SIAM J. SCI. COMPUT.
Vol. 38, No. 5, pp. S62–S77
© 2016 Society for Industrial and Applied Mathematics

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

FLUCTUATING HYDRODYNAMICS METHODS FOR DYNAMIC COARSE-GRAINED IMPLICIT-SOLVENT SIMULATIONS IN LAMMPS*

Y. WANG†, J. K. SIGURDSSON†, AND P. J. ATZBERGER†

Abstract. We introduce a software package integrated with the molecular dynamics software LAMMPS for fluctuating hydrodynamics simulations of fluid-structure interactions subject to thermal fluctuations. The package is motivated to provide dynamic thermostats to extend implicit-solvent coarse-grained (IS-CG) models by incorporating kinetic contributions from the solvent to facilitate their use in a wider range of applications. To capture the thermal and hydrodynamic contributions of the solvent to dynamics, we introduce momentum conserving thermostats and computational methods based on fluctuating hydrodynamics and the stochastic Eulerian Lagrangian method (SELM). SELM couples the coarse-grained microstructure degrees of freedom to continuum stochastic fields to capture both the relaxation of hydrodynamic modes and thermal fluctuations. Features of the SELM software include (i) numerical time-step integrators for SELM fluctuating hydrodynamics in inertial and quasi-steady regimes, (ii) Lees–Edwards-style methods for imposing shear, (iii) a Java-based graphical user interface (GUI) for setting up models and simulations, (iv) standardized XML formats for parametrization and data output, and (v) standardized formats VTK for continuum fields and microstructures. The SELM software package facilitates for pre-established models in LAMMPS easy adoption of the SELM fluctuating hydrodynamics thermostats. We provide here an overview of the SELM software package, computational methods, and applications.

Key words. fluctuating hydrodynamics, implicit solvent, coarse-grained, stochastic Eulerian Lagrangian method, immersed boundary method, soft materials

AMS subject classification. 92-08

DOI. 10.1137/15M1026390

1. Introduction.

We introduce a computational package for fluctuating hydrodynamics thermostats for dynamic simulations of implicit-solvent coarse-grained (IS-CG) models. IS-CG models have been developed to study phenomena relevant to soft materials and biophysics on length and time scales difficult to attain with fully atomistic molecular dynamics. IS-CG models explicitly model microstructures at a coarse-grained level and remove the solvent degrees of freedom to treat instead the solvent contributions implicitly in the effective free energy of interaction between the microstructures. Gains in computational efficiency are achieved through (i) a reduction in the number of degrees of freedom as a consequence of the removed solvent and coarse-graining of the microstructure and (ii) reducing the roughness and complexity of the energy landscape that results in less stiff mechanics and more rapid equilibration. The IS-CG approach has worked well to reveal insights into diverse phenomena relevant to soft materials and biophysics [12, 14, 16, 19, 23, 30, 38, 41].

IS-CG models have primarily been motivated by and used for studying equilibrium properties of soft materials using Monte Carlo sampling or Langevin dynamics. For kinetic studies, IS-CG models simulated with Langevin dynamics neglect important

*Received by the editors June 6, 2015; accepted for publication (in revised form) January 25, 2016; published electronically October 27, 2016.
http://www.siam.org/journals/sisc/38-5/M102639.html
†Department of Mathematics and Department of Mechanical Engineering, University of California Santa Barbara, Santa Barbara, CA 93106 (lalalaoo@gmail.com, jonksig@gmail.com, atzberg@math.ucsb.edu). The second and third authors acknowledge support from research grant NSF CAREER - 0956210 and DOE ASCR CM4 DE-SC0009254. The first and third authors acknowledge support from the W.M. Keck Foundation.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

contributions in the kinetics arising from the missing solvent degrees of freedom. The solvent contributes not only to the free energy of interaction but also to the kinetics by mediating lateral momentum transport as manifested in hydrodynamics. The Langevin thermostat uses local sources and sinks of momentum that suppress such lateral correlations between microstructures [44]. To capture consistently at the level of hydrodynamics the momentum transport and thermal fluctuations, we introduce a momentum conserving thermostat based on fluctuating hydrodynamics referred to as the stochastic Eulerian Lagrangian method (SELM) [7]. In SELM, we introduce continuum stochastic fields that are coupled to the implicit-solvent models to thermostat the system in a manner which conserves momentum [7].

2. Stochastic Eulerian Lagrangian method (SELM). SELM provides a framework for modeling fluid-structure interactions subject to thermal fluctuations. To obtain a tractable description, approximate operators modeling the fluid-structure interaction can be used as in the immersed boundary (IB) method [34]. A Lagrangian description of the microstructure, typically a collection of markers in the fluid, is coupled to an Eulerian mesh for the hydrodynamics; see Figure 1. The thermal fluctuations are accounted for by stochastic driving fields introduced in a manner consistent with the approximation and statistical mechanics [7]. This facilitates the development of efficient stochastic numerical methods building upon deterministic computational fluid dynamics solvers. Microstructures can include point particles, slender filaments, or solid bodies [7, 13, 34].

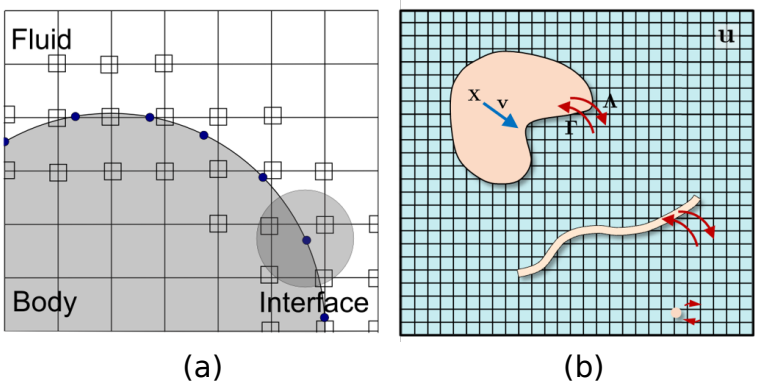

FIG. 1. *SELM.* (a) *Coupling of a Lagrangian body with the Eulerian discretization mesh.* (b) *Can represent extended bodies, filaments, or point particles.*

2.1. Inertial regime. In the inertial description of the fluid-structure system, we model the microstructure dynamics similar to Langevin by

$$\frac{d\mathbf{X}}{dt} = \mathbf{v}, \tag{1}$$

$$m\frac{d\mathbf{v}}{dt} = -\Upsilon(\mathbf{v} - \Gamma\mathbf{u}) - \nabla_X \Phi[\mathbf{X}] + \mathbf{F}_{thm}. \tag{2}$$

A key difference with Langevin is that we reference the drag force relative to the solvent hydrodynamic field $\mathbf{u}$. The contributions of the solvent fluid are modeled by

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

the incompressible fluctuating hydrodynamics

$$\rho \frac{\partial \mathbf{u}}{\partial t} = \mu \Delta \mathbf{u} - \nabla p + \Lambda[\Upsilon(\mathbf{v} - \Gamma \mathbf{u})] + \mathbf{f}_{thm}, \tag{3}$$

$$\nabla \cdot \mathbf{u} = 0. \tag{4}$$

In the notation, $\mathbf{X}$ denotes the collective degrees of freedom of the microstructures, $\mathbf{v}$ the microstructure velocity, and m the microstructure excess mass [7, 42]. The fluid velocity is denoted by $\mathbf{u}$, the fluid density by ρ, and the dynamic viscosity by μ. The pressure acts as a Lagrange multiplier to enforce the incompressibility constraint given in (4). The Υ denotes the coefficient of microstructure drag with respect to the fluid and Φ the potential energy associated with the microstructure configuration $\mathbf{X}$.

