43rd AIAA Aerospace Sciences Meeting and Exhibit, 10-13 Jan. 2005, Reno Hilton

Heterogeneous Domain Decomposition for Multi-Scale Problems *

M. Garbey †, C. Picard ‡

Department of Computer Science,University of Houston, Houston, Tx 77204,USA

R. Tran Son Tay §

Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, Fl 32611,USA

The objective of this work is to develop numerical tools useful for the understanding of fundamental phenomena in Blood Flow, including some biochemistry and cellular dynamics. An open problem that may benefit from multi-scale numerical tools is for example the understanding of the relation ship between inflammation and cholesterol plaque. The corresponding topics we address in this paper are (1) Heterogeneous Domain Decomposition for Multi-Scale Problem (2) Fluid-Structure Interaction in the Boundary Layer and (3) Immersed Boundary Method for the dynamic of cells (ref. C.Peskin et Al). In this present paper we concentrate on (1) and give preliminary results on (2) and (3) in the framework of heterogeneous domain decomposition.

I. Introduction and Motivation

The understanding of blood flow phenomena relevant to vascular diseases is a critical and yet difficult issue. The initial formation of arteriosclerosis plaques, for example, depends on a combination of various factors. The impact of physical and geometrical properties is well documented experimentally. There is a number of evidence relating shear stress to the pathophysiology of arteriosclerosis: arteriosclerosis plaques are frequently located at or near regions of bifurcations, multiple intersections, and high vessel wall curvatures.[1,2] However, arteriosclerosis is a slow process influenced accumulatively by many parameters, including chemical agents, mutagen, hypoxemia, immunological factors, and microbiological factors such as bacteria, virus, or endotoxin, which act on a much shorter time scale and in a localized manner. Therefore, in order to perform realistic simulations for cardiovascular physiology, one must address issues related to fluid and vessel wall interactions, and coupling of the aforementioned disparate time scales and length scales. Furthermore, one should develop a systematic way of calibrating the simulation to the medical imaging data. The mathematical problems to be solved are:

- The Navier-Stokes equation to describe incompressible flow and eventually quasi non-Newtonian flow.
- The coupling between the fluid flow equation and the soft tissue model for the artery wall.
- The boundary layer in fluid flow, and multiple scales in reaction diffusion convection of chemical species between the fluid flow and the multi layer material wall.
- The parameters identification and calibration of the mathematical model to medical imaging data or experimental data.

Such an endeavor involves substantial challenges arising from mathematical modeling, numerical techniques, computational modeling, image analysis, and experimental validation.

*This work was supported in part by NSF Grant ACI-0305405.

†Professor, Department of Computer Science, garbey@tlc2.uh.edu

‡PhD Candidate, Department of Computer Science, picard@tlc2.uh.edu

§Professor, Department of Mechanical and Aerospace Engineering, rtst@ufl.edu

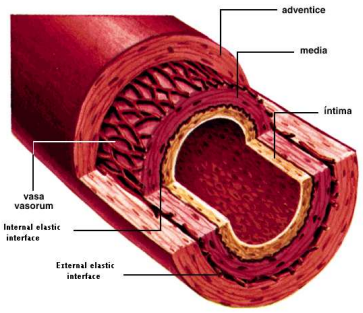

Figure 1. Structure of an artery

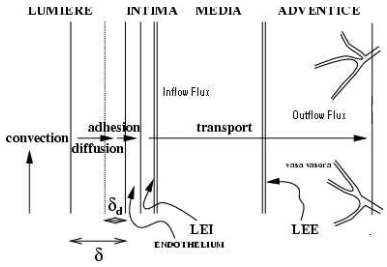

Figure 2. Structure of an Artery and Diffusion Convection Processes.

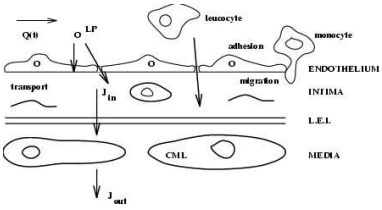

Figure 3. Detail modeling of the Intima for Wall Absorption.

The aim of this paper is to present a set of domain decomposition tools that will allow to develop numerical efficient computation of multi-scale complex blood flow phenomena. We restrict ourself to two space dimension while most of the concept can be extended to three space dimensions. In Section II, we review the model and show that its numerical solution is amenable to the design of fast solver for two classical operators. In Section III, we give the main concepts allowing us to build the heterogeneous domain decomposition. In Section IV, we present an heterogeneous domain decomposition that can match the boundary layers. In Section V, we apply our technique to Incompressible Navier-Stokes flow. Section VI gives some conclusions and perspectives.

