Lawrence Berkeley National Laboratory
LBL Publications

Title
Eulerian simulation of complex suspensions and biolocomotion in three dimensions

Permalink
https://escholarship.org/uc/item/5nj5f5s1

Authors
Lin, Yuexia L
Derr, Nicholas J
Rycroft, Chris H

Publication Date
2021-03-31

Peer reviewed

Eulerian simulation of complex suspensions and biolocomotion in three dimensions

Yuexia L. Lin[a], Nicholas J. Derr[a], and Chris H. Rycroft[*a,b]

[a]*John A. Paulson School of Engineering and Applied Sciences, Harvard University, 29 Oxford Street, Cambridge, MA 02138*

[b]*Mathematics Group, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, CA 94720*

Abstract

We present a numerical method specifically designed for simulating three-dimensional fluid–structure interaction (FSI) problems based on the reference map technique (RMT). The RMT is a fully Eulerian FSI numerical method that allows fluids and large-deformation elastic solids to be represented on a single fixed computational grid. This eliminates the need for meshing complex geometries typical in other FSI approaches, and greatly simplifies the coupling between fluid and solids. We develop the first three-dimensional implementation of the RMT, parallelized using the distributed memory paradigm, to simulate incompressible FSI with neo-Hookean solids. As part of our new method, we develop a new field extrapolation scheme that works efficiently in parallel. Through representative examples, we demonstrate the method's accuracy and convergence, as well as its suitability in investigating many-body and active systems. The examples include settling of a mixture of heavy and buoyant soft ellipsoids, lid-driven cavity flow containing a soft sphere, and swimmers actuated via active stress.

Introduction

Fluid–structure interactions (FSI) are at the heart of many important physical and biological problems, including flexible structures in flow [1, 2], blood circulation in the heart [3, 4], animal locomotion [5, 6], and cilia motion [7, 8]. The couplings between fluid and immersed solids give rise to complex nonlinear dynamics dependent on geometry and boundary conditions, material constitutive relations, and collective interactions among the solid objects. Thus analytical solutions are rare and limited to simplified settings in reduced dimensions, and numerical methods for FSI have become indispensable for understand these problems.

In designing numerical methods for fluids and solids, Eulerian and Lagrangian perspectives are the more convenient choices, respectively, due to the differences in constitutive responses. Bridging between the two perspectives is a classic dilemma in developing numerical methods for FSI. Various frameworks have been proposed to resolve this dilemma. The immersed boundary method [3, 9] and the family of immersed methods that it inspires [4, 10, 11] solve fluid on a fixed mesh, use Lagrangian points to represent the solid, and employ a coupling scheme between the two. Arbitrary Lagrangian–Eulerian methods use moving nodes for both phases, and reposition the nodes to maintain mesh quality [12, 13].

There are also fully Eulerian methods, for which two types of formulations exist. The hypoelasticity formulation [14, 15, 16] uses linear elasticity, whereas the hyperelasticity formulation employs a general large-deformation description in the solids. To compute solid stresses in hyperelasticity, a variety of quantities have been used, such as level-set functions defining the fluid–solid interface [17], the deformation gradient tensor [18], the deformation tensors [19], and the solid displacements in the undeformed configuration [20, 21, 22].

The reference map technique (RMT) is a fully Eulerian approach that uses the solid's reference coordinates as the primary simulation variables. The RMT has many favorable properties, and can simulate compressible fluid and solids [23, 24], handle contact between multiple solids [25], and simulate incompressible fluid and solids with complex geometries and actuation [26]. The RMT makes use of the level-set method [27, 28] for tracking the fluid–solid interfaces, and all prior versions have employed level-set reinitialization using the fast marching method (FMM) [28, 16]. Furthermore the FMM is used to extrapolate field values near the fluid–solid interface, which is a necessary part of the RMT.

All prior papers have demonstrated the method in two dimensions, yet a major strength of the fully Eulerian approach is the ability to avoid computational meshing of complex geometries, which becomes a compelling advantage in three

---

[*]Corresponding author. E-mail: chr@seas.harvard.edu

dimensions (3D). Here, we provide the first 3D implementation of the RMT, and use it to simulate several scenarios that would be difficult do with existing numerical approaches. Inspired by recent work on fluid-filled soft granular packings [29], we simulate a suspension of soft flexible particles, half of which are lighter and half of which are heavier than the fluid. There has also been substantial recent interest in swimming and biolocomotion, yet reduced-dimension models of the swimmer [30, 31, 32] are often the state of the art. We demonstrate a model of a swimming organism using full volumetric actuation, creating a simulation tool that can explore a broader range of swimming modalities.

3D calculations are considerably more computationally challenging than two-dimensional (2D) ones, and we therefore present the first complete parallel implementation of the method using domain decomposition. Since the RMT uses a single fixed regular grid for fluid and solid, it is well-suited to parallelization, and different processors can each handle a rectangular subdomain. In addition, the regular grid structure makes contact among solid bodies easy to detect and allows for efficient memory storage. One challenge that we faced is that the FMM is not well-suited for parallelization, since it updates field values sequentially. We therefore a developed new method for extrapolation that can be done in parallel. Our new method removes the need for level-set reinitialization and fast marching methods, and the RMT implementation is therefore simplified from all prior approaches.

Theory and numerical method

Hyperelastic formulation

In the hyperelasticity framework [33], a time-dependent mapping function $\boldsymbol{\chi}(\mathbf{X}, t)$ is introduced to determine how the undeformed configuration, $\mathbf{X}$, is transformed to its current physical configuration, $\mathbf{x}$, i.e. $\mathbf{x} = \boldsymbol{\chi}(\mathbf{X}, t)$ (Fig. 1(a, b)). The deformation gradient is defined as $\mathbf{F} = \frac{\partial \boldsymbol{\chi}}{\partial \mathbf{X}} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}}$. A constitutive relation $\boldsymbol{\sigma}_s(\mathbf{F})$ defines the Cauchy stress response in the solid material. The solid momentum balance equation in rate form is

$$\rho_s \left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} \right) = \nabla \cdot \boldsymbol{\sigma}_s \qquad (1)$$

where ρ_s is the solid density and $\mathbf{u}$ is the solid velocity. For an incompressible material $\det \mathbf{F} = 1$ and thus solid density is unaffected by the deformation. To proceed, we assume that $\boldsymbol{\chi}$ is a sufficiently smooth function of $\mathbf{X}$ and t and define its inverse as the reference map, $\mathbf{X} = \boldsymbol{\xi}(\mathbf{x}, t) = \boldsymbol{\chi}^{-1}(\mathbf{x}, t)$. The deformation gradient tensor becomes

$$\mathbf{F} = \left(\frac{\partial \boldsymbol{\xi}}{\partial \mathbf{x}} \right)^{-1} = (\nabla_{\mathbf{x}} \boldsymbol{\xi})^{-1} \qquad (2)$$

where $\nabla_{\mathbf{x}}$ is the gradient operator in physical space. The reference configuration is constant, therefore $\dot{\boldsymbol{\xi}}(\mathbf{x}, t) = \mathbf{0}$, i.e.,

$$\frac{\partial \boldsymbol{\xi}}{\partial t} + (\mathbf{u} \cdot \nabla)\boldsymbol{\xi} = \mathbf{0}. \qquad (3)$$

We discuss coupling fluid and solid phases and imposing the incompressibility constraint next.

Blurred interface method and monolithic governing equations

Consider a domain Ω containing n immersed solid objects covering subdomains $\Omega_1, \ldots, \Omega_n$. Denote the fluid domain as Ω_f and the solid domain $\Omega_s = \cup_{k=1}^n \Omega_k$ for $k = 1, 2, \ldots, n$, so that $\Omega = \Omega_s \cup \Omega_f$. The fluid–solid interface (hereinafter the interface) is denoted as $\partial \Omega_s = \cup_{k=1}^n \partial \Omega_k$ for $k = 1, 2, \ldots, n$. To avoid excessive distortion, the reference map $\boldsymbol{\xi}$ is only defined and evolved within Ω_s. The solution to Eq. (3) is the union of solutions within each Ω_k. The velocity $\mathbf{u}$ is defined as a global variable spanning Ω. The incompressibility constraint implies that

$$\nabla \cdot \mathbf{u} = 0 \qquad (4)$$

in Ω. To enforce the constraint, a global pressure field is used as a Lagrange multiplier. Thus, we need only consider the deviatoric part of the stress tensors. For the fluid, we consider the deviatoric stress of a viscous Newtonian fluid, $\boldsymbol{\tau}_f = \mu \left(\nabla \mathbf{u} + \nabla \mathbf{u}^\mathsf{T} \right)$, where μ is the fluid dynamic viscosity. The deviatoric solid stress is defined as $\boldsymbol{\tau}_s = \boldsymbol{\sigma}_s - \frac{1}{3} \operatorname{tr}(\boldsymbol{\sigma}_s)\mathbf{1}$.

In a physical system, there may be discontinuities in velocity, density, stresses, and forces across the interface. In our method, we consider a continuous velocity field, which naturally corresponds to a no-slip boundary condition at $\partial \Omega_s$. Next, we discuss the blurred interface method that ensures the traction-matching condition, i.e. $\boldsymbol{\sigma}_s \cdot \mathbf{n} = \boldsymbol{\sigma}_f \cdot \mathbf{n}$, is satisfied at the interface, and creates a smooth transition of field values across the interface.

The basic RMT equations work with a variety of interfacial coupling procedures, including sharp and blurred interface methods [24, 25, 26]. In this work we focus on a blurred interface method because it has several advantages over a sharp interface method, e.g., it is more stable to interfacial perturbations, more amenable to simulating immersed solid–solid contact, and easier to implement [25, 26]. Continuing with the blurred interface method, we make use of a blur zone of width 2ϵ across the interface. The width of the blur zone 2ϵ scales with the grid spacing so that as grid spacing approaches zero, a sharp interface representation is recovered. Without loss of generality, we consider a single solid object with interface $\partial \Omega_s$ defined by the zero contour of a signed-distance function in the undeformed configuration, $\phi_0(\mathbf{X})$. Throughout the simulation, we define the time-dependent level-set function $\phi(\mathbf{x}, t) = \phi_0(\boldsymbol{\xi}(\mathbf{x}, t))$ (Fig. 1(b)).
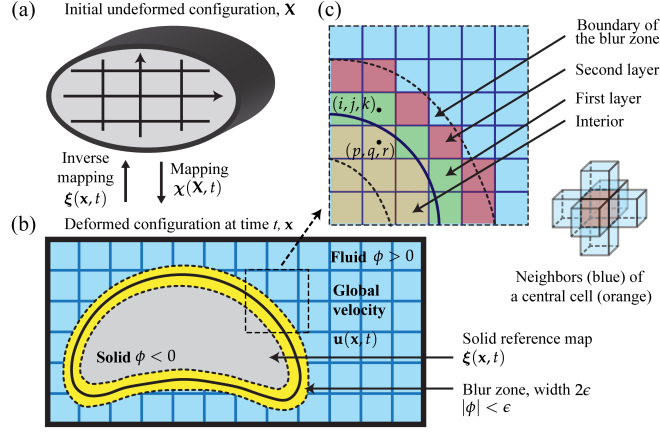
Figure 1: (a) An initially undeformed solid with reference configuration $\mathbf{X}$ undergoes a time-dependent mapping $\boldsymbol{\chi}(\mathbf{X}, t)$ to its current configuration at time t. (b) A level-set function $\phi(\mathbf{x}, t)$ distinguishes the two phases which share a global velocity $\mathbf{u}(\mathbf{x}, t)$. A blur zone (yellow), defined by $|\phi| < \epsilon$, is used to transition between phases. (c) The order in which $\boldsymbol{\xi}(\mathbf{x}, t)$ is extrapolated is defined by layers. The 1st layer (green), e.g. cell (i, j, k), are non-interior orthogonal neighbors to the interior cells (yellow), e.g. cell (p, q, r). Subsequent layers, e.g. 2nd layers (red), are constructed in the same way, until the blur zone is filled or a physical boundary is reached. 2D schematics are shown for clarity.

In the blur zone, quantities that may have jump across the interface, e.g. ρ, $\boldsymbol{\tau}$ and $\mathbf{b}$, are blended to smoothly vary between the fluid and the solid phases [34, 35, 36]. Let Q denote a scalar, or a component of a vector or a tensorial quantity in Ω. Consider Q_f in the fluid domain Ω_f, and Q_s in the solid domain Ω_k. We blend Q_f and Q_s such that Q transitions smoothly between Ω_f and Ω_k,

$$Q = Q_s + H_\epsilon(\phi)(Q_f - Q_s) \tag{5}$$

where ϕ is the value of the level-set function defining the boundary of Ω_k, and $H_\epsilon(\phi)$ is a smoothed Heaviside function,

$$H_\epsilon(\phi) = \begin{cases} 0 & \text{if } \phi \leq -\epsilon, \\ \frac{1}{2}(1 + \frac{\phi}{\epsilon} + \frac{1}{\pi}\sin\frac{\pi\phi}{\epsilon}) & \text{if } |\phi| < \epsilon, \\ 1 & \text{if } \phi \geq \epsilon, \end{cases} \tag{6}$$

which has a continuous second derivative. The momentum balance equation of the coupled fluid–structure system is

$$\rho\left(\frac{\partial\mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) = -\nabla p + \nabla\cdot\boldsymbol{\tau} + \mathbf{b} \tag{7}$$

with blended ρ, $\boldsymbol{\tau}$, and $\mathbf{b}$. The monolithic equation above satisfies the flow equation and elasticity equation, in the fluid and the solid phases, respectively. Furthermore, the traction-matching condition is also automatically satisfied due to blending of stresses near the interface. Eqs. (3), (4), & (7) form a single set of governing equations for the coupled FSI system.

In the solid bodies, we add an artificial viscosity μ_a to improve numerical stability. If μ_a scales with grid size, in the limit of very fine spatial resolution, we recover the undamped solid equation. If it is set to a grid size-independent constant, it is equivalent to simulating a Kelvin–Voigt viscoelastic solid. We also define a dimensionless constant γ_t that applies an additional multiplicative factor to the artificial viscosity in the blur zone (see SI for details).

Extrapolation of $\boldsymbol{\xi}(\mathbf{x}, t)$

Since $\boldsymbol{\xi}(\mathbf{x}, t)$ is only defined inside the solid, it needs to be extrapolated to several grid cells outside of the interface for calculating derivatives in Eq. (3) and the deformation gradient tensor $\mathbf{F}$ near the interface. We describe our new extrapolation method for a single solid occupying domain Ω_s, though it can be easily applied to any number of objects. First we simplify the spatial order in which $\boldsymbol{\xi}(\mathbf{x}, t)$ is extrapolated by making use of adjacency rules on a fixed grid. Consider a Cartesian mesh with $M \times N \times O$ grid cells indexed by i, j, k, we define orthogonal neighbors of a central cell as those cells that share a common face with the central cell (Fig. 1(c)). We label cells with $\phi < 0$ as the interior cells, or equivalently, as cells in the zeroth layer. All the other cells are left as unmarked. Then, for each interior cell, we label its unmarked orthogonal neighbors as the first layer cells, with index $l = 1$. We repeat this procedure to find subsequent layers $l = 2, 3, \ldots$, until we reach a physical boundary or the maximum number of layers, whichever occurs earlier. The maximum number of layers is chosen so that the entire blue zone can be covered by the extrapolation procedure while conserving computational resources.

The extrapolation is then performed in ascending order of layers, but it can be computed independently for each cell

within the same layer. Suppose we aim to create an extrapolated reference map, $\boldsymbol{\xi}_e = (\xi_i, \xi_j, \xi_k)$ in layer $l_{i,j,k}$, with cell index (i, j, k). Subscript e denotes an extrapolated cell. Using WLS regression, we first build a local linear model of the reference map $\boldsymbol{\xi}(\mathbf{x}, t)$ as a function of physical coordinate $\mathbf{x}$. To find eligible data points for the regression, we search within a $r_s = 2$ box centered at cell (i, j, k), where r_s is the search radius measured in number of grid cells. Consider a cell with index (p, q, r) with $\boldsymbol{\xi}_d = (\xi_p, \xi_q, \xi_r)$, where subscript d denotes a data cell. It is eligible to be a data point in the regression only if it is marked as a layer in the previous procedure, and $l_{p,q,r} < l_{i,j,k}$. In other words, reference map $\boldsymbol{\xi}_d$ must exist and be in layers lower than that of $\boldsymbol{\xi}_e$. In the case of multiple solid objects, $\boldsymbol{\xi}_d$ must emanate from the same object whose extrapolation we seek.

The weights in the regression are important to ensure the quality of the extrapolated values, especially when local deformations are large. In the weighting scheme, we use an exponential decaying kernel centered at the extrapolated cell, and near the interface we incorporate geometric information via ϕ_0. Details are provided in the SI. The WLS problem is solved to obtain a set of coefficients $\boldsymbol{\beta}$, and extrapolated reference map is calculated by $\boldsymbol{\xi}_e(\mathbf{x}_e, t) = \boldsymbol{\beta}\mathbf{x}_e$. If the linear system is degenerate, we increase the search radius r_s by 1 and repeat the procedure from searching for eligible data points, but this is rare in practice.

In multi-body simulations, the extrapolation procedure is applied to each object independently. We require that solid bodies do not co-exist at a grid cell. However, the blur zone of an object is allowed to overlap with blur zones or interiors of other objects. Thus, at a single grid cell, there can be several reference maps, extrapolated or not, each belonging to a distinct object. Since extrapolated values are only needed in a small region near the interface, we design a custom data structure that is tailored to store these extrapolated values efficiently in memory. Besides eliminating the need of reinitialization, the current extrapolation method has two additional advantages: (1) layers can be defined given a definition of adjacency on the grid; (2) the method is layer-wise and object-wise independent, thus easy to be parallelized.

Numerical procedures and implementation

The RMT implementation in 3D (RMT3D) is developed in C++ and parallelized via domain decomposition using the Message-Passing Interface (MPI) library. The numerical schemes extend our previous work on 2D RMT implementation [26] and follow established discretizations for solving hyperbolic conservative laws [37, 36]. In summary, we extend from previous works the variable arrangement on the grid, finite-difference schemes to compute spatial derivatives, and Godunov-type upwinding scheme for hyperbolic conservation laws [38] to handle the advective parts of Eqs. (3) & (7). In addition, an approximate projection method [37, 39] and a marker-and-cell projection method are used to enforce the incompressibility constraint (Eq. (4)) on the velocity solution at each timestep, and on an intermediate velocity field between two timesteps, respectively. Large linear systems from the projection methods are solved using a custom geometric multigrid solver. In many-body simulations, a collision stress-based contact model developed in the previous work [26] is used. Details of the numerical schemes and convergence tests are provided in the SI.

