Parallel Computational Dynamics
Development and Applications of Parralel Technology
C.A. Lin, A. Ecer, J. Peraux, N. Satofuka and P. Fox, editors

A fictitious domain method with distributed Lagrange multipliers for the numerical simulation of particulate flow and its parallel implementation

T.-W. Pan[a], V. Sarin[b], R. Glowinski[a], A. Sameh[b] and J. Périaux[c]

[a]Department of Mathematics, University of Houston, Houston, TX 77204, USA

[b]Department of Computer Science, Purdue University, West Lafayette, IN 47907, USA

[c]Dassault Aviation, 92214 Saint-Cloud, France

The numerical simulation of particulate flow, a mixtures of *incompressible viscous fluids* and hundreds of *rigid particles*, is computational expensive and parallelism often appears as the only way towards large scale of simulations even we have a fast Navier-Stokes solver. The method we advocate here combines *distributed Lagrange multipliers* based *fictitious domain methods*, which allows the use of *fixed structured finite element grids* on a simple shape auxiliary domain containing the actual one for the fluid flow computations, with *time discretizations* by *operator splitting à la Marchuk-Yanenko* to decouple the various computational difficulties associated to the simulation. This method offers an alternative to the *ALE* methods investigated in [5], [7], and [8] and can be easily parallelized due to the use of uniform structured grids and no need to generate mesh at each time step right after finding the new position of rigid particles. Numerical results of particulate flow obtained on a SGI Origin 2000 are presented.

1 A model problem

For simplicity, we shall consider the motion of a *unique rigid body* $B(t)$ surrounded by a *Newtonian incompressible viscous fluid* (see Figure 1) in a region $\Omega \subset I\!R^2$. But there is no basic difficulty to generalize the following considerations to 3-dimensional particulate flow. The *fluid flow* is modeled by the following *Navier-Stokes equations* (with obvious and/or classical notation):

$$\rho_f \left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right] = \rho_f \mathbf{g} + \nabla\cdot\boldsymbol{\sigma} \text{ in } \Omega\backslash\overline{B(t)}, \tag{1.1}$$

$$\nabla\cdot\mathbf{u} = 0 \text{ in } \Omega\backslash\overline{B(t)}, \tag{1.2}$$

$$\mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}), \mathbf{x} \in \Omega\backslash\overline{B(0)}, \text{ with } \nabla\cdot\mathbf{u}_0 = 0, \tag{1.3}$$

$$\mathbf{u} = \mathbf{g}_0 \text{ on } \Gamma \tag{1.4}$$

with the *stress tensor* $\boldsymbol{\sigma} = -p\mathbf{I} + \nu_f(\nabla\mathbf{u} + \nabla\mathbf{u}^t)$ for *Newtonian fluids* and $\overline{B(t)} = \partial B(t) \cup B(t)$ where $\partial B(t)$ is the boundary of the rigid body $B(t)$. Assuming that a *no-slip condition* holds on $\partial B(t)$, the *rigid body motion* of $B(t)$, combined with the *incompressibility condition* (1.2), implies that $\int_\Gamma \mathbf{g}_0 \cdot \mathbf{n} d\Gamma = 0$. Denoting by $\mathbf{V}$ (resp., ω) the

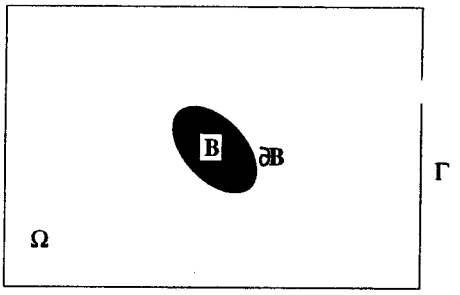

Figure 1: The rigid body B and the flow region $\Omega\backslash\overline{B}$

velocity of the center of mass G (resp., the *angular velocity*) of the rigid body B, we have for the motion of B the following *Newton's equations*:

$$M\dot{\mathbf{V}} = \mathbf{F} + M\mathbf{g}, \quad I\dot{\omega} = T, \quad \dot{G} = \mathbf{V}, \tag{1.5}$$

with the *force* $\mathbf{F}$ and *torque* T, resulting from the *fluid-particle interaction*, given by

$$\mathbf{F} = \int_{\partial B(t)} \boldsymbol{\sigma}\mathbf{n} d\gamma, \quad T = \int_{\partial B(t)} (\vec{G\mathbf{x}} \times \boldsymbol{\sigma}\mathbf{n}) \cdot \mathbf{e}_3 d\gamma, \tag{1.6}$$

where, in (1.6), $\mathbf{e}_3 = \{0, 0, 1\}$ if we assume that Ω is contained in the plane x_1Ox_2. The no-slip boundary condition mentioned above implies that on $\partial B(t)$ we have