II. Model

First, we will review briefly the different PDEs used to described some blood flow in main large arteries with bio-chemistry. In particular we would like to refer to the pioneer work of C. Peskin et Al,[3] A. Quarteroni et Al,[4,5,6] and P. Fischer et Al.[7]

The artery can be naturally decomposed into four subdomains that are the lumen, the intima, the media and the adventice more or less arranged into concentric layers. We recall that each layer has different biological functions and physical properties. For arteries of diameter more than few millimeters we can use the incompressible Navier-Stokes equations (NS),

$$\begin{aligned} \partial_t u + u.\nabla u + \nabla p - \nu \Delta u &= F, \\ \nabla . u &= 0 \end{aligned} \tag{1}$$

that is only valid a priori in the lumen. In these equation u is the velocity and p the pressure. We have the no-slip boundary condition on the wall and prescribed boundary condition on inflow and outflow that must be derived somehow.

Let us extend continuously the blood flow speed u to zero in the artery wall.

Navier-Stokes can be coupled to the reaction-convection-diffusion of chemical components transported by the blood flow in the lumen and diffused in the different layers of the artery wall,

$$\partial_t C + \nabla.(uC) = \nabla.(D\nabla C) + Q(C, E, t),$$

where C is a vector of these chemical components, D stands for the solute diffusivity. D is usually a diagonal matrix (d_i) with coefficients that are smooth piecewise functions in space. As a matter of fact, $d_i(x,t)$ have discontinuities across the interface between the different subdomains because values may differ drastically depending on the space location x in the lumen, the intima, the media, the adventice areas or outside the vessel. The source term Q might be either a source or a sink attached to the wall location.

The Reynolds number in main arteries is in the range 100-3000, and the blood flow can exhibit a complex unsteady structure with attachment and detachment of boundary layers. For chemical components, We may expect boundary layers much thinner than the boundary layer for the fluid flow - see Figure 2 and Figure 3.

In Navier-Stokes equations, the body force can take into account many different phenomena. If one follow the immersed boundary method approach, *F may represent the action of the wall into the flow:* when the wall is assimilated to an elastic membrane the force takes the following form:

$$F(x,t) = \int_0^L f(s,t)\delta(x - X(s,t))ds,$$

where L is the length of the membrane, s is a natural parametrization of the interface, $X(s,t)$ is the time dependent location of the membrane, δ is the Dirac function, and $f(s,t)$ describes the elastic properties of the membrane.

The relation $\frac{\partial X}{\partial t}(s,t) = u(X(s,t),t)$ stands for the no-slip boundary condition. We can assimilate the vessel wall as a *network of elastic and muscle fibers.* The elastic-contractile properties of the smooth muscle fibers might be modulated by concentration of vasoactive agents. Inside the lumen, F may represent a cohesion force density produced by interpolated bonds between activated platelets or by some adhesion-detachment of cells on the wall.

Using a semi-implicit projection scheme for the Navier-Stokes, and a semi-implicit or Newton scheme for the chemical components, it is easy to see that the computation of the multi-scale problem relies on the design of fast linear solvers for the following two operators:

$$\partial_t V + (\vec{a}.\nabla)V - \nabla.(D\nabla V) + \gamma V, \tag{2}$$

$$\Sigma_i \alpha \frac{\partial^2 V}{\partial x_i^2}, \tag{3}$$

where V can be a vector and $x = (x_i)$ is the space variable.

Further in each subdomain each operator (2) or (3) may have coefficients with different scalings. The artery walls are essentially thin cylindric domains with strong anisotropic properties between space direction parallel to the artery axis and to the normal direction to the wall. Further in arteries of radius larger than few millimeters, the Reynolds number is relatively large, up to few thousands, and one has a boundary layer. A biologically motivated Domain Decomposition (**DD**) consists then to decompose the lumen into a Boundary Layer subdomain (**BL**), where the gradient of flow speed is large, the chemical exchange with the wall is critical, the fluid structure interaction is strong, and into a Regular Domain (**RD**) that capture the main part of the flow outside the BL.

Our goal is then to build an heterogeneous DD for the two types of solvers corresponding to (2) and (3) in two types of subdomain such as BL or RD. One can further notice that walls subdomains are by nature BL like subdomains.

In our work we focus on the multi-scale properties of the model and make extensive used of the asymptotic analysis in the numeric. Such method are so-called *Asymptotic Induced Domain Decomposition Method.* In the next section, we show the link between domain decomposition and singular perturbations.