Results

In this section, we consider immersed viscoelastic neo-Hookean solids (constant μ_a) in various settings. We nondimensionalize the governing equations using appropriate length, time, and mass scales in each test case. We also use isotropic grid spacing h in all simulations.

Settling sphere in a square cylinder

The transport of rigid and deformable particles in fluid flow is central in many biological and physical systems. While analytical solutions are available for simple cases such as a rigid sphere in unbound creeping flow, more complex FSIs in this setting elude analytical approaches. In recent decades, the effects of confinement and multi-body interactions during settling have been a focus of much experimental and numerical work [41, 40, 42, 43, 44, 45]. In this section, we simulate a solid sphere with a high shear modulus settling in a confined geometry. We aim to demonstrate that by simply increasing the solid shear modulus, our method can easily capture the dynamics of a settling rigid sphere at various Reynolds number.

We compare simulated data with experimental measurements of positions, velocities, and wall correction factors on the terminal velocity of a rigid sphere settling in a square cylinder [40, 41]. We devise a dimensionless parameter $\zeta \equiv \rho_f G d/(\rho_s \mu u_0)$ that compares the strength of the solid elastic stress against that of the viscous stress, u_0 is the terminal velocity of a sphere in an unbound creeping flow, G is the shear modulus, and d is the sphere diameter. We find that a moderate value of G, much smaller than the experimental values, suffices to satisfy $\zeta \gg 1$ and ensure minimum elastic deformation. To accurately capture settling dynamics, it is critical to make sure that the artificial viscosity does not excessively add to the viscous drag at the interface. There are two sources of additional drag in our method: (1) viscous stress due to the use of μ_a and γ_t in the blur zone and (2) solid shear stress blended into the fluid side of the blur zone. To address (1) and still ensure stability, we restrict μ_a by $\mu_a/\mu \in [1, 4]$. In the meanwhile, we set $\gamma_t = \mu/\mu_a - 1$ and the blur zone width $\epsilon = 0.5h$ so that the total viscosity on the fluid side is exactly μ. The solid elastic energy in
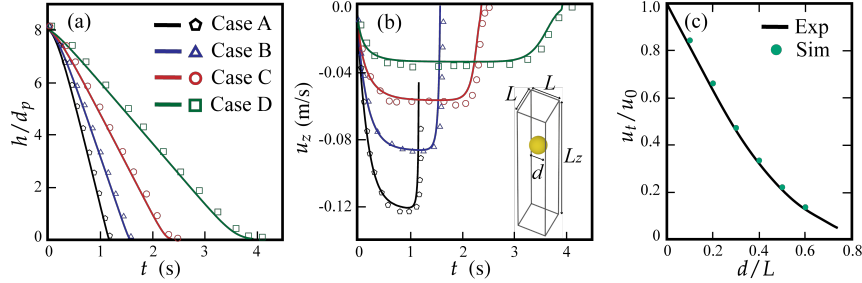
Figure 2: Comparisons of settling sphere simulations against experiments. The positions (a) and velocities (b) of a simulated sphere (solid lines) are compared with experimental results (symbols) by ten Cate *et al.* [40]. Parameters are given in Table 1. The inset in (b) shows an example simulation. (c) The wall correction factors in simulations are compared against experimental results by Miyamura *et al.* [41], using parameters $(L, L_z, \rho_s, \rho_f, G, \mu, \mu_a, \gamma_t, h, \epsilon) = (1, 6, 2, 1, 10, 0.04714, 0.04714, 0, 1/160, 1/320)$. Actual $Re_p = 0.021, 0.132, 0.319, 0.537, 0.694, 0.743$, respectively, for spheres with increasing diameters.

all simulations in this section is $< 8 \times 10^{-3}\%$ of the total energy, confirming that the elastic deformation is negligible and so is the additional drag due to (2).

The simulation domain and results are shown in Fig. 2. In Table 1 we report the physical parameters used in experiments by ten Cate *et al.* [40] and the corresponding dimensionless simulation parameters. As shown in Fig 2(a, b), positions and velocities in simulations agree well with all four experiments by ten Cate *et al.* [40]. To resolve the lubrication layer when the sphere approaches the bottom, we use an appropriate grid resolution ($h = 1/160$) but no additional treatments. In addition, we apply repulsive forces to the solid when it breaches a threshold distance to the bottom, to keep it from penetrating the wall (see SI for details).

The presence of walls drastically modifies the flow, reducing the terminal velocity from u_0 to u_t. Define the ratio of the object size to the confinement size as $\eta = d/L$, where L is the width of the square cylinder cross-section. The reduction factor u_t/u_0 can be expressed as $u_t/u_0 = f(\eta)$, and it is measured experimentally by Miyamura *et al.* [41]. To match these results, we first nondimensionalize length, time, and mass using $l_0 = 0.1$ m, $t_0 = 1$ s, $m_0 = 10^{-3}$ kg, respectively. The terminal velocity according to Stokes' law is $u_0 = (\rho_s - \rho_f)gd^2/18\mu$. To emulate the flow conditions in these experiments, μ is chosen so that the particle Reynolds number $Re_p = \rho_s u_0 d/\mu$ is small. The actual terminal velocity u_t is averaged over time after it has been reached. The apparatus used by Miyamura *et al.* has a height to width ratio of 100:1, which is hard to achieve in our simulations. Instead we use a square cylinder with aspect ratio 6:1 and only consider data from the central 2/3 of the domain. Fig. 2(c) shows that the wall correction factors in our simulations agree well with a reported curve fit of the experimental measurements [41].

Besides easily simulating a wide range of solid stiffness, another benefit of the RMT is that buoyant or neutrally buoy-

ant solids require no special treatment to address the added-mass effect, a numerical difficulty often suffered by partitioned FSI methods [46]. This is an important advantage as many FSI problems of interest involve such density ratios in the solid and fluid phases, e.g. problems in hemodynamics [47] and biomechanics [7]. To demonstrate this, as well as the RMT's ability to simulate complex suspensions [48], we show a simulation of the settling of 150 soft ellipsoids in Fig. 3.

Lid-driven cubic cavity with a sphere

In computational fluid dynamics the lid-driven cavity has long been an important benchmark problem [49, 50]. Despite its simplicity, it exhibits rich flow dynamics due to varying cavity geometries, boundary conditions, and Reynolds numbers. In stark contrast to the extensive studies on the fluid problem in both 2D and 3D, results on lid-driven cavities with deformable boundaries and immersed solids are much fewer [20, 51, 52, 53]. In this section, we turn our attention to a neutrally buoyant deformable sphere in a lid-driven cavity, which to our best knowledge is the first 3D result of its kind.

Shown in Fig. 4(a), the cavity has size $L_x \times L_y \times L_z$ and two span aspect ratios, L_z/L_x and L_y/L_x. The lid moves with velocity $\mathbf{u}_{\text{top}} = (u_x, 0, 0)$ and no-slip boundary conditions are applied on all the other walls. We rescale length and velocity by L_x and u_x, respectively. As a validation, we simulate lid-driven cavity flows without a solid, configured with various span aspect ratios and Reynolds numbers. Our results agree well with high accuracy benchmarks [49] (see SI for details).

A circular particle in a square lid-driven cavity in 2D has been investigated by Zhao *et al.* [51] and widely used as a validation case in later works [19, 54, 55, 56]. We simulate a sphere in a cubic cavity but choose parameters similar to
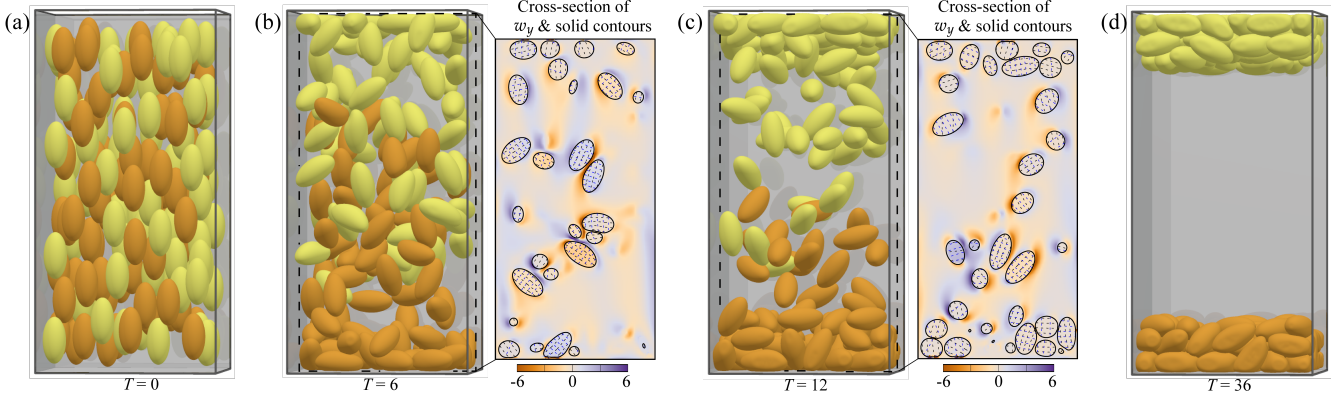
Figure 3: Various stages of 150 ellipsoids settling in a square cylinder (see SI for full movie). All ellipsoids have major axis $R = 0.13$ initially aligned with z-direction, aspect ratio 2:1:1. Half of the ellipsoids are buoyant, $\rho_s/\rho_f = 0.8$ (yellow) and the others are denser than the fluid $\rho_s/\rho_f = 1.25$ (orange). Other parameters are $(L_x, L_y, L_z, \rho_f, G, \mu, \mu_a, \gamma_t, h, \epsilon) = (1, 1, 2, 1, 10^{-1}, 10^{-3}, 10^{-3}, 0, 1/128, 1/128)$. No-slip boundary conditions are applied on all the walls. In (b) and (c) cross-sections at $y = 0.5$ are shown, color corresponds to the y component of vorticity. Fluid–solid interfaces are plotted with a thick black line, and contours of reference map components ξ_x and ξ_z are plotted with black and blue dashed lines, respectively. A contact model among solid bodies via collision stress [26] is used. The maximum particle Reynolds number is approximately 20. This simulation was run with 48 MPI processes on Intel "Cascade Lake" and took 19 hours to simulate to $T = 36$.

Table 1: Physical parameters (columns 2–3) used in the experiments by ten Cate *et al.* [40] and dimensionless simulation parameters (column 4–7). Experiment: sphere density $1120\,\mathrm{kg/m^3}$, diameter 15 mm, and domain dimensions $10\,\mathrm{cm} \times 10\,\mathrm{cm} \times 16\,\mathrm{cm}$. Simulation: M, L, T are units of mass, length, and time, respectively. Length and time are rescaled by $l_0 = 0.1$ m and $t_0 = 0.1$ s, respectively. Mass is rescaled in each case so that the fluid density is $1\,\mathrm{M/L^3}$. For all cases, $h = 1/160, \epsilon = 0.5h$.

	ρ_f (kg/m³)	μ (kg/(ms))	ρ_s (M/L³)	μ (10^{-3}M/(LT))	μ_a/μ –	G (M/LT²)
A	970	0.373	1.155	3.845	1	5.0
B	965	0.212	1.161	2.197	1	2.5
C	962	0.113	1.164	1.175	2	2.0
D	960	0.058	1.167	0.6042	4	2.0

those in the 2D test case to highlight qualitative differences in 3D (Fig. 4(b, c)). The middle cross-section of the deformed sphere (Fig. 4(b)) is qualitatively similar to the shape of the deformed circular particle at long time [56]. The distinctions between the two cases are more apparent in their centroid trajectories. Compared with the 2D case [19], although the centroid of a sphere in a cubic cavity (Fig. 4(c)) also converges to a stationary point, each spiral of its trajectory is much closer to the neighboring ones. There are several reasons for this, most notably the topological difference between an infinite cylinder and a sphere. Another reason is the reduced circulation due to lateral walls in the third dimension [50], which allows the sphere to interact with the moving lid for a longer time before being carried back to the center of the cavity by the flow.

We also simulate spheres with varying shear moduli. Snapshots of a simulation with $G = 0.03$ are shown in Fig. 4(a). To our knowledge, this is the lowest shear modulus reported in the literature for cavity flow with deformable solids. In addition, as Fig. 4(e, g) show, as the shear modulus increases, the sphere exhibits distinct centroid trajectories.

Fig. 4(d, f) offer some intuition for the qualitative changes. As a stiffer sphere moves toward and along the top lid, it deforms less and is able to separate earlier from driving flow. Consequently, a stiffer sphere is carried by the flow from the top right more toward the center than toward the bottom.

We also want to show the effect of the lateral walls on the solid deformation. We simulate a sphere with $G = 0.1$ initially at $(0.6, 0.5L_y, 0.5)$ in cavities with $L_y/L_x = 0.5, 4$. Fig. 4(h–j) show two views of the sphere at the closest approach to the top lid in each cavity. As the walls become farther apart, the sphere becomes less stretched lengthwise and less compressed vertically, again suggesting that a reduced circulation increases the strength of the interaction between the sphere and the moving boundary. We note that here we do not impose any repulsive forces on the sphere near the walls to avoid interfering with its dynamics, thus properly resolving the lubrication layer between the sphere and the boundaries is critical in keeping it from penetrating the walls. In simulations with a coarse resolution and a low G, a nonzero γ_t is needed to dampen the motion of the interface near the boundaries.
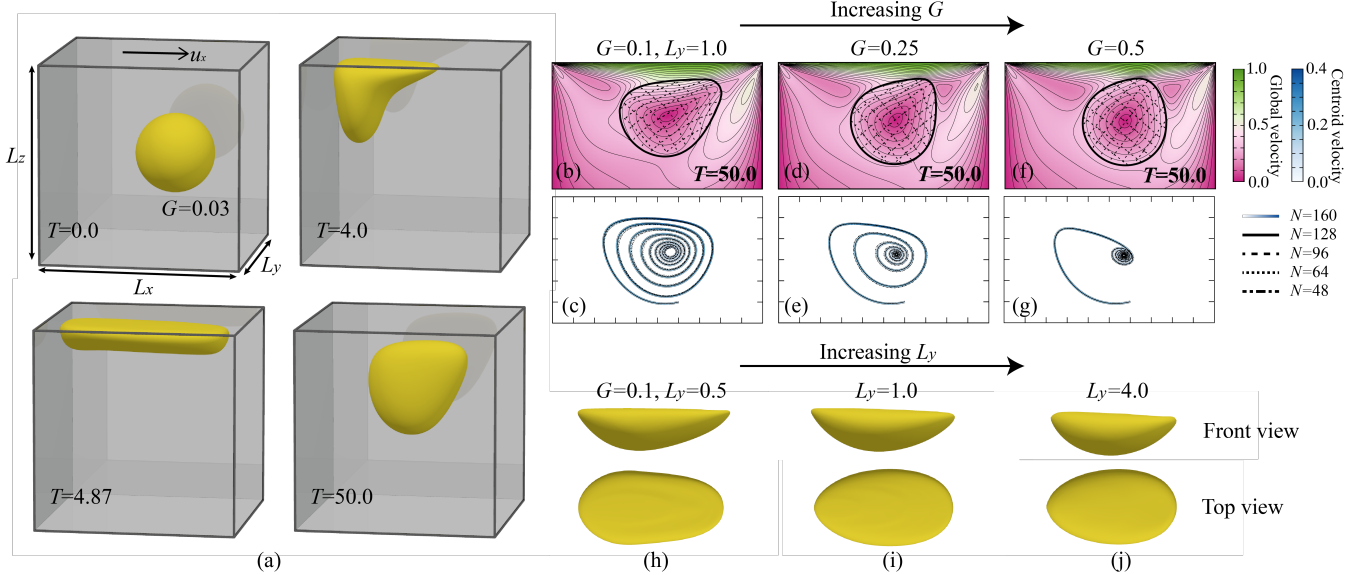
Figure 4: Simulations of a sphere in the lid-driven cavity. For all cases, parameters $(L_x, L_z, u_x, \rho_s, \rho_f, \mu, \mu_a, \epsilon) = (1, 1, 1, 1, 1, 10^{-2}, 10^{-2}, 1.5h), \gamma_t \in [0, 4], h \in [1/160, 1/48]$. The sphere has radius 0.2 and is initially centered at $(0.6, 0.5L_y, 0.5)$. (a) 3D snapshots of a simulation at various times, $(L_y, G) = (1, 0.03)$. (b) Contours of the reference map and the interface are plotted against a background of velocity magnitude (heatmap and contours). A cross-section at $y = 0.5$ and $T = 50$ is shown, $(L_y, G) = (1, 0.1)$, $h = 1/N$. (c) The trajectory of the sphere centroid from $T = 0$ to $T = 50$ of the sphere in (b). The color corresponds to the velocity magnitude of the centroid. (d), (f) The same as (b) but for solid shear moduli $G = 0.25$ and 0.5, respectively. (e), (g) The same as (c) but for solid shear moduli $G = 0.25$ and 0.5, respectively. (h), (i), (j) 3D shape of a deformed $G = 0.1$ sphere at the closest approach to the top in cavity with $L_y = 0.5, 1.0, 4.0$, respectively. See SI for additional parameters, movies, and trajectory data.

In this section, we demonstrate that our numerical schemes, including the new extrapolation method, perform well under the stress of simulating extreme deformations near boundary singularities in a dynamic flow.

Swimming

We now apply the RMT to model swimming where the active deformations in the solid phase drive the motions in the system. Swimming has been an FSI problem of interest for decades [57, 58, 59]. To address the difficulty of resolving the full FSI, especially motions of the phase boundary, modelers often apply simplifications such as asymptotic analysis and scaling arguments. A related, widely applied numerical approach in the low Reynolds number regime is to abstract the swimmer into a one-dimensional collection of regularized singularities [6, 32, 60, 61, 62, 63]. As the swimmer undergoes prescribed deformation or active forcing, the resulting flow is a good approximation to the swimmer's far field. However, this method is not well suited for contexts where the near field is of prime importance, such as dense suspensions of swimmers and swimming in tight confinement. These cases often require specializations [64, 65] or immersed boundary methods [8, 66]. Here, we show that

the RMT naturally resolves the near field around finite-size swimmers in confined geometries. Despite being an Eulerian method, the RMT provides easy access to the reference map, allowing for straightforward definition of the active stresses in the swimmer's body frame.

