

RESEARCH ARTICLE | APRIL 17 2023

Computing hydrodynamic interactions in confined doubly periodic geometries in linear time

Aref Hashemi ; Raúl P. Peláez ; Sachin Natesh ; Brennan Sprinkle ; Ondrej Maxian ;
Zecheng Gan ; Aleksandar Donev  



J. Chem. Phys. 158, 154101 (2023)

<https://doi.org/10.1063/5.0141371>



CrossMark



The Journal of Chemical Physics

Special Topic: Adhesion and Friction

Submit Today!

Computing hydrodynamic interactions in confined doubly periodic geometries in linear time

Cite as: J. Chem. Phys. 158, 154101 (2023); doi: 10.1063/5.0141371

Submitted: 5 January 2023 • Accepted: 20 March 2023 •

Published Online: 17 April 2023



Aref Hashemi,^{1,a)} Raúl P. Peláez,^{1,2} Sachin Natesh,^{1,3} Brennan Sprinkle,^{1,4,b)} Ondrej Maxian,¹ Zecheng Gan,^{1,5} and Aleksandar Donev^{1,c)}

AFFILIATIONS

¹Courant Institute, New York University, New York, New York 10012, USA

²Departamento Física Teórica de la Materia Condensada, Universidad Autónoma de Madrid, Madrid, Spain

³Department of Applied Mathematics, University of Colorado Boulder, Boulder, Colorado 80309, USA

⁴Department of Applied Mathematics and Statistics, Colorado School of Mines, Golden, Colorado 80401, USA

⁵Department of Mathematics, The Hong Kong University of Science and Technology, Hong Kong, China

^{a)}Electronic mail: aref@cims.nyu.edu

^{b)}Electronic mail: bsprinkl@mines.edu

^{c)}Author to whom correspondence should be addressed: donev@courant.nyu.edu

ABSTRACT

We develop a linearly scaling variant of the force coupling method [K. Yeo and M. R. Maxey, J. Fluid Mech. **649**, 205–231 (2010)] for computing hydrodynamic interactions among particles confined to a doubly periodic geometry with either a single bottom wall or two walls (slit channel) in the aperiodic direction. Our spectrally accurate Stokes solver uses the fast Fourier transform in the periodic xy plane and Chebyshev polynomials in the aperiodic z direction normal to the wall(s). We decompose the problem into two problems. The first is a doubly periodic subproblem in the presence of particles (source terms) with free-space boundary conditions in the z direction, which we solve by borrowing ideas from a recent method for rapid evaluation of electrostatic interactions in doubly periodic geometries [Maxian *et al.*, J. Chem. Phys. **154**, 204107 (2021)]. The second is a correction subproblem to impose the boundary conditions on the wall(s). Instead of the traditional Gaussian kernel, we use the exponential of a semicircle kernel to model the source terms (body force) due to the presence of particles and provide optimum values for the kernel parameters that ensure a given hydrodynamic radius with at least two digits of accuracy and rotational and translational invariance. The computation time of our solver, which is implemented in graphical processing units, scales linearly with the number of particles, and allows computations with about a million particles in less than a second for a sedimented layer of colloidal microrollers. We find that in a slit channel, a driven dense suspension of microrollers maintains the same two-layer structure as above a single wall, but moves at a substantially lower collective speed due to increased confinement.

Published under an exclusive license by AIP Publishing. <https://doi.org/10.1063/5.0141371>

I. INTRODUCTION

The development of more efficient, accurate, and scalable methods for suspensions of rigid and flexible particles in Stokes flow remains a key challenge in soft condensed matter physics and chemical engineering. In addition to the long-ranged nature of hydrodynamic interactions, the inclusion of Brownian motion and the presence of confining boundaries pose particular difficulties. While confining boundaries partially screen the hydrodynamic interac-

tions, they continue to decay algebraically rather than exponentially¹ and must be captured to resolve particle dynamics.

A key component of all computational methods for Stokes flow is the efficient evaluation of the action of the singular or regularized Green's function for Stokes flow among a large number of particles. Specifically, given forces (and sometimes also torques) on many *source points*, the goal is to compute the resulting (linear and sometimes also angular) velocities on many *target points*, usually the same as the source points. While boundary integral

methods typically use the singular Green's function, most computational methods used for large-scale suspensions are based on a regularized Green's function. There are four popular regularizations: the Rotne–Prager–Yamakawa (RPY)^{2,3} tensor, regularized Stokeslets,⁴ immersed boundary (IB) kernels,⁵ and the Force Coupling Method (FCM) kernel.^{6–10} The RPY, IB, and FCM kernels regularize (smooth) the singularity at both the source and target; this is crucial to maintain the symmetric positive definite (SPD) nature of the hydrodynamic mobility matrix, as is necessary for Brownian Dynamics (BD) methods.

Here, we develop a fast method for Stokes suspensions in *doubly periodic* (DP) geometries with one or two confining walls in the aperiodic direction. Our method is closely related to the force coupling method of Yeo and Maxey,⁸ but with several important differences that increase both the flexibility and the efficiency of the approach. In brief, we employ a non-Gaussian envelope function¹¹ that allows accurate computations with far fewer grid cells per particle, as in immersed boundary methods (IBM),¹² and also develop a novel spectral Stokes solver based on a recent work by some of us on fast methods for electrostatics in doubly periodic geometries.¹³

The hydrodynamic interaction between two distinct spherical particles or *blobs* of radius a at positions $\mathbf{r}^{(1)}$ and $\mathbf{r}^{(2)}$ can be captured by a 3×3 mobility tensor that gives the velocity of one of the particles for a given force acting on the other. This hydrodynamic mobility tensor can be approximated in the far field as

$$\mathcal{O}(\mathbf{r}^{(1)}, \mathbf{r}^{(2)}) = \int \delta_a(\mathbf{r}^{(1)} - \mathbf{r}') \mathbb{G}(\mathbf{r}', \mathbf{r}'') \times \delta_a(\mathbf{r}^{(2)} - \mathbf{r}'') d\mathbf{r}' d\mathbf{r}'', \quad (1)$$

where $\mathbb{G}$ is Green's function for Stokes flow with the specified boundary conditions. Here, $\delta_a(\mathbf{r})$ is a regularized “delta function” or *envelope function*⁶ that is typically radially symmetric, $\delta_a(\mathbf{r}) \equiv \delta_a(r)$, when the particles are sufficiently far from boundaries. It is important to point out that the blobs do not have to represent actual physical spherical colloids; one can build physical particles as a collection of blobs, including non-spherical rigid (passive or active) particles,^{14–18} which we have termed as *rigid multiblobs*, or flexible particles, such as semiflexible fibers.^{19,20} If the blobs represent actual spherical colloids, additional near-field lubrication corrections can be added to improve upon the far-field accuracy;^{10,15,21–23} this requires torques in addition to forces, as we do in Sec. II.

Regardless of the context, a key task is to evaluate for N blobs their far-field velocity $\mathbf{U}$ from the applied forces $\mathbf{F}$ through the action of the *mobility matrix* $\mathcal{M}$, $\mathbf{U} = \mathcal{M}\mathbf{F}$,

$$\forall i: \quad \mathbf{U}^{(i)} = \sum_{j=1}^N \mathcal{O}(\mathbf{r}^{(i)}, \mathbf{r}^{(j)}) \mathbf{F}^{(j)}, \quad (2)$$

in time linear in the number of blobs N . Note that the form of (1) guarantees that the mobility matrix is SPD by construction since $\mathbb{G}$ is an SPD kernel, and the regularization is applied both at the source and the target.

The specific choice of δ_a as a delta function on the surface of a sphere of radius a , $\delta_a(\mathbf{r}) = (4\pi a^2)^{-1} \delta(r - a)$, results in the widely used Rotne–Prager–Yamakawa (RPY) tensor.^{24–26} When the

kernels do not overlap the boundaries, one can transform (1) into a differential form by employing the Faxén differential operator

$$\mathcal{O}(\mathbf{r}^{(1)}, \mathbf{r}^{(2)}) \approx \left(\mathbf{I} + \frac{a^2}{6} \nabla_{\mathbf{r}'}^2 \right) \left(\mathbf{I} + \frac{a^2}{6} \nabla_{\mathbf{r}''}^2 \right) \mathbb{G}(\mathbf{r}', \mathbf{r}'') \Big|_{\mathbf{r}'=\mathbf{r}^{(1)}, \mathbf{r}''=\mathbf{r}^{(2)}}. \quad (3)$$

This analytical simplification allows for explicit evaluation of the RPY kernel, not just in an unbounded domain but also in a half-space above a no-slip wall²⁴ because for a single no-slip boundary there is a relatively simple image construction for $\mathbb{G}$ due to Blake.²⁷ This image construction has also enabled the development of fast methods for evaluating (2) for a single wall, based on either (flexible periodicity) fast multipole method^{28,29} or the fast Fourier transform (FFT).^{30,31} For triply periodic (TP) domains, the Positively Split Ewald (PSE) method²⁶ for evaluating (2) (and also generating Brownian velocities) for the RPY kernel provides the basis for fast Stokesian dynamics.¹⁵ However, it remains a challenge to construct a similarly efficient method for confined suspensions since PSE uses Fourier representations in a key way for each component of the method (Stokes solver, generating Brownian increments, and Ewald splitting).

In principle, greater flexibility with respect to boundary conditions can be achieved by replacing analytical Green's functions with a grid-based Stokes solver.³² This requires replacing the singular surface delta function form of $\delta_a(\mathbf{r})$ with a smooth function that can be resolved on a grid. In the FCM, a Gaussian envelope function δ_a is used, which allows for analytical calculation of $\mathcal{O}$ in an unbounded domain;^{6,7} in numerical methods, the Gaussian is truncated. By contrast, in the IBM, δ_a is a discrete grid function specifically constructed to maximize grid invariance,^{5,12,17} and the double convolution in (1) is discrete. This makes all IBM results grid- and solver-dependent, without a direct continuum limit.

We develop a method that combines favorable features of FCM and IBM. Namely, we maintain the continuum representation (1) from FCM; however, we do not use a Gaussian envelope but rather use the “exponential of a semi-circle” (ES) kernel proposed by Barnett for its efficiency in the context of non-uniform FFTs.¹¹ Not only does this kernel allow for greater flexibility in tuning the hydrodynamic radius of the particles³³ but also allows us to use many fewer grid cells per particle than for a Gaussian kernel, comparable to the IBM, while still solving the continuum equations to several digits of accuracy. Specifically, for about two–three digits of accuracy, we need four grid cells per particle (in each dimension) if only translational velocity is required, and five or six if rotational velocities are also required. The grid independence of the results makes them transferable and allows us to separate the Stokes solver from the physics (i.e., the problem has a solution without a discretization). We find that $\mathcal{O}$ computed with the ES and Gaussian kernels are indistinguishable, in practice. We employ an image construction following Yeo and Maxey⁸ (also used in the IBM^{17,34}) to generalize (1) to the relevant-in-practice case when some of the blob kernels overlap the boundaries; this is considerably harder to do for the RPY kernel and has required *ad hoc* fixes in past work by some of us.³⁵

For TP domains, one can easily solve the Stokes equations spectrally in Fourier space, but this is considerably harder when boundaries are present. In Sec. III, we use some of the ideas applied to the Poisson equation in Ref. 13 to develop a Stokes solver for DP

geometries that are unbounded or confined by one or two boundaries in the aperiodic direction. Specifically, we focus on *bottom wall* (BW) DP geometries, with a single wall at $z = 0$, and *slit channel* (SC) DP geometry, confined by two walls at $z = 0$ and $z = H$, but the Stokes solver can handle more flexible boundary conditions in the aperiodic direction. Our novel fluid solver makes it possible to handle geometries that are partially unbounded in one direction, unlike in existing IBM or FCM implementations based on more traditional grid solvers, such as finite differences or finite elements.³⁶ Our fluid solver has the additional advantage that its implementation requires only calls to the three-dimensional (3D) FFT with an oversampling factor of 2 in the aperiodic direction, combined with trivially parallelizable one-dimensional pentadiagonal boundary value solvers in the z direction. We use this to implement the method on graphical processing units, achieving linear scaling up to as many as one million particles, with scaling constants much better than the existing fast methods²⁹ for the types of problems we study here. It is important to note that Srinivasan and Tornberg have recently developed a sophisticated method based on 3D FFTs for the singular Stokes Green's function with a bottom wall geometry (using images) and *flexible periodicity*.^{30,31} Our approach is different and specialized to doubly periodic geometries, is conceptually simpler, and affords flexibility in the boundary conditions in the unbounded direction. It is beyond the scope of this work to give a thorough comparison of the different approaches.

In Sec. III, we also study how to choose the parameters of the ES kernel to achieve the desired accuracy and effective blob hydrodynamic radius with the coarsest possible solver grid. In Sec. IV, we perform several validation tests examining the self and pair mobility of particles in BW and SC geometries and show that the method produces results in agreement with existing theoretical or numerical predictions. In Sec. V, we demonstrate that it is possible to generate stochastic (Brownian) particle displacements with covariance proportional to $\mathcal{M}$ using the Lanczos method^{37,38} in a modest number of iterations independent of the number of particles, for both BW and SC geometries. This allows us to replace the core hydrodynamic routines used in the Stokesian dynamics method developed in Ref. 22 and the rigid multiblob BD methods developed in Ref. 18 with new linear-scaling implementations that are substantially more efficient for a sufficiently large number of particles.

These new computational developments enable us to perform larger-scale studies of the dynamics of confined microroller suspensions than previously feasible (see Sec. V). After validating that the results presented by some of us in Ref. 22 are free of finite-size effects, we explore what a top wall does to a driven dense suspension of colloidal microrollers. We find that while the suspension maintains a two-layer structure above a single bottom wall, in a slit channel, the collective velocity is substantially reduced due to the increased confinement, as expected.

II. MODEL FORMULATION

We develop a method to solve the Stokes equations

$$\eta \nabla^2 \mathbf{u} - \nabla p = -\mathbf{f}, \quad (4)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (5)$$

in an xy doubly periodic domain of size $[x \ y] \in [-L_x, L_x] \times [-L_y, L_y]$,³⁹ with $z \in [0, \infty)$; we refer to this as a *bottom wall* (BW) geometry. We will return later to the case of a *slit channel* (SC) geometry with two walls at $z = 0$ and $z = H$, for which $z \in [0, H]$. Here, $\mathbf{u} = [u \ v \ w]^T$ is the velocity field of the fluid, p the pressure, η the viscosity, and the body force $\mathbf{f} = [f \ g \ h]^T$ represents particles. To close the problem, we impose tangential slip boundary conditions (BCs) at $z = 0$, i.e.,

$$\mathbf{u}|_{z=0} = \mathbf{u}^{\text{wall}} = [u^{\text{wall}} \ v^{\text{wall}} \ 0]^T, \quad (6)$$

where $\mathbf{u}^{\text{wall}}(x, y)$ is a smooth function giving a prescribed slip velocity ($\mathbf{u}^{\text{wall}} = \mathbf{0}$ for no-slip boundaries) along the wall (e.g., electrophoretic slip), and we require boundedness of $\mathbf{u}$ as $z \rightarrow \infty$. For all of the results and tests presented in this study, we use $\mathbf{u}^{\text{wall}} = \mathbf{0}$.

The source term $\mathbf{f}$ is restricted to the domain limits (i.e., $\mathbf{f}(z \notin [0, H]) = \mathbf{0}$) and consists of regularized monopoles and dipoles with envelope functions Δ_M and Δ_D centered around particles or blobs, i.e.,

$$\mathbf{f}(\mathbf{x}) = \sum_{j=1}^N \left[\mathbf{F}^{(j)} \Delta_M(\mathbf{x} - \mathbf{r}^{(j)}) + \frac{1}{2} \nabla \times (\boldsymbol{\tau}^{(j)} \Delta_D(\mathbf{x} - \mathbf{r}^{(j)})) \right], \quad (7)$$

where $\mathbf{x} = [x \ y \ z]^T$ and $\mathbf{r}^{(j)} = [x^{(j)} \ y^{(j)} \ z^{(j)}]^T$ is the j th particle's location with $j = 1, \dots, N$, and $\mathbf{F}^{(j)}$ and $\boldsymbol{\tau}^{(j)}$ are the force and torque of the j th particle. Here, Δ_M and Δ_D are compactly supported, smooth, regularized delta functions called kernels in the immersed boundary method (IBM), or envelopes in the Fast Coupling Method (FCM). In the classical FCM, the envelopes are (truncated) Gaussians,

$$\Delta_{M/D}(\mathbf{x}) = \frac{1}{\sqrt{8\pi^3 g_{M/D}^6}} \exp\left(-\frac{\|\mathbf{x}\|^2}{2g_{M/D}^2}\right), \quad (8)$$

where $g_M = R_h/\sqrt{\pi}$ and $g_D = R_h/(6\sqrt{\pi})^{1/3}$, with R_h as the effective hydrodynamic radius of a particle/blob.

If a particle is closer to the wall than a distance z_{im} so that its kernel overlaps the wall, we add the negative of the particle's envelope centered at its image point about the wall. This ensures that the force/torque decays to zero as a particle approaches the wall. As a result, if $z^{(j)} < z_{\text{im}}$, we replace the envelopes Δ_M and Δ_D in (7) with

$$\Delta_{M/D}^W(\mathbf{x} - \mathbf{r}^{(j)}) = \Delta_{M/D}(\mathbf{x} - \mathbf{r}^{(j)}) - \Delta_{M/D}(\mathbf{x} - \mathbf{r}_{\text{im}}^{(j)}), \quad (9)$$

where $\mathbf{r}_{\text{im}}^{(j)} = \mathbf{r}^{(j)} - 2\hat{\mathbf{e}}_z(\hat{\mathbf{e}}_z \cdot \mathbf{r}^{(j)})$ is the particle's point of reflection about the bottom wall, and $\hat{\mathbf{e}}_z = [0 \ 0 \ 1]^T$. In the FCM, Yeo and Maxey⁸ put a positive sign in (9) for Δ_D to ensure that the angular velocity of a particle in simple shear flow is unaffected by the proximity to the wall. We put a negative sign to ensure that all components of the particle mobility go to zero at the wall. This is a modeling choice, and it is important to realize that (9) is not an exact image construction for a no-slip wall.³⁰ For a free-slip boundary, such as a gas-liquid interface, it is straightforward to make an exact image construction; see, for example, Appendix A in Ref. 40.

Our goal is to compute the motion of each of the particles for dynamic simulations. Suppose we have solved the Stokes equations

for the fluid velocity $\mathbf{u}$; then, the linear and angular velocities of the particles, denoted as $\mathbf{U}^{(j)}$ and $\mathbf{\Omega}^{(j)}$, can be obtained through the following volume integrals over the envelopes:

$$\mathbf{U}^{(j)} = \int_{\mathbf{x}} \mathbf{u}(\mathbf{x}) \Delta_M(\mathbf{x} - \mathbf{r}^{(j)}) d\mathbf{x}, \quad (10)$$

$$\mathbf{\Omega}^{(j)} = \frac{1}{2} \int_{\mathbf{x}} (\nabla \times \mathbf{u}(\mathbf{x})) \Delta_D(\mathbf{x} - \mathbf{r}^{(j)}) d\mathbf{x}, \quad (11)$$

with the understanding that $\Delta_{M/D}$ get replaced by $\Delta_{M/D}^W$ for particles close to a wall.

In our own variant of FCM, we will not use a Gaussian kernel; rather, we will employ a compactly supported Exponential of a Semicircle (ES) kernel¹¹ optimized for numerical efficiency in spectral methods, as we discuss in detail in Sec. III E. Other kernels can be used as well, as long as their Fourier transform decays sufficiently fast in the wavenumber.

