol{n}$.

5. Solution of the Subproblem (17)

The problem (17) has an interesting computational property: it is equivalent to uncoupled linear advection-diffusion problems:

Find $u_i^{n+1} \in V_{gh}^i$, $1 \le i \le 2$, such that

$$\frac{1}{\delta t}(u_{ih}^{n+1}, v_h)_{\Omega_i} + a_i(u_{ih}^{n+1}, v_h) + c_i(u_{ih}^{n+1/2}, u_{ih}^{n+1}, v_h) = (f_i^{n+1}, v_h)_{\Omega_i} + \frac{1}{\delta t}(u_{ih}^{n+1/2}, v_h)_{\Omega_i}, \quad \forall v_h \in V_{0h}^i. \tag{18}$$

The parallelization of the substep 2 is therefore obvious.

The linear advection-diffusion problem (18) can be solved by an iterative linear solver for nonsymmetric systems like the GMRES method. But we prefer the least squares/conjugate gradient method based on iterative solution of Poisson problems (Glowinski, 2003, Ch. 3). For a subdomain i, define the objective functional (after dropping time related superscripts and setting $\boldsymbol{u}_h^* = \boldsymbol{u}_h^{n+1/2}$):

$$F_i(u_h) = \frac{1}{2\delta t}(z_h, z_h)_{\Omega_i} + \frac{1}{2}a_i(z_h, z_h),$$

where $z_h \in V_{0h}^i$ is the solution to the Poisson problem

$$\frac{1}{\delta t}(z_h, v_h)_{\Omega_i} + a_i(z_h, v_h) = \frac{1}{\delta t}(u_h, v_h)_{\Omega_i} + a_i(u_h, v_h) + c_i(\boldsymbol{u}_h^*, u_h, v_h) - (f_i, v_h)_{\Omega_i} - \frac{1}{\delta t}(u_h^*, v_h)_{\Omega_i}, \quad \forall v_h \in V_{0h}^i. \tag{19}$$

Note that (19) reduces to two (uncoupled) scalar Poisson problems. The least-squares problem for the linear advection diffusion problem (18) is given by

$$u_{ih} \in V_{gh}^i, \quad F_i(u_{ih}) \le F_i(v_h), \quad \forall v_h \in V_{gh}^i, \ i = 1, 2. \tag{20}$$

The conjugate gradient algorithm for solving the least-squares problem (20) is outlined in (Glowinski, 2003, Ch. 3). Each iteration of the least-squares conjugate gradient method requires solving two Poisson problems (sensitivity and adjoint problems) of the form (19).

6. Solution of the Subproblem (15)–(16)

After dropping time related superscripts and setting $\boldsymbol{u}_h^* = \boldsymbol{u}_h^n$, let us introduce the functionals

$$J(\boldsymbol{u}_h) = \frac{1}{2\delta t}(\boldsymbol{u}_h, \boldsymbol{u}_h) - \frac{1}{\delta t}(\boldsymbol{u}_h^*, \boldsymbol{u}_h), \quad \forall \boldsymbol{u}_h \in \mathbf{V}_{gh},$$

$$\Phi(\phi_h) = \frac{1}{2}\int_S \tilde{\kappa}\left|\phi_h\right|^2 \mathrm{d}s, \qquad \forall \phi_h \in \Lambda_h,$$

where $\tilde{\kappa} = \kappa|[\boldsymbol{u}_h^*]|$. The subproblem (15)–(16) is the saddle-point formulation of the following constrained minimization problem:

Find $\boldsymbol{u}_h \in \mathbf{V}_{gh}$, such that

$$J(\boldsymbol{u}_h) + \Phi([\boldsymbol{u}_h]) \le J(\boldsymbol{v}_h) + \Phi([\boldsymbol{v}_h]), \ \ \forall \boldsymbol{v}_h \in \mathbf{V}_{gh}, \quad (21)$$

$$\nabla \cdot u_{ih} = 0 \ \text{ in } \ \Omega_i, \quad i = 1, 2, \quad (22)$$

with $\boldsymbol{p}_h$ as the Lagrange multiplier associated with the divergence constraint (22). If we set

$$F(\boldsymbol{u}_h, \phi_h) = J(\boldsymbol{u}_h) + \Phi(\phi_h), \quad \forall (\boldsymbol{u}_h, \phi_h) \in \mathbf{V}_{gh} \times \Lambda_h,$$

it is clear that (21)–(22) is equivalent to the following constrained minimization problem:

Find $(\boldsymbol{u}_h, \phi_h) \in \mathbf{V}_{gh} \times \Lambda_h$, such that

$$F(\boldsymbol{u}_h, \phi_h) \le F(\boldsymbol{v}_h, \varphi_h), \ \ \forall (\boldsymbol{v}_h, \varphi_h) \in \mathbf{V}_{gh} \times \Lambda_h, \quad (23)$$

$$\nabla \cdot u_{ih} = 0, \ \text{ in } \ \Omega_i, \ i = 1, 2, \quad (24)$$

$$\phi_h - [\boldsymbol{u}_h] = 0, \ \text{ on } \ S. \quad (25)$$

Our domain decomposition method is based on solving (23)–(25) by a Lagrange multiplier method. The Lagrangian functional associated with the constrained minimization problem (23)–(25) is given by

$$\mathcal{L}(\boldsymbol{u}_h, \phi_h, \boldsymbol{p}_h, \lambda_h) = F(\boldsymbol{u}_h, \phi_h) - \boldsymbol{b}(\boldsymbol{u}_h, \boldsymbol{p}_h) - (\lambda_h, \phi_h - [\boldsymbol{u}_h])_S, \quad (26)$$

with the Lagrange multiplier λ_h in Λ_h. Setting

$$\mathbf{X}_h = \mathbf{V}_{gh} \times \Lambda_h, \quad \mathbf{M}_h = \mathbf{P}_h \times \Lambda_h,$$

the solution to (23)–(25) reduces to the following saddle-point problem:

