Communications in Applied Mathematics and Computational Science

msp.org/camcos

See inside back cover or msp.org/camcos for submission instructions.

msp

A FRONT-TRACKING SHOCK-CAPTURING METHOD FOR TWO GASES

MEHDI VAHAB AND GREGORY H. MILLER

We present a new high-order front-tracking method for hyperbolic systems of conservation laws for two gases separated by a tracked contact discontinuity, using a combination of a high-order Godunov algorithm and level set methods. Our approach discretizes the moving front and gas domains on a Cartesian grid, with control volumes determined by the intersection of the grid with the front. In cut cells, a combination of conservative and nonconservative finite volume quadratures provide small-cell stability. Global conservation is maintained using redistribution. We demonstrate second-order convergence in smooth flow and first-order convergence in the presence of shocks.

1. Introduction

The predictive simulation of two-fluid flows with sharp material interfaces is necessary to understand processes such as shock-induced mixing that are important in ballistic impact and inertial confinement processes. Numerical methods that address sharp interfaces using discrete step functions are zero-order accurate in capturing interface jumps and cannot detect jump-sensitive features such as the onset of instability [42]. To address such problems while properly resolving jumps we present a new 2-D front-tracking method for hyperbolic systems of conservation laws specialized to gas dynamics.

Any such method may be distinguished by its front representation and the way it handles gas volume geometric calculations (see [24] for an early review of front tracking approaches). One class of front-tracking methods is surface tracking [13; 17; 19]. While highly successful, complex schemes are involved for front entanglement and difficulties could arise for generalization to higher dimensions. For instance, the topology of the solution to Riemann problem is not known for the general case [32]. Volume-tracking methods introduce a simpler front representation, but lack subgrid scale resolution [22]. Interface reconstruction methods have been

1

used to extract geometrical information from volume-tracking methods [22; 41]. With the introduction of level set methods, new front-tracking schemes have been developed for high-order implicit front evolution [15; 44]. Our approach is based on a new interface reconstruction method that extracts accurate geometry information from a time sequence of discrete level set data [30; 36].

Depending on the representation of the front, finite volume and finite difference methods have developed different approaches to calculate state variables and fluxes near and on the front. Since the material interface moves, there may be small fractional cells near the interface. The small volume of those cells force severe limitations on unmodified finite volume and finite difference methods. Several approaches have been developed to resolve or circumvent this problem, such as cell merging [18], the h-box method [4], the ghost fluid method [15], and the single phase approximation [8; 38]. Another approach is the hybrid conservative method of Chern and Colella [7] and Bell et al. [2], which combines the conservative finite volume method with a nonconservative but stable update, and maintains global conservation using a redistribution algorithm. This idea has been successfully used for embedded boundary methods for static [11] and time-dependent [35; 36; 37] domains, and for a second-order conservative front-tracking method in one dimension [16].

We base our approach on the hybrid conservative finite volume method of Chern and Colella [7] and Pember et al. [45], which solves single phase dynamics using an unsplit Godunov method in irregular domains. The time-dependence of the domain uses space-time finite volume quadratures, generalizing the 1-D algorithm of Gatti-Bono et al. [16]. The geometric quantities that support these quadratures are derived from level sets using algorithms presented by Ligocki et al. [30] and Miller and Trebotich [36]. The material interface is indicated by the zero of a level set, which is advanced in time using well-established algorithms [14; 25]. Likewise, the extended velocity field that governs the level set evolution uses a well-established method [1] to extend a velocity computed from two-gas Riemann problems across the front. In the following technical description of the algorithm, we assume that the reader is familiar with the high-order Godunov algorithm [7; 45] and the PDE-based level set algorithms [1; 14; 25; 46] that we employ essentially as they appear in the literature. Here the emphasis will be placed on the interconnectivity of these algorithmic components, and new specific details of implementation. We have implemented our algorithm in two spatial dimensions. The algorithm description is presented in a more general form, and we believe its implementation in 3-D will be straight-forward.

Our approach results in a robust high-order front-tracking method. Using convergence tests, we demonstrate that our method is second-order accurate for smooth flow, and first-order accurate in the presence of shocks.

2. Governing equations

2.1. *Gas dynamics.* We are developing a hybrid numerical method for a hyperbolic system of conservation laws:

$$\frac{\partial U}{\partial t} + \nabla \cdot \vec{F} = 0, \tag{1}$$

$$U = U(x, t), \quad x \in \Omega \subset \mathbb{R}^D, \tag{2}$$

$$\vec{F} = (F^1 \dots F^D) = \vec{F}(U), \tag{3}$$

$$U, F^d \in \mathbb{R}^m, \tag{4}$$

where U is a vector of conserved variables, and $\vec{F}$ is the corresponding flux vector defined in the problem domain Ω. A front $\mathcal{F}(t)$ separates the problem domain into two subdomains $\Omega_1(t)$ and $\Omega_2(t)$. Each gas is governed by the equation system stated above while fulfilling the Rankine–Hugoniot jump condition on the front,

$$(\vec{F}_1 - \vec{F}_2) \cdot \vec{n}_s - s^f (U_1 - U_2) = 0, \tag{5}$$

where $\vec{n}_s$ is the spatial normal vector on the front from Ω_1 to Ω_2, and s^f is the front velocity in the direction of $\vec{n}_s$ (Figure 1).

We restrict the presentation here to the 2-D Euler equations:

$$U = (\rho, \ \rho u, \ \rho v, \ E)^T, \tag{6}$$

$$F^1 = \left(\rho u, \ \rho u^2 + p, \ \rho u v, \ (E + p)u\right)^T, \tag{7}$$

$$F^2 = \left(\rho v, \ \rho u v, \ \rho v^2 + p, \ (E + p)v\right)^T, \tag{8}$$

where ρ is the gas density, u and v are the velocities in the x and y direction respectively, p is the pressure, and E is the total energy defined by the equation of
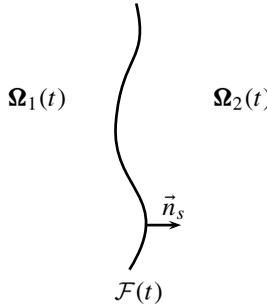
Figure 1. Two-gas flow domain and the moving interface.

state for an ideal polytropic gas,

$$E \equiv \frac{p}{\gamma - 1} + \frac{1}{2}\rho(u^2 + v^2), \tag{9}$$

where γ is the ratio of specific heats, and may be different for each gas. We use primitive variables $\boldsymbol{W} = (\rho,\ u,\ v,\ p)^T$ for flux calculations. The equation system in primitive variables is

$$\frac{\partial \boldsymbol{W}}{\partial t} + \sum_{d=1}^{D} \boldsymbol{A}^d \frac{\partial \boldsymbol{W}}{\partial x_d} = 0, \tag{10}$$

with

$$\boldsymbol{A}^1 = \begin{pmatrix} u & \rho & 0 & 0 \\ 0 & u & 0 & \rho^{-1} \\ 0 & 0 & u & 0 \\ 0 & \gamma p & 0 & u \end{pmatrix}, \quad \boldsymbol{A}^2 = \begin{pmatrix} v & 0 & \rho & 0 \\ 0 & v & 0 & 0 \\ 0 & 0 & v & \rho^{-1} \\ 0 & 0 & \gamma p & v \end{pmatrix}, \tag{11}$$

for $D = 2$.

2.2. *Front dynamics.* The level set method was introduced by Osher and Sethian to implicitly evolve an interface [44] (see the books by Sethian [52], and Osher and Fedkiw [43] for a complete introduction). We use a level set function to implicitly represent the front. The level set $\phi(\vec{x}, t)$ is a continuous function with

$$\begin{cases} \phi(\vec{x}, t) < 0 & \text{in } \boldsymbol{\Omega}_1(t), \\ \phi(\vec{x}, t) = 0 & \text{on } \mathcal{F}(t), \\ \phi(\vec{x}, t) > 0 & \text{in } \boldsymbol{\Omega}_2(t). \end{cases} \tag{12}$$

The interface is represented by the zero level set $\mathcal{F}(t) = \{\vec{x} \mid \phi(\vec{x}, t) = 0\}$. The level set is updated using the level set equation [44],

$$\phi_t + \vec{v}_{\text{ext}} \cdot \nabla\phi = 0, \tag{13}$$

where $\vec{v}_{\text{ext}}$ is the extended velocity defined in $\mathbb{R}^D$. The extended velocity represents the movement of the whole level set function and is defined to match the velocity of the front,

$$\vec{v}_{\text{ext}} = \vec{v}^f \quad \text{on } \mathcal{F}(t). \tag{14}$$

The level set function ϕ is used for calculating accurate geometric features of the front. Therefore, it is helpful to define and maintain it as a smooth function. It is initialized as a signed distance function that satisfies (12),

$$\phi(\vec{x}, 0) = \pm l(\vec{x}), \tag{15}$$

where l is the distance of point $\vec{x}$ to the front $\mathcal{F}(0)$. The level set equation (13) moves the zero level set correctly, but with time it will cease to be a distance

function, i.e., with $|\nabla \phi| = 1$. We maintain ϕ as a distance function by carefully generating the extended velocity [1], and by using a redistancing procedure [14]. The details are described in Section 4.3 and Section 4.4.

3. Discretization

3.1. *Control volumes, quadrature points, and special notation.* The physical problem domain is discretized with a Cartesian grid with size h in space and Δt in time. That is, a Cartesian (or regular) cell i is defined as $\Upsilon_i = [ih, (i + \Delta)h]$, $i \in \mathbb{Z}^D$, where Δ is a vector of ones. Let $T = [t^n, t^{n+1}]$ be the time interval. The spatial and space-time control volumes are then defined as

$$V_{i,\alpha}(t) = \Upsilon_i \cap \Omega_\alpha(t), \tag{16}$$

$$C_{i,\alpha} = \{\vec{x} \in \mathbb{R}^D, \ t \in T \mid \vec{x} \in V_{i,\alpha}(t)\}. \tag{17}$$

A regular space-time control volume is a rectangular cube in $\mathbb{R}^D \times T$ with $2D$ *faces* in $\mathbb{R}^{D-1} \times T$. The center of the cell Υ_i at time t^n is $(\vec{x}_i, t^n)$. The faces are shown by $A_{i \pm \frac{1}{2} e^d}$ where e^d is the unit vector in direction d and sign $\pm$ indicates that the face located on lower or higher side in direction d. Face centers are located at $(\vec{x}_{i \pm \frac{1}{2} e^d}, t^{n+\frac{1}{2}})$, and are at time $n + \frac{1}{2}$ (Figure 2b).
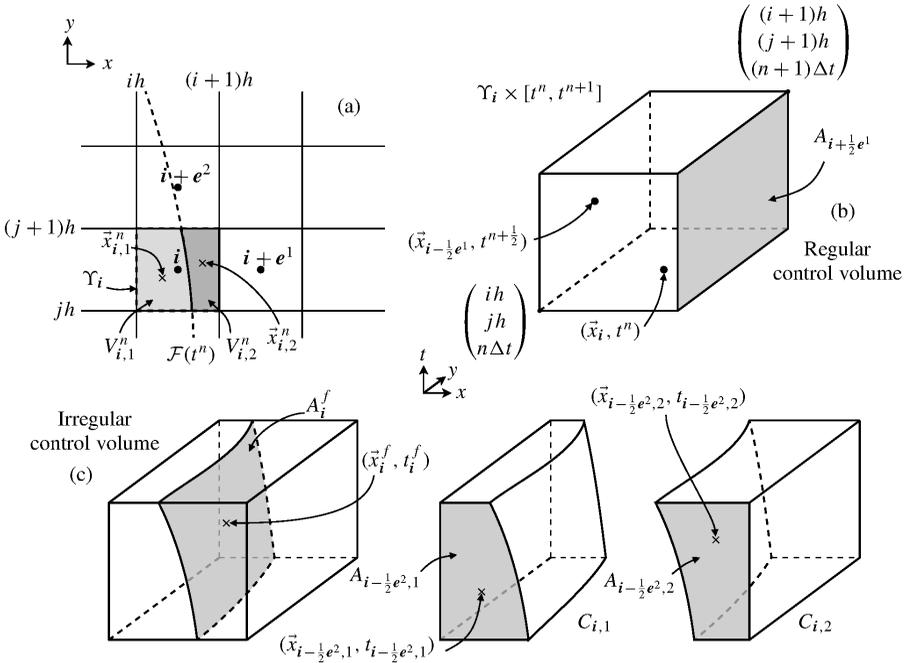
Figure 2. Geometry discretization and definitions.

A cell, face, or volume is called irregular or cut if it intersects the front. In Figure 2a, $V_{i,1}^n$ and $V_{i,2}^n$ denote the irregular cells corresponding to gases 1 and 2, respectively. Since such cells are fractional, the position of the center and centroid are different. The centroid of $V_{i,\alpha}^n$ is located at $(\vec{x}_{i,\alpha}, t^n)$. For faces of an irregular control volume, if a face is coincident only with the Cartesian grid, it is denoted as $A_{i\pm\frac{1}{2}e^d,\alpha}$, and its centroid as $(\vec{x}_{i\pm\frac{1}{2}e^d,\alpha}, t_{i\pm\frac{1}{2}e^d,\alpha})$. If a face is coincident with the front, it is written as A_i^f with the centroid at $(\vec{x}_i^f, t_i^f)$ (Figure 2c). Based on the above definitions, the cell and face fractions are specified as follows:

$$\Lambda_{i,\alpha}^n = \frac{|V_{i,\alpha}^n|}{h^D}, \quad a_{i\pm\frac{1}{2}e^d,\alpha} = \frac{|A_{i\pm\frac{1}{2}e^d,\alpha}|}{\Delta t\, h^{D-1}}, \quad \text{and} \quad a_i^f = \frac{|A_i^f|}{\Delta t\, h^{D-1}}. \tag{18}$$

The cell-centroid $(\vec{x}_{i,\alpha}, t^n)$ is defined as the center of $V_{i,\alpha}^n$,

$$\vec{x}_{i,\alpha} = \frac{1}{|V_{i,\alpha}^n|} \int_{V_{i,\alpha}^n} \vec{x}\, dV, \tag{19}$$

and face-centroids are given by

$$(\vec{x}_{i+\frac{1}{2}e^d,\alpha}, t_{i+\frac{1}{2}e^d,\alpha}) = \frac{1}{|A_{i+\frac{1}{2}e^d,\alpha}|} \int_{A_{i+\frac{1}{2}e^d,\alpha}} (\vec{x}, t)\, dA\, dt, \tag{20}$$

$$(\vec{x}_i^f, t_i^f) = \frac{1}{|A_i^f|} \int_{A_i^f} (\vec{x}, t)\, dA\, dt, \tag{21}$$

where $dV = dx^D$ and $dA = dx^{D-1}$ for $D > 1$. The average space-time normal vector on the front for cell i is defined as

$$\vec{n}_{i,\alpha} = \frac{1}{|A_i^f|} \int_{A_i^f} \vec{n}_\alpha\, dA\, dt, \tag{22}$$

where $\vec{n}_\alpha$ is the outward normal vector for gas α.

We will distinguish fluxes $\boldsymbol{F}$ defined on the interface, on face centers, and on face centroids, by $\boldsymbol{F}^f$, $\boldsymbol{F}^{cr}$, and $\boldsymbol{F}^{cd}$, respectively.

Note that the quantities related to irregular geometries, having subscript α or superscript f, are inherently time dependent variables and defined in the interval $[t^n, t^{n+1}]$. The notation is summarized in Table 12.

3.2. Gas dynamics. The conservation equation may be written beginning with the integral form

$$\int_{C_{i,\alpha}} \left(\nabla, \frac{\partial}{\partial t}\right) \cdot (\vec{F}, U)\, dV\, dt = 0. \tag{23}$$

To derive the finite volume method, the divergence theorem is applied to (23) giving

$$\oint_{\partial C_{i,\alpha}} \vec{n}_\alpha \cdot (\vec{F}, U)\, dA = 0, \tag{24}$$
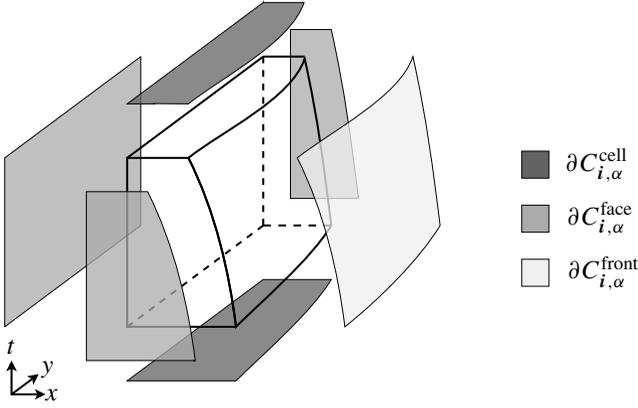
Figure 3. Separation of the integral over $\partial C_{i,\alpha}$ based on the type of surfaces.

where $\vec{n}_\alpha$ and $\mathrm{d}A$ are the outward space-time normal vector and surface area differential on region $\partial C_{i,\alpha}$, respectively. Separating this integral into three parts according to surface type gives

$$\int_{\partial C_{i,\alpha}^{\mathrm{cell}}} \vec{n}_\alpha \cdot (\vec{F}, U) \, \mathrm{d}A + \int_{\partial C_{i,\alpha}^{\mathrm{face}}} \vec{n}_\alpha \cdot (\vec{F}, U) \, \mathrm{d}A + \int_{\partial C_{i,\alpha}^{\mathrm{front}}} \vec{n}_\alpha \cdot (\vec{F}, U) \, \mathrm{d}A = 0. \quad (25)$$

The first integral is over the cells at time t^n and t^{n+1}, the second integral is over the intersection of $\partial C_{i,\alpha}$ with the regular faces, and the third integral is over the intersection of $\partial C_{i,\alpha}$ with the front (Figure 3).

Discretizing in space and time gives

$$|V_{i,\alpha}^{n+1}| U_{i,\alpha}^{n+1} - |V_{i,\alpha}^n| U_{i,\alpha}^n$$
$$+ \sum_{\pm,d=1}^D \left(\pm |A_{i \pm \frac{1}{2} e^d, \alpha}| F_{i \pm \frac{1}{2} e^d, \alpha}^{\mathrm{cd}} \right) + |A_i^f| \vec{n}_{i,\alpha} \cdot (\vec{F}_{i,\alpha}^f, U_{i,\alpha}^f) = \mathcal{O}(h^{D+1} \Delta t). \quad (26)$$

Here, $U_{i,\alpha}^n$ represents the average value of $U(\vec{x}, t)$ in $V_{i,\alpha}^n$, and is centered at $(\vec{x}_{i,\alpha}, t^n)$, the centroid of $V_{i,\alpha}^n$. Flux $F_{i \pm \frac{1}{2} e^d, \alpha}^{\mathrm{cd}}$ is the average flux at the space-time centroid of the corresponding faces in direction d, and $\vec{F}_{i,\alpha}^f$ and $U_{i,\alpha}^f$ are the average front flux and conservative state variable respectively, at the centroid of the front. Following the free-stream-preserving discretization of Pember et al. [45] the calculation on the front may be written in terms of fractional face areas on the

Cartesian grid instead of relying on an estimation of the front area itself:

$$|V_{i,\alpha}^{n+1}|U_{i,\alpha}^{n+1} - |V_{i,\alpha}^{n}|U_{i,\alpha}^{n}$$
$$+ \sum_{\pm,d=1}^{D} \left(\pm|A_{i\pm\frac{1}{2}e^d,\alpha}|F_{i\pm\frac{1}{2}e^d,\alpha}^{\mathrm{cd}}\right) - \sum_{d=1}^{D}(|A_{i+\frac{1}{2}e^d,\alpha}| - |A_{i-\frac{1}{2}e^d,\alpha}|)F_{i,d,\alpha}^{f}$$
$$- (|V_{i,\alpha}^{n+1}| - |V_{i,\alpha}^{n}|)U_{i,\alpha}^{f} = \mathcal{O}(h^{D+1}\Delta t). \quad (27)$$

Dividing by h^D and substituting cell and face areas with unitless cell and face fractions gives

$$\Lambda_{i,\alpha}^{n+1}U_{i,\alpha}^{n+1} - \Lambda_{i,\alpha}^{n}U_{i,\alpha}^{n} + \frac{\Delta t}{h}\sum_{\pm,d=1}^{D}\left(\pm a_{i\pm\frac{1}{2}e^d,\alpha}(F_{i\pm\frac{1}{2}e^d,\alpha}^{\mathrm{cd}} - F_{i,d,\alpha}^{f})\right)$$
$$- (\Lambda_{i,\alpha}^{n+1} - \Lambda_{i,\alpha}^{n})U_{i,\alpha}^{f} = \mathcal{O}(h\Delta t). \quad (28)$$

An interim conserved variable $U^{n,n+1}$ is defined as the conserved variable evaluated at time t^n and centered at the cell-centroid position at time t^{n+1}

$$U_{i,\alpha}^{n,n+1} = U(\vec{x}_{i,\alpha}(t^{n+1}), t^n), \quad (29)$$

which is calculated by interpolating cell-centroid conservative variables, $U_{i,\alpha}^n$, using the local grid interpolation method explained in Section 4.1. The purpose of this unusually centered variable, introduced in [16], is to make the following explicit update equation (compare (28)) uniformly centered in space:

$$U_{i,\alpha}^{n+1} = U_{i,\alpha}^{n,n+1} - \Delta t\, DF_{i,\alpha}^{C}, \quad (30)$$

where $DF_{i,\alpha}^{C}$ is the conservative flux difference defined as

$$DF_{i,\alpha}^{C} = \frac{\Lambda_{i,\alpha}^{n+1}U_{i,\alpha}^{n,n+1} - \Lambda_{i,\alpha}^{n}U_{i,\alpha}^{n}}{\Lambda_{i,\alpha}^{n+1}\Delta t}$$
$$+ \frac{1}{\Lambda_{i,\alpha}^{n+1}h}\left(\sum_{\pm,d=1}^{D}\pm\left(a_{i\pm\frac{1}{2}e^d,\alpha}(F_{i\pm\frac{1}{2}e^d,\alpha}^{\mathrm{cd}} - F_{i,d,\alpha}^{f})\right)\right.$$
$$\left. - \frac{h}{\Delta t}(\Lambda_{i,\alpha}^{n+1} - \Lambda_{i,\alpha}^{n})U_{i,\alpha}^{f}\right). \quad (31)$$

Note that $DF_{i,\alpha}^{C}$ may become unstable for small cell fraction $\Lambda_{i,\alpha}^{n+1}$. Therefore, a nonconservative but stable flux difference is introduced,

$$DF_{i,\alpha}^{\mathrm{NC,cr}} = \frac{1}{h}\sum_{\pm,d=1}^{D}\pm F_{i\pm\frac{1}{2}e^d,\alpha}^{\mathrm{cr}}, \quad (32)$$

where $\boldsymbol{F}^{\mathrm{cr}}_{i \pm \frac{1}{2}e^d, \alpha}$ is the flux evaluated at $(\vec{x}_{i \pm \frac{1}{2}e^d}, t^{n+\frac{1}{2}})$, the center of the regular face $A_{i \pm \frac{1}{2}e^d}$, for gas α. This calculation results in a cell-centered estimate of the nonconservative flux. Then, it is extrapolated to the cell-centroid position at time t^{n+1}, which gives $\boldsymbol{DF}^{\mathrm{NC}}_{i, \alpha}$. A linear combination of $\boldsymbol{DF}^{C}_{i, \alpha}$ and $\boldsymbol{DF}^{\mathrm{NC}}_{i, \alpha}$ is used for the update equation to avoid the small cell problem of the finite volume methods [10]. By picking $\Lambda^{n+1}_{i, \alpha}$ as the linear coefficient of $\boldsymbol{DF}^{C}_{i, \alpha}$ a stable update method is achieved:

$$U^{n+1}_{i, \alpha} = U^{n,n+1}_{i, \alpha} - \Delta t \big[\Lambda^{n+1}_{i, \alpha} \, \boldsymbol{DF}^{C}_{i, \alpha} + (1 - \Lambda^{n+1}_{i, \alpha}) \boldsymbol{DF}^{\mathrm{NC}}_{i, \alpha} \big]. \tag{33}$$

To maintain global conservation, the mass difference between the conservative update (30) and the hybrid update (33) is calculated

$$\delta M_{i, \alpha} = \Lambda^{n+1}_{i, \alpha} \big([U^{n+1}_{i, \alpha} - U^{n,n+1}_{i, \alpha}]_{\mathrm{Conservative}} - [U^{n+1}_{i, \alpha} - U^{n,n+1}_{i, \alpha}]_{\mathrm{Hybrid}} \big),$$

$$\delta M_{i, \alpha} = \Delta t \, \Lambda^{n+1}_{i, \alpha} (1 - \Lambda^{n+1}_{i, \alpha}) (\boldsymbol{DF}^{\mathrm{NC}}_{i, \alpha} - \boldsymbol{DF}^{C}_{i, \alpha}), \tag{34}$$

and this excess mass is redistributed to the appropriate neighbor cells of cell i.

3.3. Front dynamics. The level set equation (13) is discretized in space on the same Cartesian grid using the WENO method for Hamilton–Jacobi equations by Jiang and Peng [25]. The level set function at any time step is only needed in a band around the zero level set [31]. The local level set method of Peng et al. [46] is applied to reduce the computational work for updating the level set function. This means the update method applies to the cells in a band around the zero level set. The update band should be wide enough to provide enough information to support the geometry calculation. With our choice of parameters, a 5×5 stencil of level set values is needed centered on each irregular cell.

For time discretization a two-step Adams–Bashforth method is used [34]. For the ODE system

$$\phi_t = L(\phi), \quad \phi(\vec{x}, 0) = \mathcal{F}(0), \tag{35}$$

the update algorithm at time step n is

$$\begin{cases} \tilde{\phi}^{n+1} = \phi^n + \Delta t L(\phi^n) \\ (D\phi)^n = \tilde{\phi}^{n+1} - \phi^n \\ \phi^{n+1} = \phi^n + \frac{3}{2}(D\phi)^n - \frac{1}{2}(D\phi)^{n-1}. \end{cases} \tag{36}$$

The above method is second-order accurate in time with the initial starting value $(D\phi)^{-1} = (D\phi)^0$. For the level set update equation the L operator is defined as $L(\phi) = -\vec{v}_{\mathrm{ext}} \cdot \nabla \phi$. With this choice of time stepping, L is evaluated only once per time step, which means that the two-gas front Riemann problem need be solved only once per time step to compute $\vec{v}^f$, which is extended to make $\vec{v}_{\mathrm{ext}}$.

3.4. *Geometry.* To calculate the update equation (33) we need the geometrical quantities defined in Section 3.1. Using $k+2$ discrete time samples of level set data ϕ^{n+1}, ϕ^n, ..., ϕ^{n-k}, and applying the algorithm developed by Miller and Trebotich [36], we may evaluate the required geometry information with accuracy as high as $\mathcal{O}(h^{k+2})$. We use $k = 2$ to compute geometry information with fourth-order accuracy in our test cases. To obtain fourth-order accuracy by this approach requires a 5×5 stencil of discrete level set values at each time level.

4. Numerical algorithms

At the start of time step n we assume that the following data are available: a centroid-centered average of the conservative variables for each gas $U_{i,\alpha}^n(\vec{x}_{i,\alpha}^n, t^n)$; a time sequence of $k+1$ discrete level sets ϕ^n, ..., ϕ^{n-k}; the difference $(D\phi)^{n-1}$ (36); and spatial geometry information constructed from ϕ^n. The goal is to find $U_{i,\alpha}^{n+1}(\vec{x}_{i,\alpha}^{n+1}, t^{n+1})$ and ϕ^{n+1}. A brief overview of the strategy follows. Additional details are given in [57].

First, we extrapolate the conservative data from cell centroids to cell centers to support a standard high-order Godunov method. Primitive cell-centered data is extrapolated to the front centroids at time t^n to pose Riemann problems that determine the front velocity $\vec{v}^f$.

The front centroid velocity is extended in a band of cells around the zero level set to make $\vec{v}_{\text{ext}}$, which is used in (36) to compute ϕ^{n+1} to second-order. The time sequence ϕ^{n+1}, ϕ^n, ..., ϕ^{n-k} then supports the computation of space-time geometry information in all irregular control volumes.

