A Multi-Scale Method for Dynamics Simulation in Continuum Solvent Models I: Finite-Difference Algorithm for Navier-Stokes Equation

Li Xiao,[1,2] Qin Cai,[1,2] Zhilin Li,[5] Hongkai Zhao,[3] and Ray Luo[1,2,4]

1. Departments of Biomedical Engineering, 2. Molecular Biology and Biochemistry, 3. Mathematics , and 4. Chemical Engineering and Materials Science,

University of California, Irvine, CA 92697

5. Department of Mathematics, North Carolina State University, Raleigh, NC 27695

Abstract

A multi-scale framework is proposed for more realistic molecular dynamics simulations in continuum solvent models by coupling a molecular mechanics treatment of solute with a fluid mechanics treatment of solvent. This article reports our initial efforts to formulate the physical concepts necessary for coupling the two mechanics and develop a 3D numerical algorithm to simulate the solvent fluid via the Navier-Stokes equation. The numerical algorithm was validated with multiple test cases. The validation shows that the algorithm is effective and stable, with observed accuracy consistent with our design.

Introduction

To model a biomolecular system, there are different levels of approximation. The most accurate approach is the high level quantum description [1], which explicitly accounts for the electronic structures in the molecules. However the computational cost of this approach is high for large and complex biomolecular systems. The most efficient approach is the coarse-grained description [2], which averages atomic structures and properties. The intermediate approach is the atom-based description, which preserves the atomic details in molecular structures and models the effects of electronic structure through a few empirical energy terms [3]. On the other hand, continuum description is widely used for solvents to capture average solvation properties [4].

Since each level of description has its own advantages and limitations, multi-resolution or multi-scale modeling has also been used to study complex molecular systems [2]. In this type of strategies, researchers aim to calculate material properties or system behavior at one resolution/level using information or models from different resolutions/levels. At each level a particular approach is used for the description of a system of interest [2]. For example, the QM/MM (quantum mechanics/molecular mechanics) model is widely used [5, 6].

Another popular multi-scale model is the MM/CM (molecular mechanics/continuum mechanics) model [4]. Langevin and Brownian dynamics are often used to model the motion of particles in the MM/CM model [7]. The development of Langevin/Brownian dynamics dates back to Einstein's original work, in which the collective motion of particles in water is modeled by the diffusion theory [8]

$$\frac{\partial \rho}{\partial t} = D \frac{\partial^2 \rho}{\partial x^2}. \tag{1}$$

Here D is the mass diffusivity, ρ is the particle density. The average distance after a period of time t is given by

$$\bar{x}^2 = 2Dt. \tag{2}$$

and the mass diffusivity is

$$D = \frac{k_B T}{m\gamma}. \tag{3}$$

Here k_B is the Boltzmann constant, T is the temperature, m is the particle mass, γ is the particle viscosity coefficient. Eqn (3) indicates that the viscous force is proportional to the particle velocity. Subsequently Langevin proposed a random force term to study the Brownian motion of a single particle in water [9], which can be summarized as the Langevin equation

$$m\frac{d^2x}{d^2t} = -\nabla U(x) - \gamma\frac{dx}{dt} + \sqrt{2\gamma k_B T m}R(t), \tag{4}$$

where the first term on the right-hand side of the equation is the force due to the particle interaction potential, the second term is the viscosity force in proportion to the particle velocity, and the third term is a stochastic noise term representing the effect of the collisions between the particle and the molecules of the solvent fluid. The solution of the Langevin equation is consistent with eqn (2) when there is no interaction potential force. Note that constant γ in the viscous force term is modeled as being proportional to the particle velocity as indicated by Eisterin's previous study [8]. In general, it depends on the geometry of the particle, i.e., shape and size, and also depends on how the particle is exposed to the solvent. Thus, it is an approximation to assume all atoms are the same size and shape, and are all exposed to the solvent for simplicity when the Langevin equation is used in MM/CM simulations. Brownian dynamics is also widely used to describe long-time inter-molecular motions [10-12]. It is a

simplified version of Langevin dynamics where no acceleration of collective motion is assumed to take place [7]

$$0 = -\nabla U(x) - \gamma \frac{dx}{dt} + \sqrt{2\gamma k_B T m} R(t). \tag{5}$$

The constant γ in Langevin/Brownian dynamics algorithms can be obtained via the Stokes equation from a more universal constant of bulk solvent viscosity that is solute independent [13-15]. The Stokes equation is a special case of the Navier-Stokes equation for low Reynold number and steady state incompressible flow. Brune and Kim modeled proteins as rigid bodies immersed in a viscous incompressible fluid [13]. They computed the motility tensors distributed on the protein surface, which can be further incorporated into diffusion coefficient (including both diffusional and rotational). An alternative way adopted by Delatorre et. al. can also provide many interesting physical properties [15]. Hubbard and Douglas [14] presented a simple way to compute the viscosity by extending an analogy between hydrodynamics and electrostatics. They showed that the translational friction coefficient γ in eqn (4) and eqn (5) has a simple relationship with the bulk solvent viscosity η_0 and the solute capacity C as

$$\gamma = 6\pi\eta_0 C. \tag{6}$$

Intrinsic viscosity is another important physical property for solvated biomolecular systems and can be obtained by solving the Stokes equation. Douglas and Garboczi [16] found a simple expression of the intrinsic viscosity in term of polarizability, which can be calculated directly according to molecular geometry. Along a similar direction, Zhou et. al presented an interesting method to calculate the translational friction and intrinsic viscosity, and found a good agreement with experiment [17]. Interestingly, viscosity also plays an important role in the

stability of solute-solvent interface, which is recently demonstrated by a model with cylindrical solute-solvent interface [18].