Find $(\boldsymbol{u}_h, \phi_h, \boldsymbol{p}_h, \lambda_h) \in \mathbf{X}_h \times \mathbf{M}_h$ such that

$$\mathcal{L}(\boldsymbol{u}_h, \phi_h, \boldsymbol{q}_h, \mu_h) \le \mathcal{L}(\boldsymbol{v}_h, \varphi_h, \boldsymbol{q}_h, \mu_h) \le \mathcal{L}(\boldsymbol{v}_h, \varphi_h, \boldsymbol{p}_h, \lambda_h), \quad \forall (\boldsymbol{v}_h, \varphi_h, \boldsymbol{q}_h, \mu_h) \in \mathbf{X}_h \times \mathbf{M}_h. \quad (27)$$

Solving (27) is equivalent to solving the following (uncoupled) saddle-point equations:

Find $(\boldsymbol{u}_h, \phi_h, \boldsymbol{p}_h, \lambda_h) \in \mathbf{X}_h \times \mathbf{M}_h$ such that

$$\frac{1}{\delta t}(\boldsymbol{u}_h, \boldsymbol{v}_h) - \boldsymbol{b}(\boldsymbol{v}_h, \boldsymbol{p}_h) = \frac{1}{\delta t}(\boldsymbol{u}_h^*, \boldsymbol{v}_h) - (\lambda_h, [\boldsymbol{v}_h])_S, \quad \forall \boldsymbol{v}_h \in \mathbf{V}_{0h}, \quad (28)$$

$$-b_i(u_{ih}, q_h) = 0, \ \forall q_h \in P_h^i, \ i = 1, 2, \quad (29)$$

$$(\tilde{\kappa}\phi_h, \varphi_h)_S = (\lambda_h, \varphi_h), \quad \forall \varphi_h \in \Lambda_h, \quad (30)$$

$$(\mu_h, \lambda_h - [\boldsymbol{u}_h])_S = 0, \quad \forall \mu_h \in \Lambda_h. \quad (31)$$

The main advantage of the saddle-point equations (28)–(31) is that, for a known λ_h, Eqn. (28) is uncoupled. From (30), we deduce that

$$\phi_h = \kappa^* \lambda_h,$$

where $\kappa^* = \tilde{\kappa}^{-1} = (\kappa|[\boldsymbol{u}_h^*]|)^{-1}$. Substituting ϕ_h into (26), we obtain the reduced Lagrangian functional

$$\mathcal{L}(\boldsymbol{u}_h, \boldsymbol{p}_h, \lambda_h) = J(\boldsymbol{u}_h) - \boldsymbol{b}(\boldsymbol{u}_h, \boldsymbol{p}_h) + (\lambda_h, [\boldsymbol{u}_h])_S - \frac{1}{2}\int_S \kappa^* |\lambda_h|^2 \, \mathrm{d}s. \quad (32)$$

The reduced saddle-point problem is then as follows:

Find $(\boldsymbol{u}_h, \boldsymbol{p}_h, \lambda_h) \in \mathbf{V}_{gh} \times \mathbf{P}_h \times \Lambda_h$:

$$\mathcal{L}(\boldsymbol{u}_h, \boldsymbol{q}_h, \mu_h) \le \mathcal{L}(\boldsymbol{v}_h, \boldsymbol{q}_h, \mu_h) \le \mathcal{L}(\boldsymbol{v}_h, \boldsymbol{p}_h, \lambda_h), \quad \forall (\boldsymbol{v}_h, \boldsymbol{q}_h, \mu_h) \in \mathbf{V}_{gh} \times \mathbf{P}_h \times \Lambda_h, \quad (33)$$

with the corresponding saddle-point equations

$$\frac{1}{\delta t}(u_{ih}, v_h) - b_i(v_h, p_{ih}) = \frac{1}{\delta t}(u_{ih}^*, v_h) + (-1)^i(\lambda_h, v_h)_S, \quad \forall v_h \in V_{0h}^i, \quad (34)$$

$$-b_i(u_{ih}, q_h) = 0, \quad \forall q_h \in P_h^i, \quad (35)$$

$$(\kappa^* \lambda_h, \mu_h)_S = ([\boldsymbol{u}_h], \mu_h)_S, \ \forall \mu_h \in \Lambda_h. \quad (36)$$

Note that (34) corresponds to the minimization step in (33) whereas (35)–(36) correspond to maximization. The domain decomposition algorithms proposed in this paper are Uzawa/conjugate gradient type algorithms applied to the following constrained maximization problems:

DDM/S

$$\max \ell(\lambda_h) = \mathcal{L}(\boldsymbol{u}_h(\lambda_h), \boldsymbol{p}_h(\lambda_h), \lambda_h)$$

subject to

$$(u_{ih}, p_{ih}) \in V_{gh}^i \times P_h^i;$$

$$\frac{1}{\delta t}(u_{ih}, v_h) - b_i(v_h, p_{ih}) = \frac{1}{\delta t}(u_{ih}^*, v_h) + (-1)^i(\lambda_h, v_h)_S, \quad \forall v_h \in V_{0h}^i, \quad i = 1, 2,$$

$$-b_i(u_{ih}, q_h) = 0, \ \forall q_h \in P_h^i, \quad i = 1, 2.$$

DDM/P

$$\max \ell(\boldsymbol{p}_h, \lambda_h) = \mathcal{L}(\boldsymbol{u}_h(\boldsymbol{p}_h, \lambda_h), \boldsymbol{p}_h, \lambda_h)$$

subject to

$$u_{ih} \in V_{gh}^i;$$

$$\frac{1}{\delta t}(u_{ih}, v_h) - b_i(v_h, p_{ih}) = \frac{1}{\delta t}(u_{ih}^*, v_h) + (-1)^i(\lambda_h, v_h)_S, \quad \forall v_h \in V_{0h}^i, \quad i = 1, 2.$$

In DDM/S we have a decomposition in "degenerated" Stokes subproblems whereas in DDM/P we have a decomposition in Poisson subproblems. We refer, e.g., to (Daniel, 1970; Ekeland and Temam, 1999; Luenberger, 1989; Polak, 1971) for detailed studies of conjugate gradient methods and duality theory.