Thermal fluctuations are taken into account by Gaussian stochastic driving fields $\mathbf{F}_{thm}$ and $\mathbf{f}_{thm}$ with mean zero and moments

$$\begin{aligned} \langle \mathbf{f}_{\text{thm}}(s) \mathbf{f}^T_{\text{thm}}(t) \rangle &= -\left(2k_B T\right)\left(\mathcal{L} - \Lambda \Upsilon \Gamma\right) \delta(t-s), \\ \langle \mathbf{F}_{\text{thm}}(s) \mathbf{F}^T_{\text{thm}}(t) \rangle &= \left(2k_B T\right) \Upsilon \, \delta(t-s), \\ \langle \mathbf{f}_{\text{thm}}(s) \mathbf{F}^T_{\text{thm}}(t) \rangle &= -\left(2k_B T\right) \Lambda \Upsilon \, \delta(t-s). \end{aligned} \tag{5}$$

We denote $\mathcal{L} = \mu \Delta \mathbf{u}$. The stochastic equations are to be given the Ito interpretation throughout [22, 31]. This particular spatial covariance was derived for SELM using the fluctuation-dissipation principle of statistical mechanics [7, 37].

The operators Γ and Λ model the fluid-structure interactions through the equal and opposite dissipative terms $-\Upsilon(\mathbf{v} - \Gamma \mathbf{u})$ acting as a drag force on the microstructures and $\Lambda\Upsilon(\mathbf{v} - \Gamma \mathbf{u})$ acting as a drag force density on the fluid [7, 42]. To achieve desirable properties in the mechanics and numerics we require the coupling operators to be adjoints throughout [7, 34, 42]. The fluid-structure interactions and particular choice of Γ, Λ contribute important correlations in the thermal fluctuations; see (5).

Many types of operators can be used to couple the microstructure and fluid depending on the problem [7]. For simplicity, we take the widely used *immersed boundary (IB) method* [34] which is based on a kernel function to perform averages using markers in the fluid to obtain a reference velocity and to perform force spreading (see Figure 1):

$$\Gamma \mathbf{u} = \int_{\Omega} \eta\left(\mathbf{y} - \mathbf{X}(t)\right) \mathbf{u}(\mathbf{y}, t) d\mathbf{y}, \tag{6}$$

$$\Lambda \mathbf{F} = \eta\left(\mathbf{x} - \mathbf{X}(t)\right) \mathbf{F}. \tag{7}$$

The kernel functions $\eta(\mathbf{z})$ are chosen to be the Peskin δ-function, which has a number of important properties, such as near-translational invariance over the mesh, which is useful in numerical methods [5, 34].

2.2. Quasi-steady regime. A central challenge in the development of viable numerical methods for (1)–(4) is the significant temporal stiffness that arises from the stochastic driving fields that excite diverse scales in the fluid-structure system [7]. This has been handled through the development of stiff numerical time-step integrators [5] and, alternatively, through the development of stochastic asymptotics that exploit a separation of time-scales to obtain reduced stochastic equations having less stiff dynamics [7, 42].

In problems where the overall hydrodynamic coupling is important but not the relaxation dynamics of the hydrodynamic modes, the SELM equations can be reduced

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

to [7, 42]

$$\frac{d\mathbf{X}}{dt} = H_{\text{SELM}}[-\nabla_{\mathbf{X}}\Phi(\mathbf{X})] + (\nabla_{\mathbf{X}} \cdot H_{\text{SELM}})k_BT + \mathbf{h}_{\text{thm}}, \tag{8}$$

$$H_{\text{SELM}} = \Gamma(-\wp\mathcal{L})^{-1}\Lambda, \tag{9}$$

$$\langle \mathbf{h}_{\text{thm}}(s)\mathbf{h}_{\text{thm}}^T(t)\rangle = (2k_BT)\, H_{\text{SELM}}\, \delta(t-s). \tag{10}$$

$\mathcal{L} = \mu\Delta$, and $\wp$ denotes a projection operator that imposes the incompressibility constraint in (4) [13, 42].

This provides a mesh-based approach to computing the quasi-steady hydrodynamic coupling in a manner especially useful for complex geometries or when imposing special boundary conditions [8, 36]. This formulation of SELM treats a physical regime similar to Brownian–Stokesian dynamics simulations [4, 11, 18]. For a more detailed discussion and SELM methods for other physical regimes, see [5, 7, 42].

2.3. Computational methods. In the current SELM package release, we consider numerical methods and implementations for the two extremal regimes: (i) fully inertial dynamics of the microstructure and hydrodynamics, and (ii) overdamped dynamics of the microstructure subject to quasi-steady hydrodynamics. For SELM methods for other physical regimes and more details, see [7, 42].

A central challenge in developing viable computational methods for the fluctuating hydrodynamic equations (1)–(4) is that solutions $\mathbf{u}$ are highly irregular in space and time. Technically, the fields are solutions of the stochastic partial differential equations only in a weak generalized sense described by distributions [29, 39]. This requires special consideration in the development of discretizations and in the approximation of the stochastic driving fields [5, 7].

2.3.1. Spatial discretization. Many different approaches can be used to discretize SELM, including spectral methods, finite differences, and finite elements [5, 7, 36]. For simplicity, we discuss here the case of finite difference methods on a uniform periodic mesh. We approximate the Laplacian $\Delta\mathbf{u} \sim L\mathbf{u}$ where

$$[L\mathbf{u}]_{\mathbf{m}} = \sum_{j=1}^{3} \frac{\mathbf{u}_{\mathbf{m}+\mathbf{e}_j} - 2\mathbf{u}_{\mathbf{m}} + \mathbf{u}_{\mathbf{m}-\mathbf{e}_j}}{\Delta x^2}. \tag{11}$$

We approximate the fluid incompressibility constraint $\nabla \cdot \mathbf{u} = 0$ by the divergence operator $\nabla \cdot \mathbf{u} \sim D \cdot \mathbf{u}$ where

$$[D \cdot \mathbf{u}]_{\mathbf{m}} = \sum_{j=1}^{3} \frac{\mathbf{u}^j_{\mathbf{m}+\mathbf{e}_j} - \mathbf{u}^j_{\mathbf{m}-\mathbf{e}_j}}{2\Delta x}. \tag{12}$$

$\mathbf{m} = (m_1, m_2, m_3)$ denotes the index of the lattice site. $\mathbf{e}_j$ denotes the standard basis vector in three dimensions. We spatially semidiscretize the SELM equations by replacing the operators in (1)–(4) with the corresponding discrete operators. We approximate the stochastic driving fields by replacing the continuum fields with a Gaussian process on the lattice sites of the mesh with moments imposed by (5) corresponding to the discrete operators. This ensures the discretization approximates' fluctuation-dissipation balance and can be shown to have other desirable properties. For a more detailed discussion, see [5, 7].