Following a standard high-order unsplit Godunov method, modified on irregular boundaries [9; 11], we calculate the slopes and fluxes on the face and front centroids and apply the update equation (33). Excess mass is calculated and redistributed.

4.1. *Local grid interpolation.* The interpolation steps used in the algorithm are linear estimates based on the local grid data. If we want to interpolate the value of function G at target point $\vec{x}_t$ based on the support data at $\vec{x}_s$ we write the Taylor expansion of $G(\vec{x}_s)$ centered at $\vec{x}_t$,

$$G(\vec{x}_s) \approx G(\vec{x}_t) + \nabla G(\vec{x}_t)(\vec{x}_s - \vec{x}_t). \tag{37}$$

By gathering enough support point data around the target we create a least-squares problem to solve for $G(\vec{x}_t)$,

$$\begin{pmatrix} 1 & (\vec{x}_1 - \vec{x}_t) \\ \vdots & \vdots \\ 1 & (\vec{x}_m - \vec{x}_t) \end{pmatrix} \begin{pmatrix} G(\vec{x}_t) \\ \nabla G(\vec{x}_t) \end{pmatrix} = \begin{pmatrix} G(\vec{x}_1) \\ \vdots \\ G(\vec{x}_m) \end{pmatrix}, \tag{38}$$

9	8	7	6	7	8	9
8	5	4	3	4	5	8
7	4	2	1	2	4	7
6	3	1	0	1	3	6
7	4	2	1	2	4	7
8	5	4	3	4	5	8
9	8	7	6	7	8	9

(a)

(b)
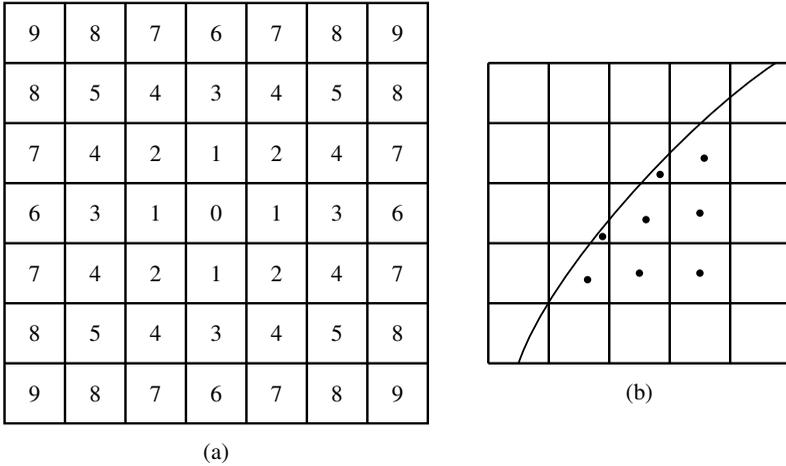
Figure 4. (a) Cell stencil showing sets of nearest neighbors. Neighbor cells with the same approximate distance to the center cell are indicated by a common number. (b) Interpolation from cell centroid data to cell center: Target point (the cell center) is shown with × and support points are shown with ●. In this example only sets 1 and 2 of nearest neighbor cells are used.

where $\vec{x}_1, \ldots, \vec{x}_m$ are support points. The support points are chosen based on the nearest cell criteria to the target cell position. We start from the most compact stencil possible and try solve the above least-square system with QR decomposition (Figure 4). More support points are added if values on the diagonal of R are too small, which signals large numerical errors. The heuristic cut-off values that we used are 10^{-10} for R_{ii} (no sum), and 10^{-15} for $\det(R)$.

4.2. *Solution on the front and velocity matching.* We need the state variables on the front to calculate the front speed at time t^n, and the flux on the front centroid. In both cases, the state variables from each gas are extrapolated to the front centroids. We solve the Riemann problem in front normal direction for a contact discontinuity to determine the states on the front, using a primitive variable Riemann solver [56].

The Riemann solver provides a unique normal velocity u^*, from which we determine $\vec{v}^f = u^*\mathbf{n}$. The tangential component of the front velocity is double valued, and not needed for level set advection.

When computing $\boldsymbol{DF}^C$ (31) we require the flux at the space-time front centroid. This flux is computed by first setting up and solving a Riemann problem as described above with left-hand and right-hand values interpolated as described below in (48). This computation also yields a unique normal velocity u^*, however this velocity is

not consistent with the space-time slope of the front, which gives velocity

$$s^f = \frac{-n_t}{\sqrt{n_x^2 + n_y^2}}, \quad \vec{n}_i^n = (n_x, n_y, n_t)^T. \tag{39}$$

We substitute s^f for u^* in this Riemann problem calculation to give front values that are internally consistent and make the method fully conservative. Additional details are given in [57].

4.3. *Velocity extension.* We need the velocity defined on cell centers in a band around the zero level set to update the level set equation. The solution on the front at time t^n gives the velocity on front centroids.

To keep ϕ a signed distance function after updating the level set equation one needs to impose the following condition on the velocity [1]:

$$\nabla(\vec{v}_{\text{ext},d}) \cdot \nabla\phi = 0, \quad d = 1, \dots, D, \tag{40}$$

which means that velocity is constant along the level set gradient. We use the extension method developed by Peng et al. [46] and Zhao et al. [60]. To extend the quantity q with the condition $\nabla q \cdot \nabla\phi = 0$ we solve

$$q_t + S(\phi)\frac{\nabla\phi}{|\nabla\phi|} \cdot \nabla q = 0, \tag{41}$$

as an internal boundary value problem, where $S(\phi)$ is a regularized sign function [54]. Since the characteristics of (41) are normal to the front and point outward from the front, we only need to initialize q on a initialization band around $\mathcal{F}$, with a width of 2 or 3 cells, and then solve (41) while the velocity is anchored on the initialization band.

To set up the initialization band around the zero level set we pick cells that have amongst their $3^D - 1$ neighbors at least one cell with the opposite level set sign. Applying the extension condition (40) means the extended velocity field does not change in the direction of the level set gradient. Therefore, if we project from a point outside the front, such as the center of a cell in the initialization band $\vec{x}_c$, in the direction of the level set gradient toward the front and reach a point on the front such as $\vec{x}_f$,

$$\vec{x}_f \approx \vec{x}_c - \phi(\vec{x}_c)\frac{\nabla\phi(\vec{x}_c)}{|\nabla\phi(\vec{x}_c)|} \tag{42}$$

(see Figure 5), then the extension condition (40) recovers

$$\vec{v}(\vec{x}_c) = \vec{v}(\vec{x}_f). \tag{43}$$

To find the velocity at the projected point $\vec{x}_f$, we apply linear interpolation using the velocity values on the front centroids in the 3^D cells near $\vec{x}_f$. If m front centroid
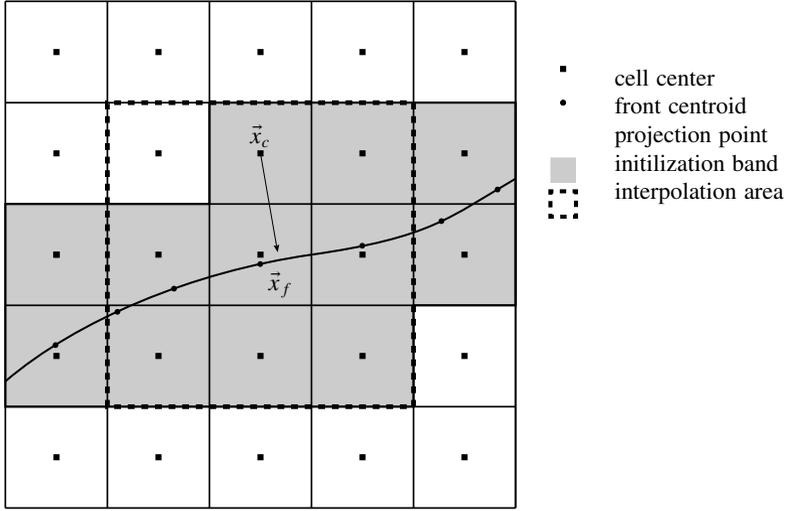
Figure 5. Initialization of velocity field in a narrow band around zero level set.

points reside in the vicinity of $\vec{x}_f$, we solve the least-squares equation system (38) with $G = v_d(\vec{x}_f)$. In this case, we solve system (38) using singular value decomposition because the matrix may be singular, e.g., if the support points are collinear.

Extension equation (41) is solved iteratively as described by Peng et al. [46] to extend the velocity from the initialization band to a narrow band around the zero level set while applying the high-order WENO discretization of Jiang and Peng [25] to increase the accuracy of the algorithm and decrease the number of iterations that are needed.

4.4. _Redistancing._ A level set function initialized as a distance function and driven with a nonconstant velocity field may deviate from being a distance function [46; 50], causing poor evaluation of geometric quantities. Sussman et al. [54] introduced the redistancing equation in artificial time τ

$$\phi_\tau + S(\tilde{\phi})(|\nabla\phi| - 1) = 0, \tag{44}$$

where $\tilde{\phi} = \phi(\vec{x}, \tau = 0)$. We solve the redistancing equation (44) using the improved Hamilton–Jacobi WENO algorithm with Godunov approximation in space and the third-order TVD Runge–Kutta scheme in time by du Chéné et al. [14].

4.5. _Flux calculation._ There are four different types of fluxes needed to apply the hybrid update method, varying in centering location.

4.5.1. _Flux on a regular face._ A regular face is not intersected by the front in $[t^n, t^{n+1}]$. The most common flux calculated is $\boldsymbol{F}^{\mathrm{cr}}_{i\pm\frac{1}{2}e^d,\alpha}$ for a regular face, which is

the flux on the face center $(\vec{x}_{i\pm\frac{1}{2}e^d}, t^{n+\frac{1}{2}})$. This is done using a high-order Godunov method with transverse flux correction in primitive variables [9; 11].

4.5.2. *Flux on a covered face.* If cell i is irregular, but the face at $i \pm \frac{1}{2}e^d$ does not intersect the front, then that face is said to be *covered* with respect to the gas α that is not present at $i \pm \frac{1}{2}e^d$. Fluxes on covered faces may be used in (32). The upwinding extrapolation algorithm that is used for flux calculation on a regular face only provides the state variable on one side of the covered face. A method similar to the procedure described in [11, §5.2] is applied to extrapolate the state variables on the other side using the interior face-centered values. We use the spatial projection of $\vec{n}_{i,\alpha}$ in 2-D as the normal vector needed in the method.

4.5.3. *Flux on an irregular face.* A face is irregular if it intersects the front in $[t^n, t^{n+1}]$. We evaluate the average flux on the irregular face using the primitive states on the face centroid,

$$F^{\text{cd}}_{i\pm\frac{1}{2}e^d,\alpha} = F^d(W^{\text{cd}}_{i\pm\frac{1}{2}e^d,\alpha}). \tag{45}$$

We extrapolate the primitive variable on the face center $W^{\text{cr}}_{i\pm\frac{1}{2}e^d,\alpha}$, calculated in Section 4.5.1, to the face centroid using the average slopes of two surrounding cells in time and space,

$$
\begin{aligned}
W^{\text{cd}}_{i\pm\frac{1}{2}e^d,\alpha} &= W_\alpha\left(\vec{x}_{i\pm\frac{1}{2}e^d,\alpha}, t_{i\pm\frac{1}{2}e^d,\alpha}\right) \\
&= W^{\text{cr}}_{i\pm\frac{1}{2}e^d,\alpha} + \left(\vec{x}_{i\pm\frac{1}{2}e^d,\alpha}(t^n) - \vec{x}_{i\pm\frac{1}{2}e^d}\right) \cdot e^d \left(\frac{\Delta_x^{d'} W^n_{i,\alpha} + \Delta_x^{d'} W^n_{i\pm e^d,\alpha}}{2h}\right) \\
&\quad + \left(t_{i\pm\frac{1}{2}e^d,\alpha} - (n+\frac{1}{2})\Delta t\right)\left(\frac{\Delta_t W^n_{i,\alpha} + \Delta_t W^n_{i\pm e^d,\alpha}}{2\Delta t}\right), \quad d \neq d', \tag{46}
\end{aligned}
$$

where $\Delta_t W^n_{i,\alpha}$ is an estimate of time difference calculated as

$$\Delta_t W^n_{i,\alpha} = 2\left(\frac{1}{2D}\sum_{\pm}\sum_{d=1}^{D} W^{\text{cr}}_{i\pm\frac{1}{2}e^d,\alpha} - W^n_{i,\alpha}\right). \tag{47}$$

4.5.4. *Flux on the front.* The flux on the front is calculated using the solution of the Riemann problem on the contact discontinuity described in Section 4.2. The left-hand and right-hand states of that Riemann problem are derived by extrapolation from cell-centered primitive states using the local slopes in space and time for each gas,

$$
\begin{aligned}
\widetilde{W}^f_{i,\alpha} &= \widetilde{W}_\alpha(\vec{x}^f_i, t^f_i) \\
&= W^n_{i,\alpha} + \frac{1}{h}\sum_{d=1}^{D}(\vec{x}^f_i - \vec{x}_{i,\alpha}(t^n)) \cdot e^d \Delta_x^d W^n_{i,\alpha} + \frac{1}{\Delta t}(t^f_i - n\Delta t)\Delta_t W^n_{i,\alpha}. \tag{48}
\end{aligned}
$$

The solution to the Riemann problem in the front normal direction with initial states $\widetilde{W}_{i,1}^f$ and $\widetilde{W}_{i,2}^f$ gives the primitive state on both sides of the front. These values, modified as described in Section 4.2, determine $W_{i,1}^f$ and $W_{i,2}^f$, which are used for the calculation of $F_{i,\alpha}^f$ and $U_{i,\alpha}^f$.

4.6. *Excess generalized mass and redistribution.*

To maintain global conservation the generalized mass (i.e., $\sum_i |V_{i,\alpha}| U_{i,\alpha}$) difference between the hybrid method and the conservative method (34) should be redistributed. First we balance the excess generalized mass on the front cells based on the characteristics in the front normal coordinate systems using the approach of Chern and Colella [7]. The excess generalized mass on each side is projected on the characteristics of the cell centered values in the front normal coordinate system,

$$\delta M_{i,\alpha} = \sum_{k=0}^{m-1} b_{k,\alpha} r_{k,\alpha}, \tag{49}$$

where $r_{k,\alpha}$ are the right eigenvectors of $\frac{\partial \vec{F}}{\partial U}(U_{i,\alpha}^n)$. Then, the contribution of excess generalized mass to each side is calculated by considering the characteristic speed of each projected term. Assuming that $\tilde{u}_1$ and $\tilde{c}_1$ are the normal velocity and sound speed of $U_{i,1}^n$ in the front normal coordinate system, the projected components corresponding to $\tilde{u}_1 - \tilde{c}_1$ and $\tilde{u}_1$ remain in gas 1, and the projected component corresponding to $\tilde{u}_1 + \tilde{c}_1$ moves to gas 2. With a similar argument for gas 2, the excess generalized mass values are:

$$\delta M_{i,1} := (b_{0,1} r_{0,1} + b_{1,1} r_{1,1} + b_{2,1} r_{2,1}) + b_{0,2} r_{0,2}, \tag{50}$$

$$\delta M_{i,2} := b_{3,1} r_{3,1} + (b_{1,2} r_{1,2} + b_{2,2} r_{2,2} + b_{3,2} r_{3,2}). \tag{51}$$

Note that physical mass (i.e., $\sum_i |V_{i,\alpha}| \rho_{i,\alpha}$) is not redistributed across a front since the characteristic carrying mass has velocity $\tilde{u}_\alpha$.

In general, the redistribution method is

$$U_{j,\alpha}^{n+1} := U_{j,\alpha}^{n+1} + w_{i,j}\, \delta M_{i,\alpha}, \quad j \in N(i), \tag{52}$$

where $N(i)$ indicates a set of cells in the neighborhood of cell i and the weight coefficients satisfy [11],

$$w_{i,j} \geq 0, \quad \text{and} \quad \sum_{j \in N(i)} w_{i,j} \Lambda_j^{n+1} = 1. \tag{53}$$

We use

$$w_{i,j} = \frac{1}{\sum_{l \in N(i)} \Lambda_l^{n+1}}. \tag{54}$$

With these weight coefficients we get

$$\Lambda_{j,\alpha}^{n+1} U_{j,\alpha}^{n+1} := \Lambda_{j,\alpha}^{n+1} U_{j,\alpha}^{n+1} + \frac{\Lambda_{j,\alpha}^{n+1}}{\sum_{k \in N(i)} \Lambda_{k,\alpha}^{n+1}} \delta M_{i,\alpha}, \quad j \in N(i), \qquad (55)$$

which means the excess mass of cell i is redistributed into neighbor cells in proportion to their volume at time $n + 1$.

5. Results

In this section we present numerical tests of the method for gas dynamics problems. The first two tests are for smooth and steady flows, where no discontinuity other than the material interface is present. The third and fourth test include the interaction of the front with a shock wave. In the fifth and sixth test cases, front dynamics are unstable by nature, and the geometry of the front is changing more than in the previous cases.

To make quantitative comparisons we define the error for conservative state variables

$$E_{i,\alpha}^{2h} = U_{i,\alpha}^{2h}(t) - U_{i,\alpha}^{e}(t), \qquad (56)$$

where $U_{i,\alpha}^{2h}(t)$ is calculated with spatial grid size $2h$ and $U_{i,\alpha}^{e}$ is the exact solution. If we do not have the exact solution in hand, we replace it with the solution from the simulation with a finer grid. If we choose grid size h for our finer grid calculation

$$U_{i,\alpha}^{e}(t) \approx \frac{\sum_{j \in M(i)} \Lambda_{j,\alpha}^{h} U_{j,\alpha}^{h}(t)}{\sum_{j \in M(i)} \Lambda_{j,\alpha}^{h}}, \qquad (57)$$

where $M(i) = [2i, 2i + \Delta]$ are the fine cells underlying coarse cell i.

The total error in the L_1, L_2 and L_∞ norms is

$$\epsilon_{L_1}^{h} = \sum_{\alpha, i \in \Omega_\alpha} h^d \Lambda_{i,\alpha}^{h} |E_{i,\alpha}^{h}|, \qquad (58)$$

$$\epsilon_{L_2}^{h} = \left(\sum_{\alpha, i \in \Omega_\alpha} h^d \Lambda_{i,\alpha}^{h} |E_{i,\alpha}^{h}|^2 \right)^{\frac{1}{2}}, \qquad (59)$$

$$\epsilon_{L_\infty}^{h} = \max_{\alpha, i \in \Omega_\alpha} |E_{i,\alpha}^{h}|, \qquad (60)$$

and the convergence rate is calculated by

$$p = \frac{\log(\epsilon^{2h}/\epsilon^{h})}{\log(2)}. \qquad (61)$$

5.1. *Convergence test — advection.* With this simple test, we examine the convergence and conservation of the method for smooth gas and front dynamics. The problem domain is a 1 by 1 square. The initial material interface is a circle with radius 0.2 centered at $(0.25, 0.25)$. With the initial condition shown in Table 1 and appropriate inflow and outflow boundary conditions, simulations are done up to $t = 0.5$ with CFL number 0.5, which results in diagonal advection of the circular front without any deformation across the domain. Table 2 shows the error and convergence rates. The results show second order convergence for all norms. Fedkiw et al. [15] did a similar test and showed second order convergence in L_1. Here we also measure the convergence in L_∞ which shows the method is second order convergent away from and near the interface.

The final front position is displayed in Figure 6 for different resolutions. Since the final solution is trivial, we measured the least squares error of the front centroid positions using the exact solution. Numerical convergence analysis shows the front position is converging with $\mathcal{O}(h^{1.74})$ (see Figure 7).

At the request of a reviewer we measured the total amount of each conserved variable for each gas. Taking into consideration the boundary fluxes, the mass, momentum, and energy of each gas is conserved to within machine precision in each time step, and the cumulative loss of conservation is less than 10^{-16} for all conserved variables in all resolutions tested.

5.2. *Convergence test — smooth perturbation.* We test our method for a 2-D perturbation problem to observe the interaction of the front with smooth flows. The problem domain is a 1 by 1 square. The initial front is a circle with radius 0.27 centered at $(0.5, 0.5)$. The gas inside the front has a smooth bell-shaped perturbation in the pressure component of the form

$$p_{\text{in}}(r) = p_{\text{base}}\left(1 + 256\alpha\left(\frac{r+r_p}{2r_p} - \left(\frac{r+r_p}{2r_p}\right)^2\right)^4\right), \qquad r \le r_p, \qquad (62)$$

where $p_{\text{base}} = 1$ is the background value of pressure, $\alpha = 0.15$ determines the peak-to-base value of the perturbation, $r_p = 0.25$ is the radius of the perturbation and r is the distance from the center of the problem domain. The initial density is

	Gas 1: inside	Gas 2: outside
ρ	0.138	1
u	1	1
v	1	1
p	1	1
γ	1.67	1.4

Table 1. Initial values for the advection problem.

			ρ			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	5.24×10^{-5}		1.84×10^{-4}		4.41×10^{-3}	
1/64	4.95×10^{-6}	3.40	2.05×10^{-5}	3.16	6.47×10^{-4}	2.77
1/128	1.19×10^{-6}	2.06	3.22×10^{-6}	2.67	1.47×10^{-4}	2.36
1/256	2.73×10^{-7}	2.12	7.93×10^{-7}	2.27	3.48×10^{-5}	2.48

			ρu & ρv			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	1.12×10^{-4}		2.73×10^{-4}		4.12×10^{-3}	
1/64	1.00×10^{-5}	3.49	3.43×10^{-5}	2.99	6.42×10^{-4}	2.68
1/128	2.18×10^{-6}	2.20	8.05×10^{-6}	2.09	1.36×10^{-4}	2.24
1/256	5.38×10^{-7}	2.02	1.91×10^{-6}	2.08	3.30×10^{-4}	2.04

			E			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	2.70×10^{-4}		7.09×10^{-4}		1.34×10^{-2}	
1/64	2.80×10^{-5}	3.27	1.09×10^{-4}	2.70	2.45×10^{-3}	2.45
1/128	6.14×10^{-6}	2.19	1.91×10^{-5}	2.51	4.77×10^{-4}	2.40
1/256	1.40×10^{-6}	2.13	4.30×10^{-6}	2.15	1.11×10^{-4}	2.10

Table 2. Error and convergence rate for the advection problem.

constant and equals 1 for both gases, and the initial velocities are zero. The initial pressure for the gas outside the front is p_{base}. We simulate the state variables to a fixed time $t = 0.6$ to allow the perturbation to pass the front with a CFL number of 0.5. We simulated the expansion of the perturbation in seven grid sizes varying from $\frac{1}{32}$ to $\frac{1}{2048}$, and calculated the solution errors and convergence rates based on the Richardson method explained above.

Two cases for gas materials are considered. First, gases have different ratios of specific heats, $\gamma_{\text{in}} = 1.67$ and $\gamma_{\text{out}} = 1.4$ (see Table 3). Second, both gases are the same material, $\gamma_{\text{in}} = \gamma_{\text{out}} = 1.4$ (see Table 4). Our method shows second-order convergence in L_1, L_2 and L_∞ for all conserved variables in either case.

In the second case, since we chose the same material for both gases, we expect to have continuous states on the front. It is also verified by comparing with a single gas simulation, where no front tracking is performed and the same initial condition is used. Figure 8 shows the final results, which also exhibit the expected rotational symmetry.
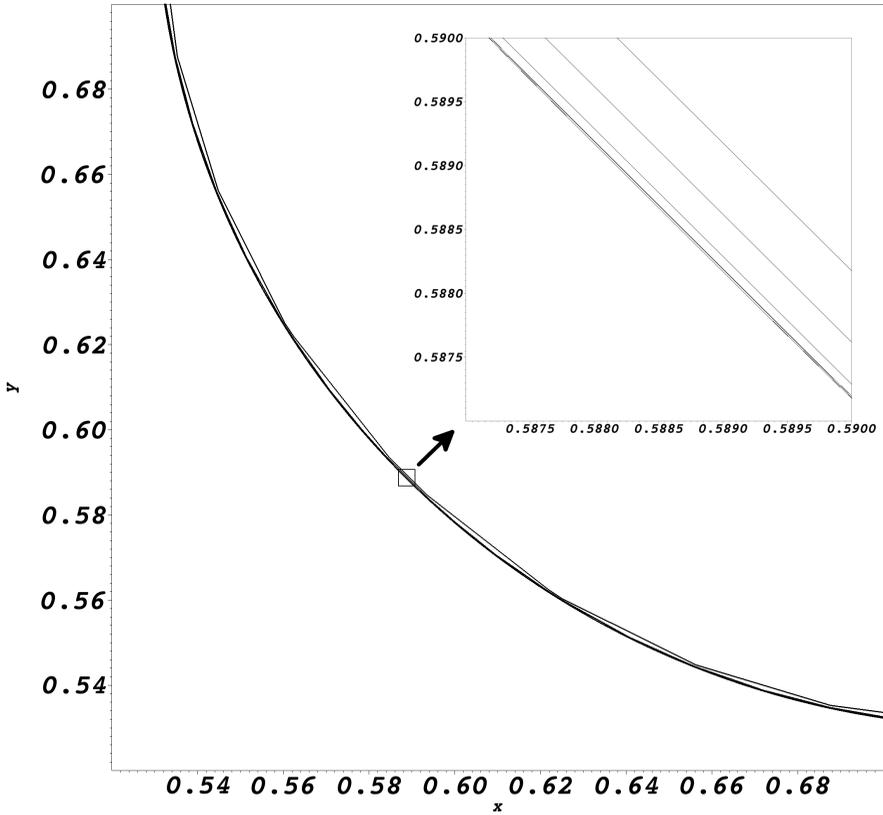
Figure 6. Final front position in the front advection problem. Inset: blowup of the front position. From right to left for h equal to $\frac{1}{32}$, $\frac{1}{64}$, $\frac{1}{128}$, $\frac{1}{256}$ and $\frac{1}{512}$.

Using finer resolution simulations as the reference (Richardson error), we calculated the least squares error of the front centroid positions, which shows a convergence rate of $\mathcal{O}(h^{1.60})$ (see Figure 7).

5.3. *Shock test — 2-D Sod problem.* Sod [53] introduced this benchmark 1-D problem to compare different numerical methods. Variations of this problem are also in common use [15; 16; 21; 28; 56]. Here we use this test in 2-D for two different materials. The problem domain is a 1 by 1 square. The initial front is a circle with radius 0.25 centered at (0.5, 0.5). The gas inside the circle has higher pressure and density (Table 5), causing an outward shock wave and inward rarefaction wave. The material discontinuity is tracked by our front tracking method. With CFL number 0.5 and solid wall boundary condition, simulations are done for five resolutions up to $t = 15$, before the shock wave reaches the domain boundary
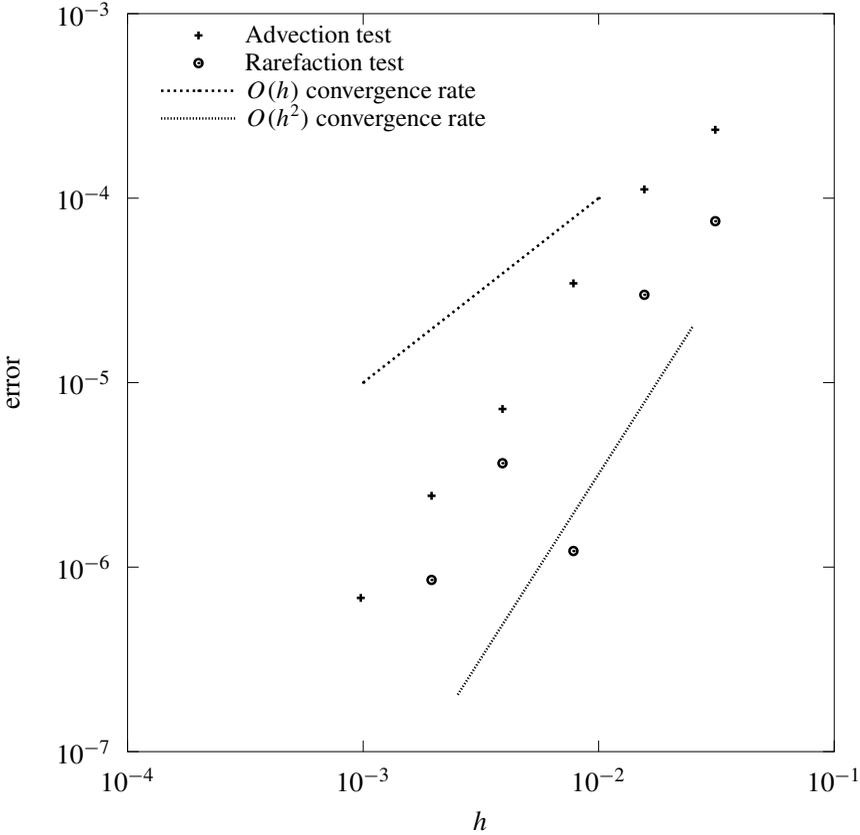
Figure 7. Convergence rate for front position.