Limitations of the Langevin/Brownian dynamics algorithms are obvious. One is that the solute molecules are assumed to be rigid bodies in most studies, while the conformations of biomolecules are quite flexible. Another one is that all atoms of biomolecules are assumed to be 100% surrounded by the solvent, i.e. the solvent dragging force and the stochastic term are applied to all atoms, which is not a good approximation since buried atoms only interact with solute atoms. Besides, all the particles are unrealistically assumed to be of the same size. These approximations often lead to incorrect dynamics or sampling of biomolecular conformations. Finally an important feature lacking in many Langevin and Brownian dynamics studies is the hydrodynamic effect, a must in the studies of many biochemical processes that are diffusion-dorminant. Indeed the hydrodynamic effect plays an important role in determining the rate constants of diffusion-controlled reactions, and it cannot be derived from an intermolecular potential [19]. To date only a few approximated strategies are available to estimate the hydrodynamic effect [20, 21]. These models all rely on the ground-breaking algorithm proposed by Ermak and McCammon [10].

This paper reports our exploration of a more physical approach in simulating dynamics of biomolecular systems in continuum solvent models. Specifically, the Navier-Stokes equation or the Stokes equation is used to simulate the continuum solvent and is coupled to the Newtonian equation for the particle-based biomolecular solute. In this first attempt, we formulated the necessary physical model to couple molecular mechanics and fluid mechanics. We also developed and validated a new three-dimensional numerical algorithm for the Navier-Stokes equation before our next step to couple it to the Newtonian equation of a molecular solute.

Theoretical Model

1. Physical model

We proceed by first defining a Hamiltonian for the entire system. Its degrees of freedom are atomic positions ($\mathbf{x}$) and their velocities ($\mathbf{v}$) for the solute molecular dynamics (MD) region; and fluid element displacements ($\mathbf{y}$) and their velocities ($\mathbf{u}$) for the solvent fluid dynamics (FD) region. For the MD region, all-atom molecular mechanics force fields, either polarizable [22-25] or nonpolarizable [26-28], can be used. Reduced models can certainly be used as well [29, 30]. For the FD region, the solvent fluid can be modeled as an incompressible viscous fluid. Equations of motion for all the relevant variables in the system are obtained by taking the appropriate derivatives of this Hamiltonian. All variables can then be updated in lock step as a function of time with time integration. Thus the entire time history of the system can be obtained numerically given an appropriate set of initial conditions and boundary conditions.

Conceptually, the Hamiltonian can be written as

$$H = H_{MD}(\mathbf{x}, \mathbf{p}_\mathbf{x}) + H_{FD}(\mathbf{y}, \mathbf{p}_\mathbf{y}) + H_{MD/FD}(\mathbf{x}, \mathbf{p}_\mathbf{x}; \mathbf{y}, \mathbf{p}_\mathbf{y}). \tag{7}$$

where $\mathbf{p}_\mathbf{x}$ is the momentum of molecule region and $\mathbf{p}_\mathbf{y}$ is the momentum of fluid region.

H_{MD} is the Hamiltonian of for the MD region that is modeled by molecular mechanics $H_{MD} = U + K$, where U is the force field potential energy and K is the kinetic energy. H_{FD} represents the Hamiltonian for the incompressible solvent fluid. There have been different strategies adopted to formulate a Hamiltonian for the Navier-Stokes equation [31, 32]. Here we follow the formulation from [32] by explicitly considering the constraints of the mass and momentum conservations, and the first and second laws of thermodynamics. Briefly, for an adiabatic system, H_{FD} can be written as

$$H_{FD} = \int \left[\frac{1}{2} \rho \mathbf{u}^2 + U_{\text{int}}(\rho, s) \right] dV, \tag{8}$$

where U_{int} is the internal energy density, ρ is the fluid density, and s is the entropy. The coupling Hamiltonian, $H_{MD/FD} = U_{ele} + U_{vdw} + U_{hse}$, consists of three terms. U_{ele} is the Poisson-Boltzmann electrostatic solvation energy [33-36]. The nonelectrostatic solvation energy is modeled as two components: the van del Waals component U_{vdw} and the hard sphere entropy/cavity component U_{hse} [37-41].

It should be pointed out that non-Hamiltonian systems are also widely used in biomolecular simulations, for which additional terms are used to generate specific statistical ensembles, e.g. the constant temperature and constant pressure ensemble. Among those methods, extended Lagrangian, with artificial coordinates and velocities added into a Lagrangian [42], is widely used. Meanwhile the fluid region governed by the Navier-Stokes equation should also incorporate the thermal fluctuations into its macroscopic hydrodynamics. A common way is to use a stochastic and temperature-dependent noise term in the Navier-Stokes equation as proposed by Landau and Lifshitz [43].

Next we proceed to derive the dynamics equation by first setting $\mathbf{q} = (\mathbf{x}, \mathbf{y})$ as the position vector of the system and $\mathbf{p} = (\mathbf{p}_\mathbf{x}, \mathbf{p}_\mathbf{y})$ as the momentum vector of the system. The familiar Newtonian dynamics equation can be derived from the Hamiltonian equation

$$\dot{p} = -\frac{\partial H}{\partial q}. \tag{9}$$

In the molecule dynamics region, the equation of motion can be expressed symbolically as

$$\dot{\mathbf{p}}_x = -\frac{\partial H_{MD}}{\partial x} - \frac{\partial H_{MD/FD}}{\partial x}. \tag{10}$$

$-\frac{\partial H_{MD}}{\partial x}$ represents the usual force field terms in molecule dynamics simulations. Since U_{hse} does not depend on atomic positions, the coupling force terms that the atoms feel are only those of electrostatics and van del Waals in nature, i.e., $-\frac{\partial H_{MD/FD}}{\partial x} = -\frac{\partial U_{ele}}{\partial x} - \frac{\partial U_{vdw}}{\partial x}$. An interesting property is that the electrostatic forces $-\frac{\partial U_{ele}}{\partial x}$ are simply $q\mathbf{E}$ forces, where q's are "free" charges, i.e. atomic point charges in molecular mechanics models [35]. $-\frac{\partial U_{vdw}}{\partial x}$ are the van del Waals forces from the solvent molecules modeled as continuum [39].