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

2.3.2. Temporal discretization. For the SELM dynamics in (1)–(4), we develop a temporal integrator that extends the Velocity-Verlet method used in molecular dynamics [43]. The Velocity-Verlet method was originally developed for integrating deterministic time-reversible dynamics such as Newton's equations of mechanics to preserve symmetries to achieve advantageous stability and energy conservation [2, 21, 43]. In the stochastic setting, the time-reversible symmetry is broken by the dissipative terms and the stochastic driving fields. However, despite this broken symmetry, the scheme still offers some advantages over Euler–Marayuma [27]. For the SELM equations (1)–(4), we use the Verlet-style integrator

$$
\begin{aligned}
\mathbf{v}^{n+\frac{1}{2}} &= \mathbf{v}^n + \frac{\Delta t}{2} m^{-1}\mathbf{F}^n \\
&\quad + \frac{\Delta t}{2}\left(-m^{-1}\Upsilon\left(\mathbf{v}^{n-\frac{1}{2}} - \Gamma^n \mathbf{u}^{n-\frac{1}{2}}\right)\right. \\
&\quad \left. + m^{-1}\mathbf{g}^{n-\frac{1}{2}}\right), \\
\mathbf{u}^{n+\frac{1}{2}} &= \mathbf{u}^n + \frac{\Delta t}{2}\rho^{-1}\mu L \mathbf{u}^{n-\frac{1}{2}} \\
&\quad - \frac{\Delta t}{2}\left(\rho^{-1}\Lambda^n\left[-\Upsilon\left(\mathbf{v}^{n-\frac{1}{2}} - \Gamma^n \mathbf{u}^{n-\frac{1}{2}}\right)\right.\right. \\
&\quad \left.\left. + \mathbf{g}^{n-\frac{1}{2}}\right]\right) \\
&\quad + \mathbf{h}^{n-\frac{1}{2}}, \\
\mathbf{X}^{n+1} &= \mathbf{X}^n + \mathbf{v}^{n+\frac{1}{2}}\Delta t, \\
\mathbf{v}^{n+1} &= \mathbf{v}^{n+\frac{1}{2}} + \frac{\Delta t}{2} m^{-1}\mathbf{F}^{n+1} \\
&\quad + \frac{\Delta t}{2}\left(-m^{-1}\Upsilon\left(\mathbf{v}^{n+\frac{1}{2}} - \Gamma^{n+1} \mathbf{u}^{n+\frac{1}{2}}\right)\right. \\
&\quad \left. + m^{-1}\mathbf{g}^{n+\frac{1}{2}}\right), \\
\mathbf{u}^{n+1} &= \mathbf{u}^{n+\frac{1}{2}} + \frac{\Delta t}{2}\rho^{-1}\mu L \mathbf{u}^{n+\frac{1}{2}} \\
&\quad - \frac{\Delta t}{2}\left(\rho^{-1}\Lambda^{n+1}\left[-\Upsilon\left(\mathbf{v}^{n+\frac{1}{2}} - \Gamma^{n+1} \mathbf{u}^{n+\frac{1}{2}}\right)\right.\right. \\
&\quad \left.\left. + \mathbf{g}^{n+\frac{1}{2}}\right]\right) \\
&\quad + \mathbf{h}^{n+\frac{1}{2}},
\end{aligned}
\tag{13}
$$

where

$$
\langle \mathbf{g}^{n-\frac{1}{2}} \mathbf{g}^{n-\frac{1}{2}T} \rangle = 4k_B T \Upsilon/\Delta t, \tag{14}
$$

$$
\langle \mathbf{h}^n \mathbf{h}^{nT} \rangle = 4k_B T \rho^{-2}\mu L/\Delta t. \tag{15}
$$

The $\mathbf{F}^n$ gives the forces for particle configuration $\mathbf{X}^n$. The scheme extends the Velocity-Verlet method to include the dissipative and stochastic terms by sampling them at the half-time-steps in a staggered manner relative to the microstructure configurations. The numerical integrator is momentum conserving even in the presence of the dissipative and stochastic driving terms which can be shown to only transfer momentum between the microstructure and hydrodynamic fields. This can be contrasted with the Langevin dynamics which uses local sources and sinks of momentum

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

to thermostat. Finally, to temporally discretize the quasi-steady SELM dynamics in (8)–(10), we use the Euler–Marayuma method [8, 27].

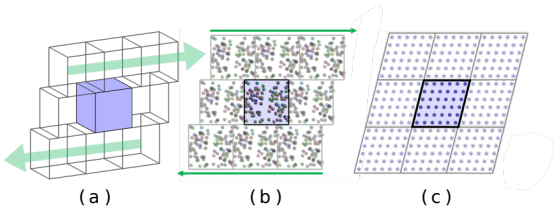

FIG. 2. *Lees–Edwards boundary conditions.* (a) *"Sliding bricks" model for imposed shear;* (b) *microstructure interactions with shifted periodic images;* (c) *deforming discretization mesh for hydrodynamics.*

2.4. Shear boundary conditions: Lees–Edwards for SELM. To model imposed shear stress on a simulation domain, Lees and Edwards introduced methods for molecular dynamics [28]. The central idea is to use a "sliding bricks model," where a periodic-like boundary condition is imposed on interactions near the boundary but with a time-dependent shift of the periodic images. In addition, the velocities of particles in the periodic images are accordingly adjusted; see Figure 2. We have developed a similar approach in the context of SELM by imposing in the hydrodynamic equations the condition [8]

$$\mathbf{u}(x, y, L, t) = \mathbf{u}(x - vt, y, 0, t) + v\mathbf{e}_x. \tag{16}$$

This corresponds to a domain of size L with shear along the z-axis in the x-direction at the shear rate $\dot{\gamma} = v/L$. However, in numerical discretizations on a Cartesian mesh the shift $x - vt$ is inconvenient and results in interpolation error from a mismatch of lattice points [8]. To avoid this issue, the SELM fluctuating hydrodynamic equations are reformulated and solved on a deforming mesh for the equivalent hydrodynamic field $\mathbf{w}(\mathbf{q}, t) = \mathbf{u}(\phi(\mathbf{q}, t), t)$, where $\phi(\mathbf{q}, t) = (q_1 + q_3\dot{\gamma}t, q_2, q_3)$ and $\mathbf{q} = (q_1, q_2, q_3)$ parametrizes the unit cell. The jump in velocity at the boundary is handled by introducing a localized source term in the SELM equations. This reformulation allows for the field $\mathbf{w}$ to be treated numerically as periodic $\mathbf{w}(q_1, q_2, L, t) = \mathbf{w}(q_1, q_2, 0, t)$. This allows for efficient computational methods using FFTs [8].

An important feature of the Lees–Edwards-style approach is that shear is imposed by modifying interactions only locally near the domain boundary. This is in contrast to imposing a global affine transformation of the entire simulation domain as is sometimes done in studies of polymeric networks [20, 40]. This local-global distinction can be important since shear stresses can induce nonaffine deformations in systems [9, 25, 40]. The approach above allows for incorporating the Lees–Edwards-style conditions for imposing shear into SELM fluctuating hydrodynamic simulations [8]. We give an example simulation using these methods in section 5.2.

3. SELM software package for LAMMPS. To facilitate use by a wide community, we have integrated implementation of the SELM computational methods with the LAMMPS molecular dynamics software [35]. The methods have been implemented in C++. An overview of how the codes are used to set up models, interact with LAMMPS, and produce simulation output is shown in Figure 3.