6.1. Decomposition in Degenerated Stokes Subproblems. We assume that $(u_{ih}, p_{ih}) = (u_{ih}(\lambda_h), p_{ih}(\lambda_h))$ $(1 \le i \le 2)$ is a solution to the following degenerated Stokes subproblem:

Find $(u_{ih}, p_{ih}) \in V_{gh}^i \times P_h^i$ such that

$$\frac{1}{\delta t}(u_{ih}, v_h) - b_i(v_h, p_{ih}) = \frac{1}{\delta t}(u_{ih}^*, v_h) + (-1)^i(\lambda_h, v_h)_S, \quad \forall v_h \in V_{0h}^i, \tag{37}$$

$$-b_i(u_{ih}, q_h) = 0, \quad \forall q_h \in P_h^i. \tag{38}$$

For convenience, we suppress the dependence of (u_{ih}, p_{ih}) on λ_h. The mapping $\lambda_h \mapsto (\boldsymbol{u}_h(\lambda_h), \boldsymbol{p}_h(\lambda_h))$ is linear and we have

$$\boldsymbol{u}_h(\lambda_h + t\mu_h) = \boldsymbol{u}_h(\lambda_h) + t\tilde{\boldsymbol{u}}_{\boldsymbol{h}},$$
$$\boldsymbol{p}_h(\lambda_h + t\mu_h) = \boldsymbol{p}_h(\lambda_h) + t\tilde{\boldsymbol{p}}_{\boldsymbol{h}},$$

where $\tilde{\boldsymbol{u}}_h = (\tilde{u}_{1h}, \tilde{u}_{2h})$ and $\tilde{\boldsymbol{p}}_h = (\tilde{p}_{1h}, \tilde{p}_{2h})$ are the solutions of the (uncoupled) sensitivity problems $(1 \le i \le 2)$:

Find $(\tilde{u}_{ih}, \tilde{p}_{ih}) \in V_{0h}^i \times P_h^i$, such that

$$\frac{1}{\delta t}(\tilde{u}_{ih}, v_h)_{\Omega_i} - b_i(v_h, \tilde{p}_{ih}) = (-1)^i(\mu_h, v_h)_S, \quad \forall v_h \in V_{0h}^i, \tag{39}$$

$$-b_i(\tilde{u}_{ih}, q_h) = 0, \quad \forall q_h \in P_h^i. \tag{40}$$

Then the directional derivative of $\ell(\lambda_h) = \mathcal{L}(\boldsymbol{u}_h(\lambda_h), \boldsymbol{p}_h(\lambda_h), \lambda_h)$ is given by

$$\begin{aligned}\frac{\partial \ell(\lambda_h)}{\partial \lambda_h} \cdot \mu_h = {} & \frac{1}{\delta t}(\boldsymbol{u}_h, \tilde{\boldsymbol{u}}_h) - \frac{1}{\delta t}(\boldsymbol{u}_h^*, \tilde{\boldsymbol{u}}_h) \\ & - \boldsymbol{b}(\boldsymbol{u}_h^*, \boldsymbol{p}_h) - \boldsymbol{b}(\boldsymbol{u}_h, \tilde{\boldsymbol{p}}_h) \\ & + (\mu_h, [\boldsymbol{u}_h])_S + (\lambda_h, [\tilde{\boldsymbol{u}}_{\boldsymbol{h}}])_S \\ & - (\kappa^* \lambda_h, \mu_h)_S, \end{aligned} \tag{41}$$

where $\tilde{\boldsymbol{u}}_h$ is the solution to the sensitivity problem (39)–(40). Setting $v_h = \tilde{u}_{ih}$ and $q_h = \tilde{p}_{ih}$ in (37)–(38) and substituting the result in (41), we get

$$\frac{\partial \ell(\lambda_h)}{\partial \lambda_h} \cdot \mu_h = ([\boldsymbol{u}_h] - \kappa^* \lambda_h, \mu_h)_S, \quad \forall \mu_h \in \Lambda_h.$$

Thus we deduce that

$$\nabla \ell(\lambda_h) = [\boldsymbol{u}_h] - \kappa^* \lambda_h.$$

With $\nabla \ell(\lambda_h)$ we can now construct a search direction. Since ℓ is quadratic, a more suitable search direction is a conjugate gradient direction.

When a search direction μ_h is constructed, the step size t is computed as the maximizer of the real-valued function $\rho(t) = \ell(\lambda_h + t\mu_h)$. Since ℓ is quadratic and concave, the maximizer of ρ is the unique solution of the linear equation

$$(\nabla \ell(\lambda_h + t\mu_h), \mu_h)_S = 0,$$

i.e.,

$$t = -\frac{(\nabla \ell(\lambda_h), \mu_h)_S}{([\tilde{\boldsymbol{u}}_h] - \kappa^* \mu_h, \mu_h)_S}.$$

The following algorithm is simply the conjugate gradient algorithm applied to the maximization of $\ell(\lambda_h)$:

Algorithm DDM/S

Iteration 0. Initialization: λ_h^0 given.

Compute $(\boldsymbol{u}_h^0, \boldsymbol{p}_h^0) \in \mathbf{V}_{gh} \times \mathbf{P}_h$ via $(i = 1, 2)$

$$\frac{1}{\delta t}(u_{ih}^0, v_h)_{\Omega_i} - b(v_h, p_{ih}^0) = \frac{1}{\delta t}(u_{ih}^*, v_h)_{\Omega_i} + (-1)^i(\lambda_h^0, v_h)_S, \quad \forall v_h \in V_{0h}^i,$$

$$-b_i(u_{ih}^0, q_h) = 0, \quad \forall q_h \in P_h^i,$$

$$\gamma_h^0 = [\boldsymbol{u}_h^0] - \kappa^* \lambda_h^0,$$
$$\mu_h^0 = \gamma_h^0.$$

Iteration $k \ge 0$. Assuming that λ_h^k, $(\boldsymbol{u}_h^k, \boldsymbol{p}_h^k)$, γ_h^k and μ_h^k are known, do the following:

Sensitivity:

