An integral-based spectral method for inextensible slender fibers in Stokes flow

Ondrej Maxian[*a], Alex Mogilner[†a,b], and Aleksandar Donev[‡a]

[a]*Courant Institute of Mathematical Sciences, New York University, New York, NY 10012*
[b]*Department of Biology, New York University, New York, NY 10012*

October 28, 2020

Abstract

Every animal cell is filled with a cytoskeleton, a dynamic gel made of inextensible fibers, such as microtubules, actin fibers, and intermediate filaments, all suspended in a viscous fluid. Numerical simulation of this gel is challenging because the fiber aspect ratios can be as large as 10^4. We describe a new method for rapidly computing the dynamics of inextensible slender filaments in periodically-sheared Stokes flow. The dynamics of the filaments are governed by a nonlocal slender body theory which we partially reformulate in terms of the Rotne-Prager-Yamakawa hydrodynamic tensor. To enforce inextensibility, we parameterize the space of inextensible fiber motions and strictly confine the dynamics to the manifold of inextensible configurations. To do this, we introduce a set of Lagrange multipliers for the tensile force densities on the filaments and impose the constraint of no virtual work in an L^2 weak sense. We augment this approach with a spectral discretization of the local and nonlocal slender body theory operators which is linear in the number of unknowns and gives improved spatial accuracy over approaches based on solving a line tension equation. For dynamics, we develop a second-order semi-implicit temporal integrator which requires at most a few evaluations of nonlocal hydrodynamics and a few block diagonal linear solves per time step. After demonstrating the improved accuracy and robustness of our approach through numerical examples, we apply our formulation to a permanently cross-linked actin mesh in a background oscillatory shear flow. We observe a characteristic frequency

---

[*]Email: om759@nyu.edu

[†]Email: mogilner@cims.nyu.edu

[‡]Email: donev@courant.nyu.edu

at which the network transitions from quasi-static, primarily elastic behavior to dynamic, primarily viscous behavior. We find that nonlocal hydrodynamics increases the viscous modulus by as much as 25%. Most of this increase, in contrast to the smaller (about 10%) increase in the elastic modulus, is due to short-ranged intra-fiber interactions.

1 Introduction

Interactions of long, thin, inextensible filaments with a viscous fluid abound in biology, engineering, physics, and medicine. In biology, the swimming mechanisms of flagellated organisms have been of interest for decades, with an initial cluster of studies on how force and torque balances lead to swimming [13, 9, 11, 44], and a more recent focus on flagellar bundling and propulsion [48, 54, 56]. In physics and engineering, suspensions of high-aspect-ratio fibers have been observed to display non-Newtonian, viscoelastic behavior both experimentally [10] and computationally [53, 80].

Our particular area of interest is the simulation of semi-flexible filaments that make up the cell cytoskeleton. These inextensible filaments, which include microtubules and actin filaments, maintain the cell's structure, control the mechanics of the cell division process, and have aspect ratios from 10^2 to 10^4 [4]. In vivo, actin filaments are generally bound together into networks by cross-linking proteins, the properties of which determine the viscoelastic behavior of the cytoskeleton [86, 33, 3]. While there has been much work recently on microtubule systems [65, 76], there has yet to be, to our knowledge, a systematic study of the influence of hydrodynamic interactions on the mechanics and rheology of cross-linked actin networks. One of the goals of this paper is to develop an efficient numerical technique that can simulate a cross-linked network of thousands of inextensible actin filaments and take into account the filament interactions with a viscous solvent at zero Reynolds number. Our method not only handles the inextensibility and stiffness of the fibers robustly and efficiently, but also accounts for, in near-linear time with respect to the number of fibers, the long-ranged hydrodynamic interactions between fibers, which we show can increase the viscous modulus of the network by as much as 25%. We will not consider thermal fluctuations or elastic twisting of the filaments in this work.

Prior to the year 2000, tools for analytical analysis and numerical simulation of filaments in Stokes flow were developed in parallel by several authors. For slender filaments, a useful approach for both analysis and computation is to reduce the problem from three dimensions to one by assuming a certain distribution of singularities along the filament centerline. This approach, referred

to as "slender body theory" (SBT), was first introduced by Hancock [31] and later expanded upon by Batchelor [7]. By using the method of matched asymptotics, Keller and Rubinow were the first to derive an SBT that is uniformly accurate in the fiber slenderness ratio $\epsilon = \text{radius/length}$ [38]. Johnson further developed the theory by adding higher order corrections and correctly treating a fiber with free ends [36], and Götz re-derived the SBT of Keller and Rubinow in a more general context, allowing him to apply the theory to Oseen's and Poisson's equations [26].

Because the SBTs of Keller and Rubinow, Johnson, and Götz are uniformly accurate in powers of ϵ, they have formed the basis of most of the more recent analysis. To this end, Mori et al. recently showed that these singularity solutions solve a well-posed Stokes problem with non-standard boundary conditions on the filament surface [62, 63]. Koens and Lauga also showed that the SBT singularity solution can be recovered by matched asymptotic expansion of the full surface boundary integral formulation of Stokes flow [41].

On the numerical side, non-SBT based techniques for the simulation of fibers in Stokes flow have been in use for many decades. The most prevalent among these are regularized singularity methods, in which the fibers are discretized by a series of marker points, each of which is assigned a force according to the fiber physics. The force on each marker is then regularized, and the Stokes equations are solved to obtain a fluid velocity on the marker points due to the collection of regularized forces. The type of regularization determines the particular numerical method. For example, in the immersed boundary (IB) method of Peskin and collaborators, the force is regularized by smearing it onto a background grid on which the fluid equations are solved, and this velocity field is then interpolated back onto the marker points [67, 68]. In the special case when the spreading and interpolation are done with a Gaussian kernel, the method is referred to as a force coupling method (FCM) [50, 57]. For regularization and interpolation over the surface of a sphere, the force to velocity relationship (mobility matrix) can be computed analytically and is known as the Rotne-Prager-Yamakawa (RPY) tensor [73, 75, 21]. Finally, the method of regularized Stokeslets describes the case when the Stokes equations are solved analytically for a given regularization function, and the resulting velocity field is evaluated directly on the marker points without an interpolation kernel[1] [15, 17].

All of these regularization methods have been used to model immersed rods, but generally with penalty terms to enforce inextensibility [66, 47]. To our knowledge, only the recent approaches of

---

[1]This destroys the symmetry of hydrodynamic interactions, which is otherwise preserved in the IB, FCM, and RPY approaches.

Schoeller et al. [75] and Jabbarzadeh and Fu [34] enforce inextensibility rigorously with Lagrange multipliers. In the case of [34], the fiber is broken up into segments of regularized point forces, and each segment is updated via a rotation that preserves inextensibility exactly. Yet both [75] and [34] suffer from the same pitfall as all regularization methods: when modeling slender fibers, the width/radius of the regularization function must be on the order of the fiber radius [12]. Since the regularization width is also tied to the fiber discretization spacing (and, in the IB method, the fluid grid spacing), semi-flexible slender fibers must be discretized with many more points than would be necessary in a continuum, SBT-based approach. While this limitation can be partially overcome with adaptive mesh refinement [28], grid coarsening with local velocity correction [58], and kernel-independent fast multipole methods (to accelerate many-body sums) [72], the fact remains that to achieve controlled accuracy for dilute suspensions of many fibers, discretizing the fiber by a collection of marker points is much less efficient than treating it as a continuum with SBT. This is especially true for semi-flexible or stiff fibers, where the smooth fiber shapes are well represented in a spectral basis with rapidly decaying coefficients.

Despite their limitations, regularized singularities are sometimes convenient to work with since they are nonsingular on the fiber centerline, and can therefore be easily evaluated there. A natural workaround to the regularization lengthscale issue is to take a continuum limit of many regularized point forces along the fiber [18] or along segments of a fiber [30, 16, 88]. For example, Walker et al. recently derived an SBT that uses regularized singularities along the fiber centerline and can be used for fibers of non-uniform cross section [89]. In still more recent work [88], they combined this theory with the regularized Stokeslet segments approach of Cortez [16] to yield a numerical method which is more efficient than that of regularized point forces, but still quadratic-complexity in the total number of segments.

Since the shapes of biological filaments are smooth, a better approach is to represent the entire shape using an interpolating polynomial, rather than breaking into segments, and take a continuum limit of regularized singularities along the entire fiber. We show in Appendix A that applying this procedure to the RPY tensor yields a formula for the fiber velocity that is identical to SBT away from the fiber endpoints. For the reasons just listed, efficient simulation of many slender fibers requires a numerical method that can handle an SBT-type formulation for the fiber velocity.

To our knowledge, Shelley and Ueda were the first to derive such a method and use it to simulate immersed slender fibers. By designing a numerical method around the analytical results of slender body theory, they reduced the complexity of the numerical computations from three dimensions to

one [77, 78]. Their formulation, however, relies on the filament being a closed loop, thus excluding many problems from biology, engineering, and physics where the filament ends are free.

Tornberg and Shelley treated inextensible filaments with free ends using an SBT-based numerical method [84]. In their approach, inextensibility is preserved by deriving an auxiliary (integro-)differential equation for the line tension in the filament, which acts as a Lagrange multiplier to preserve inextensibility. This method has since been used in applications with flexible (and sometimes fluctuating) filaments [55, 95], and was also extended to simulate falling rigid fibers, the novelty there being that many of the SBT-related integrals can be done analytically [83]. More recently, Nazockdast et al. modified the approach of Tornberg and Shelley to make it feasible to simulate many-body cellular fiber assemblies. By replacing the second-order spatial discretization of Tornberg and Shelley with a spectral spatial discretization and utilizing a kernel-independent FMM to accelerate sums, Nazockdast et al. developed a parallel algorithm that makes it possible to simulate $\mathcal{O}(1000)$ fibers in linear time [65].

Despite these recent advances, imposing inextensibility via a tension boundary value problem (BVP) leads to a number of drawbacks which are present in all of the prior SBT-based numerical methods. To begin, the line tension equation of [84] involves multiplications of high-order (as high as four) derivatives of the fiber position function. This leads to severe aliasing problems and a loss of spatial accuracy in the spectral formulation [65]. In addition, the "inextensibility" of the filaments is still subject to discretization error and requires inserting a penalty term into the line tension equation that reduces the discrete extensibility [84]. For fibers tugged by cross-linkers or strong extensional flows, this penalty parameter will be large, introducing artificial stiffness into the problem.

The primary focus of this paper is on a new formulation for inextensible filaments. In our approach, the fibers are evolved via a rotation of the tangent vector on the unit sphere, and the fiber positions are then obtained by integration. This approach is similar to that of [34], but unique because we consider the fiber as a continuum, rather than a collection of discrete line segments. In this way, we maintain strict inextensibility of the fibers without introducing a penalty parameter. To close our formulation, we treat the force due to tension as a Lagrange multiplier and enforce the principle that the constraint forces do no work [42]. We couple this advance with recent techniques [1, 81] for efficient evaluation of nonlocal integrals appearing in SBT to develop a method that is both accurate and robust. In our spectrally accurate numerical method, we use, as in [65], Chebyshev polynomials of relatively low degree $(16 - 32)$ to represent the fiber centerlines *and*

the force densities acting on them. This assumes that all of these quantities are smooth enough to be represented in the spectral basis. This assumption fails for truly cylindrical fibers near the endpoints, and also when the fibers experience localized forces such as those due to steric repulsion, electrostatics, friction, and molecular motors. In our concluding Section 7, we discuss some possible ways to extend our method to account for these important biophysical forces and fiber shapes.

The rest of this paper is laid out as follows. We begin in Section 2 by introducing the necessary SBT equations for both local and nonlocal hydrodynamics. We modify the classical SBT formulation [38, 36, 26] to regularize the local drag coefficient for cylindrical fibers and account for inter-fiber interactions through the RPY tensor. In Section 3, we parameterize inextensible motions of the fiber as rotations of the unit tangent vector, thus strictly enforcing inextensibility. We then discuss how to determine the Lagrange multiplier forces for inextensibilty by imposing the principle of virtual work in a weak L^2 sense. Section 4 is devoted to numerical methods. We show how to incorporate a fast method for evaluating far-field hydrodynamic interactions (positively-split Ewald summation [21]), and how to use specialized quadrature schemes for accurate evaluation of finite part and near-fiber quadratures. In Section 4.5, we design a semi-implicit, second-order temporal integrator that treats bending elasticity implicitly, yet for dilute systems only requires solving a block-diagonal linear system with a single evaluation of the nonlocal hydrodynamics per time step. For more concentrated systems, we use GMRES to solve a dense linear system, but show in Section 5.2 that at most a few iterations are needed per time step to maintain stability. In the other numerical tests of Section 5, we also show how our "weak formulation" of inextensibility gives improved spatial accuracy over the traditional "strong formulation" of inextensibility in [84, 65]. In Section 6, we study the rheology of a cross-linked network of filaments in oscillatory shear by introducing cross-linkers into the SBT formulation. Section 7 gives our conclusions and discusses future work.

2 Slender body theory

We begin here by summarizing the slender body theories of [38, 36, 26], here following in particular Johnson [36] and Götz [26]. These SBTs derive a global fluid velocity due to a single slender fiber, then evaluate this velocity asymptotically on the fiber surface to obtain a fiber evolution equation. It remains an open question, however, how to efficiently evaluate the fluid velocity generated by one filament on another filament. Here we formulate a modified treatment of these fiber-fiber interactions that is more physical and motivated by our observation in Appendix A that classical

SBT can be formulated in terms of a line integral involving the RPY kernel. In this section we will not consider any time dependence and look at the velocity of the filament and the Stokes fluid at a specific instant in time. We therefore omit explicit time dependence in our notation for simplicity.

2.1 Single filament

We denote with $\boldsymbol{X}(s)$ the position of the centerline of a filament, parameterized by arclength $s \in [0, L]$, where L is the fiber length. The tangent vector is $\boldsymbol{\tau}(s) = \partial \boldsymbol{X}/\partial s$ and has unit length. The fiber has physical radius $a(s) = r\rho(s)$, where $0 \le \rho(s) \le 1$, and slenderness ratio $\epsilon = r/L$. Let the force per unit length on the fiber centerline be denoted by $\boldsymbol{f}(s)$ and the background flow (e.g. shear flow) at an arbitrary point in the fluid be denoted by $\boldsymbol{u}_0(\boldsymbol{x})$.

We recall the Stokeslet and doublet (Laplacian of the Stokeslet) kernels, which are the fundamental solutions to the Stokes equations for a point force and mass source dipole, respectively. If we center the kernels at $\boldsymbol{x}_0$, take $\boldsymbol{x}$ to be an arbitrary point in the fluid, and introduce $\boldsymbol{R} = \boldsymbol{x} - \boldsymbol{x}_0$ with $\hat{\boldsymbol{R}} = \boldsymbol{R}/\|\boldsymbol{R}\|$, we have that

$$\boldsymbol{S}(\boldsymbol{x}, \boldsymbol{x}_0) = \frac{\boldsymbol{I} + \hat{\boldsymbol{R}}\hat{\boldsymbol{R}}}{\|\boldsymbol{R}\|} \qquad \text{and} \qquad \boldsymbol{D}(\boldsymbol{x}, \boldsymbol{x}_0) = \frac{\boldsymbol{I} - 3\hat{\boldsymbol{R}}\hat{\boldsymbol{R}}}{\|\boldsymbol{R}\|^3}. \tag{1}$$

The idea of SBT is to introduce an ansatz for the flow field away from the fiber centerline of the form

$$\boldsymbol{u}(\boldsymbol{x}) - \boldsymbol{u}_0(\boldsymbol{x}) = \frac{1}{8\pi\mu} \int_0^L \left(\boldsymbol{S}(\boldsymbol{x}, \boldsymbol{X}(s)) + \beta(s)\boldsymbol{D}(\boldsymbol{x}, \boldsymbol{X}(s)) \right) \boldsymbol{f}(s)\, ds \tag{2}$$

$$:= \frac{1}{8\pi\mu} \int_0^L \boldsymbol{S}_D(\boldsymbol{x}, \boldsymbol{X}(s); \beta(s))\, \boldsymbol{f}(s)\, ds, \tag{3}$$

where μ is the fluid viscosity. In Eq. (3), we have defined a kernel $\boldsymbol{S}_D(\boldsymbol{x}, \boldsymbol{x}_0; \beta(s))$ that is a combination of a Stokeslet and a doublet with strength β. Using the method of matched asymptotic expansions, the velocity integral (2) can be computed analytically on the surface of the fiber to $\mathcal{O}(\epsilon)$ (see [26, 41] for details on these integrals). The value of β comes from imposing the boundary condition that the velocity on the fiber surface be constant to $\mathcal{O}(\epsilon)$; Mori et al. [62, 63] refer to this as the "fiber integrity condition." For cylindrical [26] or ellipsoidally-tapered [36] filaments, this yields the solution for the velocity in the fluid as

$$\boldsymbol{u}(\boldsymbol{x}) - \boldsymbol{u}_0(\boldsymbol{x}) = \frac{1}{8\pi\mu} \int_0^L \boldsymbol{S}_D\left(\boldsymbol{x}, \boldsymbol{X}(s); \frac{(\epsilon L)^2}{2}\right) \boldsymbol{f}(s)\, ds. \tag{4}$$

The fluid velocity $\boldsymbol{u}(\boldsymbol{x})$ in (4) does not apply inside of the fiber volume; in fact the kernel $\boldsymbol{S}_D$ in (4) is not even defined on the fiber centerline. Physically, however, the velocity of the fiber centerline,

7

which we denote with $\boldsymbol{U}(s)$, should be equal to the average of $\boldsymbol{u}(\boldsymbol{x})$ around a ring cross section of the fiber with radius $a(s)$. Equivalently, since the function $\boldsymbol{u}(\boldsymbol{x})$ is constant on the cross section surface to $\mathcal{O}(\epsilon)$, averaging $\boldsymbol{u}(\boldsymbol{x})$ is equivalent to throwing out all terms in its expansion of $\mathcal{O}(\epsilon)$ or higher. Another approach, which is based on the Rotne-Prager-Yamakawa (RPY) kernel and matched asymptotics, is presented in Appendix A. The RPY tensor approximates the hydrodynamic interaction between two spheres of radius b centered at $\boldsymbol{x}$ and $\boldsymbol{y}$ with the kernel [73, 87, 21]

$$8\pi\mu\boldsymbol{S}_{\mathrm{RPY}}\left(\boldsymbol{x},\boldsymbol{y};b\right) = \begin{cases} \boldsymbol{S}_D\left(\boldsymbol{x},\boldsymbol{y};2b^2/3\right) & r > 2b \\ \left(\frac{4}{3b} - \frac{3r}{8b^2}\right)\boldsymbol{I} + \frac{1}{8b^2r}\boldsymbol{r}\boldsymbol{r} & r < 2b \end{cases}, \tag{5}$$

where $\boldsymbol{r} = \boldsymbol{x} - \boldsymbol{y}$, $r = \|\boldsymbol{r}\|$, and we set the sphere radius (see (A.27)) to

$$b = \epsilon L \frac{e^{3/2}}{4} \approx 1.12\epsilon L. \tag{6}$$

We express the velocity $\boldsymbol{U}(s)$ on the fiber as a line integral of the RPY kernel

$$\boldsymbol{U}\left(s\right) - \boldsymbol{u}_0\left(\boldsymbol{X}(s)\right) := \frac{1}{8\pi\mu}\int_0^L \boldsymbol{S}_{\mathrm{RPY}}\left(\boldsymbol{X}(s),\boldsymbol{X}\left(s'\right);2b^2/3\right)\boldsymbol{f}\left(s'\right)\,ds'. \tag{7}$$

Using matched asymptotics to approximate (7) for slender fibers (see Appendix A), we obtain the same result as classical SBT [26]

$$\boldsymbol{U}(s) - \boldsymbol{u}_0\left(\boldsymbol{X}(s)\right) = \boldsymbol{M}^{\mathrm{L}}\left(\boldsymbol{\tau}(s);c(s)\right)\boldsymbol{f}(s) + \left(\boldsymbol{\mathcal{M}}^{\mathrm{FP}}\left[\boldsymbol{X}(\cdot)\right]\boldsymbol{f}(\cdot)\right)(s), \text{ where} \tag{8}$$

$$\boldsymbol{M}^{\mathrm{L}}(\boldsymbol{\tau};c) = \frac{1}{8\pi\mu}\left(c(\boldsymbol{I} + \boldsymbol{\tau}\boldsymbol{\tau}) + (\boldsymbol{I} - 3\boldsymbol{\tau}\boldsymbol{\tau})\right), \text{ and} \tag{9}$$

$$\left(\boldsymbol{\mathcal{M}}^{\mathrm{FP}}\left[\boldsymbol{X}\right]\boldsymbol{f}\right)(s) = \frac{1}{8\pi\mu}\int_0^L \left(\boldsymbol{S}\left(\boldsymbol{X}(s),\boldsymbol{X}(s')\right)\boldsymbol{f}(s') - \left(\frac{\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|s - s'|}\right)\boldsymbol{f}(s)\right)\,ds'. \tag{10}$$

Here $\boldsymbol{M}^{\mathrm{L}}$ is a 3×3 local drag matrix that gives the velocity contribution from the force density $\boldsymbol{f}$ at points $\mathcal{O}(\epsilon)$ away from $\boldsymbol{X}(s)$. The integral operator $\boldsymbol{\mathcal{M}}^{\mathrm{FP}}\left[\boldsymbol{X}\right]$ gives the contribution from the rest of the fiber in the form of a finite part integral. The first term in the integrand is the Stokeslet, and the second term is the "common" part in the matched asymptotic expansion that comes from expansion of the Stokeslet around $s' = s$. Physically, the finite part integral gives the velocity contribution from forcing at points $\mathcal{O}(1)$ away from $\boldsymbol{X}(s)$. Thus while both terms in the integrand are singular, their difference is finite [84] (see also Section 4.2.1).

In the local drag matrix (9), the leading order local drag coefficient is given by [26]

$$c(s) = \ln\left(\frac{4s(L - s)}{a(s)^2}\right) \tag{11}$$
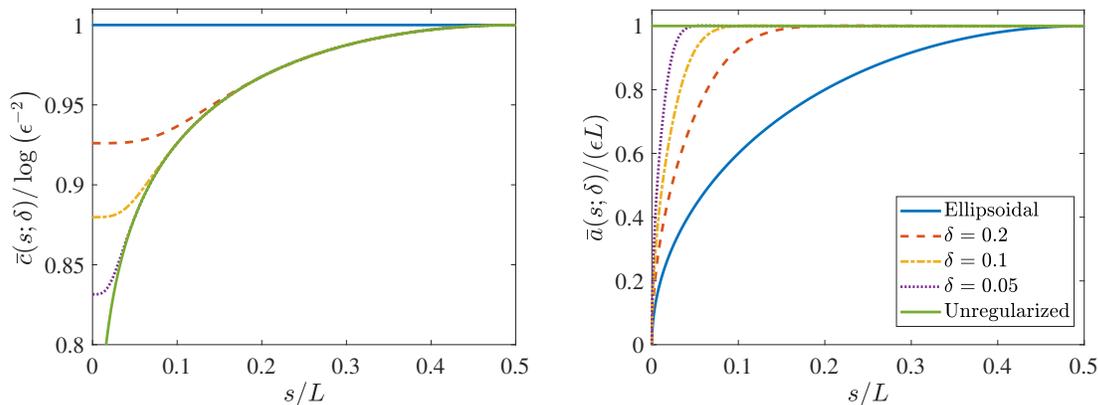
Figure 1: Regularized drag coefficients and effective radius functions. (Left:) The regularized local drag coefficients $\bar{c}(s; \delta)$ for various δ. (Right:) The effective fiber radius $\bar{a}(s; \delta)$ as defined in (14). We show only one side of the fiber, with the other being the former's mirror image about $s/L = 0.5$.

and is singular without proper decay of $a(s)$ at $s = 0$ and $s = L$. Clearly, if $a(s)$ decays near the fiber endpoints as $2\epsilon\sqrt{s(L-s)}$, then the leading order coefficient (11) is finite at the fiber endpoints [36]. This fact is the basis for a general assumption across the SBT literature that the filaments have ellipsoidal shape, so that in most studies $c(s) = -\ln(\epsilon^2)$ is constant for all s [84, 65, 19].

Actin filaments are best modeled as cylinders with constant radius, so that $a(s) = r = \epsilon L$ on $s \in [0, L]$. In this case, the coefficient (11) becomes singular at the filament ends, and so we modify the local drag coefficient by effectively tapering the fiber radius over a distance $\sim \delta L$ near the endpoints. Specifically, to regularize $c(s)$, we set $\eta = 2s/L - 1$, so that $\eta \in [-1, 1]$, and compute a weight function

$$w(s; \delta) = \tanh\left(\frac{\eta(s) + 1}{\delta}\right) - \tanh\left(\frac{\eta(s) - 1}{\delta}\right) - 1, \tag{12}$$

which is 1 near the fiber center ($\eta = 0$) and zero at the fiber ends ($\eta = \pm 1$). We then assign to each s a regularized fiber centerline coordinate by

$$\bar{s}(s; \delta) = w(s; \delta)s + \left(1 - w(s; \delta)^2\right)\frac{\delta L}{2} \tag{13}$$

on $0 \leq s \leq L/2$, with the corresponding reflection for $s > L/2$. The regularized coefficient for a given δ is then given by

$$\bar{c}(s; \delta) := c(\bar{s}(s; \delta)) = \ln\left(\frac{4\bar{s}(L - \bar{s})}{(\epsilon L)^2}\right) := \ln\left(\frac{4s(L - s)}{\bar{a}(s; \delta)^2}\right). \tag{14}$$

Figure 1 shows how the choice of δ impacts the local drag coefficient $\bar{c}(s)$ and effective radius function $\bar{a}(s)$. We see that the fibers are cylindrical ($\bar{a}(s) = \epsilon L$) for $s/L \gtrsim \delta$, while for $s/L \lesssim \delta$

the effective radius smoothly decays to zero, as it would for an ellipsoidal fiber. Larger values of δ yield smoother radius functions and smoother local drag coefficients. Throughout this paper, we will use $\delta = 0.1$, unless otherwise stated.

2.2 Multiple filaments

It remains to include in the fiber centerline velocity (8) the perturbed flow due to other filaments, i.e., to account for hydrodynamic interactions between fibers. We require more involved notation in this case to distinguish between fibers. In general, we use the symbol $\boldsymbol{X}$ in an equation whenever it is localized to a single fiber and is the same on all fibers. For example, $\mathcal{H}[\boldsymbol{X}] = 0$ implies that $\mathcal{H}$ is a functional of a single fiber's position and is zero on every fiber individually. When multiple fibers are involved, we index the ith fiber by the superscript $\boldsymbol{X}^{(i)}$. For example, $\mathcal{H}\left[\boldsymbol{X}^{(i)}, \boldsymbol{X}^{(j)}\right] \boldsymbol{f}^{(i)}$ implies that $\mathcal{H}$ is a functional of a pair of fiber positions which acts on the force density on fiber i. Whenever an equation applies to one fiber, but involves all other fibers, we will use the notation $\boldsymbol{X}^{(i)}$ for the single fiber and $\boldsymbol{X}$ to refer to the collection of all fibers. Our use of $\boldsymbol{X}$ to refer to both a general fiber and collection of fibers is a slight abuse of notation, but the meaning should be clear from the context, specifically whether an equation involves a single fiber or multiple fibers.

The simplest approach for including hydrodynamic interactions in SBT is the one taken by Tornberg and Shelley [84], in which the fluid velocity due to one fiber (4) is simply evaluated on the centerline of the other fibers. Nazockdast et al. also adopted this, except they dropped the doublet term completely and included only the Stokeslet term. Inspired by the fact that the classical SBT for a single fiber (8) can be reformulated as a regularized singularity method using the RPY line integral (7), we use the same line integral for interaction with other fibers. In (A.28), we substitute the choice of RPY radius (6) into the RPY kernel (5) to obtain the inter-fiber interaction velocity

$$\boldsymbol{v}^{(j)}\left(\boldsymbol{X}^{(i)}(s)\right) := \frac{1}{8\pi\mu} \int_0^L \boldsymbol{S}_D\left(\boldsymbol{X}^{(i)}(s), \boldsymbol{X}^{(j)}(s'), \frac{e^3}{24}(\epsilon L)^2\right) \boldsymbol{f}^{(j)}(s')\, ds'$$
$$:= \left(\boldsymbol{\mathcal{M}}^c\left[\boldsymbol{X}^{(j)}(\cdot)\right] \boldsymbol{f}^{(j)}(\cdot)\right)\left(\boldsymbol{X}^{(i)}(s)\right) \tag{15}$$

Here we have again defined a linear integral operator $\boldsymbol{\mathcal{M}}^c$ which acts on $\boldsymbol{f}^{(j)}$ to give the velocity on the centerline of filament i solely due to filament j. More generally, we will denote by $\boldsymbol{v}^{(j)}(\boldsymbol{x})$ the velocity induced by filament j at any point $\boldsymbol{x}$ on the centerline of any other filament. This "filament interaction velocity" differs from the slender body fluid velocity (4) in that the coefficient of the dipole term is $e^3/24 \approx 0.84$ instead of $1/2$ (see (A.28)). Since $\boldsymbol{v}^{(j)}(\boldsymbol{x})$ can be evaluated everywhere

in the fluid, we use a lowercase letter to denote it and distinguish it from the *fluid* velocity $\boldsymbol{u}^{(j)}$ induced by fiber j, which is given in (4).

We have used a constant radius b when defining the inter-fiber mobility (15) using the RPY kernel. While our use of the regularized local drag coefficient (14) implies that the fibers are tapered at the endpoints, and therefore that b decays to zero at the endpoints, we will use a constant radius for nonlocal hydrodynamics. This assumption, together with the assumption that all of the fibers have the same maximum radius ϵL, can both be relaxed, as the RPY kernels derived in [96] and associated fast methods [23] can be used to generalize (15) to tapered filaments or fibers with different radii.

Summing the interaction kernel (15) over filaments $j \neq i$ and adding the terms from local drag, we get a slender body theory for the velocity $\boldsymbol{U}^{(i)}(s)$ at position s on filament i,

$$\boldsymbol{U}^{(i)}(s) - \boldsymbol{u}_0\left(\boldsymbol{X}^{(i)}(s)\right) = \boldsymbol{M}^{\mathrm{L}}\left(\boldsymbol{\tau}^{(i)}(s); c(s)\right)\boldsymbol{f}^{(i)}(s) \tag{16}$$
$$+ \left(\boldsymbol{\mathcal{M}}^{\mathrm{FP}}\left[\boldsymbol{X}^{(i)}(\cdot)\right]\boldsymbol{f}^{(i)}(\cdot)\right)(s) + \sum_{j \neq i}\left(\boldsymbol{\mathcal{M}}^{\mathrm{c}}\left[\boldsymbol{X}^{(j)}(\cdot)\right]\boldsymbol{f}^{(j)}(\cdot)\right)\left(\boldsymbol{X}^{(i)}(s)\right),$$

where $\boldsymbol{M}^{\mathrm{L}}$ is defined in Eq. (9), $\boldsymbol{\mathcal{M}}^{\mathrm{FP}}$ is defined in Eq. (10), and $\boldsymbol{\mathcal{M}}^{\mathrm{c}}$ is defined in Eq. (15). Because $\boldsymbol{U}^{(i)}$ is only defined on the centerline of filament i (and not everywhere in the fluid), we denote it with a capital letter.