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{V}(t) + \omega(t)\mathbf{e}_3 \times \vec{G\mathbf{x}}, \ \forall \mathbf{x} \in \partial B(t). \tag{1.7}$$

Of course, I is the *moment of inertia* of B, with respect to G.

2 A fictitious domain formulation

Through fictitious domain methods, the actual problems with complex geometries can be solved on a simple shape auxiliary domain containing the actual one. Therefore we can use *structured uniform finite element grids* and then *parallel fast solvers* designed on the structured uniform grids, and avoid mesh generations at each time step right after finding the new position of rigid particles. The *fictitious domain method* described below, offers an alternative to the *ALE* methods investigated in [5], [7], and [8]. The basic idea is quite simple and can be summarized as follows:

(i) *Fill each particle with the surrounding fluid.*

(ii) *Impose a rigid body motion to the fluid inside each particle.*

(iii) *Relax the rigid body motion inside each particle by using a distributed Lagrange multiplier defined over the space region occupied by the particles.*

In the following, we shall assume that the particle B is made of an homogeneous material of density ρ_s. Starting from the variational formulation of (1.1)-(1.7) and following steps (i) to (iii) lead to the following *generalized variational problem* (for detail, see [4]) on Ω, where $\boldsymbol{\lambda}(t)$ is the *distributed Lagrange multiplier* forcing at time t rigid body motion for the fluid "filling" body B:

For a.e. $t > 0$, *find* $\{\mathbf{U}(t), P(t), \boldsymbol{\lambda}(t), \mathbf{V}(t), G(t), \omega(t)\}$ *such that*
$\mathbf{U}(t) \in W_{\mathbf{g}_0(t)} = \{\mathbf{v} | \mathbf{v} \in H^1(\Omega)^2,\ \mathbf{v} = \mathbf{g}_0(t)\ on\ \Gamma\},\ P(t) \in L^2_0(\Omega) = \{q | q \in L^2(\Omega),$
$\int_\Omega q d\mathbf{x} = 0\},\ \boldsymbol{\lambda}(t) \in \Lambda(t) = H^1(B(t))^2,\ \mathbf{V}(t) \in I\!R^2,\ G(t) \in I\!R^2,\ \omega(t) \in R$

and

$$\begin{cases} \rho_f \int_\Omega \frac{\partial \mathbf{u}}{\partial t} \cdot \mathbf{v} d\mathbf{x} + \rho_f \int_\Omega (\mathbf{U} \cdot \nabla)\mathbf{U} \cdot \mathbf{v} d\mathbf{x} - \int_\Omega P \nabla \cdot \mathbf{v} d\mathbf{x} + 2\nu_f \int_\Omega \mathbf{D}(\mathbf{U}) : \mathbf{D}(\mathbf{v}) d\mathbf{x} \\ \quad + (1 - \rho_f/\rho_s) M(\dot{\mathbf{V}} - \mathbf{g}) \cdot \mathbf{Y} + (1 - \rho_f/\rho_s) I \dot{\omega} \theta - < \boldsymbol{\lambda}, \mathbf{v} - \mathbf{Y} - \theta \mathbf{e}_3 \times \vec{G\mathbf{x}} >_{B(t)} \\ = \rho_f \int_\Omega \mathbf{g} \cdot \mathbf{v} d\mathbf{x}, \forall \mathbf{v} \in H^1_0(\Omega)^2,\ \forall \{\mathbf{Y}, \theta\} \in I\!R^3, \end{cases} \tag{2.1}$$

$$\int_\Omega q \nabla \cdot \mathbf{U} d\mathbf{x} = 0,\ \forall q \in L^2(\Omega), \tag{2.2}$$

$$< \boldsymbol{\mu}, \mathbf{U} - \mathbf{V} - \omega \mathbf{e}_3 \times \vec{G\mathbf{x}} >_{B(t)} = 0,\ \forall \boldsymbol{\mu} \in \Lambda(t), \tag{2.3}$$