III. Domain Decomposition and Singular Perturbation

To illustrate the concept, let us consider the momentum equation in Navier-Stokes equation. The time integration of

$$\partial_t U + (\vec{a}.\nabla)U - \nabla.(D\nabla U),$$

leads to the singular perturbation problem,

$$-\epsilon\nabla.(D\nabla U) + \nu(\vec{a}.\nabla)U + U = F,$$

with boundary conditions such as the no-slip condition along the walls, and prescribed velocity at inlet and outlet. The singular perturbation parameter is $\epsilon = \frac{dt}{Re} \ll 1,\ \nu = dt \ll 1$. Similarly, for steady problem, the corresponding problem is

$$-\epsilon\nabla.(D\nabla U) + \nu(\vec{a}.\nabla)U = F,$$

with analogous boundary conditions but $\epsilon = \frac{1}{Re} \ll 1$.

Small ϵ , i.e. large Reynolds number are responsible for Boundary Layers.

Singular perturbation analysis of such operators has been extensively reported in the literature. One popular technique is the Matched Asymptotic Expansion method.[8,9,10] This method can be seen as an analytical equivalent of the domain decomposition method for the numerical solution of PDEs - http://www.ddm.org. From the matched asymptotic method, we have:

- The domain Ω should be decomposed into

  $$\Omega = \Omega^R \cup \Omega^{BL}.$$

  where Ω^{BL} stands for a thin domain where fast scales gives large variation of the solution, and Ω^R stands for the subdomain where the solution has derivatives of order one.

- The Boundary Layer approximation uses a local orthogonal coordinate system η, ξ in Ω^{BL} attached to the wall $\partial\Omega$, with normal-tangential coordinates.

- Most often, the normal variable should be stretched as

  $$\xi = \frac{\eta}{\epsilon^p},$$

  where ϵ^p is the so-called **thickness of the layer.**

- The choice of the space variable coordinate system in each subdomain is such that the solution has a regular asymptotic expansion in each subdomain:

  $$U^R = \Sigma_{j=1..n}\delta_i^R(\epsilon)U_i^R,\ U^{BL} = \Sigma_{j=1..n}\delta_i^{BL}(\epsilon)U_i^{BL},$$

- Regular expansion means separation of variable between space and ϵ parameter: in other words U_i is independent of ϵ

- Both asymptotic expansion of U in Ω^R and Ω^{BL} should be asymptotically valid in an overlap area $\Omega^R \cap \Omega^{BL} \neq 0$

- Matching conditions define a composite uniform expansion:

  $$U = \mathcal{H}U^R + (1 - \mathcal{H})U^{BL},$$

  where $\mathcal{H}$ is a partition of unity, consistent with the PDE problem.

- A stability estimate is necessary to prove the convergence of this formal composite asymptotic expansion.

One can motivate a DD numerical procedure using these analytical results. Our objective is to get a numerically efficient numerical method from the matched asymptotic method idea. This is particularly easy to do with the Schwarz algorithm as a DD solver.

Following the asymptotic analysis framework, we discretize Ω^R and Ω^{BL} with grids topologically equivalent to Cartesian meshes using the local coordinate system: the overall grid is composite. Because we have regular data structure, we can use cache efficient numerical algorithm that leads to dramatic performance improvement on scalar processors. Ω^R is the restriction of a Cartesian grid to a polygonal domain. The background Cartesian grid on the box can be used as a preconditioner with fictitious domain decomposition.[11] Difference in space scaling is taken care by the DD which avoid possible local FV cells with

flat angles. The Schwarz algorithm is an iterative scheme providing automatically the matching condition between subdomains. Artificial interfaces in the Schwarz method with (artificial boundary layers) give superlinear speed of convergence. The first order asymptotic approximation of the operator in the subdomain is a good preconditioner.

However, conservativity of the physical quantities with overset grids might be an issue. We refer to the work of G. Chesshire and W. Henshaw for a general discussion of this problem that is common to the Chimera approach. In this paper we use interface conditions derived from interpolators that satisfy a maximum principle. The maximum principle allows us to keep the convergence properties of the Schwarz method for the non matching grid case analogue to the matching grid case. We have relied on extensive solution verification here to check that these non conservative interface boundary conditions leads, in our benchmark problems, to second order accuracy provided that the grids is fine enough. Further the DD that is driven by the singular perturbation problem, does not provide a priori a fast convergence of the elliptic solver. As a matter of fact the Multiplicative Schwarz algorithm has slow convergence for the elliptic operator (3) as opposed to singular perturbation problems corresponding to (2). In the next section, we discuss the optimized choice of subdomain algebraic solvers and an acceleration method for the multiplicative Schwarz algorithm.