For a single fiber i, we will write the mobility (16) abstractly as

$$\boldsymbol{U}^{(i)}(s) - \boldsymbol{u}_0\left(\boldsymbol{X}^{(i)}(s)\right) = \sum_j\left(\boldsymbol{\mathcal{M}}_{ij}\left[\boldsymbol{X}\right]\boldsymbol{f}^{(j)}\right)(s), \tag{17}$$

where the mobility operator $\boldsymbol{\mathcal{M}}\left[\boldsymbol{X}\right]$ is a functional of the positions of all fibers. We will compactly write the velocity (17) for the entire collection of fibers as

$$\boldsymbol{U} - \boldsymbol{u}_0\left(\boldsymbol{X}\right) := \boldsymbol{\mathcal{M}}\left[\boldsymbol{X}\right]\boldsymbol{f}. \tag{18}$$

This mobility equation can be closed by defining a constitutive equation for the fiber force densities $\boldsymbol{f}$, which we do next.

3 Inextensible filaments

In this paper, we consider inextensible filaments $\boldsymbol{X}^{(i)}(s, t)$ which can bend, but not stretch, as they evolve in time. We assume the fibers are in a constant twist-free equilibrium, since in the absence of

externally-applied or internally-generated torques the timescale of twist relaxation is $\mathcal{O}(\epsilon^{-2})$ faster than bending [69].

At every instant in time, each fiber resists bending with bending force density (per unit length) $\boldsymbol{f}^{\kappa}[\boldsymbol{X}]$. Inextensibility can be enforced by introducing Lagrange multiplier force densities on each fiber $\boldsymbol{\lambda}^{(i)}(s,t)$, where we will again write $\boldsymbol{\lambda} = \left\{\boldsymbol{\lambda}^{(i)}\right\}$ whenever we refer to the collection of Lagrange multipliers on all fibers. Thus the PDE that we need to solve on every fiber is given by (using the abstract notation of (18)),

$$\frac{\partial \boldsymbol{X}}{\partial t} - \boldsymbol{u}_0(\boldsymbol{X}, t) = \boldsymbol{\mathcal{M}}[\boldsymbol{X}](\boldsymbol{f}^{\kappa}[\boldsymbol{X}] + \boldsymbol{\lambda}), \tag{19}$$

where the mobility operator $\boldsymbol{\mathcal{M}}$ is defined in (16) and the background flow function $\boldsymbol{u}_0$ can in general vary in time. The fibers are also constrained to be inextensible, so that for every fiber

$$\boldsymbol{\tau}(s,t) \cdot \boldsymbol{\tau}(s,t) = 1, \tag{20}$$

for all s and t. We still need to specify boundary conditions for the evolution equation (19) and additional conditions on $\boldsymbol{\lambda}$ to make the solution unique, as we explain shortly.

3.1 Bending elasticity

For fiber mechanics, we use the Euler beam model, in which the bending force density on every fiber is given by

$$\boldsymbol{f}^{\kappa}[\boldsymbol{X}] = -\kappa \boldsymbol{X}_{ssss} := \boldsymbol{\mathcal{F}} \boldsymbol{X}, \tag{21}$$

where the constant linear operator $\boldsymbol{\mathcal{F}}$ gives $\boldsymbol{f}^{\kappa}$ taking into account the "free fiber" boundary conditions [84]

$$\boldsymbol{X}_{ss}(s = 0, t) = \boldsymbol{X}_{sss}(s = 0, t) = \boldsymbol{0}, \tag{22}$$

$$\boldsymbol{X}_{ss}(s = L, t) = \boldsymbol{X}_{sss}(s = L, t) = \boldsymbol{0}.$$

Again, because the boundary conditions (22) apply to every fiber without dependence on other fibers, we use the notation $\boldsymbol{X}_{ss}$ and $\boldsymbol{X}_{sss}$ to refer to the arclength derivatives along the fiber and drop the superscript (i).

It is easy to see that the boundary conditions (22) cause the total force and torque on every

fiber due to $\boldsymbol{f}^\kappa$ to be zero,

$$\int_0^L \boldsymbol{f}^\kappa \, ds = -\kappa \boldsymbol{X}_{sss}\Big|_0^L = \boldsymbol{0}, \quad \text{and} \tag{23}$$

$$\left(\int_0^L \boldsymbol{f}^\kappa \times \boldsymbol{X} \, ds\right)^\ell = -\kappa \int_0^L \left(X^j X^k_{ssss} - X^k X^j_{ssss}\right) ds = -\kappa \int_0^L \left(X^j_{ss} X^k_{ss} - X^k_{ss} X^j_{ss}\right) = 0. \tag{24}$$

Here the set of superscripts (j, k, ℓ) denote vector components and are a cyclic permutation of $(1, 2, 3)$. In the torque equation, the free fiber boundary conditions lead to the cancellation of boundary terms that arise in integration by parts.

3.2 Traditional formulation of inextensibility

In the traditional formulation of inextensibility [84], the inextensibility constraint (20) is differentiated with respect to time. Then, s and t derivatives are interchanged to yield

$$\left(\frac{\partial \boldsymbol{X}}{\partial t}\right)_s \cdot \boldsymbol{\tau} = 0. \tag{25}$$

In [84], the system was closed by substituting the mobility equation (19) into the differentiated inextensibility constraint (25). On each fiber, Tornberg and Shelley then assume that $\boldsymbol{\lambda}^{(i)} = \left(T^{(i)} \boldsymbol{\tau}^{(i)}\right)_s$, where $T^{(i)}(s, t)$ is an unknown scalar tension [84]. This results in the *line tension equation*,

$$\frac{\partial}{\partial s}\left(\boldsymbol{u}_0\left(\boldsymbol{X}^{(i)}\right) + \sum_j \boldsymbol{\mathcal{M}}_{ij}[\boldsymbol{X}]\left(\boldsymbol{\mathcal{F}}\boldsymbol{X}^{(j)} + \left(T^{(j)} \boldsymbol{\tau}^{(j)}\right)_s\right)\right) \cdot \boldsymbol{\tau}^{(i)} = 0. \tag{26}$$

which holds for each fiber i [84]. While the second-order BVP (26) is linear in T, it is highly nonlinear in $\boldsymbol{X}$, since the operation $\boldsymbol{\mathcal{F}}\boldsymbol{X}$ gives fourth derivatives of $\boldsymbol{X}$. Even in the absence of any nonlocal hydrodynamic interactions (i.e. if $\boldsymbol{\mathcal{M}} = \boldsymbol{M}^\mathrm{L}$) and zero background flow ($\boldsymbol{u}_0 = \boldsymbol{0}$), the line tension equation still has terms of the form $\boldsymbol{X}_{sss} \cdot \boldsymbol{X}_{sss}$ (see [84, Eq. (13)]), which lead to aliasing errors in spectral numerical methods. Because the line tension equation (26) enforces inextensibility pointwise along the fiber, we refer to it as a *strong formulation of inextensibility*.

3.3 Kinematics of inextensible fibers

In our approach, we evolve the tangent vector $\boldsymbol{\tau}(s, t)$, rather than $\boldsymbol{X}(s, t) = \boldsymbol{X}(0, t) + \int_0^s \boldsymbol{\tau}(s', t) \, ds'$. Considering the evolution of $\boldsymbol{\tau}(s, t)$, the differentiated inextensibility constraint (25) implies that, for every fiber,

$$\frac{\partial \boldsymbol{\tau}}{\partial t}(s, t) = \boldsymbol{\Omega}(s, t) \times \boldsymbol{\tau}(s, t), \tag{27}$$

i.e., that the fiber evolution can be thought of as rotations of $\boldsymbol{\tau}$ on the unit sphere.

At each fiber point, we uniquely define an orthonormal coordinate system using spherical angles $\theta(s,t)$ and $\phi(s,t)$. We represent the unit tangent vector $\boldsymbol{\tau}(s,t)$ as

$$\boldsymbol{\tau}(s,t) = \begin{pmatrix} \cos\theta\cos\phi \\ \sin\theta\cos\phi \\ \sin\phi \end{pmatrix}, \tag{28}$$

where we define θ to be single-valued at $\phi = \pi/2$ by setting $\theta\,(\phi = \pm\pi/2) = 0$. A choice of normal vectors that are always orthonormal to $\boldsymbol{\tau}$ on the unit sphere is

$$\boldsymbol{n}_1 = \begin{pmatrix} -\sin\theta \\ \cos\theta \\ 0 \end{pmatrix}, \qquad \boldsymbol{n}_2 = \begin{pmatrix} -\cos\theta\sin\phi \\ -\sin\theta\sin\phi \\ \cos\phi \end{pmatrix}. \tag{29}$$

Because $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ can be determined uniquely from $\boldsymbol{\tau}$, we denote them henceforth with $\boldsymbol{n}_j\,(\boldsymbol{\tau}(s,t))$, for $j = 1, 2$. Since θ is single-valued at $\phi = \pi/2$, each component of the orthonormal coordinate system $(\boldsymbol{\tau}, \boldsymbol{n}_1, \boldsymbol{n}_2)$ is a smooth function of $\boldsymbol{X}$ when $\boldsymbol{\tau}$ is smooth. Importantly, our method does not depend on the particular choice of normal vectors (29); any choice that gives smooth $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$ for a smooth $\boldsymbol{X}$ is equally acceptable. For example, the Frenet or Bishop frames could be used [43].

Because $\boldsymbol{\tau} \times \boldsymbol{\tau} = \boldsymbol{0}$, and since we are not considering twist, $\boldsymbol{\Omega}(s,t)$ can be restricted to linear combinations of $\boldsymbol{n}_1$ and $\boldsymbol{n_2}$. We let

$$\boldsymbol{\Omega}(s,t) := \boldsymbol{\Omega}\,(\boldsymbol{\tau}(s,t), \boldsymbol{g}(s,t)) := g_1(s,t)\boldsymbol{n}_2\,(\boldsymbol{\tau}(s,t)) - g_2(s,t)\boldsymbol{n}_1\,(\boldsymbol{\tau}(s,t))\,, \tag{30}$$

where $g_1(s,t)$ and $g_2(s,t)$ are two specific unknown functions and $\boldsymbol{g} = \{g_1, g_2\}$. Equation (30) implies that, by the right-handedness of the coordinate system $(\boldsymbol{\tau}, \boldsymbol{n}_1, \boldsymbol{n}_2)$,

$$\frac{\partial\boldsymbol{\tau}}{\partial t} = \boldsymbol{\Omega} \times \boldsymbol{\tau} = g_1\boldsymbol{n}_1 + g_2\boldsymbol{n}_2. \tag{31}$$

Any inextensible velocity of the fiber centerline can now be written in the form

$$\boldsymbol{U}(s,t) = \frac{\partial\boldsymbol{X}}{\partial t}(s,t) = \overline{\boldsymbol{U}}(t) + \int_0^s \sum_{j=1}^2 g_j(s',t)\boldsymbol{n}_j\,(\boldsymbol{\tau}(s',t))\,ds', \tag{32}$$

where $\overline{\boldsymbol{U}}(t) = \partial\boldsymbol{X}/\partial t(s = 0, t)$ is a rigid body translation.

3.4   Principle of virtual work

The kinematic formulation of Section 3.3 can still be used to solve for the line tensions and fiber velocities. In particular, by substituting the inextensible velocity (32) into the left hand side of the evolution equation (19) and setting $\boldsymbol{\lambda} = (T\boldsymbol{\tau})_s$, a PDE results with unknowns g_1, g_2, $\overline{\boldsymbol{U}}$, and T. We choose to close our formulation differently, in the process eliminating the need to solve for tension explicitly.

On every fiber, the principle of virtual work states that the constraint forces $\boldsymbol{\lambda}$ do no work for any choice of g_1, g_2, and $\overline{\boldsymbol{U}}$ [42]. Because this constraint holds for all time, for simplicity we drop for the moment the explicit dependence on t in the notation. To impose the principle of virtual work, we use the L^2 inner product to compute the total power dissipated in the fluid from $\boldsymbol{\lambda}$,

$$\mathcal{P} = \left\langle \boldsymbol{\lambda}, \frac{\partial \boldsymbol{X}}{\partial t} \right\rangle = \int_0^L ds' \left(\overline{\boldsymbol{U}} + \int_0^{s'} (g_1(s)\boldsymbol{n}_1\left(\boldsymbol{\tau}(s)\right) + g_2(s)\boldsymbol{n}_2\left(\boldsymbol{\tau}(s)\right)) \, ds \right) \cdot \boldsymbol{\lambda}\left(s'\right). \tag{33}$$

Changing integration variables, we can rewrite this as

$$\mathcal{P} = \overline{\boldsymbol{U}} \cdot \int_0^L \boldsymbol{\lambda}\left(s'\right) \, ds' + \int_0^L ds \int_s^L (g_1(s)\boldsymbol{n}_1\left(\boldsymbol{\tau}(s)\right) + g_2(s)\boldsymbol{n}_2\left(\boldsymbol{\tau}(s)\right)) \cdot \boldsymbol{\lambda}(s') \, ds' \tag{34}$$

$$= \overline{\boldsymbol{U}} \cdot \int_0^L \boldsymbol{\lambda}\left(s'\right) \, ds' + \int_0^L (g_1(s)\boldsymbol{n}_1\left(\boldsymbol{\tau}(s)\right) + g_2(s)\boldsymbol{n}_2\left(\boldsymbol{\tau}(s)\right)) \cdot \left(\int_s^L \boldsymbol{\lambda}(s') \, ds' \right) ds = 0. \tag{35}$$

3.4.1   Pointwise formulation

Since the principle of virtual work (35) must hold for any inextensible motion, it must hold for all $\overline{\boldsymbol{U}}$ and all sufficiently smooth g_1 and g_2. Therefore, we must have, for all s,

$$\begin{pmatrix} \left(\int_s^L \boldsymbol{\lambda}(s') \, ds' \right) \cdot \boldsymbol{n}_1\left(\boldsymbol{\tau}(s)\right) \\ \left(\int_s^L \boldsymbol{\lambda}(s') \, ds' \right) \cdot \boldsymbol{n}_2\left(\boldsymbol{\tau}(s)\right) \\ \int_0^L \boldsymbol{\lambda}(s') ds' \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \boldsymbol{0} \end{pmatrix}. \tag{36}$$

The first and second components of the constraints (36) taken together tell us that $\int_s^L \boldsymbol{\lambda}\left(s'\right) \, ds'$ is orthogonal to both normal vectors. Therefore, $\int_s^L \boldsymbol{\lambda}\left(s'\right) \, ds'$ is in the direction of $\boldsymbol{\tau}(s)$ and can be written as

$$\int_s^L \boldsymbol{\lambda}\left(s'\right) \, ds' = -T(s)\boldsymbol{\tau}(s), \tag{37}$$

for some scalar function $T(s)$ with $T(s = L) = 0$. This gives

$$\boldsymbol{\lambda}(s) = (T(s)\boldsymbol{\tau}(s))_s, \tag{38}$$

which is the form assumed in Tornberg and Shelley [84]. Thus our derivation shows that the form of $\boldsymbol{\lambda}$ taken in [84] is equivalent to the principle that the constraint forces perform no virtual work, if the work is given by the standard L^2 inner product [42].

Now, returning to the third of the constraints (36), $\int_0^L \boldsymbol{\lambda}(s) \, ds = \mathbf{0}$, and substituting the derived form of $\boldsymbol{\lambda}$ in (38), we obtain

$$T(L)\boldsymbol{\tau}(L) - T(0)\boldsymbol{\tau}(0) = \mathbf{0}. \tag{39}$$

Since $T(L) = 0$, Eq. (39) implies that $T(0) = 0$ as well, since neither of the tangent vectors is identically 0. So we obtain $T(0) = T(L) = 0$, which is exactly the boundary condition for the line tension equation in [84]. The form of $\boldsymbol{\lambda} = (T\boldsymbol{\tau})_s$ and the tension boundary conditions imply that the total torque induced by the constraint forces is zero in continuum, $\int_0^L \boldsymbol{X}(s) \times \boldsymbol{\lambda}(s) \, ds = \mathbf{0}$.

In this sense, the constraint equation (33) is equivalent to the line tension equation used in prior work [84]. Because we showed the equivalence by enforcing constraint (33) for every choice of $g_1(s)$ and $g_2(s)$, we refer to the inextensibility constraint (33) as a *weak formulation of inextensibility*. In the next section, we choose a suitable basis for $g_1(s)$ and $g_2(s)$ to obtain a linear system of equations instead of the pointwise constraint (36).

3.4.2 L^2 weak formulation

In this section, we introduce an L^2 weak formulation that is suitable for a numerical discretization of the weak inextensibility constraint (33). The key idea is to expand the unknown functions $g_1(s)$ and $g_2(s)$ as,

$$g_j(s) = \sum_k \alpha_{jk} T_k(s), \quad \text{for } j = 1, 2, \tag{40}$$

where $T_k(s)$ are sufficiently smooth scalar-valued basis functions for $L^2 : [0, L]$. Substituting the basis function expansion (40) into the inextensible velocity (32), we obtain

$$\boldsymbol{U}(s) = \frac{\partial \boldsymbol{X}}{\partial t}(s) = \overline{\boldsymbol{U}} + \int_0^s \sum_{j=1}^2 \sum_k \alpha_{jk} T_k\left(s'\right) \boldsymbol{n}_j\left(\boldsymbol{\tau}(s')\right) \, ds' := (\mathcal{K}\left[\boldsymbol{X}(\cdot)\right] \boldsymbol{\alpha})(s), \tag{41}$$

where we have defined a linear operator $\mathcal{K}\left[\boldsymbol{X}\right]$ on every fiber that acts on $\boldsymbol{\alpha} = \left(\alpha_{jk}, \overline{\boldsymbol{U}}\right)$ to give an inextensible velocity on the filament centerline (i.e., $\boldsymbol{\alpha}$ parameterizes the space of inextensible fiber motions). Note the functional dependence of $\mathcal{K}$ on $\boldsymbol{X}$ since $\mathcal{K}$ involves the normal vectors $\boldsymbol{n}_1$ and $\boldsymbol{n}_2$. Substituting the inextensible velocity (41) into the dynamical equation (19), we obtain

$$\mathcal{K}\left[\boldsymbol{X}\right] \boldsymbol{\alpha} = \boldsymbol{u}_0\left(\boldsymbol{X}\right) + \mathcal{M}\left[\boldsymbol{X}\right]\left(\mathcal{F}\boldsymbol{X} + \boldsymbol{\lambda}\right). \tag{42}$$

16

This constrained dynamical equation is supplemented by enforcing the principle of virtual work (36) in an L^2 weak sense. We begin by substituting the representation of g_j in (40) into the power equation (33) to obtain, for every fiber,

$$\mathcal{P} = \left\langle \boldsymbol{\lambda}, \mathcal{K}[\boldsymbol{X}]\boldsymbol{\alpha} \right\rangle := \left\langle \mathcal{K}^*[\boldsymbol{X}]\boldsymbol{\lambda}, \boldsymbol{\alpha} \right\rangle \tag{43}$$

$$= \overline{\boldsymbol{U}} \cdot \int_0^L \boldsymbol{\lambda}(s)\, ds + \int_0^L \left(\int_0^s \sum_{j=1}^2 \sum_k \alpha_{jk} T_k\left(s'\right) \boldsymbol{n}_j\left(\boldsymbol{\tau}\left(s'\right)\right)\, ds' \right) \cdot \boldsymbol{\lambda}(s)\, ds = 0,$$

where we have defined $\mathcal{K}^*$ as the L^2 adjoint of $\mathcal{K}$. Since the power from the constraint forces must be zero for any inextensible motion (any $\boldsymbol{\alpha}$), each term of the constraint (43) must be zero. This gives the set of constraints on every fiber

$$\mathcal{K}^*[\boldsymbol{X}]\boldsymbol{\lambda} := \begin{pmatrix} \int_0^L \left(\int_0^s T_k(s')\boldsymbol{n}_1\left(\boldsymbol{\tau}(s')\right)\, ds' \right) \cdot \boldsymbol{\lambda}(s)\, ds \\ \int_0^L \left(\int_0^s T_k(s')\boldsymbol{n}_2\left(\boldsymbol{\tau}(s')\right)\, ds' \right) \cdot \boldsymbol{\lambda}(s)\, ds \\ \int_0^L \boldsymbol{\lambda}(s)\, ds \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \boldsymbol{0} \end{pmatrix}, \tag{44}$$

where the first two constraints hold for all k and the last constraint holds for each of the three Cartesian directions.

3.5 Summary of dynamical equations

In our abstract notation, the evolution of the fiber system can be obtained by solving the following system for $\boldsymbol{\alpha}(t) = \left\{ \alpha_{jk}^{(i)}(t), \overline{\boldsymbol{U}}^{(i)}(t) \right\}$ and $\boldsymbol{\lambda} = \left\{ \boldsymbol{\lambda}^{(i)}(s,t) \right\}$,

$$\frac{\partial \boldsymbol{X}}{\partial t} = \mathcal{K}[\boldsymbol{X}]\boldsymbol{\alpha} = \boldsymbol{u}_0(\boldsymbol{X}, t) + \mathcal{M}[\boldsymbol{X}](\mathcal{F}\boldsymbol{X} + \boldsymbol{\lambda}) \tag{45}$$

$$\mathcal{K}^*[\boldsymbol{X}]\boldsymbol{\lambda} = \boldsymbol{0}. \tag{46}$$

The first equation (45) is the mobility equation. The left hand side is the velocity of a fiber centerline, restricted to the space of inextensible motions via the operator $\mathcal{K}$ defined in (41). The right hand side involves all fiber positions and force densities because of hydrodynamic interactions. The second equation (46) is the principle of virtual work and applies on each fiber separately, $\mathcal{K}^*\left[\boldsymbol{X}^{(i)}\right]\boldsymbol{\lambda}^{(i)} = \boldsymbol{0}$.

On a single fiber i, the mobility equation (45) takes the explicit form

$$\frac{\partial \boldsymbol{X}^{(i)}}{\partial t}(s,t) = \overline{\boldsymbol{U}}^{(i)}(t) + \int_0^s \sum_{j=1}^2 \sum_k \alpha_{jk}^{(i)}(t) T_k\left(s'\right) \boldsymbol{n}_j\left(\boldsymbol{\tau}(s',t)\right)\, ds' = \boldsymbol{u}_0\left(\boldsymbol{X}^{(i)}(s),t\right) \tag{47}$$

$$+\frac{1}{8\pi\mu}\left(\left(\bar{c}(s;\delta)\left(\boldsymbol{I}+\boldsymbol{\tau}^{(i)}(s,t)\boldsymbol{\tau}^{(i)}(s,t)\right) + \left(\boldsymbol{I} - 3\boldsymbol{\tau}^{(i)}(s,t)\boldsymbol{\tau}^{(i)}(s,t)\right)\right)\boldsymbol{f}^{(i)}(s,t)\right. \tag{48}$$

$$+\int_0^L\left(\boldsymbol{S}\left(\boldsymbol{X}^{(i)}(s,t),\boldsymbol{X}^{(i)}\left(s',t\right)\right)\boldsymbol{f}^{(i)}\left(s',t\right) - \left(\frac{\boldsymbol{I}+\boldsymbol{\tau}^{(i)}(s,t)\boldsymbol{\tau}^{(i)}(s,t)}{|s-s'|}\right)\boldsymbol{f}^{(i)}(s,t)\right)ds' \tag{49}$$

$$+\sum_{j\neq i}\int_0^L\left.\left(\boldsymbol{S}\left(\boldsymbol{X}^{(i)}(s,t),\boldsymbol{X}^{(j)}\left(s',t\right)\right) + \frac{e^3}{24}(\epsilon L)^2\,\boldsymbol{D}\left(\boldsymbol{X}^{(i)}(s,t),\boldsymbol{X}^{(j)}\left(s',t\right)\right)\right)\boldsymbol{f}^{(j)}\left(s',t\right)ds'\right),$$

$$\tag{50}$$

with $\boldsymbol{f}^{(i)}(s,t) = \boldsymbol{\mathcal{F}}\boldsymbol{X}^{(i)}(s,t) + \boldsymbol{\lambda}^{(i)}(s,t)$. The Stokeslet and doublet kernels $\boldsymbol{S}$ and $\boldsymbol{D}$ are defined in (1), and the local drag coefficient $\bar{c}$ is regularized at the endpoints as defined in (14). The mobility equation for fiber i is supplemented by the principle of virtual work (46) which is localized to fiber i and takes the explicit form

$$\int_0^L\left(\int_0^s T_k(s')\boldsymbol{n}_j\left(\boldsymbol{\tau}^{(i)}(s',t)\right)ds'\right)\cdot\boldsymbol{\lambda}^{(i)}(s,t)\,ds = 0, \quad \forall k \text{ and } j = 1,2, \tag{51}$$

$$\int_0^L \boldsymbol{\lambda}^{(i)}(s,t)\,ds = \boldsymbol{0}. \tag{52}$$

4 Numerical Methods

Our goal in this section is to write the evolution equations (45) and (46) in the form of a block-matrix saddle point system. We will replace the operators with matrices and the position functions $\boldsymbol{X}^{(i)}(s)$ with discrete vectors of collocation points $\boldsymbol{X}^{(i)}$. The fiber evolution is then given by

$$\frac{\partial \boldsymbol{X}^{(i)}}{\partial t}(t) = \boldsymbol{K}\left(\boldsymbol{X}^{(i)}\right)\boldsymbol{\alpha}^{(i)}(t). \tag{53}$$

The coefficients $\boldsymbol{\alpha} = \left\{\boldsymbol{\alpha}^{(i)}\right\}$ can be determined by solving a saddle point system of the form

$$\begin{pmatrix} -\boldsymbol{M}\left(\boldsymbol{X}\right) & \boldsymbol{K}\left(\boldsymbol{X}\right) \\ \boldsymbol{K}^*\left(\boldsymbol{X}\right) & \boldsymbol{0} \end{pmatrix}\begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\alpha} \end{pmatrix} = \begin{pmatrix} \boldsymbol{u}_0\left(\boldsymbol{X},t\right) + \boldsymbol{M}\left(\boldsymbol{X}\right)\boldsymbol{F}\boldsymbol{X} \\ \boldsymbol{0} \end{pmatrix}, \tag{54}$$

where as before $\boldsymbol{X} = \left\{\boldsymbol{X}^{(i)}\right\}$ and $\boldsymbol{\lambda} = \left\{\boldsymbol{\lambda}^{(i)}\right\}$. In a slight abuse of notation, we will write $\boldsymbol{K}\left(\boldsymbol{X}\right)$ to represent the block diagonal matrix of kinematic operators for each fiber i, $\boldsymbol{K}\left(\boldsymbol{X}\right) = \text{Diag}\left\{\boldsymbol{K}\left(\boldsymbol{X}^{(i)}\right)\right\}$, and likewise for $\boldsymbol{K}^*\left(\boldsymbol{X}\right)$.

Since we expect the fiber shapes to be smooth, we use a spectral spatial discretization, described in Section 4.1. In Section 4.2, we break the discretized mobility matrix $\boldsymbol{M}$ into three components: the local drag mobility $\boldsymbol{M}^{\mathrm{L}}$ given in (48), the finite part mobility (49), and the cross-fiber mobilities (50). The local drag matrix $\boldsymbol{M}^{\mathrm{L}}$ is the 3×3 matrix whose definition is the same as in continuum. The finite part mobility and cross-fiber mobilities require more specialized quadrature schemes since the integrals involved are near singular or singular and therefore too expensive or impossible to evaluate with direct quadrature. The basic idea of the specialized schemes is to factor out the (near) singularity, expand what remains in a monomial expansion, and compute the integrals involving monomials times the singularity analytically. In Section 4.2.1, we discuss this special quadrature scheme for the singular integrals appearing in the finite part mobility (49).

In Section 4.2.2, we write a quadratic complexity discretization which uses direct quadrature to compute the SBT interaction kernels (15). In Section 4.3, we then discuss how to make the complexity linear over a triply periodic, sheared domain using a spectral Ewald method. Since we reformulated the inter-fiber hydrodynamics in terms of the RPY tensor (see Section 2.2), our Ewald splitting method is exactly the positively split Ewald method of [21], with some modifications for a non-orthogonal coordinate system [22]. In Section 4.4 we return to the case when the direct quadrature is insufficiently accurate and corrections are required, for which we use a recently developed monomial-expansion-based special quadrature scheme [1] similar to that used for the finite part integral.

Finally, in Section 4.5, we present a semi-implicit second-order temporal discretization that avoids nonlinear solves and requires a minimum number of evaluations of the nonlocal hydrodynamics for each timestep. For dilute suspensions, our temporal integration strategy is essentially to treat the local drag part of the mobility $\boldsymbol{M}^{\mathrm{L}} \boldsymbol{F} \boldsymbol{X}$ implicitly using an implicit trapezoidal method. We treat all of the terms involving the finite part and cross-fiber mobilities explicitly. This leaves a linear system to be solved on each fiber separately. When the suspension becomes more concentrated, this scheme breaks down as the nonlocal hydrodynamics adds stiffness to the problem. When this occurs, we treat the nonlocal and local hydrodynamics implicitly using an implicit trapezoidal method and use GMRES to solve for $\boldsymbol{\alpha}$ and $\boldsymbol{\lambda}$. By converting to a residual form based on the solutions of the block diagonal system for dilute suspensions, we are able to use only the minimum number of GMRES iterations necessary to achieve stability without altering accuracy.

4.1 Spectral spatial discretization

Because the fibers are semi-flexible, their shapes are relatively smooth and can be well represented by a finite number of basis functions. This makes a spectral spatial discretization the logical choice.[2] We therefore use a first-kind Chebyshev grid for the collocation points on each fiber and Chebyshev polynomials for the basis functions $T_k(s)$, as described in Section 4.1.1. For indefinite integration, we use the pseudo-inverse of the Chebyshev differentiation matrix, and for definite integration we use Clenshaw-Curtis quadrature. Once these choices are made, the discretization of the kinematic operators $\mathcal{K}$ and $\mathcal{K}^*$ follows naturally in Section 4.1.2. The discretization of the elastic force operator $\mathcal{F}$ is more subtle as the boundary conditions must be treated correctly; for this we use the rectangular spectral collocation approach of [20, 5] that is described in Section 4.1.3. Throughout this section, we consider the discretization on a single fiber.

4.1.1 Collocation discretization

Because we use a collocation discretization, each fiber is discretized as a collection of nodes s_p, $p = 1, \ldots N$, where s_p is a node on a type 1 Chebyshev grid (i.e. a grid that does not include the endpoints). Our notation will shift slightly here to reflect the change from continuous to discrete. We use $\boldsymbol{X}$ to refer to the $N \times 3$ matrix of fiber positions at the collocation points. The pth row of this matrix will be denoted by $\boldsymbol{X}_p = \boldsymbol{X}(s_p)$. Likewise, $\boldsymbol{\tau}$ refers to the $N \times 3$ matrix of tangent vectors at the collocation points with $\boldsymbol{\tau}_p = \boldsymbol{\tau}(s_p)$, and $\boldsymbol{f}$ refers to a matrix of force densities evaluated at the nodes with rows $\boldsymbol{f}_p = \boldsymbol{f}(s_p)$. Meanwhile, $\boldsymbol{X}(s)$ refers to the *Chebyshev interpolant* for $\boldsymbol{X}$ (this is actually three interpolants, one for each direction), and likewise for $\boldsymbol{\tau}(s)$. We will not try to distinguish between the unknown "true" fiber shape (which could have more than N Chebyshev modes) and its Chebyshev approximation $\boldsymbol{X}(s)$.