(Figure 9), and error and convergence rates are calculated (Table 6). First-order convergence is observed, as expected for a problem containing captured shocks.

5.4. *Shock test — shock/front interaction.* We chose this problem to observe the nontrivial 2-D interaction of a shock wave with a tracked contact discontinuity. The domain is a [2, 0.5] channel. We use the Sod shock tube problem [53] to initialize the shock wave in gas 1 at $x = 1$. The initial material interface $\mathcal{F}(0)$ is a sine wave $\eta_0 \sin(\omega y + \theta) + x_0$ at $x_0 = 1.5$. The wavelength is the same as the width of the channel $\lambda = 0.5$, which gives $\omega = 2\pi/\lambda = 4\pi$. The interface amplitude is set to $\eta_0 = 0.16$ and $\theta = \pi/2$ (see Figure 10). The initial values for each gas are shown in Table 7. Solid wall and periodic boundary conditions are used for vertical and horizontal boundaries, respectively. The simulation is done up to time $t = 0.6$, allowing the shock wave to pass through the interface. Using five grid

			ρ			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	1.51×10^{-3}		2.40×10^{-3}		8.07×10^{-3}	
1/64	2.18×10^{-4}	2.79	2.65×10^{-4}	3.18	8.94×10^{-4}	3.17
1/128	3.45×10^{-5}	2.66	4.24×10^{-5}	2.64	1.64×10^{-4}	2.45
1/256	7.93×10^{-6}	2.12	1.05×10^{-5}	2.01	6.52×10^{-5}	1.33
1/512	2.03×10^{-6}	1.97	2.93×10^{-6}	1.84	1.74×10^{-5}	1.90
1/1024	5.26×10^{-7}	1.95	7.79×10^{-7}	1.91	3.76×10^{-6}	2.21

			ρu & ρv			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	9.89×10^{-4}		1.48×10^{-3}		7.48×10^{-3}	
1/64	1.69×10^{-4}	2.55	2.47×10^{-4}	2.58	1.26×10^{-3}	2.57
1/128	2.53×10^{-5}	2.74	3.83×10^{-5}	2.69	3.37×10^{-4}	1.90
1/256	5.36×10^{-6}	2.24	7.64×10^{-6}	2.33	1.52×10^{-4}	1.15
1/512	1.39×10^{-6}	1.95	1.95×10^{-6}	1.97	5.89×10^{-5}	1.37
1/1024	3.66×10^{-7}	1.92	5.05×10^{-7}	1.95	1.36×10^{-5}	2.11

			E			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	5.10×10^{-3}		8.38×10^{-3}		2.85×10^{-2}	
1/64	6.96×10^{-4}	2.87	8.55×10^{-4}	3.29	2.65×10^{-3}	3.43
1/128	1.08×10^{-4}	2.69	1.33×10^{-4}	2.68	4.60×10^{-4}	2.53
1/256	2.52×10^{-5}	2.10	3.47×10^{-5}	1.94	2.30×10^{-4}	1.00
1/512	6.55×10^{-6}	1.94	9.91×10^{-6}	1.81	6.10×10^{-5}	1.91
1/1024	1.72×10^{-6}	1.93	2.68×10^{-6}	1.89	1.32×10^{-5}	2.21

Table 3. Error and convergence rate for perturbation test with $\gamma_{\text{in}} = 1.67$ and $\gamma_{\text{out}} = 1.4$.

sizes varying from 1/32 to 1/512, error and convergence rates are calculated (see Table 8). The final result for the finest resolution is shown in Figure 10. Again, first-order convergence is observed as expected for a problem containing captured shocks.

5.5. _Deforming interface — Richtmyer–Meshkov instability._ To test our algorithm in a case comparable to experiment we simulated the initial growth of Richtmyer–Meshkov instability(RMI) that occurs when a sudden acceleration is forced on an interface separating fluids with different densities. It is observed that an initial small perturbation on the interface grows after passage of a shock wave in a wide range of Mach numbers [5]. Such growth has been characterized as having a linear phase

			ρ			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	1.59×10^{-3}		2.43×10^{-3}		1.56×10^{-2}	
1/64	1.92×10^{-4}	3.05	2.73×10^{-4}	3.15	1.96×10^{-3}	2.99
1/128	3.13×10^{-5}	2.61	4.42×10^{-5}	2.63	5.49×10^{-4}	1.83
1/256	6.25×10^{-6}	2.33	8.56×10^{-6}	2.37	8.61×10^{-5}	2.67
1/512	1.55×10^{-6}	2.01	2.18×10^{-6}	1.98	2.32×10^{-5}	1.90
1/1024	3.98×10^{-7}	1.96	5.68×10^{-7}	1.94	5.73×10^{-6}	2.02
			ρu & ρv			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	7.74×10^{-4}		1.12×10^{-3}		5.61×10^{-3}	
1/64	1.76×10^{-4}	2.14	2.50×10^{-4}	2.17	2.33×10^{-3}	1.27
1/128	2.57×10^{-5}	2.78	4.19×10^{-5}	2.58	9.44×10^{-4}	1.30
1/256	5.46×10^{-6}	2.23	8.01×10^{-6}	2.39	2.54×10^{-4}	1.89
1/512	1.44×10^{-6}	1.92	2.03×10^{-6}	1.98	4.89×10^{-5}	2.38
1/1024	3.76×10^{-7}	1.94	5.41×10^{-7}	1.91	1.21×10^{-5}	2.01
			E			
h	L_1	rate	L_2	rate	L_∞	rate
1/32	5.54×10^{-3}		8.54×10^{-3}		5.51×10^{-2}	
1/64	6.56×10^{-4}	3.08	9.53×10^{-4}	3.16	6.93×10^{-3}	2.99
1/128	1.07×10^{-4}	2.62	1.53×10^{-4}	2.64	1.92×10^{-3}	1.85
1/256	2.09×10^{-5}	2.36	2.94×10^{-5}	2.38	3.00×10^{-4}	2.68
1/512	5.21×10^{-6}	2.00	7.53×10^{-6}	1.96	8.29×10^{-5}	1.86
1/1024	1.35×10^{-6}	1.95	1.98×10^{-6}	1.93	2.17×10^{-5}	1.94

Table 4. Error and convergence rate for perturbation test with $\gamma_{in} = \gamma_{out} = 1.4$.

	Gas 1: inside	Gas 2: outside
ρ	3	1
u	0	0
v	0	0
p	3	1
γ	1.4	1.276

Table 5. Initial values for the 2-D Sod problem.

at the beginning of the process [49], before the nonlinear evolution of spike and bubble development, and of turbulent mixing.
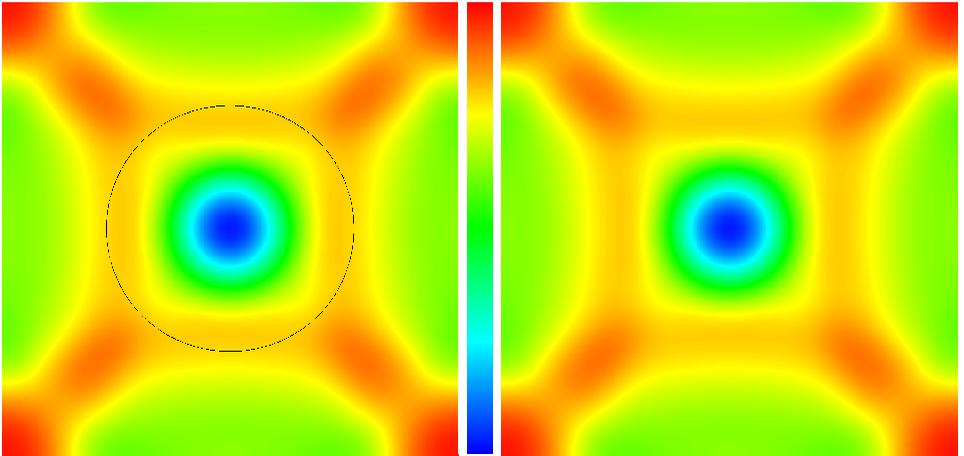
Figure 8. Density profile for smooth perturbation test at $t = 0.6$ on a 512×512 grid. Left: dual gas simulation, front position shown with solid line. Right: result for single gas simulation with no front-tracking. The color table limits are [0.9, 1.04].

h	ρ	rate	ρu & ρv	rate	E	rate
1/32	2.48×10^{-2}		1.73×10^{-2}		8.28×10^{-2}	
1/64	9.74×10^{-3}	1.35	6.49×10^{-3}	1.42	3.08×10^{-2}	1.43
1/128	4.91×10^{-3}	0.99	3.03×10^{-3}	1.10	1.44×10^{-2}	1.09
1/256	2.40×10^{-3}	1.03	1.46×10^{-3}	1.05	6.90×10^{-3}	1.07

Table 6. Error in L_1 and convergence rate for the 2-D Sod problem.

	Gas 1: postshock	Gas 1: preshock	Gas 2
ρ	3	1	1
u	0	0	0
v	0	0	0
p	3	1	1
γ	1.4	1.4	1.276

Table 7. Initial values for the shock channel problem.

Numerous experimental [12; 26; 39; 58] and numerical studies [23; 27; 33; 55] have been conducted on RMI. Here, we set up the simulation to model an experiment done by Collins and Jacobs [12]. The domain for this simulation is a [23.73 cm, 5.93 cm] channel. The initial shock is located at $x = 10$ cm and the interface is placed at $x = 12$ cm with a preshock sinusoidal perturbation of amplitude $a_0^- = 0.18$ cm and wavelength $\lambda = 5.93$ cm. The initial condition for gas 1 (preshock) and gas 2 are picked to be comparable to the cited experiment, and gas
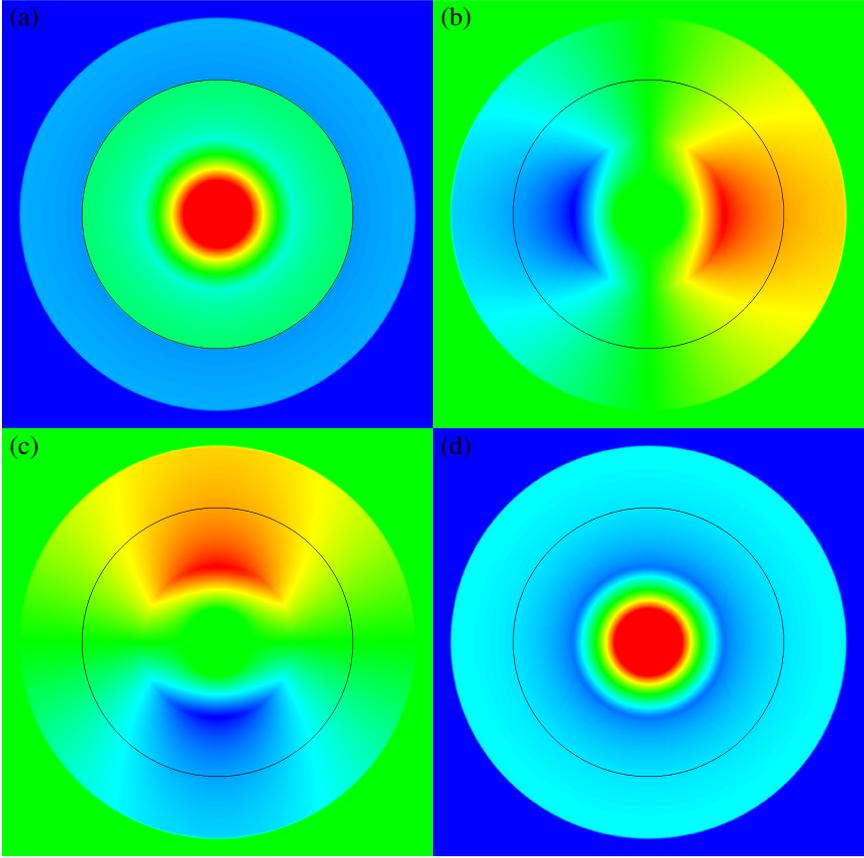
Figure 9. Results for 2-D Sod problem at finest resolution. (a) Density, (b) x-momentum, (c) y-momentum and (d) pressure. The same color table as Figure 8 is used with limits [1.0, 3.0] for density and pressure and [-0.614, 0.614] for momentum.

1 (postshock) initial conditions were chosen based on the Rankine–Hugoniot jump condition to have a shock with Ma $= 1.21$ in gas 1 (Table 9). With CFL number 0.5 we simulated the growth of perturbation up to time 0.4 ms (see Figure 11) and measured the convergence rate of the method (see Table 10).

Richtmyer [49] derived the impulsive growth rate relation for the amplitude of the perturbation based on the linear theory which describes the development of the instability after refraction of shock while the perturbation is small enough to be considered in the linear regime of the process,

$$\frac{d\eta}{dt} = kA^{+}\Delta V_{f}\eta_{0}^{+}, \tag{63}$$

where k is the wavenumber of the perturbation, A^{+} is the postshock Atwood number, ΔV_{f} is the velocity jump on the front following the shock refraction and η_{0}^{+} is the
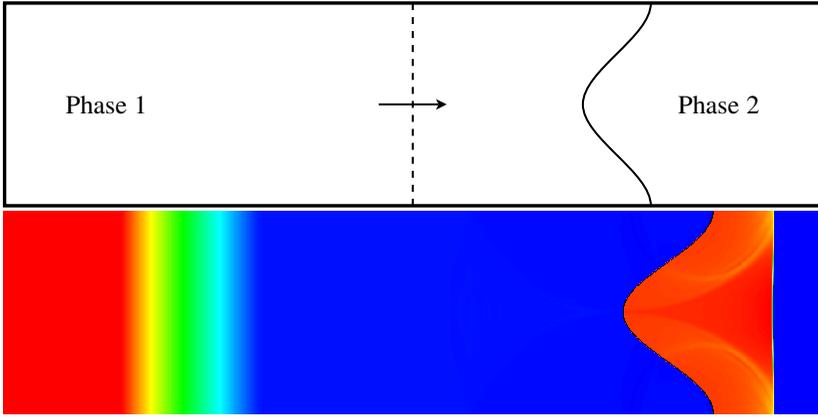
Figure 10. Shock channel initial configuration (top). Pressure profile at $t = 0.6$. The same color table as Figure 8 is used with limits [1.67, 3.0] for gas 1 and [1.0, 1.72] for gas 2 (bottom).

h	ρ	rate	ρu	rate
1/32	8.91×10^{-3}		8.24×10^{-3}	
1/64	4.02×10^{-3}	1.15	3.63×10^{-3}	1.18
1/128	2.17×10^{-3}	0.88	1.89×10^{-3}	0.94
1/256	1.16×10^{-3}	0.91	9.82×10^{-4}	0.95

h	ρv	rate	E	rate
1/32	1.87×10^{-3}		2.50×10^{-2}	
1/64	9.09×10^{-4}	1.04	1.08×10^{-2}	1.22
1/128	5.21×10^{-4}	0.80	5.56×10^{-3}	0.95
1/256	2.77×10^{-4}	0.91	2.77×10^{-3}	1.00

Table 8. Error in L_1 and convergence rate for the shock channel problem.

postshock initial perturbation amplitude. For our simulation we have $k = 2\pi/\lambda$, $\rho_1^+ = 2.07 \times 10^{-3}$, $\rho_2^+ = 9.05 \times 10^{-3}$, $A^+ = (\rho_2^+ - \rho_1^+)/(\rho_2^+ + \rho_1^+) = 0.63$, $\Delta V_f = 6356.24$, and $\eta_0^+ = 0.15$ in CGS units. The calculated amplitude growth using Equation (63) is $d\eta/dt = 624.82$ cm/s.

From the simulation results we plotted the amplitude of the interface in its linear regime (Figure 12). A linear fit shows a growth rate of $d\eta/dt = 606.64$ cm/s, while the amplitude growth rate from [12] is 628.64 cm/s. The simulation results show a good match ($\approx 3\%$ difference) with experimental and analytical results.

After the initial linear regime of the RMI test, where crests and troughs are symmetric, the interface grows nonlinearly. It becomes visible by the appearance of a bubble and spike, followed by the spike rolling up. The RMI simulation

	Gas 1: postshock	Gas 1: preshock	Gas 2
ρ (g/cm^3)	1.872×10^{-3}	1.351×10^{-3}	5.494×10^{-3}
u (cm/s)	1.013×10^4	0	0
v (cm/s)	0	0	0
p (g/(cm.s^2))	1.453×10^6	9.650×10^4	9.650×10^4
γ	1.276	1.276	1.4

Table 9. Initial values for the RMI problem.

N_x	ρ	rate	ρu	rate
128	8.81×10^{-6}		1.49×10^{-1}	
256	3.91×10^{-6}	1.17	6.62×10^{-2}	1.17
512	3.05×10^{-6}	0.36	5.19×10^{-2}	0.35
1024	1.27×10^{-6}	1.27	2.64×10^{-2}	1.08
N_x	ρv	rate	E	rate
128	2.69×10^{-2}		1.83×10^4	
256	1.13×10^{-2}	1.25	8.26×10^3	1.15
512	5.90×10^{-3}	0.94	6.63×10^3	0.32
1024	3.65×10^{-3}	0.70	2.87×10^3	1.21

Table 10. Error in L_1 and convergence rate for RMI problem. N_x is the number of cells in the x direction.
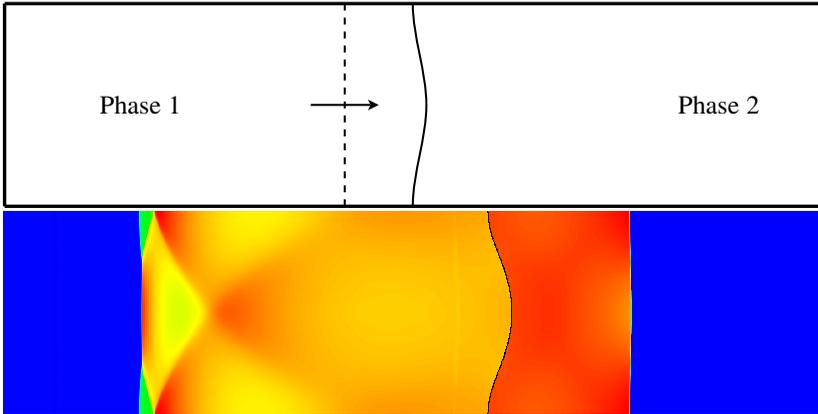
Figure 11. RMI problem initial configuration (top). Pressure profile at $t = 0.4$ ms. The same color table as Figure 8 is used with limits $[1.87, 2.11] \times 10^{-3}$ (g/cm^3) for gas 1 and $[5.49, 9.28] \times 10^{-3}$ (g/cm^3) for gas 2 (bottom).

is continued to time $t = 2.8$ ms to observe the nonlinear evolution of the front (Figure 13).
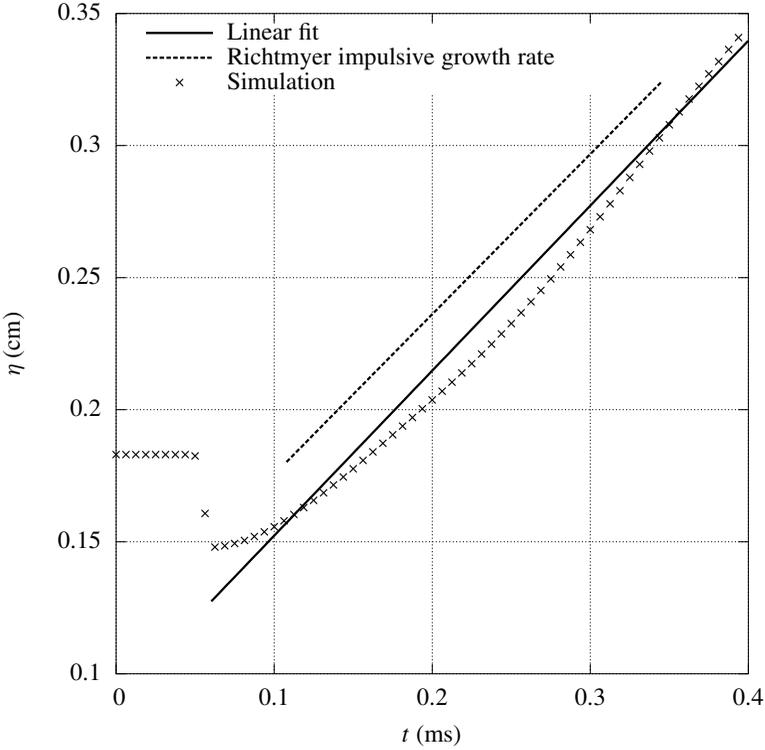
Figure 12. Amplitude of the perturbation on the front in the RMI simulation.
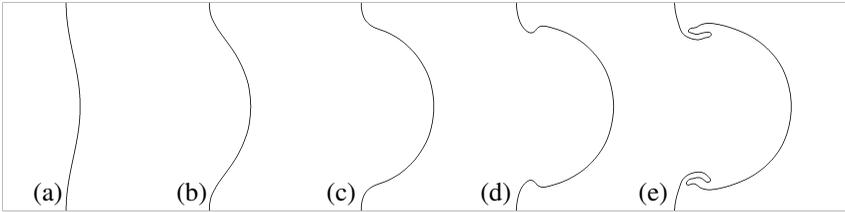
Figure 13. RMI problem front evolution. (a) $t = 0$, initial single mode perturbation; (b) $t = 0.7$ ms, linear growth of perturbation; (c) $t = 1.4$ ms, asymmetric growth of crests and troughs; (d) $t = 2.1$ ms, formation of bubble and spike; and (e) $t = 2.8$ ms, spike roll-up.

5.6. *Deforming interface — shock-bubble interaction.* The interaction of a shock wave in air with a bubble of helium has become popular for testing multiphase numerical methods. The experimental observations come from Haas and Sturtevant [20], and theoretical and numerical studies include [47; 48]. Numerical studies using this benchmark problem include [3; 15; 51; 55]. The initial conditions given in Table 11 induce an incident Mach 1.22 shock (Figure 14).
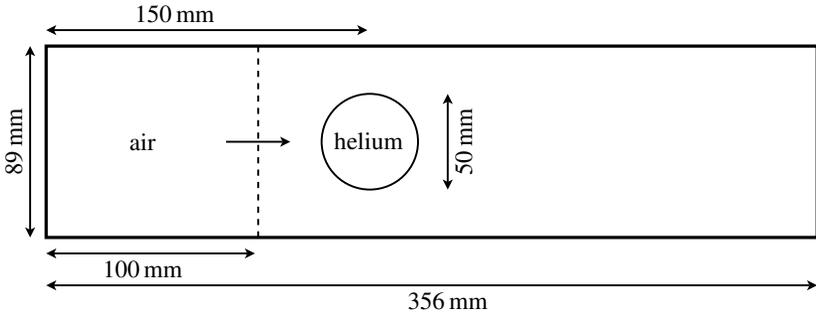
Figure 14. Initial configuration of the shock-bubble problem.

	Air: postshock	Air: preshock	Helium
ρ	1.3764	1.0	0.1819
u	0.336	0	0
v	0	0	0
p	1.1213	0.7142	0.7142
γ	1.4	1.4	1.648

Table 11. Initial values for the shock-bubble problem.

Using a 512 by 128 grid for the physical domain and CFL number 0.3, the simulation is done to $t = 427\,\mu$s. The results are shown in Figure 15. Comparing the density profiles to the experimental work of Haas and Sturtevant [20], and the adaptive mesh simulations of Quirk and Karni [48], our method accurately captures the dynamics of material interface, and the reflected and transmitted waves.

At later times in the RMI and shock-bubble simulations interface roll-ups are visible (spike roll-up in Figure 13e and jet roll-up in Figure 15j). When the roll-up happens on the material interface, the gradient of the level set becomes discontinuous within the 5×5 stencil of cells used to compute geometric information. The assumptions underlying the geometry algorithms are violated when this occurs so we terminate the simulation. With local grid refinement this loss of accuracy might be postponed. However, in these experiments perturbations grow at all scales so the loss of continuity will always occur.

6. Conclusion

We presented a new front-tracking method for contact discontinuities using the finite volume approach on a Cartesian grid. An essential feature of our method is the computation of geometry information to support space-time finite volume quadratures, using a sequence of discrete level sets [30; 36]. The level sets were advected using a velocity derived from interface-normal Riemann problems.
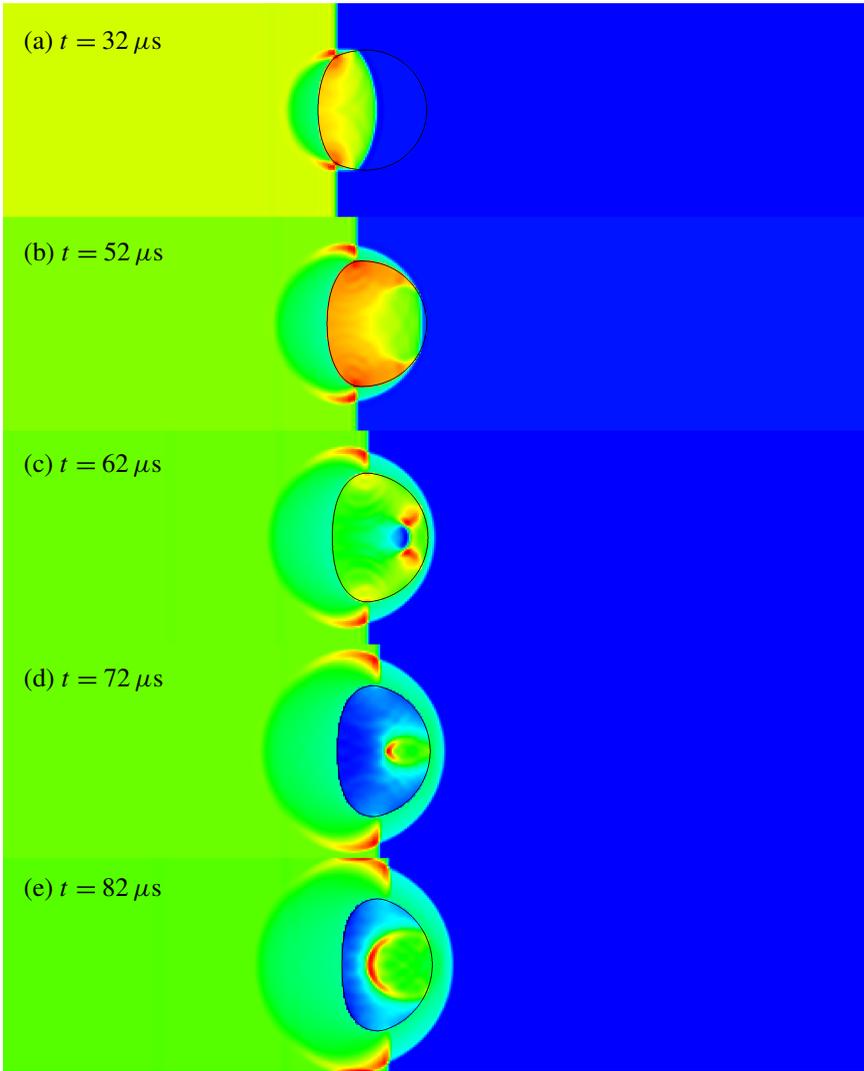
Figure 15. Density profiles of the shock-bubble test. The same color table as Figure 8 is used with limits (a)[0.997, 1.534], (b)[0.988, 1.607], (c)[0.999, 1.625], (d)[1.000, 1.623], and (e)[1.000, 1.646] for air and (a)[0.181, 0.221], (b)[0.178, 0.219], (c)[0.206, 0.221], (d)[0.215, 0.234], and (e)[0.218, 0.230] for helium.