We begin with a description of our model swimmer, a cylindrical flagellum of length L and radius R with a spherical head of radius $R_h > R$ as shown in Fig. 5(a), and the active stress driving its cyclic deformation. We decompose the solid stress tensor into passive and active parts $\boldsymbol{\sigma}_s = \boldsymbol{\sigma}^{(\mathrm{p})} + \boldsymbol{\sigma}^{(\mathrm{a})}$, where $\boldsymbol{\sigma}^{(\mathrm{p})} = \boldsymbol{\sigma}^{(\mathrm{p})}(\mathbf{F})$ is the elastic stress tensor from previous sections. We seek to define a time-dependent active stress field $\boldsymbol{\sigma}^{(\mathrm{a})} = \boldsymbol{\sigma}^{(\mathrm{a})}(t, \boldsymbol{\xi}, \mathbf{F})$ as a function of body position and local deformation, which will induce a planar bending wave traveling along the cylinder flagellum. First, we specify how body orientation is determined in the deformed frame. We assume without loss of generality that the reference frame coordinate system is centered on the flagellum and aligned with the swimmer. Denoting the reference frame's orthonormal basis as $\{\tilde{\mathbf{e}}_I\}$, $I \in X, Y, Z$, we orient $\{\tilde{\mathbf{e}}_I\}$ so $\tilde{\mathbf{e}}_X$ points along the body toward the head and $\tilde{\mathbf{e}}_Z$ vertically up. The directions of these vectors in the deformed space, $\{\mathbf{e}_I\}$, where $\mathbf{e}_I = \mathbf{F} \cdot \tilde{\mathbf{e}}_I / \|\mathbf{F} \cdot \tilde{\mathbf{e}}_I\|$, are also body-aligned as shown in Fig. 5(a).
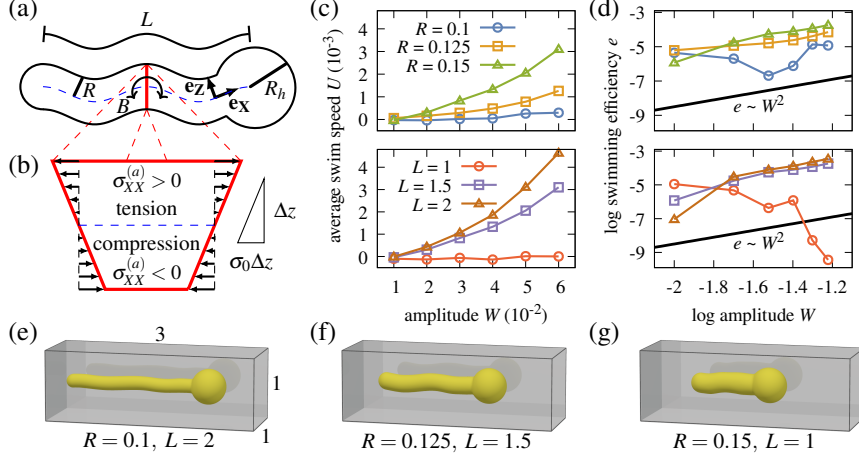
Figure 5: Swimming of flagellar objects driven by active bending moments. (a) Schematic of a swimmer. Swimmers have a spherical head of radius R_h and a cylindrical body of length L and a body radius $R < R_h$. The body is actuated by traveling wave of bending moments B applied to vertical cross-sections (marked in red). (b) Expanded view showing the active in-plane stresses $\sigma_{XX}^{(a)}$ directed along the body's long axis $\tilde{\mathbf{e}}_X$. The stress magnitude varies linearly with the height above the horizontal midplane (a dashed blue line). The swim speed U (c) and efficiency e (d) as a function of W, the bending wave amplitude, are presented for a variety of body shapes. Simulations in the top row vary R with $L = 1.5$, and those in the bottom row vary L with $R = 0.15$. (e–g) Body shapes, mid-stroke, are shown for several R, L combinations. See SI for simulation movies. Steady-state measurements are taken at time T. $T = 20$ for $R = 0.15$, and $T = 10$ otherwise. For all cases, $(L_x, L_y, L_z, R_h, \rho, G, \mu, \mu_a, \gamma_t, h, \epsilon) = (3, 1, 1, 0.25, 1, 1, 10^{-2}, 10^{-2}, 0, 1/96, 1/192)$.

Now, we define the active stress in terms of the reference map coordinates $X(\mathbf{x}, t) = \tilde{\mathbf{e}}_X \cdot \boldsymbol{\xi}(\mathbf{x}, t)$ and $Z(\mathbf{x}, t) = \tilde{\mathbf{e}}_Z \cdot \boldsymbol{\xi}(\mathbf{x}, t)$, which denote the reference distance along the cylinder and above the midplane, respectively. Bending moments are induced by axial stresses $\sigma_{XX}^{(a)} = \mathbf{e}_X \cdot \boldsymbol{\sigma}^{(a)} \cdot \mathbf{e}_X$ of opposing sign about the midplane $Z = 0$, as shown in Fig. 5(b), so we set $\sigma_{XX}^{(a)} = \sigma_0 Z$ for simplicity. Letting the full tensor be traceless, we set

$$\boldsymbol{\sigma}^{(\mathrm{a})} = \frac{3Z\sigma_0}{2}\left(\mathbf{e}_X \otimes \mathbf{e}_X - \frac{1}{3}\mathbf{1}\right), \qquad (8)$$

where $\sigma_0 \propto \cos(kX - \omega t)$ for some wavenumber k and frequency ω. Introducing the amplitude parameter $W = B/3GIk^2$, where I is the cylinder's cross-sectional area moment of inertia and B a bending moment magnitude, we set

$$\sigma_0 = 3WGk^2 \cos(kX - \omega t), \qquad (9)$$

to systematically set bending moments across a range of R. We emphasize that W is not a prescribed or measured vertical displacement, but a parameter derived from scaling arguments describing linear rod bending (see SI for details).

We nondimensionalize variables and parameters using length, time, and stress scales $l = 2\pi/k$, $\tau = 2\pi/\omega$, $\Sigma = G$. We simulate the dimensionless, uniform density system at constant ρ, μ and R_h with varying R, W, and L. For each combination of parameters, we calculate the average swim speed U, shown in Fig. 5(c), and active power

$P = \left\langle -\int_{\Omega_s} \boldsymbol{\sigma}^{(a)} : \nabla \mathbf{u}\, dV \right\rangle$, where $\langle * \rangle$ denotes time averaging over many cycles of oscillation. We also calculate an approximate Lighthill efficiency $e = CU^2/P$ [67], shown in Fig. 5(d), where C is a drag coefficient used to estimate the force required to tow the swimmer at velocity U. Mid-stroke body shapes for selected parameters are shown in Fig. 5(e–g). See SI for details and movies.

Broadly, swimming with the prescribed active stress is faster and more efficient at larger body sizes. This may be in part due to the changing Reynolds number Re_s corresponding to the time-averaged object motion, since it grows as $Re_s \sim W^2$ over the range of simulated W values; in contrast, the Reynolds number Re_o describing the swimming gait varies more slowly as $Re_o \sim W$. Both $Re_s, Re_o \in [10^{-4}, 0.1]$ for the results presented in Fig. 5 (see SI for details.) At large amplitudes, the efficiency scales as $e \sim W^2$, consistent with $U, P \sim W^2$. Notably, there is effectively no motion at $L = 1$, suggesting a minimum length is required for efficient locomotion.

Conclusion and future work

The reference map technique is an efficient and flexible numerical method for FSI problems that involve many bodies, complex solid geometries, large deformations, and actuation. We have demonstrated the accuracy of its first 3D implementation, RMT3D, by convergence tests and comparisons

against experimental data of settling spheres. We have also presented its applications to simulate settling of a large number of ellipsoids with varying density, a soft sphere in several lid-driven cavity flows, and swimmers with different body geometries, actuated by active stress.

A major future direction for the RMT is to address the issue of solid self-contact, which is a common occurrence in geometrically large deforming bodies, such as long slender structures and active swimmers. Other future work includes developing a more accurate contact model with physical boundaries to capture rebound behavior, as well as applying adaptive mesh refinement techniques to increase computational efficiency in many-body and multi-scale systems.

Acknowledgement

Author contributions
Y. L. L., N. J. D. and C. H. R. designed and performed research, wrote simulation codes, analyzed data, and wrote the paper.

Competing interest
The authors declare no conflict of interest.

Code availability
The simulation codes, RMT3D, are available on GitHub at https://github.com/ylunalin/rmt3D
The custom linear solver, Parallelized Geometric Multigrid (PGMG), which is required by RMT3D, is available on GitHub at https://github.com/chr1shr/pgmg

References

[1] Mitul Luhar and Heidi Nepf. Flow-induced reconfiguration of buoyant and flexible aquatic vegetation. *Limnology and Oceanography*, 56:2003–2017, 11 2011.

[2] Marek Bukowicki and Maria L. Ekiel-Jeżewska. Sedimenting pairs of elastic microfilaments. *Soft Matter*, 15(46):9405–9417, 2019.

[3] Charles S Peskin. Flow patterns around heart valves: A numerical method. *Journal of Computational Physics*, 10(2):252–271, 1972.

[4] Boyce E. Griffith, Xiaoyu Luo, David M. McQueen, and Charles S. Peskin. Simulating the fluid dynamics of natural and prosthetic heart valves using the immersed boundary method. *International Journal of Applied Mechanics*, 01(01):137–177, 2009.

[5] Kelsey N. Lucas, Nathan Johnson, Wesley T. Beaulieu, Eric Cathcart, Gregory Tirrell, Sean P. Colin, Brad J. Gemmell, John O. Dabiri, and John H. Costello. Bending rules for animal propulsion. *Nature Communications*, 5(1), December 2014.

[6] Yunyoung Park, Yongsam Kim, and Sookkyung Lim. Locomotion of a single-flagellated bacterium. *Journal of Fluid Mechanics*, 859:586–612, 2019.

[7] Janna C. Nawroth, Hanliang Guo, Eric Koch, Elizabeth A. C. Heath-Heckman, John C. Hermanson, Edward G. Ruby, John O. Dabiri, Eva Kanso, and Margaret McFall-Ngai. Motile cilia create fluid-mechanical microhabitats for the active recruitment of the host microbiome. *Proceedings of the National Academy of Sciences*, 114(36):9510–9516, 2017.

[8] Hanliang Guo, Lisa Fauci, Michael Shelley, and Eva Kanso. Bistability in the synchronization of actuated microfilaments. *Journal of Fluid Mechanics*, 836:304–323, 2018.

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

[10] Thomas G. Fai and Chris H. Rycroft. Lubricated immersed boundary method in two dimensions. *J. Comput. Phys.*, 356:319–339, 2018.

[11] Boyce E. Griffith and Neelesh A. Patankar. Immersed methods for fluid–structure interaction. *Annual Review of Fluid Mechanics*, 52(1):421–448, 2020.

[12] Cyril W. Hirt, Anthony A. Amsden, and J. L. Cook. An arbitrary Lagrangian Eulerian computing method for all flow speeds. *J. Comput. Phys.*, 14:227–253, 1974.

[13] Sandra Rugonyi and Klaus-Jürgen Bathe. On finite element analysis of fluid flows fully coupled with structural interactions. *Comput. Model. Eng. Sci.*, 2:195–212, 2001.

[14] Clifford Truesdell. Hypo-elasticity. *Indiana Univ. Math. J.*, 4:83–133, 1955.

[15] H S Udaykumar, L Tran, D M Belk, and K J Vanden. An Eulerian method for computation of multimaterial impact with ENO shock-capturing and sharp interfaces. *Journal of Computational Physics*, 186(1):136–177, 2003.

[16] Chris H. Rycroft and Frédéric Gibou. Simulations of a stretching bar using a plasticity model from the shear transformation zone theory. *Journal of Computational Physics*, 231(5):2155–2179, 2012.

[17] Emmanuel Maitre, Thomas Milcent, Georges-Henri Cottet, Annie Raoult, and Yves Usson. Applications of level set methods in computational biophysics. *Mathematical and Computer Modelling*, 49(11–12):2161–2169, 2009. Trends in Application of Mathematics to Medicine.

[18] Chun Liu and Noel J. Walkington. An Eulerian description of fluids containing visco-elastic particles. *Archive for Rational Mechanics and Analysis*, 159(3):229–252, 2001.

[19] Kazuyasu Sugiyama, Satoshi Ii, Shintaro Takeuchi, Shu Takagi, and Yoichiro Matsumoto. A full Eulerian finite difference approach for solving fluid–structure coupling problems. *J. Comput. Phys.*, 230(3):596–627, 2011.

[20] Thomas Dunne. An Eulerian approach to fluid–structure interaction and goal-oriented mesh adaptation. *Int. J. Numer. Methods Fluids*, 51(9-10):1017–1039, 2006.

[21] Thomas Richter. A fully Eulerian formulation for fluid–structure interaction problems. *J. Comput. Phys.*, 233:227–240, 2013.

[22] Thomas Wick. Fully Eulerian fluid–structure interaction for time-dependent problems. *Comput. Method. Appl. M.*, 255:14–26, 2013.

[23] Ken Kamrin. *Stochastic and Deterministic Models for Dense Granular Flow*. PhD thesis, MIT, 2008.

[24] Ken Kamrin, Chris H. Rycroft, and Jean-Christophe Nave. Reference map technique for finite-strain elasticity and fluid–solid interaction. *J. Mech. Phys. Solids*, 60(11):1952–1969, 2012.

[25] Boris Valkov, Chris H. Rycroft, and Ken Kamrin. Eulerian method for multiphase interactions of soft solid bodies in fluids. *J. Appl. Mech.*, 82(4):041011, 04 2015.

[26] Chris H Rycroft, Chen-Hung Wu, Yue Yu, and Ken Kamrin. Reference map technique for incompressible fluid–structure interaction. *Journal of Fluid Mechanics*, 898:A9, 2020.

[27] Stanley Osher and James A. Sethian. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations. *J. Comput. Phys.*, 79(1):12–49, 1988.

[28] James A. Sethian. *Level Set Methods and Fast Marching Methods: Evolving interfaces in computational geometry, fluid mechanics, computer vision and materials science*. Cambridge University Press, 1999.

[29] Christopher W. MacMinn, Eric R. Dufresne, and John S. Wettlaufer. Fluid-driven deformation of a soft granular material. *Phys. Rev. X*, 5:011020, Feb 2015.

[30] Eric D. Tytell, Chia-Yu Hsu, Thelma L. Williams, Avis H. Cohen, and Lisa J. Fauci. Interactions between internal forces, body stiffness, and fluid environment in a neuromechanical model of lamprey swimming. *Proc. Natl. Acad. Sci.*, 107(46):19832–19837, 2010.

[31] Becca Thomases and Robert D. Guy. Mechanisms of elastic enhancement and hindrance for finite-length undulatory swimmers in viscoelastic fluids. *Phys. Rev. Lett.*, 113:098102, Aug 2014.

[32] Cole Jeznach and Sarah D Olson. Dynamics of swimmers in fluids with resistance. *Fluids (Basel)*, 5(1):14, 2020.

[33] Morton E. Gurtin, Eliot Fried, and Lallit Anand. *The Mechanics and Thermodynamics of Continua*. Cambridge University Press, 2010.

[34] Mark Sussman, Peter Smereka, and Stanley Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.*, 114(1):146–159, 1994.

[35] Mark Sussman, Ann S Almgren, John B Bell, Phillip Colella, Louis H Howell, and Michael L Welcome. An adaptive level set approach for incompressible two-phase flows. *Journal of Computational Physics*, 148(1):81–124, 1999.

[36] Jiun-Der Yu, Shinri Sakai, and James A. Sethian. A coupled level set projection method applied to ink jet simulation. *Interfaces and Free Boundaries*, 5(4):459–482, 2003.

[37] Ann S. Almgren, John B. Bell, and William G. Szymczak. A numerical method for the incompressible Navier–Stokes equations based on an approximate projection. *SIAM J. Sci. Comput.*, 17(2):358–369, 1996.

[38] Phillip Colella. Multidimensional upwind methods for hyperbolic conservation laws. *Journal of Computational Physics*, 87(1):171–200, 1990.

[39] Elbridge Gerry Puckett, Ann S. Almgren, John B. Bell, Daniel L. Marcus, and William J. Rider. A high-order projection method for tracking fluid interfaces in variable density incompressible flows. *J. Comput. Phys.*, 130(2):269–282, 1997.

[40] A. ten Cate, C. H. Nieuwstad, J. J. Derksen, and H. E. A. Van den Akker. Particle imaging velocimetry experiments and lattice-Boltzmann simulations on a single sphere settling under gravity. *Physics of Fluids*, 14(11):4012–4025, nov 2002.

[41] A. Miyamura, S. Iwasaki, and T. Ishii. Experimental wall correction factors of single solid spheres in triangular and square cylinders, and parallel plates. *International Journal of Multiphase Flow*, 7(1):41–46, feb 1981.

[42] Robert M. MacMeccan, J. R. Clausen, G. P. Neitzel, and C. K. Aidun. Simulating deformable particle suspensions using a coupled lattice-Boltzmann and finite-element method. *Journal of Fluid Mechanics*, 618:13–39, January 2009.

[43] R.G.M. van der Sman. Drag force on spheres confined on the center line of rectangular microchannels. *Journal of Colloid and Interface Science*, 351(1):43–49, November 2010.

[44] Sudeshna Ghosh and John M. Stockie. Numerical simulations of particle sedimentation using the immersed boundary method. *Communications in Computational Physics*, 18(2):380–416, August 2015.

[45] Suresh Alapati, Woo Seong Che, and Yong Kweon Suh. Simulation of Sedimentation of a Sphere in a Viscous Fluid Using the Lattice Boltzmann Method Combined with the Smoothed Profile Method. *Advances in Mechanical Engineering*, 7(2):794198, February 2015.

[46] P. Causin, J.F. Gerbeau, and F. Nobile. Added-mass effect in the design of partitioned algorithms for fluid–structure problems. *Computer Methods in Applied Mechanics and Engineering*, 194(42-44):4506–4527, October 2005.