IV. Heterogeneous Domain Decomposition for Navier-Stokes

We are going to present successively results on the performance of subdomain algebraic solver, interface solver for the domain decomposition.

A. Fast subdomain solver

Let us consider an incompressible Navier-Stokes flow in a curved pipe. We present performances for the Poisson solver used in a pressure solve, that is the most time consuming part of the code. As mentioned earlier, we generate a composite mesh with the classical normal-tangential coordinate system in Boundary Layer theory and a regular mesh in the regular domain (Figure 11).

• On figures 4 and 5 are represented the performance of different solvers in Cartesian and curvilinear subdomain. In each figure, the left bar is for LU decomposition, middle bar for BICGSTAB with incomplete LU preconditioning and right bar is for Algebraic multigrid. These results show that the optimum choice of the algebraic solver depends critically on the nature of the subdomain.

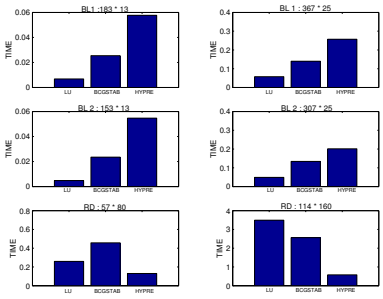

Figure 4. Comparison of the elapsed time for each subdomain with preconditioning

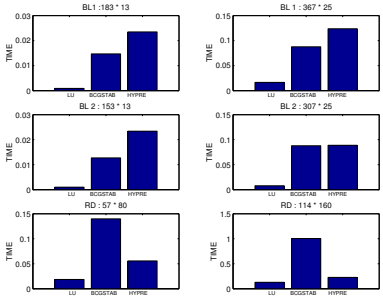

Figure 5. Comparison of the elapsed time for each subdomain with a precomputed preconditioner

We notice that the (naive) use of algebraic multigrid is much slower solver than LU decomposition in the boundary layer domain. We can conclude from extensive experiments and performance modeling[12] that

- For the Poisson solver multigrid gives the best result for the polygonal domain Ω^R, while direct solvers can be used advantageously in the boundary layer. As a matter of fact, since the Poisson operator does not change in time stepping, the LU decomposition can be computed once and for all. Further the boundary layer subdomain have many more grid points in the tangential direction than in the normal direction. We can then take advantage of the fact that the matrix of the linear system has a small bandwidth relative to the global size of the problem.

- For the momentum's equation solver, we can use as a preconditioner in the boundary layer the one dimensional approximation of the operator that neglect all tangential derivatives. It is indeed the first order term in the asymptotic expansion of the operator with respect to ϵ. Multigrids give also excellent result in the regular domain.

The Schwarz iterative solver for the pressure equation needs still to be speed up. We report on the next section on the performance of the Aitken acceleration method.

B. Iterative scheme for the domain decomposition

In this section we illustrate the performance of our iterative scheme for the DD with the test case of a flow past a cylinder.

- The implementation is keeping modular and easy to debug by using a multiplicative Schwarz algorithm.
- The convergence of the Schwarz algorithm is fast for the momentum or vorticity equation since operator is a perturbation of identity.[13,14]
- The convergence is slow for the pressure or stream function equation:

  We accelerate the convergence by an Aitken like method[15,16]

  We get a better initial guess for the interface solution by using extrapolation in time.

We need then an average of two to three iterates of the multiplicative Schwarz algorithm per time step for the flow past a cylinder in a 2D channel with a Reynolds ~ 300 using the $\omega - \Psi$ formulation.

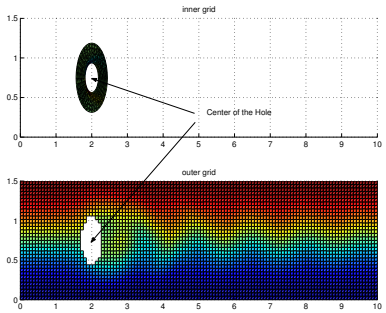

Figure 6. Composite mesh for a two D flow past a cylinder in a long channel.