A variety of convergence tests show that our method is second-order accurate in the L_1, L_2 and L_∞ norms, provided there are no shocks present. In the presence of a captured shock the convergence rate reduces to first-order in L_1.

It should be noted that many parts of the above algorithms may be implemented differently. For example the interpolation and extrapolation algorithms may include limiting operators [45] or different redistribution algorithms may be applied [11].
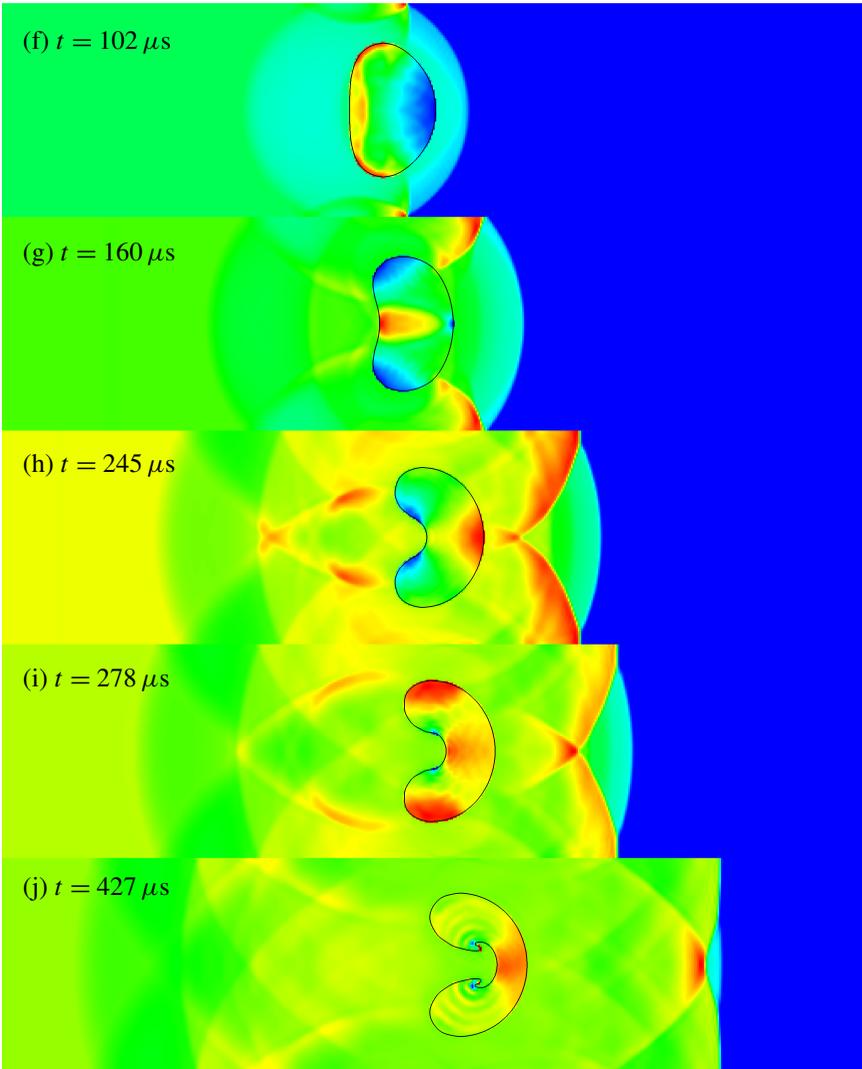
Figure 15. (continued) Density profiles of the shock-bubble test. The same color table as Figure 8 is used with limits (f)[1.000, 1.905], (g)[1.000, 1.664], (h)[1.000, 1.514], (i)[1.000, 1.554], and (j)[1.000, 1.584] for air and (f)[0.223, 0.231], (g)[0.231, 0.244], (h)[0.231, 0.238], (i)[0.230, 0.236], and (j)[0.229, 0.247] for helium.

Although such details may vary from case to case, we have shown that a second-order method is achievable if we consider the geometrical information and incorporate such data in the algorithm. While the level set methods we use are well established, there have been many new developments in this field too. Notable developments include the gradient-augmented level set method [40], distance regularized level set

Notation	Description
Υ_i	Cartesian (regular) cell
$V_{i,\alpha}^n$	Spatial irregular control volume for gas α at time n in cell Υ_i
$C_{i,\alpha}$	Space-time control volume for gas α in $\Upsilon_i \times [t^n, t^{n+1}]$
$(\vec{x}_i, t^n)$	Position of a cell center in space and time at time step n
$(\vec{x}_{i,\alpha}, t^n)$	Position of a cell centroid in space and time at time step n for gas α
$A_{i \pm \frac{1}{2}e^d}$	Cartesian (regular) face of $\Upsilon_i \times [t^n, t^{n+1}]$ in lower/higher side in direction d
$A_{i \pm \frac{1}{2}e^d,\alpha}$	Face of $C_{i,\alpha}$ in lower/higher side in direction d for gas α that coincide with Cartesian grid
A_i^f	Front face of $C_{i,\alpha}$
$(\vec{x}_{i \pm \frac{1}{2}e^d}, t^{n+\frac{1}{2}})$	Position of a face center in space and time for a regular face
$(\vec{x}_{i \pm \frac{1}{2}e^d,\alpha}, t_{i \pm \frac{1}{2}e^d,\alpha})$	Position of a face centroid in space and time for a irregular control volume for gas α
$(\vec{x}_i^f, t_i^f)$	Position of a front centroid position in space and time
$\boldsymbol{F}_{i \pm \frac{1}{2}e^d,\alpha}^{\mathrm{cr}}$	Flux at face center
$\boldsymbol{F}_{i \pm \frac{1}{2}e^d,\alpha}^{\mathrm{cd}}$	Flux at face centroid
$\boldsymbol{F}_{i,d,\alpha}^f$	Flux at front centroid

Table 12. Geometrical notation.

evolution [29], and the use of queuing algorithms [59] or hash table data structures for fast marching methods [6].

We validated our method with the simulation of single mode Richtmyer–Meshkov instability and shock-bubble interaction test. Simulation results are in good agreement with theoretical predictions and with experimental results.

Acknowledgements

References

[1]   D. Adalsteinsson and J. A. Sethian, *The fast construction of extension velocities in level set methods*, J. Comput. Phys. **148** (1999), no. 1, 2–22. MR 99j:65189 Zbl 0919.65074

[2]   J. B. Bell, P. Colella, and M. L. Welcome, *Conservative front-tracking for inviscid compressible flow*, 10-th AIAA Computational Fluid Dynamics Conference (A. D. Vakili and C. Gauthier, eds.), 1991, pp. 814–822.

[3]   J. Bell, M. Berger, J. Saltzman, and M. Welcome, *Three-dimensional adaptive mesh refinement for hyperbolic conservation laws*, SIAM J. Sci. Comput. **15** (1994), no. 1, 127–138. MR 95d:65070 Zbl 0793.65072

[4]   M. J. Berger, C. Helzel, and R. J. Leveque, *h-box methods for the approximation of hyperbolic conservation laws on irregular grids*, SIAM J. Numer. Anal. **41** (2003), no. 3, 893–918. MR 2004g:65103 Zbl 1066.65082

[5]   M. Brouillette, *The Richtmyer–Meshkov instability*, Annual review of fluid mechanics (J. L. Lumley, S. H. Davis, and P. Moin, eds.), no. 34, Annual Reviews, Palo Alto, CA, 2002, pp. 445–468. MR 2003c:76059 Zbl 1047.76025

[6]   E. Brun, A. Guittet, and F. Gibou, *A local level-set method using a hash table data structure*, J. Comput. Phys. **231** (2012), no. 6, 2528–2536. MR 2881030 Zbl 1242.65156

[7]   I.-L. Chern and P. Colella, *A conservative front tracking method for hyperbolic conservation laws*, Tech. Report LLNL Report No. UCRL-97200, Lawrence Livermore National Laboratory, 1987.

[8]   P. Colella, H. M. Glaz, and R. E. Ferguson, *Multifluid algorithms for Eulerian finite difference methods*, Tech. report, Lawrence Berkeley National Laboratory, 1996.

[9]   P. Colella, *Multidimensional upwind methods for hyperbolic conservation laws*, J. Comput. Phys. **87** (1990), no. 1, 171–200. MR 91c:76087 Zbl 0694.65041

[10] ———, *Volume-of-fluid methods for partial differential equations*, Godunov methods (E. Toro, ed.), Springer, New York, 2001, pp. 161–177. MR 1963590 Zbl 0989.65118

[11] P. Colella, D. T. Graves, B. J. Keen, and D. Modiano, *A Cartesian grid embedded boundary method for hyperbolic conservation laws*, J. Comput. Phys. **211** (2006), no. 1, 347–366. MR 2006i:65142 Zbl 1120.65324

[12] B. D. Collins and J. W. Jacobs, *PLIF flow visualization and measurements of the Richtmyer–Meshkov instability of an air/SF6 interface*, J. Fluid Mech. **464** (2002), 113–136.

[13] J. Du, B. Fix, J. Glimm, X. Jia, X. Li, Y. Li, and L. Wu, *A simple package for front tracking*, J. Comput. Phys. **213** (2006), no. 2, 613–628. MR 2006j:65306 Zbl 1089.65128

[14] A. du Chéné, C. Min, and F. Gibou, *Second-order accurate computation of curvatures in a level set framework using novel high-order reinitialization schemes*, J. Sci. Comput. **35** (2008), no. 2–3, 114–131. MR 2009e:65038 Zbl 1203.65043

[15] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher, *A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method)*, J. Comput. Phys. **152** (1999), no. 2, 457–492. MR 2000c:76061 Zbl 0957.76052

[16] C. Gatti-Bono, P. Colella, and D. Trebotich, *A second-order accurate conservative front-tracking method in one dimension*, SIAM J. Sci. Comput. **31** (2009/10), no. 6, 4795–4813. MR 2011k:65118 Zbl 05834168

[17] J. Glimm, E. Isaacson, D. Marchesin, and O. McBryan, *Front tracking for hyperbolic systems*, Adv. in Appl. Math. **2** (1981), no. 1, 91–119. MR 82i:76097 Zbl 0459.76069

[18] J. Glimm, X. L. Li, Y. Liu, and N. Zhao, *Conservative front tracking and level set algorithms*, Proc. Natl. Acad. Sci. USA **98** (2001), no. 25, 14198–14201. MR 1867514 Zbl 1005.65091

[19] J. Glimm, Y. Liu, Z. Xu, and N. Zhao, *Conservative front tracking with improved accuracy*, SIAM J. Numer. Anal. **41** (2003), no. 5, 1926–1947. MR 2004j:76109 Zbl 1053.35093

[20] J.-F. Haas and B. Sturtevant, *Interaction of weak shock waves with cylindrical and spherical gas inhomogeneities*, J. Fluid Mech. **181** (1987), 41–76.

[21] A. Harten, *High resolution schemes for hyperbolic conservation laws*, J. Comput. Phys. **49** (1983), no. 3, 357–393. MR 84g:65115 Zbl 0565.65050

[22] C. W. Hirt and B. D. Nichols, *Volume of fluid (VOF) method for the dynamics of free boundaries*, J. Comput. Phys. **39** (1981), no. 1, 201–225.

[23] R. L. Holmes, G. Dimonte, B. Fryxell, M. L. Gittings, J. W. Grove, M. Schneider, D. H. Sharp, A. L. Velikovich, R. P. Weaver, and Q. Zhang, *Richtmyer–Meshkov instability growth: experiment, simulation and theory*, J. Fluid Mech. **389** (1999), 55–79. MR 1707239 Zbl 0954.76026

[24] J. M. Hyman, *Numerical methods for tracking interfaces*, Physica D: Nonlinear Phenomena **12** (1984), no. 1–3, 396–407. Zbl 0604.65092

[25] G.-S. Jiang and D. Peng, *Weighted ENO schemes for Hamilton–Jacobi equations*, SIAM J. Sci. Comput. **21** (2000), no. 6, 2126–2143. MR 2001e:65124 Zbl 0957.35014

[26] M. A. Jones and J. W. Jacobs, *A membraneless experiment for the study of Richtmyer–Meshkov instability of a shock-accelerated gas interface*, Physics of Fluids **9** (1997), no. 10, 3078–3085.

[27] M. Latini, O. Schilling, and W. S. Don, *High-resolution simulations and modeling of reshocked single-mode Richtmyer–Meshkov instability: Comparison to experimental data and to amplitude growth model predictions*, Physics of Fluids **19** (2007), no. 2, 024104. Zbl 1146.76456

[28] R. J. LeVeque, *Finite volume methods for hyperbolic problems*, Cambridge University Press, 2002. MR 2003h:65001 Zbl 1010.65040

[29] C. Li, C. Xu, C. Gui, and M. D. Fox, *Distance regularized level set evolution and its application to image segmentation*, IEEE Trans. Image Process. **19** (2010), no. 12, 3243–3254. MR 2011m:94019

[30] T. J. Ligocki, P. O. Schwartz, J. Percelay, and P. Colella, *Embedded boundary grid generation using the divergence theorem, implicit functions, and constructive solid geometry*, Journal of Physics: Conference Series **125** (2008), no. 1, 012080.

[31] R. Malladi, J. Sethian, and B. Vemuri, *Shape modeling with front propagation: a level set approach*, IEEE Trans. Pattern Anal. and Mach. Intel. **17** (1995), no. 2, 158–175.

[32] R. Menikoff and B. J. Plohr, *The Riemann problem for fluid flow of real materials*, Rev. Modern Phys. **61** (1989), no. 1, 75–130. MR 90a:35142 Zbl 1129.35439

[33] K. O. Mikaelian, *Numerical simulations of Richtmyer–Meshkov instabilities in finite-thickness fluid layers*, Physics of Fluids **8** (1996), no. 5, 1269–1292. Zbl 1086.76018

[34] G. Miller, *Numerical analysis for engineers and scientists*, Cambridge University Press, 2014. MR 3290285 Zbl 1294.00002

[35] G. H. Miller and P. Colella, *A conservative three-dimensional Eulerian method for coupled solid-fluid shock capturing*, J. Comput. Phys. **183** (2002), no. 1, 26–82. MR 2003j:76080 Zbl 1057.76558

[36] G. H. Miller and D. Trebotich, *An embedded boundary method for the Navier–Stokes equations on a time-dependent domain*, Commun. Appl. Math. Comput. Sci. **7** (2012), no. 1, 1–31. MR 2893419 Zbl 1273.35215

[37] ———, *A front tracking embedded boundary method for two-fluid incompressible Navier–Stokes problems with surface tension*, preprint, 2014, submitted.

[38] G. H. Miller and E. G. Puckett, *A high-order Godunov method for multiple condensed phases*, J. Comput. Phys. **128** (1996), no. 1, 134–164. Zbl 0861.65117

[39] B. Motl, J. Oakley, D. Ranjan, C. Weber, M. Anderson, and R. Bonazza, *Experimental validation of a Richtmyer–Meshkov scaling law over large density ratio and shock strength ranges*, Physics of Fluids **21** (2009), no. 12, 126102. Zbl 1183.76364

[40] J.-C. Nave, R. R. Rosales, and B. Seibold, *A gradient-augmented level set method with an optimally local, coherent advection scheme*, J. Comput. Phys. **229** (2010), no. 10, 3802–3827. MR 2011a:65256 Zbl 1189.65214

[41] W. Noh and P. Woodward, *SLIC (simple line interface calculation)*, Proceedings of the fifth international conference on numerical methods in fluid dynamics (A. I. van de Vooren and P. J. Zandbergen, eds.), Lecture Notes in Physics, no. 59, Springer, Berlin, 1976, pp. 330–340.

[42] R. R. Nourgaliev, M.-S. Liou, and T. G. Theofanous, *Numerical prediction of interfacial instabilities: sharp interface method (SIM)*, J. Comput. Phys. **227** (2008), no. 8, 3940–3970. MR 2009d:76054 Zbl 1275.76164

[43] S. Osher and R. Fedkiw, *Level set methods and dynamic implicit surfaces*, Applied Mathematical Sciences, no. 153, Springer, New York, 2003. MR 2003j:65002 Zbl 1026.76001

[44] S. Osher and J. A. Sethian, *Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations*, J. Comput. Phys. **79** (1988), no. 1, 12–49. MR 89h: 80012 Zbl 0659.65132

[45] R. B. Pember, J. B. Bell, P. Colella, W. Y. Crutchfield, and M. L. Welcome, *An adaptive Cartesian grid method for unsteady compressible flow in irregular regions*, J. Comput. Phys. **120** (1995), no. 2, 278–304. MR 96d:76081 Zbl 0842.76056

[46] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang, *A PDE-based fast local level set method*, J. Comput. Phys. **155** (1999), no. 2, 410–438. MR 2000j:65104 Zbl 0964.76069

[47] J. M. Picone and J. P. Boris, *Vorticity generation by shock propagation through bubbles in a gas*, J. Fluid Mech. **189** (1988), 23–51.

[48] J. J. Quirk and S. Karni, *On the dynamics of a shock-bubble interaction*, J. Fluid Mech. **318** (1996), 129–163. Zbl 0877.76046

[49] R. D. Richtmyer, *Taylor instability in shock acceleration of compressible fluids*, Comm. Pure Appl. Math. **13** (1960), 297–319. MR 22 #5275

[50] V. Sabelnikov, A. Y. Ovsyannikov, and M. Gorokhovski, *Modified level set equation and its numerical assessment*, J. Comput. Phys. **278** (2014), 1–30. MR 3261080

[51] R. Saurel and R. Abgrall, *A simple method for compressible multifluid flows*, SIAM J. Sci. Comput. **21** (1999), no. 3, 1115–1145. MR 2001b:65089 Zbl 0957.76057

[52] J. A. Sethian, *Level set methods and fast marching methods*, 2-nd ed., Cambridge Monographs on Applied and Computational Mathematics, no. 3, Cambridge University Press, 1999. MR 2000c:65015 Zbl 0973.76003

[53] G. A. Sod, *A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws*, J. Comput. Phys. **27** (1978), no. 1, 1–31. MR 58 #13770 Zbl 0387.76063

[54] M. Sussman, P. Smereka, and S. Osher, *A level set approach for computing solutions to incompressible two-phase flow*, J. Comput. Phys. **114** (1994), no. 1, 146–159. Zbl 0808.76077

[55] H. Terashima and G. Tryggvason, *A front-tracking/ghost-fluid method for fluid interfaces in compressible flows*, J. Comput. Phys. **228** (2009), no. 11, 4012–4037. Zbl 1171.76046

[56] E. F. Toro, *Riemann solvers and numerical methods for fluid dynamics*, 2-nd ed., Springer, Berlin, 1999. MR 2000f:76091 Zbl 0923.76004

[57] M. Vahab, *A front-tracking shock-capturing method for two fluids*, Ph.D. thesis, University of California, Davis, 2014, p. 63. MR 3295245

[58] M. Vetter and B. Sturtevant, *Experiments on the Richtmyer–Meshkov instability of an air/SF$_6$ interface*, Shock Waves **4** (1995), no. 5, 247–252.

[59] L. Yatziv, A. Bartesaghi, and G. Sapiro, *O(N) implementation of the fast marching algorithm*, J. Comput. Phys. **212** (2006), no. 2, 393–399. Zbl 1083.65083

[60] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher, *A variational level set approach to multiphase motion*, J. Comput. Phys. **127** (1996), no. 1, 179–195. MR 97g:80013 Zbl 0957.76057

MEHDI VAHAB: mvahab@fsu.edu
Department of Mathematics, Florida State University, Tallahassee, FL 32306

GREGORY H. MILLER: grgmiller@ucdavis.edu
Department of Chemical Engineering and Materials Science, University of California, Davis, 1 Shields Avenue, Davis, CA 95616, United States

msp

IDENTIFYING TURBULENT STRUCTURES THROUGH TOPOLOGICAL SEGMENTATION

PEER-TIMO BREMER, ANDREA GRUBER,
JANINE C. BENNETT, ATTILA GYULASSY, HEMANTH KOLLA,
JACQUELINE H. CHEN AND RAY W. GROUT

A new method of extracting vortical structures from a turbulent flow is proposed whereby topological segmentation of an indicator function scalar field is used to identify the regions of influence of the individual vortices. This addresses a long-standing challenge in vector field topological analysis: indicator functions commonly used produce a scalar field based on the local velocity vector field; reconstructing regions of influence for a particular structure requires selecting a threshold to define vortex extent. In practice, the same threshold is rarely meaningful throughout a given flow. By also considering the topology of the indicator field function, the characteristics of vortex strength and extent can be separated and the ambiguity in the choice of the threshold reduced. The proposed approach is able to identify several types of vortices observed in a jet in cross-flow configuration simultaneously where no single threshold value for a selection of common indicator functions appears able to identify all of these vortex types.

1. Introduction

Defining and extracting vortices from a turbulent flow is a long standing challenge with implications for a wide variety of applications including study of turbulent boundary layers (e.g., del Álamo et al. [7]) and turbulence-chemistry interaction (e.g., Grout et al. [12]). Several intuitive descriptions of a vortex have been articulated. An early description of a vortex was as a coherent volume of material spinning around a common core [19]. A more recent description is as a coherent structure of the turbulent flow field; a connected, large-scale fluid mass with phase-correlated vorticity over its spatial extent [15]. Directly extracting vortices based on such definitions has proven practically infeasible especially in the complex, time-dependent flows of greatest interest. Instead, a number of indicator functions have been proposed such as vorticity magnitude [1; 9], eigenvalues of the velocity gradient tensor [6], the second invariant of the velocity gradient tensor [14; 18],

and the λ_2 criterion [16]. Given any of these derived quantities, vortices are defined using isosurfaces at a particular function threshold. While each indicator function has particular advantages and disadvantages they share two common problems. First, selecting a threshold determines both the strength of the vortices one is interested in as well as their spatial extent. A single threshold is rarely meaningful throughout the entire flow and a given threshold selection can have undesirable consequences. For example, when a weak threshold is selected, high strength vortices can balloon and ultimately merge erroneously. Conversely, when a strong threshold is selected it can cause lower-strength vortices to be ignored. Second, the common indicator functions are known to have a large dynamic range in which case the absolute strength of a vortex may not necessarily be equivalent to its importance. For example, in our test data involving a flame in a turbulent jet in cross flow configuration, the jet break-down region contains significantly stronger vortices in absolute terms than the wake vortices below. However, the wake vortices are an essential part of the description of the overall flow pattern. Combined, these two issues lead to the dilemma of choosing a strong threshold that ignores many important features or a weak one which includes a large number of spurious and/or potentially severely distorted features (see Figures 3 and 4).

In practice, there typically does not exist an optimal threshold, which leads to frequent criticism about the influence of the ultimate choice of threshold on the results of a subsequent analysis [16]. Álamo et al. [7] found this problem particularly frustrating in their analysis of a turbulent boundary layer. In order to compute the volume fraction occupied by vortex regions across a boundary layer, they noted that threshold values of their indicator function suitable for the near wall region detected few vortices in the outer part of the boundary layer. Furthermore, when suitable thresholds for the outer part were selected, the near wall region became cluttered and obfuscated. To mitigate this effect, they scaled the threshold used based on the standard deviation of their indicator function across planes parallel to the wall (i.e., their threshold was dependent on the distance from the wall). This approach was very effective although it depended on leveraging the single direction of inhomogeneity in their specific configuration.

To avoid ambiguities associated with ad hoc vortex extraction methods this paper proposes a general, simple, yet highly flexible vortex detection technique based on the topological analysis of an arbitrary scalar indicator function. A key advantage of our approach is that it decouples the identification of a vortex from the specification of its extent. We first find seeds for all potential vortices as local maxima/minima of the indicator function. Subsequently, we use a topological encoding of the indicator function to optimize a local threshold for each vortex according to some user defined criterion such as size or volume constraints, relative thresholds, or deviations from some assumed model. The resulting vortices can then be explicitly included or

excluded from any analysis depending on various selection criteria such as size, strength, location, or shape. Our contributions in detail are:

- a new methodology for defining and extracting vortices using topological encodings;

- using a simple but effective metric called *relevance* to define vortices in terms of a relative threshold that is scaled according to the per-vortex extrema rather than the global indicator range; and

- a case study using a complex, highly turbulent flow that demonstrates the effectiveness of the relevance metric in extracting well-formed, nonoverlapping vortices from many standard indicator functions while requiring no a priori knowledge of the flow geometry.

The remainder of this paper is organized as follows: we first introduce the necessary mathematical concepts and show how a broad range of vortex detection techniques can benefit from topological encodings. We then introduce *relevance* as an example of a simple to compute metric designed to improve upon traditional thresholding. After a brief survey of common indicator functions, we show how the relevance metric and topological segmentation improve on simple thresholding for a representative direct numerical simulation (DNS) dataset. We conclude with a discussion of the generality of the approach and future work.

2. Topological segmentations

This section introduces some basic concepts from scalar topology used throughout the discussion. We then show how vortices defined through any indicator function are equivalent to so called sub- or superlevel sets of the indicator and how these can be efficiently encoded in a merge tree. The advantage of the topological techniques is that they partition the domain into isolated features independent of feature strength. Finally, we define a general class of tree transforms that lead to a notion of localized thresholds and define *relevance* as a particularly simple yet powerful transform to define vortex extents.

Merge trees. Given a connected domain $\mathbb{M}$ and a smooth function $f : \mathbb{M} \to \mathbb{R}$ the region $\text{Sup}_f(c) = \{p \in \mathbb{M} \mid f(p) \geq c\}$ of $\mathbb{M}$ with function value greater c is called the *superlevel set* of c. We call a connected component of a superlevel set a *supercontour*. The merge tree of f encodes the evolution of the supercontours as c is swept from ∞ to $-\infty$. Each time c passes a local maxima of f a new supercontour is created and supercontours merge at selected saddles sometimes referred to as *merge-saddles*. Collectively, this structure is typically represented as a tree with local maxima forming the leaves, merge saddles the internal nodes, and the global minimum of f the root (see Figure 1(a)). The *arcs* of this tree
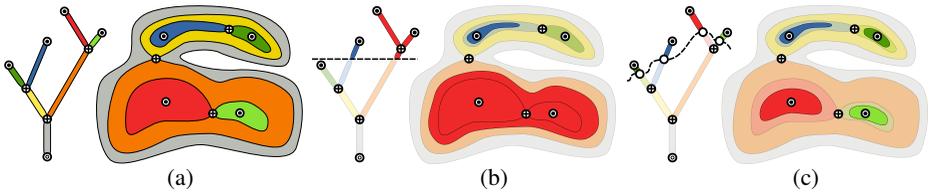
Figure 1. (a) A merge tree forms a compact encoding of an indicator function. It encodes the merging of contours as the function is lowered through its range. Each branch represents a portion of the domain as indicated by the colors. (b) The selection of a threshold is equivalent to performing a horizontal cut through the tree. Each subtree corresponds to a vortex. (c) Rather than use a single global threshold, localized thresholds can be used to identify the extents of vortices.

represent evolving supercontours that may change their shape but do not touch other supercontours. Equivalently, the *sublevel sets*, $\mathrm{Sub}_f(c) = \{p \in \mathbb{M} \mid f(p) \leq c\}$ give rise to the *split-tree* containing local minima as leafs, split-saddles as internal nodes, and the global maximum as root.

Given an indicator function I (see Section 3) vortices are traditionally defined using a threshold c as regions with the $I \geq c$ or $I \leq c$, or equivalently as the super-/subcontours of I. The following discussion will assume the former case, concentrating on superlevel sets but the results for sublevel sets are symmetric.