[47] Thomas G. Fai, Alejandra Leo-Macias, David L. Stokes, and Charles S. Peskin. Image-based model of the spectrin cytoskeleton for red blood cell simulation. *PLOS Computational Biology*, 13(10):1–25, 10 2017.

[48] Anke Lindner. Flow of complex suspensions. *Physics of Fluids*, 26(10):101307, 2014.

[49] S. Albensoeder and H.C. Kuhlmann. Accurate three-dimensional lid-driven cavity flow. *Journal of Computational Physics*, 206(2):536–558, July 2005.

[50] Hendrik C. Kuhlmann and Francesco Romanò. The Lid-Driven Cavity. In Alexander Gelfgat, editor, *Computational Modelling of Bifurcations and Instabilities in Fluid Dynamics*, volume 50, pages 233–309. Springer International Publishing, Cham, 2019. Series Title: Computational Methods in Applied Sciences.

[51] Hong Zhao, Jonathan B. Freund, and Robert D. Moser. A fixed-mesh method for incompressible flow–structure systems with finite solid deformations. *Journal of Computational Physics*, 227(6):3114–3140, March 2008.

[52] Zhi-Qian Zhang, G. R. Liu, and Boo Cheong Khoo. A three dimensional immersed smoothed finite element method (3D IS-FEM) for fluid–structure interaction problems. *Computational Mechanics*, 51(2):129–150, February 2013.

[53] Massimiliano M. Villone and Pier Luca Maffettone. Dynamics, rheology, and applications of elastic deformable particle suspensions: a review. *Rheologica Acta*, 58(3-4):109–130, April 2019.

[54] H. Esmailzadeh and M. Passandideh-Fard. Numerical and Experimental Analysis of the Fluid-Structure Interaction in Presence of a Hyperelastic Body. *Journal of Fluids Engineering*, 136(11):111107, November 2014.

[55] Iman Farahbakhsh, Hassan Ghassemi, and Fereidoun Sabetghadam. A vorticity based approach to handle the fluid–structure interaction problems. *Fluid Dynamics Research*, 48(1):015509, February 2016.

[56] Hugo Casquero, Yongjie Jessica Zhang, Carles Bona-Casas, Lisandro Dalcin, and Hector Gomez. Non-body-fitted fluid–structure interaction: Divergence-conforming B-splines, fully-implicit dynamics, and variational formulation. *Journal of Computational Physics*, 374:625–653, December 2018.

[57] Geoffrey Taylor. Analysis of the swimming of microscopic organisms. *Proceedings of the Royal Society of London. Series A, Mathematical and physical sciences*, 209(1099):447–461, 1951.

[58] E. M Purcell. Life at low Reynolds number. *American Journal of Physics*, 45(1):3–11, 1977.

[59] James Lighthill. Flagellar hydrodynamics. *SIAM Review*, 18(2):161–230, 1976.

[60] Michael J Shelley and Tetsuji Ueda. The Stokesian hydrodynamics of flexing, stretching filaments. *Physica D*, 146(1):221–245, 2000.

[61] 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):031504–031504–14, 2005.

[62] Karin Leiderman and Sarah D Olson. Swimming in a two-dimensional Brinkman fluid: Computational modeling and regularized solutions. *Physics of Fluids*, 28(2):21902, 2016.

[63] Qiang Yang and Lisa Fauci. Dynamics of a macroscopic elastic fibre in a polymeric cellular flow. *Journal of Fluid Mechanics*, 817:388–405, 2017.

[64] Hoang-Ngan Nguyen, Hoang-Ngan Nguyen, Sarah D Olson, Sarah D Olson, Karin Leiderman, and Karin Leiderman. Computation of a regularized Brinkmanlet near a plane wall. *Journal of Engineering Mathematics*, 114(1):19–41, 2019.

[65] B. J. Walker, K. Ishimoto, H. Gadêlha, and E. A. Gaffney. Filament mechanics in a half-space via regularised Stokeslet segments. *Journal of Fluid Mechanics*, 879:808–833, 2019.

[66] Arash Alizad Banaei, Marco Edoardo Rosti, and Luca Brandt. Numerical study of filament suspensions at finite inertia. *Journal of Fluid Mechanics*, 882, 2020.

[67] M. J. Lighthill. *Mathematical biofluiddynamics*. Regional conference series in applied mathematics 17. Society for Industrial and Applied Mathematics, Philadelphia, 1975.

Eulerian simulation of complex suspensions and biolocomotion in three dimensions
Supplementary Information

Yuexia L. Lin[a], Nicholas J. Derr[a], and Chris H. Rycroft[a,b]

[a]John A. Paulson School of Engineering and Applied Sciences, Harvard University, 29 Oxford Street, Cambridge, MA 02138
[b]Mathematics Group, Lawrence Berkeley National Laboratory, 1 Cyclotron Road, Berkeley, CA 94720

1 Details of numerical algorithms on Cartesian grid

The numerical algorithms that we use for modeling fluid–structure interactions (FSI) are built upon a modern implementation for simulating incompressible fluid mechanics that is based upon Chorin's projection method [1, 2]. Since its introduction in 1968, a variety of extensions have been explored in the literature to improve accuracy and stability [3]. We make use of a mature implementation that incorporates many such advancements, developed by Almgren, Bell, and coworkers [4, 5, 6, 7, 8]. We refer the reader to the papers by Yu *et al.* [9, 10] that use this implementation to simulate an inkjet printer nozzle; these papers provide a comprehensive description of the numerical approaches.

In our algorithms, we keep the fluid simulation component the same as in this existing body of work. We then build the reference map technique (RMT) on top of the same framework for handling solid objects, using similar principles in the numerical discretization. The numerical methods and approach are similar to the two-dimensional implementation of the RMT that we previously developed [11]. We use a second-order accurate discretization in space and first-order accurate explicit scheme in time for both the fluid and the solid phases. Due to error contributions from the interfacial coupling procedure, the overall FSI method is approximately first-order accurate in both space and time. Convergence rates are reported for various test cases in sections below.

As described in the main text, the treatment of the reference map field and extrapolation is improved and simplified, and removes the need for specialized level-set methods and reinitialization techniques that have been required in previous implementations [12, 13, 11]. A major benefit of the new implementation is that it is more amenable for parallelization in the distributed memory paradigm, a necessity in 3D simulations.

1.1 Overview

Governing equations

We first provide a broad sketch of the overall numerical algorithm and introduce the staggered arrangement of variables on a Cartesian grid. We discretize the following governing equations for the coupled fluid–structure system,

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u}\right) = -\nabla p + \nabla\cdot\boldsymbol{\tau} + \mathbf{b}, \tag{S1}$$

$$\frac{\partial \boldsymbol{\xi}}{\partial t} + (\mathbf{u}\cdot\nabla)\boldsymbol{\xi} = \mathbf{0}, \tag{S1}$$

$$\nabla\cdot\mathbf{u} = 0, \tag{S1}$$

where ρ is the mass density, $\mathbf{u}$ is the global velocity, $\boldsymbol{\xi}$ is the reference map variable in the solid bodies, p is the global hydrostatic pressure, $\mathbf{b}$ is a body force density if applicable, and $\boldsymbol{\tau}$ is the deviatoric stress tensor. The deviatoric stress tensor $\boldsymbol{\tau}$ is the fluid

stress tensor $\boldsymbol{\tau}_f$ in the fully fluid phase, and it is the solid stress tensor $\boldsymbol{\tau}_s$ in the fully solid phase. Near the fluid–solid interface, $\boldsymbol{\tau}$ is a mixed quantity using a smooth transition function over a blur zone of characteristic width 2ϵ,

$$\boldsymbol{\tau} = H_\epsilon(\phi)\boldsymbol{\tau}_f + (1 - H_\epsilon(\phi))(\boldsymbol{\tau}_s + \boldsymbol{\tau}_a) \tag{S1}$$

where $\boldsymbol{\tau}_a$ denotes the artificial viscous stress tensor added to the solid. $H_\epsilon(\phi)$ is the smoothed Heaviside function, which is defined in the main text and has been used in other works [14, 9, 11]. To simplify notation, we refer to ϕ_0, a signed-distance defined in the reference configuration of the solid object, as ϕ hereinafter. The artificial viscous stress is defined as

$$\boldsymbol{\tau}_a = \mu_a(1 + \gamma_t\epsilon H_\epsilon'(\phi))(\nabla\mathbf{u} + \nabla\mathbf{u}^\mathsf{T}) \tag{S2}$$

where μ_a is the solid artificial viscosity and γ_t is a dimensionless multiplier to amplify the artificial viscosity in the blur zone. The contribution of $\nabla\mathbf{u}^\mathsf{T}$ term to the divergence of viscous stress is negligible due to the incompressibility constraint $\nabla\cdot\mathbf{u} = 0$. Therefore, in the actual computation, we do not compute $\nabla\mathbf{u}^\mathsf{T}$ even though it is present in the fluid stress and the artificial stress tensors.

Variable arrangement

The placement of variables on the computational grid is illustrated in Fig. S1. A domain $[a_x, b_x] \times [a_y, b_y] \times [a_z, b_z]$ is divided in M, N, O number of grid cells, yielding grid spacings $\Delta x, \Delta y, \Delta z$, respectively. To integrate in time, we take steps of size Δt. We denote the nth timestep t_n, and the total number of timesteps U. Thus $t_n = n\Delta t$, $n = 1, 2, \ldots, U$. The discretized velocity solution is denoted $\mathbf{u}_{i,j,k}^n$, where the subscripts i, j, k indicate cell-centered position on the grid, and the superscript n indicates the number of timesteps. Other quantities on the grid are indexed in a similar fashion unless specified otherwise. When applicable, $1/2$ in the indices indicate either cell faces between to grid cells or the midpoint between two timesteps. When spatial indices are omitted, we refer to the solutions in the entire computation domain. Primary variables are mass density $\rho_{i,j,k}^n$, global velocity $\mathbf{u}_{i,j,k}^n$, and reference map variables in solids and blur zones $\boldsymbol{\xi}_{i,j,k}^n$, which are all placed at the cell centers. In addition, there is also a global pressure variable $p_{i,j,k}^{n-1/2}$ placed at the nodes. Note that subscripts i, j, k denote nodal quantities for the pressure field only.

Spatial and temporal discretizations

We discretize Eqs. (S1) & (S1) by

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + [(\mathbf{u}\cdot\nabla)\mathbf{u}]^{n+1/2} = \frac{1}{\rho(\phi^{n+1/2})}\left[-\nabla p^{n+1/2} + \nabla\cdot\left((1 - H_\epsilon(\phi^{n+1/2}))(\boldsymbol{\tau}_s^{n+1/2} + \boldsymbol{\tau}_a^{\tilde{n}}) + H_\epsilon(\phi^{n+1/2})\boldsymbol{\tau}_f^n \right) + \mathbf{b}^{n+1/2} \right], \tag{S2}$$

$$\frac{\boldsymbol{\xi}^{n+1} - \boldsymbol{\xi}^n}{\Delta t} + [(\mathbf{u}\cdot\nabla)\boldsymbol{\xi}]^{n+1/2} = \mathbf{0}. \tag{S2}$$

Note that $\phi^{n+1/2}$ refers to $\phi_0(\boldsymbol{\xi}^{n+1/2})$, not the reinitialized level-set function. A number of terms on the right hand side (RHS) of the above equations are calculated at the half-timestep $n + \frac{1}{2}$ for improved accuracy. If applicable, a body force density can be prescribed at cell centers at mid timestep, $\mathbf{b}_{i,j,k}^{n+1/2}$. Since this is an explicit scheme it is necessary to compute $\boldsymbol{\tau}_f^n$ and $\boldsymbol{\tau}_a^{\tilde{n}}$ using information at timestep n, which results in $\boldsymbol{\tau}_a^{\tilde{n}}$ using quantities at two different times, i.e.,

$$\boldsymbol{\tau}_a^{\tilde{n}} = \mu_a(1 + \gamma_t\epsilon H_\epsilon'(\phi^{n+1/2}))(\nabla\mathbf{u}^n + (\nabla\mathbf{u}^n)^\mathsf{T}). \tag{S2}$$

Better accuracy could be achieved by using a Crank–Nicolson-type update formula [15], but since this would depend on $\mathbf{u}^{n+1}$ it would result in an implicit numerical scheme, which is outside the scope of this work.

To handle the advective terms $[(\mathbf{u}\cdot\nabla)\mathbf{u}]^{n+1/2}$ and $[(\mathbf{u}\cdot\nabla)\boldsymbol{\xi}]^{n+1/2}$, we require at cell faces

1. intermediate velocities $\mathbf{u}_{i\pm1/2,j,k}^{n+1/2}, \mathbf{u}_{i,j\pm1/2,k}^{n+1/2}, \mathbf{u}_{i,j,k\pm1/2}^{n+1/2}$,

2. intermediate reference maps $\boldsymbol{\xi}_{i\pm1/2,j,k}^{n+1/2}, \boldsymbol{\xi}_{i,j\pm1/2,k}^{n+1/2}, \boldsymbol{\xi}_{i,j,k\pm1/2}^{n+1/2}$.

We use a Godunov-type upwinding scheme to compute these quantities, which will be discussed in subsection 1.2.

MAC projection

Using the intermediate face velocities, we can compute the discrete divergence in the cell centers using centered finite differences. However, due to discretization error, this discrete divergence field may not evaluate to precisely zero as expected by Eq. (S1). We employ an intermediate marker-and-cell (MAC) projection to correct the face velocities to satisfy the discrete divergence-free property. This has been shown to improve accuracy and volume conservation in previous work [11]. To do this we first solve the equation

$$\nabla \cdot \left(\frac{1}{\rho}\nabla q^{n+1/2}\right) = \frac{\nabla \cdot \mathbf{u}^{n+1/2}}{\Delta t/2} \tag{S3}$$

for an intermediate solution $q^{n+1/2}$ at the cell centers. The RHS of the equation above is evaluated using the intermediate face velocities. We subtract $\frac{\Delta t}{2\rho}\nabla q$ from the intermediate velocities, thus ensuring the discrete divergences are zero to machine precision. In Eq. (S3) we have dropped the subscripts that indicate cell faces, and we will continue to do so for brevity.

Approximate pressure-Poisson projection

Using update equations (S2) & (S2), an intermediate velocity update $\mathbf{u}^*$ at $t = t_n$ can be computed. Once this is computed, we apply the projection step making use of the approximate finite-element projection introduced by Almgren *et al.* [5]. This involves solving the Poisson equation

$$\nabla \cdot \left(\frac{1}{\rho}\nabla \psi\right) = \frac{\nabla \cdot (\mathbf{u}^* - \mathbf{u}^n)}{\Delta t} \tag{S3}$$

for the pressure correction ψ, after which the pressure is updated using $p^{n+1/2} = p^{n-1/2} + \psi$. The finite-element projection is implemented using trilinear basis functions centered on each cell corner. At a particular pressure point $p_{i,j,k}^n$ located at a cell corner, the corresponding basis function is non-zero over the $2 \times 2 \times 2$ block of adjacent grid cells.

Stress calculations

The fluid deviatoric stress $\boldsymbol{\tau}_f^n$ at time t_n is computed at the cell faces by using centered finite differences on the cell centered velocities $\mathbf{u}^n$. To compute solid deviatoric stress $\boldsymbol{\tau}_s^n$, deformation gradient $\mathbf{F}^n$ is first computed at the cell faces by using centered finite differences on the reference maps, $\mathbf{F} = (\nabla_x \boldsymbol{\xi})^{-1}$. Then $\boldsymbol{\tau}_s^n$ is computed at the cell faces directly using the incompressible neo-Hookean constitutive relation

$$\boldsymbol{\tau}_s^n = G \left(\mathbf{B}^n - \tfrac{1}{3}\operatorname{tr}(\mathbf{B}^n)\right) \tag{S3}$$

where $\mathbf{B}^n = \mathbf{F}^n(\mathbf{F}^n)^\mathsf{T}$. If blur zones of S solid objects overlap, $S > 1$, computations of solid stress and artificial viscous stress remain the same for each object. To avoid notation confusion, we drop the spatial indices, and use subscript m to represent different solid objects. But it is assumed that the discussion below only applies at cell faces where it is needed. In each solid object, we separately compute

$$\boldsymbol{\tau}_{s,m}^n = G_m \left(\mathbf{B}_m^n - \tfrac{1}{3}\operatorname{tr}(\mathbf{B}_m^n)\right), \tag{S3}$$

$$\boldsymbol{\tau}_{a,m}^{\tilde{n}} = \mu_a(1 + \gamma_t \epsilon H_\epsilon'(\phi_m^{n+1/2}))(\nabla \mathbf{u}^n + (\nabla \mathbf{u}^n)^\mathsf{T}), \tag{S3}$$

where $\phi_m^{n+1/2}$ is the specific level-set function defining the fluid–solid interface of solid m, evaluated on the reference map variables of this solid object at time $n + 1/2$. In the overlapping region, the overall stress tensors $\boldsymbol{\tau}_s$ and $\boldsymbol{\tau}_a$ are combined using the respective stress tensors from each of the overlapping solid objects,

$$\boldsymbol{\tau}_s^n = \begin{cases} \sum_{m=1}^S (1 - H_\epsilon(\phi_m^n))\boldsymbol{\tau}_s^n & \text{if } \sum_{m=1}^S (1 - H_\epsilon(\phi_m^n)) \le 1, \\[2ex] \frac{\sum_{m=1}^S (1 - H_\epsilon(\phi_m^n))\boldsymbol{\tau}_s^n}{\sum_{m=1}^S (1 - H_\epsilon(\phi_m^n))} & \text{otherwise.} \end{cases} \tag{S3}$$

The artificial viscous stress tensor $\boldsymbol{\tau}_a^n$ is computed similarly. To blend multiple solid blur zones with the fluid, we modify the volume fraction used in Eq. (S1) to

$$H_\epsilon(\phi^n) = \max\left[1 - \sum_{s=1}^S (1 - H_\epsilon(\phi_s^n)), 0\right]. \tag{S3}$$