III. STOKES PROBLEM

We develop a doubly periodic+correction Stokes solver based on the observation that the original problems (4) and (5) can be separated into two subproblems, namely, the doubly periodic “DP” problem and the “correction” problem. For the DP subproblem, we keep the source terms $\mathbf{f}$ but there is no wall present (i.e., open BCs in z). The correction subproblem has walls but has no forcing, so it can be solved analytically using a plane-wave expansion. The total solution is then the sum of the solutions to the two subproblems.

Below, we detail our solution method for the bottom wall (BW) geometry with a no-slip wall. Appendix A summarizes the method for the slit channel (SC) geometry with somewhat more general partial slip BCs. The Stokes solver can also handle open boundaries in the z direction if the total force on the domain is zero. In order to uniquely define the self-mobility of a particle, one must therefore add a negative tail to the kernel to account for the *backflow* around the particle, similar to how in TP domains the $\mathbf{k} = [k_x \ k_y \ k_z]^T = \mathbf{0}$ component of the solution is set to zero, where $\mathbf{k}$ is the wavenumber. We defer the discussion of backflow in unbounded DP geometries to future work.

A. Doubly periodic Stokes solver

For the “DP” subproblem, we solve a doubly periodic Stokes problem for $\mathbf{u}_{DP}$ and p_{DP} ,

$$\eta \nabla^2 \mathbf{u}_{DP} - \nabla p_{DP} = -\mathbf{f}, \quad (12)$$

$$\nabla \cdot \mathbf{u}_{DP} = 0. \quad (13)$$

Here, the source term $\mathbf{f}$ is the same as (4), and the doubly periodic domain is $[x \ y \ z] \in [-L_x, L_x] \times [-L_y, L_y] \times (-\infty, +\infty)$ with *free-space* BCs in z , i.e., p_{DP} and $\mathbf{u}_{DP}$ are bounded as $z \rightarrow \pm\infty$ and there is no wall present. Note that in our spectral DP solver, we will solve $\mathbf{u}_{DP}$ in a bounded domain $[x \ y \ z] \in [-L_x, L_x] \times [-L_y, L_y] \times [0, H]$ for either bottom wall or slit channel geometry. For a slit channel, (9) ensures that $\mathbf{f}$ is smooth on and vanishes outside of $z \in [0, H]$, and for the bottom wall case, we will assume that it is possible to find a maximum height H so that $\mathbf{f}$ smoothly goes to zero for $z \notin [0, H]$.

Also, note that the system represented by (12) and (13) is not well-posed if $\int_{\mathbf{x}} \mathbf{f}(\mathbf{x}) d\mathbf{x} \neq \mathbf{0}$, where the integration is over the simulation box $[-L_x, L_x] \times [-L_y, L_y] \times [0, H]$; however, this problem manifests in our method only for $[k_x \ k_y]^T = \mathbf{0}$, which we will handle separately in Sec. III C. As we now show, the open boundary condition can be reduced to $z \in [0, H]$ through the Dirichlet-to-Neumann map, as proposed for the Poisson equation by Maxian *et al.*¹³

1. Solution for p_{DP}

First, take the divergence of (12) and use (13) to obtain

$$\nabla^2 p_{DP} = \nabla \cdot \mathbf{f}. \quad (14)$$

Fourier transforming in the xy domain yields

$$(\partial_z^2 - k^2) \hat{p}_{DP}(\mathbf{k}, z) = \left[\begin{array}{c} i\mathbf{k} \\ \partial_z \end{array} \right] \cdot \hat{\mathbf{f}}(\mathbf{k}, z). \quad (15)$$

Here, $\mathbf{k} = [k_x \ k_y]^T = [n_x \ n_y]^T \pi / L_{x/y}$ with $n_{x/y} \in \mathbb{Z}$, and $k = \|\mathbf{k}\|$. For $z \notin [0, H]$, $\mathbf{f} = \mathbf{0}$ and the general solution to (15) is

$$\hat{p}_{DP}(\mathbf{k}, z \geq H) = C_1 e^{-kz}, \quad \hat{p}_{DP}(\mathbf{k}, z \leq 0) = C_2 e^{kz}, \quad (16)$$

where we used the boundedness of $\hat{p}_{DP}$ as $z \rightarrow \pm\infty$. The solutions (16) imply simple Dirichlet-to-Neumann maps for $\hat{p}_{DP}$ at $z = 0^-$ and $z = H^+$. Since $\hat{p}_{DP}$ is continuously differentiable across the (artificial) computational boundaries at $z = 0$ and $z = H$, the same maps apply at $z = H^-$ and $z = 0^+$ as well, giving

$$(\partial_z + k) \hat{p}_{DP}(\mathbf{k}, z = H) = 0, \quad (\partial_z - k) \hat{p}_{DP}(\mathbf{k}, z = 0) = 0. \quad (17)$$

Therefore, we can solve the boundary value problem (BVP) (15) on $z \in [0, H]$ subject to the boundary conditions given by (17). This approach was proposed by some of us for the Poisson equation in Ref. 13.

2. Solution for $\mathbf{u}_{DP}$

To find the velocity, we take an xy Fourier transform of (12),

$$-\eta(\partial_z^2 - k^2) \hat{\mathbf{u}}_{DP}(\mathbf{k}, z) + \left[\begin{array}{c} i\mathbf{k} \\ \partial_z \end{array} \right] \hat{p}_{DP}(\mathbf{k}, z) = \hat{\mathbf{f}}(\mathbf{k}, z). \quad (18)$$

Again, $\mathbf{f} = \mathbf{0}$ outside domain $z \in [0, H]$, giving

$$\partial_z^2 \hat{\mathbf{u}}_{DP} - k^2 \hat{\mathbf{u}}_{DP} = \frac{i\mathbf{k}_x}{\eta} \hat{p}_{DP}, \quad (19)$$

$$\partial_z^2 \hat{w}_{DP} - k^2 \hat{w}_{DP} = \frac{1}{\eta} \partial_z \hat{p}_{DP}. \quad (20)$$

Using the solution for $\hat{p}_{DP}$ in (16), and enforcing boundedness of $\hat{\mathbf{u}}_{DP}$ as $z \rightarrow \pm\infty$, we find the general solution for $\mathbf{k} \neq \mathbf{0}$,

$$\hat{\mathbf{u}}_{DP}(\mathbf{k}, z \geq H) = A_x e^{-kz} - i \frac{k_x C_1}{2k\eta} z e^{-kz}, \quad (21a)$$

$$\hat{\mathbf{u}}_{DP}(\mathbf{k}, z \leq 0) = B_x e^{kz} + i \frac{k_x C_2}{2k\eta} z e^{kz}. \quad (21b)$$

Due to the xy symmetry of the problem, solutions to $\hat{v}_{DP}$ are analogous and obtained by replacing x with y in (21).

From (21), we get the following Dirichlet-to-Neumann maps at the boundaries:

$$\begin{aligned} (\partial_z + k) \begin{bmatrix} \hat{u}_{DP}(\mathbf{k}, H) \\ \hat{v}_{DP}(\mathbf{k}, H) \end{bmatrix} &= -\frac{i\mathbf{k}}{2k\eta} \hat{p}_{DP}(\mathbf{k}, H), \\ (\partial_z - k) \begin{bmatrix} \hat{u}_{DP}(\mathbf{k}, 0) \\ \hat{v}_{DP}(\mathbf{k}, 0) \end{bmatrix} &= \frac{i\mathbf{k}}{2k\eta} \hat{p}_{DP}(\mathbf{k}, 0). \end{aligned} \quad (22)$$

Similarly, the general solution for $\hat{w}_{DP}$ is

$$\hat{w}_{DP}(\mathbf{k}, z \geq H) = A_z e^{-kz} + \frac{C_1}{2\eta} z e^{-kz}, \quad (23a)$$

$$\hat{w}_{DP}(\mathbf{k}, z \leq 0) = B_z e^{kz} + \frac{C_2}{2\eta} z e^{kz}. \quad (23b)$$

which implies the following boundary conditions:

$$\begin{aligned} (\partial_z + k) \hat{w}_{DP}(\mathbf{k}, H) &= \frac{1}{2\eta} \hat{p}_{DP}(\mathbf{k}, H), \\ (\partial_z - k) \hat{w}_{DP}(\mathbf{k}, 0) &= \frac{1}{2\eta} \hat{p}_{DP}(\mathbf{k}, 0). \end{aligned} \quad (24)$$

We can solve the BVP problems (18) on $z \in [0, H]$ with boundary conditions (22) and (24) to find the velocity field using a spectral (Chebyshev) solver, as described in Sec. III D. It is important to note that solving the pressure and velocity BVPs only requires solving pentadiagonal linear systems.

B. Correction solve

The correction subproblem solves the following homogeneous Stokes problem for $\mathbf{u}_{corr}$ and p_{corr} :

$$\eta \nabla^2 \mathbf{u}_{corr} - \nabla p_{corr} = 0, \quad (25)$$

$$\nabla \cdot \mathbf{u}_{corr} = 0, \quad (26)$$

where the doubly periodic domain is $[x \ y \ z] \in [-L_x, L_x] \times [-L_y, L_y] \times [0, +\infty)$ and with slip on the bottom wall

$$\mathbf{u}_{corr}|_{z=0} = \mathbf{u}^{wall} - \mathbf{u}_{DP}|_{z=0}, \quad (27)$$

where $\mathbf{u}_{DP}$ is the solution to the DP problem. Note that, by linearity, it is clear that the sum of the solution to the two subproblems in the upper-half plane gives the solution to the original problem (4)–(6), i.e.,

$$\mathbf{u} = \mathbf{u}_{DP} + \mathbf{u}_{corr}, \quad \text{and} \quad p = p_{DP} + p_{corr}. \quad (28)$$

The correction subproblem (25)–(27) can be solved analytically.

1. Solution for p_{corr}

Taking the divergence of (25) we obtain a Laplace equation for pressure

$$\Delta p_{corr} = 0. \quad (29)$$

After a Fourier transform in xy , we get a 1D equation for each Fourier mode $\mathbf{k}$,

$$\partial_z^2 \hat{p}_{corr}(\mathbf{k}, z) - k^2 \hat{p}_{corr}(\mathbf{k}, z) = 0. \quad (30)$$

Using the boundedness of $\hat{p}_{corr}$ as $z \rightarrow \infty$, the general solution of (30) is

$$\hat{p}_{corr}(\mathbf{k}, z) = C_0 e^{-kz}. \quad (31)$$

2. Solution for $\mathbf{u}_{corr}$

Taking the Fourier transform of (25) in xy and using (31), we obtain

$$\eta(k^2 - \partial_z^2) \hat{\mathbf{u}}_{corr}(\mathbf{k}, z) = -ik_x C_0 e^{-kz}, \quad (32)$$

$$\eta(k^2 - \partial_z^2) \hat{w}_{corr}(\mathbf{k}, z) = k C_0 e^{-kz}, \quad (33)$$

with general solutions for $\mathbf{k} \neq \mathbf{0}$ given by

$$\hat{\mathbf{u}}_{corr}(\mathbf{k}, z) = -\frac{C_0 i k_x}{2\eta k} z e^{-kz} + C_x e^{-kz}, \quad (34)$$

$$\hat{w}_{corr}(\mathbf{k}, z) = \frac{C_0}{2\eta} z e^{-kz} + C_z e^{-kz}, \quad (35)$$

and analogously for $\hat{v}_{corr}$. To determine the unknown coefficients, we consider the boundary conditions on the wall. Let $\mathbf{u}_{corr}|_{z=0} = \mathbf{u}^{wall} - \mathbf{u}_{DP}|_{z=0} = [u_0(x, y) \ v_0(x, y) \ w_0(x, y)]^T$, i.e.,

$$\begin{aligned} \hat{\mathbf{u}}_{corr}(\mathbf{k}, 0) &= \hat{\mathbf{u}}_0(\mathbf{k}), \quad \hat{v}_{corr}(\mathbf{k}, 0) = \hat{v}_0(\mathbf{k}), \\ \hat{w}_{corr}(\mathbf{k}, 0) &= \hat{w}_0(\mathbf{k}). \end{aligned} \quad (36)$$

Substituting (34) and (35) into (36) gives three of the unknown coefficients

$$C_x = \hat{u}_0(\mathbf{k}), \quad C_y = \hat{v}_0(\mathbf{k}), \quad C_z = \hat{w}_0(\mathbf{k}). \quad (37)$$

To find the coefficient C_0 , we take an xy Fourier transform of (26),

$$ik_x \hat{\mathbf{u}}_{corr}(\mathbf{k}, z) + ik_y \hat{v}_{corr}(\mathbf{k}, z) + \partial_z \hat{w}_{corr}(\mathbf{k}, z) = 0. \quad (38)$$

Evaluating (38) at $z = 0$ and also using the boundary conditions (36), we obtain

$$\partial_z \hat{w}_{corr}(\mathbf{k}, z = 0) = -ik_x \hat{u}_0(\mathbf{k}) - ik_y \hat{v}_0(\mathbf{k}), \quad (39)$$

which coupled with (35) gives

$$C_0 = 2\eta(k \hat{w}_0(\mathbf{k}) - ik_x \hat{u}_0(\mathbf{k}) - ik_y \hat{v}_0(\mathbf{k})). \quad (40)$$

Substituting the coefficients back to (34) and (35), we obtain an explicit correction solution in xy Fourier space,

$$\hat{p}_{corr}(\mathbf{k}, z) = 2\eta(k \hat{w}_0(\mathbf{k}) - ik_x \hat{u}_0(\mathbf{k}) - ik_y \hat{v}_0(\mathbf{k})) e^{-kz}, \quad (41)$$

$$\begin{aligned} \hat{\mathbf{u}}_{corr}(\mathbf{k}, z) &= -\frac{k_x}{k} (ik \hat{w}_0(\mathbf{k}) + k_x \hat{u}_0(\mathbf{k}) + k_y \hat{v}_0(\mathbf{k})) z e^{-kz} \\ &\quad + \hat{\mathbf{u}}_0(\mathbf{k}) e^{-kz}, \end{aligned} \quad (42)$$

$$\hat{v}_{\text{corr}}(\mathbf{k}, z) = -\frac{k_y}{k} (ik\hat{u}_0(\mathbf{k}) + k_x\hat{u}_0(\mathbf{k}) + k_y\hat{v}_0(\mathbf{k}))ze^{-kz} + \hat{v}_0(\mathbf{k})e^{-kz}, \quad (43)$$

$$\hat{w}_{\text{corr}}(\mathbf{k}, z) = (k\hat{u}_0(\mathbf{k}) - ik_x\hat{u}_0(\mathbf{k}) - ik_y\hat{v}_0(\mathbf{k}))ze^{-kz} + \hat{w}_0(\mathbf{k})e^{-kz}. \quad (44)$$

C. Zero mode ($\mathbf{k} = \mathbf{0}$)

Sections III A. and III B provide the DP and correction solutions for $\mathbf{k} \neq \mathbf{0}$. We treat the $\mathbf{k} = \mathbf{0}$ mode here for the overall solution without separating it into subproblems.

To find $\hat{w}(\mathbf{0}, z)$, we use the continuity equation (5) in Fourier space for $k_x = k_y = 0$,

$$\partial_z \hat{w}(\mathbf{0}, z) = 0. \quad (45)$$

As a result, $\hat{w}(\mathbf{0}, z)$ is just a constant, and the no-slip boundary conditions at $z = 0$, given in Fourier space by

$$\hat{\mathbf{u}}(\mathbf{k}, 0) = [\hat{u}^{\text{wall}}(\mathbf{k}) \quad \hat{v}^{\text{wall}}(\mathbf{k}) \quad 0]^T, \quad (46)$$

indicate that

$$\hat{w}(\mathbf{0}, z) = 0. \quad (47)$$

The momentum equation (4) in Fourier space for $\mathbf{k} = \mathbf{0}$ can be expressed as

$$-\eta \partial_z^2 \hat{\mathbf{u}}(\mathbf{0}, z) + \begin{bmatrix} \mathbf{0} \\ \partial_z \end{bmatrix} \hat{\mathbf{p}}(\mathbf{0}, z) = \hat{\mathbf{f}}(\mathbf{0}, z) = \begin{bmatrix} \hat{f}(\mathbf{0}, z) \\ \hat{g}(\mathbf{0}, z) \\ \hat{h}(\mathbf{0}, z) \end{bmatrix}. \quad (48)$$

Using (47), the z component of (48) simplifies to

$$\partial_z \hat{p}(\mathbf{k} = \mathbf{0}, z) = \hat{h}(\mathbf{k} = \mathbf{0}, z), \quad (49)$$

which yields the zero mode solution for the pressure,

$$\hat{p}(\mathbf{0}, z) = \int_0^z \hat{h}(\mathbf{0}, z') dz', \quad (50)$$

where we set $\hat{p}(\mathbf{0}, 0) = 0$ as a convention.

To find the $\mathbf{k} = \mathbf{0}$ mode solution for $\hat{u}$ (or $\hat{v}$ due to symmetry in xy), note that the x component of (48) is

$$\partial_z^2 \hat{u}(\mathbf{0}, z) = -\frac{\hat{f}(\mathbf{0}, z)}{\eta}. \quad (51)$$

We have a no-slip BC at $z = 0$ given by (46). To find a boundary condition at $z = H$, note that for $z > H$, $\partial_z^2 \hat{u}(\mathbf{0}, z) = 0$, which along with the boundedness of velocity as $z \rightarrow \infty$, indicate that $\hat{u}(\mathbf{0}, z \geq H)$ is some constant. Since $\partial_z \hat{u}(\mathbf{0}, z)$ is continuous across $z = H$ and it is a constant for $z > H$, we conclude that $\partial_z \hat{u}(\mathbf{0}, H) = 0$. As a result, the $\mathbf{k} = \mathbf{0}$ mode of $\hat{u}$ can be computed by solving the BVP (51) with the following boundary conditions:

$$\hat{u}(\mathbf{0}, 0) = \hat{u}^{\text{wall}}(\mathbf{0}), \quad \partial_z \hat{u}(\mathbf{0}, H) = 0, \quad (52)$$

and analogously for $\hat{v}$.

D. Stokes solver

Our spectral doubly periodic Stokes solver is based on the method proposed for the Poisson equation in Ref. 13, where more details are provided. We use FFT in the doubly periodic xy plane and Chebyshev polynomials in the z direction to achieve spectral accuracy. The grid is uniform in the xy plane, with spacing $h_{x/y} = 2L_{x/y}/N_{x/y}$, where $N_{x/y}$ is the total number of points in each direction, and nonuniform in the z direction, with grid points placed on a type-2 Chebyshev grid, including the endpoints at $z = 0$ and $z = H$. The number of Chebyshev grid points, N_z , is chosen such that the coarsest spacing in the z direction (occurring at the midplane) is comparable to $h_{x/y}$ [see Eq. (108) in Ref. 13].

In principle, we could take a 2D FFT of (12) and (14) to find the discrete forms of (15) and (18), respectively, and solve the resulting BVPs in the z direction in Chebyshev space. Instead, following Maxian *et al.*,¹³ we take advantage of the Chebyshev compatibility with the Fourier transform⁴¹ and solve the problem in 3D Fourier–Chebyshev space.⁴² This allows us to use 3D FFTs in the implementation, which gives greater flexibility for optimizing parallel performance. We have implemented the method in CUDA for Graphical Processing Units (GPUs) in the UAMMD library,⁴³ using the NVIDIA cuFFT library. Both central processing unit (CPU) and GPU implementations are freely available at <https://github.com/stochasticHydroTools/DoublyPeriodicStokes>.