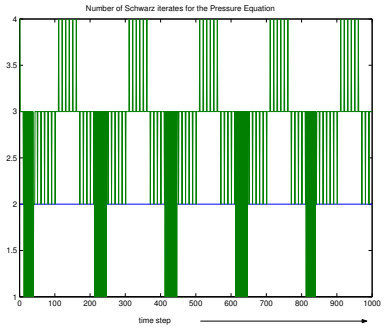

Figure 7. Number of Schwarz iterates for the stream solution procedure as function of time step's number. The times step is 0.005, and tolerance for the stop criteria is such that the difference between last two iterates is less than 10^{-6} in maximum norm.

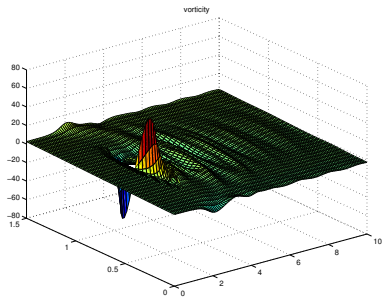

Figure 8. Vorticity function at Reynolds number 300.

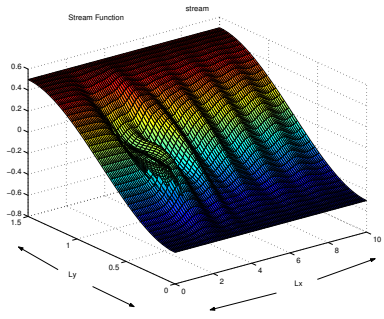

Figure 9. Stream function at Reynolds number 300.

The next section gives some accuracy assessment of our solution.

V. Solution verification

A. Description of the test case and conservativity issue

Let us consider the Navier-Stokes model in $\omega - \Psi$ formulation.

$$\frac{\partial \omega}{\partial t} + (\mathbf{u} \cdot \nabla)\omega - \nu \nabla^2 \omega = 0 \tag{4a}$$

$$\nabla^2 \Psi + \omega = 0 \tag{4b}$$

$$\frac{\partial \Psi}{\partial \eta} = -u_\eta \ , \ \frac{\partial \Psi}{\partial \xi} = u_\xi \tag{4c}$$

$$+\text{B.C.} \tag{4d}$$

where ω stands for the vorticity, ψ for the stream, $\mathbf{u} = (u_\xi, u_\eta)$ for the velocity field, and ξ and η the coordinates within a given referential (i.e. $\xi = x$, $\eta = y$ in Cartesian coordinates, and ξ is the tangential coordinate, η the normal coordinate in curvilinear coordinates). We use a classical discretization method:

- A finite centered difference scheme of order 2 for diffusion and convection terms.
- A Euler scheme with implicit diffusion and explicit convection for equation (4a).
- An Poisson Problem for equation (4b) with ω solution of 4a at the current time.

The algorithm for solving Navier-Stokes equations writes:

1. Generate the mesh for the different subdomains
   - Boundary layers meshes are generated first.
   - The remaining space is meshed using a Cartesian grid overlapping the boundary layers grids. The size of the overlap is two Cartesian cells.

   For each time step:
2. Solve vorticity (ω) equations in the whole domain using multiplicative Schwarz:
   - Compute the RHS for equation (4a) in curvilinear and Cartesian domains.
   - Solve equation (4a) for curvilinear domains.
   - Send the new virtual boundaries to the Cartesian domain using a bilinear interpolation.
   - Solve equation (4a) for Cartesian domain.
   - Send the new virtual boundaries to the curvilinear domains using a bilinear interpolation.
   - Continue the process until the boundary residue go below the fixed tolerance.
3. Solve stream (Ψ) equations in the whole domain using multiplicative Schwarz:
   - Compute the RHS for equation (4b) in curvilinear and Cartesian domains.
   - Solve equation (4b) for curvilinear domains.
   - Send the new virtual boundaries to the Cartesian domain using a bilinear interpolation.
   - Solve equation (4b) for Cartesian domain.
   - Send the new virtual boundaries to the curvilinear domains using a bilinear interpolation.
   - Continue the process until the boundary residue go below the fixed tolerance.
4. Check the divergence of the velocity

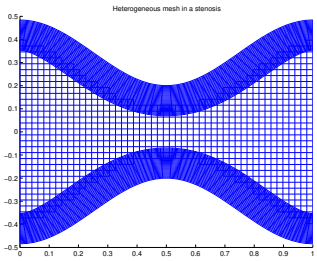

Figure 10. Overset mesh for NS's flow in a stenosis.

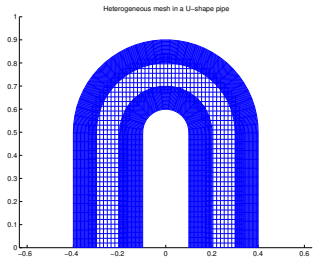

Figure 11. Overset mesh for NS's flow in a pipe.