Models can be set up in a few different ways, including (i) custom commands in the LAMMPS script, (ii) Python codes to generate input data and control SELM-

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

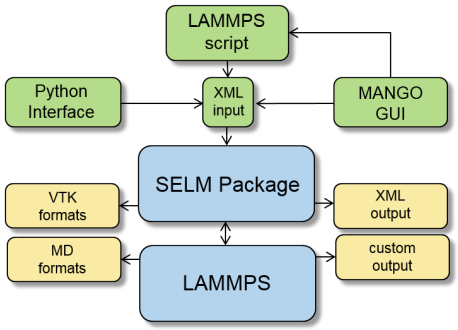

FIG. 3. *Package interactions and data flow. SELM simulations can be set up with Python, LAMMPS scripts, or the MANGO GUI. Standardized XML formats are used for input and output.*

LAMMPS, or (iii) the MANGO graphical user interface (GUI). The main SELM module interfaces with LAMMPS through a custom "fix class" referred to as USER-SELM in the terminology of LAMMPS. These codes provide the hooks for the time-stepping routines, force interactions, calculations of statistics, and data input/output. The SELM module obtains model geometry and parameters through standard LAMMPS data structures and by reading select parameter files having a standardized XML format that closely follows the object classes of SELM.

LAMMPS-SELM Interface	XML Interface
fix_SELM.cpp	Atz_XML_Helper_ParseData.cpp
fix_SELM_XML_Handler.cpp	Atz_XML_Package.cpp
SELM_Package.cpp	Atz_XML_Parser.cpp
Atz_XML_Handler_Example_A.cpp	Atz_XML_SAX_DataHandler.cpp
Atz_XML_Helper_DataHandler_List.cpp	Atz_XML_SAX_Handler_Multilevel.cpp
Atz_XML_Helper_Handler_SkipNextTag.cpp	Atz_XML_SAX_Handler_PrintToScreen.cpp
Eulerian Mechanics	Lagrangian Mechanics
SELM_Eulerian.h	SELM_Lagrangian.h
SELM_Eulerian_Types.h	SELM_Lagrangian_Delegator_XML_Handler.h
SELM_Eulerian_Delegator_XML_Handler.h	SELM_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE.h
SELM_Eulerian_LAMMPS_SHEAR_UNIFORM1_FFTW3.h	SELM_Lagrangian_LAMMPS_ATOM_ANGLE_STYLE_XML_Handler.h
SELM_Eulerian_LAMMPS_SHEAR_UNIFORM1_FFTW3_XML_Handler.h	SELM_Lagrangian_Types.h
SELM_Eulerian_Uniform1_Periodic.h	SELM_Package.h
SELM_Eulerian_Uniform1_Periodic_XML_Handler.h	
Time-Step Integration	Fluid-Structure Coupling
SELM_Integrator.h	SELM_CouplingOperator.h
SELM_Integrator_Delegator_XML_Handler.h	SELM_CouplingOperator_Delegator_XML_Handler.h
SELM_Integrator_FFTW3_Period.h	SELM_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1.h
SELM_Integrator_LAMMPS_SHEAR_QUASI_STEADY1_FFTW3.h	SELM_CouplingOperator_LAMMPS_SHEAR_UNIFORM1_FFTW3_TABLE1_XML_Handler.h
SELM_Integrator_LAMMPS_SHEAR_QUASI_STEADY1_FFTW3_XML_Handler.h	

FIG. 4. *Source codes in C++ for the SELMs.*

The C++ classes can be organized into roughly six categories (i) Eulerian mechanics, (ii) Lagrangian mechanics, (iii) coupling operators, (iv) force interactions, (v) time-step integrators, and (vi) XML processors. We show a typical collection of source files from our first release in Figure 4. The specific C++ classes and source files for the current release can be found in the distribution package. The classes are designed to operate with few interdependencies and interact through a standardized programming interface. In addition, each of the classes receives parameter values through a standardized XML interface.

The implementation has been designed for each of the general class categories to be easily extended for the creation of new spatial-temporal numerical methods, types of Eulerian–Lagrangian descriptions, and physical models. Each category has a "delegator class" that is responsible for interpreting the class type from an identify string passed along from a script or XML data associated with a given physical model [10, 17]. In practice, this is done easily by creating a new derived class imple-

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

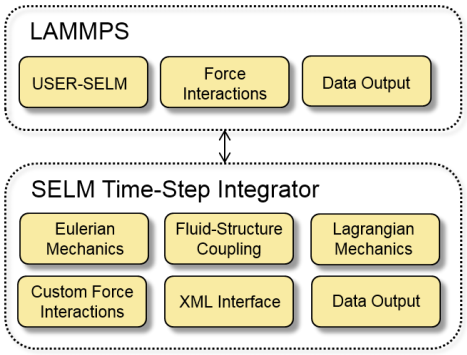

FIG. 5. *The package USER-SELM and the SELM time-step integrator classes coordinate the simulation. Shown are the broad categories of C++ classes and the interactions between SELM and LAMMPS.*

menting the standardized interface and by updating the delegator class to include an identifier string linked with this new class.

The primary LAMMPS-SELM interface is implemented in the class *fix_SELM.cpp*. The time-step integrator class coordinates primarily the software components shown in Figure 5. In a typical simulation of the Vertlet style, the integrator class performs the following operations: (i) receives input concerning the physical state from LAMMPS; (ii) integrates the initial half-time-step for the stochastic dynamics of the microstructure and hydrodynamic fields; (iii) computes the microstructure-fluid hydrodynamic interactions using the specified fluid-structure coupling operators; (iv) computes any custom interaction forces acting on the microstructures or hydrodynamic fields; (v) returns output data and control to LAMMPS to complete the initial half-time-step; (vi) receives final half-time-step input from LAMMPS; (vii) integrates the final half-time-step for the stochastic dynamics of the microstructure and hydrodynamic fields similar to steps (iii) and (iv); and (viii) returns output data and control to LAMMPS to repeat the above steps. An important task handled by LAMMPS is to efficiently compute the bonded and nonbonded interactions for different types of potentials and boundary conditions using specialized data structures and sorting methods [35]. In summary, the modular design of the package facilitates future extensions and development of the SELM fluctuating hydrodynamics methods.

4. Model specification. Models can be set up using the SELM software package in the following ways: (i) custom commands in the LAMMPS script, (ii) Python codes to generate input data and control SELM-LAMMPS, or (iii) the Java-based MANGO GUI.

4.1. LAMMPS scripts. For simple models, the LAMMPS script can be modified easily so that the integrator is used from the SELM package. This can done by use of a command of the form

```
fix 1 all SELM FENE_Dimer.SELM_params
```

This gives the name of a master XML file that specifies the model. The master XML file specifies the Eulerian mechanics for the hydrodynamics, fluid-structure coupling, and other aspects of the SELM model and parametrization. An example demonstrating this approach can be found in the folder */USER-SELM/examples/FENE_Dimer/*. This provides a particularly simple way to convert an existing model already set up

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

in LAMMPS.