The following steps summarize our solution algorithm for the Stokes equation:

- (1) Spread the right-hand side (RHS) $\mathbf{f}$ of the momentum equation (12) on the grid using the ES kernel (cf. Sec. III E). Since the ES kernel is compactly supported, each particle only spreads to $\sim m^3$ grid points. In the z direction, this is only true if near the midplane; more than m points will be required near the boundaries where the Chebyshev grid is finer. When the kernel extends outside of the domain over a physical boundary,⁴⁴ we fold the contribution outside of the domain back into the domain using a negative mirror image [see (9)]. We will return to spreading torques after this list.
- (2) Take a 3D FFT of $\mathbf{f}$ to compute the Chebyshev coefficients of $\hat{\mathbf{f}}(\mathbf{k}, z)$. Then, use Chebyshev differentiation to compute the Chebyshev coefficients of $\partial_z \hat{\mathbf{f}}(\mathbf{k}, z)$.⁴²
- (3) For each $\mathbf{k} \neq \mathbf{0}$, solve the BVP (15) with boundary conditions (17) for the Chebyshev coefficients of $\hat{p}_{\text{DP}}(\mathbf{k}, z)$. Then, evaluate $\hat{p}(\mathbf{k}, z = 0/H)$ by direct summation to assemble boundary conditions (22) and (24), and solve the BVP (18) for the Chebyshev coefficients of $\hat{\mathbf{u}}_{\text{DP}}(\mathbf{k}, z)$. See Appendix A in Ref. 13 for details of the BVP solver, which is based on Ref. 45; this efficient and stable solver only requires solving pentadiagonal systems.
- (4) Evaluate, by direct summation, $\hat{\mathbf{u}}_{\text{DP}}(\mathbf{k}, z = 0)$ for the BW geometry and also $\hat{\mathbf{u}}_{\text{DP}}(\mathbf{k}, z = H)$ for the SC geometry,⁴⁶ and compute the RHSs of (36) and (A2), respectively. Evaluate the correction solutions, $\hat{p}_{\text{corr}}(\mathbf{k}, z)$ and $\hat{\mathbf{u}}_{\text{corr}}(\mathbf{k}, z)$ on the Chebyshev grid, using the analytical expressions (41)–(44) for BW and (A4)–(A6) for SC geometries.
- (5) Evaluate the $\mathbf{k} = \mathbf{0}$ mode solution for $\hat{p}(\mathbf{0}, z)$ using (50) and Chebyshev integration, and solve the BVP given by (51) and (52) for the BW and (A12) for the SC

- geometry to find $\hat{u}(\mathbf{0}, z)$, and analogously, $\hat{v}(\mathbf{0}, z)$. Set $\hat{w}(\mathbf{0}, z) = 0$.
- (6) For all $\mathbf{k}$, take a 1D FFT in the z direction to compute the Chebyshev coefficients of the correction and zero-mode solutions and add them to $\hat{p}_{DP}$ and $\hat{u}_{DP}$.
 - (7) Back transform to real space using a 3D iFFT to get $p(\mathbf{x})$ and $\mathbf{u}(\mathbf{x})$.
 - (8) Interpolate $\mathbf{u}(\mathbf{x})$ at the particle positions via (10) and trapezoidal+Chebyshev quadrature to find the linear velocity of the particles. For each particle, this requires the same grid values (but with additional z -dependent quadrature weights) as in the spreading in step 1.

When torques are also applied, they can be spread in two ways. The first way, which is more straightforward to implement but also more computationally expensive, is to compute the derivatives needed for the curl operator in Fourier–Chebyshev space; this requires an additional forward 3D FFT. Analogously, in the interpolation step, the curl of the velocity is first computed in Fourier–Chebyshev space and then transferred back to real space with an additional 3D iFFT. These additional FFTs can be avoided by instead using the derivative of the ES kernel to spread and interpolate velocity.^{31,47} We have employed both of these approaches to spread torques and interpolate velocity in a CPU-based implementation and confirm that both preserve the desired accuracy. Due to the higher implementation complexity of the second approach,⁴⁸ in the GPU code, we have presently implemented only the first way.

It is important to emphasize that all steps in the method scale linearly, or log-linearly, in either the number of particles or the number of grid points. The BVP solutions along the z direction require solving pentadiagonal linear systems (see Appendix A in Ref. 13), which makes that step need only a linear amount of memory and computational effort in N_z . Combined with the independence of the BVP solves for each $\mathbf{k}$, this makes the algorithm particularly well-suited for efficient execution on GPUs.

E. Exponential of a semicircle kernel (ES)

In order to reduce the costs associated with representing particles on the grid, we replace the Gaussian kernel traditionally used in the FCM formulation with the exponential of a semicircle (ES) kernel.^{11,49} Specifically, as in the IBM, we take $\Delta_{M/D}$ to be tensor products of a scalar kernel ϕ ,

$$\Delta_{M/D}(\mathbf{x}; \alpha_{M/D}, \beta_{M/D}) = \prod_{i=1}^3 \phi(x_i; \alpha_{M/D}, \beta_{M/D}), \quad (53)$$

where

$$\phi(z; \alpha, \beta) = \left(\int_{-\alpha}^{\alpha} e^{\beta \left(\sqrt{1 - \left(\frac{z}{\alpha} \right)^2} - 1 \right)} dz \right)^{-1} \times \begin{cases} e^{\beta \left(\sqrt{1 - \left(\frac{z}{\alpha} \right)^2} - 1 \right)}, & \left| \frac{z}{\alpha} \right| \leq 1 \\ 0, & \text{otherwise.} \end{cases} \quad (54)$$

Here, (α, β) are fixed parameters uniquely tied to the kernel, which is compactly supported on $[-\alpha, \alpha]$. If the kernel we use for spreading

and interpolation has a simple Fourier transform, we can determine the effective hydrodynamic radius R_h of a particle represented by such a kernel analytically. For example, in the seminal works on FCM applied to Stokes flow,^{6,7} the authors show that for Gaussian monopole and dipole envelopes given by (7) with standard deviations g_M and g_D , the length scale of the kernels is related to the particle radius through $g_M = R_h/\sqrt{\pi}$ and $g_D = R_h/(6\sqrt{\pi})^{1/3}$. However, the ES kernel does not have a simple analytical Fourier transform, so we will numerically determine the relationship between R_h and (α, β) .

Let us denote with m the number of grid cells in the support of the kernel in each dimension. For a regular grid with spacing h , $\alpha \leq \frac{1}{2}hm$ for the ES kernel, but we typically take $\alpha = \frac{1}{2}hm$. Kernels, such as the Peskin kernels used in IBM,⁵ have no free parameters to change R_h once m is chosen. In contrast, the ES kernel affords increased flexibility with the additional parameter β . So, particles of different radii can be represented with the same support α , but different β values. Moreover, the ES kernel has numerically compact support (to some tolerance) in real and Fourier space, and is, therefore, well suited for spectral discretizations of the Stokes problem. That is, the discrete RHS of the Stokes problem (4) will decay rapidly in Fourier space.

To find an appropriate value for β , we consider the mobility of an isolated sphere. A sphere of radius R_h will translate and rotate with linear and angular velocities $\mathbf{U}$ and $\mathbf{\Omega}$ in an unbounded space under the action of a force $\mathbf{F} = 6\pi\eta R_h \mathbf{U}$ and torque $\mathbf{\tau} = 8\pi\eta R_h^3 \mathbf{\Omega}$. For a TP domain with a cubic box of size L^3 , periodic corrections to Stokes' law can be expressed as an asymptotic expansion in R_h/L ,⁵⁰

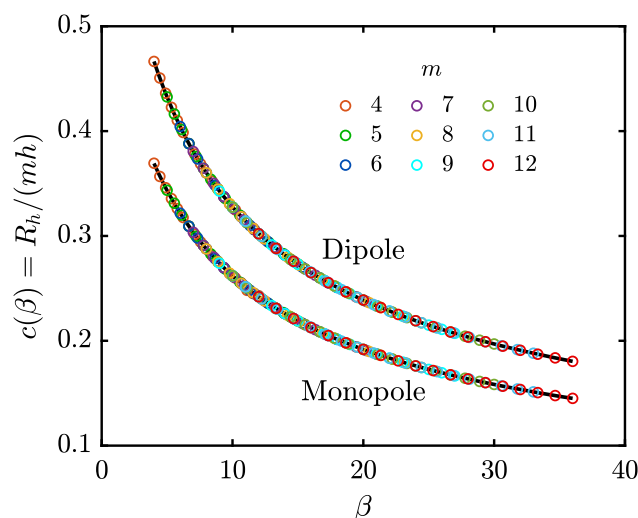


FIG. 1. Dimensionless hydrodynamic radius function $c(\beta)$ for the ES kernel for translation and rotation. For 30 particle positions distributed uniformly in one grid cell, and for several values of m and β , we extrapolate the computed radii to $L \rightarrow \infty$. Specifically, we take $h = 1$, $L \in \{60, 95, 130, 165, 200\}$, $m \in \{4, 5, 6, 7, 8, 9, 10, 11, 12\}$, and β/m values linearly spaced in $[1, 3]$. All of the data collapse on a master curve (depicted in solid black) that can be fit by a polynomial, up to a small error due to loss of translational invariance.

$$\mathbf{F} \approx \frac{6\pi\eta R_h \mathbf{U}}{1 - K_M \frac{R_h}{L}}, \quad \boldsymbol{\tau} \approx \frac{8\pi\eta R_h^3 \boldsymbol{\Omega}}{1 - K_D \left(\frac{R_h}{L}\right)^3}, \quad (55)$$

to leading order, where K_M and K_D are constants independent of R_h and L . Our numerical results indicate that the same form applies to doubly periodic domains as well, but with different constants; in this section, we use a TP system so that the grid spacing h is uniform.

By the scale invariance of the ES kernel under changing the unit of length, there exists a function $c(\beta)$ such that

$$R_h(\alpha, \beta) = 2\alpha c(\beta) = (hm) c(\beta). \quad (56)$$

To obtain $c(\beta)$ empirically, and determine an “optimal” β , we take a single particle in a TP domain and apply a unit force or torque on it, and measure its linear and angular velocities. If we keep the position of the particle (relative to the grid) and α fixed, we can use extrapolation to $L \rightarrow \infty$ based on (55) to obtain R_h . Due to numerical errors, R_h varies slightly based on the exact position of the particle relative to the grid, resulting in a numerical loss of translational invariance. In Fig. 1, we show $c(\beta)$ estimated from the mean $\langle R_h \rangle$, along with a polynomial fit. We emphasize that $c(\beta)$ is an intrinsic property of the ES kernel and is independent of the Stokes solver used, as long as the solver is sufficiently accurate. This is assured to about three digits by our use of a spectral solver for any $m \geq 4$; smaller m will have too few points per particle to resolve the envelope. The loss of translational invariance is purely numerical and gets worse for smaller m .

In Fig. 2, we evaluate percent errors in the extrapolated radii at each m and for each β in terms of a 95% confidence interval. That is, we report

$$\% \text{ error} = \frac{4\sigma(m)}{\langle R_h(m) \rangle} \times 100\%, \quad (57)$$

in which $\sigma(m)$ is the standard deviation from the mean radius $\langle R_h(m) \rangle$. We see that for the monopole, $m = 12$ and $\beta/m \approx 1.9$ gives the smallest %-error, while for the dipole, $m = 12$ and $\beta/m \approx 1.7$ are the optimal settings. From an extrapolation with these highly resolved settings, we find $K_M = 2.83$ and $K_D = 4.19$.⁵¹

Note, in practice and for the sake of efficiency, we use $m = 4, 5, 6$, for which $\beta/m \approx 1.75$ and $\beta/m \approx 1.6$ are optimal settings for the monopole and dipole kernels, respectively. This is to be compared to the optimal value $\beta/m \approx 2.7$ suggested by Barnett *et al.*¹¹ for the non-uniform FFT algorithm. The precise values of β and m for each particle species need to be selected by balancing several considerations. First, decreasing $R_h/h = c(\beta)m$ means fewer grid cells in the support of the particle kernel and, therefore, higher efficiency. However, translational invariance ought to be preserved to some target tolerance, requiring keeping β/m in a narrow range around the minima of error seen in Fig. 2. Finally, L/h must be an integer, preferably an FFT-friendly integer. The target tolerance for us is 2–3 digits of accuracy. A final challenge is that the hydrodynamic radius needs to be matched between the monopole and dipole terms. If we use $m = 4$ for the monopole, which is the smallest possible to ensure sufficient grid invariance, we find that we require $m \geq 5$ for

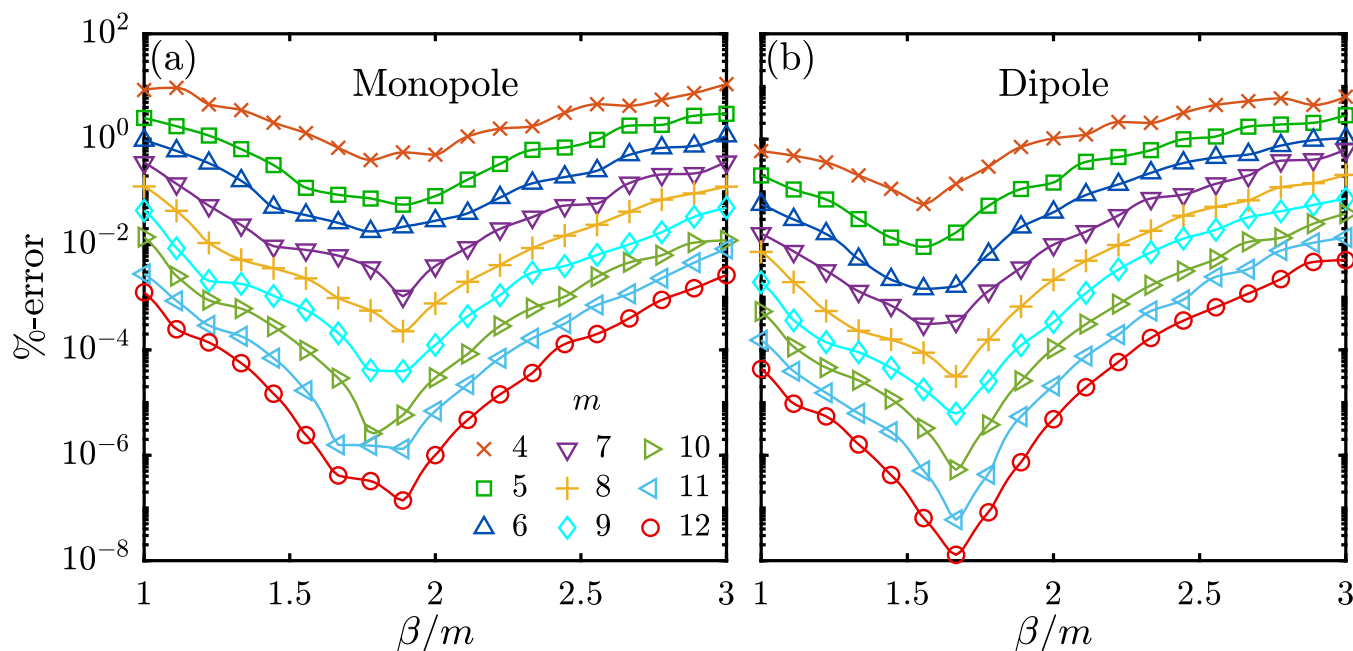


FIG. 2. Percent error (variability) in the effective hydrodynamic radius of a particle represented by the ES kernel as the particle moves relative to an underlying triply periodic (TP) uniform grid of spacing h . For a given width m , there is an optimal β/m that yields the smallest spread in the calculated effective radius (caused by the loss of translational invariance).

TABLE I. Optimal combinations of $m_M, m_D, \beta_M, \beta_D$ along with minimal errors in translational invariance (57) of the effective hydrodynamic radius R_h (equal for monopole and dipole up to 3 digits), when both force and torque are applied (M and D).

$m_M (= m_D)$	5	6
R_h/h	1.560	1.731
β_M/m_M	1.305	1.327
β_D/m_D	2.232	2.216
% error _M	0.90	0.15
% error _D	0.81	0.21

TABLE II. Same as Table I but only forces are applied (M).

m_M	4	5	6
R_h/h	1.205	1.344	1.554
β_M/m_M	1.785	1.886	1.714
% error _M	0.37	0.05	0.02

the dipole to match R_h and get acceptable accuracy. Table I summarizes the values we suggest for $m = 5$ and $m = 6$ (it is best to keep m the same for the monopole and dipole kernels). When only translational motion is required (e.g., with the rigid multiblob method¹⁷), $m = 4$ can be used; Table II summarizes the suggested values. Our public domain software release contains a script that suggests “good” values for m and β given L and R_h as inputs. When there are more than one particle species, the selection becomes more involved but still possible.³³

IV. SELF AND PAIR MOBILITIES

We validate our solver against theoretical predictions on basic mobility problems in the bottom wall (BW) and slit channel (SC) geometries with $N = 1$ or $N = 2$ particles. Specifically, for several particle configurations, we determine the mobility matrix $\mathcal{M} \in \mathbb{R}^{6N \times 6N}$, relating velocities to forces,

$$\begin{bmatrix} \mathbf{U} \\ \mathbf{\Omega} \end{bmatrix} = \begin{bmatrix} \mathcal{M}^{tt} & \mathcal{M}^{tr} \\ \mathcal{M}^{rt} & \mathcal{M}^{rr} \end{bmatrix} \begin{bmatrix} \mathbf{F} \\ \mathbf{\tau} \end{bmatrix}, \quad (58)$$

where $\mathcal{M}^{(-)} \in \mathbb{R}^{3N \times 3N}$ with superscripts t and r denoting the translational and rotational mobility components, respectively, and $\mathbf{F}, \mathbf{\tau} \in \mathbb{R}^{3N}$ and $\mathbf{U}, \mathbf{\Omega} \in \mathbb{R}^{3N}$ are the vectors of force and torque on the particles and the resulting translation and rotational velocities. We will refer to certain elements of $\mathcal{M}$ using the notation μ_{ab}^{cd} for diagonal blocks (self-mobility) or v_{ab}^{cd} for off-diagonal blocks (pair-mobility). This corresponds to prescribing a force or torque ($d = t$ or r) in the b direction ($b = x, y$, or z) on one particle, and measuring the linear or rotational velocity ($c = t$ or r) in the a direction of the second particle (or the same particle, in the case of self-mobility). For example, v_{yx}^{tr} corresponds to the linear velocity in the y direction of the second particle due to a torque about the x axis on the first particle, i.e., $U_y^{(2)} = v_{yx}^{tr} \tau_x^{(1)}$. Note that $v_{ab}^{cd} = \mu_{ab}^{cd}$ if the two particles have the same position.

In all tests, we use a box with dimensions $L_x/L_y = L = 76.8R_h$, $H = 19.2R_h$ to allow direct comparison with prior IBM results.¹⁷ The components of $\mathcal{M}^{tt}$, $\mathcal{M}^{tr/r}$, and $\mathcal{M}^{rr}$ are normalized by $1/(6\pi\eta R_h)$, $1/(6\pi\eta R_h^2)$, and $1/(8\pi\eta R_h^3)$, respectively. For problems involving rotation, we use the $m_M = m_D = 6$ optimal ES monopole and dipole kernels (cf. Table I). If the problem only involves translation, we can use the optimal $m_M = 4$ monopole kernel (cf. Table II).

A. Self-mobility

Here, we examine the behavior of the linear and rotational velocity of a particle given some prescribed forces or torques, for several particle positions relative to the wall(s), in both the BW and SC geometries.