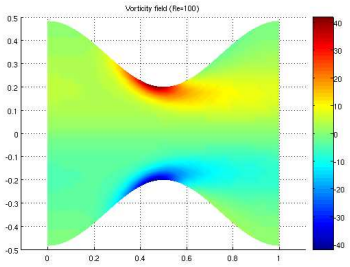

Figure 12. Vorticity field for NS's flow in a stenosis.

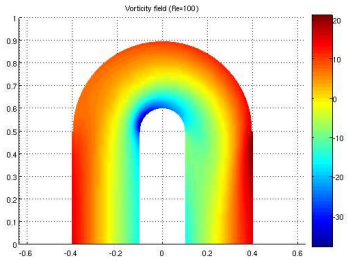

Figure 13. Vorticity field for NS's flow in a pipe.

The solution given in this code is tested on the two configurations given in Figure 10 and Figure 11.

The vorticity field of the domain for a Reynolds number equal to 100, is given in Figure 12 and Figure 13.

The computations are done with three different Reynolds numbers: $Re = 1$, $Re = 50$, $Re = 100$. The velocity at the inlet is defined by $u_x = (0.5 - y)(0.5 + y), u_y = 0$. We also apply this boundary condition at the outlet. No-slip boundary conditions are applied at other boundaries.

Each run is repeated on three sets of grids. Let us denote $G_0 = (N_x \times N_y)$ for the Cartesian grids where N_x and N_y are the number of nodes respectively in x and y directions, $G_k = (N_i \times N_e)$ for the curvilinear grids where N_i and N_e are the number of nodes respectively in tangent and normal directions to the boundary k. These three grids are

$$G_0 = (25 \times 20),\ G_1 = (25 \times 15),\ G_2 = (25 \times 15)$$
$$G_0 = (50 \times 40),\ G_1 = (50 \times 30),\ G_2 = (50 \times 30)$$
$$G_0 = (75 \times 60),\ G_1 = (75 \times 45),\ G_2 = (75 \times 45)$$

The order of convergence in L2 norm for each test case are given in Figure 14 and Figure 15.

One can notice that, as expected, the computed order of convergence in L2 norm based on these three overset grids, deteriorates as the Reynolds number increases. In practice, refining the grid improves the convergence order toward its asymptotic prediction.

We also notice that in both cases, the maximum error are localized next to the artificial and physical boundaries.

B. Grid convergence and solution verification with Adina Software

Despite the fact that the results in section V.A exhibit grid convergence, we still need to do comparison with numerical solutions produced independently of our code. For this purpose, we use *The ADINA System* - http://www.adina.com/. *The ADINA System* is a finite element analysis software for multi-physics problems. It allow us to solve Navier-Stokes solver in velocity-pressure formulation.

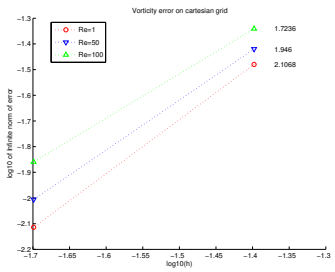

Figure 14. Convergence's order for the vorticity field for NS's flow in a stenosis.

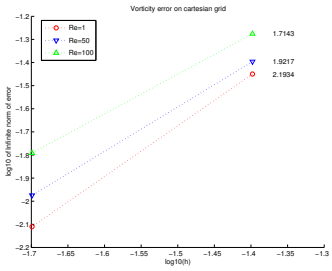

Figure 15. Convergence's order for the vorticity field for NS's flow in a pipe.

The following procedure gives the different steps done to verify the solution produce by the heterogeneous domain decomposition method:

- First, we compute the numerical solution (ω,Ψ) using our algorithm.
- Then, we compute the numerical solution ($\mathbf{u_a}$,p_a) using ADINA Software on unstructured mesh (quadrilateral bilinear elements).
- Thus, we can calculate the vorticity (ω_a) and stream (Ψ_a) fields on the unstructured ADINA grid using the solution ($\mathbf{u_a}$,p_a).
- The next step it is to interpolate the calculated fields (ω_a,Ψ_a) on the heterogeneous structured grids using bilinear interpolation. We obtain ($\tilde{\omega}_a$,$\tilde{\Psi}_a$).
- The last step is to compute the relative errors in L_2 norm $e_\omega = ||\omega - \tilde{\omega}_a||_2$ and $e_\Psi = ||\Psi - \tilde{\Psi}_a||_2$

Our calculations show that the relative errors e_ω and e_Ψ decreased as the grid become finer for both codes. However, this comparison between the error doesn't provide the convergence order of our method. Indeed, errors e_ω and e_Ψ can be decomposed into consistency error, interpolation error from one grid to an other, and the change of variable (($\mathbf{u_a}$,p_a)$\rightarrow$ (ω_a,Ψ_a)).