In the fluid dynamics region, as it is modeled as a continuum, the partial derivative is equivalent to the variation at each volume element

$$\dot{\mathbf{p}}_y = -\frac{\delta H_{FD}}{\delta y} - \frac{\delta H_{MD/FD}}{\delta y}. \tag{11}$$

As discussed in Supplemental Material S.1, given a small fluid volume element at position y and volume V, we impose a variation δy on the element. The derivative of the Hamiltonian gives

$$-\frac{\delta H_{FD}}{\delta y} = -\frac{\partial \sigma_{ij}}{\partial y_j} V, \tag{12}$$

where $\sigma_{ij} = -p\delta_{ij} + \mu\left(\frac{\partial(\mathbf{u}\cdot y_i)}{\partial y_j} + \frac{\partial(\mathbf{u}\cdot y_j)}{\partial y_i}\right)$ is the stress tensor of the fluid [44]. The only significant derivative of $H_{MD/FD}$ is the van del Waals force, which is termed as the external force density ($\mathbf{F}$) on the fluid element, i.e.

$$-\frac{\delta H_{MD/FD}}{\delta y} = -\frac{\delta U_{vdw}}{\delta y} = \mathbf{F}V. \tag{13}$$

The change of momentum of the fluid volume element is

$$\dot{\mathbf{p}}_y = \rho V \frac{\partial \mathbf{u}}{\partial t} + \rho V (\mathbf{u} \bullet \nabla)\mathbf{u}. \tag{14}$$

Combining eqns (11) and (12) – (14) and dropping the element volume leads to

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho (\mathbf{u} \bullet \nabla)\mathbf{u} = \frac{\partial \sigma_{ij}}{\partial y_j} + \mathbf{F}. \tag{15}$$

Given the conservation of volume/mass for the given volume element, i.e. $\nabla \bullet \mathbf{u} = 0$, the incompressible Navier-Stokes equation is expressed as

$$\begin{gathered} \rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} \right) = -\nabla p + \mu \Delta \mathbf{u} + \mathbf{F} \\ \nabla \cdot \mathbf{u} = 0, \end{gathered} \tag{16}$$

where p is pressure and μ is viscosity.

As shown in S.2, the interface conditions can also be obtained using the variational principle. Introducing the local coordinate system, which consists of one normal direction ($\mathbf{n}$) and two tangential directions ($\mathbf{t}, \tau$) at a certain point on the interface, i.e.,

$$\begin{cases} \mathbf{n} = \cos\alpha_1 \mathbf{i} + \cos\alpha_2 \mathbf{j} + \cos\alpha_3 \mathbf{k} \\ \mathbf{t} = \cos\beta_1 \mathbf{i} + \cos\beta_2 \mathbf{j} + \cos\beta_3 \mathbf{k} \\ \tau = \cos\gamma_1 \mathbf{i} + \cos\gamma_2 \mathbf{j} + \cos\gamma_3 \mathbf{k}. \end{cases} \tag{17}$$

The interface conditions can be summarized as

$$\begin{gathered} -p + p_g - \gamma\kappa + 2\mu \frac{\partial \mathbf{u}}{\partial \mathbf{n}} \bullet \mathbf{n} + f_{dielec} = 0 \\ \frac{\partial (\mathbf{u} \cdot \mathbf{n})}{\partial \mathbf{t}} + \frac{\partial (\mathbf{u} \cdot \mathbf{t})}{\partial \mathbf{n}} = 0 \qquad \text{on } \partial\Omega \\ \frac{\partial (\mathbf{u} \cdot \mathbf{n})}{\partial \tau} + \frac{\partial (\mathbf{u} \cdot \tau)}{\partial \mathbf{n}} = 0, \end{gathered} \tag{18}$$

where κ is the curvature, γ is surface tension coefficient and p_g is internal pressure. $\mathbf{f}_{dielec}$ is the dielectric boundary electrostatic force, which is in the normal direction of the interface [35].

2. Mathematical model for the fluid dynamic region

In the fluid domain of the physical model, we need to solve a 3D incompressible Navier-Stokes equation (eqn(16)). The fluid domain is typically contained in a rectangular box, with one or multiple interior domain(s) excluded due to the presence of solute molecules or a different phase of fluids, e.g., air bubbles. The conditions at the outer boundary of the rectangular R are

$$\begin{aligned} \mathbf{u}|_{x=x_{\min}} &= 0, \quad & \mathbf{u}|_{x=x_{\max}} &= 0 \\ \left.\frac{\partial \mathbf{u}}{\partial y}\right|_{y=y_{\min}} &= 0, \quad & \left.\frac{\partial \mathbf{u}}{\partial y}\right|_{y=y_{\max}} &= 0 \\ \mathbf{u}|_{z=z_{\min}} &= 0, \quad & \mathbf{u}|_{z=z_{\max}} &= 0 \\ p &= 0, \end{aligned} \qquad \text{on } \partial R \tag{19}$$

which represents a pipe flow in the y direction. Inside the domain R there are the cavities, the union of which is denoted by Ω. For the sake of simplicity in developing the fluid dynamics algorithm, the interior Ω, which is occupied by biomolecules or another phase of fluid, is modeled as ideal gas with pressure $p_g = \lambda / V_g$ in this initial development. Here V_g is the volume of the air bubble, λ is a gas constant. Eqn (18) can also be written in component form as

$$\begin{aligned} u_n \cos\alpha_1 + v_n \cos\alpha_2 + w_n \cos\alpha_3 &= \frac{1}{2\mu}(p - p_g + \gamma\kappa) \\ u_n \cos\beta_1 + v_n \cos\beta_2 + w_n \cos\beta_3 &= -u_t \cos\alpha_1 - v_t \cos\alpha_2 - w_t \cos\alpha_3 \\ u_n \cos\gamma_1 + v_n \cos\gamma_2 + w_n \cos\gamma_3 &= -u_\tau \cos\alpha_1 - v_\tau \cos\alpha_2 - w_\tau \cos\alpha_3 . \end{aligned} \tag{20}$$

where $\mathbf{u} = u\mathbf{i} + v\mathbf{j} + w\mathbf{k}$, and the subscript of the velocity denotes the derivative with respect to the designated direction.

Algorithms

The solution for the above mathematical problem consists of three steps at each time step. The first step is to solve the nonlinear equation for velocity. The backward Euler method instead

of the Crank-Nicholson method was adopted to achieve better stability [45]. To decouple velocity and pressure, we used a semi-implicit method [46], i.e. the nonlinear convection term and pressure term are treated explicitly and the viscosity term is treated implicitly when updating velocity. The second step is to compute pressure, which can be obtained by solving a simple Poisson equation [44]. The third step is to update the interface implicitly via the level set method [47].