First, we consider translation–translation self-mobility. In the BW geometry, we compare our numerical results to the periodized Rotne–Prager–Blake (periodized RPB) tensor. The RPB formulas²⁴ are for a half-space with no-slip conditions on the bottom wall, valid only for $z \geq R_h$. We periodize them by directly summing over 400 images in the x and y directions to account for the periodic boundary conditions in our doubly periodic simulations (400^2 images in total). We also compare to Faxén’s power series expansions for the parallel self-translational mobility of a sphere at half and quarter channel locations [see Eq. (24) in Ref. 34]. In Fig. 3, we plot the parallel [with respect to the wall(s)], $\mu_{xx}^{tt} = \mu_{yy}^{tt}$, and perpendicular, μ_{zz}^{tt} , self mobilities of a blob. All of our numerical results agree with the corresponding reference values for both $m = 4$ and $m = 6$ for $z \gtrsim 2R_h$. By the image construction (9), the particle mobility goes to zero

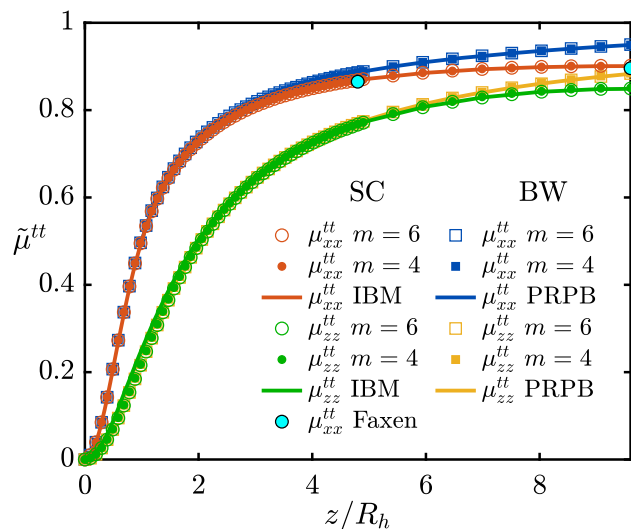


FIG. 3. Normalized translation–translation self-mobility, $\tilde{\mu}^{tt} = (6\pi\eta R_h) \mu^{tt}$, above the lower wall ($z \in [0, \frac{1}{2}H]$) in the bottom wall (BW) and slit channel (SC) geometries for $m = 6$ (empty markers) and $m = 4$ (filled markers). The reference data (shown with solid curves) for the BW and SC geometries are from a periodized Rotne–Prager–Blake (PRPB) tensor²⁴ and numerical results from the immersed boundary method (IBM),¹⁷ respectively. Faxén’s results⁵² for μ_{xx}^{tt} at $z = \frac{1}{4}H$ and $z = \frac{1}{2}H$ in the SC geometry are shown in cyan circles.

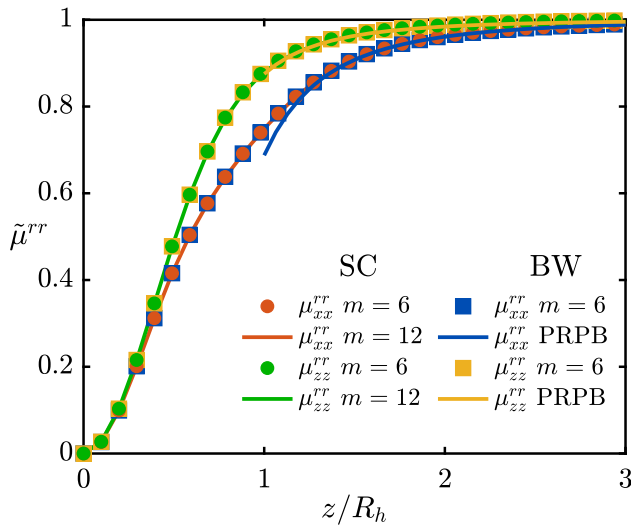


FIG. 4. Normalized rotation–rotation self-mobility, $\tilde{\mu}^{rr} = (8\pi\eta R_h^3) \mu^{rr}$, above the lower wall at $z = 0$ in the BW and SC geometries. The reference results, from PRPB for the BW geometry and a twice resolved grid ($m = 12$) for the SC geometry, are solid curves. Since $H \gg R_h$, the results are nearly identical between the BW and SC geometries.

smoothly as $z \rightarrow 0$ for both the FCM and IBM, but in a kernel/solver-dependent manner. Nevertheless, only small differences in μ_{zz}^{tr} near the wall are noticeable in the figure between the RPB, IBM, and FCM mobilities.

In Fig. 4 we show the parallel, $\mu_{xx}^{rr} = \mu_{yy}^{rr}$, and perpendicular, μ_{zz}^{rr} , rotation–rotation mobility components. Again, our numerical

results are in excellent agreement with the reference evaluations for $z \gtrsim 2R_h$. Since the wall corrections to the rotational components of the free space RPY tensor in the RPB tensor decay fast like $(R_h/z)^3$, μ_{xx}^{rr} and μ_{zz}^{rr} in the BW geometry are nearly indistinguishable from the same components in the SC geometry. As the blob approaches the wall, we see a noticeable difference between RPB and FCM for μ_{xx}^{rr} . Lacking a reference result from other methods, in the SC geometry, we compare to a numerical reference computation on a doubly refined grid (i.e., $h := \frac{1}{2}h$ and $m := 2m$ such that the support α and the hydrodynamic radius remain constant).

In Fig. 5, we show μ_{yx}^{tr} , the only nontrivial translation–rotation self-coupling component. Interestingly, the mobility does not decay to zero as $z \rightarrow \infty$, as it would for a half-space domain; we display the non-periodized RPB kernel as dashed red lines in the left panel to highlight the effect of periodic boundary conditions in x, y . Furthermore, our numerical results for $m = 6$, and to a lesser extent, even a doubly refined result with $m = 12$, exhibit small numerical oscillations as the particle moves away from the wall.

To better understand the periodic artifacts in μ_{yx}^{tr} , we analyze the contributions of periodic images of a particle to its self-mobility. Consider the analytical form of the pairwise mobility v_{yx}^{tr} (from the RPB tensor) for two particles, far away from each other, at the same height z ,

$$\eta v_{yx}^{tr}(z) \approx \frac{1}{6\pi} \frac{3\hat{z}(\hat{r}_x)^2}{\|\mathbf{r}\|^2}, \quad (59)$$

where

$$\mathbf{r} = [x^{(1)} - x^{(2)} \quad y^{(1)} - y^{(2)} \quad 2z]^\top, \quad \hat{\mathbf{r}} = \frac{\mathbf{r}}{\|\mathbf{r}\|}. \quad (60)$$

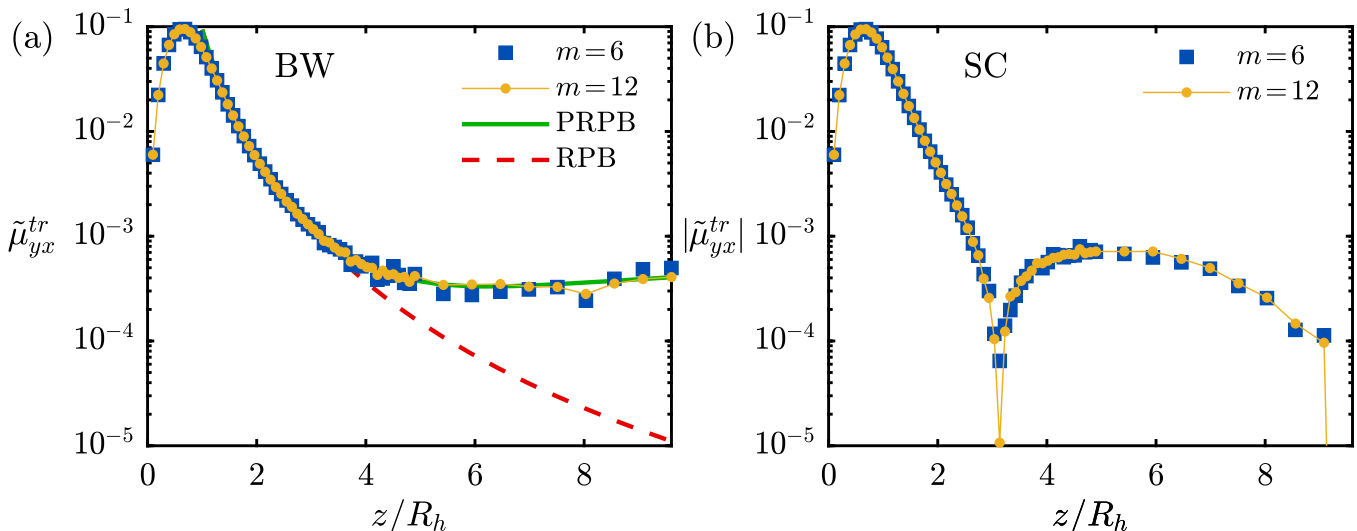


FIG. 5. Normalized translation–rotation self-mobility, $\tilde{\mu}_{yx}^{tr} = (6\pi\eta R_h^2) \mu_{yx}^{tr}$, above the lower wall ($z \in [0, \frac{1}{2}H]$) in the BW (a) and SC (b) geometries for $m = 6$ and $m = 12$ (a doubly refined grid). For the BW geometry (a), we also show the results from PRPB and the half-space *non-periodized* RPB tensors. Although not shown in this figure, for both BW and SC geometries, we find that the symmetric analog μ_{xy}^{tr} agrees with μ_{yx}^{tr} well within five digits. For the SC geometry (b), we show the absolute value due to a sign change for $z \gtrsim 3R_h$. The x axis extends to $z = \frac{1}{2}H$, for which $\tilde{\mu}_{yx}^{tr} = 0$ due to symmetry.

We can approximate the sum of the RHS of (59) over the periodic images in the xy plane with an integral in polar coordinates. We find that as $z \rightarrow \infty$, the approximate sum converges to a constant $\sim L^{-2}$, and for $z \ll L$, it is $\sim z/L^3$. The self-mobility μ_{yx}^{tr} itself decays like R_h^2/z^4 . Therefore, the combined mobility (self plus images) is dominated by the self-term for small z and by the images for large z . This results in a minimum in $\mu_{yx}^{tr}(z)$, at roughly $z_{\min} \sim (L^3 R_h^2)^{1/5}$, followed by a plateau. If the particle goes above $z_{\min}$, the translation–rotation self-coupling is dominated by the periodic images.

For the slit channel, symmetry is always broken by the presence of the upper and lower walls, except at the exact center of the channel $z/R_h = H/2$. That is, $\mu_{yx}^{tr}(H/2) = 0$, and we exclude this point from our plot due to the log scale on the y axis. As expected, we have similar behavior between SC and BW near the wall, and the numerical oscillations are diminished for both the $m = 6$ and $m = 12$ reference computations in the SC geometry. Interestingly, there is a sign change in μ_{yx}^{tr} in the slit channel for $z \gtrsim 3R_h$ (likely due to periodic images), although the coupling is weak.

B. Pair-mobility

Because we use a non-Gaussian tensor product kernel (53), the pair-mobility for two blobs in an unbounded domain does not have a strictly isotropic form, even in the absence of discretization errors, and, furthermore, the pair-mobility is difficult to compute analytically. In Appendix B, we examine the pair-mobility in the absence of walls numerically using a (large) TP domain. Our main conclusion is that the results obtained with the ES kernel are within a percent of those for a Gaussian kernel and are isotropic to an accuracy of at least two digits for the kernel and grid choices we selected.

Here, we briefly investigate the behavior of the linear and rotational relative mobility of pairs of identical particles at the same distance z from the bottom wall and separated by a distance $d = 3R_h, 4R_h, 8R_h$ in the x direction. The presented results here serve solely as consistency checks and validation of our solver against available analytical and well-established numerical solutions. In Sec. V B, we investigate in more detail some of the interesting behavior of the translation–translation pair-mobility in a slit channel.

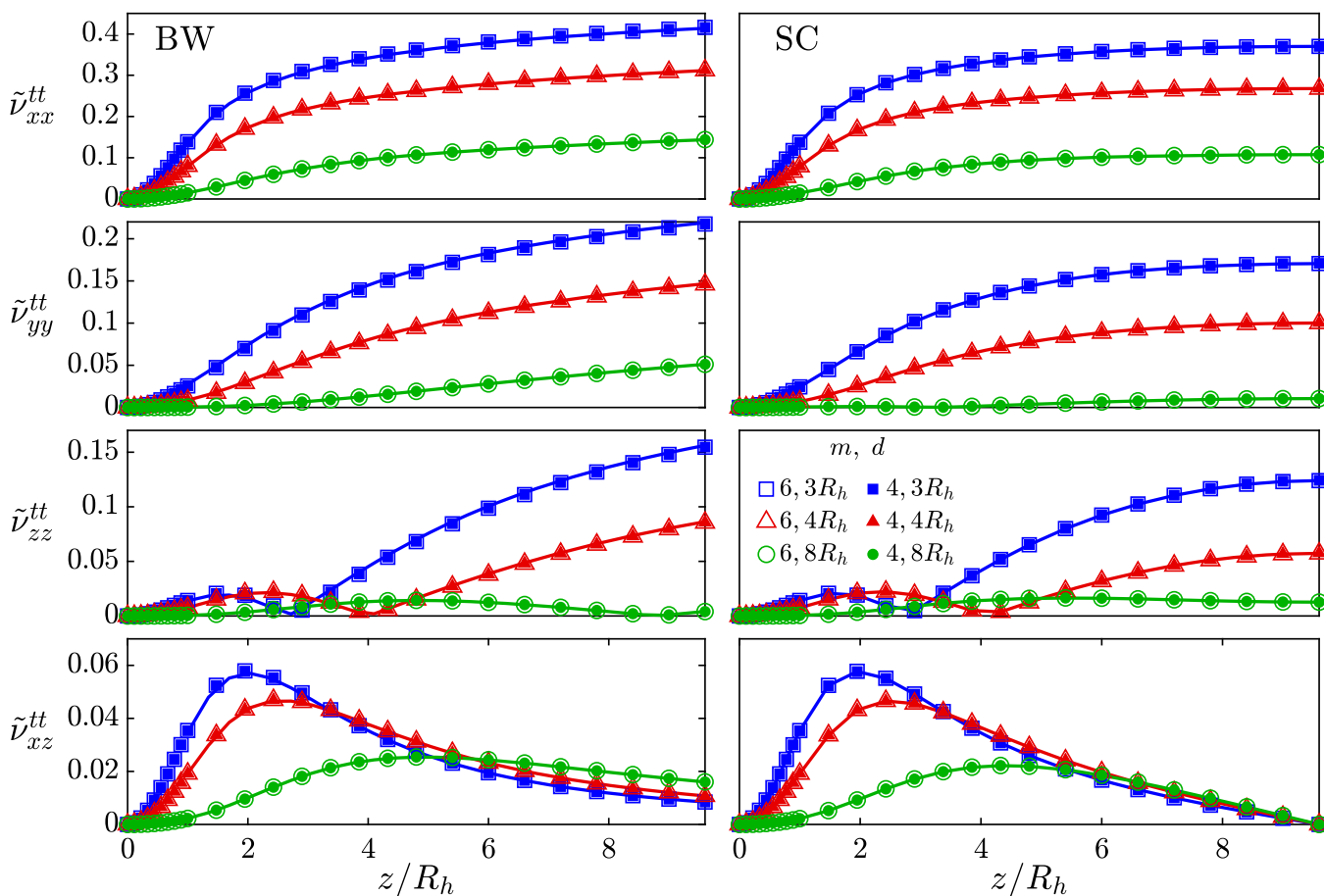


FIG. 6. Normalized translation–translation pair mobility, $\tilde{\nu}^{tt} = (6\pi\eta R_h) \nu^{tt}$, in the BW (left panel) and SC (right panel) geometries, for a pair of particles, a distance z from the bottom wall ($z \in [0, \frac{1}{2}H]$), the same y coordinate, and distance $d = 3R_h, 4R_h, 8R_h$ away from each other along the x direction. Our results using the ES kernel are shown with empty and filled markers for $m = 6$ and $m = 4$, respectively. The reference results, PRPB for the BW geometry, and $m = 12$ for the SC geometry, are solid curves.

First, we examine the pairwise components of ν^t in Fig. 6. All numerical results compare well to the reference results, i.e., RPB in BW geometry, and a doubly refined result in SC geometry, for both $m = 4$ and $m = 6$. For each mobility component considered, we see that when the particles are near the wall, greater separation leads to a reduced magnitude of coupling. This is also the case further away from the bottom wall, except for ν_{xz}^t . In Fig. 7, we inspect the pairwise components of ν^r . Our numerical results compare very well to the reference results in either geometry. As for translation–translation, the coupling decreases with particle separation, except in the case of the perpendicular–parallel component ν_{xz}^r . Finally, in Fig. 8, we consider the pairwise components of ν^{tr} and see that the reference results compare well to our numerical results. We highlight the same effects due to periodicity in xy as seen in Fig. 5 through the non-periodized RPB kernel (dotted lines) for ν_{yx}^r and ν_{xy}^r . Both ν_{yx}^r and ν_{xy}^r decay in a half-space as the particles are further from the bottom wall, but the periodicity in xy leads to a plateau in the mobility.

V. COLLOIDAL ROLLERS

In this section, we consider a case study of a driven, dense suspension of colloidal microrollers as recently investigated by some of us.²² In this system, magnetic spherical colloidal particles sediment into a dense monolayer near a bottom wall/floor. A magnetic field is used to spin the particles with the axis of rotation parallel to the wall, which induces a translational motion of the particles along the wall due to the rotation–translation coupling induced by the presence of the floor (see μ_{yx}^r in Fig. 5, and ν_{yx}^r and ν_{xy}^r in Fig. 8). A collective driven steady state is established, and both experimental and numerical results show the existence of two layers of rollers, a slower layer close to the wall, and a faster layer above the slow one;²² see the inset in Fig. 13. Since the colloidal layer remains close to the wall in the z direction but extends far in the xy directions, this system is an ideal target for our doubly periodic solver.