1.2 Evaluating the advective term

Advective derivatives

To evaluate advective derivatives on the left hand side of Eqs. (S2), (S2), we use the formulae

$$
\begin{aligned}
[(\mathbf{u}\cdot\nabla)\mathbf{u}]^{n+1/2} =\ & \frac{u_{i+1/2}^{n+1/2}+u_{i-1/2}^{n+1/2}}{2}\frac{\mathbf{u}_{i+1/2}^{n+1/2}-\mathbf{u}_{i-1/2}^{n+1/2}}{\Delta x}\\
&+\frac{v_{j+1/2}^{n+1/2}+v_{j-1/2}^{n+1/2}}{2}\frac{\mathbf{u}_{j+1/2}^{n+1/2}-\mathbf{u}_{j-1/2}^{n+1/2}}{\Delta y}\\
&+\frac{w_{k+1/2}^{n+1/2}+w_{k-1/2}^{n+1/2}}{2}\frac{\mathbf{u}_{k+1/2}^{n+1/2}-\mathbf{u}_{k-1/2}^{n+1/2}}{\Delta z},
\end{aligned}
\tag{S3}
$$

$$
\begin{aligned}
[(\mathbf{u}\cdot\nabla)\boldsymbol{\xi}]^{n+1/2} =\ & \frac{u_{i+1/2}^{n+1/2}+u_{i-1/2}^{n+1/2}}{2}\frac{\boldsymbol{\xi}_{i+1/2}^{n+1/2}-\boldsymbol{\xi}_{i-1/2}^{n+1/2}}{\Delta x}\\
&+\frac{v_{j+1/2}^{n+1/2}+v_{j-1/2}^{n+1/2}}{2}\frac{\boldsymbol{\xi}_{j+1/2}^{n+1/2}-\boldsymbol{\xi}_{j-1/2}^{n+1/2}}{\Delta y}\\
&+\frac{w_{k+1/2}^{n+1/2}+w_{k-1/2}^{n+1/2}}{2}\frac{\boldsymbol{\xi}_{k+1/2}^{n+1/2}-\boldsymbol{\xi}_{k-1/2}^{n+1/2}}{\Delta z},
\end{aligned}
\tag{S3}
$$

where $\mathbf{u}=(u,v,w)$. For brevity, only the indices that differ from i,j,k are shown in the subscripts. To obtain $\mathbf{u}^{n+1/2}$ and $\boldsymbol{\xi}^{n+1/2}$ at a cell face, we use a Taylor expansion using values straddling the cell face, keeping terms up to first order in time and space. Without loss of generality, we consider a face in x-direction, indexed by $i+1/2,j,k$. Two Taylor expansions are constructed, centered at values at cell i,j,k and cell $i+1,j,k$. Expanding from the left side of the face, we have

$$
\mathbf{u}_{i+1/2,j,k}^{n+1/2,\mathrm{L}} = \mathbf{u}_{i,j,k}^{n} + \frac{\Delta x}{2}\frac{\partial \mathbf{u}}{\partial x}\bigg|_{i,j,k}^{n} + \frac{\Delta t}{2}\frac{\partial \mathbf{u}}{\partial t}\bigg|_{i,j,k}^{n}.
\tag{S4}
$$

From the momentum balance equation (Eq. (S1)) we obtain an expression for $\partial\mathbf{u}/\partial t$ as

$$
\frac{\partial \mathbf{u}}{\partial t}\bigg|_{i,j,k}^{n} = -\left(u\frac{\partial \mathbf{u}^n}{\partial x} + \overline{\left(v\frac{\partial \mathbf{u}^n}{\partial y}\right)} + \overline{\left(w\frac{\partial \mathbf{u}^n}{\partial z}\right)}\right)_{i,j,k} + \frac{1}{\rho(\phi^n)_{i,j,k}}\left(-\nabla p^{n-1/2}+\nabla\cdot\boldsymbol{\tau}^n(\mathbf{u}^n,\boldsymbol{\xi}^n,\phi^n)+\mathbf{b}^n\right)_{i,j,k}.
\tag{S4}
$$

The terms on the RHS of Eq. (S4) are evaluated at step n, except for the pressure, which is evaluated at step $n-1/2$. Since we are expanding in the x-direction, the barred terms in the bracket are the tangential derivatives (also called transverse derivatives) and they are treated differently; details will be discussed later in this subsection. Substituting Eq. (S4) into Eq. (S4), we have

$$
\begin{aligned}
\mathbf{u}_{i+1/2,j,k}^{n+1/2,\mathrm{L}} =\ & \mathbf{u}_{i,j,k}^{n} + \left(\frac{\Delta x}{2} - \frac{\Delta t}{2}u_{i,j,k}^n\right)\frac{\partial \mathbf{u}}{\partial x}\bigg|_{i,j,k}^{n} - \frac{\Delta t}{2}\left(\overline{\left(v\frac{\partial \mathbf{u}^n}{\partial y}\right)} + \overline{\left(w\frac{\partial \mathbf{u}^n}{\partial z}\right)}\right)_{i,j,k}\\
&+ \frac{\Delta t}{2\rho(\phi^n)_{i,j,k}}\left(-\nabla p^{n-1/2}+\nabla\cdot\boldsymbol{\tau}^n(\mathbf{u}^n,\boldsymbol{\xi}^n,\phi^n)+\mathbf{b}^n\right)_{i,j,k}.
\end{aligned}
\tag{S4}
$$

Similarly, Taylor expanding from the right side of the cell face, we have

$$
\begin{aligned}
\mathbf{u}_{i+1/2,j,k}^{n+1/2,\mathrm{R}} =\ & \mathbf{u}_{i+1,j,k}^{n} - \left(\frac{\Delta x}{2} + \frac{\Delta t}{2}u_{i+1,j,k}^n\right)\frac{\partial \mathbf{u}}{\partial x}\bigg|_{i+1,j,k}^{n} - \frac{\Delta t}{2}\left(\overline{\left(v\frac{\partial \mathbf{u}^n}{\partial y}\right)} + \overline{\left(w\frac{\partial \mathbf{u}^n}{\partial z}\right)}\right)_{i+1,j,k}\\
&+ \frac{\Delta t}{2\rho(\phi^n)_{i+1,j,k}}\left(-\nabla p^{n-1/2}+\nabla\cdot\boldsymbol{\tau}^n(\mathbf{u}^n,\boldsymbol{\xi}^n,\phi^n)+\mathbf{b}^n\right)_{i+1,j,k}.
\end{aligned}
\tag{S4}
$$

The Taylor expansion procedure to create values of $\boldsymbol{\xi}_{i+1/2,j,k}^{n+1/2}$ makes use of Eq. (S1), which has a much simpler form. Expanding from the left and right sides respectively gives

$$\boldsymbol{\xi}_{i+1/2,j,k}^{n+1/2,\mathrm{L}} = \boldsymbol{\xi}_{i,j,k}^{n} + \left(\frac{\Delta x}{2} - \frac{\Delta t}{2} u_{i,j,k}^{n} \right) \frac{\partial \boldsymbol{\xi}}{\partial x}\bigg|_{i,j,k}^{n} - \frac{\Delta t}{2} \left(\overline{\left(v \frac{\partial \boldsymbol{\xi}^{n}}{\partial y} \right)} + \overline{\left(w \frac{\partial \boldsymbol{\xi}^{n}}{\partial z} \right)} \right)_{i,j,k}, \tag{S4}$$

$$\boldsymbol{\xi}_{i+1/2,j,k}^{n+1/2,\mathrm{R}} = \boldsymbol{\xi}_{i+1,j,k}^{n} - \left(\frac{\Delta x}{2} + \frac{\Delta t}{2} u_{i+1,j,k}^{n} \right) \frac{\partial \boldsymbol{\xi}}{\partial x}\bigg|_{i+1,j,k}^{n} - \frac{\Delta t}{2} \left(\overline{\left(v \frac{\partial \boldsymbol{\xi}^{n}}{\partial y} \right)} + \overline{\left(w \frac{\partial \boldsymbol{\xi}^{n}}{\partial z} \right)} \right)_{i+1,j,k}. \tag{S4}$$

Godunov-type upwinding scheme

Now there are two choices for velocity and two for reference map on the cell face indexed by $i + 1/2, j, k$, we perform Godunov upwinding to select one. We define the normal velocity as $u_{\mathbf{n}}^{n+1/2,\mathrm{L}}$ and $u_{\mathbf{n}}^{n+1/2,\mathrm{R}}$, expanded from the left and the right cells, respectively. We use $q^{n+1/2,\mathrm{L}}$ and $q^{n+1/2,\mathrm{R}}$ to represent other components of the velocity or the reference map, expanded from the left and the right, respectively. To choose the normal velocity, we follow

$$u_{\mathbf{n}}^{n+1/2} = \begin{cases} u_{\mathbf{n}}^{n+1/2,\mathrm{L}} & \text{if } u_{\mathbf{n}}^{n+1/2,\mathrm{L}} > 0 \text{ and } u_{\mathbf{n}}^{n+1/2,\mathrm{L}} + u_{\mathbf{n}}^{n+1/2,\mathrm{R}} > 0, \\ u_{\mathbf{n}}^{n+1/2,\mathrm{R}} & \text{if } u_{\mathbf{n}}^{n+1/2,\mathrm{R}} < 0 \text{ and } u_{\mathbf{n}}^{n+1/2,\mathrm{L}} + u_{\mathbf{n}}^{n+1/2,\mathrm{R}} < 0, \\ 0 & \text{otherwise.} \end{cases} \tag{S5}$$

After this, we select a value for the other quantities at the cell face based on $u_{\mathbf{n}}^{n+1/2}$ according to

$$q^{n+1/2} = \begin{cases} q^{n+1/2,\mathrm{L}} & \text{if } u_{\mathbf{n}}^{n+1/2} > 0, \\ \frac{q^{n+1/2,\mathrm{L}}+q^{n+1/2,\mathrm{R}}}{2} & \text{if } u_{\mathbf{n}}^{n+1/2} = 0, \\ q^{n+1/2,\mathrm{R}} & \text{if } u_{\mathbf{n}}^{n+1/2} < 0. \end{cases} \tag{S5}$$

Here we have dropped all spatial index subscripts since all terms are evaluated at $i + 1/2, j, k$.

Normal derivatives

To compute the normal derivatives (in the example given here, $\partial \mathbf{u}/\partial x$ and $\partial \boldsymbol{\xi}/\partial x$), we employ the fourth-order monotonicity-limited scheme of Collela [7], which is described in detail by Yu *et al.* [9] and Rycroft *et al.* [11].

Tangential derivatives

To ensure stability, especially in intermediate to high Reynolds number regime, we compute the tangential derivatives (shown as the barred terms in Eqs. (S4), (S4), & (S4)) using an upwinding scheme which is commonly used for solving hyperbolic conservative laws [4, 5, 8, 9]. Our approach to construct an upwinding scheme in 3D is only one of the many possibilities, and in selecting the following scheme, we prioritize algorithmic simplicity and the ease implementation.

Without loss of generality, we consider the quantities on the transverse faces indexed by $i, j + 1/2, k$, required in Eq. (S4), (S4), (S4) when applied to faces indexed by $i + 1/2, j, k$. The procedure to compute tangential derivative terms along the z-direction is similar. We start by performing a Taylor expansion again to construct the velocities and reference maps on the transverse cell faces, but neglect contributions from pressure, stress, body forces, and tangential derivatives to obtain

$$\bar{\mathbf{u}}_{i,j+1/2,k}^{n+1/2,\mathrm{D}} = \mathbf{u}_{i,j,k}^{n} + \left(\frac{\Delta y}{2} - \frac{\Delta t}{2} v_{i,j,k}^{n} \right) \frac{\partial \mathbf{u}}{\partial y}\bigg|_{i,j,k}^{n}, \tag{S5}$$

$$\bar{\mathbf{u}}_{i,j+1/2,k}^{n+1/2,\mathrm{U}} = \mathbf{u}_{i,j+1,k}^{n} - \left(\frac{\Delta y}{2} + \frac{\Delta t}{2} v_{i,j+1,k}^{n} \right) \frac{\partial \mathbf{u}}{\partial y}\bigg|_{i,j+1,k}^{n}, \tag{S5}$$

$$\bar{\boldsymbol{\xi}}_{i,j+1/2,k}^{n+1/2,\mathrm{D}} = \boldsymbol{\xi}_{i,j,k}^{n} + \left(\frac{\Delta y}{2} - \frac{\Delta t}{2} v_{i,j,k}^{n} \right) \frac{\partial \boldsymbol{\xi}}{\partial y}\bigg|_{i,j,k}^{n}, \tag{S5}$$

$$\bar{\boldsymbol{\xi}}_{i,j+1/2,k}^{n+1/2,\mathrm{U}} = \boldsymbol{\xi}_{i,j+1,k}^{n} - \left(\frac{\Delta y}{2} + \frac{\Delta t}{2} v_{i,j+1,k}^{n} \right) \frac{\partial \boldsymbol{\xi}}{\partial y}\bigg|_{i,j+1,k}^{n}, \tag{S5}$$

where the superscripts D and U denote down and up, respectively. We perform a similar Godunov upwinding procedure to select a normal advective velocity at each face, so that

$$\bar{v}_{\text{adv}}^{n+1/2} = \begin{cases} \bar{v}^{n+1/2,\text{D}} & \text{if } \bar{v}^{n+1/2,\text{D}} > 0 \text{ and } \bar{v}^{n+1/2,\text{D}} + \bar{v}^{n+1/2,\text{U}} > 0, \\ \bar{v}^{n+1/2,\text{U}} & \text{if } \bar{v}^{n+1/2,\text{U}} < 0 \text{ and } \bar{v}^{n+1/2,\text{D}} + \bar{v}^{n+1/2,\text{U}} < 0, \\ 0 & \text{otherwise,} \end{cases} \tag{S5}$$

where we have dropped the subscripts since all terms are evaluated at $i, j + 1/2, k$. Next, we select between Taylor expansions $\bar{\mathbf{u}}^{n+1/2,\text{D}}$ and $\bar{\mathbf{u}}^{n+1/2,\text{U}}$, as well as between expansions $\bar{\boldsymbol{\xi}}^{n+1/2,\text{D}}$ and $\bar{\boldsymbol{\xi}}^{n+1/2,\text{U}}$, based on $\bar{v}_{\text{adv}}^{n+1/2}$. For two generic vector quantities at the face denoted by $\bar{\mathbf{q}}^{n+1/2,\text{D}}$ and $\bar{\mathbf{q}}^{n+1/2,\text{U}}$, we define

$$\bar{\mathbf{q}}^{n+1/2} = \begin{cases} \bar{\mathbf{q}}^{n+1/2,\text{D}} & \text{if } \bar{v}_{\text{adv}}^{n+1/2} > 0, \\ \left(\bar{\mathbf{q}}^{n+1/2,\text{D}} + \bar{\mathbf{q}}^{n+1/2,\text{U}} \right)/2 & \text{if } \bar{v}_{\text{adv}}^{n+1/2} = 0, \\ \bar{\mathbf{q}}^{n+1/2,\text{U}} & \text{if } \bar{v}_{\text{adv}}^{n+1/2} < 0. \end{cases} \tag{S6}$$

Finally, we compute the tangential derivative terms,

$$\overline{\left(v \frac{\partial \mathbf{q}}{\partial y} \right)} \bigg|_{i,j,k}^{n+1/2} = \frac{\bar{v}_{i,j-1/2,k,\text{adv}}^{n+1/2} + \bar{v}_{i,j+1/2,k,\text{adv}}^{n+1/2}}{2} \frac{\bar{\mathbf{q}}_{i,j+1/2,k}^{n+1/2} - \bar{\mathbf{q}}_{i,j-1/2,k}^{n+1/2}}{\Delta y}. \tag{S6}$$

1.3 Contact with a wall

We use subscript α to represent walls of the rectangular domain, with $\alpha = 1, 2, 3, 4, 5, 6$ referring to walls with normal vectors $\mathbf{e}_x, -\mathbf{e}_x, \mathbf{e}_y, -\mathbf{e}_y, \mathbf{e}_z, -\mathbf{e}_z$, respectively. The inward unit normal at a wall is denoted by $\mathbf{n}_\alpha$. When a part of the solid body is within a threshold distance from a wall with $\mathbf{n}_\alpha$, we impose a repulsive acceleration, $\mathbf{a}_{\text{rep},\alpha}$, to the grid cells in the solid body that have breached the threshold, acting at the cell centers. This acceleration is multiplied by a transition function in a small transition zone of width Δh_α, where Δh_α is the grid spacing in the Cartesian direction aligned with $\mathbf{n}_\alpha$.

For those solid grid cells that experience wall acceleration,

$$\mathbf{a}_{\text{rep},\alpha} = \begin{cases} A_{\text{rep}} \mathbf{n}_\alpha & \text{if } \phi \leq -\Delta h_\alpha, \\ \left(\frac{1}{2} - \frac{\phi}{2\Delta h_\alpha} \right) A_{\text{rep}} \mathbf{n}_\alpha & \text{if } |\phi| < \Delta h_\alpha, \\ 0 & \text{if } \phi \geq \Delta h_\alpha. \end{cases} \tag{S6}$$

If a solid body is in close approach to more than one wall, the repulsive accelerations are added together. As the solid approaches the wall and deforms to form a contact area, a small region of cells experiences the repulsive acceleration described by Eq. (S6).

We use the following approach to choose an appropriate value for A_{rep}. First, given the velocity of an object at its center of mass, the repulsive acceleration should be sufficient to stop the object from going through any physical boundary. In a rigid body, this means that the center of mass velocity is zero before or when the edge of the solid reaches the boundary. Though the solid objects will deform in our simulations, the rigid body case offers good guiding intuition. Second, we impose the condition that the repulsive acceleration, at a minimum, should be able to support an object resting against a wall.

We first define a critical threshold, d_w. When the distance between any cell belonging to the solid object and the wall falls below d_w, the wall repulsive acceleration comes into effect. In practice, we set $d_w = \Delta h_\alpha$ so that the solid objects are sufficiently close to the wall and experience effects due to boundaries, however, d_w can be increased so that the repulsive acceleration can be applied to more grid cells.

Let us consider an object of volume V approaches the wall with a normal velocity component U, which has a momentum along the normal direction, $\rho_s V U$. Suppose the volume of the object that actually experiences repulsive acceleration is V_{rep}, the effective acceleration at the center of mass is $\frac{V_{\text{rep}} A_{\text{rep}}}{V}$. Following the reasoning of the first case above,

$$U - \frac{V_{\text{rep}} A_{\text{rep}}}{V} t_s = 0,$$
$$U t_s - \frac{1}{2} \frac{V_{\text{rep}} A_{\text{rep}}}{V} t_s^2 = d_w. \tag{S6}$$

Solving these equations yields $A_{\text{rep}} = \frac{V U^2}{2 d_w V_{\text{rep}}}$ and $t_s = 2 d_w / U$.