A merge tree that encodes the indicator function topology forms a highly compact representation of vortices at all thresholds [2]. Given the duality between vortices and supercontours it is convenient to think of choosing a threshold as a cut through the tree. More specifically, consider laying out the tree vertically according to the function values of the nodes. The selection of a threshold is equivalent to drawing a horizontal line through the tree cutting it into a forest of subtrees (Figure 1(b)). Each subtree corresponds to a vortex, and therefore, finding a good threshold is equivalent to finding an appropriate cut through the tree. The challenge of finding a global horizontal cut through the tree nicely illustrates the issues that can arise when choosing a global threshold to define all vortices. Selecting a high cut will miss all lower branches of the tree, yet moving the cut too low will result in overly large (or rather high) subtrees that no longer represent meaningful vortices. Instead, we propose to use the flexibility of merge trees to define *local* thresholds.

Local thresholds. The main insight from the discussion above is that there is no need to use a horizontal cut to define vortices. In fact any disjunct set of subtrees will define a set of vortices, where each vortex is given by the supercontour that contains the root of its corresponding subtree. Therefore, selecting subtrees is one convenient way of defining *local* thresholds. Nevertheless, the problem of finding a "good" set of local thresholds remains challenging.

As mentioned in Section 1, traditionally the threshold performs two important tasks. It defines which regions of the data set are considered "vortex-like" and it determines the spatial extent of a vortex. For global thresholds these two choices are inextricably linked resulting in two competing and sometimes mutually exclusive sets of goals. Local thresholds allow us to ignore the first condition of which regions to consider. Instead, we initially consider all possible vortices as identified by all leaves in the merge tree, i.e., all regions with a local maximum of the indicator. Conceptually, one can consider this a cut that runs just below all leaves of the tree which corresponds to a (maximal) set of tiny, "seed" vortices. Subsequently, we can determine the spatial extent of each of these vortices by lowering the local threshold, which effectively grows and merges vortices. This results in an arbitrary cut of the tree which intersects the path from each leaf to the root exactly once (Figure 1(c)). Since in this manner we are guaranteed to not miss a potential vortex one can use an arbitrary user defined metric to define the "optimal" spatial extent and the corresponding local threshold. As well be demonstrated later, the results tend to be insensitive to the choice of the metric.

Metric indicators. Defining a cut as described above provides a highly flexible means to select vortices. For example, one could use a goodness of fit to an idealized structure, such as an Oseen vortex to identify well formed structures. However, the merge tree also allows to go beyond identifying a single set of vortices. In particular, rather than finding an "optimal" cut point along each path from a leaf to the root one can evaluate a given metric for all possible subtrees. As long as these values respect the structure of the merge tree, i.e., each subtree being assigned a higher metric than its parent, the metric itself effectively defines a new indicator function. Conceptually, assigning values to all subtrees defines not just a single cut through the tree but a nested set of cuts according to the metric. One can now "straighten" these cuts by mapping the original indicator values of all the nodes of the tree to the metric values. Furthermore, since there exists a one-to-one correspondence of the merge tree branches to the original domain, the mapping of the tree induces a mapping of the original indicator field to a new function. The practical advantage of this approach is that in the new *metric indicator* traditional isosurfaces correspond to a potentially highly sophisticated feature definition but can be extracted with any traditional tool and for various metric values. Note that the metric indicator does not need to be smooth or even continuous as long as it is monotone. For example, evaluating an arbitrary metric and then inflating or deflating its values to enforce monotonicity is a perfectly valid approach.

Relevance. To demonstrate the effectiveness of the general approach we show how *relevance*—a metric initially introduced to study extinction regions in a turbulent flame [17]—can be described in this framework and used to improve

vortex detection. Relevance computes a local threshold as the relative difference in the indicator with respect to the local maximum. More formally, for every node n in the tree its relevance $R(n)$ with respect to an indicator I is defined using the relative difference in indicator between a node and the local maximum in its subtree

$$R(n) = 1 - \frac{\max\{I(x) \mid x \in \text{subtree}(n)\} - I(n)}{\max\{I(x) \mid x \in \text{subtree}(n)\} - \min(I)}.$$

Intuitively, relevance describes the relative strength of a vortex compared to its neighborhood. It is not smooth as the local maximum changes discretely at saddle points but it is naturally monotone: Assume that m, n are two nodes of the merge tree part of the same subtree with $I(n) < I(m)$. Let M_n and M_m be the local maxima of n and m respectively. It follows that $I(M_n) \geq I(M_m)$ which leads to $R(n) \leq R(m)$:

$$\begin{aligned} 1 - R(m) = \frac{I(M_m) - I(m)}{I(M_m) - \min(I)} &\leq \frac{I(M_m) - I(n)}{I(M_m) - \min(I)} \\ &\leq \frac{I(M_n) - I(n)}{I(M_n) - \min(I)} \\ &\leq \frac{I(M_n) - I(n)}{I(M_n) - \min(I)} = 1 - R(n). \end{aligned}$$

As will be demonstrated below, this identifies a significantly larger number of well-formed and well-separated vortices than traditional techniques as vortices of all strengths are identified using the same relevance. In practice, this set would likely be filtered by volume, strength or other secondary criteria. While relevance has proven highly useful [17] and simple to compute, other metrics could be substituted. For example, one could extract vortices of equal volume or compute other integrated properties to define local thresholds.

Implementation. A reference implementation of the local thresholds including the relevance transform is publicly available as part of the adaptive thresholds for feature extraction (ADAPT) package at http://github.com/scalability-llnl/ADAPT. The framework uses a variant of [3] which first sorts all vertices according to their indicator function followed by global union-find to construct the tree. One advantage of this scheme is that it produces a *fully augmented* tree, meaning one that contains not just critical points but all vertices. While this is slightly less efficient (especially in terms of memory) it allows one to compute the chosen metric, (i.e., relevance) for all vertices in the original field. For convenience ADAPT directly outputs the transformed field of the metric indicator which can then be analyzed and processed with any of the standard visualization or analysis tools without modifications.

3. Scalar fields for identifying vortices

For convenience and precision, we turn now to a brief specification of the indicator functions that we extract from the dataset and use to explore the segmentation technique in the next section. These arise from a decomposition of the velocity gradient tensor $\nabla \vec{u} = \mathbf{S} + \mathbf{\Omega}$ into a symmetric part (rate-of-strain tensor),

$$\mathbf{S} = \tfrac{1}{2}[\nabla\vec{u} + (\nabla\vec{u})^T], \tag{1}$$

and an antisymmetric one (vorticity tensor),

$$\mathbf{\Omega} = \tfrac{1}{2}[\nabla\vec{u} - (\nabla\vec{u})^T]. \tag{2}$$

The "invariants" of $\nabla\vec{u}$ [4] are

$$P = u_{ii}, \quad Q = \tfrac{1}{2}(u_{ii}^2 - u_{ij}u_{ji}) = -\tfrac{1}{2}u_{ij}u_{ji}, \quad R = \det(u_{ij}). \tag{3}$$

Several indicator functions have been proposed; Chakraborty et al. [4] explored the connections between several of these and showed that they are closely related. In the following sections, we will consider several of these commonly referred to as: the Q criterion, the λ_2 criterion, the Δ criterion, and the vorticity magnitude. We leave aside others such as the kinematic vorticity number, and the $\mathbf{M}_Z$ criterion, proposed by Haller [13]. The $\mathbf{M}_Z$ criterion defines a vortex based on Lagrangian considerations; while interesting, here we are focused on Eulerian measures.

The Q criterion. The second invariant, indicating the rotational character, is often used as an indicator of vorticity alone. In compressible flows it is also appropriate to include a pressure criterion, defining a vortex as a region where $Q > 0$ and pressure is less than ambient pressure. As discussed by Haller [13], in two dimensions this reduces to an elliptic version of the "Okubo–Weiss" criterion.

The λ_2 criterion. Proposed by Jeong and Hussain [16], the λ_2 criterion defines a vortex core as a connection region with two negative eigenvalues of the symmetric tensor $(\mathbf{S}^2 + \mathbf{\Omega}^2)$. This is equivalent to saying a vortex exists in regions where $\lambda_2 < 0$, and λ_2 is the intermediate eigenvalue of $(\mathbf{S}^2 + \mathbf{\Omega}^2)$. Although this appears unambiguous, in practice the field is noisy and an alternate threshold is often used. In proposing the λ_2 criterion, the inadequacy of earlier definitions was discussed. Some examples are that the that the $|\omega|$-definition is inappropriate for wall layer flows, that the $|\omega|$-definition can not simultaneously capture rib vortices following an elliptic vortex ring and the main structure due to the large variation in vorticity, and that the Δ-definition (see below) is unable to properly identify a conically symmetric vortex. Jeong et al. also considered various DNS data sets (e.g., of a temporal mixing layer and of a circular jet) to show the improvements made by the λ_2 criterion.

The Δ criterion. The Δ criterion, proposed by Chong et al. [6], defines a vortex as *the region of space where the vorticity is sufficiently strong to cause the rate-of-strain tensor to be dominated by the rotation tensor, i.e., the rate-of-deformation tensor has complex eigenvalues*. This occurs when the discriminant is positive:

$$\Delta = \left(\tfrac{1}{3}Q\right)^3 + \left(\tfrac{1}{2}R\right)^2 > 0. \tag{4}$$

Vorticity. The vorticity tensor of the velocity field in our DNS is given by

$$\omega_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i}\right), \tag{5}$$

and the enstrophy by

$$\Omega = 2\omega_{ij}\omega_{ij}. \tag{6}$$

The topology for the enstrophy is matched by the vorticity magnitude which we operate on in the following section where the vorticity magnitude is defined as $|\omega| = \left(\tfrac{1}{2}\Omega\right)^{0.5}$.

4. Results

This section will first introduce the test DNS dataset in more detail before comparing vortices identified through relevance with the traditional global thresholds for different indicator functions.

4.1. *Test dataset: reacting turbulent jet in crossflow.* The test dataset we have chosen is a turbulent reacting jet-in-crossflow configuration described in detail by Grout et al. [11] and depicted in Figure 2. The dataset was constructed using a fully resolved direct numerical simulation (DNS) of a turbulent boundary layer of air with a nitrogen-diluted hydrogen jet introduced into the domain through a nozzle hole on the wall. The domain size was $25 \times 20 \times 20$ mm and a Cartesian grid with $1408 \times 1080 \times 1100$ points was used. The simulation was conducted using S3D [5], a finite-difference code which solves the compressible reacting Navier–Stokes equations with multispecies transport and chemical reaction. The turbulent boundary layer inflow conditions for the three spatial components of the velocity vector were developed by an auxiliary simulation of a periodic flow above a flat plate driven by a pressure gradient. This solution was sampled on a fixed wall-normal planar section of the cross flow that was then used to specify the inflow velocity for the main simulation.

Of note, in addition to the turbulent boundary layer structures, the flow features the jet-in-cross-flow vortical structures identified by Fric [8]: the horseshoe vortices upstream of the jet, the wake vortices between the wall and plume downstream of the jet, the jet shear-layer vortices formed between the jet and the crossflow and the
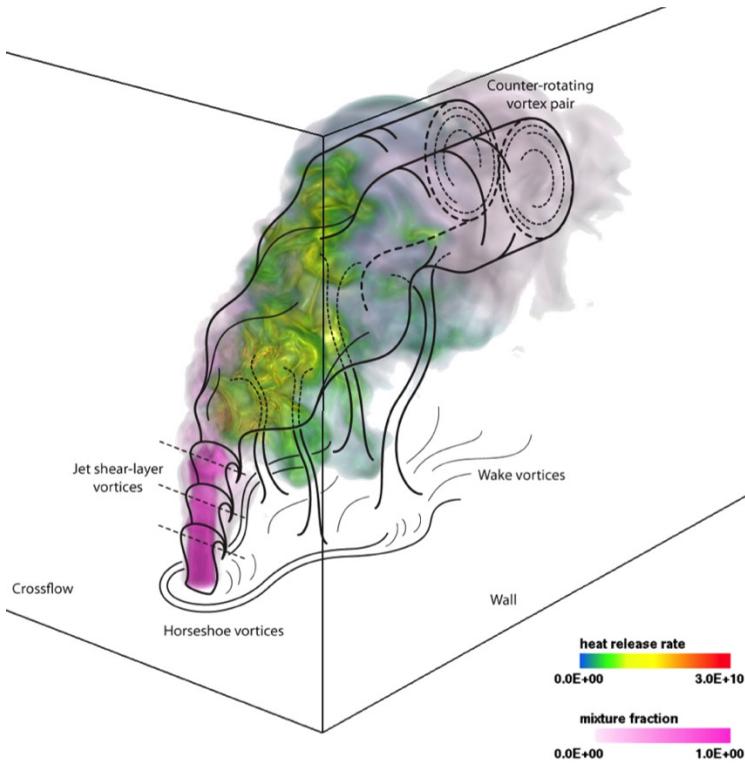
Figure 2. Jet in cross flow test DNS data set [11].

counter-rotating vortex pair in the far-field. In addition, the intense turbulence and vortical structure present in the jet break-down region is interacting with the heat-release driven expansion (dilatation) from the presence of a flame anchored in the jet [11]. Partially due to the wide dynamic range and multitude vortices embedded in vortices (as in practical turbulent flows), this dataset has resisted analysis using common indicator functions. The approach proposed herein is a formalization of the analysis presented by Grout et al. [10] where topological segmentation of indicator function scalar fields was used to identify vortices for qualitative comparison to the flame position.

4.2. *Comparison to previous approaches.* Here we compare vortices defined using relevance to determine local thresholds with vortices defined through different global thresholds. It is important to point out that, at their core, merge trees simply allow a convenient abstraction to choose local thresholds. Ultimately, they still depend on the chosen indicator function and inherit all its advantages or disadvantages. Obviously, for any vortex that can be defined using a local threshold, there exists the corresponding global threshold and vice versa. However, what merge trees provide
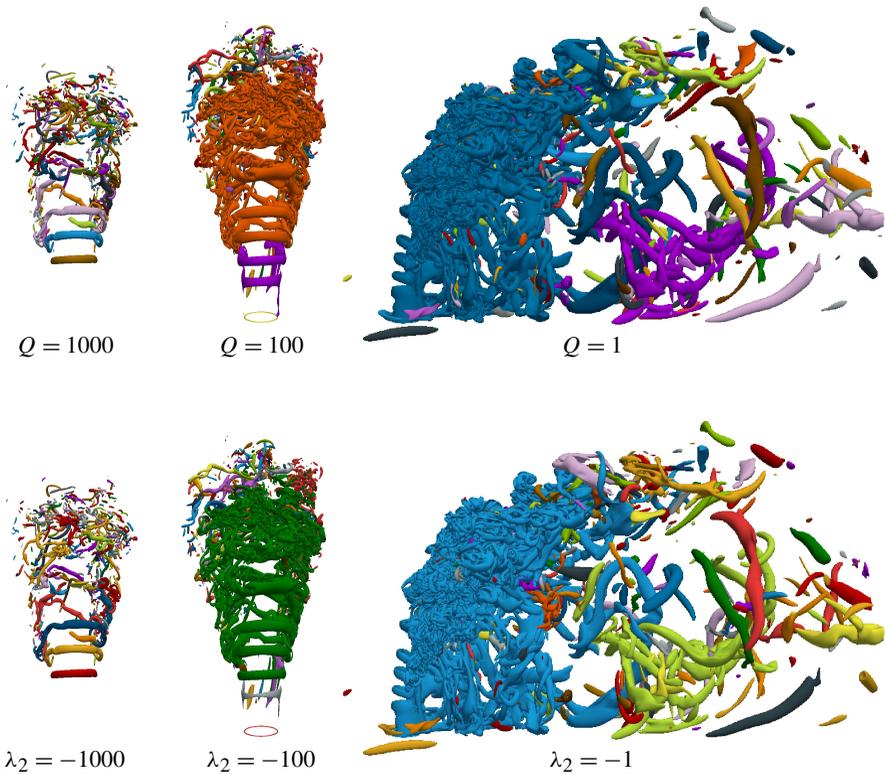
Figure 3. Vortices extracted using Q and λ_2 indicator functions at different thresholds. Vortices are randomly assigned one of 27 colors.

is a means to *simultaneously* extract all vortices of interest rather than having to explore a wide range of global thresholds. Here we use relevance as a particularly simple metric, but, as discussed above, other approaches could be substituted.

Figures 3 and 4 show vortices extracted using different indicators at various global thresholds. While some are clearly better than others at finding well formed vortices all indicators exhibit the same general behavior. In order to correctly identify the strong shear-layer vortices and those in the center of the flame, very high thresholds are required. However, these exclude all weaker vortices of interest. Yet, lowering the threshold results in the entire breakdown region being erroneously identified as a single vortex. Instead, relevance enables one to extract vortices of all strengths simultaneously as shown in Figure 5. The separation of identification of a vortex existence and strength from any enlargement due to strength is possible because the relevance field encodes local information. At any arbitrary location in the global relevance field, the value indicates the vortex strength *relative to the associated local extrema*. This theoretically limits the highest value of relevance
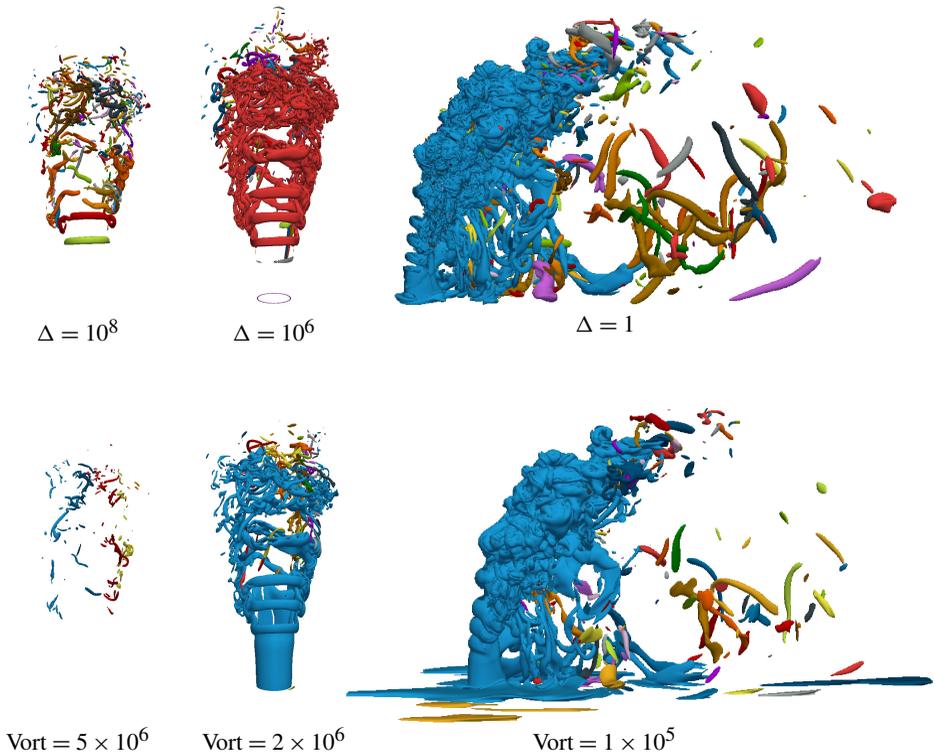
Figure 4. Vortices extracted using the Δ and Vorticity indicator functions at different thresholds. Vortices are randomly assigned one of 27 colors.

(largest vortex extent) that can be selected without ambiguity about which vortex is identified, but in practice we have found that all of the indicator functions drop to values far from the local maxima/minima between vortices so this does not seem to pose a problem. This is supported quantitatively by Figure 6, where the number of vortices identified is reasonably insensitive to the relevance threshold over a wide range. Figure 6 also shows the number of vortices each indicator identifies for a range of thresholds for the raw indicator function (i.e., not processed by our algorithm). For the global indicator function thresholds, the number of vortices identified varies significantly and nonmonotonically: slight changes in the threshold cause unpredictable changes in which vortices are considered distinct.

The relevance plots all show the expected behavior of monotonically decreasing vortex counts as a zero relevance identifies all possible seed vortices and lower thresholds progressively merge seeds into more realistic structures. This merging of seeds at low relevance also serves to eliminate numerical noise. The absolute thresholds on the other hand all show significantly fewer individual vortices for most

Q criterion at relevance 0.8

λ_2 criterion at relevance 0.8

Δ criterion at relevance 0.8

Vorticity criterion at relevance 0.8
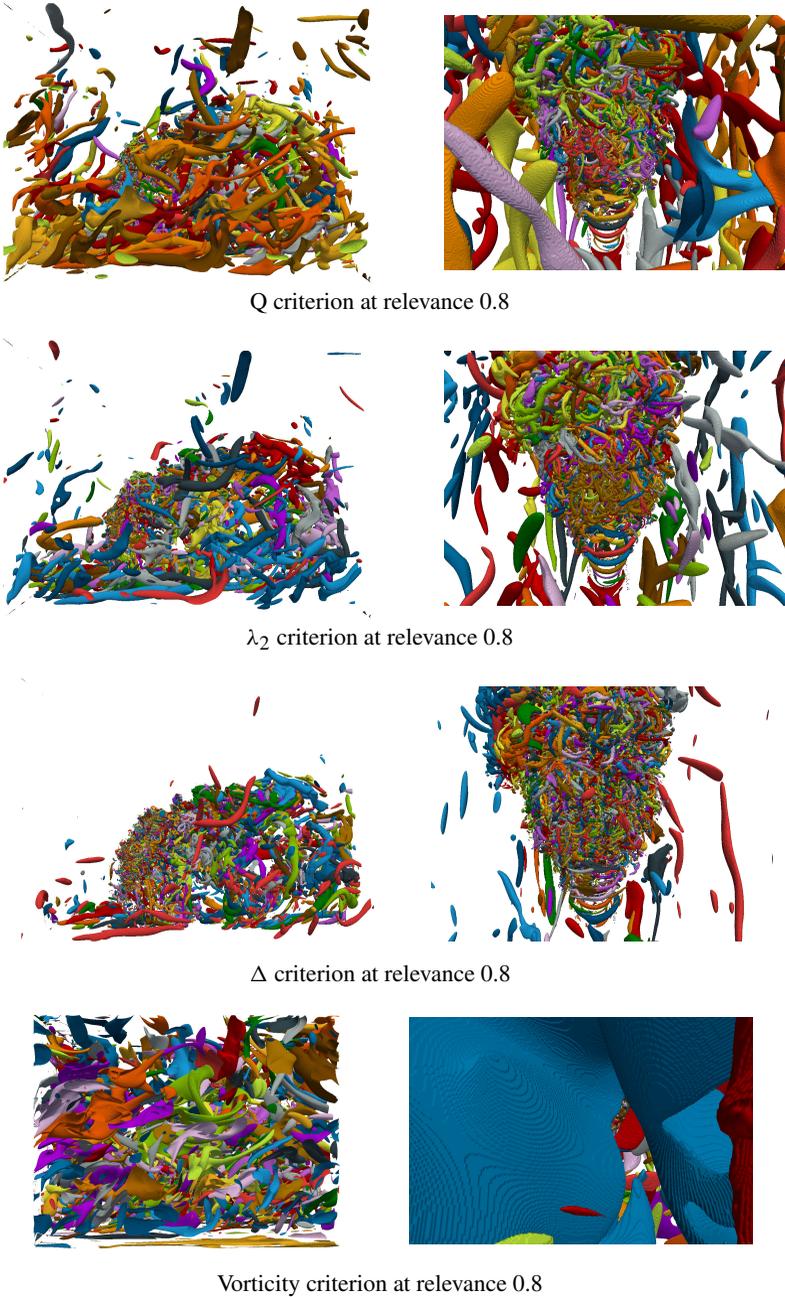
Figure 5. Side (left) and zoomed-in front view (right) of the vortices extracted at relevance 0.8 for the Q, λ_2, Δ and Vorticity indicator indicator functions. Both the shear-layer and central flame vortices as well as the low strength vortices on the side and in the back are well separated. Δ produces similar results to Q and λ_2 but the vorticity vortices tend to be distorted and pancake-like rather tube-shaped.
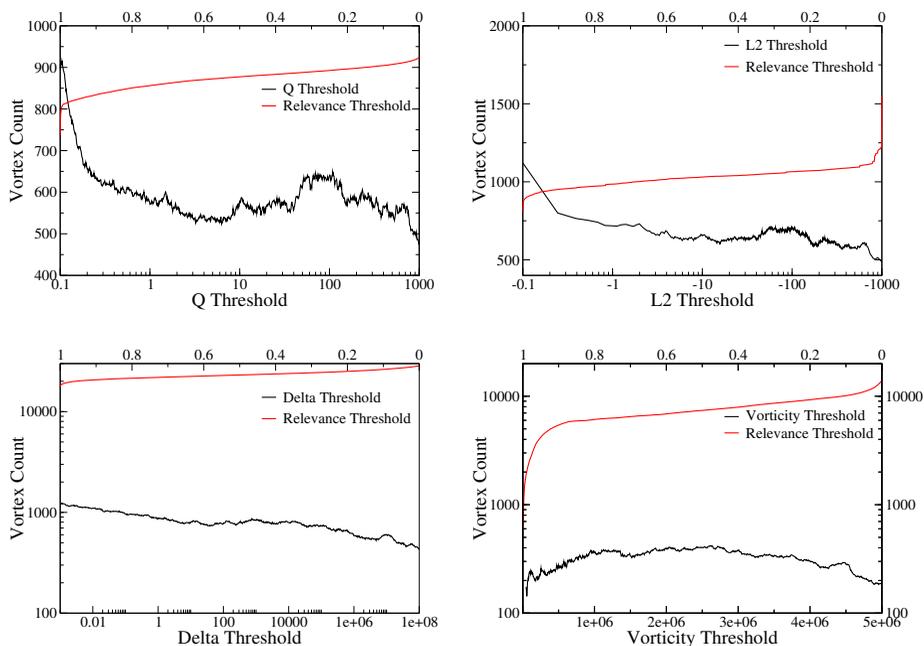
Figure 6. Sensitivity of number of vortices identified to a single scalar threshold and the relevance based threshold. In all cases the number of vortices detected varies smoothly with the relevance parameter (red line, top axis) and is much less sensitive than to the raw threshold (black line, bottom axis).

parameter ranges. Since for lower parameter values significantly more regions are considered vortex-like, this means that simultaneously strong vortices are combined into large superstructures as expected by the images of Figures 3 and 4. To further quantify the differences Figure 7 shows unnormalized cumulative distribution functions (CDFs) of the vortex sizes at different thresholds for vortices covering more than ten voxels to remove artifacts. For all but the λ_2 criterion the relevance thresholds identify significantly more individual vortices and far fewer large outliers. Overall, this confirms the impression of the visualizations that relevance is able to simultaneously detect a much large number of well formed vortices. It is worth noting that while we see separation of identification of the vortices from the process of filtering based on vortex intensity as a positive, information about the strength of the vortex is in some sense lost in this process. It falls to the analyst to use the input fields to label the vortices based on some opportune criteria to indicate their strength. For example, we have found it convenient to first identify vortices based on a relevance criteria and then select vortices based on the strength of the local maxima contained within the each structure. Figure 8 shows the unnormalized CDF of the per-vortex maximal vorticity for all indicators at the default Relevance
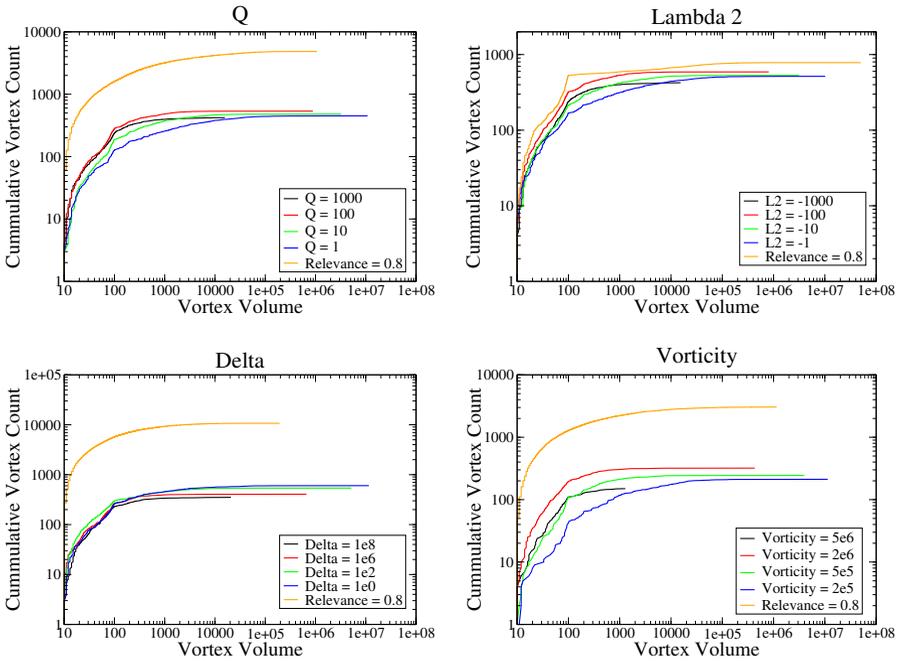
Figure 7. Vortex size distribution for $R = 0.8$ compared to distribution for various threshold values for four common indicator functions. In all cases, the relevance metric detects more vortices of each size.
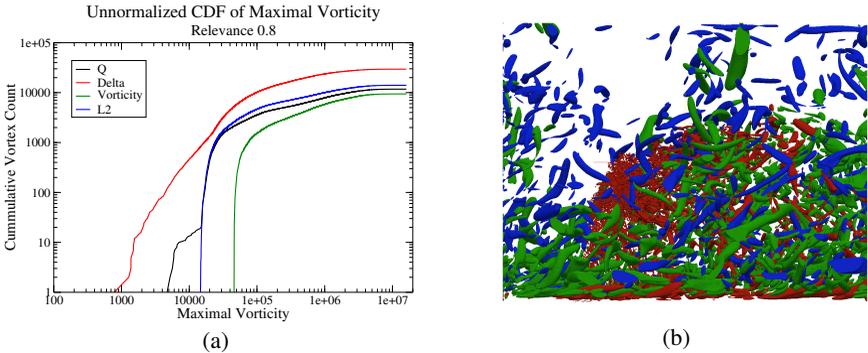
Figure 8. (a) Unnormalized CDF of maximal per-vortex vorticity for different indicators for $R = 0.8$. (b) Rendering of the vortices identified by the Q criterion color by max vorticity: (red) $[10^5, \text{max}]$; (green) $[3 \times 10^4, 10^5]$; and (blue) $[0, 3 \times 10^4]$.

of 0.8. All indicators show roughly the same behavior even though the Δ and Q criteria are able to extract a few noticeably weaker vortices. More important than this specific example is the fact that using local thresholds other selection criteria can be specified individually and without the problem of creating massive artificial

structures in high indicator regions. Clearly, other criteria such as vortex radius, length, etc, could also be used for subselection. A convenient aspect of encoding the results of the topological analysis in a relevance field is that the single field can then be fed into existing downstream analysis and visualization tools for such analysis.