4.2. Python interface to LAMMPS-SELM. Another approach to setting up models is to use a Python interface to LAMMPS and the SELM package. This allows for models to be specified programmatically. LAMMPS provides an interface allowing for any script command to be called interactively from Python. In the current release, Python interacts with SELM through the standard LAMMPS interface and through the generation of custom XML data files. In a typical simulation, the model is specified by developing a custom Python script that generates the needed LAMMPS data structures and XML files that control the SELM package and performing a LAMMPS simulation run. This provides a straightforward way to readily adopt models already set up in LAMMPS using Python.

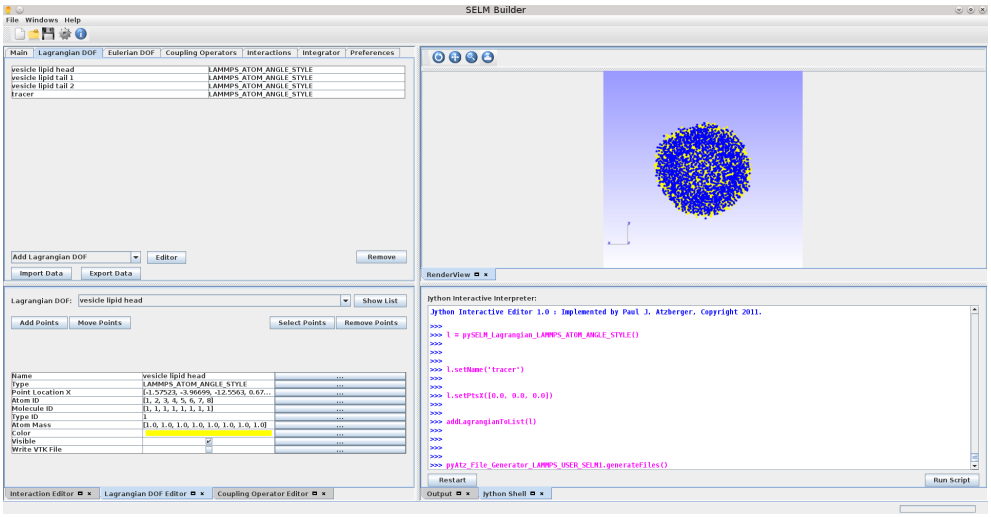

FIG. 6. *Screenshot of the MANGO GUI for setting up models and simulations.*

4.3. Graphical modeling software: MANGO. We have developed a Java-based [24] GUI for setting up SELM-LAMMPS models which is referred to as MANGO (Modeling and Numerical Graphical Orchestrator). The MANGO software allows for spreadsheet-like specification of parameters and interactive construction and visualization of models; see Figure 6. MANGO has been implemented in the Java programming language [24] using a modular design, readily allowing for extension mirroring developments in the SELM codes. In the current release simulations can be set up for overdamped shear simulations. The interface allows for interactive editing of the geometry of the Lagrangian microstructures. For instance, the interface allows for new control nodes in a model to be created or deleted and to be moved interactively. We also developed in MANGO an interface that allows for Python-style scripting through a mimetic language called Jython [32]. Many Python scripts can be run directly in Jython or with minor modifications. For running simulations, the MANGO interface automatically generates both the LAMMPS script driving the simulation and all needed XML data files for the SELM package. The MANGO graphical interface provides a particularly easy entry-point for new users to set up SELM models and perform simulations. An example project and simulation using the MANGO interface can be found in the folder */USER-SELM/examples/mango-project_FENE_Dimer/* by

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

opening *FENE_Dimer.SELM_Builder_Project*.

5. Applications. We discuss a few computational simulations performed using the SELM fluctuating hydrodynamics numerical methods. Many of these simulation results have been reported in more detail in prior papers [6, 7, 8, 44]. To demonstrate the core capabilities of the SELM methods, we discuss two particular applications. The first is a basic model for a polymeric material consisting of short polymer segments that have bonds that can be irreversibly broken when subjected to shear [8]. We study how the shear viscosity of the material changes over time as bonds are broken and the microstructure rearranges. The second is a dynamic extension of the IS-CG model for lipids developed by Cooke, Kremer, and Deserno [14]. For a self-assembled vesicle, we show how the SELM fluctuating hydrodynamics captures important collective dynamics of the lipids that are missing in implicit-solvent simulations using Langevin dynamics [44].

5.1. Physical benchmarks. We briefly discuss features of how the SELM methods capture hydrodynamic interactions and thermal fluctuations. We benchmark SELM against other hydrodynamic models used in the literature and with results from statistical mechanics.

The effective hydrodynamic interactions in SELM when using IB coupling in (6) yields interactions similar to the Rotne–Prager–Yamakawa tensor [8, 45]; see Figure 7.

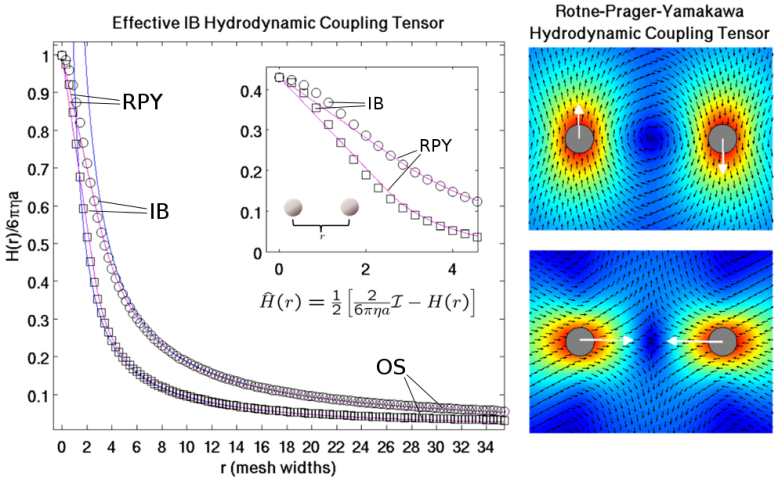

FIG. 7. *Hydrodynamic interactions. IB coupling for the parallel and perpendicular components of the pair-mobility tensor (data points), Rotne–Prager–Yamakawa (RPY) tensor* [45]*, and Oseen (OS) tensor* [1].

The IB coupling used with SELM exhibits in the far-field the same behavior as the Oseen tensor and in the near-field a regularized interaction similar to Rotne–Prager–Yamakawa [8, 45].

For a particle tethered by a harmonic spring, we benchmark the results of SELM to the predictions of equilibrium statistical mechanics [6, 37]; see Figure 8. For SELM within the inertial regime, we find good agreement with the Gibbs–Boltzmann distribution of statistical mechanics, both for the exhibited distribution of particle positions and for the distribution of particle velocities. For more details, see [6, 37].