We use the smallest length scale of the solid, $L_{\min}$, to obtain a conservative estimate of the contact area between the solid and the wall, and use d_w as the thickness of the region of cells that breach this threshold. Thus a conservative estimate of V_{rep} is $V_{\text{rep}} = d_w L_{\min}^2$, which yields

$$A_{\text{rep}} = \frac{VU^2}{2d_w^2 L_{\min}^2}. \tag{S6}$$

In the second case described above, when the object is at rest at a wall, we need to make sure that the forces experienced by the region of cells that have breached the d_w threshold is large enough to support the entire object, so that $|\rho_s - \rho_f| V g = \rho_s V_{\text{rep}} A_{\text{rep}}$ and hence

$$A_{\text{rep}} = \frac{|\rho_s - \rho_f| V g}{\rho_s V_{\text{rep}}}. \tag{S7}$$

In practice, we also find that it is helpful to set a minimum value of $A_{\text{re}} = U_{\text{safe}}^2/d_w$, $U_{\text{safe}} \approx 4.5$, as an additional safety measure to prevent solid objects from moving through physical boundaries. Given this lower limit and Eqs. (S6), (S7), we choose the maximum value among the three to be the value of A_{rep},

$$A_{\text{rep}} = \max\left(U_{\text{safe}}^2/d_w, \frac{VU^2}{2d_w^2 L_{\min}^2}, \frac{|\rho_s - \rho_f| V g}{\rho_s V_{\text{rep}}} \right). \tag{S7}$$

For simulations with multiple objects, A_{rep} is computed for each object separately.

1.4 Weighting scheme for weighted least squares-based extrapolation

In the absence of the reinitialized level-set function, we use an exponential decay kernel centered at the extrapolated cell to weigh data points in the linear squares regression. Near the fluid–solid interface we incorporate approximate geometric information via $\phi_0(\boldsymbol{\xi})$.

Consider a cell (p, q, r) which provides data $\boldsymbol{\xi}_d$ at position $\mathbf{x}_d = (x_p, x_q, x_r)$ in the weighted least squares linear regression to extrapolate to $\boldsymbol{\xi}_e$ at cell (i, j, k) with position $\mathbf{x}_e = (x_i, x_j, x_k)$. First, we compute the normalized gradient vector $\mathbf{n}_e$ of $\phi_0(\boldsymbol{\xi}^n)$ at cell (i, j, k) as well as a physical vector $\mathbf{x}' = \mathbf{x}_e - \mathbf{x}_d$. Then, depending on which layer extrapolation $\boldsymbol{\xi}_e$ resides in, the weight of data provided by cell (p, q, r) is defined as

$$\omega = \begin{cases} \max\left(0, \frac{\mathbf{x}' \cdot \mathbf{n}_e}{|\mathbf{x}'|} 2^{-(r_i + r_j + r_k)} \right) & \text{if } l_{i,j,k} \leq 2, \\ 2^{-(r_i + r_j + r_k)} & \text{if } l_{i,j,k} > 2, \end{cases} \tag{S7}$$

where $r_i = |i - p|, r_j = |j - q|, r_k = |k - r|$ and $l_{p,q,r}$ is the index of the extrapolation layer as defined in the main text. The exponential kernel accounts for locality on the Cartesian grid, while $\mathbf{n}_e$ approximates a surface normal and accounts for geometric information near the interface. However, we only make use of $\mathbf{n}_e$ in the extrapolating to the first two layers, since $\phi(\boldsymbol{\xi}^n)$ approximates a signed-distance function more poorly farther away from the interface in the presence of deformation.

1.5 Configuring a simulation

Before commencing a simulation, we need expressions of level-set functions that represent the fluid–solid interface. Each solid body is associated with such as level-set function. In addition, we need to configure the simulation domain, and prescribe material properties, boundary conditions, and initial conditions. Besides physics-related parameters, we also need to specify the type of data to collect and the frequency of outputting data files and checkpoint files. Main parameters that specify the physics in a simulation are documented in Table S1.

Initialization

1. Set initial conditions for velocity $\mathbf{u}_{i,j,k}^0$, and for reference map $\boldsymbol{\xi}_{i,j,k}^0$ for all i, j, k such that $\phi_0(\boldsymbol{\xi}_{i,j,k}^0) \leq 0$. We refer to these reference map variables as primary reference maps (PRMs). Using the undeformed state as the reference state of a solid body, the primary reference map variable is set to be the physical coordinate on the grid, $\boldsymbol{\xi}_{i,j,k}^0 = \mathbf{x}_{i,j,k}$, where $\mathbf{x} = (x_i, y_j, z_k)$.

2. If a test case has known pressure, also set $p_{i,j,k}^{-1/2}$, otherwise, we may set pressure to 0 everywhere. However, since pressure is computed as an auxiliary field rather than from an equation of state, the initial pressure profile is not known in general. Therefore, $p_{i,j,k}^{-1/2} = 0$ is generally not compatible with the initial velocity. The incompatibility can degrade the accuracy of the solution.

   To address this, we perform initial iterations to estimate $p_{i,j,k}^{-1/2}$. We run the timestepping algorithm, described in subsection 1.6, with an initial guess of $p_{i,j,k}^{-1/2} = 0$. After an iteration, we keep the pressure estimate $\tilde{p}_{i,j,k}^{1/2}$ but discard changes to all other fields. In the next iteration, let $p_{i,j,k}^{-1/2} = \tilde{p}_{i,j,k}^{1/2}$, and repeat until a tolerance on the divergence of $\mathbf{u}_{i,j,k}^0$ is reached, or until a maximum number of iterations is reached. Note that in order to carry out initial iterations, the variant of approximate projection method that computes the pressure increments should be used [16].

3. If applicable, initialize $\mathbf{b}_{i,j,k}^0$.

4. Extrapolate reference map variables to the blur zones. Extrapolated reference maps (ERMs) and the PRMs (together denoted as $\boldsymbol{\xi}_{i,j,k}^0$) are both required to begin the simulation.

5. Evaluate $\phi_0(\boldsymbol{\xi}^0)$ everywhere reference map variables $\boldsymbol{\xi}_{i,j,k}^0$ exist.

6. Initialize $\rho_{i,j,k}^0$ by blending ρ_f and ρ_s using using $H_\epsilon(\phi_0(\boldsymbol{\xi}_{i,j,k}^0))$ (Eq. (S3)).

1.6 Timestepping algorithm

After initializing the simulation, we perform the following steps for timestep $n = 0, 1, 2, \ldots, U$ of size Δt. This procedure is also used in the iterations to create an initial pressure profile.

1. Compute $\boldsymbol{\tau}_f^n$, $\boldsymbol{\tau}_s^n$, and $\boldsymbol{\tau}_a^n$, and mix them to get $\boldsymbol{\tau}^n$, as described in subsection 1.1.

2. Using the Godunov-type upwinding scheme described in subsection 1.2, we compute $\mathbf{u}_{i\pm1/2,j,k}^{n+1/2}$, $\mathbf{u}_{i,j\pm1/2,k}^{n+1/2}$, $\mathbf{u}_{i,j,k\pm1/2}^{n+1/2}$ everywhere, and $\boldsymbol{\xi}_{i\pm1/2,j,k}^{n+1/2}$, $\boldsymbol{\xi}_{i,j\pm1/2,k}^{n+1/2}$, $\boldsymbol{\xi}_{i,j,k\pm1/2}^{n+1/2}$ only within solids.

3. Perform the MAC projection, then update the normal face velocities. Any boundary conditions should be enforced before the projection step and after the velocity update.

4. Compute the advective term $[(\mathbf{u} \cdot \nabla)\mathbf{u}]_{i,j,k}^{n+1/2}$ and $[(\mathbf{u} \cdot \nabla)\boldsymbol{\xi}]_{i,j,k}^{n+1/2}$ according to Eqs. (S3).

5. Compute the predictor values of PRMs, i.e. $\boldsymbol{\xi}_{i,j,k}^{n+1}$ within solids (Eq. (S2)).

6. Extrapolate PRMs to yield ERMs to cover the blur zones, these are the predictor values of ERMs.

7. Apply corrector to all available reference maps (PRMs and ERMs), $\boldsymbol{\xi}_{i,j,k}^{n+1/2} = \frac{1}{2}\left(\boldsymbol{\xi}_{i,j,k}^n + \boldsymbol{\xi}_{i,j,k}^{n+1}\right)$.

8. Compute solid stress $\boldsymbol{\tau}_s^{n+1/2}$ and artificial viscous stress $\boldsymbol{\tau}_a^{\tilde{n}}$, then mix with $\boldsymbol{\tau}_f^n$ to create $\boldsymbol{\tau}^{n+1/2}$, as described in subsection 1.1.

9. Update density to $\rho_{i,j,k}^{n+1/2}$ using $H_\epsilon(\phi_0(\boldsymbol{\xi}_{i,j,k}^{n+1/2}))$ (Eq. (S3)).

10. Compute the intermediate velocity for approximate projection, $\mathbf{u}_{i,j,k}^*$, by

$$\mathbf{u}_{i,j,k}^* = \mathbf{u}_{i,j,k}^n - \Delta t[(\mathbf{u} \cdot \nabla)\mathbf{u}]_{i,j,k}^{n+1/2} - \frac{1}{\rho_{i,j,k}^{n+1/2}}\left(\nabla \cdot \boldsymbol{\tau}^{n+1/2} - \nabla p_{i,j,k}^{n-1/2} + \mathbf{b}_{i,j,k}^{n+1/2}\right) \tag{S8}$$

11. Perform the approximate projection, then update cell center velocities to $\mathbf{u}_{i,j,k}^{n+1}$ and nodal pressure to $p_{i,j,k}^{n+1/2}$. Any boundary conditions should be enforced before the projection step and after updating velocity and pressure.

12. Restore all PRMs and ERMs to their predictor values, $\boldsymbol{\xi}_{i,j,k}^{n+1}$.

13. Update the density to $\rho_{i,j,k}^{n+1}$ using $H_\epsilon(\phi_0(\boldsymbol{\xi}_{i,j,k}^{n+1/2}))$ (Eq. (S3)).

2 Test cases

In this section, we first present simulations of a pre-stretched, immersed sphere to demonstrate that the convergence of our method, as well as the scaling behavior of the simulated FSI problem, are as expected. We also provide additional details and results of examples mentioned in the main text.

2.1 Pre-stretched sphere

We simulate a viscoelastic neo-Hookean solid sphere, subject to initial strain, relaxing to equilibrium in a cubic periodic domain. Parameters reported here are dimensionless.

1. The sphere has a radius $R = 0.2$ and a shear modulus $G = 1$, positioned at the center of a periodic box with unit length, fluid viscosity $\mu = 0.01$. The sphere has the same density as the fluid, and is pre-strained at $T = 0$ in the x-direction by a stretch $\lambda = 1.21$. We use isotropic grid spacing $\Delta x = \Delta y = \Delta z = h$. The solution with the smallest h is used as the reference solution in the convergence plot.

   Elastic energy and volume of the sphere over time are reported in Fig. S2(a), and the convergence in surface area, volume, and $\|J - 1\|_2$ is reported in Fig. S2(b), where J is the determinant of the deformation gradient. The pressure-Poisson projection method we use to enforce incompressibility is an approximate one, which means locally, it is possible to have $J \neq 1$. To account for the local violation of incompressibility, we compute strain energy density with a logarithmic correction factor [17],

$$\Phi_{\text{elas}} = \frac{1}{2} G \big(\operatorname{tr}(\mathbf{C}) - 3 - 2 \ln(J) \big) \tag{S9}$$

   where $\mathbf{C} = \mathbf{F}^{\mathsf{T}} \mathbf{F}$ is the right Cauchy–Green tensor. This strain energy density equation is used throughout our analysis.

   Since we use a reference solution rather than the exact solution, only part of the error can be captured by a Richardson error model [18, 19, 20, 11]. For other sources of error, e.g. reference map extrapolations, Godunov-type upwinding procedures, Richardson model is not a good fit. Thus, we adopt the 3-parameter error model proposed by Rycroft *et al.* [11]

$$E(h) = B(h^s - \alpha h_{\text{ref}}^s) + \mathcal{O}(h^{s+1}) \tag{S9}$$

   where α is the Richardson correction factor which indicates the proportion in the error that can be captured by a Richardson error model, and s is the overall convergence rate. We report the fitted convergence parameters in the caption of Fig. S2.

2. We demonstrate scaling of elastic energy of the immersed sphere in various combinations of G, μ, ρ_s / ρ_f, and R (Fig. S3). First, we find the pertinent parameter in our test case by dimensional analysis. Consider a standalone, ideal, incompressible neo-Hookean elastic body with density $\rho_s = \rho$, shear modulus G, and a characteristic length scale R. In the absence of viscous damping, uniaxial tension or compression induces an oscillatory response due to elastic restoring forces in the solid body. The oscillation time scale is $\tau_{\text{osc}} = R\sqrt{\rho/G}$. In addition to being the characteristic oscillation period, this time scale can also be interpreted as the time it takes for elastic waves to traverse the elastic body. Now, suppose the elastic body is immersed in a viscous fluid, with viscosity μ and density $\rho_f = \rho$. In our method, the solid material is made viscoelastic by using a constant artificial viscosity μ_a. We impose $\mu_a = \mu$ in this case. Out of the parameters relevant to the fluid ρ, μ, R, we can build another time scale, $\tau_{\text{diss}} = \rho R^2 / \mu$. We can interpret τ_{diss} as the time scale of the fluid dissipating the kinetic energy generated by the oscillating sphere as it returns to equilibrium.

   The ratio between these two time scales is

$$\gamma = \tau_{\text{diss}} / \tau_{\text{osc}} = R\sqrt{\rho G}/\mu.$$

In the simple system we have considered here, having assumed solid and fluid have identical densities ρ, and viscosity $\mu_a = \mu$, we have four dimensional parameters μ, G, ρ, R. By the Buckingham Π Theorem, γ is the only dimensionless parameter in this system. It can be interpreted as the number of oscillations a viscoelastic sphere undergoes before returning to the equilibrium. Incidentally, we can also consider the relaxation time scale of the viscoelastic solid, $\tau_{\text{rel}} = \mu/G$. Taking the ratio $\tau_{\text{osc}} / \tau_{\text{rel}}$ also yield the dimensionless parameter γ.

We carry out a series of simulations (parameters documented in Table S2) to demonstrate the scaling behavior in this system. In Fig. S3(a), we show solid elastic energy from various simulations normalized by its initial value E_0. Values of G, R, and μ are varied across simulations, but dimensionless parameter γ is kept the same. The number of visible oscillations before energy is fully dissipated by the fluid is the same across these simulations. After rescaling time by τ_{diss}, we also find that peaks and valleys of the elastic energy fall at similar places on the rescaled time axis. Furthermore, given the same γ, if the sphere radius is kept the same, which implies $\sqrt{G}/\mu$ is constant, elastic energy curves collapse onto the same one (simulation A_1 and A_2).

In Fig. S3(b), we show the number of visible oscillations is qualitatively linearly proportional to γ. Simulation B_1, which has $\gamma = 10$, is compared to simulation B_2, A_1, and B_3, which have have $\gamma = 15.8, 20, 30$, respectively. The orange dashed line indicates approximately one oscillation period in simulation B_1 after initial release. Within the same rescaled time (in units of respective τ_{diss}), we observe about 1.5, 2, and 3 times more periods in simulation B_2, A_1, and B_3, respectively. The increase in the number of periods follows the increase in γ approximately linearly.

If the solid has a different density from fluid, the matter becomes more complex. Though the oscillation time scale τ_{osc} should only depend on ρ_s, the energy dissipation, influenced by dynamics in both the solid and the fluid phases, now depends on both ρ_s and ρ_f. We demonstrate this effect in Fig. S3(c). In simulation C_1 and C_2, we have chosen to use ρ_s in calculating γ and τ_{diss}. While the number of visible oscillations remains similar among simulations A_1, C_1, and C_2, which have identical γ, rescaling time with τ_{diss} no longer lead to the alignment of energy peaks and valleys on the rescaled time axis across simulations.

2.2 Settling

We provide `si_movie_1.mov` Settling of 150 ellipsoids, described in main text Fig. 3.

2.3 Lid-driven Cavity

Here we provide results in comparing lid-driven cavity flow simulated with RMT3D without any solid objects with 3D benchmarks as a validation test for our Navier–Stokes solver code. In addition, we present convergence tests and centroid positions data of an immersed sphere in cubic lid-driven cavity flow. Convergence rates are computed using Eq. (S9) and reported in the figure caption when applicable. Finally, we provide additional simulation parameters used in Fig. 4 in the main text, and include movies mentioned there.

1. Comparison of normal velocities and pressure profiles in a cubic lid-driven cavity with Reynolds number $Re = 1000$ against benchmark results is shown in Fig. S4; convergence of normal velocity extrema and steady state kinetic energy is shown in Fig. S5.

2. Comparison of normal velocities and pressure profiles in a cubic lid-driven cavity with Reynolds number $Re = 100, 400$ against benchmark results is shown in Fig. S6.

3. Comparisons of normal velocities and pressure profiles in a lid-driven cavity of various aspect ratios with $Re = 1000$ against benchmark results are shown in Figs. S7 & S8.

4. Convergence in surface area, volume, elastic energy, and $\|J - 1\|_2$, where J is the determinant of the deformation gradient, for a sphere with $G = 0.1$ in a cubic lid-driven cavity with $Re = 100$ is shown in Fig. S9.

5. Additional parameters for simulations presented in Fig. 4 in the main text are documented in Table S3.

6. `trajectories.txt` trajectories.txtTrajectories of the centroid of a sphere with $G = 0.1, 0.25, 0.5$ in a cubic lid-driven cavity with $Re = 100$. Isotropic grid spacing $h = 1/160$.

7. `si_movie_2.mov` A sphere of radius 0.2 in unit cubic lid-driven cavity with $Re = 100$ and solid shear modulus $G = 0.03$, described in main text Fig. 4(a).

8. `si_movie_3.mov` A sphere of radius 0.2 in unit cubic lid-driven cavity with $Re = 100$ and solid shear modulus $G = 0.1$, described in main text Fig. 4(b)&(c).

9. `si_movie_4.mov` A sphere of radius 0.2 in unit cubic lid-driven cavity with $Re = 100$ and solid shear modulus $G = 0.25$, described in main text Fig. 4(d)&(e).