In Ref. 22, we developed a lubrication-corrected variant of the Stokesian dynamics method (without stresslet corrections) to simulate the collective driven dynamics of colloidal rollers and

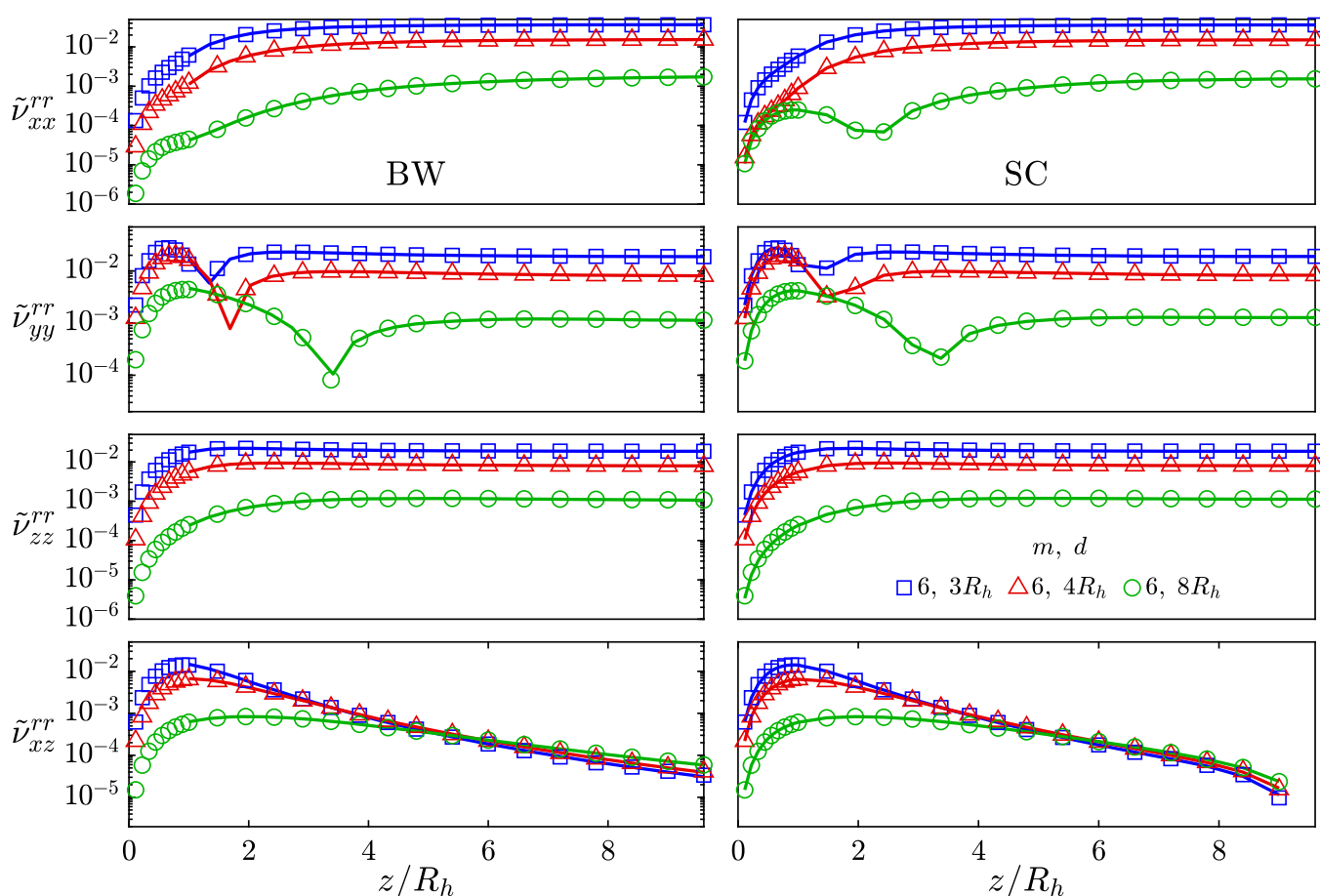


FIG. 7. Normalized rotation–rotation pair mobility, $\tilde{\nu}^{rr} = (8\pi\eta R_h^3)\nu^{rr}$, for the same system as in Fig. 6. Our results using the ES kernel with $m = 6$ are shown with empty markers. The reference results, PRPB for the BW geometry, and $m = 12$ for the SC geometry, are solid curves. Note that μ_{xz}^r goes to zero at the midplane due to symmetry.

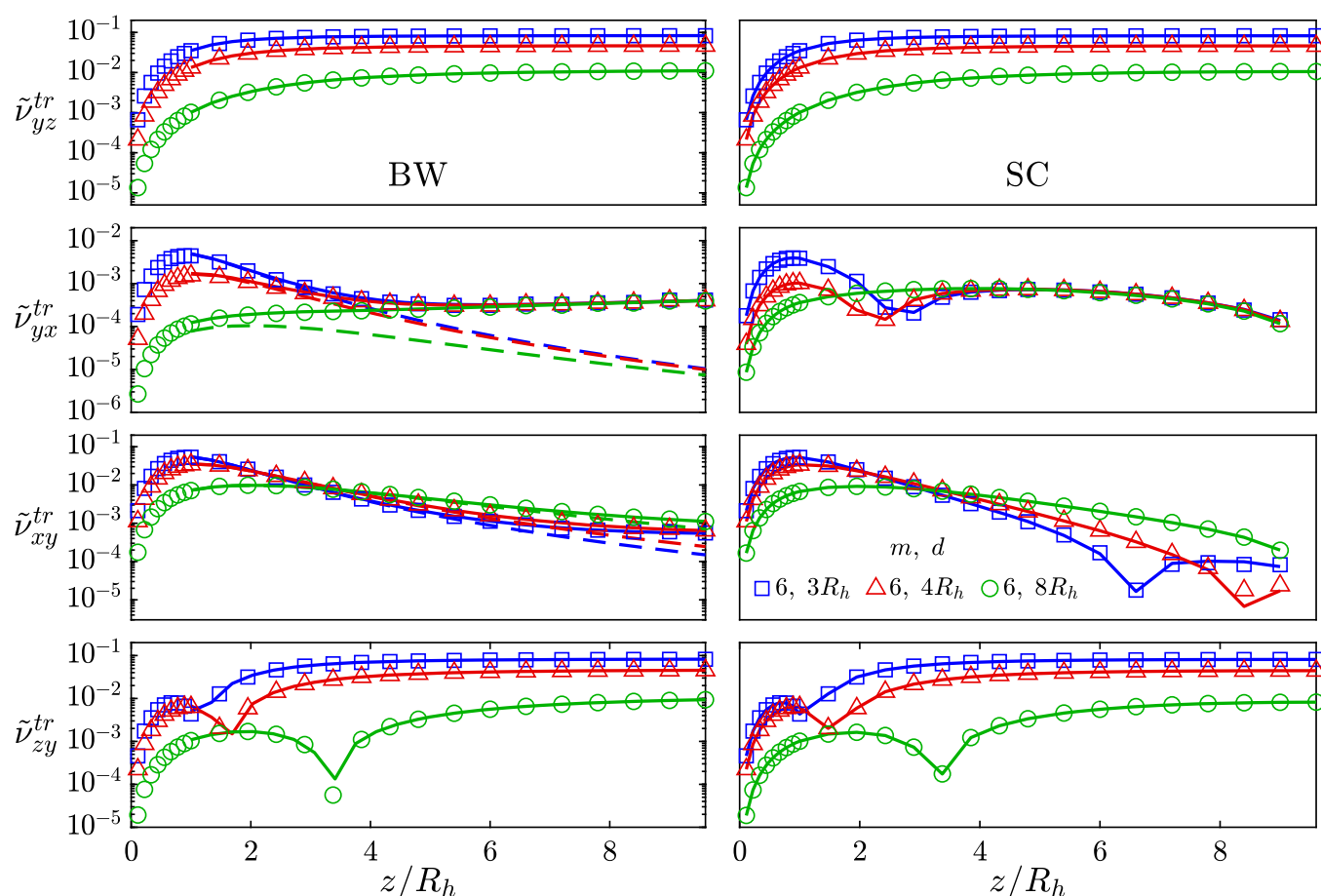


FIG. 8. Normalized translation–rotation pair-mobility, $\tilde{\nu}^{tr} = (6\pi\eta R_h^2)\nu^{tr}$, for the same system and labels as in Fig. 6. The half-space non-periodized RPB is also shown for mobility components ν_{yx}^{tr} and ν_{xy}^{tr} in BW as dashed curves. Although not shown here, for both BW and SC geometries, we find that the symmetric components of ν^{tr} agree with each other to well within five digits at each height. Note that the yx , xy , and zy components of ν^{tr} go to zero at the midplane due to symmetry.

found good agreements between computational and experimental results. In these prior studies, we computed the far-field hydrodynamic interactions using a GPU-accelerated direct pairwise summation of the Rotne–Prager–Blake tensor²⁴ generalized to account for torques/rotation. To approximate the doubly periodic conditions, in these prior works,³⁵ each particle interacted only with $3 \times 3 = 9$ periodic copies of every other particle. Because of the quadratic scaling of the direct summation, this increased the cost of the calculation by as much as two orders of magnitude (i.e., a factor of $9^2 = 81$), severely limiting scalability to larger numbers of particles.

Here, we apply our linearly scaling method to the same problem. In Sec. V A, we validate the linear scaling of our method up to the memory limitations on the GPU by periodically replicating a representative configuration from the steady-state conditions found numerically in Ref. 22. In Sec. V B, we study the convergence of an iterative method to compute the (far-field) Brownian

(stochastic) particle displacements, since efficient and (log-)linear-scaling dynamic simulations require rapid convergence of the iterative method with several iterations independent of the number of particles. While we have previously established this for BW geometries,³⁵ here we confirm this to be the case also in SC geometries, despite the slower decay of the hydrodynamic interactions with particle distance.¹ In Sec. V C, we perform dynamic simulations with a larger number of particles than previously feasible and compare to previous results based on the Rotne–Prager–Blake tensor. We also add a top wall and investigate its influence on the collective dynamics and structure of the driven suspension.

A. Linear scaling

In this section, we demonstrate the linear scaling of our GPU-based solver with the number of particles. Figure 9 shows the computation time as we periodically replicate a representative

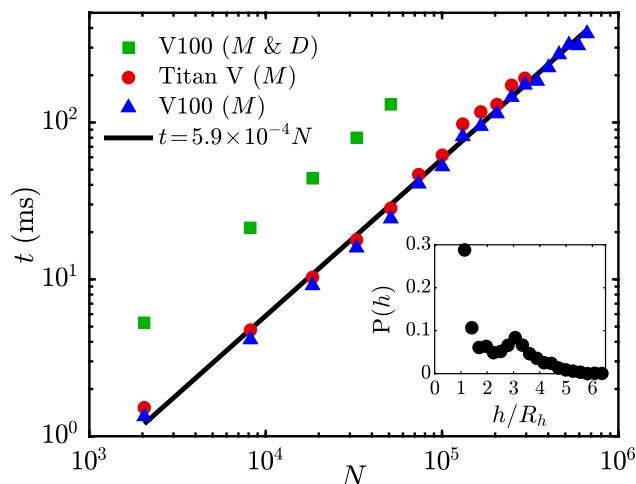


FIG. 9. Time per mobility solve vs number of particles with parameters given in Table III, on V100 and Titan V NVIDIA GPUs. We replicate a system of 2048 particles in the xy plane. Particles are uniformly distributed in the planar direction, with an area packing fraction of $\phi = 0.4$, and a bimodal (two-layer) distribution of heights $P(h)$ above the wall, as shown in the inset. The biggest system contains 663 552 particles and takes 370 ms to compute with UAMMD on a V100 GPU.

configuration of $N = 2048$ particles²² in the xy plane direction to increase the number of particles. The original configuration before replication has a system size $L_{x/y} = L = 128.8R_h$ and $H \approx 9R_h$ (see Table III). We test here the scaling for the case when there are only forces applied (M , monopole-only, kernel width $m = 4$), and when both forces and torques are applied (M and D , monopole and dipole terms, kernel width $m = 6$). The computational cost for the dipole case is significantly larger than the monopole-only case, primarily because a larger grid size is required for kernel width $m = 6$, and because additional FFTs are required in the present implementation of spreading and interpolation for the dipole terms.⁵³ For the monopole-only case, the biggest system has $L = 18 \times 128.8R_h = 2318.4R_h$ (H remains the same) and contains 663 552 particles, the maximum size that fits in the 32 GB of memory in a V100 GPU. After 169 replicas (346 112 particles), the 12 GB of memory available in the Titan V GPU is not enough.

Our performance test results clearly show a linear scaling with the number of particles, allowing computations with on the order of a million particles in less than a second. This sort of a large number of particles is infeasible with the direct summation method used by Sprinkle *et al.*²² It is difficult to perform direct timing comparisons with the linear-scaling, FMM-based multicore, and STKFMM

TABLE III. Simulation parameters for the performance tests when only forces are applied (monopole-only, M), and when both forces and torques are applied (monopole and dipole, M and D).

	H	$L_{x/y}$	N_z	$N_{x/y}$	m_M	β_M/m_M	m_D	β_D/m_D
M	$9.1R_h$	$128.8R_h$	19	150	4	1.87		
M and D	$9.2R_h$	$128.8R_h$	26	216	6	1.38	6	2.3

code of Wen *et al.*,^{28,29} since the performance of this CPU-based code depends heavily on the type of supercomputer and number of cores used, and optimization options. Nevertheless, rough timing comparisons suggest our simple GPU-friendly Stokes solver is at least an order of magnitude faster for this example on standard workstations.

B. Lanczos convergence

Brownian Dynamics (BD) simulations require a method to generate Brownian particle “velocities,” which are Gaussian random variables with mean zero and covariance $\sim \mathcal{M}$, i.e., $\mathbf{g} = \mathcal{M}^{\frac{1}{2}} \mathbf{W}$, where $\mathcal{M}^{\frac{1}{2}}$ is a square root of the mobility matrix and $\mathbf{W}$ is a collection of independent identically distributed (i.i.d) random numbers with zero mean and unit standard deviation.⁵⁴ A Lanczos iterative algorithm³⁷ can be used to estimate $\mathbf{g}$ in the Krylov subspace of $\mathcal{M}$. Prior work has shown that the number of iterations required to achieve a certain tolerance is independent of the number of particles in suspensions of particles near a single wall (BW geometry, see Fig. 1 in Ref. 35), due to the hydrodynamic screening of the wall, but grows with the number of particles in the absence of a wall, i.e., in a TP environment (see Fig. 1 in Ref. 26 and the inset of Fig. 1 in Ref. 35).

One could assume that the convergence rate of the Lanczos algorithm depends on how fast the hydrodynamic interactions decay with distance. In particular, we expect that the number of iterations is controlled by the slowest decaying hydrodynamic interactions. Therefore, for our problem of thin colloidal layers, we focus on the translation–translation components of the hydrodynamic interactions (no torques) in the xy plane, for which the decay is slowest relative to the other components. In Fig. 10, we consider a pair of particles located at the same height $z = \frac{1}{2}H$, same y , and distance d away from each other in the x direction. We apply a unit force on one of the particles in either x or y directions and measure the resulting translational velocity of the other one in the same direction, i.e., v_{xx}^t and v_{yy}^t , which we refer to as parallel, $v_{\parallel}^t$, and perpendicular, $v_{\perp}^t$, mobility components, respectively [see Fig. 10(a)]. We compare our results to the available asymptotic formulas in the literature for $d \gg R_h$. For SC geometry, both parallel and perpendicular mobilities approach the same asymptote $3H/(32\pi\eta d^2)$ [see Eq. (51) in Ref. 55]. For BW geometry, the parallel and perpendicular mobilities approach the asymptotes $3H^2/(8\pi\eta d^3)$ and $3H^4/(64\pi\eta d^5)$, respectively.²⁷ Also, note that the data are normalized with the corresponding $v^t(d=0)$, which is the mobility of a single particle located at height $z = \frac{1}{2}H$.⁵⁶

An interesting observation is that $v_{\perp}^t$ becomes negative (presence of backflow) for $d/R_h \gtrsim 4$ in the SC geometry [see the inset of Fig. 10(a)]. To provide a better picture of this phenomena, in Figs. 10(b) and 10(c), we show representative streamlines of a ghost particle velocity near a particle that is being dragged along the x direction. We see the existence of fluid vortices in the SC geometry case. Another important, and counter-intuitive, point is that the SC geometry shows a slower asymptotic decay rate compared to the BW geometry¹ (d^{-2} vs d^{-3}). Such a slow decay warrants an examination of the Lanczos convergence rate in the SC geometry and how it compares with that in the BW and TP geometries.

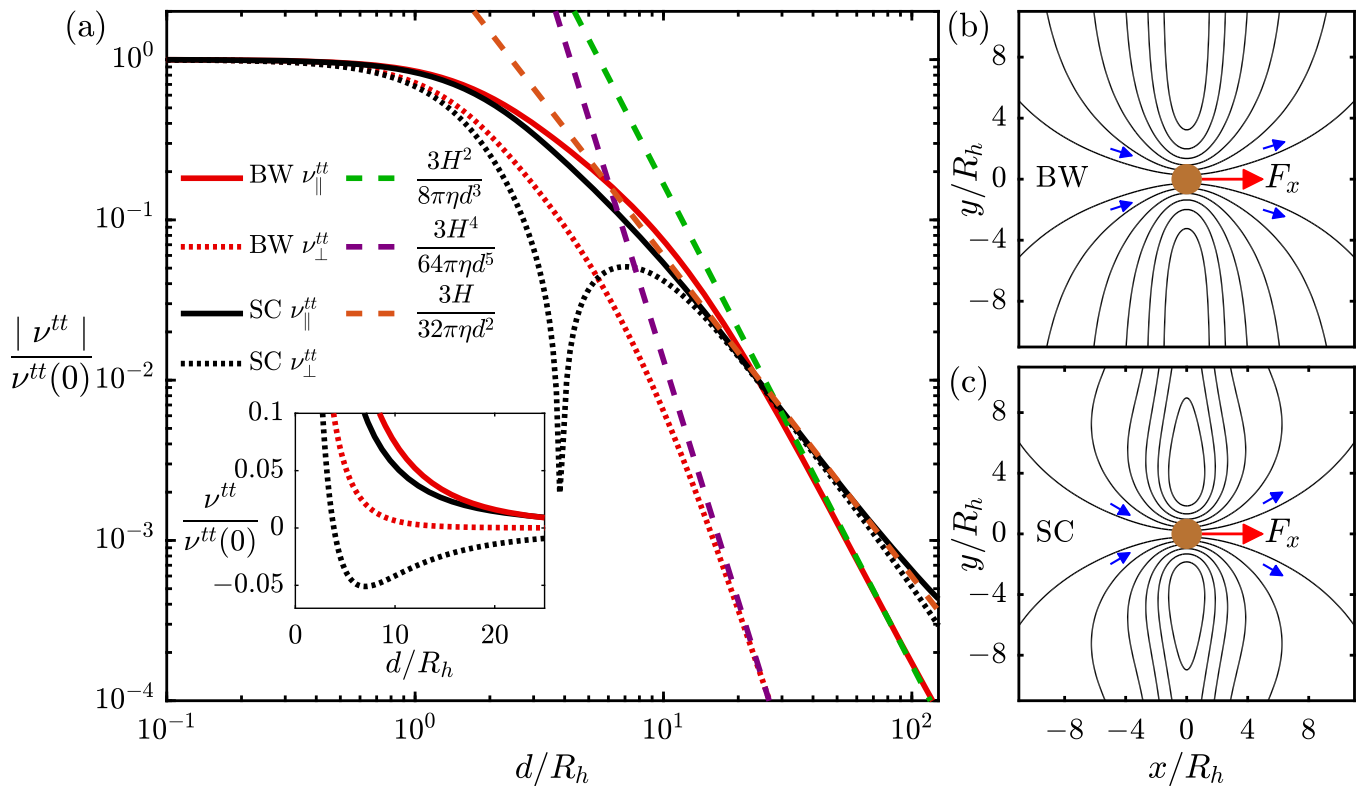


FIG. 10. (a) Two particles are placed at the same height $z = \frac{1}{2}H$, same y , and a distance d apart from each other along the x direction ($L = 512R_h$, $H = 8R_h$, and $m = 6$). One is pulled with a force in the x or y direction and the induced mobility on the other particle is measured in, respectively, x (parallel component, $\nu_{\parallel}^{tt} = \nu_{xx}^{tt}$) and y (perpendicular component, $\nu_{\perp}^{tt} = \nu_{yy}^{tt}$) directions. The dashed lines correspond to theoretical asymptotes for $d \gg R_h$. [(b) and (c)] Streamlines of a ghost particle velocity about a particle being pulled along the x direction with a unit force F_x , calculated using the x and y velocity components $U_x^{(2)} = \nu_{xx}^{tt}F_x$ and $U_y^{(2)} = \nu_{yx}^{tt}F_x$.

Figure 11 shows the relative difference $\epsilon_n = \|\mathbf{g}_n - \mathbf{g}_{n-1}\|/\|\mathbf{g}_{n-1}\|$ for the Lanczos algorithm vs the number of iterations n in the BW, SC, and TP geometries. For the BW geometry, we observe the same phenomenon showcased in Ref. 35, where the hydrodynamic screening of the wall makes the convergence of the Lanczos algorithm essentially independent of the number of particles. The number of iterations required to achieve a 3-digit accuracy remains less than 10 for $N = 2^{11}$ to 2^{17} particles. Surprisingly, the SC geometry shows a faster convergence rate than the BW one, despite the slower decay rate of the pair mobilities in this geometry (see Fig. 10). Perhaps this can be explained by the fact that the slowly decaying component is *not* of hydrodynamic origin, but rather comes from incompressibility; the actual hydrodynamic interactions are exponentially screened in the SC geometry.¹ Finally, consistent with prior work,²⁶ the convergence rate drops significantly (and fails at times) for large TP systems due to the absence of the wall effects.