In the next section, we discuss briefly the potential applications of our method to fluid-structure interaction.

VI. Future work and conclusion

We are currently expanding this work to include fluid structure interactions.

A. Basic equation of fluid structure interaction

The general equations used to describe the deformation of the structure[17] are

$$\rho s_{tt} - \nabla \cdot \tau = b \text{ in } \Omega_s(0) \quad (5a)$$

$$n \cdot \tau = g \text{ in } \Gamma_{fs} \quad (5b)$$

where τ is the stress tensor, s is the displacement, ρ is the density.

This equation are coupled with Navier-Stokes equations, either in pressure-velocity formulation or vorticity-stream formulation.

The interface condition between the fluid domain and the solid domain is given by

$$\phi = s \quad (6a)$$

$$\varphi = s^{\cdot} \quad (6b)$$

$$n \cdot \tau = \nu(\nabla u + \nabla u^T) - pI)n \quad (6c)$$

where φ, ϕ, u ,p are respectively the flow domain speed, the mapping of the fluid domain, the velocity of the fluid the pressure.

To solve the coupled problem, the fluid-structure algorithm used is the following :

- First, assuming that the geometry is constant with fixed boundary conditions for the flow, we solve Navier-Stokes problem. And so, we can update (6)
- Then we solve the structure problem
- In a third step, we update the interface conditions of equations (6), and we determine the function ϕ, and the speed of the fluid domain.
- The resulting displacement velocity $\dot{s}$ is then used as a boundary condition for the fluid domain.
- We continue until convergence.

The advantage of our domain decomposition is that it allow us to consider fluid-structure interaction only in the boundary layer instead of considering the influence of the wall displacement in the whole domain. Further the BL mesh is well adapted to the nature of the interface condition[18,19] . In the mean time, we are implementing the Peskin's method in this framework.

B. Application to Peskin Method

The Immersed Boundary Method (IBM), originally developed by C.S. Peskin,[3] is a very elegant method of simulating fluid-structure interactions. It combines Eulerian and Lagrangian descriptions of flow and moving elastic boundaries using Dirac delta functions. Incompressible Navier-Stokes and Elasticity theory can be unified by the same set of equations to get a combined model of the interaction. There are numerous applications of the IBM in Bio-Engineering or in more general Computational Fluid Dynamics applications. The numerical study of the stability and accuracy of the method is based on the implementation of several mathematical tools such as Fourier filters, multigrid solvers, Newton-Krylov methods, numerical methods for ordinary differential equations with singular source terms, domain decomposition methods or multilevel discretization.[20]

In order to apply the IBM in the framework we defined in this paper, a curvilinear mesh is built along the moving boundaries (the membrane in Figure 17 for example) and a Cartesian grid to fill the remain of the domain. At each iteration, using the Peskin's model in the curvilinear domain, we can compute the external force in NS's equations. Once the NS's equations solved, we can recompute the position of the moving boundary. Thanks to the low displacement of the boundary, the curvilinear mesh need to be recompute only every few time steps instead of each time step.

Since Peskin's method is very sensitive to discontinuity on mesh stepping, one should be careful to the position of the moving membrane with respect to the virtual boundaries between domains.

This preliminary work is completed by some fluid experiments for future validation studies as well as for calibrating our model.

C. Description of the experimental system

We consider a pipe split (partially) in two parallel channels by a solid wall. In the middle of this wall, an elastic membrane is placed. We are testing different configurations with pulsating flow on one side and free flow on the other side.

To conclude, we can sum up the features of our approach with heterogeneous domain decomposition as follows:

1. Each subdomain can use a fast solver that takes full advantage of either the stretching of the mesh in one space direction for boundary layer domains, or the regular data structure with Cartesian grids used for the main part of the flow.
2. Simplicity of the implementation, grid generation, and memory allocation due to the use of the additive Schwarz method for the iteration process between overlapping non matching grids.

Figure 16. Experimental device

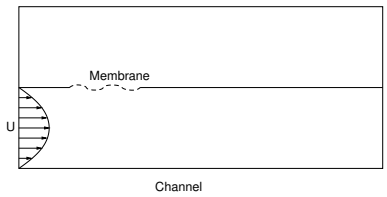

Figure 17. Schema of the experimental device

3. Fast convergence of the domain decomposition algorithm thanks to the use of an acceleration procedure to speed up the convergence of the Schwarz method.

