International Journal of Computer Mathematics, 2013
Vol. 90, No. 3, 615–629, http://dx.doi.org/10.1080/00207160.2012.727988

Taylor & Francis
Taylor & Francis Group

A parallel fully coupled implicit domain decomposition method for numerical simulation of microfluidic mixing in 3D

Feng-Nan Hwang[a], Xiao-Chuan Cai[b], Yu-Lun Cheng[a] and Chia-Wen Tsao[c]*

[a]*Department of Mathematics, National Central University, Jhongli 32001, Taiwan;* [b]*Department of Computer Science, University of Colorado, Boulder, CO 80309, USA;* [c]*Department of Mechanical Engineering, National Central University, Jhongli 32001, Taiwan*

(*Received 21 April 2011; revised version received 28 February 2012; accepted 21 August 2012*)

A parallel fully coupled implicit fluid solver based on a Newton–Krylov–Schwarz algorithm is developed on top of the Portable, Extensible Toolkit for Scientific computation for the simulation of microfluidic mixing described by the three-dimensional unsteady incompressible Navier–Stokes equations. The popularly used fractional step method, originally designed for high Reynolds number flows, requires some modification of the inviscid-type pressure boundary condition in order to reduce the divergence error near the wall. On the other hand, the fully coupled approach works well without any special treatment of the boundary condition for low Reynolds number microchannel flows. A key component of the algorithm is an additive Schwarz preconditioner, which is used to accelerate the convergence of a linear Krylov-type solver for the saddle-point-type Jacobian systems. As a test case, we carefully study a three-dimensional passive serpentine micromixer and report the parallel performance of the algorithm obtained on a parallel machine with more than one hundred processors.

Keywords: Navier–Stokes equations; microfluidics; fully implicit methods; domain decomposition; parallel processing

2010 AMS Subject Classifications: 65H10; 65N30; 65N55; 76D05

1. Introduction

Mixing enhancement is crucially important in many branches of emerging microfluidic technologies. Several different types of micromixers are available or proposed. Generally speaking, micromixers can be classified into two major groups [16,27]. One of them is the so-called active micromixer, in which mixing is achieved by some external perturbation applied, for example, through the pressure field [25] by magneto-hydrodynamics [4] or through acoustic disturbances [1,41]. The other group consists of passive micromixers, where mixing relies solely on diffusion or chaotic advection without importing external energy. In comparison with the active type, the passive-type micromixers are more popular due to several advantages, including efficient mixing performance, simple fabrication and high integration level with other lab-on-a-chip applications. To optimize the mixing performance, the main goal of the design of passive

*Corresponding author. Email: cwtsao@ncu.edu.tw

micromixers is to increase the contact surface between two fluids and to decrease the diffusion path between them by changing the microchannel geometry. The mixing enhancement techniques include using a staggered herringbone microchannel to create three-dimensional twisting flows inside the microchannel [12,17,20,35,39], placing obstacles inside the microchannel to create chaotic mixing [30,33] or changing velocity profiles with various microchannel cross-section geometries [37].

Most of the numerical simulations are carried out by using commercial software packages; e.g. ANSYS CFX [23], COMSOL [36], Fluent [8] or CFD-ACE [33,39]. These package are easy to use because of the friendly interface, but are usually not parallel, or restricted to very small number of processors, and therefore cannot be used for high-fidelity simulations, which require very fine meshes and large number of processors. Recently, Glatzel *et al.* [15] evaluated the performance of some available computational fluid dynamics software packages with emphasis on microfluidic applications and concluded that there is a real need for faster and more accurate algorithms and software for microfluidic simulations on large-scale parallel computers. Thus, the aim of this paper is to investigate a fully parallel approach for solving incompressible Navier–Stokes (NS) equations, which is used to simulate the mixing of fluids in microsystems. We hope the algorithms and software developed in this paper will provide engineers with a more efficient approach for the design of microfluidic devices, and a scalable tool for understanding the physics of fluids at the nano or micro level. As a numerical example to demonstrate the applicability of our fluid solver to microfluidic systems and to evaluate the performance on a parallel machine, we focus on a three-dimensional serpentine microchannel, which is the passive micromixing model proposed by Liu *et al.* [24].

In this paper, we develop and study some fully implicit methods that have recently gained in popularity [3,5,6,11,26,28,40], because they allow much larger time-step size compared to fully explicit or linearly implicit methods due to the stability constraint of the time integrator employed, and they are more scalable on large-scale parallel computers. Additionally, they are able to accurately capture the nonlinear coupling between components, conserve more physical quantities, such as mass, momentum or energy for a longer period of simulation time. As pointed out by the authors of [21], the inviscid-type pressure boundary condition in the classical temporal fractional step method, which is originally designed for high Reynolds number flows, needs to be modified in order to reduce the divergence error near the boundaries (see Figure 16.2 on p. 517 [21]). On the other hand, the fully coupled implicit approach considered in this paper works well for both high Reynolds number laminar flows and lower Reynolds number microflows without changing the boundary condition. The price to pay is that we have to solve a nonlinear system at every time step. The inexact Newton method is one of the popular approaches for solving such nonlinear systems arising from time-dependent PDE problems because of its robustness and fast convergence. The kernel of the Newton-type method is the linear Jacobian solver, which is the most expensive part of the algorithm and the design of an efficient preconditioner is crucial for the success of the algorithm.

Our parallel algorithm is based on a Newton–Krylov–Schwarz (NKS) algorithm [7], which consists of three key ingredients: an inexact Newton method with backtracking as the nonlinear solver, a Krylov subspace method [31] as the linear solver for the Jacobian systems together with a parallel overlapping Schwarz domain decomposition-based preconditioner [34,38] to accelerate the convergence of the linear solver. The major advantage of NKS is that it is fully parallel, since one does not need to split the velocity and pressure fields. Furthermore, NKS is extendable to simulate other more complex full microsystems involving coupled electrical, mechanical, thermal and fluid components.

The rest of this paper is organized as follows. In the next section, we briefly mention the microfluidic model based on incompressible NS equations, followed by a description of a Newton–Krylov nonlinear solver, a parallel Schwarz preconditioner for the saddle-point-type Jacobian

system, and an overview of parallel micro-flow simulator. In Section 4, numerical results for three microfluidic mixing problems are presented. Concluding remarks are given in Section 5.