The tools we use for differentiation and integration are standard [85]. For differentiation, we use the Chebyshev differentiation matrix $\boldsymbol{D}_N$. By $\boldsymbol{D}_N \boldsymbol{X}$, we mean the linear operation that takes $\boldsymbol{X}$, computes the $N - 1$ degree Chebyshev polynomial representation $\boldsymbol{X}(s)$, differentiates it, and returns $\boldsymbol{\tau}(s)$. We also define $\boldsymbol{D}_N^\dagger$, the pseudo-inverse of the Chebyshev differentiation matrix, which

---

[2]Implicit in our choice of spectral discretization is the assumption that the fiber constraint forces $\boldsymbol{\lambda}(s)$ and tangent vector rotation rates $\boldsymbol{\Omega}(s)$ are also smooth, which is the case for sufficiently large δ in the local drag regularization (14). If $\delta \ll 1$, $\boldsymbol{\lambda}$ and $\boldsymbol{\Omega}$ can become nonsmooth and oscillatory at the fiber endpoints, and a small number of Chebyshev modes can no longer resolve them.

gives the values of the indefinite integral of a function $f(s)$ modulo an unknown constant,

$$\left(\boldsymbol{D}_N^\dagger \boldsymbol{f}\right)_p \approx \int_0^{s_p} f(s')\, ds' + C. \tag{55}$$

For definite integration, we use Clenshaw-Curtis quadrature with weight w_p associated with each collocation point,

$$\int_0^L f(s')\, ds' \approx \boldsymbol{w}^T \boldsymbol{f} := \sum_{p=1}^N f_p w_p. \tag{56}$$

4.1.2 Discretization of $\mathcal{K}$ and $\mathcal{K}^*$

To construct a discretization of the kinematic operator $\mathcal{K}$ defined in (41), we first need to choose the basis functions T_k in the representation formula (40). We choose $T_k(s)$ to be the Chebyshev polynomial of the first kind of degree k on $[0, L]$. We truncate the sum at $N - 2$ basis functions,

$$g_j(s) = \sum_{k=0}^{N-2} \alpha_{jk} T_k(s). \tag{57}$$

The choice of $N - 2$ for the maximum summation index is a necessary condition to make the representation $\boldsymbol{U} = \mathcal{K}\boldsymbol{\alpha}$ unique on an N point Chebyshev grid. Increasing the number of basis functions introduces degeneracy without improving the fiber representation. In particular, if the maximum index in the sum were $N - 1$, integration of $g_j(s)$ in the inextensible velocity (41) could cause $\boldsymbol{U}$ to be zero at all N Chebyshev nodes without $\boldsymbol{\alpha}$ being zero.

Since the kinematic operators $\mathcal{K}$ and $\mathcal{K}^*$ act linearly on $\boldsymbol{\alpha}$ and $\boldsymbol{\lambda}$, respectively, they can each be discretized as matrices. Because we use a spectral discretization, however, care must be taken to avoid aliasing errors. The key step in doing this is to compute, for each $k = 0, \ldots, N - 2$ and $j = 1, 2$, the integrals that describe the inextensible motions of the fiber

$$\boldsymbol{J}_q^{(k,j)} \approx \int_0^{s_q} T_k(s') \boldsymbol{n}_j \left(\boldsymbol{\tau}\left(s'\right) \right) ds' \tag{58}$$

on a grid of size $2N$ ($q = 1, \ldots 2N$). To do so, we determine the Chebyshev polynomial representation of $\boldsymbol{\tau}(s)$ on the N point grid and upsample it to a type 1 Chebyshev grid of $2N$ points. We then compute the normal vectors on the $2N$ grid using the polar angle representations (28) and (29), and multiply the normal vectors pointwise by T_k evaluated on the $2N$ grid. This gives the integrand in (58) on the $2N$ grid. To integrate, we apply the matrix $\boldsymbol{D}_{2N}^\dagger$ to approximate the integrals (58) (modulo a constant) on the $2N$ grid. Since both T_k and $\boldsymbol{n}_j$ are Chebyshev polynomials of degree at most $N - 1$, these integrals are exact on the grid of size $2N$, modulo a constant.

21

Once $\boldsymbol{J}_p^{(k,j)}$ has been computed, the discretization of $\mathcal{K}^*$ is straightforward. The discrete form of the principle of virtual work (44) is given by inner products of $\boldsymbol{\lambda}$ with $\boldsymbol{J}^{(k,j)}$, where $\boldsymbol{\lambda}$ has been upsampled to a grid of size $2N$ to avoid aliasing errors,

$$\boldsymbol{K}^*\left(\boldsymbol{X}\right)\boldsymbol{\lambda} := \begin{pmatrix} \sum_{p=1}^{2N}\left(\boldsymbol{J}_p^{(0,1)}\cdot(\boldsymbol{U}\boldsymbol{\lambda})_p\right)w_p \\ \vdots \\ \sum_{p=1}^{2N}\left(\boldsymbol{J}_p^{(N-2,1)}\cdot(\boldsymbol{U}\boldsymbol{\lambda})_p\right)w_p \\ \hline \sum_{p=1}^{2N}\left(\boldsymbol{J}_p^{(0,2)}\cdot(\boldsymbol{U}\boldsymbol{\lambda})_p\right)w_p \\ \vdots \\ \sum_{p=1}^{2N}\left(\boldsymbol{J}_p^{(N-2,2)}\cdot(\boldsymbol{U}\boldsymbol{\lambda})_p\right)w_p \\ \hline \boldsymbol{w}^T\boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \hline 0 \\ 0 \\ \vdots \\ 0 \\ \hline \boldsymbol{0} \end{pmatrix}. \tag{59}$$

This defines $\boldsymbol{K}^*\left(\boldsymbol{X}\right)$ as a $(2N+1)\times 3N$ matrix acting on a $3N$ vector $\boldsymbol{\lambda}$, which has been upsampled to (i.e., evaluated on) the grid of size $2N$ by applying the upsampling matrix $\boldsymbol{U}$. Because $\boldsymbol{\lambda}$ discretely integrates to zero ($\boldsymbol{w}^T\boldsymbol{\lambda} = \boldsymbol{0}$), adding a constant to $\boldsymbol{J}_p^{(k,j)}$ does not change the first two rows of $\boldsymbol{K}^*\left(\boldsymbol{X}\right)\boldsymbol{\lambda}$. Thus the fact that $\boldsymbol{J}_p^{(k,j)}$ gives the integrals modulo a constant is not relevant in the formation of $\boldsymbol{K}^*$.

The discretization of $\boldsymbol{K}$ is less straightforward. The main issue is that the integrals $\boldsymbol{J}_p^{(k,j)}$ are exact on the grid of size $2N$, but the fiber velocity is defined on a grid of size N. One way around this is to simply double the grid size, i.e., define $\boldsymbol{K}\boldsymbol{\alpha}$ on a grid of size $2N$. This would, however, necessitate computing $\boldsymbol{M}\boldsymbol{f}$ on a grid of size $2N$ as well, which is unnecessarily expensive. Instead, we incur some aliasing error and downsample the velocity $\boldsymbol{J}_p^{(k,j)}$ to an N point grid. Specifically, we discretize the matrix $\boldsymbol{K}$ as

$$\left(\boldsymbol{K}\left(\boldsymbol{X}\right)\boldsymbol{\alpha}\right)_p = \overline{\boldsymbol{U}} + \sum_{j=1}^{2}\sum_{k=0}^{N-2}\alpha_{jk}\left(\boldsymbol{R}\boldsymbol{J}^{(k,j)}\right)_p \tag{60}$$

so that $\boldsymbol{K} = \boldsymbol{R}\boldsymbol{J}$ is a $3N \times (2N+1)$ matrix which acts on the $2N+1$ vector $\boldsymbol{\alpha} = \left(\alpha_{jk},\overline{\boldsymbol{U}}\right)$ to give the three components of the velocity at N points on the Chebyshev grid. The matrix $\boldsymbol{R}$ is a downsampling matrix which gives a reduced-order representation of $\boldsymbol{J}^{(k,j)}$ on the N point grid (the unknown constant in $\boldsymbol{J}^{(k,j)}$ can be folded into the constant velocity $\overline{\boldsymbol{U}}$). If we concisely write $\boldsymbol{K}^*\boldsymbol{\lambda}$ as defined in (59) as $\boldsymbol{K}^*\boldsymbol{\lambda} = \boldsymbol{J}^T\boldsymbol{W}_{2N}\boldsymbol{U}\boldsymbol{\lambda}$, where $\boldsymbol{W}_{2N}$ is the diagonal matrix of Clenshaw-Curtis quadrature weights on the $2N$ grid, and discretize the inner products in (43) as

$$\langle\boldsymbol{\alpha},\boldsymbol{K}^*\boldsymbol{\lambda}\rangle = \boldsymbol{\alpha}^T\boldsymbol{J}^T\boldsymbol{W}_{2N}\boldsymbol{U}\boldsymbol{\lambda} = \langle\boldsymbol{K}\boldsymbol{\alpha},\boldsymbol{\lambda}\rangle = (\boldsymbol{U}\boldsymbol{K}\boldsymbol{\alpha})^T\boldsymbol{W}_{2N}\boldsymbol{U}\boldsymbol{\lambda}, \tag{61}$$

22

we obtain the weighted least squares downsampling matrix

$$\boldsymbol{R} = \left(\boldsymbol{U}^T \boldsymbol{W}_{2N} \boldsymbol{U}\right)^{-1} \boldsymbol{U}^T \boldsymbol{W}_{2N}. \tag{62}$$

4.1.3 Discretization of $\mathcal{F}$

We use rectangular spectral collocation [5, 20] to discretize the bending force operator $\mathcal{F}$ with the boundary conditions (22). We recall that the matrix $\boldsymbol{X}$ gives the positions of the fiber on an N point type 1 Chebyshev grid that does *not* include the boundaries.

In rectangular spectral collocation, we compute an upsampled representation $\widetilde{\boldsymbol{X}}$ of $\boldsymbol{X}$. Since there are four boundary conditions, the upsampled representation is on a $\widetilde{N} = N + 4$ point type 2 Chebyshev grid that includes the endpoints. The unique configuration $\widetilde{\boldsymbol{X}}$ can be obtained by solving

$$\begin{pmatrix} \boldsymbol{A} \\ \boldsymbol{B} \end{pmatrix} \widetilde{\boldsymbol{X}} = \begin{pmatrix} \boldsymbol{X} \\ \boldsymbol{0} \end{pmatrix}, \tag{63}$$

where $\boldsymbol{A}$ is an $N \times \widetilde{N}$ resampling matrix and $\boldsymbol{B}$ is a $4 \times \widetilde{N}$ matrix that encodes the boundary conditions. The linear operation $\boldsymbol{A}\widetilde{\boldsymbol{X}}$ has the effect of computing the Chebyshev interpolant of $\widetilde{\boldsymbol{X}}$ on the $\widetilde{N} = N + 4$ point grid and evaluating it at the N original gridpoints. The first block equation simply states that the downsampled $\widetilde{\boldsymbol{X}}$ has to be the original $\boldsymbol{X}$. In the next block, the product $\boldsymbol{B}\widetilde{\boldsymbol{X}}$ is a vector with 4 entries. The first two entries are the Chebyshev interpolant approximation to $\partial_s^2 \widetilde{\boldsymbol{X}} (s = 0, L)$, respectively, and the second two entries are likewise an approximation to $\partial_s^3 \widetilde{\boldsymbol{X}} (s = 0, L)$. Thus the second block equation simply states that the boundary conditions are satisfied on the type 2 grid, and any modifications to the BCs would modify $\boldsymbol{B}$ in this formulation.

The use of a type 1 grid for $\boldsymbol{X}$ and a type 2 grid for $\widetilde{\boldsymbol{X}}$ is a sufficient condition for the left-hand side of system (63) to be invertible (see [20] for details). We can therefore write

$$\widetilde{\boldsymbol{X}} = \begin{pmatrix} \boldsymbol{A} \\ \boldsymbol{B} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{X} \\ \boldsymbol{0} \end{pmatrix} := \boldsymbol{E}\boldsymbol{X}. \tag{64}$$

In summary, $\widetilde{\boldsymbol{X}}$ is the unique upsampled configuration that satisfies the problem boundary conditions and gives $\boldsymbol{X}$ when downsampled. This is similar to "ghost cells" in finite difference schemes which take on unique values so that the boundary stencils satisfy the BCs to some order. The

rectangular spectral collocation method can therefore be thought of as a generalization of ghost cell techniques for finite difference methods to collocation-based spectral methods.

Once the configuration $\widetilde{\boldsymbol{X}}$ is known, the elastic force density can be computed on it as $\widetilde{\boldsymbol{f}}^{\kappa} = -\kappa \boldsymbol{D}_{\widetilde{N}}^{4} \widetilde{\boldsymbol{X}}$. The elastic force density $\widetilde{\boldsymbol{f}}^{\kappa}$ is then downsampled to the original N point type 1 grid to give the final result,

$$\boldsymbol{f}^{\kappa} = \boldsymbol{A}\widetilde{\boldsymbol{f}}^{\kappa} = -\kappa \left(\boldsymbol{A} \boldsymbol{D}_{\widetilde{N}}^{4} \boldsymbol{E} \right) \boldsymbol{X} := \boldsymbol{F} \boldsymbol{X}, \tag{65}$$

which defines $\boldsymbol{F}$, the discrete analogue of $\boldsymbol{\mathcal{F}}$. In a slight abuse of notation, we will use the notation $\boldsymbol{F}\boldsymbol{X}$ to refer to the bending force calculation on either a single fiber (where $\boldsymbol{F}$ is as defined in (65)), or a collection of fibers (where $\boldsymbol{F}$ is a diagonal block matrix composed of smaller matrices that are defined in (65)); the meaning should be clear from the context.

4.2 Discretization of $\boldsymbol{\mathcal{M}}$

We discretize the mobility operator defined in (16) by computing the relative velocity of point p on fiber i as

$$\left(\boldsymbol{M}\boldsymbol{f} \right)_{p}^{(i)} = \boldsymbol{M}^{\mathrm{L}} \left(\boldsymbol{\tau}_{p}^{(i)}; c_{p} \right) \boldsymbol{f}_{p}^{(i)} + \left(\boldsymbol{M}^{\mathrm{FP}} \left(\boldsymbol{X}^{(i)} \right) \boldsymbol{f}^{(i)} \right)_{p} + \sum_{j \neq i} \boldsymbol{M}^{\mathrm{c}} \left(\boldsymbol{X}_{p}^{(i)}, \boldsymbol{X}^{(j)} \right) \boldsymbol{f}^{(j)}. \tag{66}$$

The 3×3 matrix $\boldsymbol{M}^{\mathrm{L}}$ given in (9) is unchanged from the continuum, and $c_{p} = \bar{c}(s_{p}; \delta)$. The matrix $\boldsymbol{M}^{\mathrm{FP}}$ computes an approximation to the finite part integral (10) (see Section 4.2.1), and $\boldsymbol{M}^{\mathrm{c}} \left(\boldsymbol{X}_{p}^{(i)}, \boldsymbol{X}^{(j)} \right) \boldsymbol{f}^{(j)}$ is the velocity at point $\boldsymbol{X}_{p}^{(i)}$ induced by fiber j (see Section 4.2.2).

We will need notation to separate the local part of the discrete mobility (66), which is easy to invert, from the nonlocal part, which is not. For this we write

$$\left(\boldsymbol{M}\boldsymbol{f} \right)_{p}^{(i)} = \boldsymbol{M}^{\mathrm{L}} \left(\boldsymbol{\tau}_{p}^{(i)}; c_{p} \right) \boldsymbol{f}_{p}^{(i)} + \left(\boldsymbol{M}^{\mathrm{NL}} \left(\boldsymbol{X} \right) \boldsymbol{f} \right)_{p}^{(i)}, \tag{67}$$

where the nonlocal part of the mobility matrix $\boldsymbol{M}^{\mathrm{NL}} \left(\boldsymbol{X} \right)$ is a function of the collection of fibers $\boldsymbol{X}$ and acts on the collection of force densities $\boldsymbol{f}$. For the collection of fibers, we will simply write the splitting (67) as

$$\boldsymbol{M} \left(\boldsymbol{X} \right) = \boldsymbol{M}^{\mathrm{LD}} \left(\boldsymbol{X} \right) + \boldsymbol{M}^{\mathrm{NL}} \left(\boldsymbol{X} \right), \tag{68}$$

where the block diagonal matrix $\boldsymbol{M}^{\mathrm{LD}}$ is composed of a collection of 3×3 local drag matrices on the diagonal.

4.2.1 Discretization of $\mathcal{M}^{\mathrm{FP}}$

Here we discretize the finite part integral (10). Since the finite part integral involves only a single fiber, we use $\boldsymbol{X}$ here to denote a matrix of $N \times 3$ positions for a single fiber. Substituting the definition of the Stokeslet (1), we have

$$\left(\mathcal{M}^{\mathrm{FP}}\left[\boldsymbol{X}\right]\boldsymbol{f}\right)(s) = \frac{1}{8\pi\mu}\int_0^L \left(\frac{\boldsymbol{I} + \left(\hat{\boldsymbol{R}}\hat{\boldsymbol{R}}\right)\left(\boldsymbol{X}(s), \boldsymbol{X}(s')\right)}{\|\boldsymbol{R}\left(\boldsymbol{X}(s), \boldsymbol{X}(s')\right)\|}\boldsymbol{f}(s') - \left(\frac{\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|s - s'|}\right)\boldsymbol{f}(s)\right)ds' \quad (69)$$

for any coordinate on the fiber s. We seek to evaluate the finite part integral (69) at $s = s_p$ on a given fiber. Because each term in the integrand is singular, the integral (69) cannot directly be evaluated with Clenshaw-Curtis quadrature. One way around this difficulty is to simply skip the singular point in the quadrature, which results in a second-order accurate scheme. Because this destroys the spectral accuracy of our formulation, we seek an improved quadrature that handles the singularity analytically.

In [84], the integrand was regularized to make it non-singular, and a product integration scheme was used to compute the resulting regularized integral [84, Section 3.1]. The justification for the regularization is that the self mobility operator $\mathcal{M}$ is actually not invertible, since its null space contains force densities $\boldsymbol{f}$ with frequencies higher than $1/\epsilon$. Götz [26] and Tornberg and Shelley [84, Appendix B] show this analytically by considering a straight fiber and expanding $\boldsymbol{f}$ as a sum of Legendre polynomials (which diagonalize $\mathcal{M}^{\mathrm{FP}}$). They show that the centerline velocity $\boldsymbol{U}$ for a single fiber (8) uniquely gives the Stokeslet strength $\boldsymbol{f}(s)$ via $\boldsymbol{U} = \mathcal{M}\boldsymbol{f}$ if the maximum number of polynomials that contribute to $\boldsymbol{f}$ is less than $\mathcal{O}(1/\epsilon)$. Intuitively, adding polynomials of degree larger than $1/\epsilon$ introduces length scales into $\boldsymbol{f}$ which are less than ϵ and cannot be accounted for by SBT. In the discretization of [84], the number of points can exceed $1/\epsilon$ for $\epsilon = 10^{-2}$, and so regularization of the integrand is required.

Because we use a spectral basis with smooth fiber shapes, we never exceed the $\mathcal{O}(1/\epsilon)$ threshold for the number of Chebyshev polynomials. Indeed, having less than $\mathcal{O}(1/\epsilon)$ Chebyshev points (polynomials) is a sensible restriction on the numerics. After all, using such a large number of points (spectral modes) is antithetical to the philosophy of SBT, which, unlike IB methods, eliminates the need to resolve the length scale ϵ. Mori and Ohm analyze this "spectral truncation" approach for infinite (periodic) fibers and find that it yields errors of order at worst ϵ with respect to the true solution for a "slender body" PDE on the fiber surface [61, Eq. (20)]. This accuracy is asymptotically equivalent to that obtained using the regularization of Tornberg and Shelley [61,

Eq. (22)].

With this in mind, we do not modify the integrand of the finite part integral (69). Rather, we use a spectrally accurate method developed in [81] to compute the action of the finite part integral on the fiber force density. The key idea is to isolate the singularity by writing the integrand in (69) as $\boldsymbol{g}(s', s)(s' - s)/|s' - s|$ for some function $\boldsymbol{g}(s', s)$. In particular, we observe that the finite part integral (69) can be written as

$$\left(\boldsymbol{\mathcal{M}}^{\mathrm{FP}}\left[\boldsymbol{X}\right]\boldsymbol{f}\right)(s) = \int_0^L \boldsymbol{g}(s, s')\frac{s' - s}{|s' - s|}\, ds' = \frac{L}{2}\int_{-1}^1 \boldsymbol{g}(\eta, \eta')\frac{\eta' - \eta}{|\eta' - \eta|}\, d\eta', \tag{70}$$

where $\eta = -1 + 2s/L$ is a rescaled arclength coordinate on $[-1, 1]$ and

$$\boldsymbol{g}(s, s') = \frac{1}{8\pi\mu}\left[\left(\boldsymbol{I} + \left(\hat{\boldsymbol{R}}\hat{\boldsymbol{R}}\right)\left(\boldsymbol{X}(s), \boldsymbol{X}(s')\right)\right)\frac{|s' - s|}{\|\boldsymbol{R}\left(\boldsymbol{X}(s), \boldsymbol{X}(s')\right)\|}\boldsymbol{f}(s')\right.$$
$$\left. - \left(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right)\boldsymbol{f}(s)\right]\frac{1}{s' - s}. \tag{71}$$

The computation is now tractable since $\boldsymbol{g}$ has a limit as $s' \to s$. The limit is easily computed by adding and subtracting $(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s))\boldsymbol{f}(s')$ inside the square bracket, and Taylor expanding around $s' = s$, to obtain

$$\lim_{s' \to s}\boldsymbol{g}(s, s') = \frac{1}{8\pi\mu}\left(\frac{1}{2}\left(\boldsymbol{\tau}(s)\boldsymbol{X}_{ss}(s) + \boldsymbol{X}_{ss}(s)\boldsymbol{\tau}(s)\right)\boldsymbol{f}(s) + \left(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right)\boldsymbol{f}_s(s)\right). \tag{72}$$

Since $\boldsymbol{g}$ is smooth, we can approximate it by a polynomial expressed in a monomial basis on $[-1, 1]$,

$$\frac{L}{2}\boldsymbol{g}(\eta, \eta') \approx \sum_{k=0}^{N-1} \boldsymbol{c}_k(\eta)(\eta')^k, \tag{73}$$

where $\boldsymbol{c}_k$ is a vector of 3 coefficients for each η.

We are now ready to discretize $\boldsymbol{\mathcal{M}}^{\mathrm{FP}}\left[\boldsymbol{X}\right]$ with the matrix representation $\boldsymbol{M}^{\mathrm{FP}}\left(\boldsymbol{X}\right)$. Substituting the monomial expansion (73) into the finite part integrand (70) and computing the integrals involving monomials and the singularity analytically, we get

$$\left(\boldsymbol{M}^{\mathrm{FP}}\left(\boldsymbol{X}\right)\boldsymbol{f}\right)_p = \sum_{k=0}^{N-1}\boldsymbol{c}_k(\eta_p)\int_{-1}^1(\eta')^k\frac{\eta' - \eta_p}{|\eta' - \eta_p|}\, d\eta' = \sum_{k=0}^{N-1}\boldsymbol{c}_k(\eta_p)q_k(\eta_p), \tag{74}$$

$$\text{where}\qquad q_k(\eta_p) = \int_{-1}^1(\eta')^k\frac{\eta' - \eta_p}{|\eta' - \eta_p|}\, d\eta' = \frac{1 + (-1)^{k+1} - 2\eta_p^{k+1}}{k + 1}. \tag{75}$$

An adjoint method can be used to accelerate our computation of the product (74). Let us introduce the Vandermonde matrix $\boldsymbol{V}$ with entries $V_{pq} = \eta_p^q$. Let $\boldsymbol{g}^p\left(\boldsymbol{X}\right)$ be the $N \times 3$ matrix

with rows $\boldsymbol{g}_q^p = \boldsymbol{g}(\eta_p, \eta_q)$, with $\boldsymbol{g}$ defined in (71). The $N \times 3$ matrix $\boldsymbol{c}$ of coefficients of the three polynomial interpolants of the columns of $\boldsymbol{g}^p$ is $\boldsymbol{c} = \boldsymbol{V}^{-1}\boldsymbol{g}^p$. If $\boldsymbol{q}$ is an N vector with elements $q_k = q_k(\eta_p)$ as given in (75), then the product (74) can be computed efficiently as

$$\left(\boldsymbol{M}^{\mathrm{FP}}\left(\boldsymbol{X}\right)\boldsymbol{f}\right)_p = \boldsymbol{c}^T\boldsymbol{q} = \left(\boldsymbol{V}^{-1}\boldsymbol{g}^p\left(\boldsymbol{X}\right)\right)^T\boldsymbol{q} = \left(\boldsymbol{g}^p\left(\boldsymbol{X}\right)\right)^T\left(\boldsymbol{V}^{-T}\boldsymbol{q}\right) := \left(\boldsymbol{g}^p\left(\boldsymbol{X}\right)\right)^T\boldsymbol{b}. \tag{76}$$

Since $\boldsymbol{b} = \boldsymbol{V}^{-T}\boldsymbol{q}$ does not depend on the fiber configuration, it can be precomputed using pivoted LU factorization for each $p = 1, 2, \ldots N$ at the beginning of the simulation. The Vandermonde matrix must be sufficiently well-conditioned to do this calculation accurately; specifically, the fiber discretization can have at most $\sim$40 points in double precision. If higher accuracy is needed, then the fiber must be split into multiple panels or higher precision arithmetic must be used to compute $\boldsymbol{b}$ in (76).

4.2.2 Discretization of $\boldsymbol{\mathcal{M}}^{\mathrm{c}}$

In this section, we describe the simplest discretization of inter-fiber hydrodynamic interactions. We recall the definition of the velocity induced by fiber j at point s on fiber i from (15),

$$\boldsymbol{v}^{(j)}\left(\boldsymbol{X}^{(i)}(s)\right) = \left(\boldsymbol{\mathcal{M}}^{\mathrm{c}}\left[\boldsymbol{X}^{(j)}(\cdot)\right]\boldsymbol{f}^{(j)}(\cdot)\right)\left(\boldsymbol{X}^{(i)}(s)\right) \tag{77}$$

$$= \frac{1}{8\pi\mu}\int_0^L \boldsymbol{S}_D\left(\boldsymbol{X}^{(i)}(s), \boldsymbol{X}^{(j)}(s'), \frac{e^3}{24}(\epsilon L)^2\right)\boldsymbol{f}^{(j)}(s')\,ds'.$$

Given a discrete Chebyshev node on fiber i, $\boldsymbol{X}_p^{(i)}$, the total disturbance velocity is a sum of the flows generated by all other fibers $j \neq i$. We can therefore restrict our attention to the calculation of the velocity induced by a single "source" fiber at a single "target" point on another fiber.

The simplest approach is to discretize the interaction velocity (77) by Clenshaw-Curtis quadrature,

$$\boldsymbol{v}^{(j)}\left(\boldsymbol{X}_p^{(i)}\right) \approx \frac{1}{8\pi\mu}\sum_{q=1}^N w_q\boldsymbol{S}_D\left(\boldsymbol{X}_p^{(i)}, \boldsymbol{X}_q^{(j)}; \frac{e^3}{24}(\epsilon L)^2\right)\boldsymbol{f}_q^{(j)} := \boldsymbol{M}^{\mathrm{c}}\left(\boldsymbol{X}_p^{(i)}, \boldsymbol{X}^{(j)}\right)\boldsymbol{f}^{(j)}. \tag{78}$$

The key challenge in evaluating (78) is the quadratic complexity; for each Chebyshev point $\boldsymbol{X}_p^{(i)}$ we must sum over all others. We address this in Section 4.3 using the positively split Ewald method [21].

While the direct quadrature (78) represents the simplest way to discretize $\boldsymbol{\mathcal{M}}^{\mathrm{c}}$, it becomes inadequate when fibers i and j approach each other. In fact, the Stokeslet-doublet combination kernel $\boldsymbol{S}_D$ defined in (3) becomes singular if the Chebyshev interpolant $\boldsymbol{X}^{(j)}(s)$ of fiber j approaches

the target $\boldsymbol{X}_p^{(i)}$ for some value of s. This singularity occurs because the velocity of one fiber due to another (15) only makes sense physically if the two fiber cross sections are not overlapping (for overlapping fibers the RPY kernel is different, see the second line of (5)).

While it makes little physical sense for fiber cross sections to overlap, it is numerically possible. In this case, we set the velocity at the target point on fiber i to be equal to the centerline velocity at the closest point on fiber j. Let us denote the minimum distance from $\boldsymbol{X}_p^{(i)}$ to[3] $\boldsymbol{X}^{(j)}(s)$ by d and denote the closest point on fiber j to $\boldsymbol{X}^{(i)}$ as $\boldsymbol{X}^{(j)}(s^*)$. Then if the two cross sections are (almost) overlapping, we set

$$\boldsymbol{v}^{(j)}\left(\boldsymbol{X}_p^{(i)}\right) = \boldsymbol{M}^{\mathrm{L}}\left(\boldsymbol{\tau}^{(j)}\left(s^*\right); \bar{c}\left(s^*; \delta\right)\right)\boldsymbol{f}^{(j)}\left(s^*\right) + \left(\boldsymbol{M}^{\mathrm{FP}}\left(\boldsymbol{X}^{(j)}\right)\boldsymbol{f}^{(j)}\right)\left(s^*\right) \quad \text{for } d \le 2b, \quad (79)$$

so that the influence of fiber j on the target (which is inside the cross section of fiber j) is the same as if the target were actually on the centerline of fiber j, with the radius $b \approx 1.12\epsilon L$ given in (6). By $\boldsymbol{\tau}^{(j)}\left(s^*\right)$, we mean the Chebyshev polynomial $\boldsymbol{\tau}^{(j)}(s)$ evaluated at s^*, and likewise for the remaining terms.

For non-overlapping cross sections ($d > 2b$), the expression for the interaction velocity (15) has to be changed because evaluating (15) at $d = 2b$ may not be exactly equal to the centerline velocity on fiber j (79) because of the slenderness approximation used in SBT. We therefore set the velocity to be the interaction velocity (15) only if $d \ge 4b$. Between $2b \le d \le 4b$, we linearly interpolate between the interaction velocity (15) and centerline velocity (79). This interpolation procedure is almost identical to that of [84], except we use a different integral kernel $\boldsymbol{S}_D$ for the interaction velocity (15), and we estimate s^* using a more robust procedure described in Section 4.4.

Despite our modifications to the interaction velocity for contacting cross sections, the kernel in (15) is still nearly singular, and some quadrature scheme other than direct quadrature (78) is required to accurately determine the interaction velocity for $d > 2b$, as we discuss in Section 4.4.

4.3 Fast summation

Putting off for the moment the possible near-singular nature of the interaction integrals (15), suppose that we discretize every integral using direct quadrature (78). If F is the number of fibers, each of which is discretized using N Chebyshev points, then the direct evaluation of nonlocal hydrodynamics using (78) requires $\mathcal{O}\left((NF)^2\right)$ operations. In this section, we discuss our choice of

---

[3]Recall that we cannot and do not distinguish between the unknown "true" fiber shape and its Chebyshev interpolant and denote both with $\boldsymbol{X}^{(j)}(s)$.

fast algorithm to accelerate the evaluation of these many-body sums under triply periodic boundary conditions. We recall that the kernel $\boldsymbol{S}_D$ in (78) is the RPY kernel for non-overlapping spheres of radius $b = \epsilon L e^{3/2}/4$. Thus (78) reduces to summing the RPY kernel over all pairs of points $\left(\boldsymbol{X}_p^{(i)}, \boldsymbol{X}_q^{(j)}\right)$, which is a well-studied problem that can be treated with a number of fast algorithms [21, 45, 29, 93]. Alternatively, the kernel $\boldsymbol{S}_D$ can be viewed as a linear combination of Stokes singularities, and fast algorithms for the individual singularities can be applied [49, 2, 39]. In Section 4.3.2 we describe the Positively Split Ewald (PSE) approach of [21, 22], which assumes a constant value of b (and therefore ϵL), across all of the fibers; this assumption can be relaxed [23]. Because we are interested in rheological applications, we use the method of [22] to extend the fast Ewald summation technique of [21] to a parallelepiped sheared unit cell (see Section 4.3.1).