4. Biologically motivated domain decomposition.

References

[1]Zarins, C., Giddens, D., Bharadvaj, B., Sottiurai, V., Mabon, R., and Glagov, S., "Carotid bifurcation atherosclerosis: quantitative correlation of plaque formation with velocity profiles and wall shear stresses," *Circulation Research*, Vol. 53, 1993, pp. 502–514.

[2]R.Krams, "Evaluation of endothelial shear stress and 3d geometry as factors determining the development of arteriosclerosis and remodeling in human coronary arteries in vivo," *Thrombosis and Vascular Biology*, Vol. 117, 1997, pp. 2061–2065.

[3]C.Peskin, "The Immersed Boundary Method," *Acta Numerica*, Vol. 11, 2002, pp. 1–39.

[4]Quarteroni, A., Tuveri, M., and Veneziani, A., "Computational Vascular Fluid Dynamics: Problems, Models and Methods," *Computing and Visualization in Science*, Vol. 2, 2000, pp. 163–197.

[5]Quarteroni, A., Veneziani, A., and Zunino, P., "Mathematical and Numerical Modeling of Solute Dynamics in Blood Flow and Arterial Walls," *SIAM Journal on Numerical Analysis*, Vol. 39, No. 5, 2001, pp. 1488–1511.

[6]Quarteroni, A., Veneziani, A., and Zunino, P., "A Domain Decomposition Method for Advection-Diffusion Processes with Application to Blood Solutes," *SIAM Journal on Scientific Computing*, Vol. 23, No. 6, 2002, pp. 1960–1981.

[7]Fischer, P., Kruse, G., and Loth, F., "Spectral Element Methods for Transitional Flows in Complex Geometries," *Journal of Scientific Computing*, Vol. 17, 2002, pp. 1–3.

[8]Eckhaus, W., *Asymptotic Analysis of Singular Perturbations*, North–Holland, Amsterdam, 1979.

[9]J.Kevorkian and J.D.Cole, *Perturbation Methods in Applied Mathematics*, Springer Verlag, New York, 1981.

[10]R.J.O'Malley, *Singular Perturbation Methods for Ordinary Differential Equations*, Vol. 89, Springer Verlag Applied Mathematical Sciences, 1991.

[11]Marchuk, G., Kuznetsov, Y., and Matsokin, A., "Fictitious domain and domain decomposition methods," *Soviet Journal of Numerical Analysis and Mathematical Modelling*, Vol. 1, 1986, pp. 3–35.

[12]Garbey, M., Shyy, W., Hadri, B., and Rougetet, E., "Efficient Solution Techniques for CFD and Heat Transfer," *ASME Heat Transfer/Fluids Engineering Summer Conference*, 2004.

[13]Garbey, M., Kuznetsov, Y. A., and Vassilevski, Y. V., "A Parallel Schwarz Method for a Convection-Diffusion Problem," *SIAM Journal on Scientific Computing*, Vol. 22, No. 3, 2001, pp. 891–916.

[14]Garbey, M. and Vassilevski, Y. V., "A parallel solver for unsteady incompressible 3D Navier-Stokes equations," *Parallel Computing*, Vol. 27, No. 4, 2001, pp. 363–389.

[15]Garbey, M. and Dervout, D. T., "On some Aitken like acceleration of the Schwarz Method," *International Journal for Numerical Methods in Fluids*, Vol. 40, 2002, pp. 1493–1513.

[16]Garbey, M., "Acceleration of the Schwarz method for elliptic problem," *SIAM Journal on Scientific Computing*, To appear.

[17]Ciarlet, P., *Mathematical Elasticity Volume I:Three Dimensional Elasticity*, Elsevier, North-Holland, 1988.

[18]Fast, P. and Shelley, M., "A moving overset grid method for interface dynamics applied to non-Newtonian HeleShaw flow," *Journal of Computational Physics*, Vol. 195, No. 1, 2004, pp. 117–142.

[19]Fast, P. and Henshaw, W., "Time accurate simulation of viscous flow around deforming bodies using overset grids," *32nd AIAA Fluid Dynamics Conference*, No. AIAA-20012604, 2001.

[20]Pacull, F. and Garbey, M., "A Numerical Experimental Study of the Immersed Boundary Method," *Proceedings of the 16th Intl. Conf. on Domain Decomposition Methods*, 2005, To appear.