10. `si_movie_5.mov` A sphere of radius 0.2 in unit cubic lid-driven cavity with $Re = 100$ and solid shear modulus $G = 0.5$, described in main text Fig. 4(f)&(g).

2.4 Swimming

In this section, we expand on some of the contexts and results from the section in the main text on simulating swimming with the RMT.

Amplitude parameter

The amplitude parameter $W = B/3GIk^2$ is derived from scaling arguments and linear bending theory. Here, we explicitly list the relationships and scalings between bending moment amplitude B, displacement scaling constant W, and the vertical linear stress density σ_0.

In Euler beam theory, the stress density is related to the bending moment across a cross-section as $\sigma_0 \sim B/I$, where I is the area moment of inertia of the cross-section. There is a similar relationship to the beam curvature $\kappa \sim B/3GI$, where $3GI$ is the beam bending modulus for a shear modulus G. Given a vertical displacement scale W and length scale k^{-1}, the curvature also satisfies $\kappa \sim Wk^2$ for $Wk \ll 1$. Thus, $W \sim B/3GIk^2$ and $\sigma_0 \sim 3WGk^2$, motivating the definitions in the main text.

Swimming statistics

For each simulation, we calculate a swim speed U, active power P, drag coefficient C, and swimming efficiency e. Here, we detail the form of each and describe its calculation.

We let $\langle \psi \rangle$ describe the time-average and $\hat{\psi}$ the oscillation magnitude of a time-dependent value $\psi(t)$. To calculate these quantities, time traces $\psi(t)$ are fit to a function

$$f(t) = c_0 + c_1 t + c_2 t^2 + \sum_{j=1}^{2} \left[A_j \cos(2\pi j t) + B_j \sin(2\pi j t) \right]. \tag{S11}$$

Then $\langle \psi \rangle = c_0$, $\hat{\psi} = \sqrt{A_1^2 + B_1^2}$.

- The swim speed $U = \left\langle u_x^{(c)} \right\rangle$, where $\mathbf{u}^{(c)}$ is centroid velocity and the subscript x denotes the direction of swimming

- The power $P = \left\langle - \int_{\Omega_s} \boldsymbol{\sigma}^{(a)} : \nabla \mathbf{u} dV \right\rangle$ where $\boldsymbol{\sigma}^{(a)}$ is the active part of the solid stress

- The drag coefficient C is computed for each combination of swimmer body radius R and length L. A simulation is conducted for each body shape with no active stress subject to a density ratio $\rho_s/\rho_f = 2$ and gravitational acceleration $g = 1/10$ in the swimming direction. The centroid velocity $u_x^{(c)}(t)$ is fit to the function

$$g(t) = U_f + (U_0 - U_f)e^{-\lambda t}, \tag{S11}$$

  for an initial and final velocity U_0 and U_f and a rate λ representing the speed with which the object approaches terminal velocity. The drag coefficient is then calculated as $C = (\rho_s - \rho_f)V_s g/U_f$, where V_s is the swimmer volume.

- The swimming efficiency is typically defined $e = FU/P$, where F is the force required to tow the swimmer at velocity U. In the main text, we substitute an approximate tow force $\bar{F} = CU$ representing a linear estimate.

Reynolds numbers

We describe two Reynolds numbers: one corresponding to the steady flow induced by time-averaged motion of the object through the fluid, and another describing the oscillatory flow driven by the object's cyclic deformation. Here we describe the form of both and report the calculated values.

- The steady Reynolds number $Re_s = \rho U R_h/\mu$ uses the calculated swim speed as a velocity scaling and the swimmer head radius as a length scale.

- The oscillatory Reynolds number $Re_o = \rho \hat{u}_z^{(c)} R / \mu$ uses $\hat{u}_z^{(c)}$—the magnitude of the oscillating vertical velocity—as a velocity scale and the swimmer radius as a length scale.

For each simulation reported in Fig. 5 in the main text, the values of the two Reynolds numbers are calculated and documented in Fig. S10.

Movie

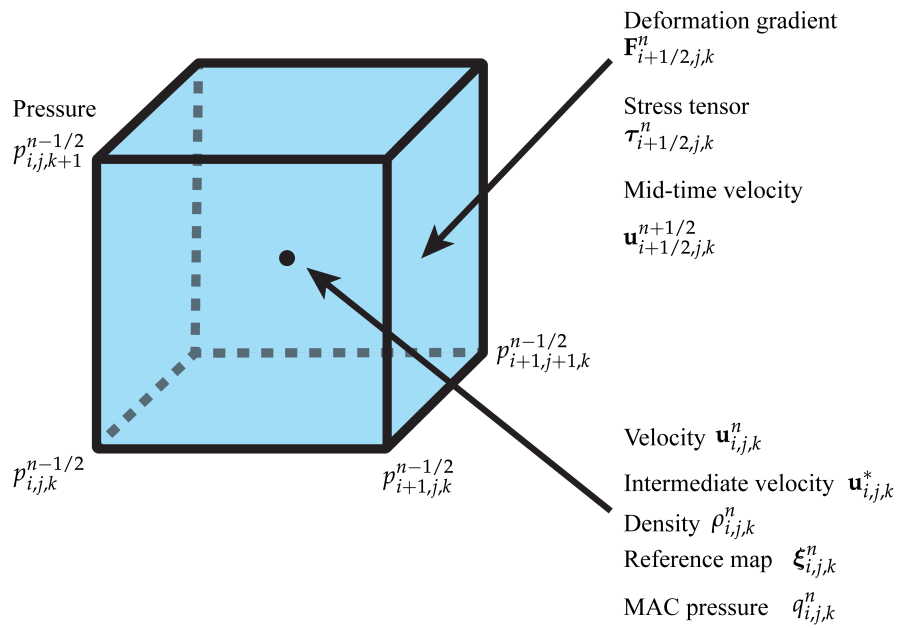Finally, we provide `si_movie_6.mov` Swimmers with various body geometries, actuated by active stress, described in main text Fig. 5.

Figure S1: Arrangement of variables in the three-dimensional reference map simulation. Depending on the finite difference scheme we use, variable can reside at cell centers, faces, or nodes.

Table S1: Main dimensionless simulation parameters in the RMT. In addition, Dirichlet and Neumann boundary conditions can be specified for each face of the rectangular simulation domain.

ρ_f	fluid density
μ	fluid viscosity
ρ_s	solid density
G	solid shear modulus
μ_a	solid artificial viscosity
γ_t	blur zone viscosity constant
ϵ	half blur zone width
L_x, L_y, L_z	simulation domain side lengths
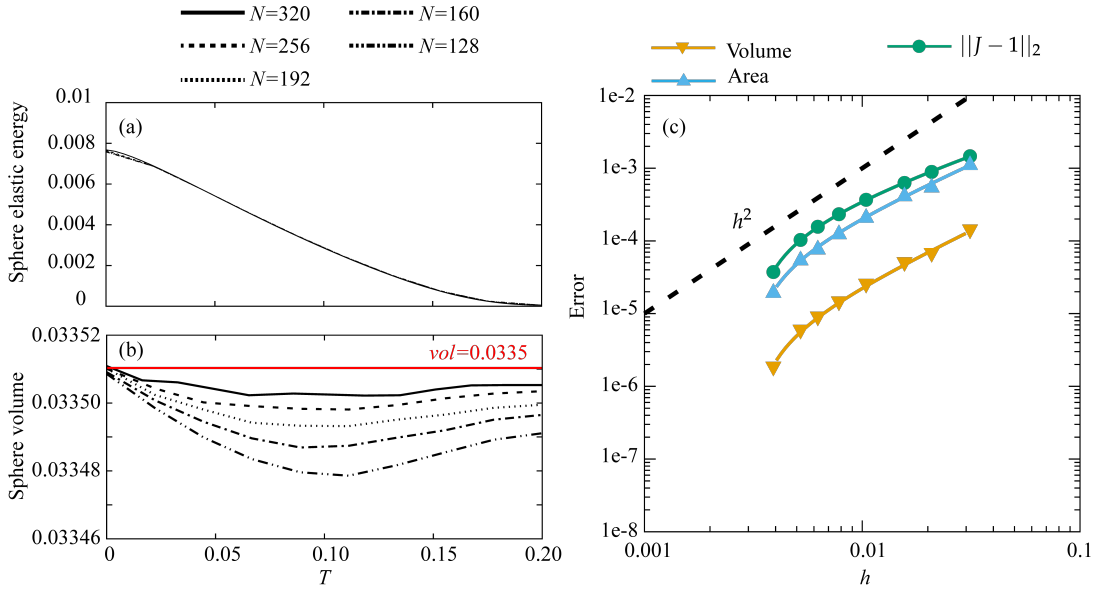
Figure S2: (a) Elastic energy and volume of a sphere with radius 0.2 and shear modulus $G = 1$ in a periodic unit box. The incompressible sphere is stretched in x-direction by a factor of 1.44, and compressed in the other directions accordingly. Fluid viscosity $\mu = 0.01$, and solid sphere has the same density as the fluid, $\rho_s = \rho_f = 1$. An incompressible neo-Hookean solid is modeled. However, $J \neq 1$ in our numerical scheme, especially near the fluid–solid interface. Therefore, a correction factor is used in the strain energy density (Eq. (S9)). The correction approaches zero as the grid is refined, since J converges to 1. (b) Convergence of errors in surface area, volume, elastic energy, and $\|J - 1\|_2$, where J is the determinant of the deformation gradient. Errors are measured at $T = 0.2$. The solid lines are fitted curves using Eq. (S9). Convergence rates (with Richardson correction factor α in brackets) are $1.34(0.97), 1.44(1.0),$ and $1.04(1.0)$, for area, volume, and $\|J - 1\|_2$, respectively.

Table S2: Dimensionless simulation parameters for pre-stretched sphere scaling test. Fluid density ρ_f is always kept at unity. Solid artificial viscosity μ_a is identical to fluid viscosity μ.