It should be noted that the convergence behavior for the SC geometry depends heavily on the thickness H and the particle height h , as we explore Fig. 12. We first place 2048 particles near the bottom wall and increase the thickness H between the two walls [see Fig. 12(a)]. We observe that as $H \rightarrow \infty$, the SC behavior approaches that of a BW geometry. We repeat this experiment with the particles

placed at the midplane [see Fig. 12(b)]. This time, as $H \rightarrow \infty$, the system behavior approaches that of a TP system, as expected. Furthermore, we find that the convergence worsens as particles start to overlap the wall; this is a rare occurrence in BD simulations with steric repulsion.

C. Microrollers

In this section, we use lubrication-corrected BD to study the dynamics of a dense suspension of colloidal microrollers in the BW and SC geometries.

1. BW geometry

As discussed earlier, in Ref. 22, some of us described a method to apply near-field lubrication corrections (without stresslets) to a blob mobility matrix in the BW geometry, and how to efficiently use the corrected mobility matrix in the context of BD simulations. Blob-based descriptions of spherical particles provide efficient and adequate resolution of far-field hydrodynamic interactions at the expense of poor resolution for near-field hydrodynamics. Following Stokesian dynamics, lubrication corrections are employed to fix the inaccurate near-field hydrodynamics and effectively capture the

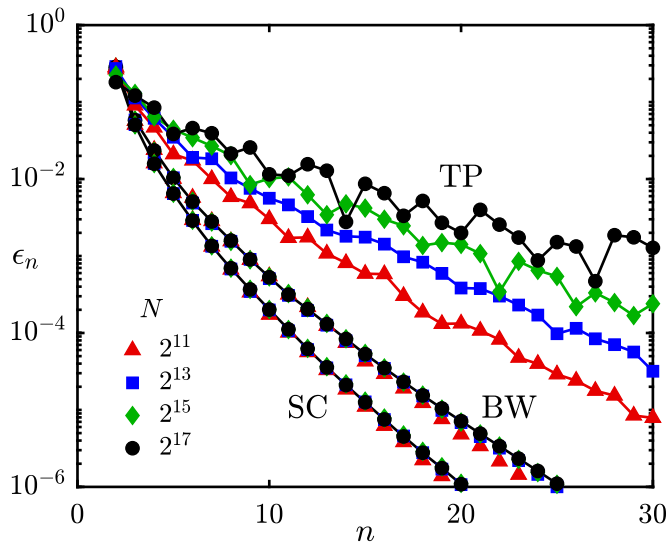


FIG. 11. Relative error ϵ_n vs number of iterations n of the Lanczos algorithm for a suspension of particles with the same configuration as in Fig. 9. Similar to Fig. 9, the system is replicated in the planar direction to study higher particle counts. The results for BW and SC geometries are obtained with the parameters given in Table III but $H = 7.5R_h$. The TP result is for the same configuration but in triply periodic mode, where we set the height of the (periodic) domain to $H = 130R_h$.

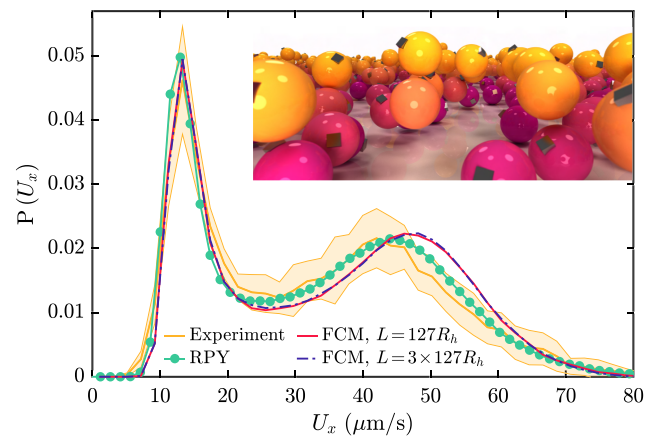


FIG. 13. A comparison of the experimentally measured (shown as a solid yellow line with a shaded area representing 95% confidence bounds) and numerically computed velocity distributions in the direction of collective motion for a suspension of microrollers above a bottom wall. Simulated data are obtained from lubrication corrected BD in a doubly periodic domain with $L = 127R_h$, using either the Rotne–Prager–Blake tensor, where periodicity is captured by directly including the 9 nearest periodic images,²² our FCM approach, or our FCM approach in a $9\times$ larger domain. Inset: A typical configuration for a uniform suspension of microrollers at planar density $\phi = 0.4$ and driving frequency $f = 9$ Hz, suspended above a bottom wall. The hematite cube embedded in the particles is overemphasized for visual clarity and to show particle orientation. Low/slow particles are colored magenta while high/fast particles are colored yellow.

true hydrodynamics of nearly touching spheres, or a sphere near a bottom wall. Specifically, we can construct the lubrication corrected FCM mobility matrix as

$$\overline{\mathcal{M}} = (\mathcal{R} + \Delta\mathcal{R})^{-1}, \quad \Delta\mathcal{R} = \mathcal{R}_{\text{sup}}^A - \mathcal{R}_{\text{sup}}^N, \quad (61)$$

where $\mathcal{R} = \mathcal{M}^{-1}$ is the resistance matrix computed using our FCM-based solver, and $\mathcal{R}_{\text{sup}}^A$ and $\mathcal{R}_{\text{sup}}^N$ are a superposition of pairwise resistance tensors for nearly touching surfaces (particle–particle or particle–wall) computed semi-analytically using the lubrication theory (or using a very refined blob-based method), and numerically by

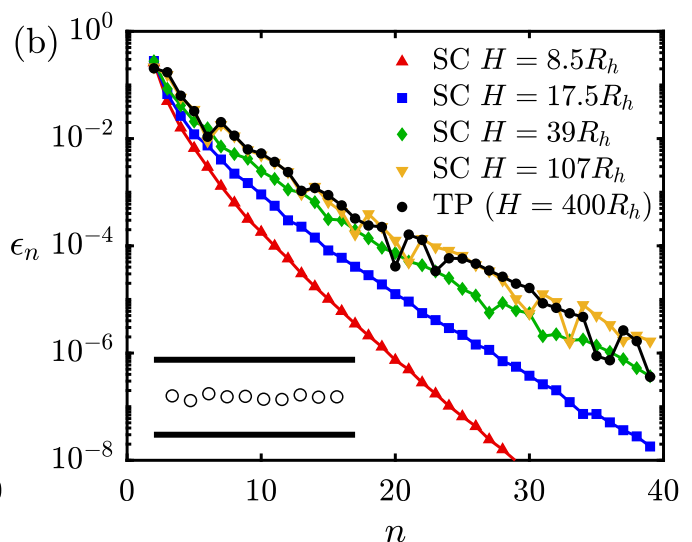
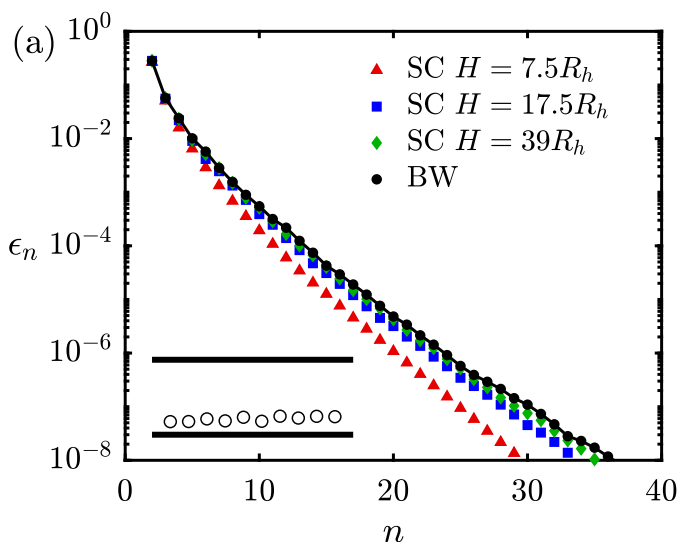


FIG. 12. Lanczos convergence for a slit channel with increasing gap H , in two scenarios: (a) Particles ($N = 2048$) are kept near the bottom wall and H increases (which becomes increasingly similar to the bottom wall geometry). (b) Particles are kept at the middle of the channel and H increases (which becomes increasingly similar to a TP geometry). Although not shown, above a relative error of $\sim 10^{-6}$, the convergence is not affected by the floating point precision (single or double precision).

our FCM method,⁵⁷ respectively. The subscript “sup” signifies superposition of isolated pair interactions; each block i, j of the resistance tensors $\mathcal{R}_{\text{sup}}^{\text{A}}$ and $\mathcal{R}_{\text{sup}}^{\text{N}}$ ($i, j \in \{1, \dots, N\}$) accounts for the near-field hydrodynamic interaction of nearby particles i and j , $\mathcal{R}_{\text{pair}, ij}^{\text{N/A}}$, while the diagonal blocks correspond to the particle–wall interactions, $\mathcal{R}_{\text{BW}, i}^{\text{N/A}}$.

To compute the correction $\Delta\mathcal{R} = \mathcal{R}_{\text{sup}}^{\text{A}} - \mathcal{R}_{\text{sup}}^{\text{N}}$, we need expressions for $\mathcal{R}_{\text{pair}, ij}^{\text{N/A}}$ and $\mathcal{R}_{\text{BW}, i}^{\text{N/A}}$. Formulas for $\mathcal{R}_{\text{pair}, ij}^{\text{A}}$ and $\mathcal{R}_{\text{BW}, i}^{\text{A}}$ are given in Appendix A of Ref. 22, while polynomial fits⁵⁸ for the coefficients of $\mathcal{R}_{\text{pair}, ij}^{\text{N}}$ for FCM can be found in Ref. 59. To compute $\mathcal{R}_{\text{BW}, i}^{\text{N}}$, we simply use our FCM approach to generate a set of data for each coefficient vs height above the wall, taking care to use a large enough $L = L_{x/y}$ so that the periodic artifacts do not pollute the data, and linearly interpolate this data as needed.

To interrogate the effectiveness of using lubrication corrections with our FCM method, we will study the uniform suspension of rotating particles above a bottom wall considered in Sec. IV of Ref. 22. The “microrollers” are magnetic since they each have a cube of hematite embedded near their surface that exerts a torque when an external, rotating, magnetic field is applied to the whole suspension. The magnetic torque is strong enough that up to a certain frequency (~ 10 Hz) the particles rotate with the same frequency of the applied field. We measure the particle velocity distribution $P(U_x)$ in the positive x direction in response to an applied

magnetic field rotating in the yz plane with a frequency of 9 Hz, for a uniform suspension of microrollers with in plane packing fraction $\phi = 0.4$. Figure 13 shows both the experimental measurements of $P(U_x)$ and the numerical results using the lubrication corrected RPY mobility given in Ref. 22, along with results from a new simulation using our lubrication corrected FCM method. The new FCM method agrees reasonably well with both the experimentally measured velocity distribution as well as the old RPY results. However, there is a clear difference that stems from the fact that lubrication corrections are only approximate and have uncontrolled precision; perhaps including stresslets as well^{10,15,40} would make the match better. Although it is tempting to compare to experimental data and choose one approximation over the other, it should be noted that some of the parameters in Ref. 22 were chosen to match experimental data, while here we reuse the same parameters as in Ref. 22 without re-estimating them.

2. SC geometry

In an SC geometry, the pair interactions remain the same, and one only needs to make modifications for the particle–wall interactions (compared to the BW geometry). In other words, we simply replace the near-field BW correction tensors $\Delta\mathcal{R}_{\text{BW}, i} = \mathcal{R}_{\text{BW}, i}^{\text{A}} - \mathcal{R}_{\text{BW}, i}^{\text{N}}$ with SC correction tensors, assuming grid invariance,

$$\Delta\mathcal{R}_{\text{SC}, i} = \mathcal{R}_{\text{SC}, i}^{\text{A}}(h) - \mathcal{R}_{\text{SC}, i}^{\text{N}}(h) = \begin{bmatrix} \Delta X^{tt}(h) \hat{z}\hat{z}^T + \Delta Y^{tt}(h) (\mathcal{I} - \hat{z}\hat{z}^T) & -\Delta Y^{tr}(h) \hat{z}\times \\ \Delta Y^{tr}(h) \hat{z}\times & \Delta X^{rr}(h) \hat{z}\hat{z}^T + \Delta Y^{rr}(h) (\mathcal{I} - \hat{z}\hat{z}^T) \end{bmatrix}, \quad (62)$$

where h is the height of the particle’s center above the bottom wall, $\hat{z}$ is the unit normal vector of the bottom wall, and $\hat{z}\times$ is the matrix that takes the cross-product with $\hat{z}$. However, the available analytical formulas for the coefficients of $\mathcal{R}_{\text{SC}, i}^{\text{A}}$ typically take the form of unwieldy and slowly convergent series,¹⁷ so we instead calculate the coefficients using a well-resolved multiblob representation of a sphere in a channel geometry. Recalling that only $\Delta\mathcal{R}_{\text{SC}, i}$ (but not $\mathcal{R}_{\text{SC}, i}^{\text{N/A}}$ individually) is needed, all calculations should be done with the same unit cell in the xy plane; this ensures that the results will rapidly converge in the limit as the periodic domain size $L_{x/y} = L \rightarrow \infty$. Note that the periodic domain size in all of these pre-calculations was taken to be $L = 32R_h$ as this was determined to be sufficient to avoid any noticeable periodic artifacts.

Figure 14 shows each component of $\Delta\mathcal{R}_{\text{SC}, i}$ in (62) for a particle in an SC geometry with channel height $H = 6R_h$ (notation for the coefficients follows Ref. 60). The components of $\Delta\mathcal{R}_{\text{SC}, i}$ are computed using an increasingly resolved multiblob¹⁷ representation of the sphere and extrapolated (Richardson) to infinite resolution. Figure 14 also shows each component of the single wall lubrication corrections to the nearest wall $\Delta\mathcal{R}_{\text{BW}, i}$, where $\mathcal{R}_{\text{BW}, i}^{\text{A}}$ is computed

with a very resolved 2562 multiblob representation of a sphere.²² Note that it is also possible to add the lubrication corrections from both walls, but we have found this to be less accurate in thin channels because the corrections are doubled in the channel’s center. Comparing the behavior of each component in $\Delta\mathcal{R}_{\text{BW}, i}$ with the behavior of $\Delta\mathcal{R}_{\text{SC}, i}$ shows that, at least for this representative problem with a channel width $H = 6R_h$, superposing single wall corrections is a very reasonable approximation to the full slit channel corrections. Still, we opt to use the channel corrections in our simulations to emphasize that this approach can be adopted even for very thin channels where superposing single wall corrections may provide a worse approximation. Numerically, we find that basing lubrication corrections on the nearest wall works surprisingly well even for channels as thin as $H = 3R_h$.

3. Effects of confinement on suspensions of microrollers

A suspension of microrollers above a BW will self-separate into a layer of slow-moving particles that remain close to the wall,

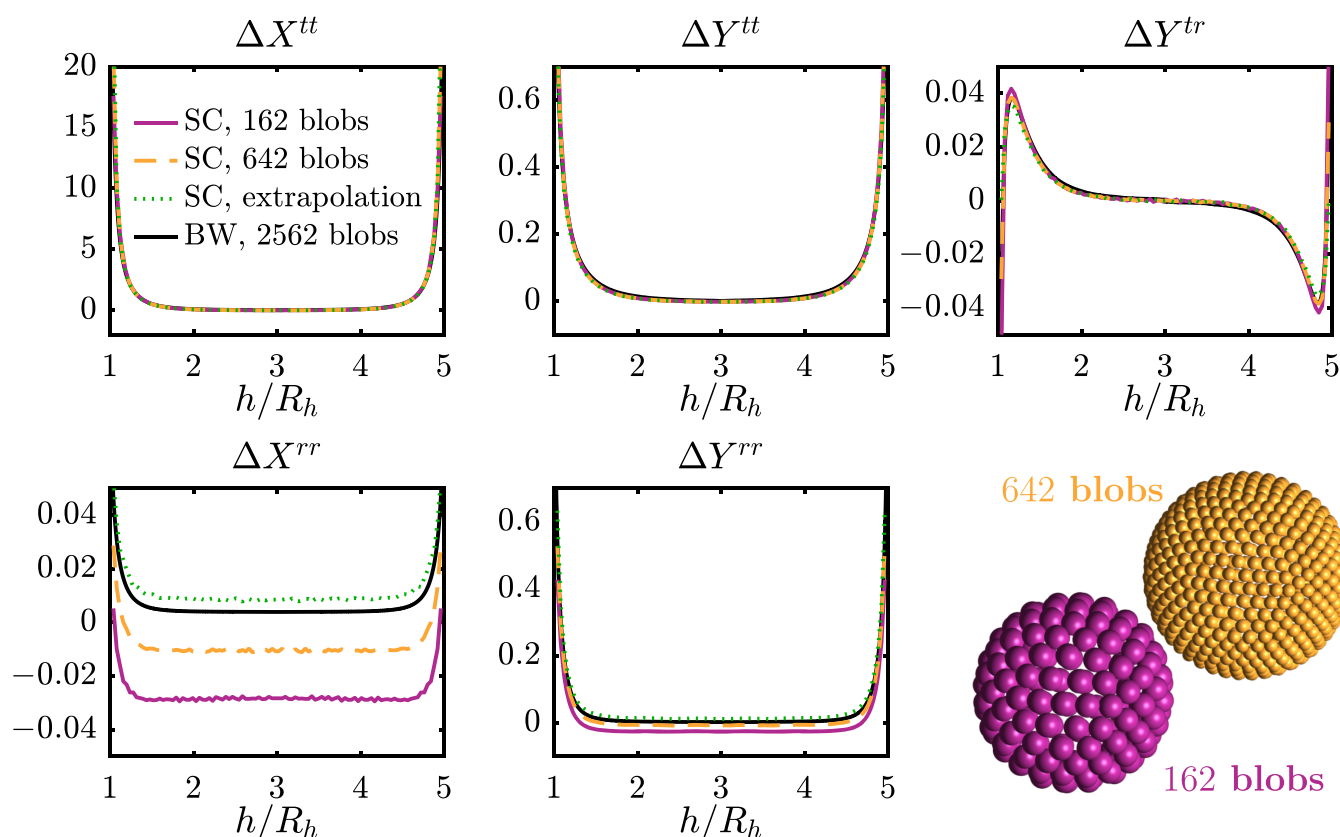


FIG. 14. For an SC geometry with height $H = 6R_h$, each panel shows one of the five components in $\Delta\mathcal{R}_{SC, i}$ [see (62)] computed using our FCM-based solver and increasingly resolved rigid multiblob simulations.¹⁷ Richardson extrapolation is used to approximate more accurate values for $\Delta\mathcal{R}_{SC, i}$ than we can presently compute with the multiblob method because large grid sizes are required. Each panel also shows components of the single wall corrections $\Delta\mathcal{R}_{BW, i}$ for the nearest wall, which provides a very reasonable approximation to the corrections in the SC geometry. Note that extrapolation is necessary (and not as accurate) for the components involving rotation since the rotational hydrodynamic radii of low-resolution multiblob spheres are different from their translational hydrodynamic radii,¹⁷ and computations with 2562 blobs and $L = 32R_h$ are infeasible due to limited GPU memory.

mainly due to gravity, and a fast layer of particles that are lifted above the slow layer, against gravity. These results are detailed in Ref. 22 and reconfirmed in this section. Our spectral FCM approach now enables us to investigate the effect of further confinement on the velocity of a microroller suspension. In this section, we will use the same simulation parameters as those for the BW geometry considered in Sec. V C, but change the geometry to an SC with height $H = 6R_h$. The $H = 6R_h$ channel width was chosen based on the 98th percentile of the particle height distribution obtained in the BW geometry (see the inset in Fig. 13) so that the top wall only provides a minor steric constraint on the suspension.