As a further benchmark, we consider the motions of a pair of ellipsoidal particles in

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

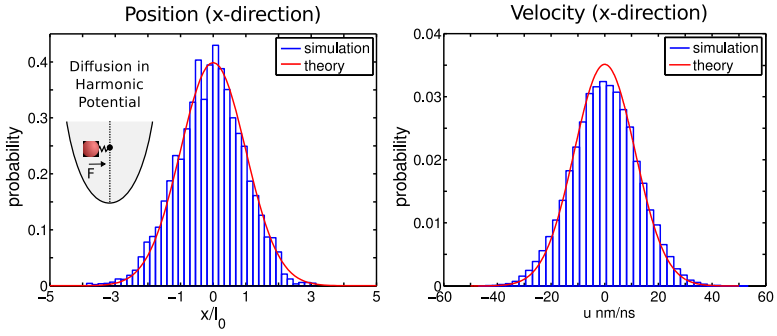

FIG. 8. *Particle subject to harmonic tether. The probability distribution generated by SELM simulations of a particle subject to a harmonic tether. The particle position is shown on the left, and the particle velocity is shown on the right. For more details, see* [6].

proximity to a wall. We compare the correlations in the passive diffusive motions with the deterministic motions associated with the hydrodynamic coupling in response to a force [6]; see Figure 9. The results confirm empirically that the stochastic dynamics generated by SELM exhibit a Stokes–Einstein relation between the mobility capturing the hydrodynamic responses and the tensor for the correlated diffusive motions. For more details, see [6]. In the bulk, we also found in [6] that the SELM hydrodynamic responses for the ellipsoidal particles are in agreement with prior fluid mechanics calculations for ellipsoid-shaped particles; see [6, 15, 33].

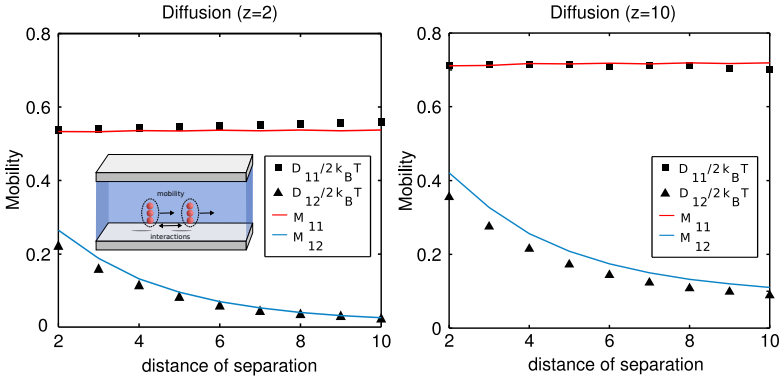

FIG. 9. *Diffusivity of ellipsoidal particles near a wall. For two interacting ellipsoidal particles, the correlated diffusivity tensor components are compared to the hydrodynamic mobility components. Good agreement is found both for particles near the channel center $z = 10nm$ and for those near the wall $z = 2nm$. For more details, see* [6].

Overall, these benchmark studies validate that the SELM methods yield reasonable results for the hydrodynamics and fluctuations consistent with prior fluid mechanics results in the literature and statistical mechanics [6, 8, 15, 33, 37, 45]. The SELM methods can be used to perform simulations for diverse applications.

5.2. Polymeric material. A basic model has been developed using SELM for a polymeric material with microstructures comprised of cross-linked polymer chains [8]. The polymeric chains are each comprised of five control points, and each have specialized binding sites at the second and fourth control points. The interpolymer bonds

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

have a preferred extension and angle of 70^o. When an interpolymer bond is strained beyond 50% of its preferred rest-length, the bond breaks irreversibly; see Figure 10.

This is modeled by the interaction energy

$$
\begin{aligned}
\Phi[\mathbf{X}] &= \Phi_{mb} + \Phi_{ma} + \Phi_{pb} + \Phi_{pa}, \\
\Phi_{mb}[\mathbf{X}] &= \sum_{(i,j)\in\mathcal{Q}_1} \phi_{mb}(r_{ij}), \\
\Phi_{ma}[\mathbf{X}] &= \sum_{(i,j,k)\in\mathcal{Q}_2} \phi_{ma}(\boldsymbol{\tau}_{ij}, \boldsymbol{\tau}_{jk}), \\
\Phi_{pb}[\mathbf{X}] &= \sum_{(i,j)\in\mathcal{Q}_3} \phi_{pb}(r_{ij}), \\
\Phi_{pa}[\mathbf{X}] &= \sum_{(i,j,k)\in\mathcal{Q}_4} \phi_{pa}(\theta_{ijk}),
\end{aligned} \tag{17}
$$

where

$$
\begin{aligned}
\phi_{mb}(r) &= \frac{1}{2} K_1 (r - r_{0,1})^2, \\
\phi_{ma}(\boldsymbol{\tau}_1, \boldsymbol{\tau}_2) &= \frac{1}{2} K_2 \left| \boldsymbol{\tau}_1 - \boldsymbol{\tau}_2 \right|^2, \\
\phi_{pb}(r) &= \sigma^2 K_3 \exp\left[-\frac{(r - r_{0,3})^2}{2\sigma^2} \right], \\
\phi_{pa}(\theta) &= -K_4 \cos(\theta - \theta_{0,4}).
\end{aligned} \tag{18}
$$

The energy terms are Φ_{mb} for monomer bonds, Φ_{ma} for monomer bond angles, Φ_{pb} for interpolymer bonds, and Φ_{pa} for interpolymer bond angles. The sets $\mathcal{Q}_k$ define the interactions according to the structure of the individual polymer chains and the topology of the interpolymer network. r is the separation distance between two monomers, θ is the bond angle between three monomers, and $\boldsymbol{\tau}$ is a tangent vector along the polymer chain. When bonds are stretched beyond the critical length 3σ, they are broken irreversibly, which results in the sets $\mathcal{Q}_3$ and $\mathcal{Q}_4$ being time dependent. For more details and the specific simulation parameters, see [8]. The model is shown in Figure 10.

To show how the methods can be used to investigate the relationship between the polymeric microstructures and contributions to the shear viscosity $\eta_p = \sigma_{xz}/\dot{\gamma}$, we used the Lees–Edwards formulation of SELM [8] in the quasi-steady regime discussed in sections 2.4 and 2.2. The shear viscosity is estimated using a variant of the approach of Irving and Kirkwood [26]; see [8]. As the polymeric network deforms under the shear, the interpolymer bonds break, and the material transitions from a gel-like material to a complex fluid. The contributions to the non-Newtonian shear viscosity η_p during this progression are shown in Figure 11.

The time progression of the viscosity under shear exhibits roughly three stages. In the first stage, the polymer-network maintains its integrity. Contributions to the shear viscosity arise from stretching of the interpolymer and intrapolymer bonds. In the second stage, the interpolymer bonds of the polymer-network begin to break. The polymers are then free to align with the direction of shear, which results in relaxation of the intrapolymer bonds to their preferred rest-length. In the third stage, steady-state is reached with the contributions to the shear viscosity arising from thermal fluctuations that drive transient misalignments of the polymers with the direction of

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

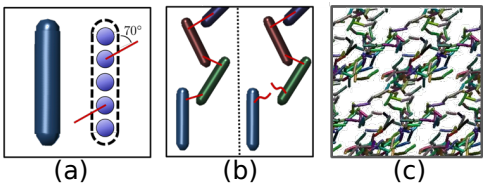

FIG. 10. *Polymeric material model.* (a) *Five-bead polymer chain with binding sites;* (b) *bonds can be irreversibly broken;* (c) *initial polymeric network.*

shear. For a more detailed discussion and specific parameters used in the simulations, see [8]. These results demonstrate how the SELM fluctuating hydrodynamics shear methods can be used to study the relationship between material microstructure and rheological properties.