For overlapping spheres the RPY kernel $\boldsymbol{S}_{\mathrm{RPY}}$ differs from the SBT kernel $\boldsymbol{S}_D$ in (78), and is given by [87]

$$\boldsymbol{S}_{\mathrm{RPY}}(\boldsymbol{x}, \boldsymbol{y}; b) = \frac{1}{8\pi\mu}\left(\left(\frac{4}{3b} - \frac{3\|\boldsymbol{R}\|}{8b^2}\right)\boldsymbol{I} + \frac{\|\boldsymbol{R}\|}{8b^2}\hat{\boldsymbol{R}}\hat{\boldsymbol{R}}\right) \quad \text{if } \|\boldsymbol{R}\| < 2b, \tag{80}$$

where $\boldsymbol{R} = \boldsymbol{x} - \boldsymbol{y}$. This means that the PSE method mistakenly computes the RPY kernel (80) between a pair of points separated by a distance less than $2b$ instead of the desired SBT kernel. We need not worry about this, however, since points that are a distance less than $2b$ yield a target-fiber pair for which we set the velocity at the target to the fiber centerline velocity (79). More generally, even for points farther apart than $2b$ there will be some number of target-fiber pairs for which the direct quadrature (78) fails. Our approach to this is to rely on Ewald splitting to periodize and accelerate the many-body summation, then subtract the free space RPY kernel between the problematic fiber and target from the result. Subtracting the free space kernel leaves the periodic images of the sum, which have been correctly accounted for by Ewald splitting (since they are distant). We then handle the free space kernel $\boldsymbol{S}_D$ for problematic pairs of fibers and targets using the special quadrature algorithm described in Section 4.4.

4.3.1 Sheared coordinate system

In order to implement a shear flow in periodic boundary conditions, a strained coordinate system is necessary. The PSE method was extended to sheared cells in [22], but here we give a more detailed description for completeness. We assume (without loss of generality) that x is the flow direction, y is the gradient direction, and z is the vorticity direction. Let the total nondimensional strain be
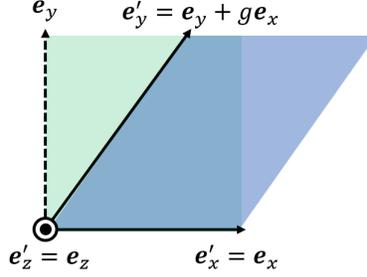
Figure 2: The sheared/parallelepiped periodic simulation cell. We denote the dimensionless shear by g. The green area shows the periodic cell (2D projection) when $g = 0$, and the blue area is the periodic cell for nonzero g.

$g(t)$. In Fig. 2 we define a strained coordinate system with axes

$$\boldsymbol{e}_{x'} = \boldsymbol{e}_x, \qquad \boldsymbol{e}_{y'} = \boldsymbol{e}_y + g(t)\boldsymbol{e}_x, \qquad \boldsymbol{e}_{z'} = \boldsymbol{e}_z, \tag{81}$$

and strained wave numbers

$$k'_x = k_x, \qquad k'_y = k_y + g(t)k_x, \qquad k'_z = k_z. \tag{82}$$

Here k_x, k_y, and k_z are the wave numbers when the periodicity is over the x, y, and z directions, while k'_x, k'_y, and k'_z are the wave numbers when the periodicity is over the x', y', and z' directions.

The transformation between the two coordinate systems is given by

$$\boldsymbol{x}' := \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 1 & -g(t) & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} := \boldsymbol{L}\boldsymbol{x}. \tag{83}$$

In the unsheared to sheared transformation (83), the sheared coordinates x', y', and z' are all periodic on $[0, L]^3$ (the blue simulation cell in Fig. 2).

Now we use the transformation (83) to transform the derivative operators in the unsheared coordinate system to the sheared one,

$$\frac{\partial}{\partial x} = \frac{\partial}{\partial x'} \qquad \frac{\partial}{\partial y} = \frac{\partial}{\partial y'} - g(t)\frac{\partial}{\partial x'} \qquad \frac{\partial}{\partial z} = \frac{\partial}{\partial z'}. \tag{84}$$

We therefore have the Laplacian in the transformed space as [40]

$$\Delta = \left(\frac{\partial^2}{\partial x'^2} + \left(\frac{\partial}{\partial y'} - g(t)\frac{\partial}{\partial x'} \right)^2 + \frac{\partial^2}{\partial z'^2} \right). \tag{85}$$

In Fourier space, $\widehat{\Delta} = \boldsymbol{k}' \cdot \boldsymbol{k}'$, where

30

$$\mathbf{k}' = (k_x', k_y' - g(t)k_x', k_z'), \tag{86}$$

Using the sheared to unsheared transformation in (82), it is easy to see that $k' := \|\mathbf{k}'\| = \|(k_x, k_y, k_z)\| := k$. It follows that we can simply replace k in any isotropic Fourier calculations by k' to use a Fourier method in the sheared coordinate system [91, 40]. In Appendix B, we verify this formulation by considering a set of particles that can be viewed periodically in two ways. This appendix verifies our correct treatment of the sheared periodic boundary conditions.

4.3.2 Ewald splitting for direct quadratures

Let $\mathbf{x}$ be a target point on the centerline of a fiber. To evaluate the direct quadrature (78) with periodic boundary conditions, we first compute

$$\mathbf{U}^{(\mathrm{PSE})}(\mathbf{x}) := \sum_P \sum_i \mathbf{S}_{\mathrm{RPY}}\left(\mathbf{x}, \mathbf{y}_i^{(P)}; b\right) \mathbf{F}_i := \sum_i \mathbf{S}_{\mathrm{RPY}}^{(P)}(\mathbf{x}, \mathbf{y}_i; b)\, \mathbf{F}_i \tag{87}$$

where $\mathbf{F}_i$ is a force (force density × weight) assigned to the point $\mathbf{y}_i$. For each target point $\mathbf{x} = \mathbf{y}_j$ for some j, the sum is over all discrete fiber points $\mathbf{y}_i$ (including $\mathbf{y}_j$) and over triply periodic images of the points $\mathbf{y}_i$ in the sheared coordinate system P. The periodized RPY kernel is denoted by $\mathbf{S}_{\mathrm{RPY}}^{(P)}$.

We use the Ewald splitting of [21, 22] to accelerate the computation of the many-body sum (87) on a periodic domain. The idea of Ewald splitting or Ewald summation is to split the RPY kernel into a smooth long-ranged part and a remaining short-ranged part. The smooth "far field" part has an exponential decay in Fourier space and can be done by standard Fourier methods (namely the non-uniform FFT), and the "near field" part decays exponentially in real space and can be truncated so that it is nonzero for $\mathcal{O}(1)$ neighboring points (sources) per target.

Let $\mathbf{x}'$ and $\mathbf{y}'$ be the coordinates of the points $\mathbf{x}$ and $\mathbf{y}$ in the sheared domain using the coordinate transformation (83). The periodic RPY tensor for a sphere with radius b can then be written on the sheared domain as

$$\mathbf{S}_{\mathrm{RPY}}^{(P)}(\mathbf{x}', \mathbf{y}'; b) = \frac{1}{V\mu} \sum_{\mathbf{k}' \neq \mathbf{0}} e^{i\mathbf{k}' \cdot (\mathbf{x}' - \mathbf{y}')} \frac{1}{k'^2} \left(\mathbf{I} - \hat{\mathbf{k}}' \hat{\mathbf{k}}'^T\right) \operatorname{sinc}^2\left(k'b\right), \tag{88}$$

where V is the domain volume and $\mathbf{k}' = 2\pi \mathbf{m}/L_d$, where $\mathbf{m}$ is a vector of three integers and L_d is the periodic domain length. Using the screening function of Hasimoto [32],

$$H(k', \xi) = \left(1 + \frac{k'^2}{4\xi^2}\right) e^{-k'^2/4\xi^2}, \tag{89}$$

31

we split the periodic kernel $\boldsymbol{S}_{\text{RPY}}^{(P)}$ into a far field and near field component, $\boldsymbol{S}_{\text{RPY}}^{(P)} = \boldsymbol{S}_{\text{RPY}}^{(\text{FF})} + \boldsymbol{S}_{\text{RPY}}^{(\text{NF})}$, where the far field is given in Fourier space by

$$\boldsymbol{S}_{\text{RPY}}^{(\text{FF})}\left(\boldsymbol{x}', \boldsymbol{y}'; b\right) = \frac{1}{V\mu} \sum_{\boldsymbol{k}' \neq \boldsymbol{0}} e^{i\boldsymbol{k}' \cdot (\boldsymbol{x}' - \boldsymbol{y}')} \frac{1}{k'^2} \left(\boldsymbol{I} - \hat{\boldsymbol{k}}' \hat{\boldsymbol{k}}'^T\right) \text{sinc}^2\left(k'b\right) H(k', \xi). \tag{90}$$

Here ξ is a splitting parameter that controls the decay of the far field kernel in Fourier space and of the near field kernel in real space, and is chosen to optimize performance. The total far field sum is obtained by summing the far field kernel (90) over all points $\boldsymbol{y}'$, with the $k' = 0$ mode set to zero since in continuum the total force on the system is zero. We use standard NUFFT algorithms (in particular the Flatiron NUFFT library [6]) to compute these sums at all points $\boldsymbol{x}'$ in log-linear time in the number of points (see [21, 22] for more details). The Flatiron NUFFT relies on a new "exponential of a semicircle" kernel to do spreading and interpolation [6], which is more efficient than the traditional Gaussian featured in [21, 22]. See [82, Eq. (31)] for error estimates using the exponential of a semicircle kernel, although these have yet to be extended to sheared domains in the manner of [22, Eq. (55)].

Assuming that the near field decays rapidly enough that Fourier series can be replaced by Fourier integrals, the near field mobility can be computed in real space by inverse transforming its Fourier space representation,

$$\boldsymbol{S}_{\text{RPY}}^{(\text{NF})}\left(\boldsymbol{x}, \boldsymbol{y}; b\right) = F(r, \xi, b)\left(\boldsymbol{I} - \hat{\boldsymbol{r}}\hat{\boldsymbol{r}}^T\right) + G(r, \xi, b)\hat{\boldsymbol{r}}\hat{\boldsymbol{r}}^T, \tag{91}$$

where $\boldsymbol{r} = (\boldsymbol{x} - \boldsymbol{y})^*$, $r = \|\boldsymbol{r}\|$, and the $*$ denotes the nearest periodic image in the sheared domain (blue in Fig. 2). The exact forms of F and G are given in [21, Appendix A]. The total near field sum is computed at $\boldsymbol{x}$ by summing the near field kernel (91) over neighboring points whose minimum image distance from $\boldsymbol{x}$ is less than a precomputed value r^*. We choose r^* so that the velocities are computed to a relative tolerance of 10^{-3}, and set ξ such that r^* is small enough that only the nearest periodic image contributes to the near field sum for each pair of points.

While the nearest image for near field calculations is over the sheared domain, $\boldsymbol{r}$ and r are computed using the Euclidean metric. We search for pairs of points closer than r^* apart in log-linear time using a kD tree implemented in SciPy for a rectangular periodic cell. To adjust for the fact that the $\boldsymbol{x}'$ coordinates are given on a non-orthogonal coordinate system, we bound the Euclidean distance between points in the unsheared coordinates by their "distance" in sheared coordinates,

$$\|\boldsymbol{r}\| = \sqrt{\boldsymbol{r}^T \boldsymbol{r}} = \sqrt{(\boldsymbol{r}')^T \boldsymbol{L}^{-T} \boldsymbol{L}^{-1} \boldsymbol{r}'} \leq \left(1 + \frac{1}{2}\left(g^2 + \sqrt{g^2(g^2 + 4)}\right)\right) \|\boldsymbol{r}'\| := \psi \|\boldsymbol{r}'\|, \tag{92}$$

where we have used the maximum eigenvalue of $\boldsymbol{L}^{-T}\boldsymbol{L}^{-1}$ to bound the norm [22]. The factor ψ defined in (92) can be thought of as a "safety factor" in the sense that points that are r^* apart using the Euclidean metric in primed coordinates are at most ψr^* apart in physical space.

For multiple interacting fibers, the velocity $\boldsymbol{U}^{(\mathrm{RPY})}\left(\boldsymbol{X}_p^{(i)}\right)$ is obtained by summing both the far field and near field over fibers j and points q. One inconvenience is that the PSE sum (87) is over all pairs of points, including a point with itself ($\boldsymbol{y}_i = \boldsymbol{x}$ is automatically included in the sum). Specifically, the PSE sum (87) will include interactions of a fiber with itself using the RPY kernel; this is not correct since those interactions should be computed by the SBT formula (8). Since we know *a priori* that the velocity at target $\boldsymbol{X}_p^{(i)}$ incorrectly includes the direct quadrature sum due to fiber i, we subtract the free space RPY kernel, defined in (5), from the RPY sum $\boldsymbol{U}^{(\mathrm{PSE})}\left(\boldsymbol{X}_p^{(i)}\right)$ defined in (87) to obtain the final Ewald sum

$$\boldsymbol{U}^{(\mathrm{RPY})}\left(\boldsymbol{X}_p^{(i)}\right) = \boldsymbol{U}^{(\mathrm{PSE})}\left(\boldsymbol{X}_p^{(i)}\right) - \sum_q \boldsymbol{S}_{\mathrm{RPY}}\left(\boldsymbol{X}_p^{(i)}, \boldsymbol{X}_q^{(i)}; b\right) \boldsymbol{f}_q^{(i)} w_q. \tag{93}$$

4.4 Near fiber quadrature

The Ewald splitting scheme (93) gives the velocity at all points $\boldsymbol{X}_p^{(i)}$ due to all other points $\boldsymbol{X}_q^{(j)}$, including target-fiber pairs where the direct quadrature scheme (78) is inaccurate. If the box size is sufficiently large, these inaccuracies only happen for the periodic image of the target $\boldsymbol{X}_p^{(i)}$ that is closest to the fiber $\boldsymbol{X}^{(j)}$. The other periodic images are handled correctly using the Ewald splitting scheme (93), since they are sufficiently far from the fiber. In this section, we describe the special quadrature scheme we use to correct the velocity for fibers $j \neq i$ that are close to target $\boldsymbol{X}_p^{(i)}$.

We need to compute the interaction velocity

$$\boldsymbol{v}\left(\boldsymbol{x}\right) = \frac{1}{8\pi\mu} \int_0^L \boldsymbol{S}_D\left(\boldsymbol{x}, \boldsymbol{X}(s), \frac{e^3}{24}(\epsilon L)^2\right) \boldsymbol{f}(s)\, ds, \tag{94}$$

where $\boldsymbol{x}$ is a target point and $\boldsymbol{X}(s)$ is the Chebyshev interpolant of the centerline of a fiber. There are several components in our scheme to compute the interaction velocities $\boldsymbol{v}\left(\boldsymbol{x}\right)$ to a guaranteed tolerance regardless of how close the target point $\boldsymbol{x}$ is to the centerline of fiber $\boldsymbol{X}(s)$. We need to:

1. Understand how far $\boldsymbol{x}$ can be from $\boldsymbol{X}(s)$ for the direct quadrature (78) to remain sufficiently accurate.

2. Obtain a reliable metric to compute or bound the minimum distance between the target $\boldsymbol{x}$ and fiber $\boldsymbol{X}(s)$,

$$d := \min_s \|\boldsymbol{x} - \boldsymbol{X}(s)\|. \tag{95}$$

Our procedure to compute d will be different for $d/L = \mathcal{O}(1)$, when fibers are far apart and minimizing over a discrete set of nodes is sufficiently accurate, than for $d/L = \mathcal{O}(\epsilon)$, where it is more efficient to actually solve the continuous minimization problem.

3. Use a special quadrature scheme to compute the integral for $d = \mathcal{O}(\epsilon)$. The special quadrature scheme will of course be more expensive than direct quadrature (78), but less expensive than actually using the requisite number of direct quadrature points required to get the same accuracy. The scheme we use here is taken directly from [1] and is based on extending the ideas used for the finite part integration in Section 4.2.1 to near-singular quadrature.

4. If $d < 4b$, compute the closest point on fiber $\boldsymbol{X}(s)$ to the target $\boldsymbol{x}$ and denote it by $\boldsymbol{X}\left(s^*\right)$. Use the distance metric for small d to correctly blend the interaction velocity (94) with the centerline velocity (79) of the fiber at $\boldsymbol{X}\left(s^*\right)$.

To begin, we define an acceptable tolerance for the integrals. Since slender body theory itself is only accurate to $\mathcal{O}(\epsilon)$, it does not make sense to set a tolerance less than ϵ. For actin filaments $\epsilon \approx 10^{-3}$, so we define the tolerance as 10^{-3} and set $\epsilon = 10^{-3}$ in our accuracy tests. That is, our goal is to compute the interaction velocities $\boldsymbol{v}\left(\boldsymbol{x}\right)$ to three digits of accuracy regardless of the distance between a target $\boldsymbol{x}$ and fiber $\boldsymbol{X}(s)$. Algorithm 1 gives the schematic flowchart of our method, the details of which are discussed next. While this method cannot guarantee 3 digits of accuracy, it does so for most target-fiber pairs of interest to us; see Appendix C for numerical results.

4.4.1 Distance where direct quadrature breaks down

Our first goal is to determine when direct quadrature breaks down. To do this, we simply measured the accuracy of direct quadrature (78) with $N = 16$ and $N = 32$ Chebyshev nodes for randomly-generated pairs of fibers and targets. Specifically (more detail is given in Appendix C), we generated 100 inextensible fibers with 16 nonzero Chebyshev coefficients decaying exponentially (in expectation) by four orders of magnitude. We placed 100 targets around each fiber a distance d in the normal direction and computed the quadratures (78). Measuring error with respect to a refined direct quadrature, we found that $N = 16$ gives 3 digits of accuracy for all test cases when the non-dimensional distance $d/L \geq 0.15$. Likewise, direct quadrature with $N = 32$ points gives 3 digits of accuracy when $d/L \geq 0.06$.
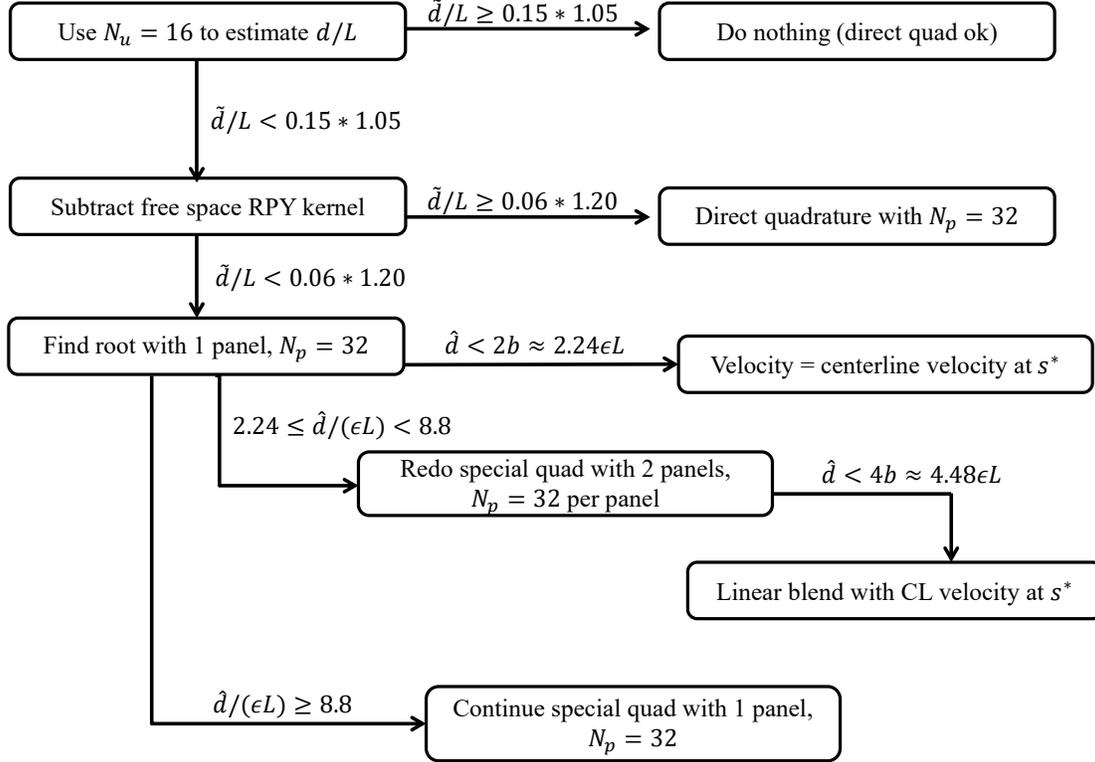
Algorithm 1: The discrete operation $v(x) = M^c(x, X) f$ for a fiber $X(s)$ and target x. Beginning with a target x and N Chebyshev points on the fiber centerline $X(s)$, we first estimate the distance d/L by $\tilde{d}/L$, where $\tilde{d}$ is computed from the discrete minimization over 16 uniform points. This determines whether the result from direct quadrature with N (assumed to be at least 16) points is sufficiently accurate. If direct quadrature is not sufficiently accurate, we subtract the free space RPY kernel (96) from the velocity (93) and, if $N < 32$, redo the calculation with direct quadrature with $N_p = 32$ points or special quadrature. If special quadrature is needed, we calculate the complex root η^* of $\|x - X(\eta)\| = 0$, use the projection (99) to find the closest point on the fiber, and compute the non-dimensional distance $\hat{d}/(\epsilon L)$ in (100). The distance $\hat{d}/(\epsilon L)$ determines whether to use special quadrature with 1 or 2 panels of 32 points each, and whether to use a linear combination with the fiber centerline velocity at s^*, given in (79). The algorithm is designed to ensure the calculation of $v(x)$ to 3 digits of accuracy most of the time, and is specific to $\epsilon \approx 10^{-3}$.

4.4.2 Estimating d for $d/L = \mathcal{O}(1)$

Since direct quadrature breaks down for $d/L < 0.15$ when $N = 16$, we need to determine whether a target x is indeed a distance less than $d/L = 0.15$ from the fiber centerline $X(s)$ (the analogous

statement holds for $d/L = 0.06$, $N = 32$). To do this quickly, we resample the fiber centerline at $N_u = 16$ uniformly spaced points (in the arclength coordinate s) and perform a discrete minimization over the uniform fiber points to estimate the distance d. We denote this approximation by $\tilde{d}$. Using the same random set of fibers and targets as in Section 4.4.1, we found the estimate of d by $\tilde{d}$ has a relative error in the distance $|\tilde{d} - d|/d$ of at most 5% for $d/L = 0.15$ and 20% for $d/L = 0.06$.

Since direct quadrature breaks down at $d/L = 0.15$ for $N = 16$, after accounting for the error in estimating d, we have that if $\tilde{d}/L \geq 0.15 \times 1.05$, direct quadrature with $N = 16$ gives the integral to 3 digits. If $\tilde{d}/L < 0.15 \times 1.05$, it is possible that the direct quadrature (78) is not accurate to 3 digits, and so the many-body Ewald sum (93) contains a direct quadrature between $\boldsymbol{x}$ and $\boldsymbol{X}$ that is not accurate enough. We therefore subtract, for the target point $\boldsymbol{x}$, this incorrect part of the sum, specifically the free space RPY kernel between the fiber $\boldsymbol{X}(s)$ and target $\boldsymbol{x}$,

$$\boldsymbol{v}^{(\mathrm{RPY})}(\boldsymbol{x}) = \sum_{p=1}^{N} \boldsymbol{S}_{\mathrm{RPY}}\left(\boldsymbol{x}, \boldsymbol{X}; b\right) w_p. \tag{96}$$

We then recompute the integral (94) to 3 digits using some other method.

For $N = 16$, our first resort to compute the integral (94) is to sample the Chebyshev polynomial $\boldsymbol{X}(s)$ at 32 points and use direct quadrature (78) with 32 points. Since direct quadrature with 32 points gives 3 digits of accuracy when $d/L \geq 0.06$, including the error bounds we have that if $\tilde{d}/L \geq 0.06 \times 1.20$, the direct quadrature gives the integral (94) to 3 digits. For $\tilde{d}/L < 0.06 \times 1.20$, we abandon direct quadrature and use a special quadrature routine.

This initial step to upsample and integrate directly is designed to take care of most of the near target and fiber pairs without incurring a significant computational cost. Our empirical correlation between the distance $\tilde{d}$ and the direct quadrature error is less rigorous than the direct quadrature error estimates of [1]. These estimates require information about the near singularity which, as discussed in the next section, must be computed using more expensive root finding.

4.4.3 Special quadrature.

The special quadrature routine is taken directly from [1]. As in Section 4.2.1, the underlying idea is to find the near singularity in the integrand, factor it out, expand what remains in a monomial basis, and compute integrals with the singularity and monomials analytically. This time, however, the integrand is not actually singular on the fiber centerline. By expanding the fiber representation to the complex plane, the nearby singularity can be found in the complex plane and the entire procedure of Section 4.2.1 can be repeated.

In more detail, the interaction velocity integral (15) can be written so that the numerator is smooth as $\boldsymbol{x}$ approaches the centerline of $\boldsymbol{X}(s)$. Starting from

$$\boldsymbol{v}\left(\boldsymbol{x}\right) = \frac{1}{8\pi\mu} \int_0^L \left(\frac{\boldsymbol{f}(s)}{\|\boldsymbol{R}\|} + \frac{\left((\boldsymbol{R}\boldsymbol{R}) + \frac{e^3}{24}(\epsilon L)^2\,\boldsymbol{I}\right)\boldsymbol{f}(s)}{\|\boldsymbol{R}\|^3} - 3\frac{e^3}{24}(\epsilon L)^2\frac{(\boldsymbol{R}\boldsymbol{R})\,\boldsymbol{f}(s)}{\|\boldsymbol{R}\|^5} \right)\,ds, \qquad (97)$$

where $\boldsymbol{R} = \boldsymbol{x} - \boldsymbol{X}(s)$, we rescale s by $\eta = -1 + 2s/L$. Then each of the terms in the integral (97) can be written in the form

$$\int_{-1}^1 \frac{\boldsymbol{h}_m\left(\boldsymbol{x},\eta\right)}{\|\boldsymbol{x} - \boldsymbol{X}(\eta)\|^m}\,d\eta, \qquad (98)$$

where $m = 1, 3, 5$. For each m, $\boldsymbol{h}_m\left(\boldsymbol{x},\eta\right)$ is a density that depends on the target point $\boldsymbol{x}$ and varies smoothly along the fiber j arclength coordinate η.

Now, the idea of [1] is to extend the representation of $\boldsymbol{X}(\eta)$ from $\eta \in [-1, 1] \subset \mathbb{R}$ to the *complex* plane $\mathbb{C}$ and compute the complex root of $\|\boldsymbol{x} - \boldsymbol{X}(\eta)\| = 0$. Because the centerline representation $\boldsymbol{X}(\eta)$ is available as a Chebyshev series, it is simple to solve for the root via Newton iteration. We denote this root by η^*, i.e., $\|\boldsymbol{x} - \boldsymbol{X}(\eta^*)\| = 0$ with $\eta^* \in \mathbb{C}$.

Once the root is known, the algorithm of [1] finds the radius of the Bernstein ellipse associated with η^*. This radius then bounds the direct quadrature error for the integral (97). If the upper bound on the direct quadrature error with $N = 32$ points is less than 10^{-3}, we proceed with direct quadrature. Otherwise, we use the special quadrature scheme of [1], which is the same as in the finite part integration in Section 4.2.1. The complex singularity is factored out of the integrand, leaving a "smooth" function which can be expanded in a monomial basis. Integrals involving monomials multipled by the singularity are computed analytically, and an inner product of the monomial coefficients with the analytical integrands yields the approximation to $\boldsymbol{v}\left(\boldsymbol{x}\right)$.

The only difference from Section 4.2.1 is that the location of the singularity (complex root η^*) now depends on the fiber position $\boldsymbol{X}$. This means that the roots and monomial coefficients must be computed at every time step using an LU factorization of the Vandermonde matrix. Since the Vandermone matrix is a function of the nodes s_p on the fiber, its LU factorization is the same for all fibers and can be precomputed once at the start of a dynamic simulation. We refer the reader to [1, Section 3] for more details on this quadrature scheme.

While error bounds exist for direct quadrature, the special quadrature scheme of [1] does not provide error bounds or a method for selecting the fiber discretization (number of panels, points per panel, etc.). Because of this, we performed an empirical study on the same set of 100 fibers and targets as in Section 4.4.1. The randomized testing described in Appendix C showed that, for

most fiber configurations of interest to us, 1 panel of 32 points is sufficient to compute the integral to 3 digits using special quadrature as long as the non-dimensional distance between the target and fiber is $d/(\epsilon L) > 8$. Otherwise, 2 panels of 32 points are required. Fibers with high curvature typically give the largest errors for a given discretization.

4.4.4 Estimating d for $d/L = \mathcal{O}(\epsilon)$

We still require a robust numerical procedure to determine when the the target point is too close to the cross section of the fiber (i.e., when $d < 4b = e^{3/2}\epsilon L$). Our idea is to use the real part of the root η^* as the closest arclength coordinate on the fiber to the target. We know the root η^* solves $\|\boldsymbol{x} - \boldsymbol{X}(\eta^*)\| = 0$. It seems sensible, therefore, for the real part of the root to approximately minimize (over real η) $\|\boldsymbol{x} - \boldsymbol{X}(\eta)\|$ when the root is close to the real line. We therefore define s^*, the closest point on the fiber to the target, from the complex root η^* by removing the imaginary part of the root and rescaling,

$$
s^* = \begin{cases} \frac{L}{2}(\mathrm{Re}(\eta^*) + 1) & -1 \leq \mathrm{Re}(\eta^*) \leq 1 \\ 0 & \mathrm{Re}(\eta^*) < -1 \\ L & \mathrm{Re}(\eta^*) > 1 \end{cases} . \tag{99}
$$

The shortest distance from $\boldsymbol{x}$ to the fiber can then be estimated as

$$
\hat{d} := \|\boldsymbol{x} - \boldsymbol{X}(s^*)\|, \tag{100}
$$

where the position $\boldsymbol{X}(s^*)$ is computed by evaluating the Chebyshev interpolant at s^*. Our randomized tests showed that this estimate gives an error of at most 10% for $d/(\epsilon L) \leq 8$, which is sufficiently accurate for our purpose. Because we use a point s^* on the fiber centerline to estimate $\hat{d}$, this 10% error is always an overestimation. For this reason we use 2 panels for special quadrature when $\hat{d}/(\epsilon L) \leq 8.8$.

Combining all of our steps, we obtain an algorithm to compute $\boldsymbol{v}(\boldsymbol{x})$ for all targets and fibers that is presented as a flowchart in Algorithm 1. In Appendix C we show that this quadrature scheme gives 3 digits of accuracy in the integrals (94) with high probability.