2. A model for incompressible micro-flows and their mixing

To simulate the motion of fluids in a microchannel, we consider the three-dimensional unsteady incompressible NS equations defined on a bounded domain Ω with the boundary $\Gamma = \Gamma_D \cup \Gamma_N$ [21],

$$\begin{aligned} \rho\left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u}\cdot\nabla\boldsymbol{u}\right) - \nabla\cdot\sigma = 0 \quad & \text{in } \Omega\times(0,T), \\ \nabla\cdot\boldsymbol{u} = 0 \quad & \text{in } \Omega\times(0,T), \\ \boldsymbol{u} = g \quad & \text{on } \Gamma_D\times(0,T), \\ \nabla\cdot\sigma = 0 \quad & \text{on } \Gamma_N\times(0,T), \\ \boldsymbol{u} = \boldsymbol{u}_0 \quad & \text{in } \Omega \text{ at } t = 0, \end{aligned} \tag{1}$$

where $\boldsymbol{u} = (u_1, u_2, u_3)^{\mathrm{T}}$ is the velocity, ρ is the fluid density and σ is the Cauchy stress tensor defined as $\sigma = -p\boldsymbol{I} + \mu[(\nabla\boldsymbol{u}) + (\nabla\boldsymbol{u})^{\mathrm{T}}]$, where p is the pressure, $\boldsymbol{I}$ is a second-order identity tensor and μ is the dynamic viscosity. Here, we impose two types of boundary conditions: the Dirichlet boundary condition on Γ_D and the homogeneous Neumann boundary condition on Γ_N and assume that the flow is stationary at the beginning of the computation. The Reynolds number, Re, is defined as $\rho(Q/A)D_h/\mu$, where Q is the volumetric flow rate through the channel, A is the cross-sectional area and $D_h = 4A/P$ (P is the wetting perimeter of the channel) is the hydraulic diameter of the channel. The Reynolds number is quite low in a microfluidic system, typically ranging from 0.01 to 100 [27]. However, due to the presence of some abrupt turns in the computational domain, e.g. a 3D micromixer, the effect of the convective acceleration term plays an important role during the numerical simulation, thus the diffusive term cannot be neglected. To measure the degree of mixing of fluids, we solve a 3D convection–diffusion equation at certain time steps,

$$\boldsymbol{u}\cdot\nabla C - D\triangle C = 0, \tag{2}$$

where $\boldsymbol{u}$ is the velocity field obtained from the solution of the NS equations, D is the diffusivity coefficient of the species and C is the concentration of the species. The corresponding mixing efficiency at a cross-section of a channel is defined as

$$M = 1 - \sqrt{\frac{1}{n_c}\sum_{i=1}^{n_c}\left(\frac{C_i - \bar{C}}{\bar{C}}\right)^2}, \tag{3}$$

where n_c is the number of mesh points on the cross-section and C_i is the concentration at the mesh point and $\bar{C}(= 0.5)$ is the average number of concentration.

3. A parallel fully coupled and fully implicit fluid solver

Our parallel time-dependent 3D incompressible fluid solver is implemented on top of the Portable, Extensible Toolkit for Scientific computation (PETSc) [2]. The solver has been validated and successfully applied to blood flows in the arteries [19]. In addition, the parallel fluid solver has

been integrated with other state-of the-art pre-processing and post-processing software packages, including (1) Cubit [9] for 3D unstructured finite element mesh generation; (2) ParMETIS [22] for mesh partitioning for the purpose of parallel processing; (3) ParaView [29] for scientific visualization of numerical results and conducting data analysis. Below we give a description of the discretization scheme and the parallel solution algorithm employed in the fluid solver.

To discretize the NS Equations (1), we use an implicit backward Euler finite difference method for the temporal variable and a stabilized $P_1 - P_1$ Galerkin/least-squares finite element method [13] in the spatial domain covered by a given tetrahedral mesh. At each time step, it is necessary to solve a large, sparse, nonlinear algebraic system

$$F(x) = 0, \tag{4}$$

where the vector x corresponds to the nodal values of $\boldsymbol{u}_h = (u_h^1, u_h^2, u_h^3)$ and p_h at the time $t = (n+1)\Delta t$. Here, only a uniform time step Δt is considered. To solve the nonlinear algebraic system a NKS algorithm is employed as follows. Let $x^{(0)}$ be a given initial guess, which is taken from the velocity and pressure solutions at the previous time step and assume $x^{(k)}$ is the current approximation of the exact solution x^*. Then a new approximation $x^{(k+1)}$ can be computed via the following steps:

Step 1: Find a Newton direction $s^{(k)}$ by solving the following preconditioned Jacobian system approximately by a Krylov subspace method, such as the generalized minimal residual method (GMRES) [32],

$$J_k M_k^{-1} y = -F(x^{(k)}) \quad \text{with} \quad s^{(k)} = M_k^{-1} y, \tag{5}$$

where J_k is the Jacobian of F evaluated at $x^{(k)}$ and the additive Schwarz preconditioner, M_k^{-1}, is defined in detail below.

Step 2: Obtain the new approximation $x^{(k+1)} = x^{(k)} + \lambda^{(k)} s^{(k)}$, where $\lambda^{(k)} \in (0, 1]$ is a damping parameter used to enhance the robustness of Newton-type methods [10].

Here, we define the additive Schwarz preconditioner. Let $\{\Omega_i^h,\ i = 1, \ldots, N\}$ be a non-overlapping subdomain partition, whose union covers the entire domain Ω and its mesh $\mathcal{T}^h$. We denote by $\mathcal{T}_i^h$ the collection of mesh points in Ω_i^h. In our implementation, the subdomain partition is obtained through ParMETIS and each subdomain problem is assigned to a single processor of the parallel computer. For many cases, such partitioning is optimal or nearly optimal, i.e. the total number of mesh points on each subdomain is roughly equal and the interface mesh points between subdomains are minimized. To obtain overlapping subdomains, we expand each subdomain Ω_i^h to a larger subdomain $\Omega_i^{h,\delta}$ with the boundary $\partial\Omega_i^{h,\delta}$. Here, δ is an integer indicating the level of overlap. We assume that $\partial\Omega_i^{h,\delta}$ does not cut any elements of $\mathcal{T}^h$. Similarly, we denote by $\mathcal{T}_i^{h,\delta}$ as the collection of mesh points in $\Omega_i^{h,\delta}$. We define the additive Schwarz preconditioner for the Jacobian system (5), which is an extension of that for the saddle-point-type Stokes equations as follows [18]. First, we introduce the subdomain velocity space

$$V_i^h = \{\boldsymbol{v}^h \in V^h \cap (H^1(\Omega_i^{h,\delta}))^3 : \boldsymbol{v}^h = 0 \text{ on } \partial\Omega_i^{h,\delta}\}$$

and the subdomain pressure space

$$P_i^h = \{p^h \in P^h \cap L^2(\Omega_i^{h,\delta}) : p^h = 0 \text{ on } \partial\Omega_i^{h,\delta} \backslash \Gamma_D\},$$

where V^h and P^h are the linear finite element spaces defined on the domain Ω for the velocity and the pressure, respectively. $L^2(\Omega)$ and $H^1(\Omega)$ are the standard notations with the usual meanings in the finite element literature [13,14]. On the physical boundaries, we impose the Dirichlet condition according to the original Equations (1). On the artificial boundaries, we assume both $\boldsymbol{u} = 0$ and $p = 0$. Similar boundary conditions were used in [18]. Although the well-posedness

of the subdomain problems defined in the Schwarz preconditioner by setting such boundary conditions for the Stokes-like problem is still an open problem, it works well in practice Barker and Cai [3], Hwang and Cai [18] and Hwang *et al.* [19].