5. Conclusion

While a single isosurface of many traditional measures of vorticity typically combines features (because the strength of the vortices varies rather than because the features are part of the same structure), the proposed approach keeps these separate. By bounding the features by a fraction of the local maximum, we decouple the identification of the vortex extents from the strength of the vortex. Furthermore, because the identification of the vortex extents is insensitive to the value chosen, this ultimately reduces the subjective nature of the analysis. Our approach can be used with any of the standard indicator functions and preserves all their desirable features, such as Galilean invariance. However, the proposed method does not identify "tiered" structures where a large scale rotational motion may contain smaller vortices where the vorticity has valleys inside of the largest structure. This is a direct consequence of basing the segmentation on scalar fields designed to extract regions of consistently dominant rotational motion. When Jeong and Hussain [16] proposed the λ_2 criterion, two of the primary criticisms of earlier methods were the treatment of the conical vertex and the inability to simultaneously represent vortices with a large variation in strength. Our method, when applied to the λ_2 fields inherits improvements made in both of these areas. As the λ_2 field in practice tends to be very noisy, it is still limited in the second respect. However, when segmented through a topological approach, it is easy to "clean up" for unambiguous identification of structures. In addition, the topological approach also addresses the difficulty of using the vorticity magnitude to simultaneously represent vortices with widely varying strengths.

Acknowledgements

References

[1] W. T. Ashurst, A. R. Kerstein, R. M. Kerr, and C. H. Gibson, *Alignment of vorticity and scalar gradient with strain rate in simulated Navier–Stokes turbulence*, Phys. Fluids **30** (1987), no. 8, 2343–.

[2] J. Bennett, V. Krishnamurthy, S. Liu, V. Pascucci, R. Grout, J. Chen, and P.-T. Bremer, *Feature-based statistical analysis of combustion simulation data*, IEEE Trans. Vis. Comp. Graph. **17** (2011), no. 12, 1822–1831.

[3] H. Carr, J. Snoeyink, and U. Axen, *Computing contour trees in all dimensions*, Comput. Geom. **24** (2003), no. 2, 75–94. MR 2004b:68162 Zbl 1052.68098

[4] P. Chakraborty, S. Balachandar, and R. J. Adrian, *On the relationships between local vortex identification schemes*, J. Fluid Mech. **535** (2005), 189–214. MR 2007f:76118

[5] J. H. Chen, A. Choudhary, B. de Supinski, M. DeVries, E. R. Hawkes, S. Klasky, W.K.Liao, K. L. Ma, J. Mellor-Crummey, N. Podhorszki, R. Sankaran, S. Shende, and C. S. Yoo, *Terascale direct numerical simulations of turbulent combustion using S3D*, Comput. Sci. Disc. **2** (2009), 015001.

[6] M. S. Chong, A. E. Perry, and B. J. Cantwell, *A general classification of three-dimensional flow fields*, Phys. Fluids A **2** (1990), no. 5, 765–777. MR 92b:76037

[7] J. del Álamo, J. Jimenez, P. Zandonade, and R. D. Moser, *Self-similar vortex clusters in the turbulent logarithmic region*, J. Fluid Mech. **561** (2006), 329–358. Zbl 1157.76346

[8] T. F. Fric and A. Roshko, *Vortical structure in the wake of a transverse jet*, J. Fluid Mech. **279** (1994), 1–47. Zbl 1303.11056

[9] C. H. Gibson, *Isoenstrophy points and surfaces in turbulent flow and mixing*, Fluid Dynamics Res. **3** (1988), no. 1-4, 331–336.

[10] R. W. Grout, P.-T. Bremer, A. Gruber, J. Bennett, A. Gyulassy, and J. H. Chen, *Heat release and turbulence statistics from a DNS of reacting jet in cross-flow parameterized in a jet natural coordinate system developed from scalar quantities*, paper at international conference on numerical combustion, corfu, 2011; sandia national laboratories report no. SAND2011-2994C/482316, Sandia National Laboratories, Livermore, CA, 2011.

[11] R. Grout, A. Gruber, H. Kolla, P.-T. Bremer, J. Bennett, A. Gyulassy, and J. Chen, *A direct numerical simulation study of turbulence and flame structure in transverse jets analysed in jet-trajectory based coordinates*, J. Fluid Mech. **706** (2012), 351–383. Zbl 1275.76147

[12] R. Grout, A. Gruber, C. Yoo, and J. Chen, *Direct numerical simulation of flame stabilization downstream of a transverse fuel jet in cross-flow*, Proc. Combustion Inst. **33** (2011), no. 1, 1629–1637.

[13] G. Haller, *An objective definition of a vortex*, J. Fluid Mech. **525** (2005), 1–26. MR 2008h:76007 Zbl 1071.76015

[14] J. C. R. Hunt, A. Wray, and P. Moin, *Eddies, stream, and convergence zones in turbulent flows*, Proceedings of the summer program 1988, Center for Turbulence Research (Stanford University), 1988, pp. 193–208.

[15] A. Hussain, *Coherent structures – reality and myth*, Phys. Fluids **26** (1983), 2816–2850.

[16] J. Jeong and F. Hussain, *On the identification of a vortex*, J. Fluid Mech. **285** (1995), 69–94. MR 96a:76017 Zbl 0847.76007

[17] A. Mascarenhas, R. W. Grout, C. S. Yoo, and J. H. Chen, *Tracking flame base movement and interaction with ignition kernels using topological methods*, J. Phys. Conf. Ser. **180** (2009), 012086.

[18] M. Tanahashi, T. Miyauchi, and J. Ikeda, *Scaling law of coherent fine scale structure in homogeneous isotropic turbulence*, Proceedings of the 11th Symposium on Turbulent Shear Flows, I, Institut National Polytechnique, Centre National de la Recherche Scientifique, Université Joseph Fourier, 1997, pp. 4–17.

[19] H. Tennekes and J. Lumley, *A first course in turbulence*, MIT Press, Cambridge, MA, 1972. Zbl 0285.76018

PEER-TIMO BREMER: bremer5@llnl.gov
Center for Applied Scientific Computing, Lawrence Livermore National Laboratories,
7000 East Avenue, L-422, Livermore, CA 94551, United States

ANDREA GRUBER: Andrea.Gruber@sintef.no
SINTEF Energy Research, K. Heiesvei 1A, 7465 Trondheim, Norway

JANINE C. BENNETT: jcbenne@sandia.gov
Sandia National Laboratories, 7011 East Ave, Livermore, CA 94551, United States

ATTILA GYULASSY: aggyulassy@ucdavis.edu
University of Utah, Salt Lake City, CA 84112, United States

HEMANTH KOLLA: hnkolla@sandia.gov
Sandia National Laboratories, 7011 East Ave, Livermore, CA 94551, United States

JACQUELINE H. CHEN: jhchen@sandia.gov
Reacting Flows, Sandia National Laboratories, 7011 East Ave, Livermore, CA 94551, United States

RAY W. GROUT: ray.grout@nrel.gov
Computational Science Center, National Renewable Energy Laboratory, 15013 Denver West Parkway,
Golden, CO 80401, United States

msp

AN ASYMPTOTIC-PRESERVING SCHEME FOR
SYSTEMS OF CONSERVATION LAWS WITH
SOURCE TERMS ON 2D UNSTRUCTURED MESHES

CHRISTOPHE BERTHON, GUY MOEBS,
CÉLINE SARAZIN-DESBOIS AND RODOLPHE TURPAULT

In this paper, finite volume numerical schemes are developed for hyperbolic systems of conservation laws with source terms. The systems under consideration degenerate into parabolic systems in large times when the source terms become stiff. In this framework, it is crucial that the numerical schemes are asymptotic-preserving, i.e., that they degenerate accordingly. Here, an asymptotic-preserving numerical scheme is proposed for any system within the aforementioned class on 2D unstructured meshes.

This scheme is proved to be consistent and stable under a suitable CFL condition. Moreover, we show that it is also possible to prove that it preserves the set of (physically) admissible states under a geometric property on the mesh. Finally, numerical examples are given to illustrate its behavior.

Introduction

The objective of this paper is to build a suitable numerical scheme for hyperbolic systems of conservation laws which can be written in the following form:

$$\partial_t \boldsymbol{U} + \mathrm{div}(\boldsymbol{F}(\boldsymbol{U})) = \gamma(\boldsymbol{U})(\boldsymbol{R}(\boldsymbol{U}) - \boldsymbol{U}), \quad (t, x) \in \mathbb{R}_+ \times \mathbb{R}^2. \quad (1)$$

Here, the Jacobian of the flux $\boldsymbol{F}$ is assumed to be diagonalizable in $\mathbb{R}$. The set of admissible states is denoted $\mathcal{A}$. Moreover, $\boldsymbol{R}$ is a smooth function of $\boldsymbol{U}$ such that for all $\boldsymbol{U} \in \mathcal{A}$, $\boldsymbol{R}(\boldsymbol{U}) \in \mathcal{A}$. Finally, $\gamma(\boldsymbol{U})$ is a positive real function which represents the stiffness of the source term.

The system (1) is assumed to fulfill the properties required in [6] so that it degenerates in long time and when the source term becomes stiff, more precisely when $\gamma(\boldsymbol{U})t \to \infty$, into a parabolic system.

55

There are numerous examples of such systems and two of them will be used throughout this article as illustrations, namely the isentropic Euler equations with friction and the M_1 model for radiative transfer:

Isentropic Euler with friction:

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \end{pmatrix}, \qquad F(U) = \begin{pmatrix} \rho u & \rho v \\ \rho u^2 + p(\rho) & \rho u v \\ \rho u v & \rho v^2 + p(\rho) \end{pmatrix},$$

$$\gamma(U) = \kappa(\rho), \qquad\qquad R(U) = \begin{pmatrix} \rho \\ 0 \\ 0 \end{pmatrix}, \qquad\qquad (2)$$

where p is a smooth function such that $p'(\rho) > 0$. The set of admissible states is

$$\mathcal{A} = \{U = (\rho, \rho u, \rho v)^\top \in \mathbb{R}^3 \mid \rho > 0\}. \qquad (3)$$

The diffusion limit whenever $\kappa(\rho)t \to \infty$ is (see [8] for instance):

$$\partial_t \rho - \mathrm{div}\left(\frac{1}{\kappa(\rho)}\nabla p(\rho)\right) = 0. \qquad (4)$$

M_1 *model for radiative transfer* (see [22] for the derivation of the model and [7] for the present form):

$$U = \begin{pmatrix} E \\ Fx \\ Fy \\ T \end{pmatrix}, \qquad F(U) = \begin{pmatrix} Fx & Fy \\ c^2 Pxx & c^2 Pxy \\ c^2 Pyx & c^2 Pyy \\ 0 & 0 \end{pmatrix},$$

$$\gamma(U) = c\sigma_m(U), \qquad R(U) = \begin{pmatrix} \dfrac{\sigma(U)aT^4 + \sigma_1(U)}{\sigma_m(U)} \\[2mm] \dfrac{\sigma_1(U)Fx}{\sigma_m(U)} \\[2mm] \dfrac{\sigma_1(U)Fy}{\sigma_m(U)} \\[2mm] \dfrac{\sigma(U)E + \sigma_2(U)\rho C_v T}{\rho C_v \sigma_m(U)} \end{pmatrix}, \qquad (5)$$

where

$$P = E\left(\frac{1 - \chi}{2}I_d + \frac{3\chi - 1}{2}\frac{F \otimes F}{\|F\|^2}\right),$$

$$\chi = \chi\left(\xi = \frac{\|F\|}{cE}\right) = \frac{3 + 4\xi^2}{5 + 2\sqrt{4 - 3\xi^2}}, \qquad F = (Fx, Fy)^\top,$$

and

$$\sigma_m = \sigma_m(\boldsymbol{U}) = \sigma(\boldsymbol{U}) \max\left(1, \frac{aT^3}{\rho C_v}\right),$$

$$\sigma_1(\boldsymbol{U}) = \sigma_m(\boldsymbol{U}) - \sigma(\boldsymbol{U}),$$

$$\sigma_2(\boldsymbol{U}) = \sigma_m(\boldsymbol{U}) - \sigma(\boldsymbol{U}) \frac{aT^3}{\rho C_v},$$

The set of admissible states is

$$\mathcal{A} = \{\boldsymbol{U} = (E, F_x, F_y, T)^\top \in \mathbb{R}^4 \mid E > 0,\ T > 0,\ \|F\| \le cE\}. \tag{6}$$

When $\sigma_m(\boldsymbol{U})t \to \infty$, the M_1 model degenerates into the so-called *equilibrium diffusion equation*:

$$\partial_t(\rho C_v T + aT^4) - \operatorname{div}\left(\frac{c}{3\sigma} \nabla aT^4\right) = 0. \tag{7}$$

The main difficulty when designing a numerical scheme for such systems is to enforce the correct degeneracy in the diffusion limit. In other words, the limit of the scheme when $\gamma(\boldsymbol{U})t \to \infty$ shall be a consistent approximation of the limit diffusion equation (see [28]). Obviously, this property is generally not fulfilled by numerical schemes hence the design of *asymptotic-preserving* (AP) schemes has been an important issue during the last decade.

For 1D applications, several asymptotic-preserving schemes were proposed in this context. The most explored way to do so is to use a modified HLL scheme [26] and cleverly control the numerical diffusion in the spirit of the work of Gosse and Toscani for the telegraph equations [25]. This technique has been widely used for the M_1 model for radiative transfer and Euler equations with friction (see for instance [11; 12; 5; 14]) and extended to general cases [7]. Other techniques have also been used, such as [9; 15] in the framework of Euler equations with friction, or [3] where the knowledge of the convergence rates towards equilibrium is extensively used.

The situation is much more difficult for 2D applications however. While it is quite straightforward in the case of Cartesian grids (see [4] for example), the situation is far more complex on unstructured grids. One of the reasons is that the classical two-point flux scheme (or FV4 [24]) which is the target of many AP schemes is not consistent anymore. The only exception is the MPFA-based AP scheme for Friedrich systems developed in [13].

Our goal is therefore to propose an AP finite volume scheme for any system of the form (1). This scheme is a natural extension of the 1D scheme proposed in [7]. It will be proved to be consistent and stable under a natural unrestrictive CFL

condition. Moreover, it is also possible to enforce the preservation of the set of admissible states provided a geometric property is satisfied by the mesh.

The rest of the paper is organized as follows. First the notation used throughout the paper is presented. Then, the scheme is introduced in the case where the mesh is admissible. This additional property indeed allows us to simplify several expressions. In this case, the scheme is proved to be consistent and to preserve the set of admissible states under a natural CFL condition. We explain how the AP property can be enforced.

Then, the scheme is extended for general meshes. Once again, it is proved to be consistent and AP. Since the target scheme in the diffusive regime is the diamond scheme [17] — which does not preserve the maximum principle — it does not preserve the set of admissible states on general meshes. However, we show that this feature can be recovered under some geometric constraint on the mesh.

Finally, the scheme is benchmarked on numerical examples and a few words on the optimization and parallelization of the code conclude this work.

Notation

Since we intend to provide a finite volume scheme which may be used in either *cell-centered* or *vertex-centered* (or *cell-vertex*) contexts, we will use the following nomenclature:

- The *primary* mesh $\mathcal{M}$ is the set of all control volumes (or cells) effectively used in the scheme. As a consequence, the *primary* mesh is the primal mesh in the context of cell-centered schemes and the dual mesh in the context of vertex-centered schemes.

- The *secondary* mesh is a set of control volumes defined around the nodes of the *primary* mesh. Practically, the *secondary* mesh is the dual mesh in the context of cell-centered schemes and the primal mesh in the context of vertex-centered schemes.

For the sake of clarity in the following, the *primary* mesh is simply called the mesh whenever no confusion is possible.

The notation used throughout this paper is summarized in Figure 1:

- N_K is the number of nodes (and interfaces) of the cell $K \in \mathcal{M}$.

- x_K is the centroid of the cell K.

- The nodes of the cell K are locally denoted $\{A_i\}_{i=1...N_K}$ with the convention $A_0 = A_{N_K}$ and $A_{N_K+1} = A_1$.

- The neighboring cells of the cell K (i.e., cells that share an edge with K) are locally numbered from 1 to N_K such that $K \cap L_i = [A_i A_{i+1}]$. Their centroids are locally denoted $\{x_i\}_{i=1...N_K}$.
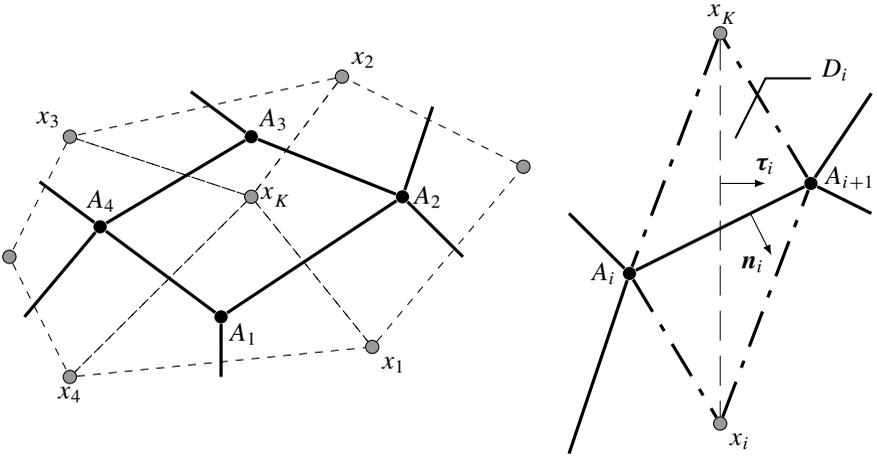
Figure 1. Local mesh notation. Left: one cell and the local notation. Right: the diamond cell associated with the i-th interface of the cell $K \in \mathcal{M}$.

- $d_i^K := \|x_K x_i\|$.
- $e_i := \|A_i A_{i+1}\|$ is the length of the i-th interface of the cell K.
- The unit outward normal to the i-th interface of the cell K is denoted $\boldsymbol{n}_i$.
- The unit normal to $x_K x_i$ is denoted $\boldsymbol{\tau}_i$. As a convention, it is chosen such that $(\boldsymbol{n}_i \wedge \boldsymbol{\tau}_i).Oz > 0$.
- The characteristic length r^K is defined by $r^K := |K|/p_k$ where $p_k = \sum_i e_i$ is the perimeter of K. Let us remark that for example, $r^K = h/4$ in a square cell of size h.

For the sake of clarity, we recall that

$$\boldsymbol{F} \cdot \begin{pmatrix} n_x \\ n_y \end{pmatrix} = \begin{pmatrix} \boldsymbol{F}_x & \boldsymbol{F}_y \end{pmatrix} \cdot \begin{pmatrix} n_x \\ n_y \end{pmatrix} = n_x \boldsymbol{F}_x + n_y \boldsymbol{F}_y.$$

1. An AP scheme on admissible meshes

In this section, the scheme is introduced on admissible meshes (see definition below). This property indeed allows us to simplify the notation and to prove the preservation of admissible states. The extension to general meshes is described in the next section. Let us mention that this scheme is explicit but an implicit version can be obtained straightforwardly using the same technique.

Definition. The mesh is said to be admissible if

$$x_K x_i \cdot A_i A_{i+1} = 0 \quad \text{for all } K \text{ and all } i \in [1, N_K],$$

i.e., if all interfaces are orthogonal to the lines which join the centroids of the cells.

Obviously, the property required for a mesh to be admissible restricts the generality of this class of meshes. However, it includes Cartesian grids and it is possible to consider unstructured admissible meshes for reasonable geometries. For instance, a mesh consisting of triangles in which the centroids are the circumcenters is often a valid possibility.

On admissible meshes, we propose to consider the following scheme:

$$U_K^{n+1} = U_K^n - \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i \alpha_i^K \mathcal{F}_{K,i}^n \cdot \boldsymbol{n}_i + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i \alpha_i^K \boldsymbol{F}(U_K^n) \cdot \boldsymbol{n}_i$$

$$+ \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i b_i^K (1 - \alpha_i^K)(\boldsymbol{R}(U_K^n) - U_K^n), \quad (8)$$

where α_i^K is defined by

$$\alpha_i^K = \frac{b_i^K}{b_i^K + \gamma_i^K r^K}, \quad (9)$$

and the numerical flux is given by

$$\mathcal{F}_{K,i}^n = \tfrac{1}{2}(\boldsymbol{F}(U_K^n) + \boldsymbol{F}(U_i^n)) - \tfrac{1}{2} b_i^K (U_i^n - U_K^n), \quad (10)$$

where $b_i^K > 0$ is a parameter larger that all characteristic speeds to be defined depending on the form of (1). This scheme is designed to keep the numerical diffusion in the normal direction to the interfaces in order to stay consistent in the diffusion limit. It is also designed to be a convex combination of 1D schemes that enter the formalism of [7]. As we will see in the proofs, this feature allows it to automatically inherit the preservation of admissible states under a suitable CFL condition.

Remark. For the M_1-model, since the physical flux is equal to zero in the last equation (therefore, the temperature T is only coupled through the source term), the corresponding component of the numerical flux is also set to zero.

Theorem 1.1. *Assume that $\gamma_i^K = \gamma > 0$ is a constant. Then the scheme* (8)–(10) *is consistent with* (1).

Proof. Let us consider a sequence of regular meshes such that

$$\lim_{\eta_K \to 0} r^K = 0,$$

where η_K is the radius of the largest circle inside the cell K. This immediately implies that

$$\lim_{\eta_K \to 0} \alpha_i^K = 1.$$

Moreover, the divergence formula gives

$$\sum_{i=1,N_K} e_i \boldsymbol{F}(\boldsymbol{U}_K^n) \cdot \boldsymbol{n}_i = 0.$$

Therefore, the first two terms of (8) are consistent with the hyperbolic part of (1).

Now, the third term is consistent with the source term. Indeed,

$$(1 - \alpha_k^K) = \frac{\gamma r^K}{b_i^K + \gamma r^K},$$

so

$$\frac{1}{|K|} \sum_{i=1}^{N_K} e_i b_i^K (1 - \alpha_i^K)(\boldsymbol{R}(\boldsymbol{U}_K^n) - \boldsymbol{U}_K^n) = \frac{1}{|K|} \sum_{i=1}^{N_K} e_i b_i^K \frac{\gamma r^K (\boldsymbol{R}(\boldsymbol{U}_K^n) - \boldsymbol{U}_K^n)}{b_i^K + \gamma r^K},$$

which tends to $\gamma(\boldsymbol{R}(\boldsymbol{U}_K^n) - \boldsymbol{U}_K^n)$ when $\eta_K \to 0$.

Finally, the scheme (8)–(10) is consistent with (1). □

Remark. This consistency result may be extended to include the case of nonconstant γ. Indeed, γ only appears in the scheme inside α_i^K. When γ is not constant, one can perform a Taylor expansion of α_i^K and the above proof is still valid up to high order terms which tend to zero with η_K.

Theorem 1.2. *The scheme preserves the set of admissible states as soon as the following CFL condition holds*:

$$\max_{K,i} b_i^K \frac{\Delta t}{|K|} p_K \le 1. \tag{11}$$

Proof. To prove this result, we establish that the scheme (8)–(10) can be written as a convex combination of 1D schemes. These 1D schemes are nothing but the ones proposed in [7] with a Rusanov flux for the hyperbolic part.

Let $\boldsymbol{U}_{K,i}^{n+1}$ be an intermediate state given by such a 1D scheme in the normal direction to the interface $A_i A_{i+1}$ and considering a space length r^K. It is therefore given by

$$\boldsymbol{U}_{K,i}^{n+1} = \boldsymbol{U}_{K,i}^n - \frac{\Delta t}{r^K}(\alpha_i^K \mathcal{F}_{K,i}^n \cdot \boldsymbol{n}_i - \alpha_K^K \mathcal{F}_{K,K}^n \cdot \boldsymbol{n}_i) - \frac{\Delta t}{r^K}(\alpha_i^K - \alpha_K^K)\boldsymbol{F}(\boldsymbol{U}_K^n) \cdot \boldsymbol{n}_i$$

$$+ \frac{b_i^K \Delta t}{r^K}\big((1 - \alpha_i^K) + (1 - \alpha_K^K)\big)(\boldsymbol{R}(\boldsymbol{U}_K^n) - \boldsymbol{U}_K^n), \quad (12)$$

where α_i^K is given by (9) and $\alpha_K^K = 1$.