1. Solver for velocity and pressure

As discussed in S.4 [45, 46], with the introduction of the backward Euler method and the second order semi-implicit integration scheme, the velocity solver is equivalent to the Helmholtz equation

$$\Delta \mathbf{u}^{k+1} - \frac{3}{2\mu\Delta t}\mathbf{u}^{k+1} = \begin{cases} \frac{1}{\mu}\left(\widetilde{(\mathbf{u}\cdot\nabla\mathbf{u})}^{k+1} + \nabla\tilde{p}^{k+1} + \frac{-4\mathbf{u}^{k}+\mathbf{u}^{k-1}}{2\Delta t} - \mathbf{F}^{k+1}\right), & \textit{outside} \\ \frac{-4\mathbf{u}^{k}+\mathbf{u}^{k-1}}{2\mu\Delta t}, & \textit{inside} \end{cases} \tag{21}$$

where

$$\begin{aligned} \tilde{p}^{k+1} &= 2p^{k} - p^{k-1}, \\ \left(\widetilde{\mathbf{u}\cdot\nabla\mathbf{u}}\right)^{k+1} &= 2(\mathbf{u}\cdot\nabla\mathbf{u})^{k} - (\mathbf{u}\cdot\nabla\mathbf{u})^{k-1}. \end{aligned} \tag{22}$$

As shown in S.5, given the updated velocity, the pressure solver is simplified to a Poisson equation as

$$\nabla^2 p^{k+1} = -\nabla\bullet((\mathbf{u}^{k+1}\bullet\nabla)\mathbf{u}^{k+1}) + \nabla\bullet\mathbf{F}^{k+1}. \tag{23}$$

2. Updating the domain interface

As shown in S.6-S.8, the interface is represented implicitly by a level set function [47]. The level set function is initialized as a signed distance to the interface. Given a moving interface with velocity **u**, the level set function can be updated by the following equation

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \bullet \nabla \phi = 0. \tag{24}$$

To enforce the level set as a signed distance function, the reinitializing algorithm is applied after each update.

3. Augmented strategy

The augmented strategy is adopted to enforce the boundary conditions when solving pressure or velocity. We set up the augmented variables $\mathbf{G}$ as the jump conditions of $\mathbf{u}_{\mathrm{n}}$ and p_n when solving $\mathbf{u}$ and p, respectively [45]. Since the degrees of freedom for $\mathbf{G}$ is much smaller than that of the entire set of grid points after discretization, we followed the strategy by solving $\mathbf{G}$ first, and then plugging it back to update the velocity or pressure. As shown in S.9-S.12, the equations for pressure or velocity together with its boundary conditions lead to two linear equations for pressure or velocity and its augmented variable **G**. Then **G** is solved by combining the two equations, symbolically, as

$$E\mathbf{G} = F, \tag{25}$$

where E and F are large dense matrices in practice, which makes the linear system computationally challenging to solve. S.9 outlines detailed procedure on how to obtain these matrices before its solution.

Results and Discussion

Several test cases were presented to illustrate the proposed physical model and the effectiveness and accuracy of the associated numerical algorithm. The numerical algorithm is first validated with an analytical system of a 3D single spherical bubble in the fluid. Next we show the oscillation dynamics of the single bubble under different solvent viscosity values. Finally we illustrate the more complex dynamics of multiple air bubble merging processes to assess the numerical procedures' robustness in handling processes that involve the molecular association.

1. Accuracy Analysis

Our discretization procedure is based on a first-order interpolation scheme and hence first order accuracy is expected for both velocity and pressure. This was demonstrated with a grid refinement analysis that leads to a quantitative estimation for the order of the accuracy of our numerical method. The analytical solution for the single sphere test case is given in S.13.

Several simulations are performed using different grid sizes. The time step of the integration was set to be in proportional to the grid size as in previous developments [45]. The error is computed as

$$\begin{aligned} \left\| E_u \right\|_\infty &= \max_{outside} \left\{ \sqrt{\left| U_{ijk} - u(x_i, y_j, z_k) \right|^2 + \left| V_{ij}^k - v(x_i, y_j, z_k) \right|^2 + \left| W_{ijk} - w(x_i, y_j, z_k) \right|^2} \right\}, \\ \left\| E_p \right\|_\infty &= \max_{outside} \left\{ \left| P_{ijk} - p(x_i, y_j, z_k) \right| \right\}, \end{aligned} \tag{26}$$

which is the difference between the analytical solution and the numerical solution in the fluid after one time step. Figure I shows the linear regression of the error data in log plot, indicating that the first order accuracy is indeed satisfied: order of 1.30 for velocity and 1.58 for pressure.

2. Single Bubble Dynamics

Since our interests are in realistic biomolecular systems, the radius of the bubble at equilibrium is chosen to be 5nm, about the radii of typical proteins. The default water parameters

are also chosen to be those under room temperature (300K): the viscosity coefficient $\mu = 8.5 \times 10^{-4} Pa \bullet s$, the surface tension coefficient $\gamma = 7.2 \times 10^{-2} N/m$ and the bulk density $\rho = 1.0 \times 10^{3} kg/m^{3}$. The simulation time step is 1ps, which is found to be sufficient for the accuracy and stability for the dynamics simulated.

The simulation was initiated from a non-equilibrium state and would proceed to the equilibrium state within finite time due to the presence of viscosity in the solvent fluid. Here the equilibrium state is reached when the air bubble radius reaches 5nm when the solvent velocity is zero and the solvent pressure is at equilibrium with the environment. Specifically, initial velocity is set as $u_0 = v_0 = w_0 = 0$, initial pressure is computed according to eqn (23), and the initial radius of the air bubble is set as 4nm and 6nm, respectively. The grid size is set as 0.625nm. Two different groups of tests were performed, the first test was conducted with the default bulk water viscosity and the second test was done with an artificially low viscosity at 20% of the default value. The time evolution of the bubble volume is plotted in Figure II.