Compute $(\tilde{\boldsymbol{u}}_h^k, \tilde{\boldsymbol{p}}_h^k) \in \mathbf{V}_{0h} \times \mathbf{P}_h$ via $(i = 1, 2)$

$$\frac{1}{\delta t}(\tilde{u}_{ih}^k, v_h) - b_i(v_h, \tilde{p}_{ih}^k) = (-1)^i(\mu_h^k, v_h)_S, \quad \forall v_h \in V_{0h}^i,$$

$$-b_i(\tilde{u}_{ih}^k, q_h) = 0, \quad \forall q_h \in P_h^i.$$

Stepsize:

$$t_k = -\frac{(\gamma_h^k, \mu_h^k)_S}{([\tilde{\boldsymbol{u}}_h^k] - \kappa^* \mu_h^k, \mu_h^k)_S}.$$

Update:

$$\lambda_h^{k+1} = \lambda_h^k + t_k \mu_h^k,$$
$$u_{ih}^{k+1} = u_{ih}^k + t_k \tilde{u}_{ih}^k, \quad i = 1, 2,$$
$$p_{ih}^{k+1} = p_{ih}^k + t_k \tilde{p}_{ih}^k, \quad i = 1, 2.$$

Gradient:

$$\gamma_h^{k+1} = [\boldsymbol{u}_h^{k+1}] - \kappa^* \lambda_h^{k+1}.$$

Conjugate gradient direction:

$$\beta_k = \frac{\| \gamma_h^{k+1} \|^2_{L^2(S)}}{\| \gamma_h^k \|^2_{L^2(S)}},$$
$$\mu_h^{k+1} = \gamma_h^{k+1} + \beta_k \mu_h^k.$$

We iterate until $\|\gamma_h^k\|_{L^2(S)}$ is sufficiently "small", i.e.,

$$\frac{\|\gamma_h^k\|_{L^2(S)}}{\|\gamma_h^0\|_{L^2(S)}} < \varepsilon.$$

At each step, we solve two uncoupled degenerated Stokes subproblems. The parallelization of the algorithm is therefore obvious.

6.2. Decomposition in the Poisson Subproblem. Now we assume that $u_{ih} = u_{ih}(p_{ih}, \lambda_h)$ $(1 \le i \le 2)$ is the solution to

$$\frac{1}{\delta t}(u_{ih}, v_h) = \frac{1}{\delta t}(u_{ih}^*, v_h) + b_i(v_h, p_{ih}) + (-1)^i(\lambda_h, v_h)_S, \quad \forall v_h \in V_{0h}^i. \quad (42)$$

For convenience, we suppress the dependence of u_{ih} on (p_{ih}, λ_h). As in Section 6.1, the mapping $(\boldsymbol{p}_h, \lambda_h) \mapsto \boldsymbol{u}_h(\boldsymbol{p}_h, \lambda_h)$ is linear and we have

$$\boldsymbol{u}_h(\boldsymbol{p}_h + t\boldsymbol{d}_h, \lambda_h + t\mu_h) = \boldsymbol{u}_h(\boldsymbol{p}_h, \lambda_h) + t\tilde{\boldsymbol{u}}_h,$$

where $\tilde{\boldsymbol{u}}_h = (\tilde{u}_{1h}, \tilde{u}_{2h})$ is the solution of the following (uncoupled) sensitivity problems $(1 \le i \le 2)$:

Find $\tilde{u}_{ih} \in V_{0h}^i$, such that

$$\frac{1}{\delta t}(\tilde{u}_{ih}, v_h) = b_i(v_h, d_{ih}) + (-1)^i(\mu_h, v_h)_S, \quad \forall v_h \in V_{0h}^i. \quad (43)$$

Then the directional derivative of $\ell(\boldsymbol{p}_h, \lambda_h) = \mathcal{L}(\boldsymbol{u}_h(\boldsymbol{p}_h, \lambda_h), \boldsymbol{p}_h, \lambda_h)$ is given by

$$\frac{\partial \ell(\boldsymbol{p}_h, \lambda_h)}{\partial(\boldsymbol{p}_h, \lambda_h)}(\boldsymbol{d}_h, \mu_h) = \frac{1}{\delta t}(\boldsymbol{u}_h, \tilde{\boldsymbol{u}}_h) - \frac{1}{\delta t}(\boldsymbol{u}_h^*, \tilde{\boldsymbol{u}}_h) - \boldsymbol{b}(\tilde{\boldsymbol{u}}_h, \boldsymbol{p}_h) - \boldsymbol{b}(\boldsymbol{u}_h, \boldsymbol{d}_h) + (\mu_h, [\boldsymbol{u}_h])_S + (\lambda_h, [\tilde{\boldsymbol{u}}_h])_S - (\kappa^* \lambda_h, \mu_h)_S, \quad (44)$$

where $\tilde{\boldsymbol{u}}_h$ is the solution of the sensitivity problem (43). Setting $v_h = \tilde{u}_{ih}$ in (42) and substituting the result in (44), we get

$$\frac{\partial \ell(\boldsymbol{p}_h, \lambda_h)}{\partial(\boldsymbol{p}_h, \lambda_h)}(\boldsymbol{d}_h, \mu_h) = -\boldsymbol{b}(\boldsymbol{u}_h, \boldsymbol{d}_h) + ([\boldsymbol{u}_h] - \kappa^* \lambda_h, \mu_h)_S.$$

Thus we deduce the gradient of ℓ with respect to λ_h:

$$\gamma_h := \nabla_\lambda \ell(\boldsymbol{p}_h, \lambda_h) = [\boldsymbol{u}_h] - \kappa^* \lambda_h.$$

In contrast, the gradient with respect to $\boldsymbol{p}_h$ is computed in $\mathbf{P}_h$ via the following uncoupled problem:

Find $g_i \in P_h^i$, $1 \le i \le 2$, such that

$$(g_{ih}, q_h)_{\Omega_i} = -(\nabla \cdot u_{ih}, q_h)_{\Omega_i}, \quad \forall q_h \in P_h^i. \quad (45)$$

A search direction for ℓ is computed from γ_h and g_{ih}. As in the previous section, a more suitable search direction is a conjugate gradient direction.