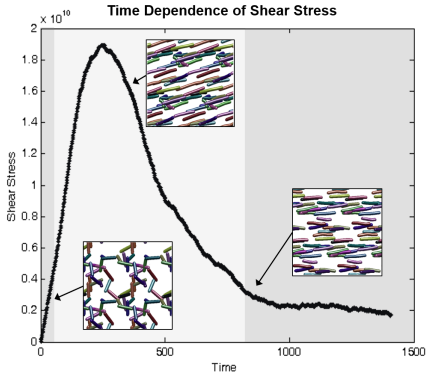

FIG. 11. *Polymer contributions to the shear viscosity.*

5.3. Lipid bilayer membrane. We use SELM to perform dynamic simulations of lipid bilayer membranes based on the implicit-solvent coarse-grained (IS-CG) model introduced for lipids by Cooke, Kremer, and Deserno [14, 44]. We consider self-assembled vesicles where the lipids are modeled by the free energy of interactions [14]

$$\Phi[\mathbf{X}] = \Phi_{rep} + \Phi_{bond} + \Phi_{bend} + \Phi_{attr}, \tag{19}$$

$$\phi_{rep}(r;b) = \begin{cases} 4\epsilon\left[(b/r)^{12} - (b/r)^{6} + \frac{1}{4}\right], & r \le r_c, \\ 0, & r > r_c, \end{cases}$$

$$\phi_{bond}(r) = -\frac{1}{2}k_{bond}r_\infty^2 \log\left[1 - (r/r_\infty)^2\right],$$

$$\phi_{bend}(r) = \frac{1}{2}k_{bend}\left(r - 4\sigma\right)^2,$$

$$\phi_{attr}(r) = \begin{cases} -\epsilon, & r < r_c, \\ -\epsilon\cos^2(\pi(r - r_c)/2w_c), & r_c \le r \le r_c + w_c, \\ 0, & r > r_c + w_c. \end{cases}$$

Each of the lipids consists of three beads that interact through the steric Weeks–Chandler–Andersen repulsion ϕ_{rep}, finitely extensible nonlinear elastic (FENE) bonds ϕ_{bond}, and bending energy ϕ_{bend}. The second and third lipids interact with other lipids

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

through a long-range attractive potential with a wide energy well near the minimum ϕ_{attr} that models the hydrophobic-hydrophilic effect [14]. The parameter b controls the steric lipid size, ϵ the energy scale of interaction, and w_c the width of the energy well of the attractive energy [14]. The IS-CG model can be used to self-assemble bilayer sheets and vesicles; see Figure 12. For more details, see [14, 44].

We perform simulations in the inertial regime using the SELM fluctuating hydrodynamics discussed in section 2.1. We make comparisons with Langevin dynamics with Stokes drag parametrized to model the same physical regime as SELM [44]. To investigate the lateral correlations within the bilayer and make comparisons, we consider

$$c_M = \langle \Delta_0 X \Delta_M X \rangle / \langle \Delta_0 X^2 \rangle . \tag{20}$$

This measures the correlations in the displacement of a reference lipid $\Delta_0 X$ over time δt with the displacement $\Delta_M X$ of the center-of-mass of a patch consisting of the M nearest neighbors, where $\Delta_M X = \frac{1}{M} \sum_{j=1}^{M} \Delta X^{I_j}$. Since the reference lipid is part of the patch, no significant correlations correspond to a decay $c_M \sim 1/M$ as M is made larger. The results of this correlation analysis is shown in Figure 13.

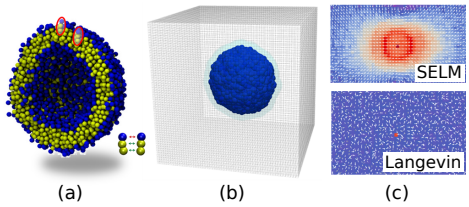

FIG. 12. *Vesicle lipid bilayer membrane model.* (a) *Self-assembled vesicle and three bead lipid model;* (b) *mesh of the SELM fluctuating hydrodynamics coupling the vesicle lipids;* (c) *lipid pair correlations.*

We can also consider the lipid pair correlations given by $\Psi(\mathbf{r}) = \langle \Delta_{\mathbf{r}} X \Delta_0 X^T \rangle$. The subscript $\mathbf{r}$ specifies the displacement vector from the center-of-mass of a reference lipid to the center-of-mass of a second lipid within the bilayer. By linear response theory, the vector field $\mathbf{w} = \Psi \mathbf{e}_1$ can be related to the flow of lipids within the bilayer that would occur in response to the force $\mathbf{e}_1 = (1, 0, 0)$. This is shown in Figure 12.

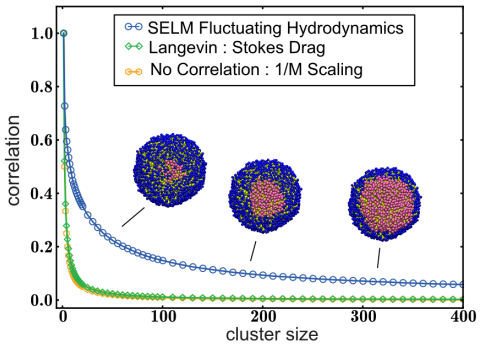

FIG. 13. *Correlations between a lipid's displacement and a cluster of nearest neighbors.*

We find that simulations with Langevin dynamics modeling the same physical regime as SELM are missing significant lateral correlations between the lipids. The

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

local non-momentum-conserving drag of Langevin greatly suppresses the collective motions of the lipids. In contrast, the SELM fluctuating hydrodynamics uses the same Stokes drag coefficient, but the momentum is conserved and instead transferred between the lipid degrees of freedom and the hydrodynamic fields modeling the solvent. This better preserves the collective dynamics and long-range spatial correlations mediated by the solvent, as seen in explicit solvent simulations [3]. For a more detailed discussion and further analysis, see [44]. These simulations demonstrate how the SELM fluctuating hydrodynamics methods can be used to extend IS-CG models to include important kinetic effects, facilitating their use in a wider range of applications.

6. Conclusions. We have developed a software package to facilitate the use of SELM fluctuating hydrodynamics methods. The package is interoperable with the widely used molecular dynamics package LAMMPS. This facilitates using SELM on existing models already set up in LAMMPS. The SELM fluctuating hydrodynamics methods provide ways to extend implicit-solvent coarse-grained (IS-CG) models to incorporate important kinetic effects, facilitating their use in a wider range of applications.

The SELM software can be downloaded from http://mango-selm.org.

REFERENCES

[1] D. J. Acheson, *Elementary Fluid Dynamics*, Oxford Applied Mathematics and Computing Science Series, Clarendon Press, Oxford University Press, New York, 1990.

[2] P. Allen and D. J. Tildesley, *Computer Simulation of Liquids*, Clarendon Press, Oxford University Press, New York, 1987.

[3] T. Apajalahti, P. Niemela, P. N. Govindan, M. S. Miettinen, E. Salonen, S.-J. Marrink, and I. Vattulainen, *Concerted diffusion of lipids in raft-like membranes*, Faraday Discuss., 144 (2010), pp. 411–430.

[4] P. J. Atzberger, *A note on the correspondence of an immersed boundary method incorporating thermal fluctuations with Stokesian-Brownian dynamics*, Phys. D, 226 (2007), pp. 144–150.

[5] P. J. Atzberger, P. R. Kramer, and C. S. Peskin, *A stochastic immersed boundary method for fluid-structure dynamics at microscopic length scales*, J. Comput. Phys., 224 (2007), pp. 1255–1292.