$$\mathbf{U}(\mathbf{x}, 0) = \mathbf{U}_0(\mathbf{x}), \mathbf{x} \in \Omega,\ (with\ \nabla \cdot \mathbf{U}_0 = 0\ and\ \mathbf{U}_0|_{\Omega \setminus \overline{B(0)}} = \mathbf{u}_0), \tag{2.4}$$

$$\mathbf{U} = \mathbf{g}_0\ on\ \Gamma,\ \mathbf{V}(0) = \mathbf{V}_0,\ \omega(0) = \omega_0, G(0) = G_0 \tag{2.5}$$

where $D(\mathbf{v}) = (\nabla \mathbf{v} + (\nabla \mathbf{v})^T)/2$, $G(t) = G_0 + \int_0^t \mathbf{V}(s)\, ds$, $\mathbf{V}_0$ (resp., ω_0) is the initial velocity (resp., initial angular velocity) of the particle $B(t)$ and G_0 is the initial center position of the particle. If (2.1)-(2.5) holds, it can be easily shown that $\mathbf{U}(t)|_{\Omega \setminus \overline{B(t)}} = \mathbf{u}(t)$, $P(t)|_{\Omega \setminus \overline{B(t)}} = p(t)$, where $\{\mathbf{u}(t), p(t)\}$ completed by $\{\mathbf{V}(t), \omega(t)\}$ is a solution of the variational problem of (1.1)-(1.7).

Remark 4.1: In the case of *Dirichlet boundary conditions* on Γ, and taking the *incompressibility condition* $\nabla \cdot \mathbf{U} = 0$ into account, we can easily show that

$$2\nu_f \int_\Omega \mathbf{D}(\mathbf{U}) : \mathbf{D}(\mathbf{v}) d\mathbf{x} = \nu_f \int_\Omega \nabla \mathbf{U} : \nabla \mathbf{v} d\mathbf{x},\ \forall \mathbf{v} \in W_0, \tag{2.6}$$

which, from a *computational* point of view, leads to a substantial simplification in (2.1)-(2.5). □

3 Approximation

With h a *space discretization step* we introduce a finite element triangulation $\mathcal{T}_h$ of $\overline{\Omega}$ and then $\mathcal{T}_{2h}$ a triangulation twice coarser (in practice we should construct $\mathcal{T}_{2h}$ first and then $\mathcal{T}_h$ by joining the midpoints of the edges of $\mathcal{T}_{2h}$, dividing thus each triangle of $\mathcal{T}_{2h}$ into 4 similar subtriangles, as shown in Figure 2, below).

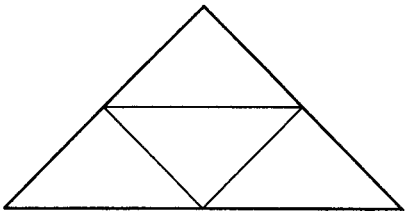

Figure 2. Subdivision of a triangle of $\mathcal{T}_{2h}$

We define the following finite dimensional spaces which approximate $W_{\mathbf{g}_0}(t)$, $(H_0^1(\Omega))^2$, $L^2(\Omega)$, $L_0^2(\Omega)$, respectively:

$$W_{\mathbf{g}_{0h}}(t) = \{\mathbf{v}_h | \mathbf{v}_h \in (C^0(\overline{\Omega}))^2,\ \mathbf{v}_h|_T \in P_1 \times P_1,\ \forall T \in \mathcal{T}_h,\ \mathbf{v}_h|_\Gamma = \mathbf{g}_{0h}(t)\}, \tag{3.1}$$

$$W_{0h} = \{\mathbf{v}_h | \mathbf{v}_h \in (C^0(\overline{\Omega}))^2,\ \mathbf{v}_h|_T \in P_1 \times P_1,\ \forall T \in \mathcal{T}_h,\ \mathbf{v}_h|_\Gamma = \mathbf{0}\}, \tag{3.2}$$

$$L_h^2 = \{q_h | q_h \in C^0(\overline{\Omega}),\ q_h|_T \in P_1,\ \forall T \in \mathcal{T}_{2h}\},\ L_{0h}^2 = \{q_h | q_h \in L_h^2,\ \int_\Omega q_h d\mathbf{x} = 0\}; \tag{3.3}$$

in (3.1)-(3.3), $\mathbf{g}_{0h}(t)$ is an approximation of $\mathbf{g}_0(t)$ verifying $\int_\Gamma \mathbf{g}_{0h}(t) \cdot \mathbf{n} d\Gamma = 0$ and P_1 is the space of the polynomials in two variables of degree ≤ 1.