Case	μ	ρ_s/ρ_f	R	G	γ	τ_{diss}
A_1	0.005	1	0.2	0.25	20.0	8
A_2	0.01	1	0.2	1	20.0	4
A_3	0.005	1	0.1	1	20.0	2
A_4	0.0075	1	0.3	0.25	20.0	12
B_1	0.005	1	0.1	0.25	10.0	2
B_2	0.02	1	0.2	2.5	15.8	2
B_3	0.005	1	0.3	0.25	30.0	18
C_1	0.005	0.5625	$\frac{4}{15}$	0.25	20.0	8
C_2	0.005	2.25	$\frac{2}{15}$	0.25	20.0	8
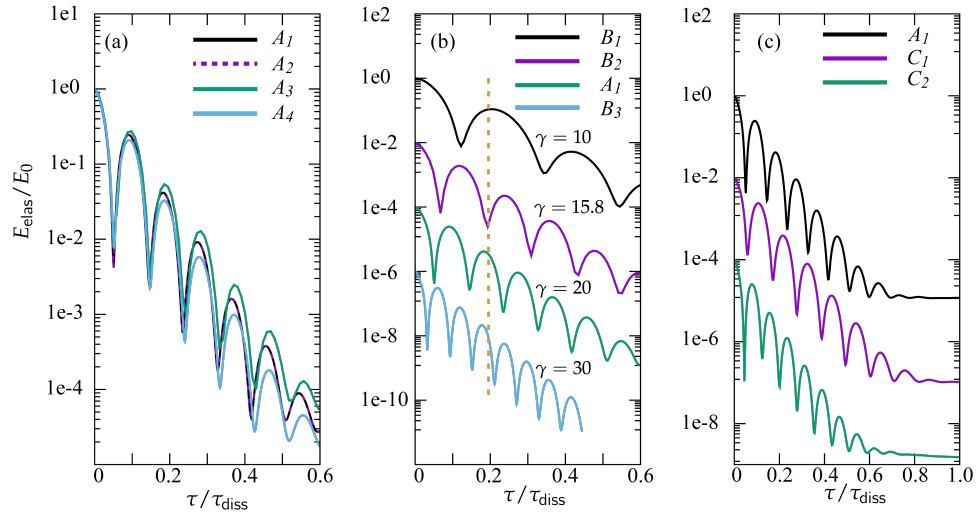
Figure S3: Elastic energy (rescaled by respective initial value E_0) over time (rescaled by respective characteristic time τ_{diss}) of pre-stretched spheres. Simulation parameters are documented in Table S2. In (b), normalized elastic energies are further reduced by a factor of 100, 10^4, 10^6 for simulation B_2, A_1, B_3 so that the curves are clearly separated. In (c), similar reduction factors of 100 and 10^4 are applied to curves from simulation C_1 and C_2 for clarity.
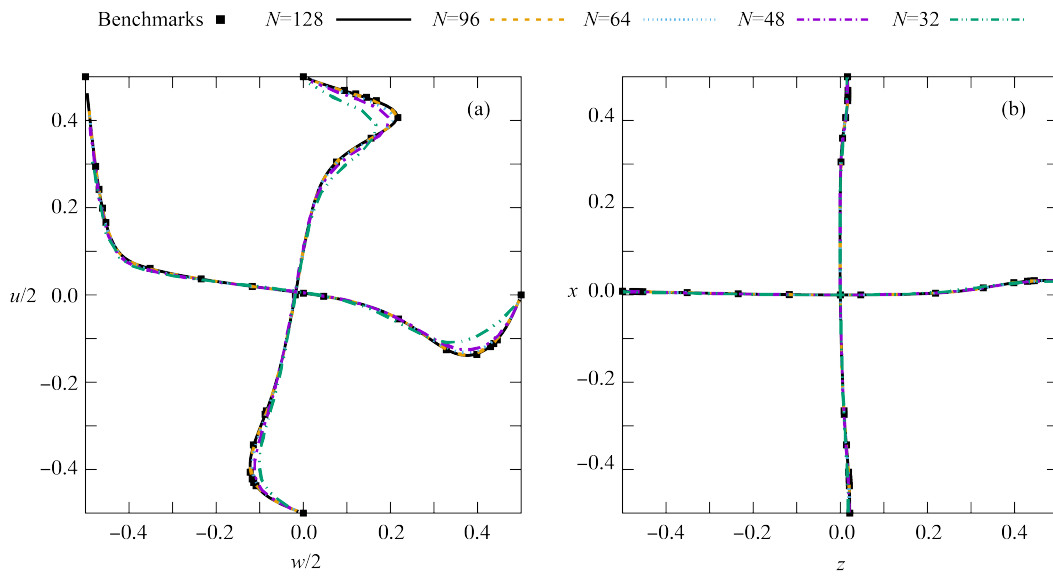
Figure S4: Comparing normal velocities and pressure profiles against benchmark results by Albensoeder and Kuhlmann [21] for cubic lid-driven cavity with Reynolds number $Re = 1000$. Spatial coordinates are shifted by 0.5 in both plots such that the domain is $[-0.5, 0.5] \times [-0.5, 0.5]$, and rotated to be aligned with figures in the work by Albensoeder and Kuhlmann [21]. (a) Normal velocities along the center lines in the central xz-plane are rescaled and plotted for five simulation resolutions, $N = 32, 48, 64, 96, 128$. (b) Pressure profiles along the center lines in the central xz-plane are plotted. Pressure values are shifted by a constant such that it is kept at zero at position $(0.0, 0.0)$.
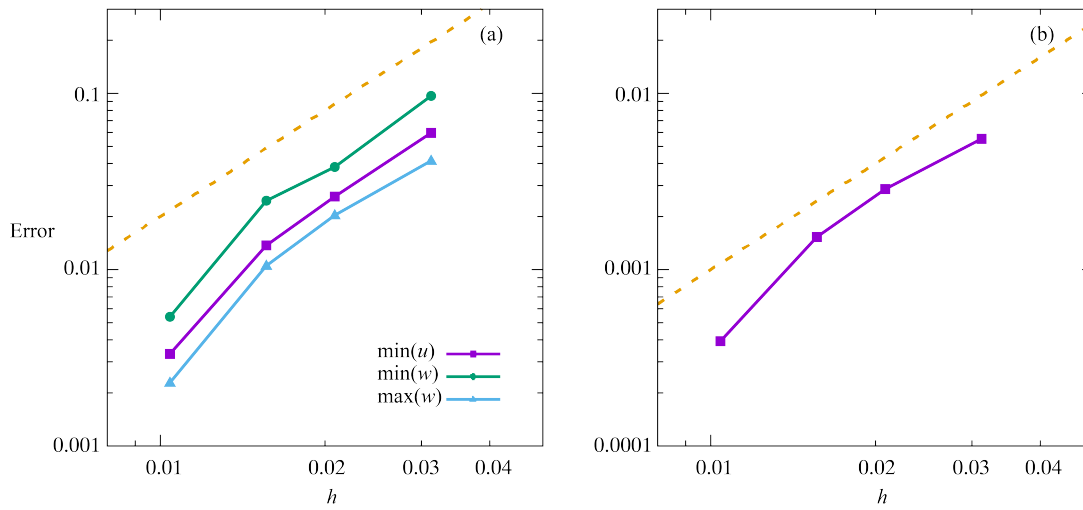
Figure S5: Error convergence in a cubic lid-driven cavity flow with $Re = 1000$, using $N = 128$ solution of as the reference solution. (a) Error in the extrema of normal velocities, $\min(u)$, $\min(w)$, and $\max(w)$, along center lines in the central xz-plane. (b) Error in the total kinetic energy after steady state has been reached.
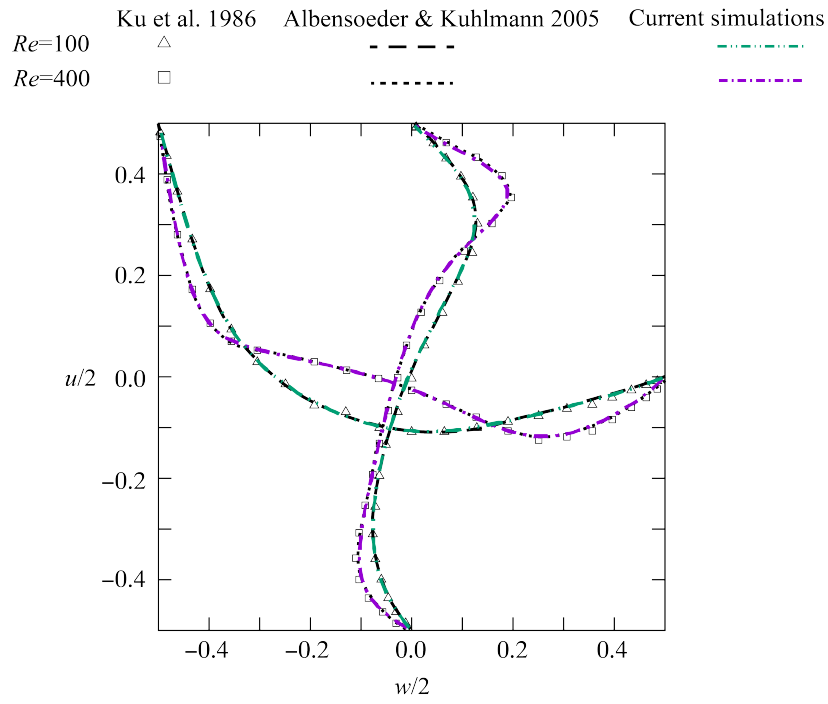
Figure S6: Comparing normal velocities against benchmark results by Albensoeder and Kuhlmann [21] for cubic lid-driven cavity with Reynolds number $Re = 100, 400$. Spatial coordinates are shifted by 0.5 in both plots such that the domain is $[-0.5, 0.5] \times [-0.5, 0.5]$, and rotated to be aligned with figures in the work by Albensoeder and Kuhlmann [21]. Normal velocities along the center lines in the central xz-plane are rescaled and plotted for resolution $N = 96$.
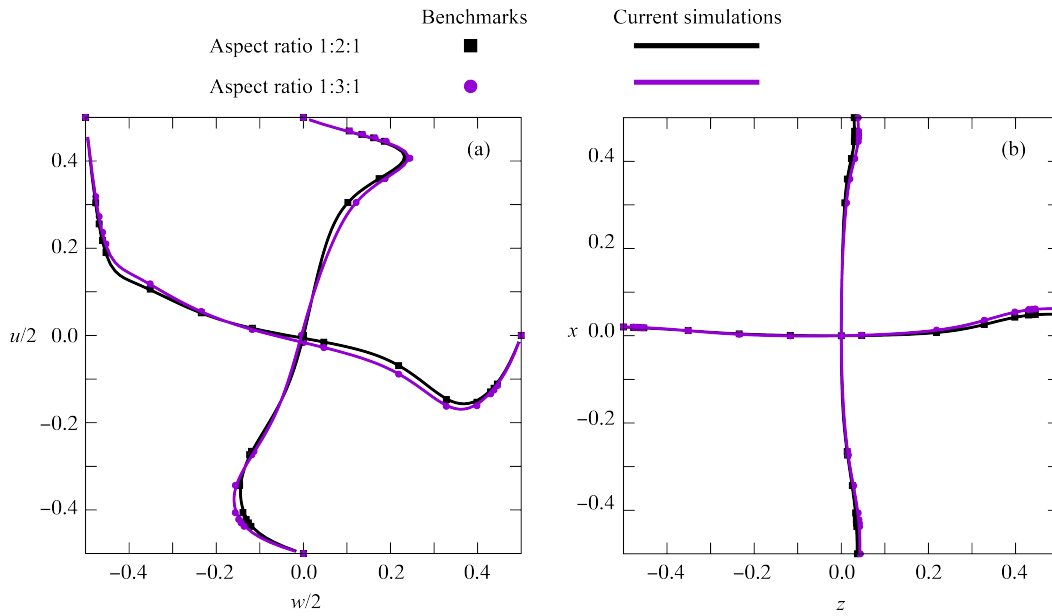
Figure S7: Comparing normal velocities and pressure profiles against benchmark results by Albensoeder and Kuhlmann [21] for lid-driven cavities with aspect ratios $1:2:1$ and $1:3:1$, with Reynolds number $Re = 1000$. Spatial coordinates are shifted by 0.5 in both plots such that the domain is $[-0.5, 0.5] \times [-0.5, 0.5]$, and rotated to be aligned with figures in the work by Albensoeder and Kuhlmann [21]. (a) Normal velocities along the center lines in the central xz-plane are rescaled and plotted for five simulation resolutions $N = 96$ per unit simulation length. (b) Pressure profiles along the center lines in the central xz-plane are plotted. Pressure values are shifted by a constant such that it is kept at zero at position $(0.0, 0.0)$.
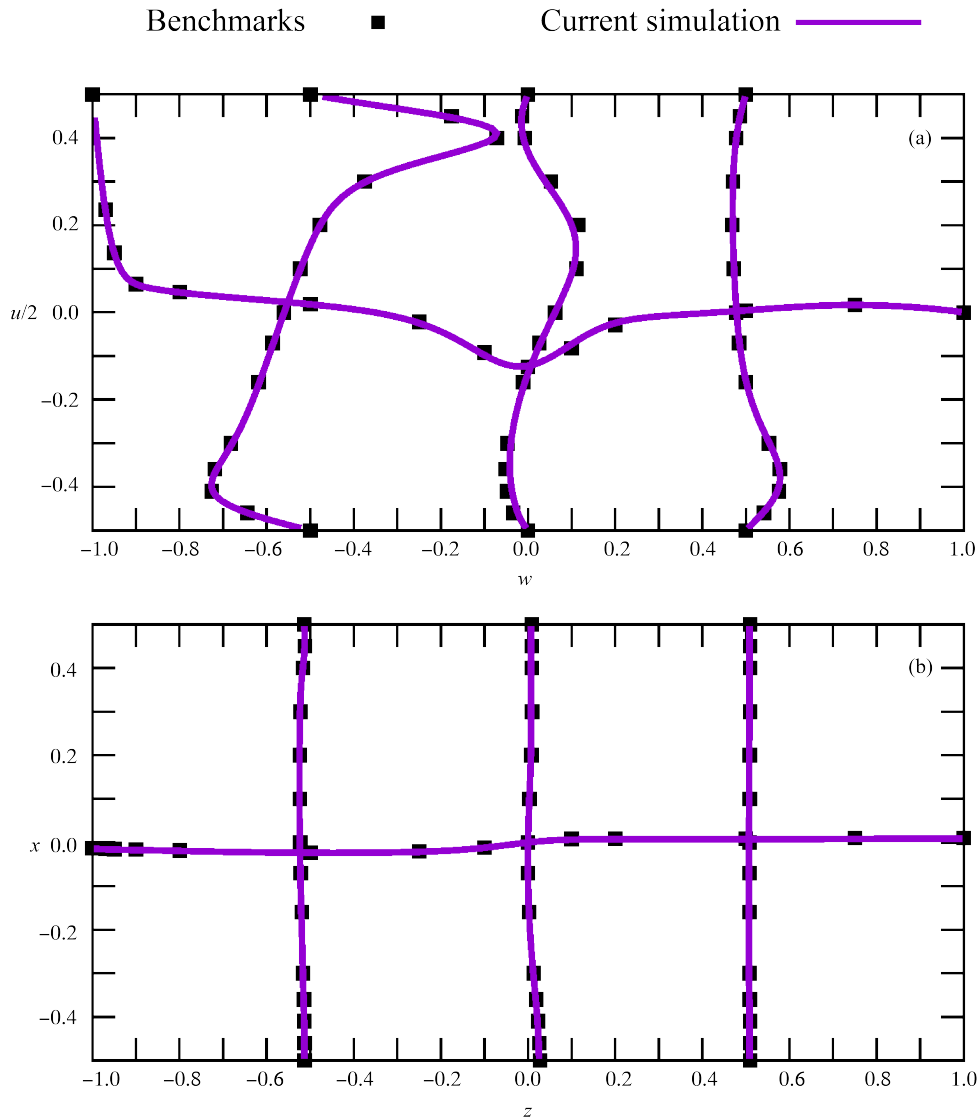
Figure S8: Comparing normal velocities and pressure profiles against benchmark results by Albensoeder and Kuhlmann [21] for lid-driven cavities with aspect ratio $1 : 1 : 2$, with Reynolds number $Re = 1000$. Spatial coordinates are shifted in both plots such that the domain is $[-0.5, 0.5] \times [-1.0, 1.0]$, and rotated to be aligned with figures in the work by Albensoeder and Kuhlmann [21]. (a) Normal velocities along the center lines in the central xz-plane are rescaled and plotted for five simulation resolutions $N = 96$ per unit simulation length. (b) Pressure profiles along the center lines in the central xz-plane are plotted. Pressure values are shifted by a constant such that it is kept at zero at position $(0.0, 0.0)$.
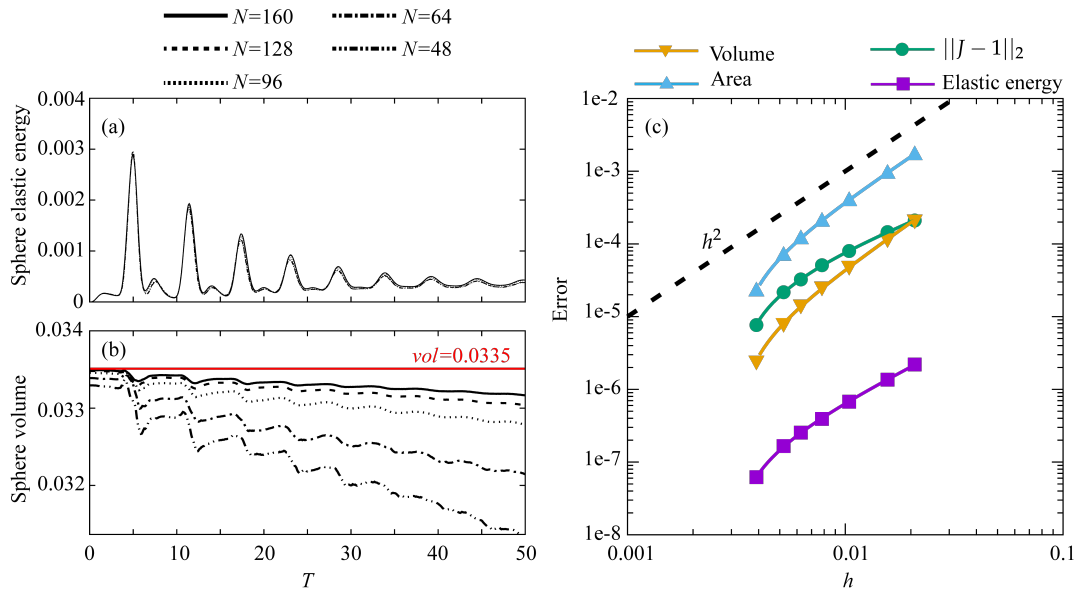
Figure S9: (a) Sphere elastic energy and volume over time for a sphere with radius 0.2 and shear modulus $G = 0.1$ in a cubic lid-driven cavity. Reynolds number $Re = 100$. An incompressible neo-Hookean solid is modeled. However, $J \neq 1$ in our numerical scheme, especially near the fluid–solid interface. Therefore, a correction factor is used in the strain energy density (Eq. (S9)). The correction approaches zero as the grid is refined, since J converges to 1. (b) Convergence of errors in surface area, volume, elastic energy, and $\|J - 1\|_2$, where J is the determinant of the deformation gradient. Errors are measured at $T = 0.5$. The solid lines are fitted curves using Eq. (S9). Convergence rates (with Richardson correction factor α in brackets) are $2.00(0.98), 2.08(1.0), 1.51(0.92),$ and $1.16(1.0)$ for surface area, volume, elastic energy, and $\|J - 1\|_2$, respectively.

Table S3: Additional parameters simulations presented in Fig. 4 in the main text. Number of grid cells N and blur zone viscosity multiplier γ_t are reported for each case in subfigure Fig. 4(a), (b), (d), (f), (h), and (j). Isotropic grid spacing $h = 1/N$.

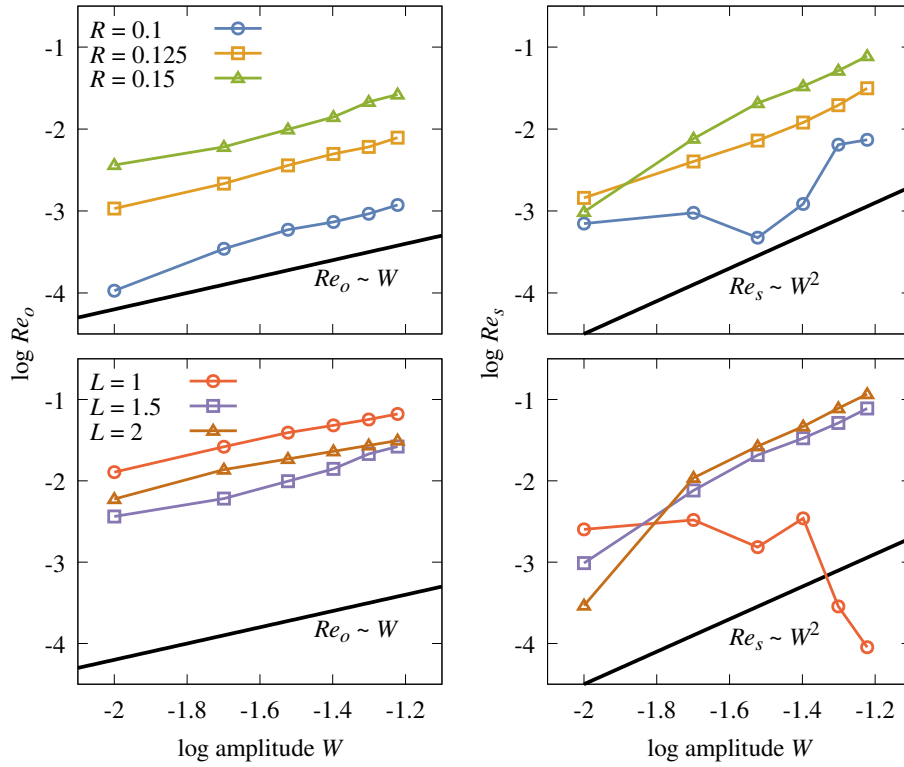| Case	N	γ_t
(a)	128	0
(b)	48	2
(b)	64	4
(b)	96	2
(b)	128	0
(b)	160	0
(d)	48	0
(d)	64	1
(d)	96	0
(d)	128	0
(d)	160	0
(f)	48	0
(f)	64	0
(f)	96	0
(f)	128	0
(f)	160	0
(h)	128	0
(j)	64	0

Figure S10: Reynolds numbers describing the oscillatory (Re_o, left column) and steady (Re_s, right column) flows about the swimmer. Top row: Swimmer body length is kept at $L = 1.5$ while its body radius R varies. Bottom row: Swimmer body radius is kept at $R = 0.15$ while its body length L varies. Other simulation parameters are reported in Fig. 5 caption in the main text.

Supplmentary Information References

[1] Alexandre J. Chorin. A numerical method for solving incompressible viscous flow problems. *J. Comput. Phys.*, 2(1):12–26, 1967.

[2] Alexandre J. Chorin. Numerical solution of the Navier–Stokes equations. *Math. Comput.*, 22(104):745–762, 1968.

[3] David L. Brown, Ricardo Cortez, and Michael L. Minion. Accurate projection methods for the incompressible Navier–Stokes equations. *J. Comput. Phys.*, 168(2):464–499, 2001.

[4] John B Bell, Phillip Colella, and Harland M Glaz. A second-order projection method for the incompressible Navier–Stokes equations. *Journal of Computational Physics*, 85(2):257–283, December 1989.

[5] Ann S. Almgren, John B. Bell, and William G. Szymczak. A numerical method for the incompressible Navier–Stokes equations based on an approximate projection. *SIAM J. Sci. Comput.*, 17(2):358–369, 1996.

[6] Ann S. Almgren, John B. Bell, Phillip Colella, Louis H. Howell, and Michael L. Welcome. A conservative adaptive projection method for the variable density incompressible Navier–Stokes equations. *J. Comput. Phys.*, 142(1):1–46, 1998.

[7] Phillip Colella. Multidimensional upwind methods for hyperbolic conservation laws. *J. Comput. Phys.*, 87(1):171–200, 1990.

[8] Mark Sussman, Ann S. Almgren, John B. Bell, Phillip Colella, Louis H. Howell, and Michael L. Welcome. An adaptive level set approach for incompressible two-phase flows. *J. Comput. Phys.*, 148(1):81–124, 1999.

[9] Jiun-Der Yu, Shinri Sakai, and James A. Sethian. A coupled level set projection method applied to ink jet simulation. *Interfaces and Free Boundaries*, 5(4):459–482, 2003.

[10] Jiun-Der Yu, Shinri Sakai, and James A. Sethian. Two-phase viscoelastic jetting. *J. Comput. Phys.*, 220(2):568–585, 2007.

[11] Chris H Rycroft, Chen-Hung Wu, Yue Yu, and Ken Kamrin. Reference map technique for incompressible fluid–structure interaction. *Journal of Fluid Mechanics*, 898, 2020.

[12] Ken Kamrin, Chris H. Rycroft, and Jean-Christophe Nave. Reference map technique for finite-strain elasticity and fluid–solid interaction. *J. Mech. Phys. Solids*, 60(11):1952–1969, 2012.

[13] Boris Valkov, Chris H. Rycroft, and Ken Kamrin. Eulerian method for multiphase interactions of soft solid bodies in fluids. *J. Appl. Mech.*, 82(4):041011, 04 2015.

[14] Mark Sussman, Peter Smereka, and Stanley Osher. A level set approach for computing solutions to incompressible two-phase flow. *J. Comput. Phys.*, 114(1):146–159, 1994.

[15] J. Crank and P. Nicolson. A practical method for numerical evaluation of solutions of partial differential equations of the heat-conduction type. *Mathematical Proceedings of the Cambridge Philosophical Society*, 43(1):50–67, 1947.

[16] Ann S. Almgren, John B. Bell, and William Y. Crutchfield. Approximate Projection Methods: Part I. Inviscid Analysis. *SIAM J. Sci. Comput.*, 22(4):1139–1159, January 2000.

[17] Paul J. Flory. *Principles of Polymer Chemistry*. Cornell University Press, Ithaca, N. Y., 1953.

[18] Lewis Fry Richardson. IX. The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Philos. Trans. Royal Soc. A*, 210(459–470):307–357, 1911.

[19] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I: Nonstiff Problems*. Springer, Berlin, 1993.

[20] Michael T. Heath. *Scientific Computing: An Introductory Survery*. McGraw-Hill, 2002.

[21] S. Albensoeder and H.C. Kuhlmann. Accurate three-dimensional lid-driven cavity flow. *Journal of Computational Physics*, 206(2):536–558, July 2005.