Let $R_i : V^h \times P^h \to V_i^h \times P_i^h$ be a global-to-local restriction operator, which returns all degrees of freedom (both velocity and pressure) associated with the subspace $V_i^h \times P_i^h$. R_i is an $4n_i \times 4n$ matrix with values of either 0 or 1, where n and n_i are the total number of mesh points in $\mathcal{T}^h$ and $\mathcal{T}_i^{h,\delta}$, respectively, and $\sum_{i=1}^{N} 4n_i \geq 4n$. Note that for linear elements, we have four variables per mesh point, three for the velocity and one for the pressure. Then, the local-to-global interpolation operator R_i^{T} can be defined as the transpose of R_i. Using the restriction and interpolation operators, we write the additive Schwarz preconditioner in the matrix form as

$$M_k^{-1} = \sum_{i=1}^{N} R_i^{\mathrm{T}} J_i^{-1} R_i,$$

where J_i^{-1} is subspace inverse of $J_i = R_i J R_i^{\mathrm{T}}$. We remark that the multiplication of R_i (and R_i^{T}) with a vector does not involve any arithmetic operation, but does involve communication in a distributed memory parallel implementation. The restriction operator R_i collects the data from neighbouring subdomains, and the local-to-global prolongation operator R_i^{T} sends partial solution to neighbouring subdomains. In practice, to save the computational cost and the memory use, the J_i^{-1} in M_k^{-1} are often replaced by an inexact solver, such as an incomplete LU decomposition (ILU) with some levels of fill-ins as we use for our numerical experiments presented in the next section.

4. Numerical results and discussions

We report the simulation results for the three-dimensional serpentine microchannel flows together with a straight microchannel and a square-wave microchannel for the purposes of comparison; see Figure 1 for the geometrical configurations for these three test cases. Note that only one unit for each case is shown in the figure and the microchannel geometric models are constructed by connecting these units repeatedly. The number of units is selected so that the travelling paths for each case are roughly the same.

The simulations are performed using $np = 64$ starting from $t = 0$ and finishing at $t = 10$ with $\Delta t = 0.1$, where np is the number of processors. Detailed information for the three test cases are summarized in Table 1. The unstructured tetrahedral finite element meshes are generated by using Cubit and the mesh size for each case is roughly equal to 0.1.

All numerical simulations are performed on the Vger PC cluster with a peak performance of 5184 Gflop/s at the National Central University in Taiwan. The system consists of 108 compute nodes, and each node has two Intel Xeon 3.0 GHz Dual-Core processor with 4 GB memory. The nodes are interconnected by a InfiniBand switch. All computations are done in double precision. The execution time is reported in seconds. The Jacobian matrices are constructed by a hybrid approach; i.e. all the linear terms and nonlinear terms associated with the Galerkin formulation are computed analytically, other stabilization terms are approximated by multicolored finite differences.

At each time step, we employ NKS to solve Equation (4) with the previous time-step solution as the initial guess. We claim the intermediate solution converges when the stopping condition for Newton

$$\|F(x^{(k)})\| \leq \max\{10^{-6}\|F(x^{(0)})\|, 10^{-10}\}$$

is satisfied. A cubic lines search technique [10] is employed to determine the step length $\lambda^{(k)}$. Note that the previous time-step solution is a good initial guess for most cases so that the line

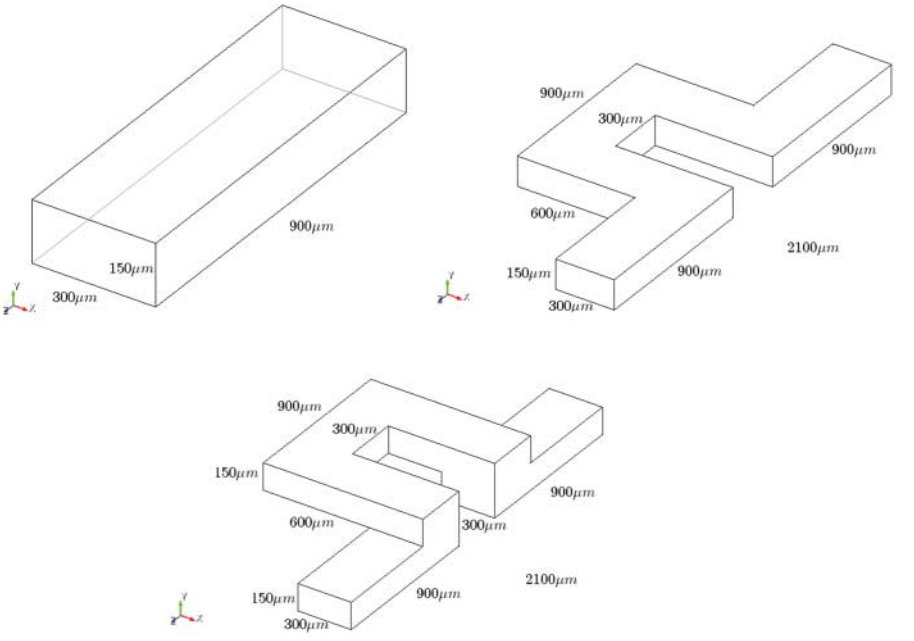

Figure 1. The dimensions of three microchannel models (only one unit is shown). From left to right: 1D straight, 2D square-wave and 3D serpentine.

Table 1. Detailed information of three test cases.

Models	# of elements	# of nodes	# of unknowns
1D straight	590,445	115,140	460,560
2D square-wave	558,695	109,704	438,816
3D serpentine	642,300	124,569	498,276

search procedure is seldom invoked. A right additive Schwarz preconditioned GMRES with a zero initial guess is employed to solve the Jacobian system. The accuracy of the solution to the Jacobian system is controlled by the stopping condition

$$\|F(x^{(k)}) + (J_k(x^{(k)})M_k^{-1})(M_k s^{(k)})\| \leq \max\{10^{-4}\|F(x^{(k)})\|, 10^{-10}\}.$$

Due to the nature of convection-dominated characteristic (typical Peclet number ranges from 10^2 to 10^5), the Galerkin/least-squares finite element method is employed to discretize Equation (2), where the stabilization parameter employed is the one suggested by Franca *et al.* [14]. The corresponding linear system is solved by one-level additive Schwarz preconditioned GMRES, with the LU decomposition as the subdomain solver.

4.1 *Simulation results*

The fluids in the channel are assumed to be stationary at $t = 0$. At the beginning of the numerical simulation, two fluids with the same velocity are injected into a single microchannel and they merge at the vertical middle line at the inlet. The concentration of the left stream is set to be 0 (blue) and the right stream is set to be 1 (red) as the inlet boundary condition for the concentration in Equation (2) and the homogeneous Neumann boundary condition is imposed on the wall and at the outlet. Such a condition implies that there is no mixing taking place before entering the

micromixer. Note that it is observed that the flow in the 3D serpentine channel reaches the quasi-steady state at around $t = 2$ for the case of $Re = 6$ and $t = 6$ for the case of $Re = 70$. All the figures for the concentration distribution and its contours at the viewing windows, the streamlines and the pressure distributions shown in this subsection are produced by using ParaView.