When a search direction $(\boldsymbol{d}_h, \mu_h)$ is constructed, the step size t is computed as the maximizer of the real-valued function $\rho(t) = \ell(\boldsymbol{p}_h + t\boldsymbol{d}_h, \lambda_h + t\mu_h)$. Since ℓ is quadratic, the maximizer of ρ is the unique solution of the linear equation

$$\rho'(t) = \partial \ell(\boldsymbol{p}_h + t\boldsymbol{d}_h, \lambda_h + t\mu_h) \cdot (\boldsymbol{d}_h, \mu_h) = 0,$$

i.e.,

$$t = -\frac{(\gamma_h, \mu_h)_S + (\boldsymbol{g}_h, \boldsymbol{d}_h)}{([\tilde{\boldsymbol{u}}_h] - \kappa^* \mu_h, \mu_h)_S - (\nabla \cdot \tilde{\boldsymbol{u}}_h, \boldsymbol{d}_h)}.$$

With the above results, we can present the conjugate gradient algorithm generating a maximizing sequence of ℓ.

Algorithm DDM/P1

Iteration 0. Initialization: $(\boldsymbol{p}_h^0, \lambda_h^0)$ given.

Compute $\boldsymbol{u}_h^0 \in \mathbf{V}_{gh}$ via

$$\frac{1}{\delta t}(u_{ih}^0, v_h) = \frac{1}{\delta t}(u_{ih}^*, v_h) + b_i(v_h, p_{ih}^0) + (-1)^i(\kappa^* \lambda_h^0, v_h)_S, \quad \forall v_h \in V_{0h}^i,\ i = 1, 2.$$

Compute $g_{ih}^0 \in P_h^i$ via

$$(g_{ih}^0, q_h)_{\Omega_i} = -(\nabla \cdot u_{ih}^0, q_h)_{\Omega_i}, \ \forall q_h \in P_h^i, i = 1, 2.$$

$$\gamma_h^0 = [\boldsymbol{u}_h^0] - \kappa^* \lambda_h^0,$$
$$d_{ih}^0 = g_{ih}^0, \quad i = 1, 2,$$
$$\mu_h^0 = \gamma_h^0.$$

Iteration $k \geq 0$. Assuming that $\boldsymbol{p}_h^k$, λ_h^k, $\boldsymbol{u}_h^k$, d_h^k and μ_h^k are known, do the following:

Sensitivity:

Compute $\tilde{\boldsymbol{u}}_h^k = (\tilde{u}_{1h}^k, \tilde{u}_{2h}^k) \in \mathbf{V}_{0h}$ via

$$\frac{1}{\delta t}(\tilde{u}_{ih}^k, v_h) = b_i(v_h, d_{ih}^k) + (-1)^i(\mu_h^k, v_h)_S, \quad \forall v_h \in V_{0h}^i,\ i = 1, 2.$$

Stepsize:

$$t_k = -\frac{(\gamma_h^k, \mu_h^k)_S + (\boldsymbol{g}_h^k, \boldsymbol{d}_h^k)}{([\tilde{\boldsymbol{u}}_h^k] - \kappa^* \mu_h^k, \mu_h^k)_S - (\nabla \cdot \tilde{\boldsymbol{u}}_h^k, \boldsymbol{d}_h^k)}.$$

Update:

$$\lambda_h^{k+1} = \lambda_h^k + t_k \mu_h^k,$$
$$p_{ih}^{k+1} = p_{ih}^k + t_k d_{ih}^k, \quad i = 1, 2,$$
$$u_{ih}^{k+1} = u_{ih}^k + t_k \tilde{u}_{ih}^k, \quad i = 1, 2.$$

Gradient:

Compute $g_{ih}^{k+1} \in P_h^i$ by solving

$$(g_{ih}^{k+1}, q_h)_{\Omega_i} = -(\nabla \cdot u_{ih}^{k+1}, q_h)_{\Omega_i}, \quad \forall q_h \in P_h^i,\ i = 1, 2,$$

$$\gamma_h^{k+1} = [\boldsymbol{u}_h^{k+1}] - \kappa^* \lambda_h^{k+1}.$$

Conjugate gradient direction:

$$\beta_k = \frac{\left\|\boldsymbol{g}_h^{k+1}\right\|^2_{\mathbf{L^2(\Omega)}} + \left\|\gamma_h^{k+1}\right\|^2_{L^2(S)}}{\left\|\boldsymbol{g}_h^{k}\right\|^2_{\mathbf{L^2(\Omega)}} + \left\|\gamma_h^{k}\right\|^2_{L^2(S)}},$$

$$\boldsymbol{d}_h^{k+1} = \boldsymbol{g}_h^{k+1} + \beta_k \boldsymbol{d}_h^k,$$

$$\mu_h^{k+1} = \gamma_h^{k+1} + \beta_k \mu_h^k.$$

We iterate until $\left\|\gamma_h^k\right\|^2_{L^2(S)} + \left\|\boldsymbol{g}_h^k\right\|^2_{\mathbf{L^2(\Omega)}}$ is sufficiently "small", i.e.,

$$\frac{\left\|\gamma_h^k\right\|^2_{L^2(S)} + \left\|\boldsymbol{g}_h^k\right\|^2_{\mathbf{L^2(\Omega)}}}{\left\|\gamma_h^0\right\|^2_{L^2(S)} + \left\|\boldsymbol{g}_h^0\right\|^2_{\mathbf{L^2(\Omega)}}} < \varepsilon^2. \tag{46}$$

At each step, we solve four uncoupled Poisson subproblems. The parallelization of the algorithm is therefore obvious.

The algorithm DDM/P1 can be improved by preconditioning the computing of the gradient with respect to p_{ih} as in the preconditioned conjugate gradient algorithm for L^2-projection (see, e.g., Glowinski, 2003, Ch. 7; Glowinski *et al.*, 2000). To this end, we define the subspace

$$P_{Sh}^i = \left\{ p_h \in P_h^i,\ p_h = 0 \ \text{ on } \ S \right\}.$$

Then, instead of (45), the gradient with respect to p_{ih} is computed via the following (uncoupled) Poisson subproblems:

Find $g_i \in P_{Sh}^i$, $1 \leq i \leq 2$, such that

$$(\nabla g_{ih}, \nabla q_h)_{\Omega_i} = -(\nabla \cdot u_{ih}, q_h)_{\Omega_i}, \quad \forall q_h \in P_{Sh}^i. \tag{47}$$

We refer to this preconditioned version as the algorithm DDM/P2.