A finite dimensional space approximating $\Lambda(t)$ is defined as follows: let $\{\mathbf{x}_i\}_{i=1}^{N_B}$ be a set of points from $\overline{B(t)}$ which cover $\overline{B(t)}$ (uniformly, for example); we define then

$$\Lambda_h(t) = \{\boldsymbol{\mu}_h | \boldsymbol{\mu}_h = \sum_{i=1}^{N_B} \boldsymbol{\mu}_i \delta(\mathbf{x} - \mathbf{x}_i),\ \boldsymbol{\mu}_i \in \mathbb{R}^2,\ \forall i = 1, ...N_B\}, \tag{3.4}$$

where $\delta(\cdot)$ is the *Dirac measure* at $\mathbf{x} = \mathbf{0}$. Then the scalar product, $< \cdot, \cdot >_{B(t)}$, is defined by

$$< \boldsymbol{\mu}_h, \mathbf{v}_h >_{B(t)} = \sum_{i=1}^{N_B} \boldsymbol{\mu}_i \cdot \mathbf{v}_h(\mathbf{x}_i),\ \forall \boldsymbol{\mu}_h \in \Lambda_h(t),\ \forall \mathbf{v}_h \in W_{\mathbf{g}_{0h}}(t) \text{ or } W_{0h}. \tag{3.5}$$

The approach, based on (3.4), (3.5), makes little sense for the continuous problem, but is meaningful for the discrete problem; it amounts to forcing the rigid body motion of $B(t)$ via a *collocation method.* A similar technique has been used to enforce Dirichlet boundary conditions by F. Bertrand, P.A. Tanguy and F. Thibault (ref. [1]).

Using above finite dimensional spaces and operator splitting *à la Marchuk-Yanenko* discussed in [6], we obtain the following discrete scheme:

$$\mathbf{U}^0 = \mathbf{U}_{0h}, \mathbf{V}^0, \omega^0, G^0 \text{ are given;} \tag{3.6}$$

for $n \geq 0$, *assuming that* $\mathbf{U}^n, \mathbf{V}^n, \omega^n, G^n$ *are known, solve*

$$\begin{cases} \rho_f \int_\Omega \dfrac{\mathbf{U}^{n+1/3} - \mathbf{U}^n}{\Delta t} \cdot \mathbf{v} d\mathbf{x} - \int_\Omega P^{n+1/3} \nabla \cdot \mathbf{v} d\mathbf{x} = 0,\ \forall \mathbf{v} \in W_{0h}, \\ \int_\Omega q \nabla \cdot \mathbf{U}^{n+1/3} d\mathbf{x} = 0,\ \forall q \in L_h^2;\ \{\mathbf{U}^{n+1/3}, P^{n+1/3}\} \in W_{\mathbf{g}_{0h}}^{n+1} \times L_{0h}^2. \end{cases} \tag{3.7}$$

Next, compute $\mathbf{U}^{n+2/3}, \mathbf{V}^{n+2/3}, G^{n+2/3}$ *via the solution of*

$$\begin{cases} \rho_f \int_\Omega \dfrac{\mathbf{U}^{n+2/3} - \mathbf{U}^{n+1/3}}{\Delta t} \cdot \mathbf{v} d\mathbf{x} + \nu_f \int_\Omega \nabla \mathbf{U}^{n+2/3} \cdot \nabla \mathbf{v} d\mathbf{x} + \\ \rho_f \int_\Omega (\mathbf{U}^{n+1/3} \cdot \nabla) \mathbf{U}^{n+2/3} \cdot \mathbf{v} d\mathbf{x} = \rho_f \int_\Omega \mathbf{g} \cdot \mathbf{v} d\mathbf{x}, \forall \mathbf{v} \in W_{0h};\ \mathbf{U}^{n+1/3} \in W^{n+1}_{\mathbf{g}_{0h}}, \end{cases} \tag{3.8}$$

$$\mathbf{V}^{n+2/3} = \mathbf{V}^n + \mathbf{g}\Delta t,\ G^{n+2/3} = G^n + (\mathbf{V}^n + \mathbf{V}^{n+2/3})\Delta t/2. \tag{3.9}$$