4.5 Temporal discretization

In this section, we discuss how we discretize the evolution equation (45) in time. We use the notation of (68) to split the mobility $\boldsymbol{M}$ into an $\mathcal{O}(\log \epsilon)$ local part $\boldsymbol{M}^{\mathrm{LD}}$ and $\mathcal{O}(1)$ non-local part

38

M^{NL}. Our goal in this section is to develop a second-order temporal integrator with the properties that:

1. A minimum number of evaluations of the nonlocal hydrodynamics are needed per time step.

2. Bending elasticity is treated implicitly.

3. Any linear solves are block-diagonal, or fiber by fiber, so that the complexity of solving evolution equation (45) scales linearly with the number of fibers and can be trivially parallelized.

Since we have separated the mobility matrix into the dominant $\mathcal{O}(\log \epsilon)$ block diagonal local drag matrix M^{LD} and the $\mathcal{O}(1)$ term that remains, we will begin by treating terms associated with M^{LD} implicitly, thereby alleviating some of the stability restrictions associated with the bending force FX. This kind of approach gives almost unconditional stability as long as the density of fibers is small enough for the local drag term to dominate the fiber's motion. When the density of fibers is larger, we will treat the bending force implicitly in the M^{NL} term as well; the resulting linear system can be solved approximately by a few iterations of GMRES [74].

To avoid nonlinear solves and still achieve second-order accuracy, we extrapolate values from previous time steps to the midpoint of the next time step and use these extrapolated values as the arguments for nonlinear functions (e.g. $M(X)$ and $K(X)$). More precisely, we define the extrapolated midpoint fiber positions as

$$X_{n+1/2,*} = \frac{3}{2}X_n - \frac{1}{2}X_{n-1}, \tag{101}$$

where we have used the notation X_n to denote the fiber positions X at the nth time step. As is our usual convention, we have not used fiber indices in the extrapolation (101) since it applies to every fiber independent of the others.

4.5.1 Semi-implicit method for dilute suspensions

To discretize the evolution equation (45) and principle of virtual work (46) in time, we split the mobility matrix into $M = M^{\mathrm{LD}} + M^{\mathrm{NL}}$. Since the elastic force density FX involves fourth derivatives, it must be treated implicitly to maintain stability as the number of Chebyshev grid points N increases. For dilute suspensions, we assume that, since M^{LD} is the dominant $\mathcal{O}(\log \epsilon)$ contribution at each point, only treating the term $M^{\mathrm{LD}}FX$ implicitly will still give improved stability.

A second-order, semi-implicit discretization of the evolution equation (45) begins by solving the saddle-point system

$$M_{n+1/2,*}^{\text{LD}} \left(\lambda_{n+1/2} + \frac{1}{2} F \left(X_n + X_{n+1,*} \right) \right) \tag{102}$$

$$+ M_{n+1/2,*}^{\text{NL}} \left(\lambda_{n+1/2,*} + F X_{n+1/2,*} \right) + u_0 \left(X_{n+1/2,*} \right) = K_{n+1/2,*} \alpha_{n+1/2},$$

$$K_{n+1/2,*}^* \lambda_{n+1/2} = 0$$

for $\lambda_{n+1/2}$ and $\alpha_{n+1/2}$, where the notation $M_{n+1/2,*}^{\text{LD}}$ means $M^{\text{LD}} \left(X_{n+1/2,*} \right)$ (and likewise for M^{NL}, K, and K^*). To obtain a second-order block-diagonal system, we extrapolate previous λ values to the midpoint of the next time step,

$$\lambda_{n+1/2,*} = 2\lambda_{n-1/2} - \lambda_{n-3/2}. \tag{103}$$

When the nonlocal mobiility in the block diagonal discretization (102) is applied to $\lambda_{n+1/2,*}$ and $F X_{n+1/2,*}$, the total force might be nonzero because of discretization errors in F. In this case, we manually set the total force to zero in the PSE method of Section 4.3. We also introduce the approximation

$$X_{n+1,*} = X_n + \Delta t K_{n+1/2,*} \alpha_{n+1/2} \tag{104}$$

to make (102) a linear system in $\alpha_{n+1/2}$ and $\lambda_{n+1/2}$. Since X_{n+1} will actually be computed by rotating the tangent vectors and integrating the result (see Section 4.5.3), the update (104) is a second-order approximation to the actual X_{n+1}. Nevertheless, it is a sufficient approximation to give the same stability properties as if the actual X_{n+1} were included in a nonlinear saddle-point system.

By substituting the approximation $X_{n+1,*}$ in (104) into saddle-point system (102), we obtain the following saddle-point linear system to solve at every time step,

$$\begin{pmatrix} -M^{\text{LD}} & K - \frac{\Delta t}{2} M^{\text{LD}} F K \\ K^* & 0 \end{pmatrix}_{n+1/2,*} \begin{pmatrix} \lambda_{n+1/2} \\ \alpha_{n+1/2} \end{pmatrix} = \tag{105}$$

$$\begin{pmatrix} M_{n+1/2,*}^{\text{LD}} F X_n + M_{n+1/2,*}^{\text{NL}} \left(\lambda_{n+1/2,*} + F X_{n+1/2,*} \right) + u_0 \left(X_{n+1/2,*} \right) \\ 0 \end{pmatrix}.$$

This system can be solved fiber by fiber, since all of the matrices M^{LD}, K, and K^* on the left hand side of (106) are block diagonal. System (106) is not invertible in general because the representation

40

$\boldsymbol{K\alpha}$ is not necessarily unique. To see this, suppose that $\boldsymbol{n}_j$ is a degree $N-1$ polynomial. Then the inextensible velocity (41) could be zero at all the nodes without $\boldsymbol{\alpha}$ being identically zero. We therefore use the least squares solution for $\boldsymbol{\alpha}$ while emphasizing that $\boldsymbol{\alpha}$ itself has no significance; only $\boldsymbol{K\alpha}$ has physical meaning.

4.5.2 Implicit method for denser suspensions

In the case when the fibers are packed densely enough to make the temporal discretization (106) unstable, we treat the bending force in the nonlocal hydrodynamics implicitly and use GMRES to solve for $\boldsymbol{\lambda}_{n+1/2}$ and $\boldsymbol{\alpha}_{n+1/2}$. The new linear system of equations is

$$\boldsymbol{M}^{\mathrm{LD}}_{n+1/2,*}\left(\boldsymbol{\lambda}_{n+1/2}+\frac{1}{2}\boldsymbol{F}\left(\boldsymbol{X}_n+\boldsymbol{X}_{n+1,*}\right)\right) \tag{106}$$

$$+\boldsymbol{M}^{\mathrm{NL}}_{n+1/2,*}\left(\boldsymbol{\lambda}_{n+1/2}+\frac{1}{2}\boldsymbol{F}\left(\boldsymbol{X}_n+\boldsymbol{X}_{n+1,*}\right)\right)+\boldsymbol{u}_0\left(\boldsymbol{X}_{n+1/2,*}\right)=\boldsymbol{K}_{n+1/2,*}\boldsymbol{\alpha}_{n+1/2},$$

$$\boldsymbol{K}^*_{n+1/2,*}\boldsymbol{\lambda}_{n+1/2}=\boldsymbol{0}.$$

Now, let us denote the solutions of the block diagonal system (106) by $\widetilde{\boldsymbol{\lambda}}_{n+1/2}$ and $\widetilde{\boldsymbol{\alpha}}_{n+1/2}$. By subtracting the fully implicit system (107) from the locally implicit system (102), we obtain the residual form of the saddle-point system

$$\begin{pmatrix}-\left(\boldsymbol{M}^{\mathrm{LD}}+\boldsymbol{M}^{\mathrm{NL}}\right) & \boldsymbol{K}-\frac{\Delta t}{2}\left(\boldsymbol{M}^{\mathrm{LD}}+\boldsymbol{M}^{\mathrm{NL}}\right)\boldsymbol{F}\boldsymbol{K} \\ \boldsymbol{K}^* & \boldsymbol{0}\end{pmatrix}_{n+1/2,*}\begin{pmatrix}\Delta\boldsymbol{\lambda}_{n+1/2} \\ \Delta\boldsymbol{\alpha}_{n+1/2}\end{pmatrix}= \tag{107}$$

$$\begin{pmatrix}\boldsymbol{M}^{\mathrm{NL}}_{n+1/2,*}\left(\boldsymbol{F}\left(\boldsymbol{X}_n+\frac{\Delta t}{2}\boldsymbol{K}_{n+1/2,*}\widetilde{\boldsymbol{\alpha}}_{n+1/2}-\boldsymbol{X}_{n+1/2,*}\right)+\widetilde{\boldsymbol{\lambda}}_{n+1/2}-\boldsymbol{\lambda}_{n+1/2,*}\right) \\ \boldsymbol{0}\end{pmatrix}$$

to be solved using GMRES for the perturbations

$$\Delta\boldsymbol{\lambda}_{n+1/2}=\boldsymbol{\lambda}_{n+1/2}-\widetilde{\boldsymbol{\lambda}}_{n+1/2}\quad\text{and} \tag{108}$$

$$\Delta\boldsymbol{\alpha}_{n+1/2}=\boldsymbol{\alpha}_{n+1/2}-\widetilde{\boldsymbol{\alpha}}_{n+1/2}.$$

The right hand side of system (108) is zero to second order in Δt. We therefore expect the perturbations to be $\mathcal{O}(\Delta t^2)$. While these perturbations have no impact on the temporal accuracy of our scheme, obtaining a good approximation to them is vital for stability. In Section 5.2, we quantify empirically (for $\epsilon=10^{-3}$) how many GMRES iterations are enough to obtain unconditional stability. Note that smaller values of ϵ require fewer GMRES iterations for stability since local drag is more dominant for smaller ϵ, and vice versa for larger ϵ.

To solve system (108) rapidly with GMRES, we use the block diagonal preconditioner

$$\boldsymbol{P}_{n+1/2} = \begin{pmatrix} -\boldsymbol{M}^{\mathrm{LD}} & \boldsymbol{K} - \frac{\Delta t}{2}\boldsymbol{M}^{\mathrm{LD}}\boldsymbol{F}\boldsymbol{K} \\ \boldsymbol{K}^* & \boldsymbol{0} \end{pmatrix}_{n+1/2,*}^{-1} \tag{109}$$

that appears in (106). Since $\boldsymbol{M}^{\mathrm{LD}}$ dominates over $\boldsymbol{M}^{\mathrm{NL}}$, this preconditioner is effective, with increased effectiveness for smaller ϵ.

Our overall scheme to solve for $\boldsymbol{\lambda}_{n+1/2}$ and $\boldsymbol{\alpha}_{n+1/2}$ can be summarized as follows:

1. Solve the block diagonal system (106) for $\boldsymbol{\lambda}_{n+1/2}$ and $\boldsymbol{\alpha}_{n+1/2}$. If the fiber suspension is sufficiently dilute (see Section 5.2), continue to the next time step.

2. Otherwise, set $\widetilde{\boldsymbol{\lambda}}_{n+1/2}$ and $\widetilde{\boldsymbol{\alpha}}_{n+1/2}$ to be the solutions of the block diagonal system (106), run a few iterations of GMRES to solve the residual system (108), and update $\boldsymbol{\lambda}_{n+1/2}$ and $\boldsymbol{\alpha}_{n+1/2}$ using (109).

To initialize, in the first and second time steps (for $n = 0, 1$), we solve system (107) by converging GMRES with a relative residual tolerance of 10^{-6}. When $n = 0$, we set $\boldsymbol{X}_{n+1/2,*} = \boldsymbol{X}_n$.

4.5.3 Updating $\boldsymbol{\tau}$ and $\boldsymbol{X}$

Once we have computed $\boldsymbol{\alpha}_{n+1/2}$, we use a discrete form of the tangent vector rotation (27) to update the tangent vectors. This is done fiber by fiber, and so here we use $\boldsymbol{X}$ to stand for a single fiber, rather than the entire collection of positions. To avoid double subscripts, in a slight abuse of notation we superscript the time step index n in this section.

Our goal is to rotate the set of tangent vectors $\boldsymbol{\tau}^n$ on the unit sphere by the (axis-angle) rotations $\boldsymbol{\Omega}^{n+1/2} = \boldsymbol{\Omega}\left(\boldsymbol{\tau}^{n+1/2,*}, \boldsymbol{\alpha}^{n+1/2}\right)$. To do this in a stable way, we cannot use $\boldsymbol{\alpha}^{n+1/2}$ directly in the computation, since the kinematic coefficients $\boldsymbol{\alpha}^{n+1/2}$ do not have physical meaning and take values which are sensitive to discretization and ill-conditioning. Since the velocity $\boldsymbol{K}^{n+1/2}\boldsymbol{\alpha}^{n+1/2}$ has a physical meaning and is less sensitive to numerical artifacts, we compute

$$\boldsymbol{\Omega}^{n+1/2} = \boldsymbol{\tau}^{n+1/2,*} \times \boldsymbol{D}_N \boldsymbol{K}^{n+1/2,*}\boldsymbol{\alpha}^{n+1/2} \tag{110}$$

on an upsampled grid, where $\boldsymbol{D}_N$ is the Chebyshev differentiation matrix on the N point grid. To do this, we upsample $\boldsymbol{\tau}^{n+1/2,*}$ and the derivative $\boldsymbol{D}_N \boldsymbol{K}^{n+1/2,*}\boldsymbol{\alpha}^{n+1/2}$ to a grid of size $2N$ and do the cross product. We then downsample the result to obtain $\boldsymbol{\Omega}_p^{n+1/2}$ for $p = 1, \ldots N$.

Once $\boldsymbol{\Omega}^{n+1/2}$ is known, we use the Rodrigues rotation formula [71] to compute the rotated tangent vectors. Letting $\Omega = \|\boldsymbol{\Omega}\|$ and $\widehat{\boldsymbol{\Omega}} = \boldsymbol{\Omega}/\Omega$, we compute the rotated tangent $\boldsymbol{\tau}$ at each node p by

$$\boldsymbol{\tau}_p^{n+1} = \boldsymbol{\tau}_p^n \cos\left(\Omega_p^{n+1/2}\Delta t\right) + \left(\widehat{\boldsymbol{\Omega}}_p^{n+1/2} \times \boldsymbol{\tau}_p^n\right) \sin\left(\Omega_p^{n+1/2}\Delta t\right) + \tag{111}$$
$$\widehat{\boldsymbol{\Omega}}_p^{n+1/2}\left(\widehat{\boldsymbol{\Omega}}_p^{n+1/2} \cdot \boldsymbol{\tau}_p^n\right)\left(1 - \cos\left(\Omega_p^{n+1/2}\Delta t\right)\right).$$

We then compute $\boldsymbol{X}^{n+1}$ from $\boldsymbol{\tau}^{n+1}$ via Chebyshev integration. Specifically, we compute the Chebyshev series coefficients of $\boldsymbol{\tau}^{n+1}$, apply the spectral integration matrix [27] to compute the Chebyshev series of $\boldsymbol{X}^{n+1}$, then evaluate this series at the nodes on the N point grid. To fix the integration constant, on each fiber we set the position at the first node

$$\boldsymbol{X}_1^{n+1} = \boldsymbol{X}_1^{n+1,*}, \tag{112}$$

where $\boldsymbol{X}^{n+1,*}$ is defined in (104).

5 Numerical tests

In this section, we validate each component of our method and demonstrate the method's spatial and temporal accuracy. We study spatio-temporal accuracy in Section 5.1 with simple examples of four falling fibers in free space and three fibers in periodic shear flow. Section 5.2 gives our most important result for computational complexity: the number of hydrodynamic evaluations per time step required to stably evolve the dynamics of a fiber suspension is at most five. By varying the fiber number density and bending modulus, we show that one block diagonal solve combined with at most three iterations of GMRES per time step are needed to maintain stability.[4]

In general, we will use an L^2 function norm to compute the differences between configurations throughout this section. Given two fiber configurations, we evaluate the Chebyshev interpolant of each on a 1000 point type 2 Chebyshev grid and calculate the discrete L^2 error using Clenshaw-Curtis quadrature. Whenever there are multiple fibers, we compute the error on the first fiber $\boldsymbol{X}^{(1)}$ unless otherwise specified.

Since fibers in shear flow are our primary interest, some of our examples will use shear flows. The general form of a time-oscillatory shear flow is given by

$$\boldsymbol{u}_0(\boldsymbol{x}, t) = \dot{\gamma}_0 \cos\left(\omega t\right)(y, 0, 0). \tag{113}$$

---

[4]The extra hydrodynamic evaluation to give a total of five comes in the conversion to residual form (108).

The corresponding strain is given by $g(t) = (\dot{\gamma}_0/\omega)\sin(\omega t)$ for $\omega > 0$ and $g(t) = \dot{\gamma}_0 t$ for $\omega = 0$. The code and input files for the examples of fibers in triply periodic shear flow presented here are available for download at `https://github.com/stochasticHydroTools/SlenderBody`.

5.1 Spatio-temporal accuracy

In this section, we study spatio-temporal convergence for two examples where nonlocal hydrodynamics has a nontrivial impact on the fiber trajectories. We first verify our method for fibers in gravity by comparing the results to those obtained using the method prescribed in [65] and in the process show improved robustness and temporal accuracy. We then demonstrate second-order temporal convergence and spectral spatial accuracy for a periodic three fiber system in shear flow. We also show that, if stable, our block diagonal solver (106) with one hydrodynamic evaluation per time step is the most efficient way to resolve the dynamics to a given tolerance.

5.1.1 Comparison to strong formulation

Our first goal is to validate our weak formulation of inextensibility by numerically comparing to the strong formulation. To do this, we consider four fibers centered around a circle of radius $d = 0.2$. The fibers have initial tangent vector $\boldsymbol{\tau}(t = 0) = (0, 0, 1)$ (they are aligned in the z direction) and positions $\boldsymbol{X}^{(1)}(t = 0) = (d, 0, s - 1)$, $\boldsymbol{X}^{(2)}(t = 0) = (0, d, s - 1)$, $\boldsymbol{X}^{(3)}(t = 0) = (-d, 0, s - 1)$, and $\boldsymbol{X}^{(4)}(t = 0) = (0, -d, s - 1)$, where $0 \leq s \leq L = 2$. For simplicity, we set $\mu = \kappa = 1$, and $\epsilon = 10^{-3}$. For this test only, we use fibers with ellipsodial cross sections and set the local drag coefficient $c(s) = -\ln(\epsilon^2)$. We simulate until $t = 0.25$.

Each fiber has a uniform gravitational force density $\boldsymbol{f}^g = (0, 0, -5)$ placed on it. In the absence of nonlocal interactions (i.e., if $\boldsymbol{M} = \boldsymbol{M}^{\text{LD}}$), the fibers fall straight downward. When the interactions between fibers are included, however, the fibers influence each other and have an x and y direction to their motion. Figure 3(a) shows the initial and final ($t = 0.25$) configurations of the fibers in this test.

Our goal here is to verify our results by comparing to results obtained using the method of [65]. Because the method of [65] uses regularization for the finite part integral (and we do not), we drop the finite part integral in this calculation and only include local drag and cross fiber interactions in the mobility. The fibers are sufficiently far from each other that the dipole term in the kernel $\boldsymbol{S}_D$, which has coefficient of 0 in [65], has a very small effect on the result (two orders of magnitude less than our smallest spatio-temporal error). We use free space boundary conditions (no periodicity)

(a) Trajectory   (b) Convergence
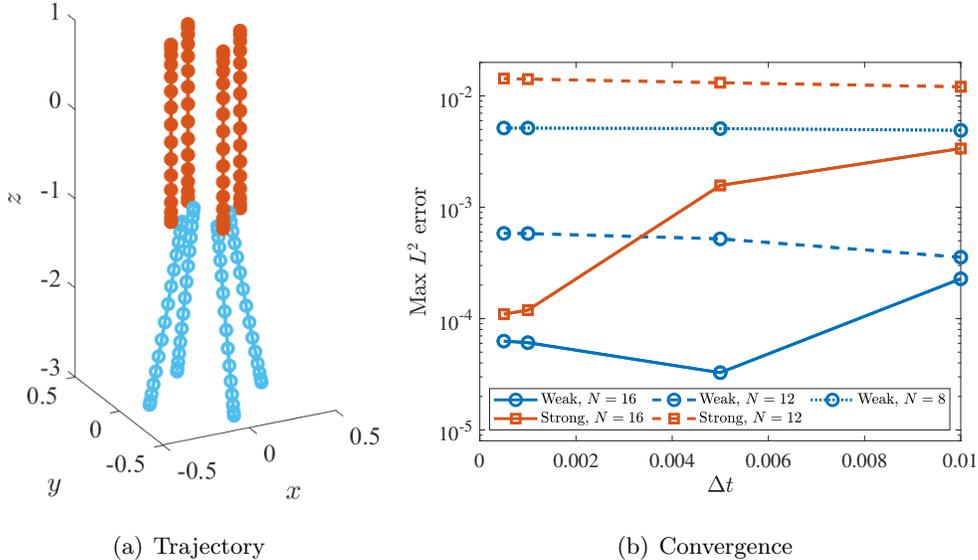
Figure 3: Four fibers in gravity. (a) Initial (filled orange) and final (light blue) configurations of the fibers. There are $N = 16$ points on each fiber. (b) Spatio-temporal convergence of our weak formulation (circles) compared to the strong formulation of [65] (squares). In both cases, the exact solution is a trajectory with $N = 24$, $\Delta t = 5 \times 10^{-4}$. For small Δt, the spatial error dominates and the spatio-temporal error saturates.

and compute all nonlocal integrals by direct quadrature (78), without any upsampling. We use the block diagonal solves (106) for temporal integration and do *not* perform any GMRES iterations.

We first verify that our results match those of the strong formulation [65] when the spatial and temporal discretizations are well-refined. Considering $N = 24$ and $\Delta t = 5 \times 10^{-4}$ in both algorithms, we obtain a maximal L^2 difference of 1.8×10^{-4} in the position of the first fiber, which, as we show in Fig. 3(b), is on the order of magnitude of the discretization error.

It is instructive to compare the spatio-temporal error between the two algorithms. We define the "exact" solution for both algorithms to be a trajectory with $N = 24$ and $\Delta t = 5 \times 10^{-4}$. Figure 3(b) shows the maximum L^2 errors over the time interval $[0, 0.25]$ for both algorithms with different spatial and temporal discretizations. For small Δt, the spatial error dominates and the spatio-temporal error saturates. We observe that our weak formulation outperforms the strong formulation of [65] in two ways. First, for coarse discretizations (e.g. $N = 12$, dashed lines in Fig. 3(b)), our saturated spatial error is more than an order of magnitude lower than the saturated spatial error of [65]. This is likely because the line tension equation of [65] has larger aliasing errors for coarser spatial discretizations, and because of our improved treatment of the free fiber boundary conditions using rectangular spectral collocation. Secondly, our errors saturate at a

much larger time step size than those of [65]. For example, when $N = 16$ our error saturates for $\Delta t = 5 \times 10^{-3}$, whereas the error from [65] does not saturate until $\Delta t = 1 \times 10^{-3}$. This occurs because our temporal integrator is second-order accurate. This simple example demonstrates the improved spatial accuracy of our new weak formulation over the strong one, and the improved accuracy of our temporal discretization, even in the absence of GMRES iterations.

5.1.2   Spatio-temporal convergence

We next verify the temporal convergence of our algorithm for periodic sheared domains by choosing an example where interactions between the fibers contribute significantly to the dynamics. We consider three sheared fibers with $L = 2$ and $\boldsymbol{X}^{(1)}(s) = (s - 1, -0.6, -0.04)$, $\boldsymbol{X}^{(2)}(s) = (0, s - 1, 0)$, and $\boldsymbol{X}^{(3)}(s) = (s - 1, 0.6, 0.06)$. As shown in Figure 4(a), this corresponds to an "I" shaped initial configuration of the fibers, with the fibers stacked in the z direction.

We set the periodic domain length $L_d = 2.4$, the Ewald parameter $\xi = 3$, and set $\mu = 1$, $\epsilon = 10^{-3}$, and $\kappa = 0.01$. We use a constant shear flow (114) with $\gamma_0 = 1$ and $\omega = 0$. Because of the small bending rigidity, the fibers deform from their straight configurations in a shear flow. In this example, two of the fibers are initially aligned with the x direction. Without nonlocal hydrodynamics, they would stay aligned with the x direction and simply translate. When nonlocal interactions are included, the flows generated by the middle fiber $\boldsymbol{X}^{(2)}$ induce deformations of the top and bottom fibers. This is evident in Fig. 4(a), which shows the final fiber positions at $t = 2.4$.

To quantify the temporal convergence, we fix $N = 16$ and simulate from $t = 0$ to $t = 2.4$ with $\Delta t = 0.4, 0.2, 0.1, 0.05$, and 0.025 using the block diagonal solver (106) (no GMRES iterations) and successive refinements to measure the error. Figure 4(b) shows that we obtain second-order temporal convergence for the block diagonal solver (106) (blue circles) and for the GMRES system (108) with 1 iteration (red squares) and 3 iterations (yellow triangles). The temporal error is about an order of magnitude smaller when we perform one GMRES iteration in addition to the block diagonal solvers. Note, however, that this comes at the cost of two additional nonlocal hydrodynamic evaluations (one to convert to residual form, one in the GMRES iteration).

For spatio-temporal convergence, we simulate a refined trajectory with $N = 32$ and $\Delta t = 0.00125$ and compute the maximum L^2 errors in trajectories with $N = 16$ and 24. As shown in Fig. 4(c), increasing the number of points by 8 decreases the spatio-temporal error by a factor of 4 (compare blue circles with purple diamonds), which is consistent with spectral spatial accuracy.

Performing one GMRES iteration with $\Delta t = 0.2$ approximately matches the spatial and tem-
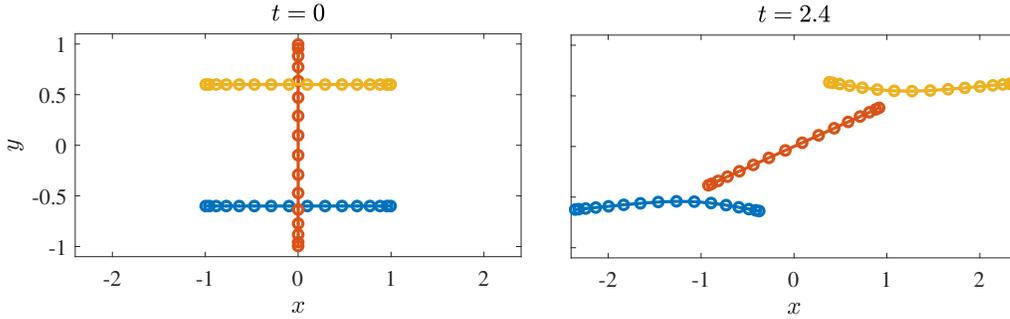
(a) Problem setup
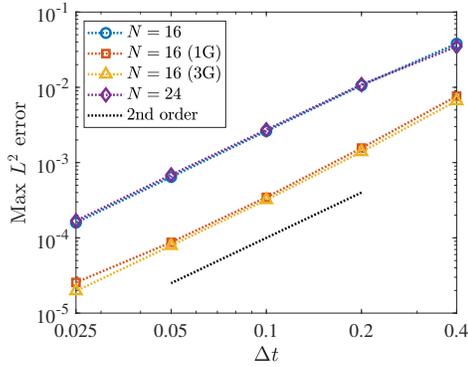
(b) Temporal error
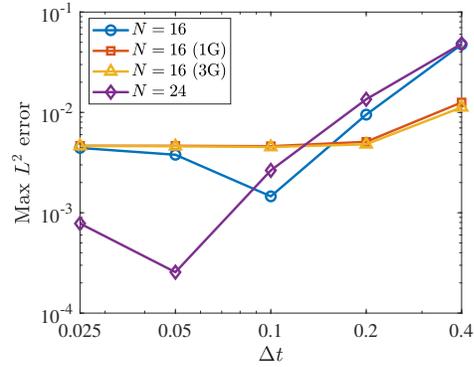
(c) Spatio-temporal error

Figure 4: Three fibers in shear flow. (a) Fiber configurations at $t = 0$ and $t = 2.4$. (b) Second-order temporal convergence. We measure the maximum L^2 error over time in the first fiber position using successive refinements and observe second-order temporal convergence for block diagonal solves with $N = 16$ (blue circles) and $N = 24$ (purple diamonds). We also see second-order convergence, and reduced temporal errors, for both 1 (red squares) and 3 (yellow triangles) GMRES iterations. (c) The spatio-temporal errors (measured against a more accurate solution with $N = 32$, $\Delta t = 0.00125$) are shown for $N = 16$ and $N = 24$.

poral errors and costs a total of three hydrodynamic evaluations per 0.2 units of time. Running the block diagonal solver with $\Delta t = 0.1$ matches the temporal error with the spatial error and costs two hydrodynamic evaluations per 0.2 units of time. We therefore conclude from Figure 4(c) that the most efficient way to obtain the maximum accuracy for a given spatial resolution is to run the block diagonal solver with the smaller time step size, assuming it is stable.

5.2 Stability

Because our block-diagonal semi-implicit temporal discretization (BDSI) described in Section 4.5.1 treats the bending force explicitly in the nonlocal term, it will become unstable when the fiber suspension is too concentrated and the cumulative effect of nonlocal hydrodynamics is comparable to that of local drag. In this case, we switch to the GMRES solver described in Section 4.5.2. As discussed there, since the block diagonal solver is already second-order accurate, the perturbations to $\boldsymbol{\alpha}$ and $\boldsymbol{\lambda}$ that come from the residual GMRES solve (108) do not impact the overall temporal accuracy (see Fig. 4), but do impact stability. For this reason, we run only a fixed number of GMRES iterations until we obtain stability. Our goal in this section is to determine an upper bound on the number of required GMRES iterations.

To do this, we consider a suspension of $F = 1000$ fibers and vary the density of fibers by changing the periodic domain length L_d. If $f = F/L_d^3$ is the number density of fibers and L is the length of a fiber, a dimensionless density $fL^3 < 1$ is considered a dilute fiber suspension, while a semi-dilute suspension is one with $rfL^2 = \epsilon fL^3 \ll 1$, and a semi-concentrated one has $\epsilon fL^3 = \mathcal{O}(1)$ [80]. Here we explore the semi-dilute and semi-concentrated regimes and derive empirical bounds on how many GMRES iterations are required to maintain stability for a variety of bending moduli. Our conclusion is that at most five nonlocal hydrodynamic evaluations are sufficient to maintain stability, even for semi-concentrated suspensions.

We simulate $F = 1000$ initially straight fibers of length $L = 2$ and radius $r = 2 \times 10^{-3}$ (so that $\epsilon = 1 \times 10^{-3}$), and use $N = 16$ points per fiber. In the oscillatory shear flow (114), we set $\omega = 2\pi$ and $\dot{\gamma}_0 = \omega/10$, so that the maximum strain is $g = 0.1$ and the time for one cycle is 1. We expect to need at least 20 time steps per cycle to obtain reasonable accuracy, so we set $\Delta t = 0.05$, although in reality we find that smaller time step sizes are needed to accurately resolve the dynamics of dense suspensions. Since we find that changing the frequency ω has a negligible impact on the results, we non-dimensionalize Δt by the bending timescale $\tau = 8\pi\mu L^4 / \left(\ln\left(\epsilon^{-2}\right)\kappa \right)$, where here we use $\mu = 1$.