[6] P. J. Atzberger and Y. Wang, *Fluctuating Hydrodynamic Methods for Fluid-Structure Interactions in Confined Channel Geometries*, preprint, University of California Santa Barbara, Santa Barbara, CA, 2014; http://atzberger.org.

[7] P. J. Atzberger, *Stochastic Eulerian Lagrangian methods for fluid-structure interactions with thermal fluctuations*, J. Comput. Phys., 230 (2011), pp. 2821–2837.

[8] P. J. Atzberger, *Incorporating shear into stochastic Eulerian Lagrangian methods for rheological studies of complex fluids and soft materials*, Phys. D, to appear.

[9] M. Bai, A. R. Missel, W. S. Klug, and A. J. Levine, *The mechanics and affine-nonaffine transition in polydisperse semiflexible networks*, Soft Matter, 7 (2011), pp. 907–914.

[10] G. Booch, *Best of Booch: Designing Strategies for Object Technology*, SIGS Reference Library, Cambridge University Press, Cambridge, UK, 1997.

[11] J. F. Brady and G. Bossis, *Stokesian dynamics*, in Annual Review of Fluid Mechanics, Annu. Rev. Fluid Mech. 20, Annual Reviews, Palo Alto, CA, 1988, pp. 111–157.

[12] G. Brannigan, L. Lin, and F. Brown, *Implicit solvent simulation models for biomembranes*, European Biophys. J., 35 (2006), pp. 104–124, doi:10.1007/s00249-005-0013-y.

[13] A. J. Chorin, *Numerical solution of Navier-Stokes equations*, Math. Comp., 22 (1968), pp. 745–762.

[14] I. R. Cooke, K. Kremer, and M. Deserno, *Tunable generic model for fluid bilayer membranes*, Phys. Rev. E, 72 (2005), 011506.

[15] J. G. De La Torre and V. A. Bloomfield, *Hydrodynamic properties of macromolecular complexes. I. Translation*, Biopolymers, 16 (1977), pp. 1747–1763.

[16] J. M. Drouffe, A. C. Maggs, and S. Leibler, *Computer simulations of self-assembled membranes*, Science, 254 (1991), pp. 1353–1356.

[17] R. Johnson, J. Vlissides, E. Gamma, and R. Helm, *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison–Wesley Professional, Boston, MA, 1994.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[18] D. L. Ermak and J. A. McCammon, *Brownian dynamics with hydrodynamic interactions*, J. Chem. Phys., 69 (1978), pp. 1352–1360.
[19] O. Farago, *"Water-free" computer model for fluid bilayer membranes*, J. Chem. Phys., 119 (2003), pp. 596–605.
[20] P. J. Flory, *Principles of Polymer Chemistry*, Cornell University Press, Ithaca, NY, 1953.
[21] D. Frenkel and B. Smit, *Molecular dynamics simulations*, in Understanding Molecular Simulation, 2nd ed., D. Frenkel and B. Smit, eds., Academic Press, San Diego, CA, 2002, pp. 63–107.
[22] C. W. Gardiner, *Handbook of Stochastic Methods*, Springer Ser. Synergetics 13, Springer, Berlin, 1985.
[23] R. Goetz and R. Lipowsky, *Computer simulations of bilayer membranes: Self-assembly and interfacial tension*, J. Chem. Phys., 108 (1998), pp. 7397–7409.
[24] J. Gosling and H. McGilton, *The Java Language Environment: A White Paper*, Sun Microsystems, Santa Clara, CA, 1996.
[25] D. A. Head, A. J. Levine, and F. C. MacKintosh, *Deformation of cross-linked semiflexible polymer networks*, Phys. Rev. Lett., 91 (2003), 108102.
[26] J. H. Irving and J. G. Kirkwood, *The statistical mechanical theory of transport processes. IV. The equations of hydrodynamics*, J. Chem. Phys., 18 (1950), pp. 817–829.
[27] P. E. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations*, Springer-Verlag, New York, 1992.
[28] A. W. Lees and S. F. Edwards, *The computer study of transport processes under extreme conditions*, J. Phys. C Solid State Phys., 5 (1972), 1921.
[29] E. H. Lieb and M. Loss, *Analysis*, 2nd ed., AMS, Providence, RI, 2001.
[30] S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. de Vries, *The martini force field: Coarse grained model for biomolecular simulations*, J. Phys. Chem. B, 111 (2007), pp. 7812–7824.
[31] B. Oksendal, *Stochastic Differential Equations: An Introduction*, Springer, New York, 2000.
[32] S. Pedroni and N. Rappin, *Jython Essentials*, O'Reilly Media, Sebastopol, CA, 2002; http://www.jython.org/.
[33] F. Perrin, *Mouvement Brownien d'un ellipsoide* (II). *Rotation libre et dépolarisation des fluorescences. Translation et diffusion de molécules ellipsoidales*, J. Phys. Rad., 7 (1936), pp. 1–11.
[34] C. S. Peskin, *The immersed boundary method*, Acta Numer., 11 (2002), pp. 1–39.
[35] S. Plimpton, *Fast parallel algorithms for short-range molecular dynamics*, J. Comput. Phys., 117 (1995), pp. 1–19.
[36] P. Plunkett, J. Hu, C. Siefert, and P. J. Atzberger, *Spatially adaptive stochastic methods for fluid-structure interactions subject to thermal fluctuations in domains with complex geometries*, J. Comput. Phys., 277 (2014), pp. 121–137.
[37] L. E. Reichl, *A Modern Course in Statistical Physics*, John Wiley and Sons, New York, 1998.
[38] B. J. Reynwar, G. Illya, V. A. Harmandaris, M. M. Muller, K. Kremer, and M. Deserno, *Aggregation and vesiculation of membrane proteins by curvature-mediated interactions*, Nature, 447 (2007), pp. 461–464.
[39] H. Royden, *Real Analysis*, Simon & Schuster, Delran, NJ, 1988.
[40] M. Rubinstein and S. Panyukov, *Elasticity of polymer networks*, Macromolecules, 35 (2002), pp. 6670–6686.
[41] B. Smit, K. Esselink, P. A. J. Hilbers, N. M. Van Os, L. A. M. Rupert, and I. Szleifer, *Computer simulations of surfactant self-assembly*, Langmuir, 9 (1993), pp. 9–11.
[42] G. Tabak and P. J. Atzberger, *Stochastic reductions for inertial fluid-structure interactions subject to thermal fluctuations*, SIAM J. Appl. Math., 75 (2015), pp. 1884–1914, doi:10.1137/15M1019088.
[43] L. Verlet, *Computer "experiments" on classical fluids.* I. *Thermodynamical properties of Lennard-Jones molecules*, Phys. Rev., 159 (1967), pp. 98–103.
[44] Y. Wang, J. K. Sigurdsson, E. Brandt, and P. J. Atzberger, *Dynamic implicit-solvent coarse-grained models of lipid bilayer membranes: Fluctuating hydrodynamics thermostat*, Phys. Rev. E, 88 (2013), 023301.
[45] H. Yamakawa, *Transport properties of polymer chains in dilute solution: Hydrodynamic interaction*, J. Chem. Phys., 53 (1970), pp. 436–443.

Downloaded 04/05/21 to 128.122.149.96. Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/page/terms

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.