Finally, compute $\mathbf{U}^{n+1}, \boldsymbol{\lambda}^{n+1}, \mathbf{V}^{n+1}, \omega^{n+1}, G^{n+1}$ *via the solution of*

$$\begin{cases} \rho_f \int_\Omega \dfrac{\mathbf{U}^{n+1} - \mathbf{U}^{n+2/3}}{\Delta t} \cdot \mathbf{v} d\mathbf{x} + (1 - \rho_f/\rho_s) I \dfrac{\omega^{n+1} - \omega^n}{\Delta t} \theta \\ \quad + (1 - \rho_f/\rho_s) M \dfrac{\mathbf{V}^{n+1} - \mathbf{V}^{n+2/3}}{\Delta t} \cdot \mathbf{Y} \\ = < \boldsymbol{\lambda}^{n+1}, \mathbf{v} - \mathbf{Y} - \theta \mathbf{e}_3 \times G^{n+2/3} \overrightarrow{\mathbf{x}} >_{B^{n+2/3}},\ \forall \mathbf{v} \in W_{0h}, \{\mathbf{Y}, \theta\} \in I\!R^3, \\ < \boldsymbol{\mu}, \mathbf{U}^{n+1} - \mathbf{V}^{n+1} - \omega^{n+1} \mathbf{e}_3 \times \overrightarrow{G^{n+2/3} \mathbf{x}} >_{B^{n+2/3}} = 0,\ \forall \boldsymbol{\mu} \in \Lambda_h^{n+2/3}; \\ \mathbf{U}^{n+1} \in W^{n+1}_{\mathbf{g}_{0h}}, \boldsymbol{\lambda}^{n+1} \in \Lambda_h^{n+2/3}, \mathbf{V}^{n+1} \in I\!R^2,\ \omega^{n+1} \in I\!R, \end{cases} \tag{3.10}$$

and

$$G^{n+1} = G^n + (V^n + V^{n+1})\Delta t/2. \tag{3.11}$$

In (3.6)-(3.11) we have $W^s_{\mathbf{g}_{0h}} = W_{\mathbf{g}_0 h}(s\Delta t)$, $\Lambda^s_h = \Lambda_h(s\Delta t)$, and $B^s = B(s\Delta t)$.

4 Parellelization

Solving problem (3.7) is *equivalent* to computing the $L^2(\Omega)$-*projection* of $\mathbf{U}^n$ on the (affine) subset of the functions $\mathbf{v} \in W^{n+1}_{\mathbf{g}_{0h}}$ such that $\int_\Omega q \boldsymbol{\nabla} \cdot \mathbf{v}\, d\mathbf{x} = 0,\ \forall q \in L^2_h$, and that $P^{n+1/3}$ is the corresponding Lagrange multiplier in L^2_{0h}. The pair $\{\mathbf{U}^{n+1/3}, P^{n+1/3}\}$ is *unique*. To compute $\{\mathbf{U}^{n+1/3}, P^{n+1/3}\}$, we can use an *Uzawa/conjugate gradient* algorithm operating in L^2_{0h} equipped with the scalar product

$$\{q, q'\} \to \int_\Omega \nabla q \cdot \nabla q'\, d\mathbf{x}.$$

We obtained then an algorithm preconditioned by the discrete equivalent of $-\Delta$ for the homogeneous Neumann boundary condition; such an algorithm is described in [10]. In this article the solution of the Laplacian for the pressure mesh that functions as preconditioner mentioned in the above algorithm is solved by a parallel multilevel Poisson solver, developed by Sarin and Sameh [9]

Problem (3.8), the advection-diffusion problem, is solved by a least-squares/conjugate-gradient algorithm [2] with two or three iterations. The arisen linear system has been solved by the Jacobi iterative method which is easy to be parallelized.

Finally, problem (3.10) has the following - classical - *saddle-point* structure

$$\begin{cases} Ax + By = b, \\ B^t x = c, \end{cases} \tag{4.1}$$

with A a *symmetric* and *positive definite matrix*. Problem (3.10) can also be solved by an *Uzawa/conjugate gradient* algorithm (in which there is no need to solve any elliptic problems); such an algorithm is described in [3] and [4].