We simulate 5 cycles of motion, until $t = 5$. Figure 5 shows the number of hydrodynamic

evaluations required for stability for a given bending modulus and fiber density. The fiber number density fL^3 is reported on the bottom x axis and on the top x axis we give ϵfL^3. For semi-dilute suspensions ($\epsilon fL^3 \approx 0.01$), we see that BDSI is stable, i.e., only a single hydrodynamic evaluation is needed for stability (red squares in Fig. 5). For a fixed number density, we first see instabilities for BDSI for smaller τ, so that stiffer fibers require more GMRES iterations for stability. As the fiber suspension becomes semi-concentrated ($\epsilon fL^3 \geq 0.1$), we see that we need at least three hydrodynamic evaluations regardless of the fiber stiffness. For our stiffest and densest suspensions ($\epsilon fL^3 \approx 0.5$), we need at most five hydrodynamic evaluations per time step to obtain stable dynamics. For comparison, Nazockdast et al. report 9 to 16 GMRES iterations for a system with a similar number of fibers [65, Table 1].

We caution that these evaluation counts are the minimum number needed for *stability*. For dense fiber suspensions, $\Delta t = 0.05$ might be too large to obtain reasonable accuracy, since fibers that are in close contact are subject to nearly nonsmooth velocity fields that come from the near singular velocity kernel (77) and the possible combination with the fiber centerline velocity (79). In practice, we find this causes fibers that are in close contact to oscillate around each other when Δt is too large. A smaller Δt can resolve these issues, and we know from Fig. 5 that the number of iterations required for stability drops with the time step size.

6   Application: cross-linked actin mesh

The cell cytoskeleton is a dynamic network of cross-linked actin filaments and myosin motors that allows cells to migrate, divide, and adapt to new environments [4, 3]. A number of experimental [24, 35, 37, 46] and computational [52, 33] studies have shown that the viscoelastic rheology of actin networks comes from specialized cross-linking proteins dynamically binding and unbinding to actin fibers, with the rates of binding and unbinding determining the ratio of viscous to elastic behavior. When the cross-linkers (CLs) are permanently attached, the network has traditionally been viewed as purely elastic, while instant unbinding of CLs has been seen as pure viscous behavior [3]. The reality is more nuanced than this, since the network is embedded in an underlying fluid which contributes to the viscous modulus of the network and affects the movement of the filaments between binding and unbinding events.

To our knowledge, there has been no systematic study of the contribution of the underlying fluid to the dynamics and viscoelastic properties of cross-linked actin networks. We defer a full
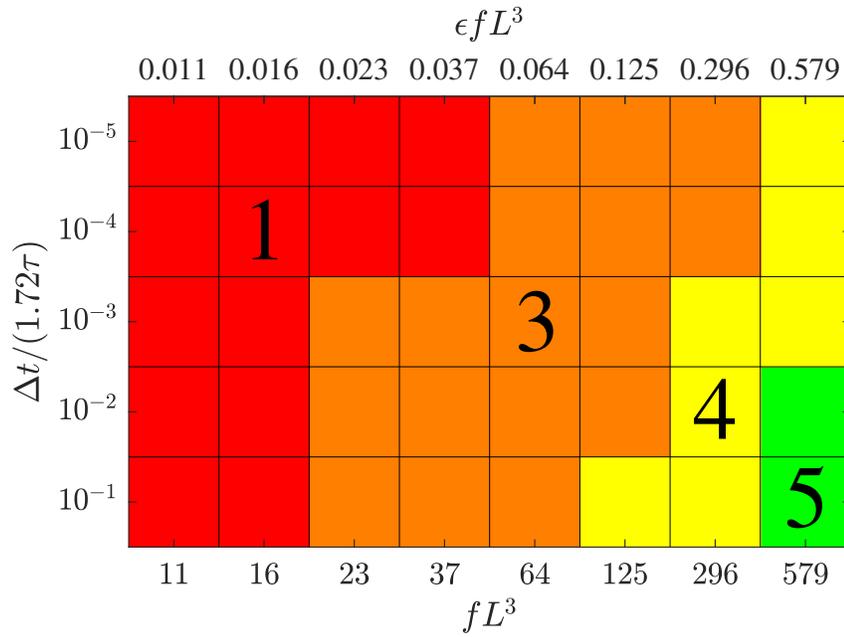
Figure 5: Number of hydrodynamic evaluations needed for stability, indicated by color. We consider a system of $F = 1000$ fibers of length $L = 2$, vary the periodic domain length L_d and fiber stiffness κ, and show the number of hydrodynamic evaluations needed for stability for each set of parameters. One hydrodynamic evaluation occurs in the block diagonal solver (106), another in the conversion to residual form (108), and one evaluation occurs per GMRES iteration. We report fiber density in units of $fL^3 = FL^3/L_d^3$ on the x axis and fiber stiffness in units of $\Delta t/(1.72\tau)$ on the y axis, where the elastic timescale $\tau = 8\pi\mu L^4 / \left(\ln\left(\epsilon^{-2}\right)\kappa\right)$.

study of this for the future; here we develop the cross linking model and provide some initial results. In particular, we will leave transient cross-linker dynamics for a future study and consider the special case of a permanently cross-linked network, which could model, for example, a network of actin fibers cross-linked by scruin proteins [24]. Since our fibers are represented by Chebyshev interpolants $\boldsymbol{X}(s)$, we seek a continuum force density on the fiber due to cross linking. This force density must be smooth relative to the discretization to preserve the spectral accuracy of our algorithm.

We begin in Section 6.1 by presenting our model of a cross-linker as an elastic spring between the fibers. Although an elastic spring model might not be appropriate for some cross-linking proteins, for example short and stiff rods like α-actinin [59], it can model longer and more flexible CLs like filamin [90]. In Section 6.2, we then discuss how we compute rheological information from our model to facilitate comparison with experiments. In Section 6.3, we also report the sensitivity of this information to the number of collocation points N, width σ of the CL Gaussian smoothing function, and local drag regularization length scale δL. We conclude this section with our results for a permanently cross-linked actin network. We consider a fixed ratio of twelve CLs per fiber and study the viscous and elastic behavior of the network using both local drag and fully nonlocal hydrodynamics to evolve the system. Our conclusion is that there exists a critical time scale τ_c on which the network relaxes to a dynamic steady state under oscillatory shear flow. For shear frequencies $\omega \ll \tau_c^{-1}$, the behavior is primarily elastic and dominated by the quasi-steady state of the network. For $\omega \gg \tau_c^{-1}$, dynamics, including hydrodynamics, matter and we see more viscous behavior.

6.1   Cross-linker model

Suppose we have two fibers, $\boldsymbol{X}^{(i)}$ and $\boldsymbol{X}^{(j)}$, and that a CL connects two fibers by attaching to arclength coordinate s_i^* on fiber i and s_j^* on fiber j, where these coordinates are not necessarily Chebyshev points. We define the force density due to the CL at arclength coordinate s_p on fiber i as

$$\boldsymbol{f}_p^{(\mathrm{CL},i)}\left(\boldsymbol{X}\right) = -K_c \left(1 - \frac{\ell}{\left\|\boldsymbol{X}^{(i)}(s_i^*) - \boldsymbol{X}^{(j)}(s_j^*)\right\|}\right) \delta_h(s_p - s_i^*) \sum_{q=1}^{N} \left(\boldsymbol{X}^{(i)}\left(s_p\right) - \boldsymbol{X}^{(j)}(s_q)\right) \delta_h(s_q - s_j^*) w_q,$$

$$(114)$$

where K_c is the spring constant for the CL (units force/length), ℓ is the rest length, and δ_h is a Gaussian smoothing function replacing a Dirac delta function. The CL force density (115) links the

point s_p on fiber i to every point on fiber j, with a weight related to the distance on fiber j between the anchor coordinate s_j^* and Chebsyshev point s_q by the Gaussian function δ_h. The prefactor outside of the sum is zero when the anchor points are exactly length ℓ apart. If the two anchor points are farther than ℓ apart, the force between them is attractive; otherwise it is repulsive.

The force density (115) exerts no net force or torque on the system both in continuum (replace the sum in (115) by an integral) and discretely. Specifically,

$$\sum_{p=1}^{N} \boldsymbol{f}_p^{(i,\mathrm{CL})} w_p + \sum_{q=1}^{N} \boldsymbol{f}_q^{(j,\mathrm{CL})} w_q = 0 \tag{115}$$

$$\sum_{p=1}^{N} \left(\boldsymbol{X}_p^{(i)} \times \boldsymbol{f}_p^{(i,\mathrm{CL})} \right) w_p + \sum_{q=1}^{N} \left(\boldsymbol{X}_q^{(j)} \times \boldsymbol{f}_q^{(j,\mathrm{CL})} \right) w_q = 0. \tag{116}$$

These identities, which imply that each pair of cross-linked fibers is force-and-torque-free, hold regardles of the form of δ_h.

For a given fiber discretization, we choose δ_h so that the forcing is smooth in the Chebyshev basis. We consider δ_h to be a Gaussian density of the form

$$\delta_h(r) = \frac{1}{Z} \exp\left(-\frac{r^2}{2\sigma^2} \right), \tag{117}$$

where σ is a parameter that controls the smoothness and spread of δ_h. The factor Z is a normalization factor that ensures δ_h discretely integrates to 1 along the fiber length, i.e., $\sum_p \delta_h \left(s_p - s_i^* \right) w_p = 1$. Far from the endpoints, $Z = \sqrt{2\pi\sigma^2}$, but if the CL is bound to the end of the fiber some of the Gaussian weight might be truncated. For a given N, we choose σ to be the minimum value that gives 3 digits of accuracy in $\boldsymbol{f}$, where the error is measured relative to a refined $\boldsymbol{f}$ computed on a 1000 point Chebyshev grid. Note that small values of σ lead to instabilities if N is not large enough to resolve δ_h. For $N = 16$, we use $\sigma/L = 0.1$; we will study the influence of σ on physical observables numerically in Section 6.3.1.

We modify the BDSI temporal integrator (102) to treat the cross linker forces in an *explicit* second-order fashion,

$$\boldsymbol{M}_{n+1/2,*}^{\mathrm{LD}} \left(\boldsymbol{\lambda}_{n+1/2} + \frac{1}{2} \boldsymbol{F} \left(\boldsymbol{X}_n + \boldsymbol{X}_{n+1,*} \right) + \boldsymbol{f}_{n+1/2,*}^{(\mathrm{CL})} \right) \tag{118}$$

$$+ \boldsymbol{M}_{n+1/2,*}^{\mathrm{NL}} \left(\boldsymbol{\lambda}_{n+1/2,*} + \boldsymbol{F} \boldsymbol{X}_{n+1/2,*} + \boldsymbol{f}_{n+1/2,*}^{(\mathrm{CL})} \right) + \boldsymbol{u}_0 \left(\boldsymbol{X}_{n+1/2,*} \right) = \boldsymbol{K}_{n+1/2,*} \boldsymbol{\alpha}_{n+1/2},$$

where $\boldsymbol{f}_{n+1/2,*}^{(\mathrm{CL})} = \boldsymbol{f}^{(\mathrm{CL})} \left(\boldsymbol{X}_{n+1/2,*} \right)$ and $\boldsymbol{X}_{n+1/2,*}$ is the extrapolation (101). As before, when the block diagonal semi-implicit temporal integrator (106) is unstable, we switch to the residual system

for GMRES (108), which is unchanged by the CLs. The time step size Δt for the CL discretization (119) is limited by stability for cases when the fibers are more flexible than the CLs; we leave implicit treatment of CLs for future work.

6.2   Rheological experiments

Experimental studies on actin networks [35, 37] typically report the viscous and elastic moduli. These quantities can be computed from the system stress tensor $\boldsymbol{\sigma}$, which in turn can be computed from the force densities on each fiber's centerline. In this section, we briefly lay out the minimum details needed for the calculation of the viscous and elastic moduli.

Recalling the time-oscillatory shear flow (114), the only nonzero component of the rate of strain tensor is constant in space and is given by

$$\dot{\gamma}_{21}(t) = \frac{\partial u_0^x}{\partial y} = \dot{\gamma}_0 \cos\left(\omega t\right), \tag{119}$$

and the relevant component of the *strain* tensor is therefore

$$\gamma_{21}(t) = \int_0^t \dot{\gamma}_{21}(t')\, dt' = \frac{\dot{\gamma}_0}{\omega} \sin\left(\omega t\right) := \gamma_0 \sin\left(\omega t\right), \tag{120}$$

where we have defined γ_0 as the maximum strain in the system. We define the bulk elastic (G') and viscous modulus (G'') from γ_0 by [64]

$$\frac{\sigma_{21}}{\gamma_0} = G' \sin\left(\omega t\right) + G'' \cos\left(\omega t\right). \tag{121}$$

Notice that the elastic modulus G' gives the part of the stress that is in phase with the strain, and the viscous modulus G'' gives the part of the stress that is in phase with the rate of strain.

The stress tensor itself can be decomposed into a part coming from the background fluid and a part coming from the internal fiber stresses,

$$\sigma_{21} = \sigma_{21}^{(\mu)} + \sigma_{21}^{(f)} = \mu \frac{\partial u_0^x}{\partial y} + \sigma_{21}^{(f)}. \tag{122}$$

For pure viscous fluid, the fiber contribution to the stress is zero and the stress is given entirely by the viscous stress tensor,

$$\frac{\sigma_{21}^{(\mu)}}{\gamma_0} = \frac{\omega \mu}{\dot{\gamma}_0} \left(\dot{\gamma}_0 \cos\left(\omega t\right)\right) = \omega \mu \cos\left(\omega t\right). \tag{123}$$

Thus the viscous modulus due to the fluid is $G'' = \omega \mu$.

The stress due to the fibers depends on the force the fibers exert on the fluid. Because (116) shows that the total force exerted by the cross-linker on the pair of fibers it connects is zero, we

use Batchelor's formula [8] for the volume-averaged stress due to the fibers and CLs. In the slender limit (i.e., the case when the surface area force density is constant on fiber cross sections), the bulk stress due to the fibers and CLs in a volume V is given by $\boldsymbol{\sigma}^{(f)} = \boldsymbol{\sigma}^{(i)} + \boldsymbol{\sigma}^{(\text{CL})}$, where

$$\boldsymbol{\sigma}^{(i)} = -\frac{1}{V} \left(\sum_{i=1}^{F} \sum_{p=1}^{N} \left(\boldsymbol{X}_p^{(i)} \left(\boldsymbol{\lambda}_p^{(i)} + \left(\boldsymbol{F} \boldsymbol{X}^{(i)} \right)_p \right) w_p \right) \right), \tag{124}$$

$$\boldsymbol{\sigma}^{(\text{CL})} = -\frac{1}{V} \left(\sum_{c=1}^{C} \sum_{p=1}^{N} \left(\boldsymbol{X}_p^{(c_1)} \boldsymbol{f}_p^{(\text{CL},c_1)} w_p + \boldsymbol{X}_p^{(c_2)} \boldsymbol{f}_p^{(\text{CL},c_2)} w_p \right) \right), \tag{125}$$

where the double sum for $\boldsymbol{\sigma}^{(i)}$ is over points on fibers and the double sum for $\boldsymbol{\sigma}^{(\text{CL})}$ is over points on pairs of cross-linked fibers. In the cross-linker stress, C is the number of CLs and the notation c_1 and c_2 means that cross-linker c links fibers c_1 and c_2. The sums must be separated since different periodic images of $\boldsymbol{X}^{(i)}$ could be involved for different cross linkers. In $\boldsymbol{\sigma}^{(CL)}$, the positions $\boldsymbol{X}_p^{(c_1)}$ and $\boldsymbol{X}_p^{(c_2)}$ must be the periodic images of the two fibers that are connected by the CL. Because (117) shows that the total torque exerted by the cross-linker on the pair of fibers it connects is zero, we expect the stress tensor (125) to be symmetric to spectral accuracy, since the constraint force $\boldsymbol{\lambda}$ and elastic forces $\boldsymbol{F}\boldsymbol{X}$ exert exactly zero torque in continuum but not discretely [8].

We discretize the stress tensor at the midpoint of each time step. Specifically, we substitute $\boldsymbol{X}^{n+1/2,*}$ for $\boldsymbol{X}$ and $\boldsymbol{\lambda}^{n+1/2}$ for $\boldsymbol{\lambda}$ in the internal fiber stress $\boldsymbol{\sigma}^{(i)}$, and evaluate the cross-linking forces at the extrapolated midpoint $\boldsymbol{X}^{n+1/2,*}$ in $\boldsymbol{\sigma}^{(\text{CL})}$. Assuming that the final time T is an integer multiple of the period $2\pi/\omega$, we then compute the bulk moduli by discretizing the integrals

$$G' = \frac{2}{\gamma_0 T} \int_0^T \sigma_{21} \sin\left(\omega t\right) dt \qquad G'' = \frac{2}{\gamma_0 T} \int_0^T \sigma_{21} \cos\left(\omega t\right) dt. \tag{126}$$

by the midpoint rule.

6.3 Results

We now consider a permanently cross-linked network of F filaments, fix the physical parameters of the network, and analyze how the viscoelastic behavior depends on the frequency of oscillations. We set the fiber length $L = 2$ μm with aspect ratio $\epsilon = 10^{-3}$ and bending stiffness $\kappa = 0.01$ pN $\cdot$ μm^2, and use a box of size $L_d = 4$ μm. For the fluid viscosity, we use $\mu = 1$ Pa$\cdot$s. The CLs have rest length $\ell = 0.5$ μm and spring constant $K_c = 1$ pN/μm. To bind the CLs, at $t = 0$ we resample each fiber centerline to 16 uniformly separated points. We then randomly select a pair of these points, $\boldsymbol{X}^{(i)}(s_i^*)$ on filament i and $\boldsymbol{X}^{(j)}(s_j^*)$ on filament j, where $i \neq j$. If the two selected

points are initially separated by a distance less than ℓ, we bind a CL connecting fibers i and j, with one end centered on $\boldsymbol{X}^{(i)}(s_i^*)$ and the other end centered on $\boldsymbol{X}^{(i)}(s_j^*)$. We continue this process until $12F$ CLs have been attached.[5] Our use of such a large number of CLs effectively makes the network into a single interconnected cluster, so that our periodic domain can be viewed as a sample of a bulk interconnected fiber gel.

6.3.1 Effect of changing N, σ, and δ

For our experiments in the rest of this section, we will typically use $N = 16$ Chebyshev points per fiber, cross-linker standard deviation $\sigma/L = 0.1$, and local drag regularization parameter $\delta = 0.1$. Here we test the effect of changing N, σ, and δ from these baseline parameters by considering a set of $F = 100$ straight fibers and 1200 CLs. In this section, we are concerned only with the spatio-temporal accuracy of our CL formulation, and not necessarily the actual values of the viscous and elastic moduli, and so for this test we start measuring stress at $t = 0$, despite the fact that the network could be far from a steady state. Since we have already tested the spatio-temporal accuracy of our hydrodynamics in Section 5, we run here with local drag only. We use $\omega = 2\pi$ and $\dot{\gamma}_0 = 0.2\pi$ and run until $T = 6$ seconds (6 periods). To obtain a set of refined trajectories, we use $\Delta t = 0.005$ for $N = 16$, $\Delta t = 0.0025$ for $N = 24$, and $\Delta t = 0.001$ for $N = 32$. In addition to $\sigma/L = 0.10$ for all discretizations, we also measure stress for $N = 24$ with $\sigma/L = 0.07$ and $N = 32$ with $\sigma/L = 0.05$.

We first study the spatio-temporal convergence of stress by fixing $\sigma/L = 0.10$. The errors in the stress under spatio-temporal refinement are shown in Figure 6(a), where we observe rapid convergence consistent with second-order convergence in time. The errors in the stress near $t = 0$, immediately after the flow is turned on, are more chaotic since the straight fibers are initially pulled by CLs into a curved shape, and our time step sizes are too large to accurately capture these dynamics.

Perhaps a more important question is the influence of the Gaussian regularization parameter σ. In Fig. 6(b), we measure the differences in stress for $N = 16$, $\sigma/L = 0.1$ and $N = 24$, $\sigma/L = 0.07$ relative to our (most resolved) reference solution with $N = 32$, $\sigma/L = 0.05$. We see a relative difference of at most 10% in the stress tensor for $\sigma/L = 0.1$ and 5% for $\sigma/L = 0.07$.

We also study the impact of the regularization parameter δ in the local drag coefficient (14) on

---

[5]Since there are only 16 sites on each fiber and one CL takes up two sites, this requires that we allow more than one CL to bind to a specific site.

the stress tensor. In Fig. 6(c), we plot the relative errors in stress using $\delta = 0.05$ as a reference solution. We observe differences of about 5% in the stress tensor when we use ellipsoidal fibers instead of $\delta = 0.05$, and we see the differences in stress decrease as δ decreases, with a difference of only about 1% between $\delta = 0.1$ and $\delta = 0.05$. This shows that the precise value of δ has a small effect on the macroscopic rheology.

6.3.2 Viscoelastic behavior

We now turn to the measurement of the viscous and elastic moduli for a network of $F = 700$ fibers and 8400 CLs. In order to avoid transient behavior, we first find a steady state configuration of the network by initializing straight fibers with CLs and running the system forward in time using local drag and $\Delta t = 0.005$ without any background flow. After $t = 2500$ seconds, the maximum L^2 norm of the fiber velocity is approximately 4×10^{-6} μm/s, which indicates a near steady state.

In order to measure the steady state viscous and elastic moduli, we must wait for some intrinsic time on which the network reaches a new steady state in the shear flow. This relaxation time scale, which we denote by τ_c, combines the cross-linker and fiber relaxation time scales. For cross-linkers, a characteristic time scale of link relaxation is $\tau_{\text{CL}} = \mu\ell/K_c = 0.5$ seconds for our parameters. For fibers, we assume that the network is sufficiently constrained that the length scale on which the fibers can relax is the mesh size ℓ_m, or characteristic distance between filaments.[6] We estimate ℓ_m by assuming that L_d/L filaments can fit in one direction while L_d/ℓ_m can fit in the other two, so that $\ell_m \sim \sqrt{L_d^3/(FL)} \approx 0.21\,\mu$m, and a characteristic time scale of fiber relaxation is $\tau_F = \mu\ell_m^4/\kappa \approx 0.2$ seconds. Thus our expectation is for τ_c to be on the order $0.1 - 1$ second.

To measure the time scale τ_c more precisely, we start with the steady state configurations, turn on a shear flow (114) with $\omega = 0.2\pi$ rad/s and $\dot{\gamma}_0 = 0.02\pi$ 1/s, and run for one cycle (until $T = 10$ seconds) with $\Delta t = 0.005$ s. We then turn off the shear flow and measure the velocity of the fibers

---

[6]Another possibility is to assume that the fiber relaxation length scale is governed by the distance between two CLs on each fiber. Since there are an average of 24 CL connections on each fiber ($12F$ CLs, each of which binds to 2 fibers), this gives a distance $\ell_c = 2/24 \approx 0.08\,\mu$m, and a time scale $\mu\ell_c^4/\kappa$ of about 0.005 seconds, which is much faster than that measured in our numerical experiments. We thank an anonymous referee for suggesting these time scale estimates.
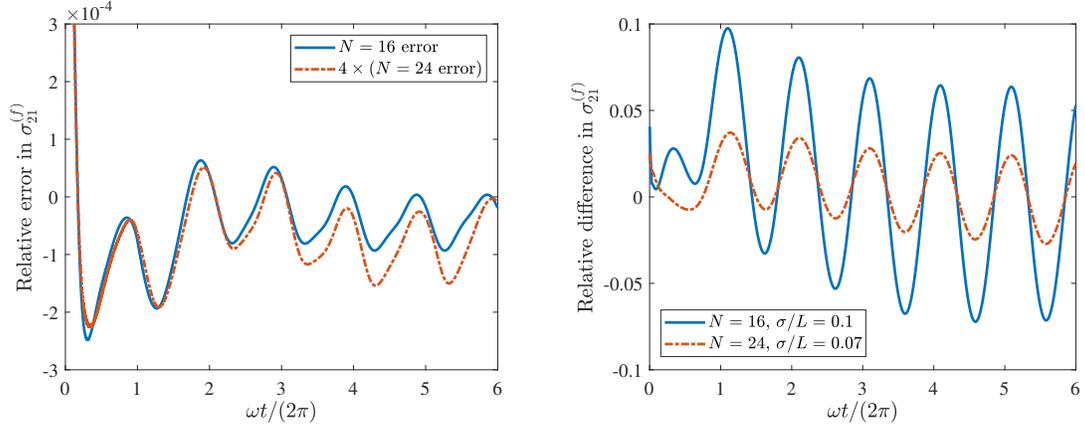
(a) Constant $\sigma/L = 0.1$ and constant $\delta = 0.1$

(b) Varying σ/L, constant $\delta = 0.1$
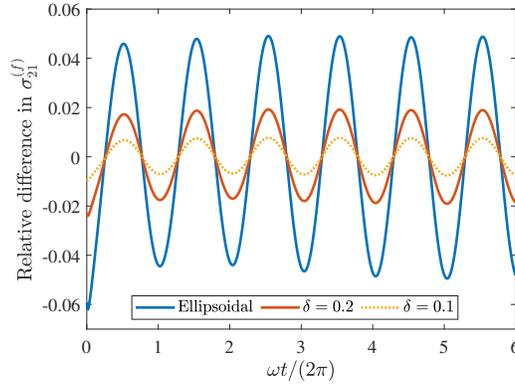
(c) Varying δ, constant $\sigma/L = 0.1$

Figure 6: Differences in the fiber stress $\sigma_{21}^{(f)}$ in a suspension of $F = 100$ fibers and 1200 CLs with (a) constant cross-linker standard deviation σ/L and local drag regularization parameter δ, (b) varying σ/L, and (c) varying δ. Normalization is the maximum absolute value of the stress for the reference solution in all cases. (a) For a reference solution with $N = 32$ and $\sigma/L = 0.1$, we see (after initial transients) rapid convergence of the stress consistent with second-order convergence in time. (b) For a reference solution with $N = 32$ and $\sigma/L = 0.05$, we see a 10% difference in stress using $N = 16$ and $\sigma/L = 0.1$, and an approximately 3% difference for $N = 24$ and $\sigma/L = 0.07$. (c) For a reference solution with $N = 16$, $\sigma/L = 0.1$, and $\delta = 0.05$ in the regularized local drag coefficient (14), we see a 5% difference in stress using ellipsoidal fibers and a difference of 1% and 2% for $\delta = 0.1$ and $\delta = 0.2$, respectively.
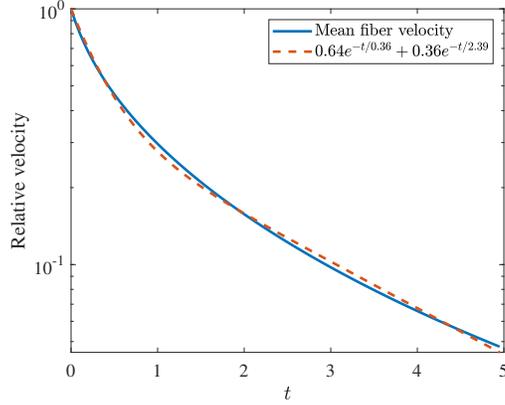
Figure 7: Decay of the mean fiber velocity to a steady state after being sheared for one cycle with $\omega = 0.2\pi$. We plot the mean fiber velocity, given by (128) and normalized by $\bar{v}(t = 0)$, over five seconds and compare the result to the double-exponential fit $0.64e^{-t/0.36} + 0.36e^{-t/2.39}$ to estimate the network relaxation time scale τ_c.

for another 5 seconds. We track the mean L^2 fiber velocity, given by

$$\bar{v}(t) = \frac{1}{F}\sum_{i=1}^{F} v^{(i)}(t), \quad \text{where} \tag{127}$$

$$v^{(i)}(t) = \left(\sum_{p=1}^{N} \left\|\boldsymbol{X}_p^{(i)}(t) - \boldsymbol{X}_p^{(i)}(t+0.05)\right\|^2 w_p\right)^{1/2},$$

and normalize by $\bar{v}(0)$ to obtain the exponential-like decay shown in Fig. 7. The time scale of relaxation to steady state is $\tau_c \approx 0.5 - 2$ s, with the best fit being a sum of two exponentials $\tau_1 = 0.4$ s and $\tau_2 = 2.4$ s. These relaxation time scales are in line with our physical estimates of $\tau_F = 0.2$ s and $\tau_{\text{CL}} = 0.5$ s.

Thus in order to measure the steady state moduli, we wait one second or one cycle (whichever is longer) prior to measuring the stress (and moduli) over three cycles of shear flow. We use a maximum strain $\gamma_0 = \dot{\gamma}_0/\omega = 0.1$ to stay in the linear regime (data not shown), and we give frequencies in Hz. To initialize for a given ω, we use the final network configuration from the previous (next smallest) frequency. A sample configuration of the fibers in the network, taken with $\omega = 1$ Hz at the point of maximum strain, is shown in Fig. 8. Dynamic movies of the simulations for varying frequencies are available for viewing at `https://cims.nyu.edu/~om759/FiberVideos`.

Figure 9(a) shows the steady-state elastic and viscous moduli when the dynamics of the network are computed with fully nonlocal hydrodynamics. Figure 9(a) also shows separately the contribution to the moduli of the CL stress (126), which is $5 - 10$ times more elastic than viscous. There is

58
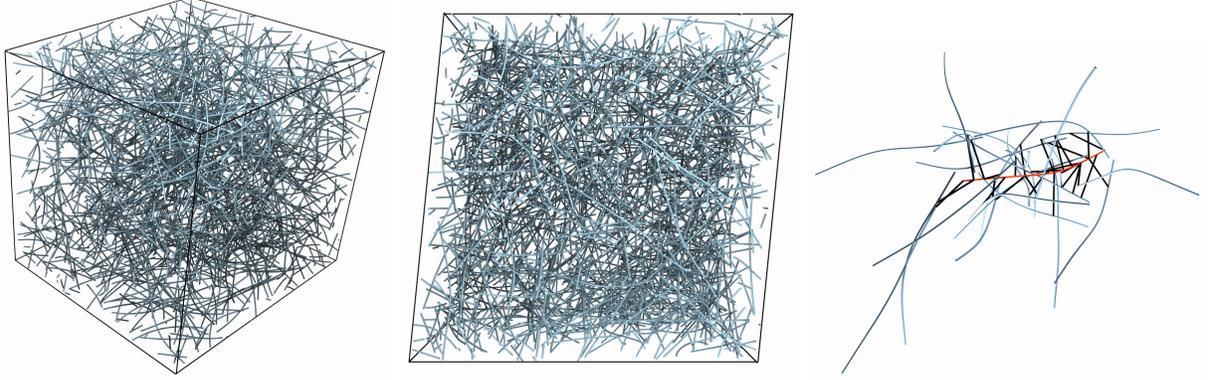
Figure 8: Steady state fiber configurations for the network of 700 fibers and 8400 CLs with $\omega = 1$ Hz and $g = 0.1$. We show (left) a three-dimensional snapshot of all the fibers in the unit cell, (middle) a view along the z axis, and (right) a snapshot of all the fibers (white/blue) and links (black) bound to a single fiber (orange) located near the center of the simulation cell.