7. Numerical Experiments

In this section we present some numerical experiments using algorithms outlined in the previous sections. The domain decomposition algorithms presented in the previous sections were implemented in Fortran 90, on an SGI Origin 200 computer, using an MPI library. MPI subroutines are used only for solving in parallel uncoupled Stokes or Poisson problems (Substep 1) and uncoupled linear advection-diffusion problems (Substep 2).

All linear systems involved are solved by a preconditioned conjugate gradient algorithm. The preconditioning is obtained by an incomplete Choleski factorization with drop tolerance varying from 10^{-5} to 10^{-3}.

In all numerical simulations, we consider an ocean/atmosphere type model consisting of two Navier–Stokes fluids of viscosities $\nu_1 = 1/2000$ and $\nu_2 = 1/200$. The traction coefficient is $\kappa = 2.45 \cdot 10^{-3}$.

7.1. Example 1. We consider a simple ocean/atmosphere model, with rectangular subdomains as shown in Fig. 3. The subdomains are $\Omega_1 = (-10,\ 10) \times (0,\ 1)$ and $\Omega_2 = (-10,\ 10) \times (-1,\ 0)$ with the interface $S = (-10,\ 10) \times \{0\}$.

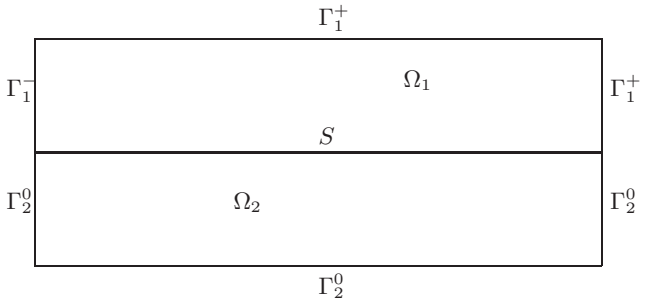

Fig. 3. First simple ocean/atmosphere model.

The boundary conditions are

$$u_{1h} = \begin{pmatrix} (1+z)/2 \\ 0 \end{pmatrix} \quad \text{on} \quad \Gamma_1^-, \tag{48}$$

$$\nu_1 \frac{\partial u_{1h}}{\partial n_1} - p_{1h} n_1 = 0 \quad \text{on} \quad \Gamma_1^+,$$

$$u_{2h} = 0 \quad \text{on} \quad \Gamma_2^0.$$

The time step is $\delta t = 0.01$. For the initial velocity, we assume that in Ω_2 the fluid is in rest whereas in Ω_1 the fluid profile is (48) , i.e.,

$$u_{1h}^0 = \begin{pmatrix} (1+z)/2 \\ 0 \end{pmatrix} \quad \text{in} \quad \Omega_1,$$

$$u_{2h}^0 = 0 \qquad\qquad \text{in} \quad \Omega_2.$$

For the solution of the degenerated Stokes problem in the algorithm DDM/S, we use the preconditioned Uzawa/conjugate gradient method (see, e.g., Glowinski, 2003, §34; Glowinski *et al.*, 2000).

To study the numerical behavior of our algorithms, subdomains are first discretized by a nonuniform mesh shown in Fig. 4 and consisting of 2×997 nodes and 2×1760 triangles. This initial mesh is successively refined to produce meshes with 2×3753, 2×14545 and 2×57249 nodes. The performances of the individual algorithms, in terms of CPU time, are presented in Table 1. The comparison was made at $t = 1$ (i.e., $n = 100$ for $\delta t = 0.01$). We first notice that the preconditioning significantly improves the performance of the algorithm DDM/P. The algorithm DDM/S requires more computational time than the algorithm DDM/P2. But while assessing this performance we must bear in mind that the preconditioned Uzawa/conjugate gradient method can be easily parallelized. Indeed, each iteration of the Uzawa/conjugate gradient method requires solving two uncoupled scalar Poisson problems for velocity and one scalar Poisson problem for pressure.

Table 1. Performances of the algorithms DDM/S, DDM/P1 and DDM/P2 at $t = 1$.

Mesh	CPU time in sec.		
	DDM/S	DDM/P1	DDM/P2
2×997	20.846	104.921	8.564
2×2865	108.758	877.991	45.918
2×11105	854.451	9647.342	417.833
2×87426	8578.686	> 10000	5169.937

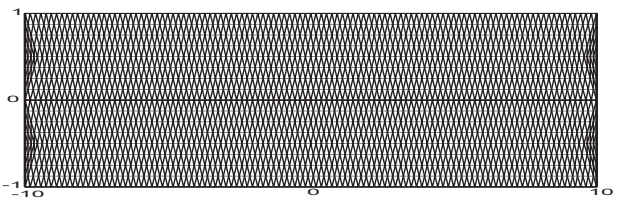

Fig. 4. Mesh sample of a simple ocean/atmosphere model with rectangular domains.

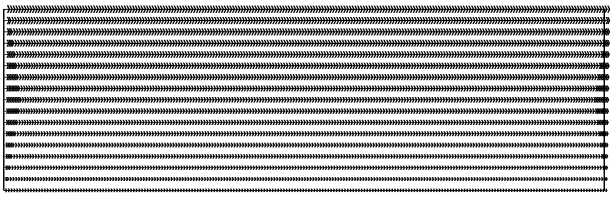

Fig. 5. Velocity field in Ω_1.

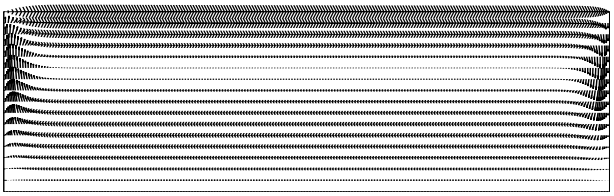

Fig. 6. Velocity field in Ω_2.