Figure II shows that the average volume gradually approaches the equilibrium volume in both cases, i.e. the spherical volume with a radius of 5nm. The oscillation dynamics with the default water viscosity is apparently over-damped at the nanometer scale as simulated. Under the low viscosity condition, the fluid flows more freely and the oscillation is apparent before reaching the equilibrium.

Note that there is around 1% error in the final equilibrium volume with respect to the analytical value, due to the numerical error in this simulation. It is also interesting to note that the error of the equilibrium volume is less for the low-viscosity simulation, which is in part due to the fact that the simulation takes less time to reach the equilibrium state.

3. Multiple Bubbles Merging Dynamics

We next studied whether the model is capable of handling the merging of multiple air bubbles due to its relevance to the molecular association. Here we tested two air bubbles and three air bubbles that are placed close to each other, and expect merging of air bubbles once their boundaries touch each other while expanding. Here the simulation box is chosen as 30nm×30nm×30nm and the grid size is set as 1nm. The time integration step remains at 1ps. During our simulations, the air bubbles first expand, and then merge together after touching with each other. The merged large bubble gradually evolves into a sphere and finally the flow becomes equilibrium. The error in the equilibrium volume is about ~2% when comparing with the analytical value based on mass conservation. The snapshots are provided in Figure III and Figure IV for two bubbles and three bubbles, respectively. The trajectory and velocity flow animations are also provided as Supplemental Material.

4. Timing Analysis of Simulations

In our algorithm, at each time step the velocity and pressure are first solved and then the level set function is updated. To solve the velocity or pressure, the augmented variables were introduced to impose the interface conditions. In particular, the linear system for the augmented variables, eqn (25), is first solved, and is then plugged back to update the velocity or pressure. As shown in S.9, solving eqn (25) consists of three steps: compute the right-hand side term F, compute the coefficient matrix E column by column and then solve the linear equation. Since the column-based calculations are highly independent, the matrix is distributed to different processors and solved it in parallel. Also the parallel linear system solver from Scalapack [48] is adopted to solve the linear equation $E\mathbf{G} = F$. All other computations in the numerical algorithm take little time and are implemented in the serial mode. Table I lists the average time per step of the three air bubbles merging dynamics as the system approaches the equilibrium state. The data

shows that computing coefficient matrix and solving the linear equation cost most of the time in the single-thread run and their computation time is greatly reduced when the program is run in parallel with 32 threads. The scaling of the parallel program is illustrated in Figure V, which plots the speedup versus the number of processors tested. The analysis shows that the parallel efficiency is quite reasonable up to 16 threads, but it starts to decay when more nodes are used. Further optimization is clearly possible and will be pursued in our next step.

Conclusions, Implications, and Future Directions

This letter reports our initial theoretical and the numerical development of a proposed multi-scale framework coupling a molecular mechanics treatment of solute molecules with a fluid mechanics treatment of continuum solvent models. There are several motivations in developing the new approach. First the hydrodynamic effect can be accurately captured by numerically solving the Navier-Stokes equation. Second the new model can capture the dynamics of moving solute particles without assumptions such as uniform size and full exposure of all solute particles as in existing approaches. Finally a general numerical framework based on the Navier-Stokes equation also facilitates the development of more robust approximated models for efficient simulations of hydrodynamic effect.

Specifically, the physical concepts necessary for coupling the two mechanics have been formulated. A 3D numerical algorithm also has been developed to describe the fluid dynamics of air bubbles immersed in an incompressible viscous fluid, which is used as a preliminary model to develop the fluid mechanics algorithm. The numerical algorithm is validated with multiple 3D test cases, including single-bubble oscillation dynamics and merging dynamics of multiple bubbles. It is noticed that for all the tested cases of dynamics, the errors in the equilibrium

volume for the air bubble as simulated are about 1%~2%, which is comparable to the error as expected based on the error analysis for the analytical system at the given finite-difference resolution. These numerical tests demonstrate that our numerical algorithm is effective and stable. Parallel implementation was explored and was shown to improve the computational speed noticeably.

Although our ultimate goal for the current development is a multiple-scale model for studying biomolecular systems, the validations reported here also demonstrates the feasibility to apply the fluid mechanics algorithm to simulate nano-scaled gas bubbles, or "nano bubbles", as long as a correct gas law is supplied. Nano-bubble has been widely used in the engineering area such as water treatment and drug delivery [49, 50]. Especially, it has been used as the targeted drug delivery in the treatments of various cancers [50]. Indeed, it is shown that nano bubbles are quite stable in water and unlikely to burst or dissolve once formed [49], which is consistent with our dynamics simulations as the simulated nano-bubbles quickly reach the static state (in the nanosecond time scales). Further efforts are still needed to validate the approach by comparing to corresponding measurable quantities from experiment.

Based on our initial development reported here, we plan to incorporate the fluid mechanics algorithm with an all-atom or coarse-grained molecular model to study the dynamics and functions of biomolecule machineries. In particular, we will implement the coupling interface with the molecular mechanics models and study the method's capability in reproducing the physical properties of solute/solvent interface in our next phase of the development.

Acknowledgements

This study was partially supported by NIGMS (GM093040 and GM079383) to R. Luo. Z. Li was partially supported by AFSOR (FA9550-09-1-0520), NIGMS (GM096195), NCSU RISF Fund (2013), and CNSF (11161036, 11371199, 11471166, BK20141443). H. Zhao was partially supported by NSF (DMS-1115698).

Supplemental Materials

Detailed discussion of physical model and mathematics is provided in the supplemental material. Animations were also generated for the bubble merging dynamics. "TwoBubble.gif" and "ThreeBubble.gif" are the animated trajectories of two bubble dynamics and three bubble dynamics, respectively. "TwoBubbleVelocity.gif" and "ThreeBubbleVelocity.gif" are the animated evolutions of velocity flows on a chosen interception plane for two bubble dynamics and three bubble dynamics, respectively.