(a) Moduli with full hydrodynamics

(b) Changes in moduli without hydrodynamics
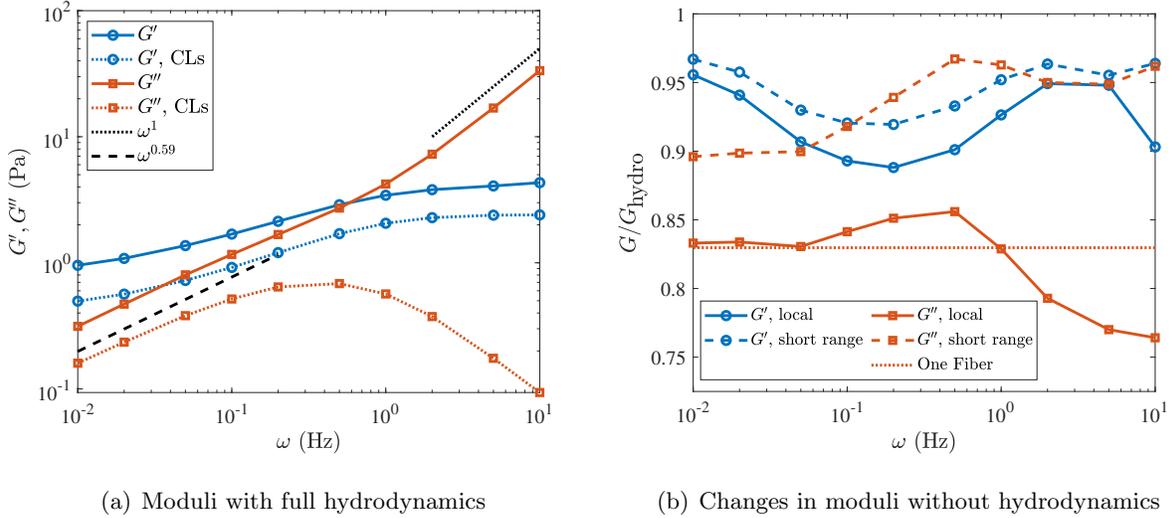
Figure 9: Elastic and viscous moduli for the network of 700 fibers and 8400 CLs. (a) Values of the elastic modulus G' (blue circles) and viscous modulus G'' (orange squares, this excludes the piece due to the background fluid stress $\sigma_{21}^{(\mu)}$ defined in (123)) for a gel of 700 fibers and 8400 CLs. We use fully nonlocal hydrodynamics to compute the dynamics and moduli. We show the contribution of the CL stress (126) to the moduli as dotted lines. (b) Fraction of G' (blue circles) and G'' (orange squares) that can be recovered when the dynamics are computed using local drag (solid lines) and short-ranged hydrodynamics only (i.e., only intra-fiber but no inter-fiber hydrodynamics). We also show the scaling of the stress for a single fiber in a shear flow as a dotted orange line.

a clear transition in both of the moduli for $\omega \approx 0.5 - 1$ Hz which can be understood using the characteristic time scales in the problem. Since the characteristic relaxation time scale $\tau_c \approx 0.5 - 2$ s, the behavior of the moduli can be divided into three regimes: low frequency (background flow time scale $\tau_\omega \gg \tau_c$), medium frequency ($\tau_\omega \approx \tau_c$), and high frequency ($\tau_\omega \ll \tau_c$), where $\tau_\omega = \omega^{-1}$.

1. In the low frequency regime ($\omega < 0.1$ Hz), τ_ω is the longest time scale and the system is in a constant quasi-static state. If the frequency is small enough, the network has the opportunity to relax at every instant, and it therefore behaves more like an elastic solid where the links constrain the network. As in an elastic solid, the elastic modulus is unchanged with frequency and changes very little (less than 10%, as shown in Fig. 9(b)) when nonlocal hydrodynamics is dropped. In this regime, the viscous modulus scales like $G'' = \omega^{0.59}$; the reason for this particular scaling is not obvious to us.

2. In the mid frequency regime ($0.1 \leq \omega \leq 1$), fibers and CLs can deform and relax on the time scale of the background flow and the dynamics involve both an elastic and viscous response. In this regime, $G'' \approx G'$, as shown in Fig. 9(a), and the change in the elastic modulus G' due to both changes in frequency (Fig. 9(a)) and the inclusion of hydrodynamics (Fig. 9(b)) is maximal.

3. In the high frequency regime ($\omega > 1$), the background flow dominates the dynamics, the network is essentially fixed on the time scale of the shear flow, and it oscillates back and forth as a viscous fluid would. Figure 9(a) shows that for $\omega \gg 1/\tau_c$, the viscous modulus scales linearly with ω, as would happen for a pure viscous fluid. In this regime, the viscous modulus decreases by as much as 25% when nonlocal hydrodynamics is not included, with higher frequencies giving larger decreases. The larger the frequency, the farther the network is from its quasi steady-state and the more important dynamics are for determining the viscous modulus.

Generally speaking, the changes in the viscous modulus when hydrodynamics is switched off are attributable to a reduction in stress on each fiber individually. The changes in the viscous modulus at most frequencies can in fact be explained by considering an isolated straight fiber that makes an angle θ with the x axis, $\boldsymbol{\tau}(s) = (\cos\theta, \sin\theta, 0)$. We put a background shear flow on the fiber $\boldsymbol{u}_0(x, y, z) = (y, 0, 0)$ and compute the resulting constraint force density $\boldsymbol{\lambda}$ on the fiber using both local drag and nonlocal hydrodynamics. Averaging over θ, we obtain a mean difference in

the corresponding stress of $\sim 17\%$, which is plotted as a dotted line in Fig. 9(b), and matches the change in the network's viscous modulus when nonlocal hydrodynamics is dropped, except at the largest frequencies. This implies that the change in the network's viscous modulus comes primarily from an increase in the stress on each fiber individually when the intra-fiber, but not necessarily inter-fiber, nonlocal hydrodynamics is included in the mobility calculation. We confirm this in Fig. 10, which shows that including nonlocal hydrodynamics causes the magnitude of the constraint forces $\boldsymbol{\lambda}$ on the fibers in the network to increase without changing their positions substantially.

While Fig. 9(b) shows that there is a change in the viscous modulus of as much as 25% when the dynamics are computed by local drag, it also shows that including short-ranged hydrodynamics only, i.e., including intra-fiber but not inter-fiber hydrodynamics, gives a viscous modulus that is at least 90% of the one computed with inter-fiber hydrodynamics. The changes in the elastic modulus when nonlocal hydrodynamics is dropped are smaller (at most 10%), but not explainable simply by adding only intra-fiber hydrodynamics. Indeed, for $\omega = 0.5$, when the change in the elastic modulus due to hydrodynamics is near maximal, we find that the short-ranged finite part integral (10) explains only one third of the change, with the rest coming from inter-fiber hydrodynamics. Physically, the elastic modulus is related to the interactions of the fibers with CLs, and the network is sufficiently connected that the CLs transmit stress across multiple links in the network, and so long-ranged hydrodynamics plays a role.

Our main findings here can be summarized as follows: there exists a critical time scale τ_c, which is on the order of a second for our parameters. On time scales longer than τ_c, the CLs are constantly in a steady state, as they would be in an elastic solid, and the network is more elastic than viscous $(G' > G'')$. On time scales shorter than τ_c, the network does not have time to respond to (penalize) strain deformations, and the network is more viscous than elastic $(G'' > G')$. On time scales comparable to τ_c, the network is equally viscous and elastic. The more viscous the network behavior and the farther the network is from a quasi-steady state, the more nonlocal hydrodynamics impacts the moduli. Indeed, if a modulus G' or G'' changes substantially with changes in frequency, then we expect dynamics, including whether they are computed with nonlocal or local hydrodynamics, to matter. Interestingly, we find that intra-fiber nonlocal hydrodynamics dominates the contributions of nonlocal hydrodynamics to the viscous, but not the elastic, modulus.
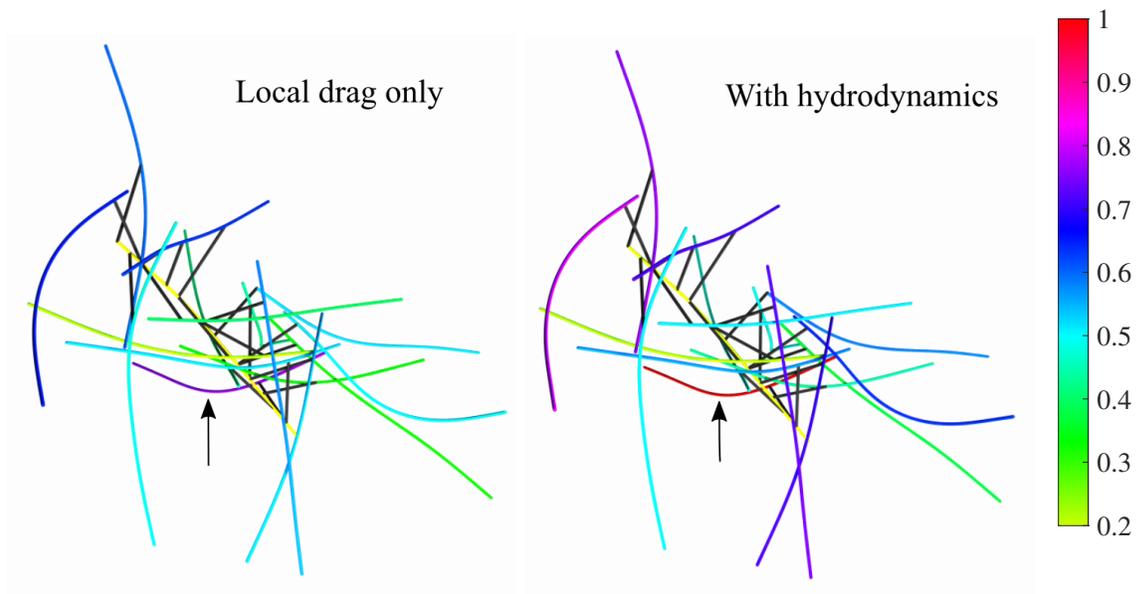
Figure 10: Subset of fibers inside the unit cell, colored by the L^2 norm of the constraint forces $\boldsymbol{\lambda}^{(i)}$ for $\omega = 1$ Hz at $t = 3.5$ s ($g = 0$). Normalization is the maximum L^2 norm of $\boldsymbol{\lambda}^{(i)}$ in the system, which is on the fiber indicated by the arrow. We show the case of local drag at left and nonlocal hydrodynamics at right, noting that the inclusion of nonlocal hydrodynamics increases the norm of the constraint forces (which is the dominant cause of the increase in stress), without changing the fiber positions significantly.

7 Conclusion

In this paper, we have developed a novel method for the simulation of slender filaments, such as actin filaments and microtubules, in the viscous environment of live cells. The key novelty is our reformulation of the continuum fiber centerline evolution in terms of tangent vector rotations, $\partial \boldsymbol{\tau} / \partial t = \boldsymbol{\Omega} \times \boldsymbol{\tau}$, where $\boldsymbol{\Omega}$ is an unknown rotation rate. We introduced constraint forces $\boldsymbol{\lambda}$ and closed the system by requiring that the constraint forces perform no work. This virtual work constraint supplements the evolution equation to give a closed saddle-point system for the unknown rotation rates and constraint forces. Here we used slender body theory to obtain the mobility operator $\boldsymbol{\mathcal{M}}$ for cytoskeletal filaments in the zero Reynolds number regime; however, our continuum formulation could be used with other mobility relationships. In fact, we show in Appendix A that the fiber evolution equation takes the same form regardless of whether we represent the fiber as a cylinder using SBT or a continuum chain of regularized point forces (using the RPY tensor).

We have also shown that our formulation of inextensibility lends itself more naturally to numerical calculations than formulations involving an auxiliary line tension equation [84]. The two main issues with the line tension equation are that it is highly nonlinear in $\boldsymbol{X}$, which can create aliasing issues when using spectral methods, and that it does not naturally preserve inextensibility after discretization, thereby requiring additional penalty forces to do so. We were able to obtain the discrete tangent vector rotation rates $\boldsymbol{\Omega}$ directly from the discretized saddle-point system, rotate the fiber tangent vectors, and obtain the positions by integration, thus preserving discrete inextensibility without penalty parameters.

While we developed an efficient spectral discretization for each fiber's centerline, the numerical method as described here breaks down when the fibers become cylindrical. In that case, which corresponds to $\delta \to 0$ in our local drag regularization (14), the contribution of the finite part integral might exceed that of local drag at the fiber endpoints. This causes the SBT mobility matrix to become ill-conditioned, and the force required to produce even a uniform velocity becomes highly oscillatory near the fiber endpoints. The corresponding lack of smoothness in $\boldsymbol{\Omega}$ and $\boldsymbol{\lambda}$ at the endpoints makes the global Chebyshev method used here inaccurate and unstable. One way to address this might be to couple a spectral discretization, such as the one developed here, for the fiber interior with a lower-order finite difference discretization near the endpoints. This would allow the smooth part of the forcing in the fiber interior to be represented efficiently using a global interpolant, while the oscillatory parts of the forcing would be represented using a more robust

local representation.

In addition to our contributions for a single fiber, we have also contributed key numerical developments in nonlocal slender body hydrodynamics. By observing that slender body theory can be interpreted as asymptotic evaluation of a line integral of regularized singularities, we reformulated the nonlocal hydrodynamics in terms of the Rotne-Prager-Yamakawa mobility tensor. We then used an Ewald splitting scheme for the RPY kernel to evaluate the hydrodynamic interactions on a triply periodic sheared domain in linear time with respect to the number of fibers. For nearby fibers, we supplemented Ewald splitting with a special quadrature scheme which is based on factoring out the near singularity and expanding what remains in a monomial series [1]. This special quadrature method allowed us to develop an algorithm to efficiently compute inter-fiber interactions to 3 digits of accuracy most of the time, while avoiding calls to the special quadrature scheme when direct quadrature is sufficiently accurate.

Despite our improved treatment of near fiber interactions, there are still several issues with our treatment of nearly contacting fibers to be explored in future work. Foremost among these is our use of slender body theory itself. Indeed, SBT is designed to *avoid* the length scale ϵL, which is the same length scale on which the fibers contact each other. While we made some adjustments to the nonlocal hydrodynamics that make it less likely for fibers to cross, these adjustments require too small of a time step to resolve the near-contacts. Nearly contacting fibers also reduce the effectiveness of our GMRES preconditioner for more concentrated suspensions, which is based on the assumption that the dynamics of a single filament are dominated by local drag. Our adjustments to nonlocal SBT also did not account for lubrication and friction (contact) forces or steric repulsion between nearby fibers. These forces, which might be a function of the local orientation, twist, and microscopic shape of the filaments, play a role on length scales ϵL and cannot be resolved by slenderness approximations. Our work has therefore left unanswered what contribution these localized forces make to the macroscopic rheology of a cross-linked fiber suspension. A possible extension to this work is to use a collision tracking algorithm [94, 92] to preserve larger time step sizes and prevent fibers from crossing, thereby enabling future studies to determine the role of lubrication, sterics, and friction.

Since the nonlocal hydrodynamics is the most expensive to compute, we designed a temporal integrator that minimizes the number of nonlocal evaluations. For dilute suspensions, we split the velocity into a local and nonlocal part. The local part, which is the leading order term in ϵ, was treated implicitly, while the sub-leading nonlocal part was treated explicitly. This scheme is

stable for any value of the fiber stiffness κ, as long as the local dynamics dominate the nonlocal hydrodynamics. For semi-dilute and semi-concentrated fiber suspensions, our strategy was to use GMRES to solve for the second-order perturbations in the kinematic coefficients $\boldsymbol{\alpha}$ and constraint forces $\boldsymbol{\lambda}$ that give unconditional stability. Since the perturbations are needed for stability and not accuracy, we simply run GMRES for a fixed number of iterations until we get stability. Our tests showed that at most five total evaluations of the nonlocal hydrodynamics are needed per time step to ensure stability, even for semi-concentrated fiber suspensions.

Our preliminary results on cross-linked actin networks in Section 6 showed that the behavior of the network revolves around a timescale τ_c on which the network relaxes to a dynamic steady state. On timescales longer than τ_c, the network is in a quasi-static state and is more elastic in nature, while on timescales shorter than τ_c, the elastic part of the network has no time to respond to the deformations exerted on it by a background flow, and the network as a whole shows viscous behavior. On timescales shorter than τ_c, including nonlocal hydrodynamics in the mobility changes the viscous modulus by as much as 25% in the system we considered. However, the increases in the viscous modulus come primarily from an increase in stress on each filament separately rather than truly long-ranged hydrodynamic interactions (i.e., the increases due to hydrodynamics are reproducible when each fiber only interacts with itself through the fluid). This behavior differs from that of the elastic modulus, where the more modest increase of 10% due to hydrodynamic interactions is only present with long-ranged hydrodynamics. While these numbers are within the error bounds of most rheology experiments, without our numerical method it would not have been possible to determine them in the first place, since we were able to perform reference simulations and evaluate the errors made by various approximations such as dropping inter-fiber or even intra-fiber nonlocal hydrodynamics (as is often done for simplicity). Omitting long-ranged inter-fiber hydrodynamic interactions, when sufficiently accurate, can speed up our calculations considerably since the more-expensive parts of our algorithm such as Ewald summation and nearly singular quadratures are no longer part of the mobility calculation[7] (this will allow for rapid scanning of parameter regimes in future work). At the same time, in a number of examples in biology, especially when activity is included, the flows generated by individual fibers add constructively to create large-scale macroscopic flows [60, 25]. In these systems, far-field hydrodynamics is necessary to capture large-scale fiber-generated flows, and the numerical method we developed here is a necessary tool for such simulations.

---

[7]Our temporal integrator can also be simplified in this case and is stable without any additional GMRES iterations.

Our findings for cross-linked actin networks are supported by a number of experimental studies. For example, Gardel et al. also observed G' and G'' to be on the same order of magnitude for an actin gel of similar density [24], and several studies have demonstrated a weak dependence of the elastic modulus G' on frequency [24, 35, 37]. Janmey et al. found an elastic modulus $G' \approx 1$ Pa for an actin gel with filaments of mean length $L \approx 2$ μm. They also obtained linear scaling of G'' with ω at high frequencies and sublinear scaling at low frequencies, with a transition occurring at a frequency near 1 rad/s [35]. Our agreement with only some of the existing experimental results is natural since we used only one set of parameters, while in vivo or in vitro experimental parameters can vary based on the system. For example, the CL α-actinin is about an order of magnitude shorter (≈ 40 nm) [70] than filamin ($150 - 200$ nm) [90], and most cross-linking proteins have been estimated to be an order of magnitude stiffer [14] than the ones we use here. Another parameter mismatch manifests itself in the viscous modulus, for which we obtained larger values than those reported experimentally [24, 35, 37] because our underlying fluid viscosity μ was larger than estimated for living cells [51]. While improvements and extensions to our numerical scheme are required to reach the entire span of biological parameters, the platform we have developed here can still be used to test the assumptions behind some of the prior physical theories [24, 35, 46].

Our analysis here was also specific to a fixed τ_c, since the CL stiffness K_c and fiber bending constant κ were fixed. This begs the question: how does τ_c change with changes in κ, K_c, and fluid viscosity μ? How do the intrinsic timescales in the problem change when the CLs are dynamically binding and unbinding to the fibers? To fully explore these questions, we need to design an efficient temporal integrator for the CL forces. For stiff CLs, explicit treatment leads to a reduction in the stable time step size, so that $\mathcal{O}(10^5)$ time steps might be necessary to measure stress for frequencies as small as 0.01 s^{-1}. To alleviate this restriction, we will develop a (semi-)implicit temporal integrator for cross-linked networks in future work. We will also investigate the utility of our spring model for shorter, stiffer CLs, which might be better modeled as actual fibers or as rigid connections between fibers.

In this work, we used Euler-Bernouilli beam theory to obtain the elastic force on a fiber due to its curvature. We neglect twist elasticity on the assumption that those deformations are in equilibrium on our timescales of interest [69]. While this assumption is reasonable for a system of fibers in shear flow, it precludes modeling flagellar beating [48] and chains twisted by external forces such as magnetic fields [79]. An extension of this work is to account for twist by using the Kirchhoff rod model instead of the Euler beam. The Kirchhoff rod model has previously been

used in combination with the immersed boundary method [47], RPY tensor [75], and method of regularized Stokeslets [66, 34] to model a bent twisted fiber interacting with a fluid, with the methods of [34, 75] even preserving discrete inextensibility. As we've discussed at length, these regularization-based models become impractical as the fiber becomes slender since $1/\epsilon$ regularized points are required to properly resolve the fiber thickness. Slender body theory is once again a natural choice, but it is unclear how to account for rotation and torque on the fiber centerline. The most rigorous (and difficult) solution is to use an SBT that accounts for twist, for example the $\mathcal{O}(\epsilon^2)$ slender body fluid velocity of Johnson [36] or the SBT of Keller and Rubinow [38, Section 10]. Another option is to take a continuum limit of the RPY tensor for translation-rotation and rotation-rotation coupling [87, 75], as we have done in Appendix A for translation-translation, and use these continuum limits in the grand mobility matrix. A still better approach would be to bridge the gap between the RPY tensor and SBT by deriving an RPY-type tensor for rings which gives the average linear and angular velocity on a ring (cross section of a fiber) due to the force/torque uniformly spread over another ring (cross section).

Because the persistence length of actin is $\mathcal{O}(10)$ μm and actin filaments in vivo are hundreds of nanometers of length, we have neglected thermal fluctuations in this work. For actin networks, some studies [35, 24] have estimated large elastic moduli even in the absence of CLs, which indicates entropic effects. One of our goals for future work is to extend the method here to account for thermal fluctuations of the filaments. There are many challenges in doing this. Most notably, the loss of smoothness of the fiber centerline function $\boldsymbol{X}(s)$ when thermal noise is present ($\boldsymbol{\tau}(s)$ has the same Hölder continuity as Brownian motion) makes it difficult to even write a well-posed stochastic equation of motion, let alone solve it accurately using a spectral method suited to smooth $\boldsymbol{X}(s)$, as we assumed here.

While there are many improvements awaiting attention, our work has provided solutions to several problems that have previously plagued numerical methods for inextensible slender fibers. From a general framework to view the dynamics of inextensible fibers to specific numerical methods for singular integrals, we have developed a platform for the efficient simulation of thousands of filaments that can be used to gain new insights into processes from sedimentation to cell division and motility.

Acknowledgments

A   Relationship between regularized singularity methods and SBT

In this appendix, we compare our SBT-based fiber representation with a line of regularized point-like forces. Previous studies have used the method of regularized Stokeslets [34] or RPY tensor [75] to represent exactly inextensible fibers. An unresolved question is how regularized singularity methods relate to an SBT-based approach for a single fiber. Bringley and Peskin [12] partially answered this by numerically comparing results from a free space immersed boundary method to SBT. Cortez and Nicholas [18] gave a more complete answer by asymptotically evaluating the velocity on a fiber made of regularized Stokeslets and doublets. They showed that the resulting velocity is a regularized form of SBT with an arbitrary constant regularization parameter. In this appendix, we extend these prior results on regularized singularity methods by considering the fiber to be a line of singularities regularized with the RPY tensor (5). We show that there is a unique choice of sphere radius b that gives, to order ϵ, the same formula as SBT for the centerline velocity. This is the sphere radius that we use in our inter-fiber mobility (15) to give a consistent formulation.

Consider a line of regularized point forces, where each of the point forces is regularized over the surface of a sphere of radius b, and the centerline velocity at a point on the fiber is computed by averaging the fluid velocity over the same sphere of radius b. This means that force can be related

to velocity via the RPY mobility tensor. The velocity at arclength coordinate s on the fiber is then obtained by integrating the RPY kernel over the fiber length,

$$8\pi\mu\boldsymbol{U}(s) = \int_{R>2b} \boldsymbol{S}_D\left(\boldsymbol{X}(s), \boldsymbol{X}(s'), 2b^2/3\right)\boldsymbol{f}(s')\,ds' \tag{A.1}$$

$$+ \int_{R\leq 2b}\left(\left(\left(\frac{4}{3b} - \frac{3R(s')}{8b^2}\right)\boldsymbol{I} + \frac{1}{8b^2 R(s')}(\boldsymbol{R}\boldsymbol{R})(s')\right)\boldsymbol{f}(s')\,ds'.$$

Here $\boldsymbol{R}(s') = \boldsymbol{X}(s') - \boldsymbol{X}(s)$ and $R = \|\boldsymbol{R}\|$. The separation of the integrals captures the change in the RPY tensor (5) when $R < 2b$. Because the RPY tensor is nonsingular for $R = 0$, it can be evaluated at any point on the centerline. This is in contrast to the asymptotics for SBT, which are based on evaluating the Stokeslet/doublet kernel on the fiber surface (i.e., ϵL away from the centerline), and then assigning this result to be the velocity of the centerline [26]. Although the kernel (A.1) is nonsingular, it is still nearly singular for $s \approx s'$, and in the limit $b \ll L$ it is more efficient to evaluate (A.1) asymptotically. In this appendix we show that this results in an SBT-type formulation with a local drag term and finite part integral. The local drag terms can be matched with a specific choice of b, and the remainder is a finite part integral that is equivalent to that of SBT to $\mathcal{O}(\epsilon)$.

Our strategy is standard matched asymptotics and similar to the approach of Gotz [26] for SBT. We compute an outer expansion to the integral (A.1) by considering the region where $|s - s'|$ is $\mathcal{O}(1)$. We then construct an inner expansion in the region where $|s - s'|$ is $\mathcal{O}(b)$. This inner solution must be constructed in two parts for $|s - s'| > 2b$ and $|s - s'| \leq 2b$. We then add the inner and outer solutions together and subtract the common part to obtain the final solution.

A.1 Outer expansion

In the outer expansion, we consider the part of the integral (A.1) where $|s - s'|$ is $\mathcal{O}(1)$. In this case, the doublet term in $\boldsymbol{S}_D$ is insignificant and we obtain the outer velocity by integrating the Stokeslet over the fiber centerline,

$$8\pi\mu\boldsymbol{U}^{(\text{outer})}(s) = \int_{R>2b} \boldsymbol{S}\left(\boldsymbol{X}(s), \boldsymbol{X}(s')\right)\boldsymbol{f}(s')\,ds'. \tag{A.2}$$

The part of the kernel (A.1) for $R \leq 2b$ makes no contribution to the outer expansion since $|s - s'|$ is $\mathcal{O}(b)$ there.

A.2 Inner expansion

In the inner expansion, we consider the part of the integral (A.1) where $|s - s'|$ is $\mathcal{O}(b)$. In this case, we follow [26] and introduce the rescaled variable

$$\xi = \frac{s' - s}{b}, \tag{A.3}$$

so that ξ is $\mathcal{O}(1)$. As in [26], we will assume that the region $[-2, 2]$ is contained in the domain of ξ, thereby ignoring the case when s is $\mathcal{O}(b)$ away from the fiber endpoints. While it is feasible to directly evaluate the RPY integral (A.1) at the fiber endpoints, our goal here is to show equivalence with SBT, which is only valid away from the fiber endpoints.

We will need the following asymptotics around $\boldsymbol{X}(s)$,

$$\boldsymbol{R} = \boldsymbol{X}\left(s'\right) - \boldsymbol{X}(s) = \xi b \boldsymbol{\tau}(s) + \mathcal{O}\left(b^2\right), \quad \boldsymbol{R}\boldsymbol{R} = \xi^2 b^2 \boldsymbol{\tau}(s)\boldsymbol{\tau}(s) + \mathcal{O}\left(b^3\right), \tag{A.4}$$

$$R^2 = \boldsymbol{R} \cdot \boldsymbol{R} = \xi^2 b^2 + \mathcal{O}\left(b^3\right), \quad R = |\xi| b + \mathcal{O}\left(b^2\right), \tag{A.5}$$

$$R^{-1} = \frac{1}{|\xi| b} + \mathcal{O}(1), \quad R^{-3} = \frac{1}{|\xi|^3 b^3} + \mathcal{O}\left(b^{-2}\right), \quad R^{-5} = \frac{1}{|\xi|^5 b^5} + \mathcal{O}\left(b^{-4}\right). \tag{A.6}$$

$$\boldsymbol{f}\left(s'\right) = \boldsymbol{f}(s) + \mathcal{O}(b) \tag{A.7}$$

We begin with the part of the integral (A.1) that uses the kernel $\boldsymbol{S}_D$ in the region $R > 2b$. For this we will need the expansion of the Stokeslet and doublet,

$$\boldsymbol{S}\left(\boldsymbol{X}(s), \boldsymbol{X}\left(s'\right)\right) = \frac{\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|\xi| b} + \mathcal{O}(1) \tag{A.8}$$

$$\boldsymbol{D}\left(\boldsymbol{X}(s), \boldsymbol{X}\left(s'\right)\right) = \frac{\boldsymbol{I} - 3\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|\xi|^3 b^3} + \mathcal{O}\left(b^{-2}\right). \tag{A.9}$$

We now integrate the Stokeslet along the centerline region

$$\int_{R>2b} \boldsymbol{S}\left(\boldsymbol{X}(s), \boldsymbol{X}(s')\right) \boldsymbol{f}\left(s'\right) ds' = \int_{|\xi|>2} \left(\frac{\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|\xi| b}\right) \boldsymbol{f}(s) b \, d\xi + \mathcal{O}(b) \tag{A.10}$$

$$= \left(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right) \boldsymbol{f}(s) \left[\int_{-s/b}^{-2} -\frac{1}{\xi} \, d\xi + \int_{2}^{(L-s)/b} \frac{1}{\xi} \, d\xi\right] + \mathcal{O}(b) \tag{A.11}$$

$$= \log\left(\frac{(L-s)s}{4b^2}\right) \left(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right) \boldsymbol{f}(s) + \mathcal{O}(b). \tag{A.12}$$

Likewise for the doublet, we have

$$\frac{2b^2}{3} \int_{R>2b} \boldsymbol{D}\left(\boldsymbol{X}(s), \boldsymbol{X}(s')\right) \, ds' = \frac{2b^2}{3} \int_{|\xi|>2} \left(\frac{\boldsymbol{I} - 3\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|\xi|^3 b^3}\right) \boldsymbol{f}(s) b \, d\xi + \mathcal{O}(b) \tag{A.13}$$

$$= \frac{2\left(\boldsymbol{I} - 3\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right) \boldsymbol{f}(s)}{3} \left[\int_{-s/b}^{-2} -\frac{1}{\xi^3} \, d\xi + \int_2^{(L-s)/b} \frac{1}{\xi^3} \, d\xi\right] + \mathcal{O}(b) \tag{A.14}$$

$$= \frac{2}{12} \left(\boldsymbol{I} - 3\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right) \boldsymbol{f}(s) + \mathcal{O}(b). \tag{A.15}$$

Combining these results, we have, to $\mathcal{O}(b)$,

$$\int_{R>2b} \boldsymbol{S}_D\left(\boldsymbol{X}(s), \boldsymbol{X}\left(s'\right), 2b^2/3\right) \, ds' = \tag{A.16}$$

$$\left(\log\left(\frac{(L-s)s}{4b^2}\right) \left(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right) + \frac{1}{6}\left(\boldsymbol{I} - 3\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right)\right) \boldsymbol{f}(s).$$