Using the algorithm DDM/P2 and meshes with 2×3753 nodes, a stationary solution is reached at $t = 61.48$ (i.e., 6148 times steps) after 2760.242 seconds of CPU time. Velocity fields are presented in Figs. 5 and 6. We observe that the transmission condition generates a recirculation within Ω_2 while the flow in Ω_1 is practically unchanged.

7.2. Example 2. We consider a coupled system with $\Omega_1 = (-10,\ 10) \times (0, 1)$ and

$$\Omega_2 = \left\{ (x, y)\ :\ \frac{x^4}{10^4} - 1 \le y \le 0,\ x \in (-10,\ 10) \right\},$$

as illustrated in Fig. 7. Subdomains are discretized by nonuniform meshes consisting of 3771 nodes for Ω_1, 3079 nodes for Ω_2 and 81 nodes on the interface S. Boundary and initial conditions are the same as in Example 1 with $\delta t = 0.01$ as the time step.

A stationary solution is reached at $t = 59.76$ (i.e., 5976 time steps). In the simulation, the number of iterations for the algorithm DDM/P2 is about 5 and the number of iterations for both linear advection-diffusion problems is 1. Figs. 8–10 show streamlines and velocity fields for the stationary solution. As in Example 1, a recirculation is established within Ω_2 while the flow in Ω_1 remains practically unchanged.

7.3. Example 3. In this example we assume that there is a submarine mountain (i.e., the subdomain Ω_2 is nonconvex) whereas Ω_1 is the same as in Examples 1 and 2, see Fig. 11. Ω_2 is given by

$$\Omega_2 = \Big\{ (x, y)\ :\ 0 \ge y \ge \alpha - 0.175x \sin(0.35x), \\ x \in (-10,\ 10) \Big\},$$

where $\alpha = 1.75 \sin(3.5)$.

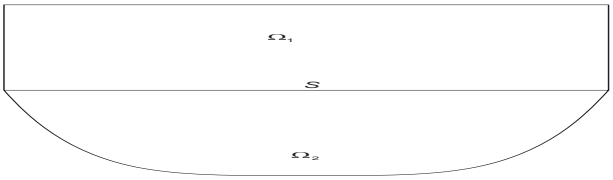

Fig. 7. Second simple ocean/atmosphere model.

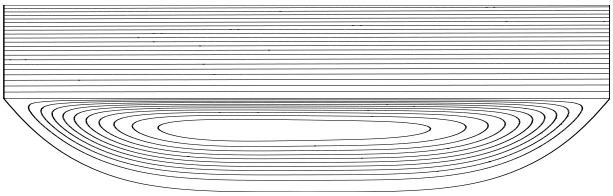

Fig. 8. Streamline plot.

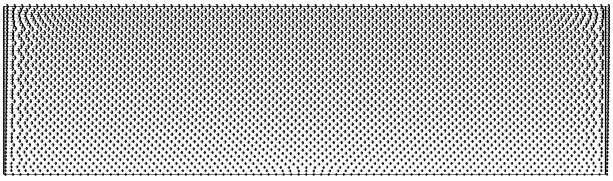

Fig. 9. Velocity field in Ω_1.

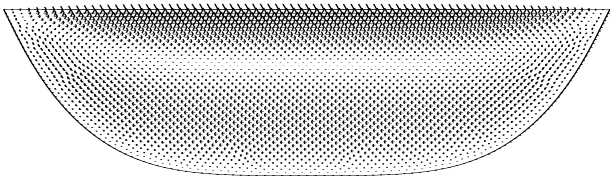

Fig. 10. Velocity field in Ω_2.

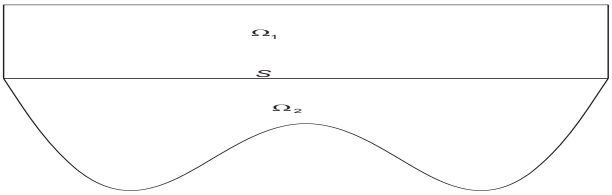

Fig. 11. Third simple ocean/atmosphere model.

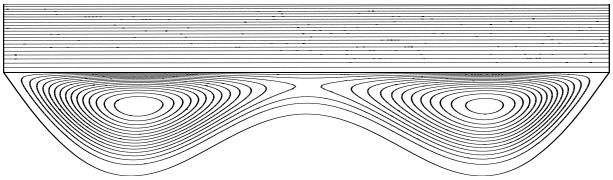

Fig. 12. Streamline plot.

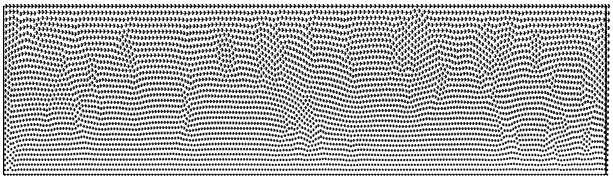

Fig. 13. Velocity field in Ω_1.

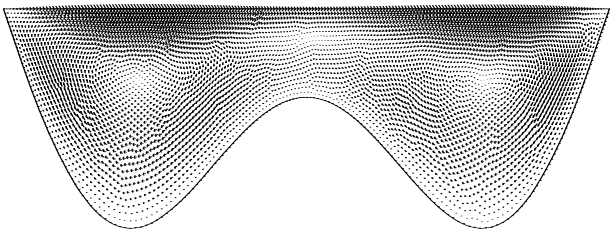

Fig. 14. Velocity field in Ω_2.

The coupled system is discretized by nonuniform meshes consisting of 5836 nodes for Ω_1, 5321 nodes for Ω_2 and 201 nodes on the interface S. The step size is $\delta t = 0.005$.