Figure 2 displays the concentration distribution in each microchannel at $t = 2$ for the case of $Re = 6$. Due to the nature of laminar flows in the micro-scale channel, mixing relies mainly on diffusion. As shown on the top picture of Figure 2, two fluid streams move forward smoothly along the 1D straight channel without any fluid perturbation involved and mixing occurs only along the interface of the two fluids near the middle of the channel. The situation is improved for the 2D square-wave micromixer (the middle picture of Figure 2), since the fluids skewing at the 90° turn result in having a larger contact surface area at the fluid interface. This induces a better mixing. This can be seen clearly from the left column of Figure 3, which shows the concentration contours at different viewing windows. About one-third of the area of the cross-section is the complete mixing region (green area in the middle) near the outlet of the channel. Finally, both the bottom picture of Figure 2 and the right column of Figure 3 suggest that in the 3D serpentine micromixer, chaotic advection occurs, which greatly increases the contact surface area and shortens the mixing path between the two fluids, in other words, the mixing performance is significantly improved. Superior mixing performance is observed at the cross-section near the outlet of the 3D serpentine microchannel.

Figures 4 and 5 show a comparison of the streamlines and the pressure distribution in the 1D straight, 2D square-wave and 3D serpentine microchannel at $t = 6$ for the case of $Re = 70$. It should be noted that the visualization of the streamlines is useful for studying the dead volume or eddies in the microchannel. And the pressure drop is of interests to the engineers, as it provides important information about the pump needed to drive the flow in the micro-device.

4.2 *Impact of the viscosity*

We next evaluate the mixing performance of the serpentine microchannel with respect to different values of Reynolds number, which is measured by the volumetric flow rate Q through the channel ranging from 0.1 to 1.2 mL/min. The mixing efficiency defined in Equation (3) at a cross-section is used as the metric. Note that the total number of mesh points n_c on the cross-section is about 500. We show in Figure 6 the computed mixing efficiency at different viewing windows for four different values of the Reynolds number. It is clear that the mixing efficiency increases as the fluids flow towards the downstream direction and is at least 70% for all four cases when the fluids reach the 10th viewing window. As expected, the higher Reynolds number implies better mixing efficiency. Similar observation obtained from experiments was also reported in [24].

4.3 *Parallel performance of the algorithm*

To achieve the optimal performance of the parallel fluid solver in terms of the computing time, several parameters need to be well tuned. Particularly, in this section, we study how the algorithmic parameters involved Krylov–Schwarz algorithms, as well as some physical parameters, affect the overall performance of the algorithm applied to the microchannel flows. These parameters include the number of levels of ILU fill-ins, the degree of overlap for the additive Schwarz preconditioner, the geometric configuration of the microchannel and the Reynolds number. Such a study provides a guideline that helps the users to choose appropriate parameters in their numerical simulations. Note that the number of levels of ILU fill-ins and the degree of overlap are related to the solution quality of the subdomain problem, which affects significantly the overall performance. Since solving the subdomain problem is the most time-consuming step in the algorithm, our goal is

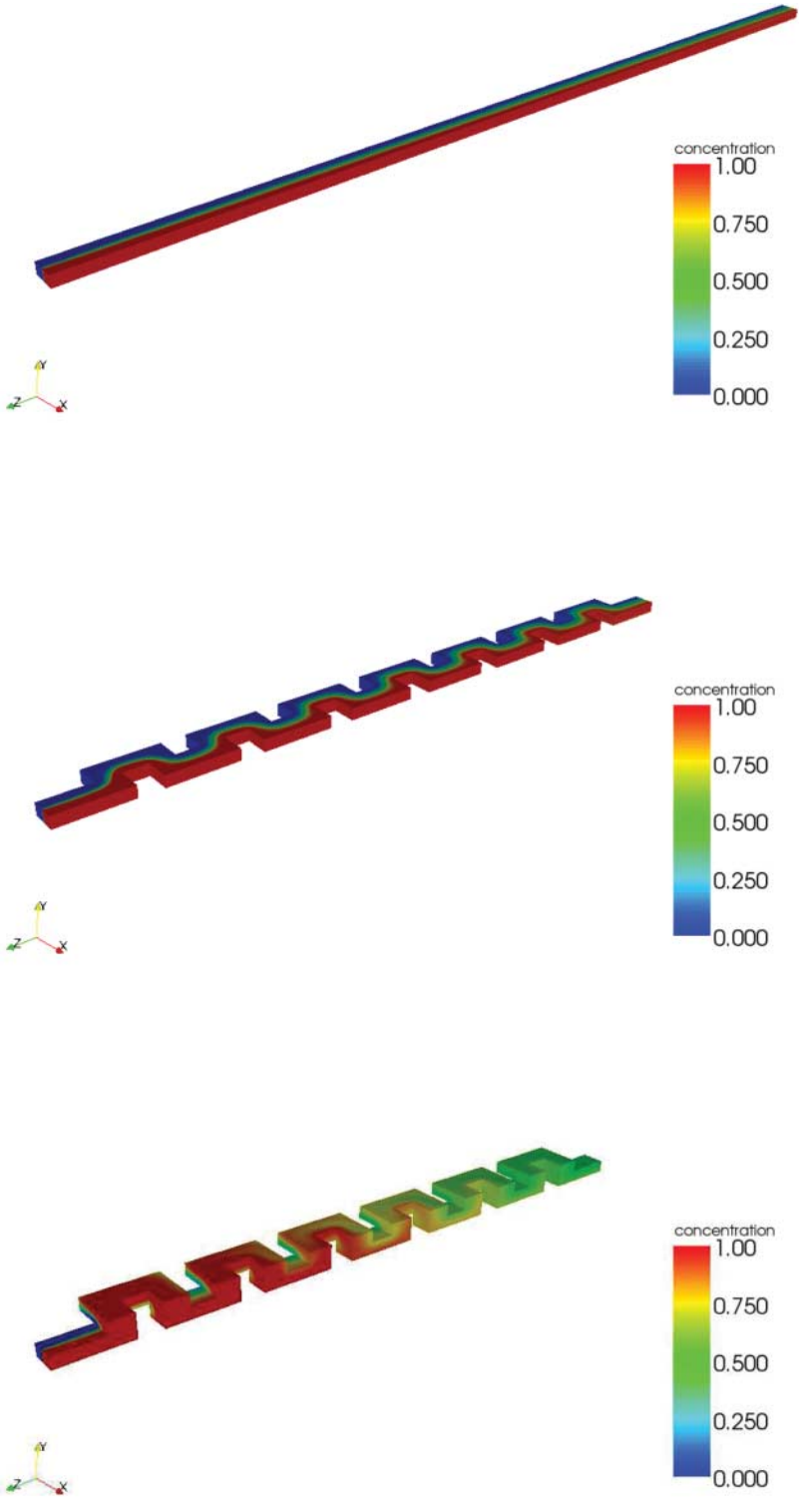

Figure 2. Concentration distribution in each microchannel for the case of $Re = 6$ at time $t = 2$. From top to bottom: 1D straight, 2D square-wave and 3D serpentine (colour online only).