References

1. P.W. Atkins and R.S. Friedman, Molecular Quantum Mechanics, Oxford, Oxford University Press, 2005.
2. S. Izvekov and G.A. Voth, *A Multiscale Coarse-Graining Method for Biomolecular Systems.* Journal of Physical Chemistry B, **109**: 2469-2473. 2005.
3. B.J. Alder and T.E. Wainwright, *Studies in Molecular Dynamics .1. General Method.* Journal of Chemical Physics, **31**: 459-466. 1959.
4. B. Honig and A. Nicholls, *Classical Electrostatics in Biology and Chemistry.* Science, **268**: 1144-1149. 1995.
5. A. Warshel and M. Levitt, *Theoretical Studies of Enzymic Reactions - Dielectric, Electrostatic and Steric Stabilization of Carbonium-Ion in Reaction of Lysozyme.* Journal of Molecular Biology, **103**: 227-249. 1976.
6. J.L. Gao and X.F. Xia, *A Priori Evaluation of Aqueous Polarization Effects through Monte-Carlo Qm-Mm Simulations.* Science, **258**: 631-635. 1992.
7. L.E. Reichl, A Modern Course in Statistical Physics, Verlag Chemie, Wiler-VCH.
8. A. Einstein, *Über Die Von Der Molekularkinetischen Theorie Der Wärme Geforderte Bewegung Von in Ruhenden Flüssigkeiten Suspendierten Teilchen.* Annalen Der Physik, **322**: 549-560. 1905.
9. P. Langevin, *Sur La Théorie Du Mouvement Brownien.* Comptes rendus de l'Académie des Sciences, **146**: 530-533. 1908.
10. D.L. Ermak and J.A. Mccammon, *Brownian Dynamics with Hydrodynamic Interactions.* Journal of Chemical Physics, **69**: 1352-1360. 1978.

11. R.C. Wade, B.A. Luty, E. Demchuk, J.D. Madura, M.E. Davis, J.M. Briggs, and J.A. Mccammon, *Simulation of Enzyme-Substrate Encounter with Gated Active-Sites.* Nature Structural Biology, **1**: 65-69. 1994.

12. R.R. Gabdoulline and R.C. Wade, *Brownian Dynamics Simulation of Protein-Protein Diffusional Encounter.* Methods-a Companion to Methods in Enzymology, **14**: 329-341. 1998.

13. D. Brune and S. Kim, *Predicting Protein Diffusion-Coefficients.* Proceedings of the National Academy of Sciences of the United States of America, **90**: 3835-3839. 1993.

14. J.B. Hubbard and J.F. Douglas, *Hydrodynamic Friction of Arbitrarily Shaped Brownian Particles.* Physical Review E, **47**: R2983-R2986. 1993.

15. J.G. Delatorre, S. Navarro, M.C.L. Martinez, F.G. Diaz, and J.J.L. Cascales, *Hydro - a Computer-Program for the Prediction of Hydrodynamic Properties of Macromolecules.* Biophysical Journal, **67**: 530-531. 1994.

16. J.F. Douglas and E.J. Garboczi, *Intrinsic Viscosity and the Polarizability of Particles Having a Wide Range of Shapes.* Advances in Chemical Physics, Vol 91, **91**: 85-153. 1995.

17. H.X. Zhou, *Calculation of Translational Friction and Intrinsic Viscosity .2. Application to Globular Proteins.* Biophysical Journal, **69**: 2298-2303. 1995.

18. B. Li, H. Sun, and S. Zhou, *Stability of a Cylindrical Solute-Solvent Interface: Effect of Geometry, Electrostatics, and Hydrodynamics.* submitted to SIAM J. Applied Math. 2014.

19. H.L. Friedman, *A Hydrodynamic Effect in the Rates of Diffusion-Controlled Reactions.* The Journal of Physical Chemistry, **70**: 3931-3933. 1966.

20. N. Wang, G.A. Huber, and J.A. McCammon, *Assessing the Two-Body Diffusion Tensor Calculated by the Bead Models.* Journal of Chemical Physics, **138**. 2013.

21. A.H. Elcock, *Molecule-Centered Method for Accelerating the Calculation of Hydrodynamic Interactions in Brownian Dynamics Simulations Containing Many Flexible Biomolecules.* Journal of Chemical Theory and Computation, **9**: 3224-3239. 2013.

22. J. Wang, P. Cieplak, J. Li, T. Hou, R. Luo, and Y. Duan, *Development of Polarizable Models for Molecular Mechanical Calculations I: Parameterization of Atomic Polarizability.* Journal of Physical Chemistry B, **115**: 3091-3099. 2011.

23. J. Wang, P. Cieplak, Q. Cai, M.J. Hsieh, J.M. Wang, Y. Duan, and R. Luo, *Development of Polarizable Models for Molecular Mechanical Calculations. 3. Polarizable Water Models Conforming to Thole Polarization Screening Schemes.* Journal of Physical Chemistry B, **116**: 7999-8008. 2012.

24. Y.-H. Tan and R. Luo, *Continuum Treatment of Electronic Polarization Effect.* Journal of Chemical Physics, **126**. 2007.

25. Y.-H. Tan, C. Tan, J. Wang, and R. Luo, *Continuum Polarizable Force Field within the Poisson-Boltzmann Framework.* Journal of Physical Chemistry B, **112**: 7675-7688. 2008.

26. Y. Duan, C. Wu, S. Chowdhury, M.C. Lee, G.M. Xiong, W. Zhang, R. Yang, P. Cieplak, R. Luo, T. Lee, J. Caldwell, J.M. Wang, and P. Kollman, *A Point-Charge Force Field for Molecular Mechanics Simulations of Proteins Based on Condensed-Phase Quantum Mechanical Calculations.* Journal of Computational Chemistry, **24**: 1999-2012. 2003.

27. M.J. Hsieh and R. Luo, *Physical Scoring Function Based on Amber Force Field and Poisson-Boltzmann Implicit Solvent for Protein Structure Prediction.* Proteins-Structure Function and Bioinformatics, **56**: 475-486. 2004.

28. T.Z. Lwin, R.H. Zhou, and R. Luo, *Is Poisson-Boltzmann Theory Insufficient for Protein Folding Simulations?* Journal of Chemical Physics, **124**. 2006.