A stationary solution is reached at $t = 96.59$ (i.e., 19300 time steps). In the simulation, the number of iterations for the algorithm DDM/P2 is about 5 and the number of iterations for both linear advection-diffusion problems is 1. Figures 12–14 show streamlines and velocity fields for the stationary solution. As in Example 1, a recirculation is established within Ω_2 while the flow in Ω_1 remains practically unchanged. We can notice that the presence of the submarine mountain dramatically affects the flow in Ω_2. Indeed, the flow slows down before arriving to the straitness and accelerates again after crossing it. Bernardi *et al.* (2004) obtained the same results with two coupled turbulent Stokes fluids.

8. Conclusion

We have demonstrated that operator-splitting and domain decomposition methods reproduce the qualitative behavior of two coupled Navier-Stokes flows. Further study is underway to improve the simulation by extending it to more realistic problems.

References

Bernardi C., Chacon T., Lewandowski R. and Murat F. (2002) *A model for two coupled turbulent fluids I: Analysis of the system*. — Nonlinear Partial Differential Equations and Their Applications, Collège de France Seminar, Vol. XIV (Paris, 1997/1998), pp. 69–102, Stud. Math. Appl., Vol. 31, Amsterdam: North–Holland.

Bernardi C., Chacon T., Lewandowski R. and Murat F. (2003): *A model for two coupled turbulent fluids II: Numerical analysis of a spectral discretization*. — SIAM J. Numer. Anal. Vol. 40, No. 6, pp. 2368–2394.

Bernardi C., Chacon-Rebello T., Gomez-Marmol, Lewandowski R. and Murat F. (2004): *A model for two coupled turbulent fluids III: Numerical approximation by finite elements*. — Numer. Math., Vol. 98, No. 1, pp. 33–66.

Bresch D. and Koko J. (2004): *An optimization-based domain decomposition method for nonlinear wall laws in coupled systems*. — Math. Models Meth. Appl. Sci., Vol. 14, No. 7, pp. 1085–1101.

Ciarlet P. (1979): *The Finite Element Method for Elliptic Problems*. — Amsterdam: North-Holland.

Daniel J. (1970): *The approximate minimization of functionals*. — Englewood Cliffs, NJ: Prentice–Hall.

Du Q. (2001): *Optimization based non-overlapping domain decomposition algorithms and their convergence*. — SIAM J. Numer. Anal., Vol. 39, No. 3, pp. 1056–1077.

Du Q. and Gunzburger M.D. (2000): *A gradient method approach to optimization-based multidisciplinary simulations and nonverlapping domain decomposition algorithms*. — SIAM J. Numer. Anal., Vol. 37, No. 5, pp. 1513–1541.

Ekeland I. and Temam R. (1999): *Convex Analysis and Variational Problems*. — Philadelphia: SIAM.

Glowinski R. (2003): *Numerical Methods for Fluids*, In: Handbook of Numerical Analysis, Vol. IX, (Ciarlet P.G. and Lions J.L., Eds.), Amsterdam: North-Holland.

Glowinski R. and Le Tallec P. (1989): *Augmented Lagrangian and Operator-splitting Methods in Nonlinear Mechanics*. — Philadelphia: SIAM.

Glowinski R. and Marocco A. (1975): *Sur l'approximation par éléments finis d'ordre un, et la résolution par pénalisation–dualité, d'une classe de problèmes de Dirichlet non linéaires*, RAIRO Anal. Num., Vol. 2, No. 2, pp. 41–76.

Glowinski R., Pan T.-W. and Périaux J. (1998): *Distributed Lagrange multiplier methods for incompressible viscous flow around moving rigid bodies*. — Comput. Meth. Appl. Mech. Eng., Vol. 151, No. 1–2, pp. 181–194.

Glowinski R., Pan T.W., Hesla T.I., Joseph D.D. and Périaux J. (2000): *A distributed Lagrange multiplier/fictitious domain method for the simulation of flow around moving rigid bodies: application to particulate flow*. — Comput. Methods Appl. Mech. Eng., Vol. 184, No. 2–4, pp. 241–267.

Gunzburger M.D. and Peterson J. (1999): *An optimization based domain decomposition method for partial differential equations*. — Comp. Math. Appl., Vol. 37, No. 10, pp. 77–93.

Gunzburger M.D. and Lee H.K. (2000): *An optimization–based domain decomposition method for Navier–Stokes equations*. — SIAM J. Numer. Anal., Vol. 37, No. 5, pp. 1455–1480.

Koko J. (2002): *An optimization based domain decomposition method for a bonded structure*. — Math. Models Meth. Appl. Sci., Vol. 12, No. 6, pp. 857–870.

Koko J. (2006): *Uzawa conjugate gradient domain decomposition methods for coupled Stokes flows*. — J. Sci. Comput., Vol. 26, No. 2, pp.195–215.

Lewandowski R. (1997): *Analyse Mathématique et Océanographie*. — Paris: Masson.

Lions J.L., Temam R. and Wang S. (1993): *Models for the coupled atmosphere and ocean. (CAO I,II)*. — Comput. Mech. Adv., Vol. 1, No. 1, pp. 120.

Luenberger D. (1989): *Linear and Nonlinear Programming*. — Reading, MA: Addison Wesley.

Marchuk G.I. (1990): *Splitting and alternating direction methods*, In: Handbook of Numerical Analysis Vol. I, (Ciarlet P.G. and Lions J.L., Eds.). — Amsterdam: North-Holland, pp. 197–462.

Miglio E., Quarteroni A. and Saleri F. (2003): *Coupling of free-surface and groundwater flows*. — Comput. Fluids, Vol. 32, No. 1, pp. 73–83.

Pan T.W., Joseph D.D. and Glowinski R. (2005): *Simulating the dynamics of fluid-ellipsoid interactions*. — Comput. Struct., Vol. 83, No. 6–7, pp. 463–478.

Polak E. (1971): *Computational Methods in Optimization*. — New York: Academic Press.

Quarteroni A. and Valli A. (1999): *Domain decomposition methods for partial differential equations*. — Oxford: Oxford University Press.

Suquet P.M. (1988): *Discontinuities and plasticity*, In: Nonsmooth Mechanics and Applications, (Moreau J.J. and Panagiotopoulos P.D., Eds.). — CISM courses and lectures, New-York: Springer, No. 302, pp. 279–340.

Received: 17 March 2006
Revised: 15 August 2006