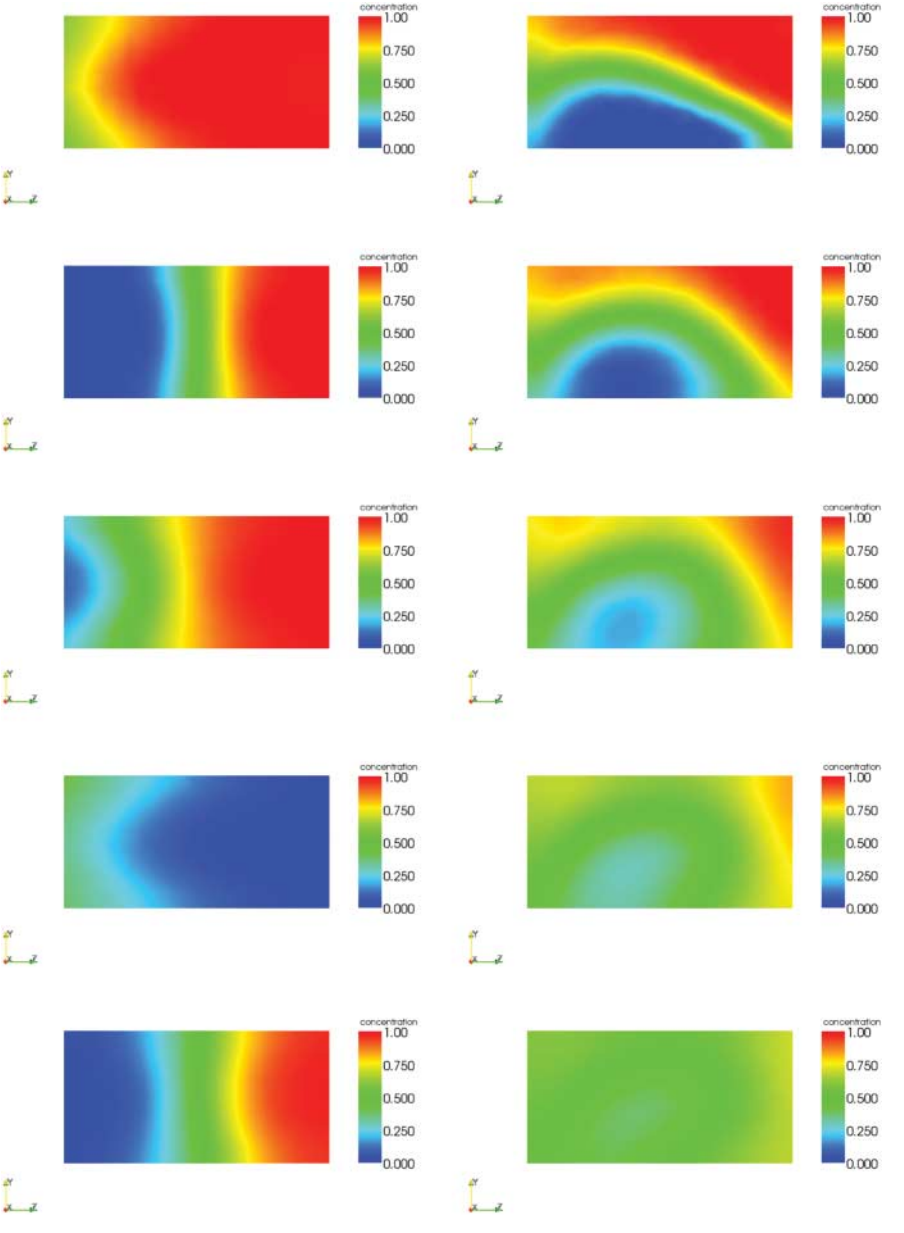

Figure 3. Concentration contours for 2D and 3D microchannels at the viewing windows 2, 4, 6, 8 and 10 at $t = 2$ from top to bottom. $Re = 6$ is considered. 2D (left) and 3D (right) (colour online only).

to reduce the computing time spent on the subdomain solution as much as possible but not to degrade too much the convergence rate of the Krylov subspace method, which is a trade-off between the number of iterations and the cost per iterations. Timing results reported in this section are obtained by running the simulation for 10 time steps. The total execution time, the average number of nonlinear iterations per time step (ANNI), the average number of linear iterations per Newton iteration (ANLI) are reported for the case of $Re = 6$ and $Re = 70$ in Table 2. As expected, the more ILU fill-ins and more overlap, the fewer GMRES iterations are required to achieve convergence. In these particular cases, ILU with a small number of fill-ins is too inexact