It still remains to include in the inner expansion the term for $R \leq 2b$. For this we have the two terms

$$\int_{R<2b} \left(\frac{4}{3b} - \frac{3R\left(s'\right)}{8b^2}\right) \boldsymbol{f}\left(s'\right) \, ds' = \boldsymbol{f}(s) \int_{-2}^2 \left(\frac{4}{3} - \frac{3|\xi|}{8}\right) \, d\xi + \mathcal{O}(b) = \frac{23}{6} \boldsymbol{f}(s) + \mathcal{O}(b), \tag{A.17}$$

$$\int_{R<2b} \frac{1}{8b^2 R\left(s'\right)} \left(\boldsymbol{R}\boldsymbol{R}\right)\left(s'\right) \boldsymbol{f}\left(s'\right) \, ds' = \frac{1}{8} \int_{-2}^2 \boldsymbol{\tau}(s)\boldsymbol{\tau}(s) \boldsymbol{f}\left(s'\right) |\xi| \, d\xi + \mathcal{O}(b) = \frac{1}{2}\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\boldsymbol{f}(s) + \mathcal{O}(b) \tag{A.18}$$

where we have used the fact that $\xi \in [-2, 2]$ is on the fiber (s is away from the endpoints). We therefore have, to $\mathcal{O}(b)$,

$$\int_{R\leq 2b} \left(\left(\frac{4}{3b} - \frac{3R\left(s'\right)}{8b^2}\right) \boldsymbol{I} + \frac{1}{8b^2 R\left(s'\right)} \left(\boldsymbol{R}\boldsymbol{R}\right)\left(s'\right)\right) \, ds' = \left(\frac{23}{6}\boldsymbol{I} + \frac{1}{2}\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right) \boldsymbol{f}(s). \tag{A.19}$$

The inner expansion is therefore, adding the terms (A.16) and (A.19),

$$8\pi\mu\boldsymbol{U}^{(\text{inner})}(s) = \left(\log\left(\frac{(L-s)s}{4b^2}\right) \left(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right) + 4\boldsymbol{I}\right) \boldsymbol{f}(s). \tag{A.20}$$

By adding and subtracting $\log(16)\left(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right)$, we obtain the same leading order coefficient as SBT,

$$8\pi\mu\boldsymbol{U}^{(\text{inner})}(s) = \left(\log\left(\frac{4(L-s)s}{b^2}\right) \left(\boldsymbol{I} + \boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right)\right. \tag{A.21}$$

$$\left. + (4 - \log 16)\boldsymbol{I} - (\log 16)\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right)\boldsymbol{f}(s).$$

A.3 Common part

The common part is the outer velocity written in terms of the inner variables. That is, to $\mathcal{O}(b)$,

$$8\pi\mu\boldsymbol{U}^{(\text{common})}(s) = \int_{R>2b}\left(\frac{\boldsymbol{I}+\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|s-s'|}\right)\boldsymbol{f}(s)\,ds'. \tag{A.22}$$

A.4 Matched asymptotic expansion

The total velocity is the sum of the inner and outer expansions, with the common part subtracted,

$$\boldsymbol{U}(s) = \boldsymbol{U}^{(\text{inner})}(s) + \boldsymbol{U}^{(\text{outer})}(s) - \boldsymbol{U}^{(\text{common})}(s). \tag{A.23}$$

This can be written as

$$8\pi\mu\boldsymbol{U}(s) = \left(\log\left(\frac{4(L-s)s}{b^2}\right)(\boldsymbol{I}+\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)) + a_I\boldsymbol{I} + a_\tau\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right)\boldsymbol{f}(s) \tag{A.24}$$

$$+ \int_{R>2b}\left(\boldsymbol{S}\left(\boldsymbol{X}(s),\boldsymbol{X}(s')\right)\boldsymbol{f}\left(s'\right) - \left(\frac{\boldsymbol{I}+\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|s-s'|}\right)\boldsymbol{f}(s)\right)ds',$$

$$\text{where}\quad a_I = 4 - \log 16 \quad\text{and}\quad a_\tau = -\log 16. \tag{A.25}$$

We can now establish equivalence with SBT by observing that the integrand in (A.24) is $\mathcal{O}(b)$ when $R < 2b$, and so we can add that part of the integral back into the velocity without changing the asymptotic accuracy of the velocity (A.24). This gives a velocity of the exact same form as SBT,

$$8\pi\mu\boldsymbol{U}(s) = \left(\log\left(\frac{4(L-s)s}{b^2}\right)(\boldsymbol{I}+\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)) + a_I\boldsymbol{I} + a_\tau\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)\right)\boldsymbol{f}(s) \tag{A.26}$$

$$+ \int_0^L\left(\boldsymbol{S}\left(\boldsymbol{X}(s),\boldsymbol{X}(s')\right)\boldsymbol{f}\left(s'\right) - \left(\frac{\boldsymbol{I}+\boldsymbol{\tau}(s)\boldsymbol{\tau}(s)}{|s-s'|}\right)\boldsymbol{f}(s)\right)ds'.$$

The velocity expression (A.26) is the same as the SBT velocity (8) when $a_I = 1$ and $a_\tau = -3$. This equivalence is accomplished by the specific choice[8] of b

$$b = \frac{e^{3/2}}{4}\epsilon L, \tag{A.27}$$

which gives the RPY doublet coefficient

$$\frac{2b^2}{3} = \frac{e^3}{24}\epsilon L, \tag{A.28}$$

that we use in Section 2.2 for inter-fiber interactions. Because we used only the leading order terms in the asymptotics, we are guaranteed that (A.26) approximates (A.1) to $\mathcal{O}(b)$, although we observe $\mathcal{O}(b^2)$ accuracy a distance $\mathcal{O}(1)$ from the endpoints.

---

[8]Observe that we have two equations (a_I and a_τ) for one variable b, so the existence of a solution is surprising.
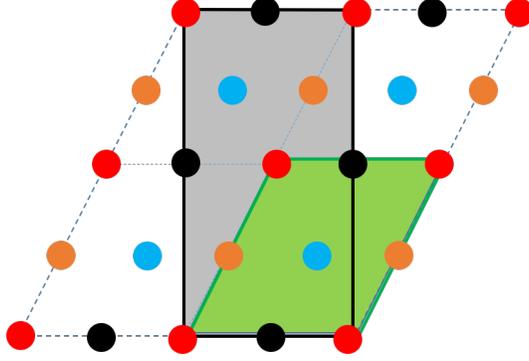
Figure 11: The lattice for the sheared unit cell test. The points are positioned on a lattice with a 1-particle (primitive) unit cell (shaded green) at $(0,0,0)$ (red), $(1,0,0)$ (black), $(0.5,1,0)$ (orange), and $(1.5,1,0)$ (blue). The lattice can also be viewed as periodic on the larger 2-particle rectangular unit cell shaded in gray. Colors indicate the magnitude of the force placed on each set of points: $+1$ in all three directions on the red points, -1 in each direction on the black points, $+2$ on the orange points, and -2 on the blue points.

B Verification for sheared unit cell

To test our implementation of sheared periodic boundary conditions, we consider a packing of points that is hexagonal in the xy plane. As shown in Fig. 11, the points are positioned on a (green) periodic slanted cell at $(0,0,0)$ (red blob), $(1,0,0)$ (black), $(0.5,1,0)$ (orange), and $(1.5,1,0)$ (sky blue). To form a periodic hexagonal packing in the xy plane, we set $g=0.5$ with periodic domain length $L_x = L_y = L_z = 2$. Using a coloring scheme (see Fig. 11), it is easy to see that this arrangement is equivalent to the same set of points on a (gray) rectangular unit cell, with additional points at $(1,2,0)$, $(0,2,0)$, $(1.5,3,0)$, $(0.5,3,0)$, with the ordering of forces in the second set of points being the same as the first and periodic length $L_y = 4$. We place a force of strength $+1$ in each direction (including z) on the first (red) pair of points, -1 in each direction on the second (black) pair, $+2$ on the third (orange), and -2 on the fourth (sky blue). Note that the z direction is also periodic in all cases with length $L_z = 2$, so that we are actually considering a set of stacked copies of Fig. 11.

We solve for the RPY velocities induced by the forces at each point using the Ewald splitting technique described in Section 4.3. We set $\xi = 5$, sphere radius $b = 10^{-2}$, and fluid viscosity $\mu = 3$. The maximum relative 2-norm error in the velocity of the four points is less than 10^{-5} for all values of the NUFFT tolerance less than 10^{-2}, with decay to 10^{-11} when the tolerance is 10^{-8}. We conclude that our modified Ewald splitting scheme of Section 4.3 properly treats the strain in the periodic coordinate system.

C Near fiber accuracy

In this appendix, we test the accuracy of Algorithm 1 for computing the slender body interaction integrals (15). We generate 100 smooth inextensible fibers by initializing an unnormalized tangent vector that is an exact Chebyshev series with 15 exponentially decaying terms. More precisely, the kth coefficient of the series is a Gaussian random variable with mean 0 and standard deviation $e^{-10k/N}$, where $k = 0, \ldots 15$. We then normalize this tangent vector to obtain $\boldsymbol{\tau}(s)$ and integrate to obtain the fiber positions $\boldsymbol{X}(s)$. To make sure the resulting fiber is smooth after tangent vector normalization, we compute the Chebyshev series of the fiber position $\boldsymbol{X}(s)$. Denoting the coefficients of the position Chebyshev series by $\hat{a}_k$, we only accept fibers with Chebyshev series coefficients $|\hat{a}_k| \leq e^{-0.61k}$ for $k = 2, \ldots 15$ (the constant and linear modes play no role in the fiber smoothness). This means that the last coefficient $\hat{a}_{15}$ has value at most 10^{-4}.

We consider fibers with $L = 2$ and $\epsilon = 10^{-3}$ in a fluid of viscosity $\mu = 1/8\pi$. Our goal is to evaluate the velocity due to a fiber $\boldsymbol{X}(s)$ at a target $\boldsymbol{x}$,

$$\boldsymbol{v}(\boldsymbol{x}) = \int_0^L \boldsymbol{S}_D\left(\boldsymbol{x}, \boldsymbol{X}(s); \frac{e^3}{24}(\epsilon L)^2\right) \boldsymbol{f}(s). \tag{C.1}$$

We choose $\boldsymbol{f}(s) = \boldsymbol{\tau}(s)$, so that the force density is sufficiently smooth.

To measure the accuracy of Algorithm 1, we place 100 targets a distance d away in a random normal direction from each fiber's centerline. To get a reference answer, we compute the integral (C.1) directly by upsampling the fiber to 6000 type 1 Chebyshev points. We then compute the integral using Algorithm 1. We show the maximum relative error, $\boldsymbol{E}_i/\|\boldsymbol{v}(\boldsymbol{x})\|_\infty$, where the maximum is over the direction $i = 1, 2, 3$ and $\boldsymbol{E}$ is the absolute difference between the approximate and reference values of the velocity (C.1).

We separate our results into short distances, $2\epsilon L < d < 10\epsilon L$, and long distances, $0.01L < d < 0.2L$ (there is overlap between the two regions since $10\epsilon L = 0.01L = 0.02$ with our parameters). Fig. 12(a) shows the errors at short distances. We see that we obtain many more digits than necessary in most cases. There are, however, a few cases where we obtain 3 digits. Since d will rarely be $\mathcal{O}(\epsilon L)$, it is acceptable to expend extra computational effort to guarantee accuracy. In Fig. 12(b), we show the errors for long distances. In particular, we see that we obtain 4-5 digits most of the time, and that $\approx 5\%$ of the time we obtain exactly 3 digits of accuracy.
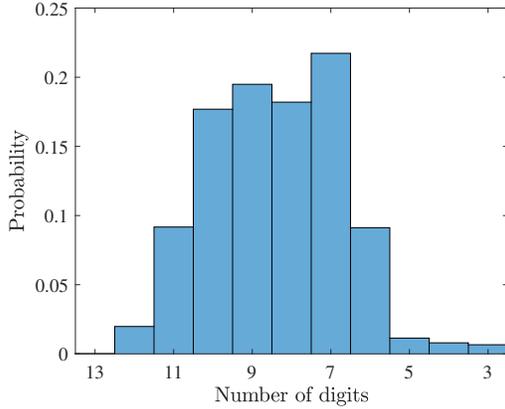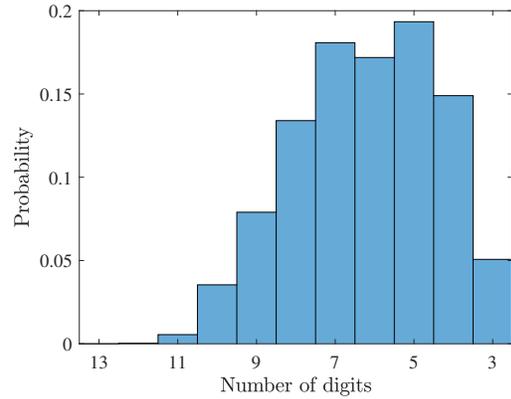
(a) Short distances   (b) Long distances

Figure 12: Errors in Algorithm 1 for near singular quadratures. We show a histogram of the number of digits obtained in the velocity $\boldsymbol{v}(\boldsymbol{x})$, given in (C.1), for 100 different fibers and 100 targets per fiber. Here d is the distance from the target point to the fiber centerline and we show histograms of the number of digits obtained in the integral $\boldsymbol{v}(\boldsymbol{x})$. The number of digits is computed as $-\log_{10}\left(\max_i \boldsymbol{E}_i / \|\boldsymbol{v}(\boldsymbol{x})\|_\infty\right)$, where the maximum is over the direction $i = 1, 2, 3$ and $\boldsymbol{E}$ is the absolute difference between the approximate and reference values of the velocity (C.1). (a) Errors from short distances $2 \leq d/(\epsilon L) \leq 10$. We see that we are over-working in most cases, since most of the time we obtain many more than 3 digits of accuracy, but there are some cases when we only obtain 3 digits. (b) Long distances $0.04 < d/L < 0.20$, where we obtain 4-7 digits most of the time.

References

[1] Ludvig af Klinteberg and Alex H Barnett. Accurate quadrature of nearly singular line integrals in two and three dimensions by singularity swapping. *BIT Numerical Mathematics*, pages 1–36, 2020.

[2] Ludvig Af Klinteberg and Anna-Karin Tornberg. Fast ewald summation for stokesian particle suspensions. *International Journal for Numerical Methods in Fluids*, 76(10):669–698, 2014.

[3] Wylie W Ahmed and Timo Betz. Dynamic cross-links tune the solid–fluid behavior of living cells. *Proceedings of the National Academy of Sciences*, 112(21):6527–6528, 2015.

[4] Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. *Molecular biology of the cell*. Garland Science, 2002.

[5] Jared L Aurentz and Lloyd N Trefethen. Block operators and spectral discretizations. *SIAM Review*, 59(2):423–446, 2017.

[6] Alexander H Barnett, Jeremy Magland, and Ludvig af Klinteberg. A parallel nonuniform fast fourier transform library based on an "exponential of semicircle" kernel. *SIAM J. Sci. Comput.*, 41(5):C479–C504, 2019.

[7] GK Batchelor. Slender-body theory for particles of arbitrary cross-section in stokes flow. *Journal of Fluid Mechanics*, 44(3):419–440, 1970.

[8] GK Batchelor. The stress system in a suspension of force-free particles. *Journal of fluid mechanics*, 41(3):545–570, 1970.

[9] Howard C Berg and Robert A Anderson. Bacteria swim by rotating their flagellar filaments. *Nature*, 245(5425):380, 1973.

[10] Miguel Angel Bibbó. *Rheology of semiconcentrated fiber suspensions*. PhD thesis, Massachusetts Institute of Technology, 1987.

[11] Christopher Brennen and Howard Winet. Fluid mechanics of propulsion by cilia and flagella. *Annual Review of Fluid Mechanics*, 9(1):339–398, 1977.

[12] Thomas T Bringley and Charles S Peskin. Validation of a simple method for representing spheres and slender bodies in an immersed boundary method for stokes flow on an unbounded domain. *Journal of Computational Physics*, 227(11):5397–5425, 2008.

[13] AT Chwang and Th Y Wu. A note on the helical movement of micro-organisms. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 178(1052):327–346, 1971.

[14] Mireille MAE Claessens, Mark Bathe, Erwin Frey, and Andreas R Bausch. Actin-binding proteins sensitively mediate f-actin bundle stiffness. *Nature materials*, 5(9):748–753, 2006.

[15] Ricardo Cortez. The method of regularized stokeslets. *SIAM Journal on Scientific Computing*, 23(4):1204–1225, 2001.

[16] Ricardo Cortez. Regularized stokeslet segments. *Journal of Computational Physics*, 375:783–796, 2018.

[17] Ricardo Cortez, Lisa Fauci, and Alexei Medovikov. The method of regularized stokeslets in three dimensions: analysis, validation, and application to helical swimming. *Physics of Fluids*, 17(3):031504, 2005.

[18] Ricardo Cortez and Michael Nicholas. Slender body theory for stokes flows with regularized forces. *Communications in Applied Mathematics and Computational Science*, 7(1):33–62, 2012.

[19] Debasish Das and Eric Lauga. Computing the motor torque of escherichia coli. *Soft matter*, 14(29):5955–5967, 2018.

[20] Tobin A Driscoll and Nicholas Hale. Rectangular spectral collocation. *IMA Journal of Numerical Analysis*, 36(1):108–132, 2015.

[21] Andrew M Fiore, Florencio Balboa Usabiaga, Aleksandar Donev, and James W Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations. *The Journal of chemical physics*, 146(12):124116, 2017.

[22] Andrew M Fiore and James W Swan. Rapid sampling of stochastic displacements in brownian dynamics simulations with stresslet constraints. *The Journal of chemical physics*, 148(4):044114, 2018.

[23] Andrew M Fiore and James W Swan. Fast stokesian dynamics. *Journal of Fluid Mechanics*, 878:544–597, 2019.

[24] ML Gardel, Jennifer Hyunjong Shin, FC MacKintosh, L Mahadevan, PA Matsudaira, and DA Weitz. Scaling of f-actin network rheology to probe single filament elasticity and dynamics. *Physical review letters*, 93(18):188102, 2004.

[25] Raymond E Goldstein. Batchelor prize lecture fluid dynamics at the scale of the cell. *Journal of Fluid Mechanics*, 807:1–39, 2016.

[26] Thomas Götz. *Interactions of fibers and flow: asymptotics, theory and numerics*. dissertation. de, 2001.

[27] Leslie Greengard. Spectral integration and two-point boundary value problems. *SIAM Journal on Numerical Analysis*, 28(4):1071–1080, 1991.

[28] Boyce E Griffith, Richard D Hornung, David M McQueen, and Charles S Peskin. An adaptive, formally second order accurate version of the immersed boundary method. *J. Comput. Phys.*, 223(1):10–49, 2007.

[29] W Guan, X Cheng, J Huang, G Huber, Wei Li, JA McCammon, and B Zhang. Rpyfmm: Parallel adaptive fast multipole method for rotne–prager–yamakawa tensor in biomolecular hydrodynamics simulations. *Computer physics communications*, 227:99–108, 2018.

[30] Atticus L Hall-McNair, Thomas D Montenegro-Johnson, Hermes Gadêlha, David J Smith, and Meurig Thomas Gallagher. Efficient implementation of elastohydrodynamics via integral operators. *Physical Review Fluids*, 4(11):113101, 2019.

[31] GJ Hancock. The self-propulsion of microscopic organisms through liquids. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 217(1128):96–121, 1953.

[32] Hidenori Hasimoto. On the periodic fundamental solutions of the stokes equations and their application to viscous flow past a cubic array of spheres. *Journal of Fluid Mechanics*, 5(2):317–328, 1959.

[33] David A Head, Alex J Levine, and FC MacKintosh. Deformation of cross-linked semiflexible polymer networks. *Physical review letters*, 91(10):108102, 2003.

[34] Mehdi Jabbarzadeh and Henry C Fu. A numerical method for inextensible elastic filaments in viscous fluids. *Journal of Computational Physics*, page 109643, 2020.

[35] Paul A Janmey, Soren Hvidt, J Käs, Dietmar Lerche, Anthony Maggs, Erich Sackmann, Manfred Schliwa, and Thomas P Stossel. The mechanical properties of actin gels. elastic modulus and filament motions. *Journal of Biological Chemistry*, 269(51):32503–32513, 1994.

[36] Robert E Johnson. An improved slender-body theory for stokes flow. *Journal of Fluid Mechanics*, 99(2):411–431, 1980.

[37] KE Kasza, CP Broedersz, GH Koenderink, YC Lin, W Messner, EA Millman, F Nakamura, TP Stossel, FC MacKintosh, and DA Weitz. Actin filament length tunes elasticity of flexibly cross-linked actin networks. *Biophysical journal*, 99(4):1091–1100, 2010.

[38] Joseph B Keller and Sol I Rubinow. Slender-body theory for slow viscous flow. *Journal of Fluid Mechanics*, 75(4):705–714, 1976.

[39] Ludvig af Klinteberg, Davoud Saffar Shamshirgar, and Anna-Karin Tornberg. Fast Ewald summation for free-space Stokes potentials. *Research in the Mathematical Sciences*, pages 1–32, December 2016.

[40] Hideki Kobayashi and Ryoichi Yamamoto. Implementation of lees–edwards periodic boundary conditions for direct numerical simulations of particle dispersions under shear flow. *The Journal of chemical physics*, 134(6):064110, 2011.

[41] Lyndon Koens and Eric Lauga. The boundary integral formulation of stokes flows includes slender-body theory. *Journal of Fluid Mechanics*, 850, 2018.

[42] Cornelius Lanczos. *The variational principles of mechanics*. Courier Corporation, 2012.

[43] Joel Langer and David A Singer. Lagrangian aspects of the kirchhoff elastic rod. *SIAM review*, 38(4):605–618, 1996.

[44] Eric Lauga and Thomas R Powers. The hydrodynamics of swimming microorganisms. *Reports on Progress in Physics*, 72(9):096601, 2009.

[45] Zhi Liang, Zydrunas Gimbutas, Leslie Greengard, Jingfang Huang, and Shidong Jiang. A fast multipole method for the rotne–prager–yamakawa tensor and its applications. *Journal of Computational Physics*, 234:133–139, 2013.

[46] O Lieleg, KM Schmoller, Mireille MAE Claessens, and AR Bausch. Cytoskeletal polymer networks: viscoelastic properties are determined by the microscopic interaction potential of cross-links. *Biophysical journal*, 96(11):4725–4732, 2009.

[47] Sookkyung Lim, Anca Ferent, X Sheldon Wang, and Charles S Peskin. Dynamics of a closed rod with twist and bend in fluid. *SIAM Journal on Scientific Computing*, 31(1):273–302, 2008.

[48] Sookkyung Lim and Charles S Peskin. Fluid-mechanical interaction of flexible bacterial flagella by the immersed boundary method. *Physical Review E*, 85(3):036307, 2012.

[49] Dag Lindbo and Anna-Karin Tornberg. Spectrally accurate fast summation for periodic Stokes potentials. *Journal of Computational Physics*, 229(23):8994–9010, 2010.

[50] Sune Lomholt and Martin R Maxey. Force-coupling method for particulate two-phase flow: Stokes flow. *Journal of Computational Physics*, 184(2):381–405, 2003.

[51] Katherine Luby-Phelps. Cytoarchitecture and physical properties of cytoplasm: volume, viscosity, diffusion, intracellular surface area. In *International review of cytology*, volume 192, pages 189–221. Elsevier, 1999.

[52] Rui Ma and Julien Berro. Structural organization and energy storage in crosslinked actin assemblies. *PLoS computational biology*, 14(5):e1006150, 2018.

[53] Michael B Mackaplow and Eric SG Shaqfeh. A numerical study of the rheological properties of suspensions of rigid, non-brownian fibres. *J. Fluid Mech.*, 329:155–186, 1996.

[54] Alexander M Maier, Cornelius Weig, Peter Oswald, Erwin Frey, Peer Fischer, and Tim Liedl. Magnetic propulsion of microswimmers with dna-based flagellar bundles. *Nano letters*, 16(2):906–910, 2016.

[55] Harishankar Manikantan and David Saintillan. Subdiffusive transport of fluctuating elastic filaments in cellular flows. *Physics of Fluids*, 25(7):073603, 2013.

[56] Tătulea-Codrean Maria and Eric Lauga. Geometrical constraints on the tangling of bacterial flagellar filaments. *Scientific Reports (Nature Publisher Group)*, 10(1), 2020.

[57] MR Maxey and BK Patel. Localized force representations for particles sedimenting in stokes flow. *International journal of multiphase flow*, 27(9):1603–1626, 2001.

[58] Ondrej Maxian and Charles S Peskin. An immersed boundary method with subgrid resolution and improved numerical stability applied to slender bodies in stokes flow. *SIAM J. Sci. Comput.*, 42(4):847–868, 2020.

[59] Rudolf K Meyer and Ueli Aebi. Bundling of actin filaments by alpha-actinin depends on its molecular length. *The Journal of cell biology*, 110(6):2013–2024, 1990.

[60] Corey E Monteith, Matthew E Brunner, Inna Djagaeva, Anthony M Bielecki, Joshua M Deutsch, and William M Saxton. A mechanism for cytoplasmic streaming: Kinesin-driven alignment of microtubules and fast fluid flows. *Biophysical journal*, 110(9):2053–2065, 2016.

[61] Yoichiro Mori and Laurel Ohm. Accuracy of slender body theory in approximating force exerted by thin fiber on viscous fluid. *arXiv preprint arXiv:2008.06829*, 2020.

[62] Yoichiro Mori, Laurel Ohm, and Daniel Spirn. Theoretical justification and error analysis for slender body theory. *Communications on Pure and Applied Mathematics*, 73(6):1245–1314, 2020.

[63] Yoichiro Mori, Laurel Ohm, and Daniel Spirn. Theoretical justification and error analysis for slender body theory with free ends. *Archive for Rational Mechanics and Analysis*, 235(3):1905–1978, 2020.

[64] Faith A Morrison et al. *Understanding rheology*. Oxford University Press, USA, 2001.

[65] Ehssan Nazockdast, Abtin Rahimian, Denis Zorin, and Michael Shelley. A fast platform for simulating semi-flexible fiber suspensions applied to cell mechanics. *J. Comput. Phys.*, 329:173–209, 2017.

[66] Sarah D Olson, Sookkyung Lim, and Ricardo Cortez. Modeling the dynamics of an elastic rod with intrinsic curvature and twist using a regularized stokes formulation. *Journal of Computational Physics*, 238:169–187, 2013.

[67] Charles S Peskin. Flow patterns around heart valves: a numerical method. *J. Comput. Phys.*, 10(2):252–271, 1972.

[68] Charles S Peskin. The immersed boundary method. *Acta Numer.*, 11:479–517, 2002.

[69] Thomas R Powers. Dynamics of filaments and membranes in a viscous fluid. *Reviews of Modern Physics*, 82(2):1607, 2010.

[70] Jr Ede A Ribeiro, Nikos Pinotsis, Andrea Ghisleni, Anita Salmazo, Petr V Konarev, Julius Kostan, Björn Sjöblom, Claudia Schreiner, Anton A Polyansky, Eirini A Gkougkoulia, et al. The structure and regulation of human muscle α-actinin. *Cell*, 159(6):1447–1460, 2014.

[71] Olinde Rodrigues. *Des lois géométriques qui régissent les déplacements d'un système solide dans l'espace: et de la variation des cordonnées provenant de ces déplacements considérés indépendamment des causes qui peuvent les produire.* 1840.

[72] Minghao W Rostami and Sarah D Olson. Kernel-independent fast multipole method within the framework of regularized stokeslets. *Journal of Fluids and Structures*, 67:60–84, 2016.

[73] Jens Rotne and Stephen Prager. Variational treatment of hydrodynamic interaction in polymers. *J. Chem. Phys.*, 50(11):4831–4837, 1969.

[74] Youcef Saad and Martin H Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869, 1986.

[75] Simon F Schoeller, Adam K Townsend, Timothy A Westwood, and Eric E Keaveny. Methods for suspensions of passive and active filaments. *arXiv preprint arXiv:1903.12609*, 2019.

[76] Michael J Shelley. The dynamics of microtubule/motor-protein assemblies in biology and physics. *Annual Review of Fluid Mechanics*, 48:487–506, 2016.

[77] Michael J Shelley and Tetsuji Ueda. The nonlocal dynamics of stretching, buckling filaments. *Advances in multi-fluid flows*, pages 415–425, 1996.

[78] Michael J Shelley and Tetsuji Ueda. The stokesian hydrodynamics of flexing, stretching filaments. *Physica D: Nonlinear Phenomena*, 146(1-4):221–245, 2000.

[79] Brennan Sprinkle, Ernest B van der Wee, Yixiang Luo, Michelle Driscoll, and Aleksandar Donev. Active dynamics in dense suspensions of microrollers. *arXiv preprint arXiv:2005.06002*, 2020.

[80] RR Sundararajakumar and Donald L Koch. Structure and properties of sheared fiber suspensions with mechanical contacts. *Journal of Non-Newtonian Fluid Mechanics*, 73(3):205–239, 1997.

[81] Anna-Karin Tornberg. Accurate evaluation of integrals in slender-body formulation for fibers in viscous flow. *to appear.*

[82] Anna-Karin Tornberg. Fft based spectral ewald methods as an alternative to fast multipole methods.

[83] Anna-Karin Tornberg and Katarina Gustavsson. A numerical method for simulations of rigid fiber suspensions. *Journal of Computational Physics*, 215(1):172–196, 2006.

[84] Anna-Karin Tornberg and Michael J Shelley. Simulating the dynamics and interactions of flexible fibers in stokes flows. *Journal of Computational Physics*, 196(1):8–40, 2004.

[85] Lloyd N Trefethen. *Spectral methods in MATLAB*, volume 10. Siam, 2000.

[86] Bernd Wagner, Rainer Tharmann, Ilka Haase, Markus Fischer, and Andreas R Bausch. Cytoskeletal polymer networks: the molecular structure of cross-linkers determines macroscopic properties. *Proceedings of the National Academy of Sciences*, 103(38):13974–13978, 2006.

[87] Eligiusz Wajnryb, Krzysztof A Mizerski, Pawel J Zuk, and Piotr Szymczak. Generalization of the rotne–prager–yamakawa mobility and shear disturbance tensors. *Journal of Fluid Mechanics*, 731, 2013.

[88] Benjamin J Walker and Eamonn A Gaffney. Regularised non-uniform segments and efficient no-slip elastohydrodynamics. *arXiv preprint arXiv:2008.12339*, 2020.

[89] BJ Walker, MP Curtis, K Ishimoto, and EA Gaffney. A regularised slender-body theory of non-uniform filaments. *Journal of Fluid Mechanics*, 899, 2020.

[90] Robert R Weihing. The filamins: properties and functions. *Canadian journal of biochemistry and cell biology*, 63(6):397–413, 1985.

[91] Dean R Wheeler, Norman G Rowley, and Richard L Fuller. Non-equilibrium molecular dynamics simulation of the shear viscosity of liquid methanol: adaptation of the ewald sum to lees-edwards boundary conditions. *Molecular Physics*, 92(1):55–62, 1997.

[92] Wen Yan, Eduardo Corona, Dhairya Malhotra, Shravan Veerapaneni, and Michael Shelley. A scalable computational platform for particulate stokes suspensions. *Journal of Computational Physics*, page 109524, 2020.

[93] Wen Yan and Michael Shelley. Universal image systems for non-periodic and periodic stokes flows above a no-slip wall. *Journal of Computational Physics*, 375:263–270, 2018.

[94] Wen Yan, Huan Zhang, and Michael J Shelley. Computing collision stress in assemblies of active spherocylinders: Applications of a fast and generic geometric method. *The Journal of chemical physics*, 150(6):064109, 2019.

[95] Y-N Young. Hydrodynamic interactions between two semiflexible inextensible filaments in stokes flow. *Physical Review E*, 79(4):046317, 2009.

[96] PJ Zuk, E Wajnryb, KA Mizerski, and P Szymczak. Rotne–prager–yamakawa approximation for different-sized particles in application to macromolecular bead models. *Journal of Fluid Mechanics*, 741, 2014.