Figure 15 compares the velocity distributions $[P(U_x)]$ of the driven particles in a suspension confined by a single BW, or an SC with height $H = 6R_h$. The velocity distributions in each case are also broken up into sub-distributions consisting of particles

whose height above the BW is above/below a certain threshold (specifically, $h \lesseqgtr 2R_h$). This shows that the faster particles in the suspension (those which have the largest positive U_x) are also the highest in the suspension. The inset of Fig. 15 indicates that the particle height distributions in the BW and SC ($H = 6R_h$) geometries are fairly similar, while the main figure shows large differences in the velocity distributions for these two geometries. Although both distributions are bimodal, there are no really fast particles in the SC case. Coincidentally, the “fast” layer is at a distance of about $3R_h$ from the bottom wall. If it were not for the “slow” layer of particles breaking the vertical symmetry, particles at $h = 3R_h$ would not move when spinning. We can thus conclude that this significant slowing of the suspension velocity in the SC geometry compared with the BW geometry is due to the presence of a top wall, which changes the character of the flow generated by driving the suspension.

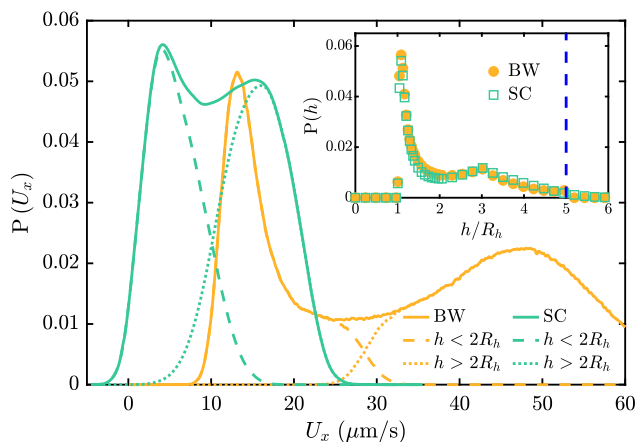


FIG. 15. Distributions of particle height (inset) and velocity (main figure) for a micro-roller suspension above a single wall (BW, gold line with circular markers), and a slit channel with width $H = 6R_h$ (SC, teal line with square markers). Particle velocity distributions are shown for each channel geometry where each distribution is also broken into sub-distributions for low (close to the bottom wall, $h < 2R_h$) and high ($h > 2R_h$) particles. These sub-distributions show that regardless of domain geometry the “high” particles make up the mode of “fast” particles in the suspension (where “fast” is to be interpenetrated as having a large, positive U_x). Inset: dimensionless particle height (h/R_h) distributions are shown for each channel geometry. The dashed line demarcates the maximum possible particle height in the SC geometry.

VI. CONCLUSIONS

In this paper, we have developed a variant of the force coupling method (FCM) for computing hydrodynamic interactions among blobs/particles suited for doubly periodic geometries, including the ones that have an open aperiodic direction. We employed a more compact non-Gaussian kernel to reduce the grid size required to resolve the particles. Our novel spectral Stokes solver uses FFTs in the two periodic directions and Chebyshev transforms in the aperiodic direction and is suited for efficient implementation on GPUs using a single 3D FFT for each Fourier–Chebyshev transformation. We used our method to show that a dense monolayer of colloidal microrollers splits into two layers even in a slit channel, but moves substantially slower.

The key ideas used in our Stokes solver, namely, the use of the Dirichlet-to-Neumann map to obtain boundary conditions in the aperiodic direction, as well as the use of correction solves to enforce boundary conditions, can be used to generalize the solver to other boundary conditions in the aperiodic direction, such as partial slip (e.g., superhydrophobic surfaces) or free slip boundaries (e.g., a fluid–fluid interface), including cases when there are particles on both sides of the boundary. However, it is important to point out that the method relies in a key way on the double periodicity in two directions to obtain a system of uncoupled one-dimensional boundary value problems and is therefore tailored to doubly periodic geometries.

It is possible to extend our work to account for the contribution of stresslets, as already implemented in the traditional FCM method; see Eqs. (7) and (11) in Ref. 7. While we have not done so, there is no difference as far as our method or any FCM method is concerned

between torque and stresslet; they are spread with the same kernel, and both are spread and interpolated using derivatives of the kernel. The real challenge is how to efficiently solve for the stresslets; see the work of Fiore and Swan¹⁵ for some ideas toward effective preconditioning. Additionally, one might include fluctuations (Brownian motion) with stresslets as described in Appendix B of Ref. 32, as well as in Refs. 40 and 61.

Two key challenges for future work are to combine the force coupling method with Ewald splitting and to devise a method to generate stochastic displacements for BD without iterations. This would allow us to decouple the blob hydrodynamic radius from the grid size and make stochastic simulations of comparable cost to deterministic simulations. Both of these challenges have been tackled for the RPY kernel in the positively split Ewald method²⁶ for triply periodic domains. Both challenges have also partially been addressed in a low-order immersed boundary approach^{33,62} including for slit channel geometries, but the accuracy is difficult to control and the Ewald splitting is approximated by a particle–particle–particle–mesh approach.

Considerable progress has been made on handling the first challenge due to its importance in boundary integral methods for Stokes flow. Spectral Ewald Stokes solvers for the singular Stokeslet and Rotlet have been developed extensively by the group of Tornberg,^{30,63} including recent work that unifies arbitrary periodicities in different directions.³¹ One can directly apply these methods to the bottom wall geometry we studied here using a method of images, but the finite size of the particles and the potential overlap of the particles with each other and the wall need to be accounted for either using the RPY kernel (as in recent fast multipole methods²⁹) or an FCM-like approach. To our knowledge, an Ewald splitting for FCM has not yet been developed even for triply periodic domains. It seems quite plausible that by combining ideas from this paper (and our work on Poisson solvers¹³) with extensive work in the field of FCM and spectral Ewald methods, in the future the community can develop an Ewald splitting method for finite-size blobs capable of handling all doubly periodic geometries.

The second challenge, generating the action of the square root of the mobility explicitly without costly iterations, has hardly been addressed for non-triply periodic domains. We have been able to achieve this for doubly periodic unconfined geometries using the approach developed in this work; however, it required switching to a Galerkin formulation of the boundary value problems in the unbounded direction. This not only leads to dense linear systems, instead of the pentadiagonal ones in the method presented here, but it also couples all three velocities and pressure in one linear solve for each planar Fourier mode. This makes the approach less suited to memory-limited GPUs and more costly if sufficient memory is available. Furthermore, accounting for the wall(s) remains a nontrivial challenge for future work.

ACKNOWLEDGMENTS

We thank Leslie Greengard, Kyongmin Yeo, and Amanda Howard for helpful discussions. This work was supported by the NSF under Award No. DMS-2011544 and through a Research and Training Group in Modeling and Simulation under Award No. RTG/DMS-1646339.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Aref Hashemi: Data curation (supporting); Methodology (supporting); Visualization (supporting); Writing – original draft (lead); Writing – review & editing (equal). **Raul Perez Pelaez:** Investigation (supporting); Methodology (supporting); Software (lead); Validation (supporting); Visualization (supporting); Writing – review & editing (supporting). **Sachin R. Natesh:** Investigation (supporting); Validation (lead). **Brennan Sprinkle:** Investigation (supporting); Methodology (supporting); Writing – review & editing (supporting). **Ondrej Maxian:** Methodology (supporting). **Zecheng Gan:** Investigation (supporting). **Aleksandar Donev:** Conceptualization (lead); Funding acquisition (lead); Supervision (lead); Writing – original draft (supporting); Writing – review & editing (equal).

DATA AVAILABILITY

All of the codes, input files, and scripts used to write this paper can be found on github at <https://github.com/stochasticHydroTools/DoublyPeriodicStokes>.

APPENDIX A: SLIT CHANNEL GEOMETRY WITH SLIP VELOCITY

For the SC geometry, for increased generality and because of its relevance to microfluidics, we show how to impose partial slip boundary conditions at $z = 0$ and $z = H$,

$$\mathbf{u}|_{z=0} - \mathbf{u}_0^{\text{wall}} = \ell_0 \left[\frac{\partial u(x, y, 0)}{\partial z} \quad \frac{\partial v(x, y, 0)}{\partial z} \quad 0 \right]^\top, \quad (\text{A1a})$$

$$\mathbf{u}|_{z=H} - \mathbf{u}_H^{\text{wall}} = -\ell_H \left[\frac{\partial u(x, y, H)}{\partial z} \quad \frac{\partial v(x, y, H)}{\partial z} \quad 0 \right]^\top. \quad (\text{A1b})$$

Here, $\mathbf{u}_{0/H}^{\text{wall}} = [u_{0/H}^{\text{wall}} \quad v_{0/H}^{\text{wall}} \quad 0]^\top$ are the wall velocities, and $\ell_{0/H}$ are the slip lengths, with $\ell = 0$ denoting a no-slip wall.

For simplicity, we write the equations explicitly only for u , with identical equations for v with the swap $x \leftrightarrow y$. Note that the DP solver remains the same as that of the single wall geometry, noting that particle kernels have images in both walls. It is the correction solve that takes care of the boundary conditions.

1. Correction solve

We have $\mathbf{u} = \mathbf{u}_{\text{DP}} + \mathbf{u}_{\text{corr}}$, which upon substitution into (A1), and some rearrangement, yields

$$\begin{aligned} \mathbf{u}_{\text{corr}}|_{z=0} - \ell_0 \left[\frac{\partial u_{\text{corr}}(x, y, 0)}{\partial z} \quad \frac{\partial v_{\text{corr}}(x, y, 0)}{\partial z} \quad 0 \right]^\top \\ = \left[u_0(x, y) \quad v_0(x, y) \quad w_0(x, y) \right]^\top, \end{aligned} \quad (\text{A2a})$$

$$\mathbf{u}_{\text{corr}}|_{z=H} + \ell_H \left[\frac{\partial u_{\text{corr}}(x, y, H)}{\partial z} \quad \frac{\partial v_{\text{corr}}(x, y, H)}{\partial z} \quad 0 \right]^\top$$

$$= \left[u_H(x, y) \quad v_H(x, y) \quad w_H(x, y) \right]^\top, \quad (\text{A2b})$$

where

$$u_0(x, y) = u_0^{\text{wall}} - u_{\text{DP}}(x, y, 0) + \ell_0 \frac{\partial u_{\text{DP}}(x, y, 0)}{\partial z}, \quad (\text{A3a})$$

$$w_0(x, y) = -w_{\text{DP}}(x, y, 0), \quad (\text{A3b})$$

$$u_H(x, y) = u_H^{\text{wall}} - u_{\text{DP}}(x, y, H) - \ell_H \frac{\partial u_{\text{DP}}(x, y, H)}{\partial z}, \quad (\text{A3c})$$

$$w_H(x, y) = -w_{\text{DP}}(x, y, H). \quad (\text{A3d})$$

Following a similar method as the one described in Sec. III B, one can find the general solutions to $\hat{p}_{\text{corr}}$ and $\hat{\mathbf{u}}_{\text{corr}}$, the pressure and velocity in Fourier space, as

$$\hat{p}_{\text{corr}}(\mathbf{k}, z) = C_0 e^{-kz} + D_0 e^{kz}, \quad (\text{A4})$$

$$\hat{\mathbf{u}}_{\text{corr}}(\mathbf{k}, z) = -\frac{C_0 i k_x}{2\eta k} z e^{-kz} + \frac{D_0 i k_x}{2\eta k} z e^{kz} + C_x e^{-kz} + D_x e^{kz}, \quad (\text{A5})$$

$$\hat{w}_{\text{corr}}(\mathbf{k}, z) = \frac{C_0}{2\eta} z e^{-kz} + \frac{D_0}{2\eta} z e^{kz} + C_z e^{-kz} + D_z e^{kz}. \quad (\text{A6})$$

To find the constants, we start by writing the boundary conditions (A2) in Fourier space

$$\hat{\mathbf{u}}_{\text{corr}}(\mathbf{k}, 0) - \ell_0 \frac{\partial \hat{\mathbf{u}}_{\text{corr}}(\mathbf{k}, 0)}{\partial z} = \hat{\mathbf{u}}_0(\mathbf{k}), \quad (\text{A7a})$$

$$\hat{w}_{\text{corr}}(\mathbf{k}, 0) = \hat{w}_0(\mathbf{k}), \quad (\text{A7b})$$

$$\hat{\mathbf{u}}_{\text{corr}}(\mathbf{k}, H) + \ell_H \frac{\partial \hat{\mathbf{u}}_{\text{corr}}(\mathbf{k}, H)}{\partial z} = \hat{\mathbf{u}}_H(\mathbf{k}), \quad (\text{A7c})$$

$$\hat{w}_{\text{corr}}(\mathbf{k}, H) = \hat{w}_H(\mathbf{k}). \quad (\text{A7d})$$

Substituting (A5) and (A6) into (A7) results in the following equations for the unknown constants:

$$\begin{aligned} \left(\frac{i k_x \ell_0}{2\eta k} \right) C_0 - \left(\frac{i k_x \ell_0}{2\eta k} \right) D_0 + (1 + k \ell_0) C_x \\ + (1 - k \ell_0) D_x = \hat{u}_0(\mathbf{k}), \end{aligned} \quad (\text{A8a})$$

$$C_z + D_z = \hat{w}_0(\mathbf{k}), \quad (\text{A8b})$$

$$\begin{aligned} \frac{i k_x H e^{-kH}}{2\eta k} \left(k \ell_H - \frac{\ell_H}{H} - 1 \right) C_0 + \frac{i k_x H e^{kH}}{2\eta k} \left(k \ell_H + \frac{\ell_H}{H} + 1 \right) D_0 \\ + e^{-kH} (1 - k \ell_H) C_x + e^{kH} (1 + k \ell_H) D_x = \hat{u}_H(\mathbf{k}), \end{aligned} \quad (\text{A8c})$$

$$\frac{H e^{-kH}}{2\eta} C_0 + \frac{H e^{kH}}{2\eta} D_0 + e^{-kH} C_z + e^{kH} D_z = \hat{w}_H(\mathbf{k}). \quad (\text{A8d})$$

Furthermore, the continuity equation in Fourier space (38) can be written at $z = 0, H$,

$$ik_x \hat{u}_{\text{corr}}(\mathbf{k}, 0/H) + ik_y \hat{v}_{\text{corr}}(\mathbf{k}, 0/H) + \frac{\partial \hat{w}_{\text{corr}}(\mathbf{k}, 0/H)}{\partial z} = 0. \quad (\text{A9})$$

Inserting the general solutions (A5) and (A6) into the above equations yields two more equations for the unknown coefficients

$$\frac{C_0}{2\eta} + \frac{D_0}{2\eta} + ik_x C_x + ik_x D_x + ik_y C_y + ik_y D_y - kC_z + kD_z = 0, \quad (\text{A10a})$$

$$\begin{aligned} \frac{e^{-kH}}{2\eta} C_0 + \frac{e^{kH}}{2\eta} D_0 + ik_x e^{-kH} C_x + ik_x e^{kH} D_x + ik_y e^{-kH} C_y \\ + ik_y e^{kH} D_y - ke^{-kH} C_z + ke^{kH} D_z = 0. \end{aligned} \quad (\text{A10b})$$

Now we have a system of eight equations given by (A8) (with the corresponding equations for v) and (A10), which can be solved to find the eight unknown constants $C_0, D_0, C_{x/y/z}, D_{x/y/z}$ for each $\mathbf{k} = [k_x \ k_y]^\top \neq 0$.

2. Zero mode ($\mathbf{k} = 0$)

The zero mode $\mathbf{k} = 0$ solutions to the velocity component normal to the walls, $\hat{w}(\mathbf{0}, z)$, and the pressure, $\hat{p}(\mathbf{0}, z)$, remain the same as those of the single wall geometry, given by (47) and (50). However, the $\mathbf{k} = 0$ mode of the tangential velocity components will be different. Fourier transform of the slip boundary conditions (A1) is

$$\hat{\mathbf{u}}(\mathbf{k}, 0) - \hat{\mathbf{u}}_0^{\text{wall}}(\mathbf{k}) = \ell_0 \begin{bmatrix} \frac{\partial \hat{u}(\mathbf{k}, 0)}{\partial z} & \frac{\partial \hat{v}(\mathbf{k}, 0)}{\partial z} & 0 \end{bmatrix}^\top, \quad (\text{A11a})$$

$$\hat{\mathbf{u}}(\mathbf{k}, H) - \hat{\mathbf{u}}_H^{\text{wall}}(\mathbf{k}) = -\ell_H \begin{bmatrix} \frac{\partial \hat{u}(\mathbf{k}, H)}{\partial z} & \frac{\partial \hat{v}(\mathbf{k}, H)}{\partial z} & 0 \end{bmatrix}^\top. \quad (\text{A11b})$$

Therefore, similar to the single-wall case [cf. (51) and (52)], one can find $\hat{u}(\mathbf{0}, z)$ (and analogously $\hat{v}(\mathbf{0}, z)$) via solving the following BVPs:

$$\begin{aligned} \frac{\partial^2 \hat{u}(\mathbf{0}, z)}{\partial z^2} &= -\frac{\hat{f}(\mathbf{0}, z)}{\eta}, \quad \hat{u}(\mathbf{0}, 0) - \ell_0 \frac{\partial \hat{u}(\mathbf{0}, 0)}{\partial z} = \hat{u}_0^{\text{wall}}(\mathbf{0}), \\ \hat{u}(\mathbf{0}, H) + \ell_H \frac{\partial \hat{u}(\mathbf{0}, H)}{\partial z} &= \hat{u}_H^{\text{wall}}(\mathbf{0}). \end{aligned} \quad (\text{A12})$$

APPENDIX B: TRANSLATION-TRANSLATION PAIR COUPLING

In this appendix, we numerically compute and investigate the invariance under rotation and translation of the pair translational mobility, $\mathcal{M}^t$, for the ES kernel in a large triply periodic domain. We will examine our numerical approximation to $\mathcal{M}^t$ as a function of $d = \|\mathbf{d}\| = \|\mathbf{d}^{(1)} - \mathbf{d}^{(2)}\|$, where $\mathbf{d}^{(1)}$ and $\mathbf{d}^{(2)}$ are the positions of the two particles. If the kernel is rotationally invariant [which our tensor product ES kernel (53) is not], then $\mathcal{M}^t$ would have the invariant form

$$\mathcal{M}^t(\mathbf{d}) = f(d)\mathcal{I} + g(d)\frac{\mathbf{d} \otimes \mathbf{d}}{d^2} \quad (\text{B1})$$

for some functions $f(d)$ and $g(d)$. Lomholt and Maxey⁷ derived analytical forms for *free space* $f(d)$ and $g(d)$ when particles