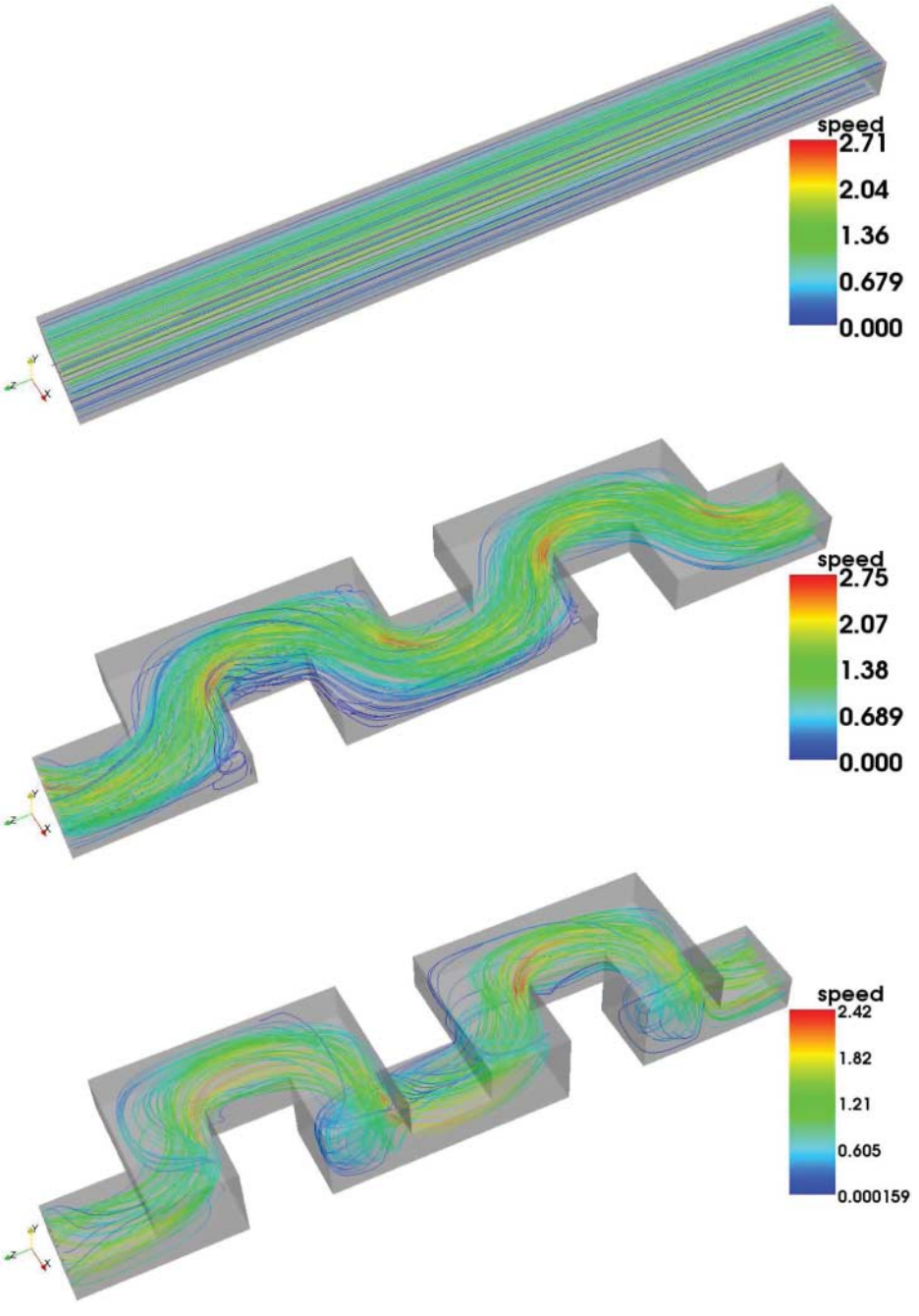

Figure 4. The streamlines for the 1D straight (top), the 2D square-wave (middle) and the 3D serpentine (bottom). Note that the last two units for all cases are shown in the figures (colour online only).

to make the Schwarz preconditioner efficient. The same trends for both the 1D straight and 2D square-wave microchannels are also observed (not shown here). Furthermore, the communication cost is high; therefore, we are not able to save any computing time by increasing the degree of overlap.

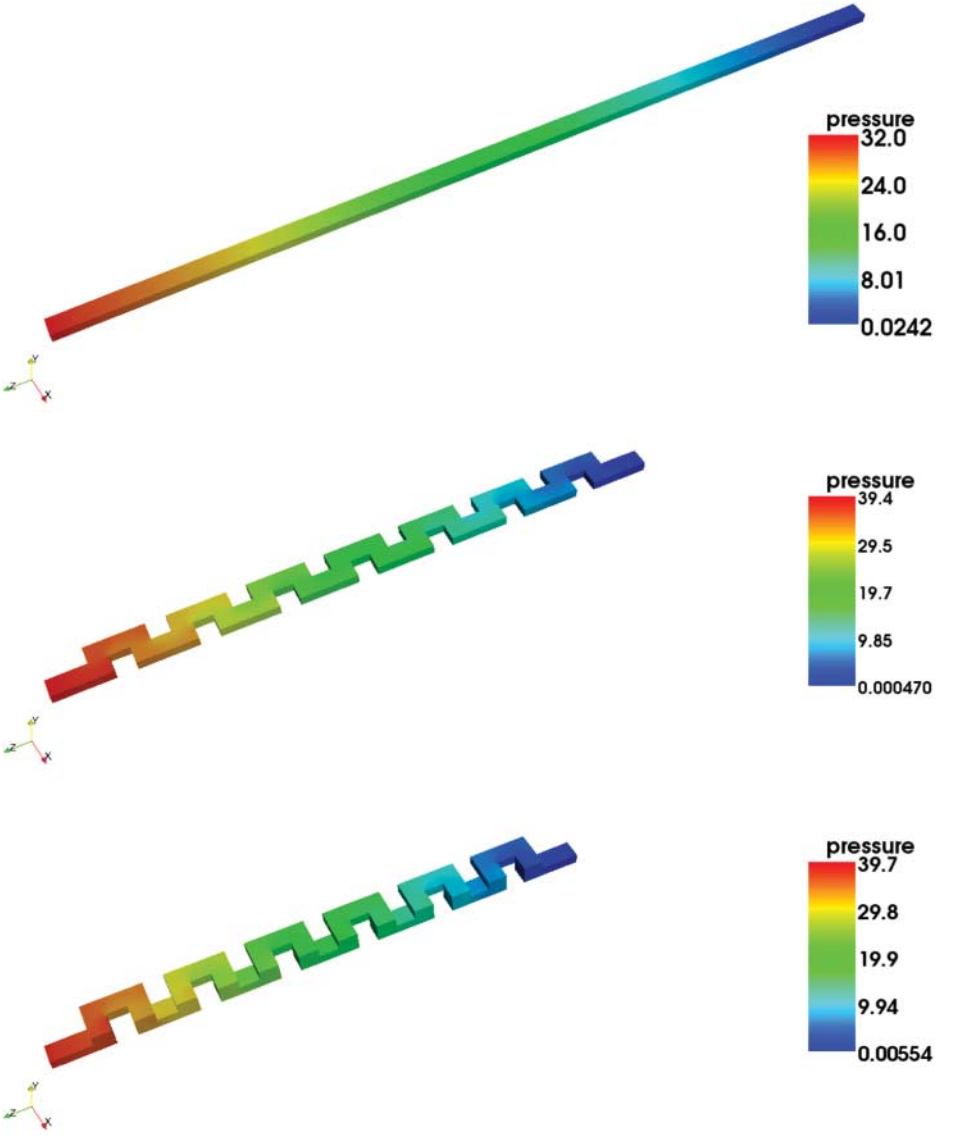

Figure 5. The pressure for the 1D straight (top), the 2D square-wave (middle) and the 3D serpentine (bottom) (colour online only).

Next we summarize the timing results for three cases for $Re = 6$ and $Re = 70$ in Table 3. This table suggests that, generally speaking, the larger the Reynolds number, the more difficult it is for the GMRES to converge. Without a coarse space, which may improve the communication between subdomains, the ANLI is usually high. Such a problem becomes more severe in the case of a long and thin computational domain such as a microchannel even for a low Reynolds number flow with $Re = 70$. In many situations, Jacobian solves reach the maximum number of iterations, which is set to be 500. A multilevel method could be used to further reduce the number of GMRES iterations for convergence. However, finding an appropriate coarse space is not an easy task for 3D complex geometries and developing efficient multilevel methods for real applications is still an active research topic in scientific computing. Some successful 2D examples for multilevel incompressible flow computations can be found in Barker and Cai [3] and Hwang and Cai [18].

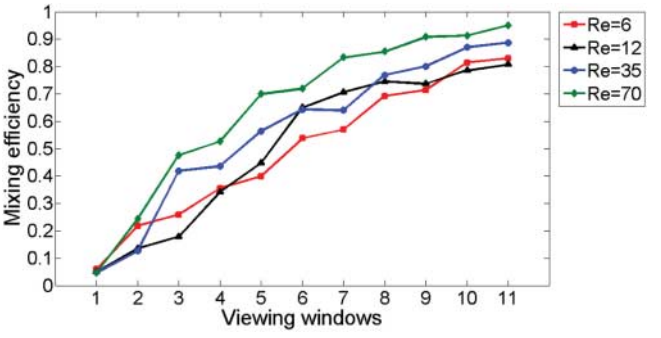

Figure 6. The mixing efficiency at different viewing windows for the cases with different values of the Reynolds number (colour online only).