According to [7], such a scheme preserves the convex set of admissible states $\mathcal{A}$ under the CFL condition

$$\max_{K,i}\left(b_i^K \frac{\Delta t}{r^K}\right) \le \frac{1}{2}. \tag{13}$$

Now, we set

$$\omega_{K,i} := \frac{e_i}{p_K},$$

so that $\omega_{K,i}$ is positive and

$$\sum_{i=1}^{N_K} \omega_{K,i} = 1.$$

Equipped with this choice, we consider the following convex combination of the 1D schemes (12):

$$\sum_{i=1}^{N_K} \omega_{K,i} U_{K,i}^{n+1} = \sum_{i=1}^{N_K} \omega_{K,i} U_{K,i}^n - \sum_{i=1}^{N_K} \omega_{K,i} \frac{\Delta t}{r^K} (\alpha_i^K \mathcal{F}_{K,i}^n \cdot \boldsymbol{n}_i - \alpha_K^K \mathcal{F}_{K,K}^n \cdot \boldsymbol{n}_i)$$

$$- \sum_{i=1}^{N_K} \omega_{K,i} \frac{\Delta t}{r^K} (\alpha_i^K - \alpha_K^K) \boldsymbol{F}(U_K^n) \cdot \boldsymbol{n}_i$$

$$+ \sum_{i=1}^{N_K} \omega_{K,i} \frac{b_i^K \Delta t}{r^K} ((1 - \alpha_i^K) + (1 - \alpha_K^K))(\boldsymbol{R}(U_K^n) - U_K^n).$$

Since

$$\frac{\omega_{K,i}}{r^K} = \frac{e_i}{|K|},$$

this convex combination is nothing but the scheme (8), i.e.,

$$U_{K,i}^{n+1} = \sum_{i=1}^{N_K} \omega_{K,i} U_{K,i}^{n+1},$$

and the choice of r^K implies that the CFL condition (13) becomes the condition (11). Therefore, if all $U_K^n \in \mathcal{A}$, this condition ensures that all $U_{K,i}^{n+1} \in \mathcal{A}$ and therefore $U_K^{n+1} \in \mathcal{A}$ since $\mathcal{A}$ is convex. $\qquad\square$

At this point, the scheme still doesn't preserve the asymptotic. Hopefully, this property may be easily recovered in the same way it was enforced in [7]. Indeed, as mentioned in [7], for any $\bar{\gamma}$ such that $\gamma + \bar{\gamma} > 0$ the scheme (8) may be applied to the system

$$\partial_t U + \mathrm{div}(\boldsymbol{F}(U)) = (\gamma + \bar{\gamma})(\bar{\boldsymbol{R}}(U) - U), \tag{14}$$

where

$$\bar{\boldsymbol{R}}(U) = \frac{\gamma \boldsymbol{R}(U) + \bar{\gamma} U}{\gamma + \bar{\gamma}}. \tag{15}$$

In fact, this system is equivalent to (1) but using the scheme (8) on it allows one to consider $\bar{\gamma}$ as a free parameter which can be used to recover the asymptotic limit.

The asymptotic limit of the scheme (8) is formally obtained by performing a Chapmann–Enskog expansion. A small parameter ε is introduced and the following rescalings are made:

$$\Delta t \leftarrow \frac{\Delta t}{\varepsilon}, \quad \gamma \leftarrow \frac{\gamma}{\varepsilon}.$$

Therefore we have

$$\alpha_i^K \leftarrow \frac{\varepsilon b_i^K}{\varepsilon b_i^K + (\gamma_i^K + \overline{\gamma_i^K})r^K}, \quad \text{and} \quad 1 - \alpha_i^K \leftarrow \frac{(\gamma_i^K + \overline{\gamma_i^K})r^K}{\varepsilon b_i^K + (\gamma_i^K + \overline{\gamma_i^K})r^K}.$$

Using these rescalings in (8), an identification gives:
terms in ε^{-1}:

$$\boldsymbol{R}(\boldsymbol{U}_K^n) = \boldsymbol{U}_K^n, \tag{16}$$

terms in ε^0:

$$\boldsymbol{U}_K^{n+1} = \boldsymbol{U}_K^n - \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i \frac{b_i^K}{(\gamma_i^K + \overline{\gamma_i^K})r^K} [\mathcal{F}(\boldsymbol{U}_K^n) \cdot \boldsymbol{n}_i - \boldsymbol{F}(\boldsymbol{U}_K^n) \cdot \boldsymbol{n}_i]\big|_{\boldsymbol{R}(\boldsymbol{U}_K^n) = \boldsymbol{U}_K^n}. \tag{17}$$

The parameters $\overline{\gamma_i^K}$ are then fixed by imposing a given scheme for the diffusion limit. As examples, we propose two corrections that allow us to recover the classical two-point flux scheme in the asymptotic regimes of the M_1 model for radiative transfer and Euler system with friction.

Remark. Let us underline that the Chapmann–Enskog expansion is nothing but a tool to obtain the correct asymptotic limit in the diffusion regime. It has to be handled with care for other purposes since it assumes that both γ and Δt are large (with the same order in ε), while the degeneracy is indeed governed by γt (see for instance [8]).

However, techniques that avoid rescalings which can be used in the continuous case (for instance [8]) cannot be extended to the discrete level.

AP correction for the M_1 model. In the case of the M_1 model for radiative transfer (5) $b_i^K = c$ and the equilibrium (16) gives $F_x = F_y = 0$ and $E = aT^4$. The sum of the first and fourth equations of (17) hence becomes

$$(\rho C_v + aT^4)_K^{n+1} = (\rho C_v T + aT^4)_K^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} \frac{ce_i}{2(\sigma_{m,i}^K + \bar{\sigma}_i^K)r^K} \big((aT^4)_i^n - (aT^4)_K^n\big).$$

This scheme is not consistent in general with the equilibrium diffusion equation (7), but it is possible to choose $\bar{\sigma}$ in order to recover the consistency in the diffusive

limit. For example, if we take

$$(\sigma_{m,i}^K + \bar{\sigma}_i^K) = \sigma_{m,i}^K \frac{3|D_i|}{2r^K e_i} > 0, \tag{18}$$

then the limit scheme in the diffusion regime is

$$(\rho C_v T + aT^4)_K^{n+1}$$
$$= (\rho C_v T + aT^4)_K^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} \frac{ce_i^2}{3\sigma_i^K |Di|} \big((aT^4)_i^n - (aT^4)_K^n\big), \tag{19}$$

which is nothing but the classic FV4 scheme (see [24]) for the diffusion equation (7) on admissible meshes.

AP correction for Euler equations with friction. Now we consider the case of Euler equations with friction (2). In this case, the equilibrium (16) gives $\rho u = \rho v = 0$, and the first equation of (17) hence becomes

$$\rho_K^{n+1} = \rho_K^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i \frac{(b_i^K)^2}{2(\kappa_i^K + \bar{\kappa}_i^K)r^K} (\rho_i^n - \rho_K^n).$$

Once again, this scheme is not consistent in general with the limit diffusive regime (4), however it is also possible to choose $\bar{\kappa}$ in order to recover the consistency in this limit. For instance if we take

$$(\kappa_i^K + \bar{\kappa}_i^K) = \begin{cases} 2\kappa_i^K \dfrac{(b_i^K)^2 |D_i|}{r^K e_i} \dfrac{\rho_i^n - \rho_K^n}{p(\rho_i^n) - p(\rho_K^n)} & \text{if } \rho_i^n \neq \rho_K^n, \\ 2\kappa_i^K \dfrac{(b_i^K)^2 |D_i|}{r^K e_i \, p'(\rho_K^n)} & \text{otherwise,} \end{cases} \tag{20}$$

then the limit scheme in the diffusion regime is

$$\rho_K^{n+1} = \rho_K^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} \frac{e_i^2}{\kappa_i^K |D_i|} \big(p(\rho_i^n) - p(\rho_K^n)\big), \tag{21}$$

which is consistent with the diffusion equation (4) on admissible meshes since it is once again nothing but the FV4 scheme in this context.

Remark. It is required for the hyperbolicity of the system that p be an increasing function of ρ. Therefore, the choice (20) provides positive values for $(\kappa_i^K + \bar{\kappa}_i^K)$, as required in robustness theorems.

2. Extension to more general meshes

When the mesh is not admissible, there is an additional difficulty since the classical two-point finite volume scheme (also known as FV4 [24]) is not consistent with the

diffusion equation anymore. The target scheme in the diffusive limit must therefore properly take into account the whole gradient. For the sake of consistency and simplicity, we choose to use the same gradient discretization in the hyperbolic part. Among the possible choices available in the literature, we adopt the *diamond scheme* strategy [17], but other strategies could be considered such as DDFV schemes [27; 18] (see also [1; 2; 10; 16]) or hybrid strategies (see [19; 20; 23] and references therein).

With the diamond scheme to approximate the gradients, it is possible to propose a natural extension of the scheme for admissible meshes (8)–(10) into the following generalized scheme:

$$U_K^{n+1} = U_i^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i \alpha_{K,i}^n \mathcal{F}_{K,i}^n \cdot \boldsymbol{n}_i + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i \alpha_{K,i}^n \boldsymbol{F}(U_K^n) \cdot \boldsymbol{n}_i$$

$$+ \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i (1 - \alpha_{K,i}^n) b_i^K (\boldsymbol{R}(U_K^n) - U_K^n), \quad (22)$$

where

$$\mathcal{F}_{K,i}^n = \tfrac{1}{2}(\boldsymbol{F}(U_K^n) + \boldsymbol{F}(U_i^n)) - \tfrac{1}{2} b_i^K \theta_i^K \nabla_i^K U_K^n \cdot \boldsymbol{n}_i \quad (23)$$

with θ_i^K a positive parameter to be specified later and

$$\nabla_i^K U_K^n \cdot \boldsymbol{n}_i = \frac{U_i^n - U_K^n}{2|D_i|} e_i + \frac{U_{A_{i+1}}^n - U_{A_i}^n}{2|D_i|} d_i^K \boldsymbol{n}_i \cdot \boldsymbol{\tau}_i, \quad (24)$$

$U_{A_i}^n$ being the value of the solution at the node A_i (see Figure 1). This value is obtained as a mean value of the solution in the cells which share A_i as a node (see [17]).

With this definition, we immediately see that the scheme for admissible meshes (8)–(10) is recovered if $\theta_i^K = 2|D_i|/e_i$.

Theorem 2.1. *Assume that $\gamma_i^K = \gamma > 0$ is a constant and $\theta_i^K \to 0$ when $\eta_K \to 0$. Then the scheme (22)–(23) is consistent with (1).*

Proof. As was pointed in the introduction of the scheme (22)–(23), the only difference compared to the scheme designed for admissible meshes (8)–(10) lies in the definition of the discrete gradient in the numerical flux. But if $\theta_i^K \to 0$, this difference between the two schemes converges to zero when $\eta_K \to 0$. Therefore, the arguments in the proof of Theorem 1.1 can still be applied to obtain the consistency. $\square$

The preservation of the set of admissible states $\mathcal{A}$ is all the more difficult since most finite volume schemes for parabolic problems do not preserve the maximum principle. Only a few examples ensure this property [29; 21]. It is therefore expected that the extension of Theorem 1.2 either does not hold for nonadmissible meshes

or is very difficult to prove. Interestingly, it is sometimes possible to recover the maximum principle under some geometric condition on the mesh.

Definition. The mesh is said to be δ-admissible if there exists a constant $\delta > 0$ such that the following property holds:

$$\forall K \in \mathcal{M}, \ \forall i \in [1, N_K], \quad 1 + \frac{e_{i-1}\overline{d_{i-1}^K}}{e_i^2}\frac{|D_i|}{3|D_i-1|} - \frac{e_{i+1}\overline{d_{i+1}^K}}{e_i^2}\frac{|D_i|}{3|D_i+1|} > \delta,$$

where

$$\overline{d_i^K} = d_i^K \boldsymbol{n}_i \cdot \boldsymbol{\tau}_i.$$

Remark. With this definition, an admissible mesh is δ-admissible for all $\delta \leq 1$ since all d_i^K are then equal to 0. While not all meshes are δ-admissible, this condition turned out to be satisfied by most of the meshes we tested that were generated with reasonable constraints on the angles.

Equipped with this definition, we can obtain a generalization of Theorem 1.2 for the scheme (22)–(23) applied on a δ-admissible mesh where the secondary mesh is made of triangles (e.g., vertex-centered schemes of a triangular mesh).

Theorem 2.2. *Assume that the mesh is δ-admissible and that the secondary mesh is made of triangles. Let us also assume that α_i^K is constant inside each cell $K \in \mathcal{M}$ ($\alpha_i^K = \alpha^K$) and set $\theta_i^K = 2|D_i|/\delta e_i$.*

Then, the scheme (22)–(23) preserves the set of admissible states $\mathcal{A}$ as long as the following CFL condition holds:

$$\max_{K \in \mathcal{A}, i \leq N_K} \{b_i^K \theta_i^K \delta_i^K\}\frac{\Delta t}{|K|}p_K \leq \frac{1}{2}. \tag{25}$$

Proof. We consider here secondary meshes made of triangles. This characteristic allows us to give a simple expression from the extrapolated solution at the vertices of the primary mesh (see Figure 1):

$$\boldsymbol{U}_{A_i} = \tfrac{1}{3}(\boldsymbol{U}_K + \boldsymbol{U}_i + \boldsymbol{U}_{i-1}).$$

Then, from the expression of the numerical flux (23), we obtain

$$\sum_{i=1}^{N_K} e_i \mathcal{F}_{K,i}^n \cdot \boldsymbol{n}_i = \frac{1}{2}\sum_{i=1}^{N_K} e_i (\boldsymbol{F}(\boldsymbol{U}_K^n) + \boldsymbol{F}(\boldsymbol{U}_i^n)) \cdot \boldsymbol{n}_i$$

$$-\frac{1}{2}\sum_{i=1}^{N_K} e_i b_i^K \theta_i^K \left(\frac{\boldsymbol{U}_i^n - \boldsymbol{U}_K^n}{2|D_i|}e_i + \frac{\boldsymbol{U}_{i+1}^n - \boldsymbol{U}_K^n + \boldsymbol{U}_K^n - \boldsymbol{U}_{i-1}^n}{6|D_i|}\overline{d_i^K} \right).$$

Then, summing by parts to reorganize the terms $U_{i\pm1}^n - U_K^n$ into $U_i^n - U_K^n$ we get

$$\sum_{i=1}^{N_K} e_i \mathcal{F}_{K,i}^n \cdot \boldsymbol{n}_i = \frac{1}{2} \sum_{i=1}^{N_K} e_i (\boldsymbol{F}(U_K^n) + \boldsymbol{F}(U_i^n)) \cdot \boldsymbol{n}_i - \frac{1}{2} \sum_{i=1}^{N_K} e_i b_i^K \theta_i^K \delta_i^K (U_i^n - U_K^n),$$

where

$$\delta_i^k = \frac{e_i}{2|D_i|} + \frac{e_{i-1}}{e_i} \frac{\overline{d_{i-1}^K}}{6|D_i - 1|} - \frac{e_{i+1}}{e_i} \frac{\overline{d_{i+1}^K}}{6|D_i + 1|}.$$

With the choice $\theta_i^K = 2|D_i|/\delta e_i$ and since the mesh is δ-admissible, we have $\theta_i^K \delta_i^K \geq 1$ for all $K \in \mathcal{M}$ and all $i \in [1, N_K]$. Moreover, the numerical flux can be expressed as

$$\mathcal{F}_{K,i}^n \cdot \boldsymbol{n}_i = \tfrac{1}{2}(\boldsymbol{F}(U_K^n) + \boldsymbol{F}(U_i^n)) \cdot \boldsymbol{n}_i - \tfrac{1}{2} b_i^K \theta^K \delta_i^K (U_i^n - U_K^n).$$

Hence the scheme (22)–(23) can be recast as a convex combination of 1D schemes as in the proof of Theorem 1.2. These 1D schemes are Rusanov schemes with a speed of $b_i^K \theta_i^K d_i^K \geq b_i^K$ from which the CFL condition follows:

$$\max_{K \in \mathcal{A}, i \leq N_K} \{b_i^K \theta_i^K \delta_i^K\} \frac{\Delta t}{|K|} p_K \leq \frac{1}{2}. \qquad \square$$

Remark. We have several comments concerning this theorem:

- The choice of θ_i^K tends to 0 when $\eta_K \to 0$ as it was requested for the sake of consistency.

- A similar theorem may be obtained on more general meshes. However, the geometric condition (equivalent to the definition of δ-admissible meshes above) quickly becomes cumbersome.

- The main restriction is to consider α_i^K that are constant per cell. As one can guess from the α_i^K chosen to obtain AP schemes in the previous section, it is not always possible to select a correction $\overline{\gamma_i^K}$ such that $\gamma_i^K + \overline{\gamma_i^K} > 0$ and does not depend on i.

- Other choices of θ_i^K allow one to recover the same result. For instance, one can consider $\theta_i^K = \max_{i \leq N_K} 2|D_i|/2\delta$.

The scheme (22)–(23) is also not asymptotic preserving in general but the procedure previously used can still be considered in order to recover this property. Indeed, a formal Chapmann–Enskog expansion will lead to the same two relations (16) and (17). Of course, in the last relation, the numerical flux is given by (23). Now, the correction is once again illustrated in the example of the M_1 model for radiative transfer and Euler equations with friction. The objective is to recover an extension of the schemes obtained in the diffusive limit in the case of admissible meshes.

AP correction for the M_1 model. We first consider the M_1 model for radiative transfer (5). We recall that in this case $b_i^K = c$ and the equilibrium (16) gives $F_x = F_y = 0$ and $E = aT^4$. The sum of the first and fourth equations of (17)–(23) hence becomes

$$(\rho C_v + aT^4)_K^{n+1} = (\rho C_v T + aT^4)_K^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} \frac{c^2 e_i}{2(\sigma_{m,i}^K + \bar{\sigma}_i^K)r^K} \nabla_i^K (aT^4)^n.$$

Once again, this scheme is not consistent in general with the equilibrium diffusion equation (7) but it is possible to choose $\bar{\sigma}$ in order to recover the consistency in the diffusive limit. For example, if we take

$$(\sigma_{m,i}^K + \bar{\sigma}_i^K) = \sigma_{m,i}^K \frac{3c\theta_i^K}{2r^K} > 0, \tag{26}$$

then the limit scheme in the diffusion regime is

$$(\rho C_v T + aT^4)_K^{n+1} = (\rho C_v T + aT^4)_K^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} \frac{ce_i}{3\sigma_i^K} \nabla_i^K (aT^4)_i^n,$$

which is consistent with the diffusion equation (7) and a clear extension of (19). Moreover, if σ is a constant and $\theta_i^K = \theta^K$ then $\bar{\sigma}_i^K = \bar{\sigma}^K$ and Theorem 2.2 can be applied. In order to meet such a requirement, one may choose the form of θ_i^K already mentioned earlier: $\theta_i^K = \max_{i \leq N_K} \frac{2|D_i|}{2\delta}$.

AP correction for Euler equations with friction. If we consider Euler equations with friction (2), the equilibrium (16) gives $\rho u = \rho v = 0$ and the first equation of (17)–(23) hence becomes

$$\rho_K^{n+1} = \rho_K^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} e_i \frac{(b_i^K)^2 \theta_i^K}{2(\kappa_i^K + \bar{\kappa}_i^K)r^K} \nabla_i^K \rho^n \cdot \boldsymbol{n}_i.$$

As previously, this scheme is not consistent in general with the limit diffusive regime (4), however it is possible to choose $\bar{\kappa}$ in order to recover the consistency in this limit. For instance if we take

$$(\kappa_i^K + \bar{\kappa}_i^K) = 2\kappa_i^K \frac{(b_i^K)^2 \theta_i^K}{r^K} \frac{\nabla_i^K \rho^n \cdot \boldsymbol{n}_i}{\nabla_i^K p(\rho^n) \cdot \boldsymbol{n}_i}, \tag{27}$$

then the limit scheme in the diffusion regime is

$$\rho_K^{n+1} = \rho_K^n + \frac{\Delta t}{|K|} \sum_{i=1}^{N_K} \frac{e_i}{\kappa_i^K} \nabla_i^K p(\rho^n) \cdot \boldsymbol{n}_i,$$

which is consistent with the diffusion equation (4) and a direct extension of (21).

Remark. The choice (27) also provides positive values for $(\kappa_i^K + \bar{\kappa}_i^K)$ since p is required to be an increasing function of ρ for the sake of hyperbolicity.

However, this choice generates an α_i^K which is not constant per cell and therefore, Theorem 2.2 cannot be applied here.

3. Numerical results

Validation tests. Validation tests are performed in this subsection in order to illustrate the behavior of the scheme. All tests share this same setup: a Riemann problem for the M_1 model for radiative transfer is considered on the rectangular domain $[0, 5] \times [0, 1]$, with

$$(E, F_x, F_y, T)^\top (0, x) = \begin{cases} (aT_L^4, cf_{x,L}aT_L^4, 0, T_L)^\top & \text{if } x < 1, \\ (aT_R^4, 0, 0, T_R)^\top & \text{otherwise.} \end{cases}$$

In the following, $T_L = 10000$, $T_R = 300$ and $f_{x,L} = 0$ unless otherwise specified and $\rho C_v = 10^{-2}$. We also recall that $c = 3 \times 10^8$.

All tests are performed at least with $\sigma = 0$ and $\sigma = 1$. Indeed, when $\sigma = 0$, the model turns out to be a hyperbolic system and the preservation of admissible states is expected to be more difficult than in the presence of the (regularizing) source term.

The end times of the simulations are $t = 2 \times 10^{-9}$ when $\sigma = 0$ and $t = 1 \times 10^{-8}$ when $\sigma = 1$.

The approximate solutions are computed on two different meshes: a coarse one (5152 triangles) and a fine one (132 006 triangles). Both are δ-admissible and their optimal values of δ are $\delta_1 = 1.095$ for the coarse grid and $\delta_2 = 5.599 \times 10^{-2}$ for the fine one.

In practice, these meshes would be called "very coarse" and "coarse", since there are only respectively 35 and 160 cells in the x direction. These choices are made in order to easily visualize the errors made by the schemes on the graphs.

Finally, the solutions are compared to reference solutions. When $\sigma = 0$, the reference solution is the exact solution of the corresponding 1D Riemann problem (see [30]). When $\sigma \neq 0$, the exact solution is not available so the reference solution is given by the grid-converged 1D asymptotic-preserving scheme described in [7].

Figure 2 shows the computed solutions along $x = \frac{1}{2}$ compared with the exact solution in the case $\sigma = 0$. Here, the conservation of admissible states is enforced by using $\theta_i^K = \max_{i \leq N_K} 2|D_i|/2\delta$ where $\delta = \delta_1$ on the coarse mesh and δ_2 on the fine one. The solution computed on the coarse grid is comparable to a 1D Rusanov scheme with a similar number of cells. On the other hand, since $\delta_2 \ll \delta_1$, the numerical diffusion of the scheme is much larger on the fine mesh than on the coarse one. As a consequence, the approximation is better on the coarse grid in this case.
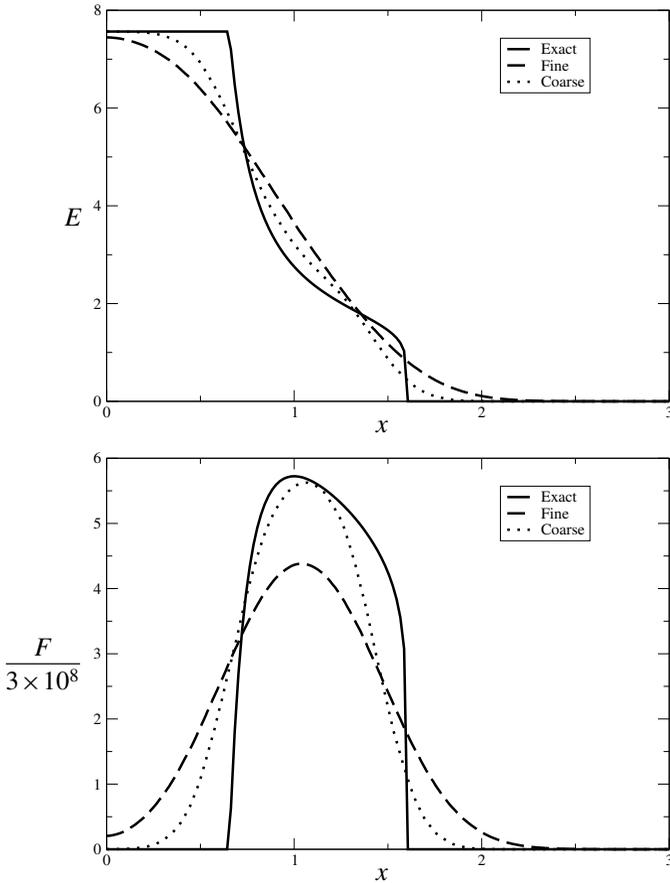
Figure 2. Exact and computed solutions along $x = \frac{1}{2}$ with $\sigma = 0$. Top: E. Bottom: F.

Now, if we set $\delta = \delta_1$ on the fine mesh, the quality of the approximation behaves as expected, i.e., the approximation is better on the fine grid (see Figure 3). Here, even if the condition required to preserve the set of admissible states is violated, the setup is not stiff enough to beget unphysical values in the scheme.

The same conclusions can be drawn when $\sigma = 1$ (see Figure 4), though the overall quality of the scheme is better than when $\sigma = 0$. Indeed, the scheme is designed to recover the diamond scheme in the limit, which is a better approximation of the equilibrium diffusion equation than the Rusanov scheme for the hyperbolic part. This is particularly true when θ_i^K is large. Therefore, the quality of the approximation is expected to increase with σ.

Finally, the asymptotic-preserving property is clarified through a last validation example. This time, we fix $\sigma = 1000$ and $t = 2 \times 10^{-9}$ and the results shown in Figure 5 are compared with a grid-converged 1D approximation of the equilibrium diffusion
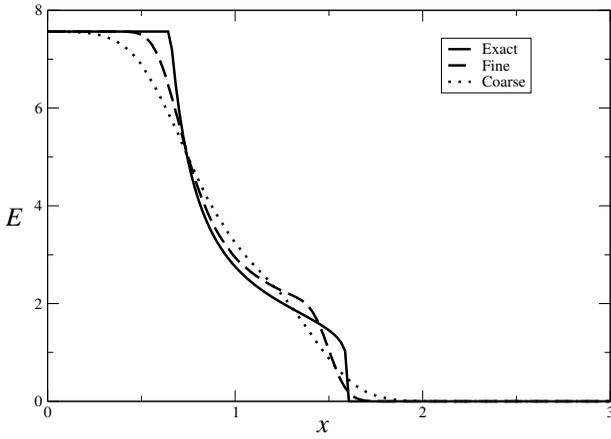
Figure 3. Reference and computed solutions along $x = \frac{1}{2}$ with $\sigma = 0$ and $\delta = \delta_1$.
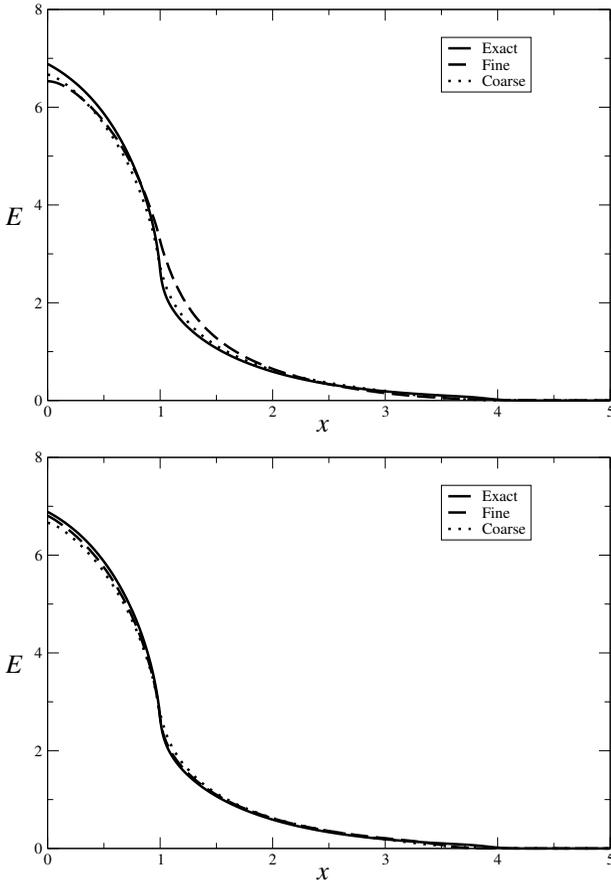
Figure 4. Reference and computed solutions along $x = \frac{1}{2}$ with $\sigma = 1$. Top: optimal choice of δ to enforce the preservation of $\mathcal{A}$. Bottom: same choice of δ for both schemes.
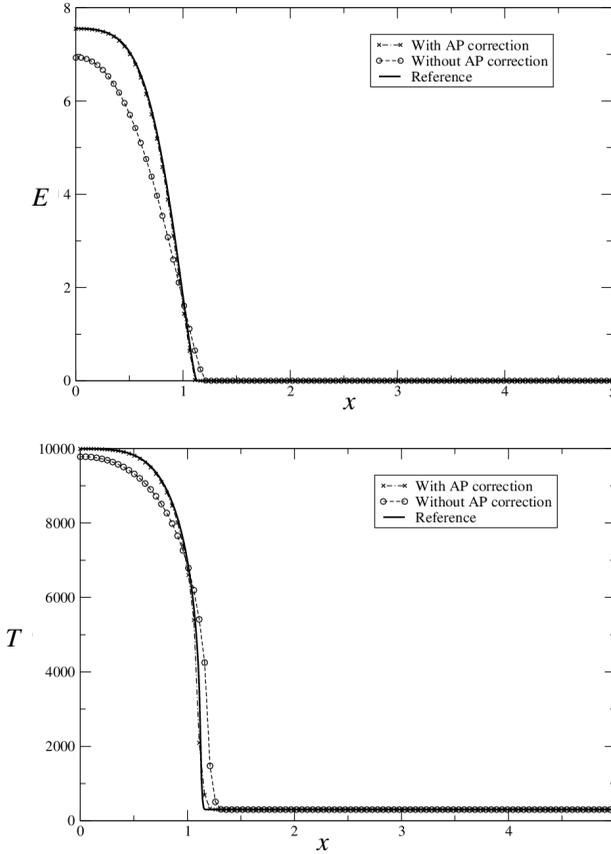
Figure 5. Reference and computed solutions with and without AP correction along $x = \frac{1}{2}$ with $\sigma = 1000$. Top: E. Bottom: T.

equation. The tests are performed with and without the asymptotic-preserving correction on the fine grid to see the impact of the asymptotic preservation. We immediately see that with the AP correction, the scheme provides an approximation which is nearly indistinguishable from the reference solution. On the other hand, as expected, if the AP correction is turned off (i.e., $\overline{\gamma^K} = 0$), there is a large discrepancy between the computed and reference solutions.