29. L.J. Yang, C.H. Tan, M.J. Hsieh, J.M. Wang, Y. Duan, P. Cieplak, J. Caldwell, P.A. Kollman, and R. Luo, *New-Generation Amber United-Atom Force Field.* Journal of Physical Chemistry B, **110**: 13166-13176. 2006.

30. M.-J. Hsieh and R. Luo, *Balancing Simulation Accuracy and Efficiency with the Amber United Atom Force Field.* Journal of Physical Chemistry B, **114**: 2886-2893. 2010.

31. R.J. Becker, *Lagrangian/Hamiltonian Formalism for Description of Navier-Stokes Fluids.* Physical Review Letters, **58**: 1419-1422. 1987.

32. H. Fukagawa and Y. Fujitani, *A Variational Principle for Dissipative Fluid Dynamics.* Progress of Theoretical Physics, **127**: 921-935. 2012.

33. E.S. Reiner and C.J. Radke, *Variational Approach to the Electrostatic Free-Energy in Charged Colloidal Suspensions - General-Theory for Open Systems.* Journal of the Chemical Society-Faraday Transactions, **86**: 3901-3912. 1990.

34. K.A. Sharp and B. Honig, *Calculating Total Electrostatic Energies with the Nonlinear Poisson-Boltzmann Equation.* Journal of Physical Chemistry, **94**: 7684-7692. 1990.

35. L. Xiao, Q. Cai, X. Ye, J. Wang, and R. Luo, *Electrostatic Forces in the Poisson-Boltzmann Systems.* Journal of Chemical Physics, **139**. 2013.

36. L. Xiao, C.H. Wang, and R. Luo, *Recent Progress in Adapting Poisson-Boltzmann Methods to Molecular Simulations.* Journal of Theoretical & Computational Chemistry, **13**: 1430001. 2014.

37. D. Chandler, *Interfaces and the Driving Force of Hydrophobic Assembly.* Nature, **437**: 640-647. 2005.

38. J. Dzubiella, J.M.J. Swanson, and J.A. McCammon, *Coupling Hydrophobicity, Dispersion, and Electrostatics in Continuum Solvent Models.* Physical Review Letters, **96**. 2006.

39. C. Tan, Y.H. Tan, and R. Luo, *Implicit Nonpolar Solvent Models.* Journal Of Physical Chemistry B, **111**: 12263-12274. 2007.

40. L.T. Cheng, J. Dzubiella, J.A. McCammon, and B. Li, *Application of the Level-Set Method to the Implicit Solvation of Nonpolar Molecules.* Journal of Chemical Physics, **127**. 2007.

41. P.W. Bates, Z. Chen, Y.H. Sun, G.W. Wei, and S. Zhao, *Geometric and Potential Driving Formation and Evolution of Biomolecular Surfaces.* Journal of Mathematical Biology, **59**: 193-231. 2009.

42. D. Frenkel and B. Smit, Understanding Molecular Simulation - from Algorithms to Applications, Academic Press, 2001.

43. L.D. Landau and E.M. Lifshits, Statistical Physics. Course of Theoretical Physics, Oxford ; New York, Pergamon Press, 1980.

44. A.L. Fette and J.D. Walecka, Theoretical Mechanics of Particles and Continua, Mineola, New York, Dover Publications, Inc., 2003.

45. Z.L. Li, M.C. Lai, G.W. He, and H.K. Zhao, *An Augmented Method for Free Boundary Problems with Moving Contact Lines.* Computers & Fluids, **39**: 1033-1040. 2010.

46. J. Liu, *Open and Traction Boundary Conditions for the Incompressible Navier-Stokes Equations.* Journal of Computational Physics, **228**: 7250-7267. 2009.

47. S. Osher and R.P. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces. Applied Mathematical Sciencesxii, 273 p., New York, Springer, 2003.

48. L.S. Blackford, J. Choi, A. Cheary, J. D'Azevedo, S. Harmmarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R.C. Whaley, Scalapack Users' Guide, Philadelphia,PA, Society for Industrial and Applied Mathematics, 1997.

49. A. Agarwal, W.J. Ng, and Y. Liu, *Principle and Applications of Microbubble and Nanobubble Technology for Water Treatment.* Chemosphere, **84**: 1175-1180. 2011.

50. E.Y. Lukianova-Hleb, A. Belyanin, S. Kashinath, X.W. Wu, and D.O. Lapotko, *Plasmonic Nanobubble-Enhanced Endosomal Escape Processes for Selective and Guided Intracellular Delivery of Chemotherapy to Drug-Resistant Cancer Cells.* Biomaterials, **33**: 1821-1826. 2012.

Tables

Table I: Average CPU time per step for the three-bubble merging dynamics. The timing data were collected for ten steps when the simulation is close to equilibrium. The total CPU time is divided into three parts, those for computing the coefficient matrix in eqn (25), solving the linear equation, and other nonparallel procedures. The CPU time for other nonparallel procedures also includes the waiting time for slower threads to finish so it is longer in a parallel run. Note that all CPU times are rounded so they may or may not add up exactly to the total values.

	32 threads	1 thread
Compute coefficient matrix(s)	7.3	149.7
Solve linear equation (s)	8.1	42.3
Other nonparallel procedures (s)	4.6	3.3
Total CPU time (s)	20.0	195.3

Figures

Figure I. Order analysis of max errors in velocity and pressure for the numerical procedures used in solving NSE for the analytical test case. Both axes are in the log scale, so the regression slope represents the order of decreasing error with decreasing grid spacing *h*.