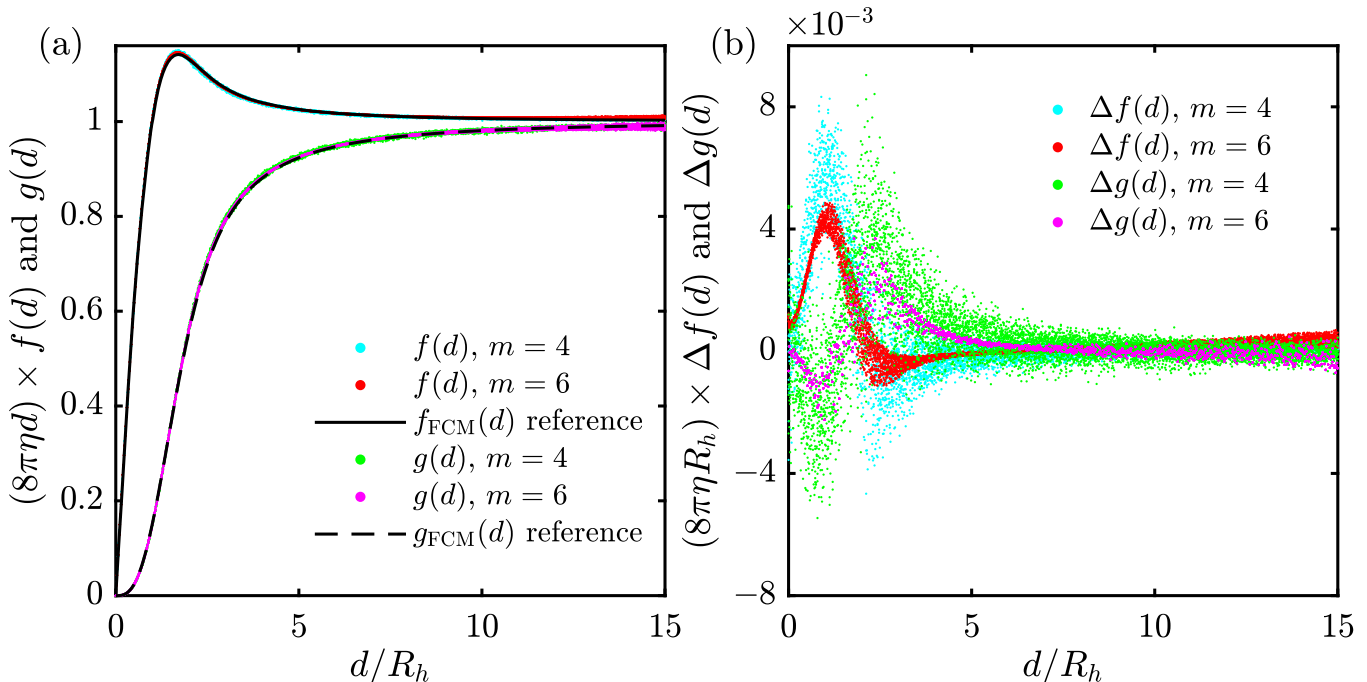


FIG. 16. (a) Numerically calculated, normalized, mobility functions $(8\pi\eta d)f(d)$ and $(8\pi\eta d)g(d)$, for two spheres in a large triply-periodic domain with ES kernel, vs the distance d between the two spheres, along with reference FCM curves given by (B2) for the continuum model in free space with Gaussian envelopes. (b) Normalized difference between the numerical and analytical values of $f(d)$ and $g(d)$ (i.e., $(8\pi\eta R_h)[f(d) - f_{\text{FCM}}(d)]$ and $(8\pi\eta R_h)[g(d) - g_{\text{FCM}}(d)]$, respectively) vs d .

are represented by Gaussian force envelopes (in the absence of discretization artifacts),

$$f(d) = \frac{1}{8\pi\eta d} \left(\left(1 + 2 \frac{R_h^2}{\pi d^2} \right) \operatorname{erf} \left(\frac{d\sqrt{\pi}}{2R_h} \right) - 2 \frac{R_h}{\pi d} \exp \left(-\frac{\pi d^2}{4R_h^2} \right) \right), \quad (\text{B2a})$$

$$g(d) = \frac{1}{8\pi\eta d} \left(\left(1 - 6 \frac{R_h^2}{\pi d^2} \right) \operatorname{erf} \left(\frac{d\sqrt{\pi}}{2R_h} \right) + 6 \frac{R_h}{\pi d} \exp \left(-\frac{\pi d^2}{4R_h^2} \right) \right). \quad (\text{B2b})$$

The invariant form (B1) does not strictly apply, in practice, for our FCM method for several reasons. The first is that the tensor product kernel we use is not rotationally invariant as a Gaussian kernel is. The second is that rotational invariance is broken for a finite system such as a periodic box. Finally, there are discretization artifacts due to the finite grid resolution in the Stokes solver, especially for the small kernel widths like $m = 4$. Here we numerically demonstrate that our method produces pair-mobility that matches (B2) to at least two digits.

The numerical domain is a cubic box of side length $L = 200$ and uniform grid spacing $h = 1$. We consider particles represented by the optimal $m = 6$ monopole kernel with $R_h = 1.55$ or the optimal $m = 4$ monopole kernel with $R_h = 1.205$ (see Table II). The two particles always use the same kernel types. We generate 10^4 random particle pair configurations at a distance $d \in [0, 25]$ from each other. For each pair, we compute the pair parallel and perpendicular mobilities $v_{\parallel}$ and $v_{\perp}$ with respect to the direction $\mathbf{p}$. According to (B1), we have

$$v_{\parallel} \approx f(d; L) + g(d), \quad (\text{B3})$$

$$v_{\perp} \approx f(d; L), \quad (\text{B4})$$

where $f(d; L)$ depends on L due to the finite size of the triply periodic box. Following Hasimoto's corrections for the self-mobility in triply periodic domains⁵⁰ [see (55)], we correct the numerically calculated $f(d; L)$ for $d \ll L$ by

$$f(d) \approx f(d; L) + \frac{1}{6\pi\eta R_h} \left(2.84 \frac{R_h}{L} \right). \quad (\text{B5})$$

In Fig. 16, we compare the corrected $f(d)$ and $g(d)$ for the ES kernel to the free space $f(d)$ and $g(d)$ for Gaussian force envelopes given by (B2). We see a good match for both $m = 4$ and $m = 6$ monopole ES kernels [see Fig. 16(a)]. Figure 16(b) shows the normalized difference between the numerically calculated $f(d)$ and $g(d)$ by the ES kernel and the traditional FCM mobility functions (B2). We see that the data clusters around a curve that is systematically different from (B2), but the difference is on the order of a percent or less. Also, the scatter of $f(d)$ and $g(d)$ due to imperfect translational and rotational invariance are well controlled within two digits. Note that the increased error at high d/R_h is due to periodic boundary conditions artifacts.

REFERENCES

- H. Diamant, "Hydrodynamic interaction in confined geometries," *J. Phys. Soc. Jpn.* **78**, 041002 (2009).
- H. Yamakawa, "Transport properties of polymer chains in dilute solution: Hydrodynamic interaction," *J. Chem. Phys.* **53**, 436 (1970).
- J. Rotne and S. Prager, "Variational treatment of hydrodynamic interaction in polymers," *J. Chem. Phys.* **50**, 4831 (1969).
- R. Cortez, L. Fauci, and A. Medovikov, "The method of regularized Stokeslets in three dimensions: Analysis, validation, and application to helical swimming," *Phys. Fluids* **17**, 031504 (2005).
- Y. Bao, J. Kaye, and C. S. Peskin, "A Gaussian-like immersed-boundary kernel with three continuous derivatives and improved translational invariance," *J. Comput. Phys.* **316**, 139 (2016), software and updated documentation available at <https://github.com/stochasticHydroTools/IBMethod>, including also a new 5-pt kernel with three continuous derivatives.
- M. R. Maxey and B. K. Patel, "Localized force representations for particles sedimenting in Stokes flow," *Int. J. Multiphase Flow* **27**, 1603 (2001).
- S. Lomholt and M. R. Maxey, "Force-coupling method for particulate two-phase flow: Stokes flow," *J. Comput. Phys.* **184**, 381 (2003).
- K. Yeo and M. R. Maxey, "Dynamics of concentrated suspensions of non-colloidal particles in Couette flow," *J. Fluid Mech.* **649**, 205 (2010).
- E. E. Keaveny, "Fluctuating force-coupling method for simulations of colloidal suspensions," *J. Comput. Phys.* **269**, 61 (2014).
- K. Yeo and M. R. Maxey, "Simulation of concentrated suspensions using the force-coupling method," *J. Comput. Phys.* **229**, 2401 (2010).
- A. H. Barnett, J. Magland, and L. af Klinteberg, "A parallel nonuniform Fast Fourier Transform library based on an 'exponential of semicircle' kernel," *SIAM J. Sci. Comput.* **41**, C479 (2019).
- C. S. Peskin, "The immersed boundary method," *Acta Numerica* **11**, 479 (2002).
- O. Maxian, R. P. Peláez, L. Greengard, and A. Donev, "A fast spectral method for electrostatics in doubly periodic slit channels," *J. Chem. Phys.* **154**, 204107 (2021), software available at <https://github.com/stochasticHydroTools/DPPoissonTests>.
- J. W. Swan and G. Wang, "Rapid calculation of hydrodynamic and transport properties in concentrated solutions of colloidal particles and macromolecules," *Phys. Fluids* **28**, 011902 (2016).
- A. M. Fiore and J. W. Swan, "Fast stokesian dynamics," *J. Fluid Mech.* **878**, 544 (2019).
- J. W. Swan, J. F. Brady, R. S. Moore *et al.*, "Modeling hydrodynamic self-propulsion with Stokesian dynamics. Or teaching Stokesian dynamics to swim," *Phys. Fluids* **23**, 071901 (2011).
- F. Balboa Usabiaga, B. Kallemov, B. Delmotte, A. P. S. Bhalla, B. E. Griffith, and A. Donev, "Hydrodynamics of suspensions of passive and active rigid particles: A rigid multiblob approach," *Commun. Appl. Math. Comput. Sci.* **11**, 217 (2016), software available at <https://github.com/stochasticHydroTools/RigidMultiblobsWall>.
- B. Sprinkle, F. Balboa Usabiaga, N. A. Patankar, and A. Donev, "Large scale Brownian dynamics of confined suspensions of rigid particles," *J. Chem. Phys.* **147**, 244103 (2017), software available at <https://github.com/stochasticHydroTools/RigidMultiblobsWall>.
- S. F. Schoeller, A. K. Townsend, T. A. Westwood, and E. E. Keaveny, "Methods for suspensions of passive and active filaments," *J. Comput. Phys.* **424**, 109846 (2021).
- O. Maxian, B. Sprinkle, C. S. Peskin, and A. Donev, "Hydrodynamics of a twisting, bending, inextensible fiber in Stokes flow," *Phys. Rev. Fluids* **7**, 074101 (2022), software available at <https://github.com/stochasticHydroTools/SlenderBody>.
- A. Sierou and J. F. Brady, "Accelerated Stokesian dynamics simulations," *J. Fluid Mech.* **448**, 115 (2001).
- B. Sprinkle, E. B. van der Wee, Y. Luo, M. M. Driscoll, and A. Donev, "Driven dynamics in dense suspensions of microrollers," *Soft Matter* **16**, 7982 (2020), software available at <https://github.com/stochasticHydroTools/RigidMultiblobsWall>.
- A. Broms, M. Sandberg, and A. Tornberg, "A locally corrected multiblob method with hydrodynamically matched grids for the Stokes mobility problem," *arXiv:2207.11210* (2022).
- J. W. Swan and J. F. Brady, "Simulation of hydrodynamically interacting particles near a no-slip boundary," *Phys. Fluids* **19**, 113306 (2007).

- ²⁵E. Wajnryb, K. A. Mizerski, P. J. Zuk, and P. Szymczak, "Generalization of the Rotne-Prager-Yamakawa mobility and shear disturbance tensors," *J. Fluid Mech.* **731**, R3 (2013).
- ²⁶A. M. Fiore, F. Balboa Usabiaga, A. Donev, and J. W. Swan, "Rapid sampling of stochastic displacements in Brownian dynamics simulations," *J. Chem. Phys.* **146**, 124116 (2017), software available at <https://github.com/stochasticHydroTools/PSE>.
- ²⁷J. R. Blake, A note on the image system for a Stokeslet in a no-slip boundary," *Proc. Cambridge Philos. Soc.* **70**, 303–310 (1971).
- ²⁸W. Yan and R. Blackwell, "Kernel aggregated fast multipole method," *Adv. Comput. Math.* **47**, 69 (2021).
- ²⁹W. Yan and M. Shelley, "Universal image systems for non-periodic and periodic Stokes flows above a no-slip wall," *J. Comput. Phys.* **375**, 263 (2018).
- ³⁰S. Srinivasan and A.-K. Tornberg, "Fast Ewald summation for Green's functions of Stokes flow in a half-space," *Res. Math. Sci.* **5**, 35 (2018).
- ³¹J. Bagge and A.-K. Tornberg, "Fast Ewald summation for Stokes flow with arbitrary periodicity," *arXiv:2210.01255* (2022).
- ³²S. Delong, F. B. Usabiaga, R. Delgado-Buscalioni, B. E. Griffith, and A. Donev, "Brownian dynamics without Green's functions," *J. Chem. Phys.* **140**, 134110 (2014), software available at <https://github.com/stochasticHydroTools/FIB>.
- ³³D. R. Ladiges, A. Nonaka, K. Klymko, G. C. Moore, J. B. Bell, S. P. Carney, A. L. Garcia, S. R. Natesh, and A. Donev, "Discrete ion stochastic continuum overdamped solvent algorithm for modeling electrolytes," *Phys. Rev. Fluids* **6**, 044309 (2021), software available at <https://github.com/AMReX-FHD/FHDeX>.
- ³⁴B. Kallemov, A. P. S. Bhalla, B. E. Griffith, and A. Donev, "An immersed boundary method for rigid bodies," *Commun. Appl. Math. Comput. Sci.* **11**, 79 (2016), software available at <https://github.com/stochasticHydroTools/RigidBodyIB>.
- ³⁵F. Balboa Usabiaga, B. Delmotte, and A. Donev, "Brownian dynamics of confined suspensions of active microrollers," *J. Chem. Phys.* **146**, 134104 (2017), software available at <https://github.com/stochasticHydroTools/RigidMultiblobsWall>.
- ³⁶S. L. Dance and M. R. Maxey, "Incorporation of lubrication effects into the force-coupling method for particulate two-phase flow," *J. Comput. Phys.* **189**, 212 (2003).
- ³⁷T. Ando, E. Chow, Y. Saad, and J. Skolnick, "Krylov subspace methods for computing hydrodynamic interactions in Brownian dynamics simulations," *J. Chem. Phys.* **137**, 064106 (2012).
- ³⁸E. Chow and Y. Saad, "Preconditioned Krylov subspace methods for sampling multivariate Gaussian distributions," *SIAM J. Sci. Comput.* **36**, A588 (2014).
- ³⁹For our results, we consider a square domain in xy plane for simplicity, i.e., $(L_{xy} = L)$.
- ⁴⁰B. Delmotte and E. E. Keaveny, "Simulating Brownian suspensions with fluctuating hydrodynamics," *J. Chem. Phys.* **143**, 244109 (2015).
- ⁴¹Chebyshev coefficients of a function can be obtained by evaluating its values at the Chebyshev points and taking a fast cosine Fourier transform [Ref. 45 and J. Makhoul, "A fast cosine transform in one and two dimensions," *IEEE Trans. Acoust., Speech, Signal Process.* **28**, 27 (1980)]. Our implementation uses a 3D FFT with $2N_z - 2$ points along the z direction for simplicity. An alternative approach is introduced by Makhoul that requires no upsampling at all at the cost of some increased complexity; such a fast DCT is, however, not implemented in the cuFFT GPU library (but is in the FFTW CPU library).
- ⁴²L. N. Trefethen, *Spectral Methods in MATLAB* (SIAM, 2000).
- ⁴³R. P. Peláez, UAMMD, 2022, <https://github.com/RaulPPelaez/UAMMD>.
- ⁴⁴Kernels must never extend outside open boundaries, which can always be placed at an appropriate z location to ensure this.
- ⁴⁵L. Greengard, "Spectral integration and two-point boundary value problems," *SIAM J. Numer. Anal.* **28**, 1071 (1991).
- ⁴⁶For partial slip walls, also compute $\partial_z \hat{u}_{DP}(\mathbf{k}, z = 0/H)$ by Chebyshev differentiation and direct summation.
- ⁴⁷Since the derivative of the ES kernel is unbounded near the boundaries of its support, we have implemented this by truncating the derivative to zero for arguments with magnitude larger than the point where the derivative has minimal magnitude before it blows up. See Ref. 31 for a more sophisticated treatment.
- ⁴⁸In particular, it requires using a different kernel along different dimensions. Additionally, to ensure all components of the particle mobility go to zero at a wall, we must use a positive sign for the mirror image for the z partial derivative of the ES kernel.
- ⁴⁹See also Ref. 31 for a related use of an ES-like kernel as a window function in spectral Ewald methods.
- ⁵⁰H. Hasimoto, "On the periodic fundamental solutions of the Stokes equations and their application to viscous flow past a cubic array of spheres," *J. Fluid Mech.* **5**, 317 (1959).
- ⁵¹The value of K_m compares well to the first term in the higher order periodic correction of Hasimoto.⁵⁰ The value of K_D is also very close to $4\pi/3$ as predicted by M. Zuzovsky, P. M. Adler, and H. Brenner ["Spatially periodic suspensions of convex particles in linear shear flows. III. Dilute arrays of spheres suspended in Newtonian fluids," *Phys. Fluids* **26**, 1714 (1983)] but with an opposite sign, which we believe to be a misprint in their work.
- ⁵²H. Faxén, *Einwirkung der Gefäßwände auf den Widerstand gegen die Bewegung einer kleinen Kugel in einer zähen Flüssigkeit* (Uppsala University, 1921).
- ⁵³The alternative approach of differentiating the kernel is implemented in our CPU code.
- ⁵⁴BD simulations in confined domains also require a method to generate a stochastic drift term $\sim \partial_r \cdot \mathbf{M}$, which can be done using two additional mobility solves per time step with Random Finite Differences.³⁵
- ⁵⁵N. Liron and S. Mochon, "Stokes flow for a Stokeslet between two parallel flat plates," *J. Eng. Math.* **10**, 287 (1976).
- ⁵⁶For large values of H and in the absence of periodic effects, $v^H(d=0)$ approaches the mobility of a particle in an unbounded domain, $1/(6\pi\eta R_h)$.
- ⁵⁷The method described in Ref. 22 used the RPY tensor to compute $\mathcal{R}$ and $\mathcal{R}_{\text{sup}}^N$.
- ⁵⁸Instead of polynomials fits, we opt to use linear interpolation on a precomputed set of values for each coefficient in both $\mathcal{R}_{\text{pair}, ij}^N$ and $\mathcal{R}_{\text{BW}, i}^N$.
- ⁵⁹A. A. Howard, M. R. Maxey, and K. Yeo, "Settling of heavy particles in concentrated suspensions of neutrally buoyant particles under uniform shear," *Fluid Dyn. Res.* **50**, 041401 (2018).
- ⁶⁰D. J. Jeffrey and Y. Onishi, "Calculation of the resistance and mobility functions for two unequal rigid spheres in low-Reynolds-number flow," *J. Fluid Mech.* **139**, 261 (1984).
- ⁶¹A. M. Fiore and J. W. Swan, "Rapid sampling of stochastic displacements in Brownian dynamics simulations with stresslet constraints," *J. Chem. Phys.* **148**, 044114 (2018).
- ⁶²D. R. Ladiges, J. G. Wang, I. Srivastava, A. Nonaka, J. B. Bell, S. P. Carney, A. L. Garcia, and A. Donev, "Modeling electrokinetic flows with the discrete ion stochastic continuum overdamped solvent algorithm," *Phys. Rev. E* **106**, 035104 (2022), software available at <https://github.com/AMReX-FHD/FHDeX>.
- ⁶³D. Lindbo and A.-K. Tornberg, "Spectrally accurate fast summation for periodic Stokes potentials," *J. Comput. Phys.* **229**, 8994 (2010).