Due to the fact that distributed Lagrange multiplier method uses *uniform meshes* on a rectangular domain and relies on matrix-free operations on the velocity and pressure unkowns, this approach simplifies the distribution of data on parallel architectures and ensures very good load balance. The basic computational kernels comprising of vector operations such as additions and dot product, and matrix-free matrix-vector products yield nice scalability on distributed shared memory computers such as the SGI Origin 2000.

5 Numerical results

The parallelized code of algorithm (3.6)-(3.11) has been used to simulate the motion of 240 particles in a 2D fluidized bed whose x_1 and x_2 dimensions are 8.182 and 40.64, respectively. The density of the fluid is $\rho_f = 1.0$ and the density of the particles is $\rho_s = 1.14$. The viscosity of the fluid is $\nu_f = 0.01$. The initial condition for the fluid flow is $\mathbf{u} = \mathbf{0}$ and The boundary condition on $\partial\Omega$ of velocity field is

$$\mathbf{u} = \begin{cases} \mathbf{0}, & \text{on two vertical walls,} \\ \begin{pmatrix} 0 \\ U_0(1.0 - e^{-50t}) \end{pmatrix}, & \text{on two horizontal walls} \end{cases}$$

with $U_0 = 0.25$. The diameter d of the particles is 0.635 and the initial position of the particles is shown in Figure 3. Initial velocity and angular velocity of the particles are $\mathbf{V}_i^0 = \mathbf{0}$, ω_i^0=0 for $i = 1, \ldots, 240$. The time step is $\Delta t = 0.001$. The mesh sizes for the velocity field are $h_v = 2.54/80$, and $2.54/160$. The mesh size for pressure is $h_p = 2h_v$.

Table 1

The averaged elapsed time per time step on a SGI Origin 2000

	$h = 1/80$	$h = 1/160$
serial code	61.90 sec.	312.28 sec.
2 processor	44.58 sec.	234.70 sec.
4 processors	23.06 sec.	121.55 sec.
8 processors	12.42 sec.	74.54 sec.
16 processors	7.33 sec.	42.20 sec.