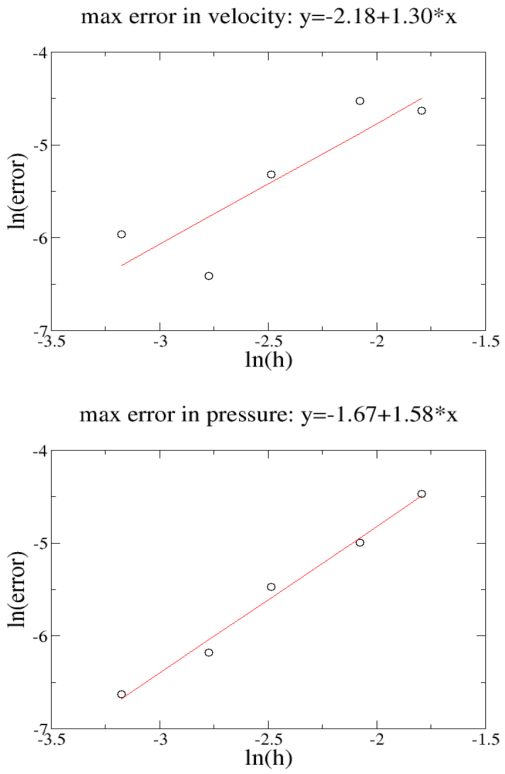

Figure II. Time evolutions (in *ps*) of the volume (in nm^3) of the tested single bubble with normal water (up) and low viscous fluid (bottom), respectively. Initial radius of the air bubble is $r_0 = 4nm$ and $r_0 = 6nm$, respectively. The equilibrium position is at $r = 5nm$.

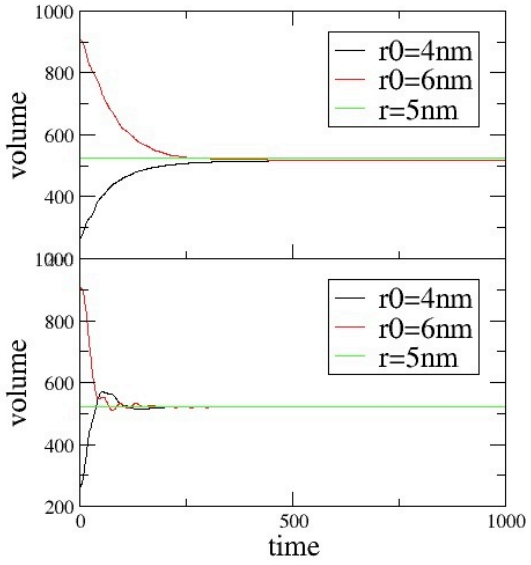

Figure III. Snapshots of two-bubble merging dynamics. Simulation is plotted in a box of 20*nm*×20*nm*×20*nm*. The bubble surface is represented as dots.

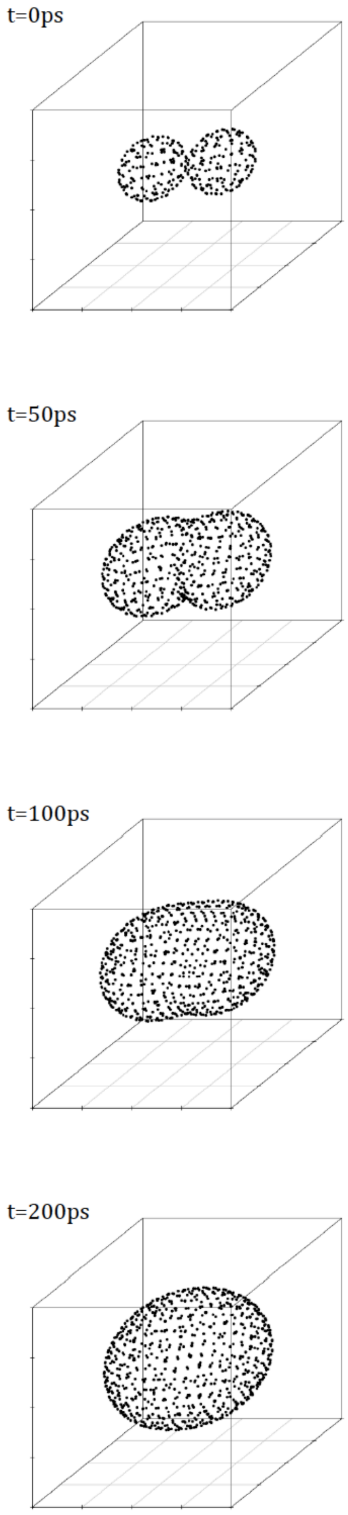

Figure IV. Snapshots of three-bubble merging dynamics. Simulation is plotted in a box of 30*nm*×30*nm*×30*nm*. The bubble surface is represented as dots.

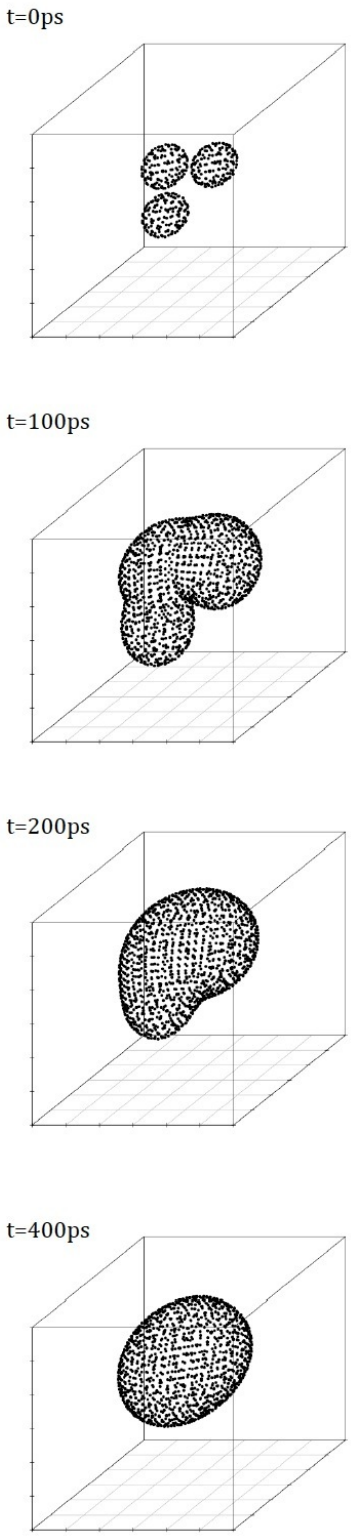

Figure V. Parallel scaling of three-bubble merging dynamics. The numbers of processes were chosen as 1, 2, 4, 8, 16, and 32. The speed up ratio is defined as the CPU time for the single-processor run over the CPU time for the tested multiple-processor run. See Table 1 for further detail.