Table 2. 3D serpentine micromixing. Different subdomain solves: LU and ILU(k), $k = 0, 1, 2, 3$ for varied sizes of overlapping for $Re = 6$ and $Re = 70$. $\Delta t = 0.1$, 10 time steps are performed. $np = 64$.

$Re = 7$		ovlp = 1	ovlp = 2	ovlp = 3	ovlp = 5
LU	ANNI (ANLI)	2.1 (131.3)	2.1 (122.0)	2.1 (119.6)	2.1 (107.0)
	Total time	875.4	1176.1	1447.8	2024.1
ILU(0)	ANNI (ANLI)	4.5 (495.4)	4.6 (495.7)	4.6 (496.0)	4.7 (496.1)
	Total time	931.4	991.3	1150.2	1256.6
ILU(1)	ANNI (ANLI)	2.8 (459.9)	2.8 (460.4)	2.8 (459.4)	2.3 (435.0)
	Total time	675.3	763.5	824.4	764.1
ILU(2)	ANNI (ANLI)	2.2 (359.5)	2.1 (320.2)	2.3 (232.1)	2.1 (186.1)
	Total time	574.1	581.4	553.6	510.7
ILU(3)	ANNI (ANLI)	2.1 (187.0)	2.1 (180.5)	2.1 (173.1)	2.1 (158.0)
	Total time	478.4	558.9	622.7	722.4
$Re = 70$		ovlp = 1	ovlp = 2	ovlp = 3	ovlp = 5
LU	ANNI (ANLI)	3.1 (382.6)	3.1 (176.4)	3.1 (163.7)	3.1 (136.6)
	Total time	2169.6	1981.0	2377.2	3200.9
ILU(0)	ANNI (ANLI)	7.1 (475.3)	7.2 (475.7)	7.2 (475.8)	6.9 (474.8)
	Total time	1266.6	1413.6	1506.5	1635.2
ILU(1)	ANNI (ANLI)	5.1 (461.2)	4.9 (461.4)	5.1 (460.2)	4.5 (454.3)
	Total time	1332.4	1501.7	1540.9	1688.6
ILU(2)	ANNI (ANLI)	4.0 (447.3)	4.0 (440.4)	3.9 (431.8)	3.5 (428.9)
	Total time	1229.6	1419.0	1521.5	1638.5
ILU(3)	ANNI (ANLI)	3.4 (434.5)	3.4 (424.5)	3.2 (404.1)	3.1 (389.7)
	Total time	1405.0	1672.2	1739.5	2040.8

Table 3. Timing results for three microchannel cases for $Re = 6$ and 70. $\Delta t = 0.1$, 10 time steps are performed.

		1D	2D	3D
$Re = 6$	ANNI (ANLI)	2.1 (188.5)	2.1 (195.0)	2.1 (187.0)
	Total time	439.6	372.0	478.4
$Re = 70$	ANNI (ANLI)	5.5 (500)	3.3 (424.5)	3.4 (496.0)
	Total time	1361.6	1053.3	1405.0

For the case of $Re = 6$, although the best choices for the level of ILU fill-ins and the degree of overlaps are slightly different, for example, ovlp = 5 and ILU(2) for 1D, ovlp = 2 and ILU(2) for 2D and ovlp = 1 and ILU(3) for 3D, the number of GMRES iterations and the number of Newton iterations are quite independent of the geometric configurations of microchannels.

Table 4. Parallel efficiency for the 3D serpentine micromixing case with $Re = 26$. The additive Schwarz preconditioner uses ovlp = 1 and ILU(3) as the subdomain solver.

np	ANNI	ANLI	Time (s)	Efficiency (%)
16	2.1	158.6	1587.3	100
32	2.1	163.5	877.3	90
64	2.1	187.0	478.4	83
128	2.2	264.2	370.0	54
256	2.4	394.2	317.5	31

Table 5. Parallel efficiency for the 3D serpentine micromixing case with $Re = 70$. The additive Schwarz preconditioner uses ovlp = 1 and ILU(1) as the subdomain solver.

np	ANNI	ANLI	Time (s)	Efficiency (%)
16	4.7	456.8	3888.3	100
32	4.4	455.7	1989.9	98
64	5.1	461.2	1224.3	79
128	5.0	460.6	734.2	66
256	5.0	464.8	505.8	48

Finally, to evaluate the parallel performance of our fluid solver, we consider the parallel efficiency defined as

$$E_f = \left(\frac{16}{np}\right)\frac{T_{16}}{T_{np}},$$

where T_{16} and T_{np} are the computing time obtained with 16 and *np* processors.

From Tables 4 and 5, we observe that the parallel efficiency reaches at least 50% with up to 128 processors but degrades slightly when 256 processors are used.

5. Conclusions

In this work, we introduced a parallel algorithm for the 3D microfluidic simulation and the corresponding software was developed on top of PETSc and several state-of-the-art open source packages, including Cubit, ParMETIS and ParaView. The core of the approach is based on a fully coupled and fully implicit scalable NKS method. Our studies showed good qualitative agreements between our numerical solutions and the experimental observation shown in [24]. Moreover, we used the three-dimensional serpentine microchannel as a numerical example to demonstrate the applicability of our software to the simulation of microfluidic mixing. Our solver achieved above 54% of parallel efficiency with up to 128 processors on a cluster of PCs. Our research focused on that the mixing of the fluids induced by a passive transport diffusion. The model is based on incompressible Navier–Stokes equations for single-phase flows. It is worth to extend our parallel solver for the case of coupled multi-phase flows with surface tension, which represents broader applications in microfluidic mixing.

Acknowledgements

This research was supported by the Center for Computational Geophysics of NCU, CCG Contribution Number: NCU-CCG101-0015. The first and third author were supported in part by the National Science Council of Taiwan, 96-2115-M-008-007-MY2 and the second author was supported in part by the Department of Energy, DE-FC02-01ER25479, and in part by the US National Science Foundation, CCR-0219190, ACI-0072089 and ACI-0305666.

References

[1] D. Ahmed, X. Mao, J. Shi, B. Juluri, and T. Huang, *A millisecond micromixer via single-bubble-based acoustic streaming*, Lab Chip 9 (2009), pp. 2738–2741.

[2] S. Balay, K. Buschelman, W. Gropp, D. Kaushik, M. Knepley, L.C. McInnes, B. Smith, and H. Zhang, *PETSc* Webpage, 2011. Available at http://www.mcs.anl.gov/petsc.

[3] A. Barker and X.-C. Cai, *Scalable parallel methods for monolithic coupling in fluid–structure interaction with application to blood flow modeling*, J. Comput. Phys. 229 (2010), pp. 642–659.

[4] H. Bau, J. Zhong, and M. Yi, *A minute magneto hydrodynamic (MHD) mixer*, Sensor Actuator B Chem. 79 (2001), pp. 207–215.

[5] S. Behara and S. Mittal, *Parallel finite element computation of incompressible flows*, Parallel Comput. 35 (2009), pp. 195–212.

[6] P. Brown, D. Shumaker, and C. Woodward, *Fully implicit solution of large-scale non-equilibrium radiation diffusion with high order time integration*, J. Comput. Phys. 204 (2005), pp. 760–783.