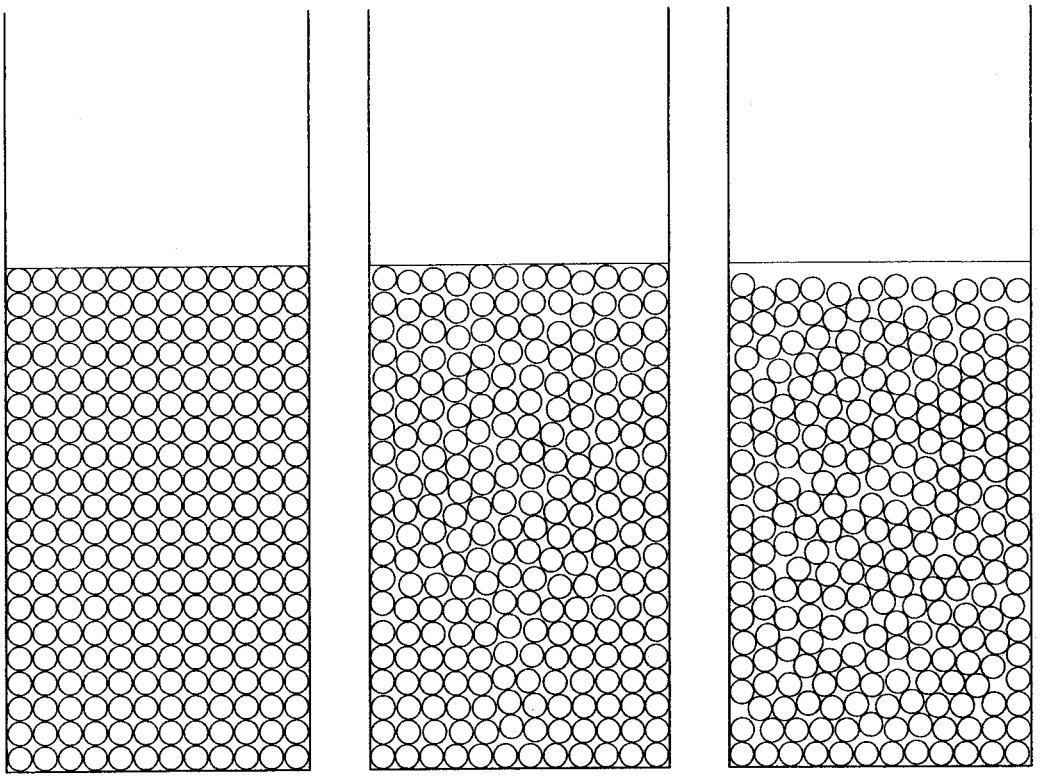

Figure 3. Particle position at $t = 0$, 2, and 4 (left to right) where the solid segment of line above the particles is the initial given bed height.

In this test case, U_0 is not strong enough to fluidize those 240 particles (see Figure 3). In Table 1, we have observed overall algorithmic speed-up of 6.08 (resp., 5.56) on 16 processors compared with the elapsed time on 2 processors when the mesh size is $h_v = 2.54/80$ (resp., $h_v = 2.54/160$). In addition, we also obtain an impressive seven to eight fold increase in speed over the serial implementation.

6 Conclusion

We have presented in this article a distributed Lagrange multiplier based fictitious domain method for the simulation of flow with moving boundaries. Some preliminary experiments of parallelized code have shown the potential of this method for the direct simulation of particulate flow with thousands of particles. In future, our goal is to develop portable 3D code with the ability to simulate large scale problems on a wide variety of architectures.

7 Acknowledgments

We acknowledge the helpful comments and suggestions of E. J. Dean, V. Girault, J. He, Y. Kuznetsov, B. Maury, and G. Rodin and also the support of NCSA concerning the use of a SGI Origin 2000. We acknowledge also the support of the NSF under HPCC Grand Challenge Grant ECS-9527123, NSF (Grants DMS 8822522, DMS 9112847, DMS 9217374), Dassault Aviation, DRET (Grant 89424), DARPA (Contracts AFOSR F49620-89-C-0125,

AFOSR-90-0334), the Texas Board of Higher Education (Grants 003652156ARP and 003652146ATP) and the University of Houston (PEER grant 1-27682).

References

[1] F. Bertrand, P.A. Tanguy, F. Thibault, A three-dimensional fictitious domain method for incompressible fluid flow problem, Int. J. Num. Meth. Fluids, 25, (1997), 719-736.

[2] R. Glowinski, Numerical methods for nonlinear variational problems, Springer-Verlag, New York, 1984.

[3] R. Glowinski, T.I. Hesla, D.D. Joseph, T.W. Pan and J. Périaux, Distributed Lagrange multiplier methods for particulate flows, in Computational Science for the 21st Century, M.O. Bristeau, G. Etgen, W. Fitzgibbon, J.L. Lions, J. Périaux and M.F. Wheeler, eds., J. Wiley, Chichester, 1997, 270-279.

[4] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph and J. Périaux, A distributed Lagrange multiplier/fictitious domain method for particulate flows, Internat. J. of Multiphase Flow (to appear).

[5] H.H. Hu, Direct simulation of flows of solid-liquid mixtures, Internat. J. Multiphase Flow, 22 (1996), 335-352.

[6] G. I. Marchuk, Splitting and alternate direction methods, in Handbook of Numerical Analysis, Vol. I, P.G. Ciarlet and J.L. Lions eds., North-Holland, Amsterdam, 1990, 197-462.

[7] B. Maury and R. Glowinski, Fluid particle flow: a symmetric formulation, C.R. Acad. Sci., Paris, t. 324, Série I (1997), 1079-1084.

[8] O. Pironneau, J. Liou and T. Tezduyar, Characteristic-Galerkin and Galerkin least squares space-time formulations for advection-diffusion equations with time-dependent domains, Comp. Meth. Appl. Mech. Eng., 16 (1992), 117-141.

[9] V. Sarin, A. Sameh, An efficient iterative method for the generalized Stokes problem, SIAM J. Sci. Comput., 19 (1998), 206-226. 2, 335-352.

[10] S. Turek, A comparative study of time-stepping techniques for the incompressible Navier-Stokes equations: from fully implicit non-linear schemes to semi-implicit projection methods, Int. J. Num. Meth. Fluids, 22 (1996), 987-1011.