[7] X.-C. Cai, W. Gropp, D. Keyes, R. Melvin, and D. Young, *Parallel Newton–Krylov–Schwarz algorithms for the transonic full potential equation*, SIAM J. Sci. Comput. 19 (1998), pp. 246–265.

[8] R. Choudhary, T. Bhakat, R. Singh, A. Ghubade, S. Mandal, A. Ghosh, A. Rammohan, A. Sharma, and S. Bhattacharya, *Bilayer staggered herringbone micro-mixers with symmetric and asymmetric geometries*, Microfluid. Nanofluid. 10 (2011), pp. 271–286.

[9] Cubit. Online CUBIT user's manual, 2011. Available at http://cubit.sandia.gov/documentation.html.

[10] J. Dennis and R. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, SIAM, Philadelphia, 1996.

[11] W. Dettmer and D. Perić, *An analysis of the time integration algorithms for the finite element solutions of incompressible Navier–Stokes equations based on a stabilised formulation*, Comput. Methods Appl. Mech. Eng. 192 (2003), pp. 1177–1226.

[12] Y. Du, Z. Zhang, C. Yim, M. Lin, and X. Cao, *A simplified design of the staggered herringbone micromixer for practical applications*, Biomicrofluidics 4 (2010), Article ID 024105, 13 pp.

[13] L. Franca and S. Frey, *Stabilized finite element methods. II: The incompressible Navier–Stokes equations*, Comput. Methods Appl. Mech. Eng. 99 (1992), pp. 209–233.

[14] L. Franca, T. Hughes, and S. Frey, *Stabilized finite element methods: I. Application to the advective-diffusive model*, Comput. Methods Appl. Mech. Eng. 95 (1992), pp. 253–276.

[15] T. Glatzel, C. Litterst, C. Cupelli, T. Lindemann, C. Moosmann, R. Niekrawietz, W. Streule, R. Zengerle, and P. Koltay, *Computational fluid dynamics (CFD) software tools for microfluidic applications – a case study*, Comput. Fluid. 37 (2008), pp. 218–235.

[16] V. Hessel, H. Löwe, and F. Schönfeld, *Micromixers – a review on passive and active mixing principles*, Chem. Eng. Sci. 60 (2005), pp. 2479–2501.

[17] S. Hossain, A. Husain, and K. Kim, *Shape optimization of a micromixer with staggered-herringbone grooves patterned on opposite walls*, Chem. Eng. J. 162 (2010), pp. 730–737.

[18] F.-N. Hwang and X.-C. Cai, *Parallel fully coupled Schwarz preconditioners for saddle point problems*, Electron. Trans. Numer. Anal. 22 (2006), pp. 146–162.

[19] F.-N. Hwang, C.-Y. Wu, and X.-C. Cai, *Numerical simulation of three-dimensional blood flows using domain decomposition method on parallel computer*, J. Chin. Soc. Mech. Eng. 31 (2010), pp. 199–208.

[20] T. Kang, M. Singh, T. Kwon, and P. Anderson, *Chaotic mixing using periodic and aperiodic sequences of mixing protocols in a micromixer*, Microfluid. Nanofluid. 4 (2008), pp. 589–599.

[21] G. Karniadakis, A. Beşkök, and N. Aluru, *Microflows and Nanoflows: Fundamentals and Simulation*, Springer Verlag, Berlin, 2005.

[22] G. Karypis, *METIS* homepage, 2011. Available at http://cubit.sandia.gov/documentation.html.

[23] L. Li, L. Lee, J. Castro, and A. Yi, *Improving mixing efficiency of a polymer micromixer by use of a plastic shim divider*, J. Micromech. Microeng. 20 (2010), Article ID 035012, 9 pp.

[24] R. Liu, M. Stremler, K. Sharp, M. Olsen, J. Santiago, R. Adrian, H. Aref, and D. Beebe, *Passive mixing in a three-dimensional serpentine microchannel*, IEEE ASME J. Microelectromech. Syst. 9 (2002), pp. 190–197.

[25] S. Müller, I. Mezi, J. Walther, and P. Koumoutsakos, *Transverse momentum micromixer optimization with evolution strategies*, Comput. Fluid. 33 (2004), pp. 521–531.

[26] M. Murillo and X.-C. Cai, *A fully implicit parallel algorithm for simulating the non-linear electrical activity of the heart*, Numer. Linear Algebra Appl. 11 (2004), pp. 261–277.

[27] N.-T. Nguyen and Z. Wu, *Micromixers – a review*, J. Micromech. Microeng. 15 (2005), pp. R1–R16.

[28] S. Ovtchinnikov, F. Dobrian, X.-C. Cai, and D. Keyes, *Additive Schwarz-based fully coupled implicit methods for resistive Hall magnetohydrodynamic problems*, J. Comput. Phys. 225 (2007), pp. 919–1936.

[29] ParaView. *ParaView* homepage, 2011. Available at http://www.paraview.org.

[30] J. Park, K. Seo, and T. Kwon, *A chaotic micromixer using obstruction-pairs*, J. Micromech. Microeng. 20 (2010), Article ID 015023, 11 pp.

[31] Y. Saad, *Iterative Methods for Sparse Linear Systems*, SIAM, Philadelphia, 2003.

[32] Y. Saad and M. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. 7 (1986), pp. 856–869.

[33] T.-R. Shih and C.-K. Chung, *A high-efficiency planar micromixer with convection and diffusion mixing over a wide Reynolds number range*, Microfluid. Nanofluid. 5 (2008), pp. 175–183.
[34] B. Smith, P. Bjørstad, and W. Gropp, *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*, Cambridge University Press, Cambridge, 1996.
[35] A. Stroock, S. Dertinger, A. Ajdari, I. Mezic, H. Stone, and G. Whitesides, *Chaotic mixer for microchannels*, Science 295 (2002), pp. 647–651.
[36] A. Tabak and S. Yesilyurt, *Simulation-based analysis of flow due to traveling-plane-wave deformations on elastic thin-film actuators in micropumps*, Microfluid. Nanofluid. 4 (2008), pp. 489–500.
[37] E. Tafti, R. Kumar, and H. Cho, *Effect of laminar velocity profile variation on mixing in microfluidic devices: The sigma micromixer*, Appl. Phys. Lett. 93 (2008), Article ID 143504, 3 pp.
[38] A. Toselli and O. Widlund, *Domain Decomposition Methods – Algorithms and Theory*, Springer, Berlin, 2005.
[39] K.-Y. Tung and J.-T. Yang, *Analysis of a chaotic micromixer by novel methods of particle tracking and FRET*, Microfluid. Nanofluid. 5 (2008), pp. 749–759.
[40] C. Yang, J. Cao, and X.-C. Cai, *A fully implicit domain decomposition algorithm for shallow water equations on the cubed-sphere*, SIAM J. Sci. Comput. 32 (2010), pp. 418–438.
[41] Z. Yang, S. Matsumoto, H. Goto, M. Matsumoto, and R. Maeda, *Ultrasonic micromixer for microfluidic systems*, Sensor Actuator Phys. 93 (2001), pp. 266–272.