Radiative flow in a channel. In this subsection, a test case involving the evolution of radiation in a channel with multiple obstacles is performed. The setup is the following (see Figure 6). The entry condition on the left side of the channel models a beam of high energy ($F_L = c E_L = c\, a \times 10^{16}$) compared to the initial state of the domain ($F_0 = 0$, $E_0 = a \times 10^4$). The opacity σ is set to 1 and the M_1 model for radiative transfer is used. Moreover, 11 obstacles (with wall boundary conditions)
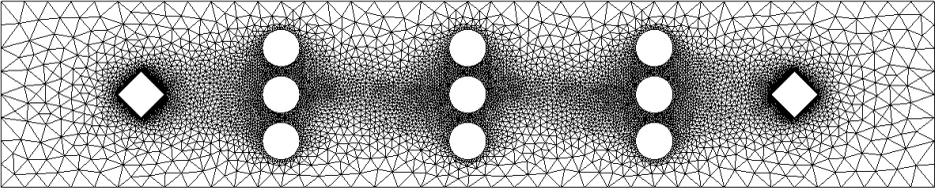
Figure 6. Mesh used for the radiative flow in a channel.

are scattered in the channel. A vertex-centered approached was used on a mesh consisting of 15348 cells refined near the obstacles.

Results for the energy E and Eddington's factor χ at time $t = 10^{-7}$ are shown on Figure 7. Let us emphasize that this case is numerically very challenging and that it is all the more critical to preserve the set of admissible states here since very small numerical errors may yield negative values for E or values of $F > cE$, which immediately cause the code to crash. Indeed, several values of θ_i^K were tested in order to investigate the optimality of the conditions in Theorem 2.2 and even a value 5% larger than the choice stated in the theorem produces inadmissible results. In this sense, it seems that the condition of Theorem 2.2 is optimal.

Optimization and parallelization. Let us conclude this work with a few words on the implementation of the scheme. First, in order to reduce the distance between a given element of the mesh and its neighbors, a renumbering procedure should be considered. We used a classic Reverse Cuthill–McKee procedure as preprocessing. Since it greatly reduces the L2 data cache misses, a significant gain may be observed. As an example, on the mesh used in the previous simulation, the code ran for 293 s without the renumbering and 235 s when renumbering was activated. This time included the Cuthill–McKee algorithm, so the gain is even better if another run is made on the same mesh.
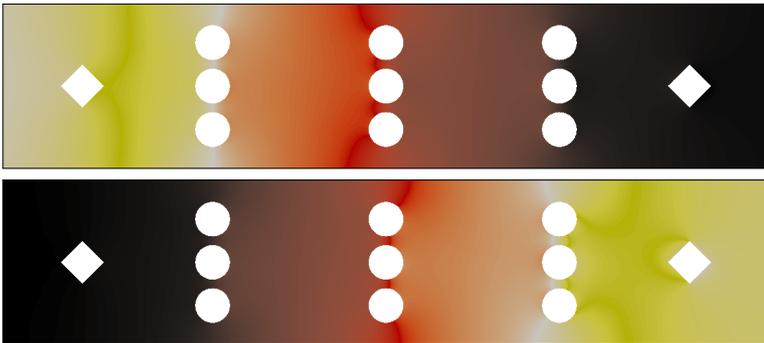
Figure 7. Radiative flow in a channel. Top: E. Bottom: χ.
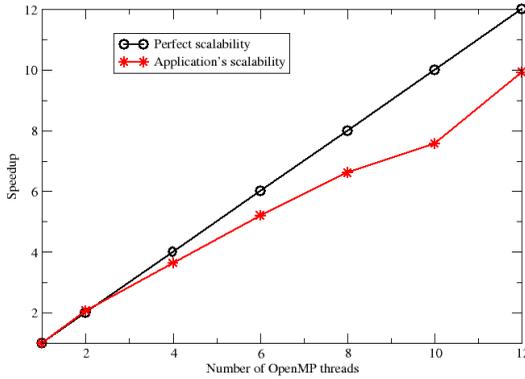
Figure 8. Scalability of the code.

Furthermore, since the scheme is explicit, the updates of the unknowns are independent from one another. Therefore, it is quite straightforward to parallelize the algorithm using OpenMP instructions (see http://openmp.org). On the one hand, such a strategy isn't suitable for massively parallel computations, but on the other hand it is very efficient on shared memory units and hence well adapted to multicore processors which are nowadays very common, even on personal computers. All the computations described above were run on a hexacore biprocessor node and a speed-up of 10 was obtained using all 12 cores of the machine. Figure 8 shows the scalability of the code on an example.

Finally, as it is classical for finite volume schemes, a loop on the edges is used (rather than on the elements for instance). In order to optimize the balance of work asked of each thread, we used an edge-based domain decomposition. This choice may save 20% or more time compared with an element domain decomposition since the number of edges per element is not constant. Table 1 shows the impact of such a choice on a representative example. Here, thread 6 has roughly 25% more work to do than thread 11.

Thread	elmt-based dec.		edge-based dec.		Thread	elmt-based dec.		edge-based dec.	
	# elmts	# edges	# elmts	# edges		# elmts	# edges	# elmts	# edges
0	5550	33879	5492	33303	6	5550	37032	4967	33301
1	5550	34713	5293	33303	7	5550	35200	5197	33304
2	5551	31906	5806	33301	8	5551	33475	5461	33300
3	5550	34391	5378	33300	9	5550	31999	5715	33305
4	5550	32865	5588	33304	10	5550	30538	6004	33299
5	5551	34282	5448	33299	11	5551	29338	6285	33299

Table 1. Impact of the domain decomposition strategy: element-based decomposition versus edge-based decomposition.

References

[1] B. Andreianov, M. Bendahmane, and K. H. Karlsen, *Discrete duality finite volume schemes for doubly nonlinear degenerate hyperbolic-parabolic equations*, J. Hyperbolic Differ. Equ. **7** (2010), no. 1, 1–67. MR 2011d:65241 Zbl 1207.35020

[2] B. Andreianov, F. Boyer, and F. Hubert, *Discrete duality finite volume schemes for Leray–Lions-type elliptic problems on general 2D meshes*, Numer. Methods Partial Differential Equations **23** (2007), no. 1, 145–195. MR 2008c:65283 Zbl 1111.65101

[3] D. Aregba-Driollet, M. Briani, and R. Natalini, *Time asymptotic high order schemes for dissipative BGK hyperbolic systems*, Numerische Mathematik (2015), 1–33 (English).

[4] C. Berthon, J. Dubois, B. Dubroca, T.-H. Nguyen-Bui, and R. Turpault, *A free streaming contact preserving scheme for the M_1 model*, Adv. Appl. Math. Mech. **2** (2010), no. 3, 259–285. MR 2011g:85009 Zbl 1262.65097

[5] C. Berthon, P. Charrier, and B. Dubroca, *An HLLC scheme to solve the M_1 model of radiative transfer in two space dimensions*, J. Sci. Comput. **31** (2007), no. 3, 347–389. MR 2008f:85004 Zbl 1133.85003

[6] C. Berthon, P. G. LeFloch, and R. Turpault, *Late-time/stiff-relaxation asymptotic-preserving approximations of hyperbolic equations*, Math. Comp. **82** (2013), no. 282, 831–860. MR 3008840 Zbl 1317.65182

[7] C. Berthon and R. Turpault, *Asymptotic preserving HLL schemes*, Numer. Methods Partial Differential Equations **27** (2011), no. 6, 1396–1422. MR 2838300 Zbl 1237.65100

[8] S. Bianchini, B. Hanouzet, and R. Natalini, *Asymptotic behavior of smooth solutions for partially dissipative hyperbolic systems with a convex entropy*, Comm. Pure Appl. Math. **60** (2007), no. 11, 1559–1622. MR 2010i:35227 Zbl 1152.35009

[9] F. Bouchut, H. Ounaissa, and B. Perthame, *Upwinding of the source term at interfaces for Euler equations with high friction*, Comput. Math. Appl. **53** (2007), no. 3-4, 361–375. MR 2008e:76120 Zbl 1213.76123

[10] F. Boyer and F. Hubert, *Finite volume method for 2D linear and nonlinear elliptic problems with discontinuities*, SIAM J. Numer. Anal. **46** (2008), no. 6, 3032–3070. MR 2009f:65231 Zbl 1180.35533

[11] C. Buet and S. Cordier, *An asymptotic preserving scheme for hydrodynamics radiative transfer models: numerics for radiative transfer*, Numer. Math. **108** (2007), no. 2, 199–221. MR 2008i:76138 Zbl 1127.76032

[12] C. Buet and B. Despres, *Asymptotic preserving and positive schemes for radiation hydrodynamics*, J. Comput. Phys. **215** (2006), no. 2, 717–740. MR 2007j:85005 Zbl 1090.76046

[13] C. Buet, B. Després, and E. Franck, *Design of asymptotic preserving finite volume schemes for the hyperbolic heat equation on unstructured meshes*, Numer. Math. **122** (2012), no. 2, 227–278. MR 2969268 Zbl 1263.65085

[14] C. Chalons, F. Coquel, E. Godlewski, P.-A. Raviart, and N. Seguin, *Godunov-type schemes for hyperbolic systems with parameter-dependent source: the case of Euler system with friction*, Math. Models Methods Appl. Sci. **20** (2010), no. 11, 2109–2166. MR 2011m:65179 Zbl 1213.35034

[15] C. Chalons, M. Girardin, and S. Kokh, *Large time step and asymptotic preserving numerical schemes for the gas dynamics equations with source terms*, SIAM J. Sci. Comput. **35** (2013), no. 6, A2874–A2902. MR 3138111 Zbl 1284.35262

[16] Y. Coudière and G. Manzini, *The discrete duality finite volume method for convection-diffusion problems*, SIAM J. Numer. Anal. **47** (2010), no. 6, 4163–4192. MR 2011e:65231 Zbl 1210. 65183

[17] Y. Coudière, J.-P. Vila, and P. Villedieu, *Convergence rate of a finite volume scheme for a two-dimensional convection-diffusion problem*, M2AN Math. Model. Numer. Anal. **33** (1999), no. 3, 493–516. MR 2000f:65086 Zbl 0937.65116

[18] K. Domelevo and P. Omnes, *A finite volume method for the Laplace equation on almost arbitrary two-dimensional grids*, M2AN Math. Model. Numer. Anal. **39** (2005), no. 6, 1203–1249. MR 2006j:65312 Zbl 1086.65108

[19] J. Droniou, R. Eymard, T. Gallouët, and R. Herbin, *A unified approach to mimetic finite difference, hybrid finite volume and mixed finite volume methods*, Math. Models Methods Appl. Sci. **20** (2010), no. 2, 265–295. MR 2011d:65318 Zbl 1191.65142

[20] J. Droniou, R. Eymard, T. Gallouet, and R. Herbin, *Gradient schemes: a generic framework for the discretisation of linear, nonlinear and nonlocal elliptic and parabolic equations*, Math. Models Methods Appl. Sci. **23** (2013), no. 13, 2395–2432. MR 3109434 Zbl 1281.65136

[21] J. Droniou and C. Le Potier, *Construction and convergence study of schemes preserving the elliptic local maximum principle*, SIAM J. Numer. Anal. **49** (2011), no. 2, 459–490. MR 2012e:65237 Zbl 1227.65100

[22] B. Dubroca and J.-L. Feugeas, *Étude théorique et numérique d'une hiérarchie de modèles aux moments pour le transfert radiatif*, C. R. Acad. Sci. Paris Sér. I Math. **329** (1999), no. 10, 915–920. MR 2000h:85005 Zbl 0940.65157

[23] R. Eymard, T. Gallouët, and R. Herbin, *Discretization of heterogeneous and anisotropic diffusion problems on general nonconforming meshes SUSHI: a scheme using stabilization and hybrid interfaces*, IMA J. Numer. Anal. **30** (2010), no. 4, 1009–1043. MR 2012d:65246 Zbl 1202.65144

[24] R. Eymard, T. Gallouët, and R. Herbin, *Finite volume methods*, Handbook of numerical analysis, 7: Solutions of equations in $\mathbb{R}$ (part 3) and Techniques of scientific computing (part 3) (P. G. Ciarlet and J. L. Lions, eds.), vol. VII, North-Holland, Amsterdam, 2000, pp. 713–1020. MR 2002e:65138 Zbl 0981.65095

[25] L. Gosse and G. Toscani, *An asymptotic-preserving well-balanced scheme for the hyperbolic heat equations*, C. R. Math. Acad. Sci. Paris **334** (2002), no. 4, 337–342. MR 2003b:65087 Zbl 0996.65093

[26] A. Harten, P. D. Lax, and B. van Leer, *On upstream differencing and Godunov-type schemes for hyperbolic conservation laws*, SIAM Rev. **25** (1983), no. 1, 35–61. MR 85h:65188 Zbl 0565.65051

[27] F. Hermeline, *Approximation of diffusion operators with discontinuous tensor coefficients on distorted meshes*, Comput. Methods Appl. Mech. Engrg. **192** (2003), no. 16-18, 1939–1959. MR 2004d:65121 Zbl 1037.65118

[28] S. Jin and C. D. Levermore, *Numerical schemes for hyperbolic conservation laws with stiff relaxation terms*, J. Comput. Phys. **126** (1996), no. 2, 449–467. MR 97g:65173 Zbl 0860.65089

[29] C. Le Potier, *Correction non linéaire et principe du maximum pour la discrétisation d'opérateurs de diffusion avec des schémas volumes finis centrés sur les mailles*, C. R. Math. Acad. Sci. Paris **348** (2010), no. 11-12, 691–695. MR 2012a:65294 Zbl 1193.65188

[30] C. Sarazin-Desbois, *Méthodes numériques pour des systèmes hyperboliques avec terme source provenant de physiques complexes autour du rayonnement*, Thèse de Doctorat, Université de Nantes Angers Le Mans, 2013.

CHRISTOPHE BERTHON: christophe.berthon@univ-nantes.fr
*Université de Nantes, Laboratoire de Mathématiques Jean Leray, 2 rue de la Houssinière,
44322 Nantes Cedex 3, France*

GUY MOEBS: guy.moebs@univ-nantes.fr
*Université de Nantes, Laboratoire de Mathématiques Jean Leray, 2 rue de la Houssinière,
44322 Nantes Cedex 3, France*

CÉLINE SARAZIN-DESBOIS: celine.sarazin@univ-nantes.fr
*Université de Nantes, Laboratoire de Mathématiques Jean Leray, 2 rue de la Houssinière,
44322 Nantes Cedex 3, France*

RODOLPHE TURPAULT: rodolphe.turpault@u-bordeaux.fr
*Bordeaux-INP, Institut de Mathématiques de Bordeaux, 351 cours de la Libération, 33400 Talence,
France*

msp

AN IMMERSED BOUNDARY METHOD FOR RIGID BODIES

BAKYTZHAN KALLEMOV, AMNEET PAL SINGH BHALLA,
BOYCE E. GRIFFITH AND ALEKSANDAR DONEV

We develop an immersed boundary (IB) method for modeling flows around fixed or moving rigid bodies that is suitable for a broad range of Reynolds numbers, including steady Stokes flow. The spatio-temporal discretization of the fluid equations is based on a standard staggered-grid approach. Fluid-body interaction is handled using Peskin's IB method; however, unlike existing IB approaches to such problems, we do not rely on penalty or fractional-step formulations. Instead, we use an unsplit scheme that ensures the no-slip constraint is enforced exactly in terms of the Lagrangian velocity field evaluated at the IB markers. Fractional-step approaches, by contrast, can impose such constraints only approximately, which can lead to penetration of the flow into the body, and are inconsistent for steady Stokes flow. Imposing no-slip constraints exactly requires the solution of a large linear system that includes the fluid velocity and pressure as well as Lagrange multiplier forces that impose the motion of the body. The principal contribution of this paper is that it develops an efficient preconditioner for this exactly constrained IB formulation which is based on an analytical approximation to the Schur complement. This approach is enabled by the near translational and rotational invariance of Peskin's IB method. We demonstrate that only a few cycles of a geometric multigrid method for the fluid equations are required in each application of the preconditioner, and we demonstrate robust convergence of the overall Krylov solver despite the approximations made in the preconditioner. We empirically observe that to control the condition number of the coupled linear system while also keeping the rigid structure impermeable to fluid, we need to place the immersed boundary markers at a distance of about two grid spacings, which is significantly larger from what has been recommended in the literature for elastic bodies. We demonstrate the advantage of our monolithic solver over split solvers by computing the steady state flow through a two-dimensional nozzle at several Reynolds numbers. We apply the method to a number of benchmark problems at zero and finite Reynolds numbers, and we demonstrate first-order convergence of the method to several analytical solutions and benchmark computations.

I. Introduction

A large number of numerical methods have been developed to simulate interactions between fluid flows and immersed bodies. For rigid bodies or bodies with prescribed kinematics, many of these approaches [78; 13; 76; 84; 71] are based on the immersed boundary (IB) method of Peskin [64]. The simplicity, flexibility, and power of the IB method for handling a broad range of fluid-structure interaction problems was demonstrated by Bhalla et al. [13]. In that study, the authors showed that the IB method can be used to model complex flows around rigid bodies moving with specified kinematics (e.g., swimming fish or beating flagella) as well as to compute the motion of freely moving bodies driven by flow. In the approach of Bhalla et al., as well as in those of others [78; 76; 71; 84], the rigidity constraint enforcing that the fluid follows the motions of the rigid bodies is imposed only *approximately*. Here and throughout this manuscript, when we refer to the *no slip condition*, we mean the requirement that the interpolated fluid velocity exactly match the rigid body velocity at the positions of the IB marker points. In this work, we develop an effective solution approach to an IB formulation of this problem that *exactly* enforces both the incompressibility and no-slip constraints, thus substantially improving upon a large number of existing techniques.

A simple approach to implementing rigid bodies using the traditional IB method is to use stiff springs to attach markers that discretize the body to tether points constrained to move as a rigid body [77]. This penalty-spring approach leads to numerical stiffness and, when the forces are handled explicitly, requires very small time steps. For this reason, a number of *direct forcing* IB methods [70] have been developed that aim to constrain the flow inside the rigid body by treating the fluid-body force as a Lagrange multiplier Λ enforcing a no-slip constraint at the locations of the IB markers. However, to our knowledge, all existing direct forcing IB methods use some form of time step splitting to separate the coupled fluid-body problem into more manageable pieces. The basic idea behind these approaches is first to solve a simpler system in which a number of the constraints (e.g., incompressibility, or no-slip along the fluid-body interface) are ignored. The solution of the unconstrained problem is then *projected* onto the constraints, which yields *estimates* of the true Lagrange multipliers. In most existing methods, the fluid solver uses a fractional time stepping scheme, such as a version of Chorin's projection method, to separate the velocity update from the pressure update [78; 76; 71]. Taira and Colonius also use a fractional time-stepping approach in which they split the velocity from the Lagrange multipliers (π, Λ). They obtain approximations to (π, Λ) in a manner similar to that in a standard projection method for the incompressible Navier–Stokes equations. A modified Poisson-type problem (see [76, Equation (26)]) determines the Lagrange multipliers and is solved using an unpreconditioned conjugate gradient

method. The method developed in [13] avoids the pressure-velocity splitting and instead uses a combined iterative Stokes solver, and in [84] (see supplementary material), periodic boundary conditions are applied, which allows for the use of a pseudospectral method. In both works, however, time step splitting is still used to separate the computation of the rigidity constraint forces from the updates to the fluid variables. In the approach described in the supplementary material to [84], the projection step of the solution onto the rigidity constraint is performed twice in a predictor-corrector framework, which improves the imposition of the constraint; however, this approach does not control the accuracy of the approximation of the constraint forces. Curet et al. [25] and Ardekani et al. [5] go a step closer in the direction of exactly enforcing the rigidity constraint by iterating the correction until the relative slip between the desired and imposed kinematics inside the rigid body reaches a relatively loose tolerance of 1%. The scheme used in [25] is essentially a fixed-point (Richardson) iteration for the constrained fluid problem, which uses splitting to separate the update of the Lagrange multipliers from a fluid update based on the SIMPLER scheme [63]. Unlike the approach developed here, fixed point iterations based on splitting are not guaranteed to converge, yet alone converge rapidly, especially in the steady Stokes regime for tight solver tolerances.

An alternative view of direct forcing methods that use time step splitting is that they are penalty methods for the unsplit problem, in which the penalty parameter is related to the time step size. Such approaches inherently rely on inertia and implicitly assume that fluid velocity has memory. Consequently, *all* such splitting methods fail in the steady Stokes limit. Furthermore, even at finite Reynolds numbers, methods based on splitting cannot exactly satisfy the no-slip constraint at fluid-body interfaces. Such methods can thereby produce undesirable artifacts in the solution, such as penetration of the flow through a rigid obstacle. It is therefore desirable to develop a numerical method that solves for velocity, pressure, and fluid-body forces in a single step with controlled accuracy and reasonable computational complexity.

The goal of this work is to develop an effective IB method for rigid bodies that does not rely on any splitting. Our method is thus applicable over a broad range of Reynolds numbers, including steady Stokes flow, and is able to impose rigidity constraints exactly. This approach requires us to solve large linear systems for velocity, pressure, and fluid-body interaction forces. This linear system is not new. For example, Equation (13) in [76] is essentially the same system of equations that we study here. The primary contributions of this work are that we do not rely on any approximations when solving this linear system, and that we develop an effective preconditioner based on an approximation of the Schur complement that allows us to solve (3). The resulting method has a computational complexity that is only a few times larger than the corresponding problem in the absence of rigid

bodies. In the context of steady Stokes flows, a rigid-body IB formulation very similar to the one we use here has been developed by Bringley and Peskin [16]; however, that formulation relies on periodic boundary conditions, and uses a very different spatial discretization and solution methodology from the approach we describe here. Our approach can readily handle a broad range of specified boundary conditions. In both [16] and a very recent work by Stein et al. [70] on a higher-order IB smooth extension method for scalar (e.g., Poisson) equations, the Schur complement is formed densely in an expensive precomputation stage. By contrast, in the method proposed here we build a simple *physics-based* approximation of the Schur complement that can be computed "on the fly" in a scalable and efficient manner.

Our basic solution approach is to use a preconditioned Krylov solver for the fully constrained fluid problem, as has been done for some time in the context of finite element methods for fluid flows interacting with elastic bodies [42; 31]. A key difficulty that we address in this work is the development of an efficient preconditioner for the constrained formulation. To do so, we construct an *analytical* approximation of the Schur complement (i.e., the mobility matrix) corresponding to Lagrangian rigidity forces (i.e., Lagrange multipliers) enforcing the no-slip condition at the positions of the IB markers. We rely on the near translational and rotational invariance of Peskin's IB method to approximate the Schur complement, following techniques commonly used for suspensions of rigid spheres in steady Stokes flow such as Stokesian dynamics [69; 75], bead methods for rigid macromolecules [46; 20; 61; 26] and the method of regularized Stokeslets [23; 21; 22]. In fact, as we explain herein, many of the techniques developed in the context of steady Stokes flow can be used with the IB method both at zero and also, perhaps more surprisingly, finite Reynolds numbers.

The method we develop offers an attractive alternative to existing techniques in the context of steady or nearly steady Stokes flow of suspensions of rigid particles. To our knowledge, most other approaches tailored to the steady Stokes limit rely on Green's functions for Stokes flow to eliminate the (Eulerian) fluid degrees of freedom and solve only for the (Lagrangian) degrees of freedom associated to the surface of the body. Because these approaches rely on the availability of analytical solutions, handling nontrivial boundary conditions (e.g., bounded systems) is complicated [86] and has to be done on a case-by-case basis [48; 72; 73; 74; 55; 2; 56]. By contrast, in the method developed here, analytical Green's functions are replaced by an "on the fly" computation that may be carried out by a standard finite-volume, finite-difference, or finite-element fluid solver.[1] Such solvers can readily handle nontrivial boundary conditions. Furthermore, suspensions at small but nonzero Reynolds

---

[1]In this work, we use a staggered-grid discretization on a uniform grid combined with multigrid-preconditioned Stokes solvers [34; 18].

numbers can be handled without any extra work. Additionally, we avoid uncontrolled approximations relying on truncations of multipole expansions to a fixed order [69; 58; 9; 32], and we can seamlessly handle arbitrary body shapes and deformation kinematics. For problems involving active particles [53], it is straightforward to add osmophoretic or electrophoretic coupling between the fluid flow and additional fluid variables such as the electric potential or the concentration of charged ions or chemical reactants. Lastly, in the spirit of fluctuating hydrodynamics [27; 50; 6], it is straightforward to generate the stochastic increments required to simulate the Brownian motion of small rigid particles suspended in a fluid by including a fluctuating stress in the fluid equations. We also point out that our method also has some disadvantages compared to methods such as boundary integral or boundary element methods. Notably, it requires filling the domain with a dense uniform fluid grid, which is expensive at low densities. It is also a low-order method that cannot compute solutions as accurately as spectral boundary integral formulations. We do believe, nevertheless, that the method developed here offers a good compromise between accuracy, efficiency, scalability, flexibility and extensibility, compared to other more specialized formulations.

II. Semicontinuum formulation

Our notation uses the following conventions where possible. Vectors (including multivectors), matrices, and operators are bolded, but when fully indexed down to a scalar quantity we no longer bold the symbol; matrices and operators are also scripted. We denote Eulerian quantities with lowercase letters, and the corresponding Lagrangian quantity with the same capital letter. We use the Latin indexes i, j, k, l, m to denote a specific fluid grid point or IB marker (i.e., physical location with which degrees of freedom are associated), the indices p, q, r, s, t to denote a specific body in the multibody context, and Greek superscripts α, β, γ to denote specific Cartesian components. For example, $\boldsymbol{v}$ denotes fluid velocity (either continuum or discrete), with v_k^α being the fluid velocity in direction α associated with the face center k, and $\boldsymbol{V}$ denotes the velocity of all IB markers, with V_i^α being the velocity of marker i along direction α. Our formulation is easily extended to a collection of rigid bodies, but for simplicity of presentation, we focus on the case of a single body.

We consider a region $\mathcal{D} \subset \mathbb{R}^d$ ($d = 2$ or 3) that contains a single rigid body $\Omega \subset \mathcal{D}$ immersed in a fluid of density ρ and shear viscosity η. The computational domain $\mathcal{D}$ could be a periodic region (topological torus), a finite box, an infinite domain, or some combination thereof, and we will implicitly assume that some consistent set of boundary conditions are prescribed on its boundary $\partial \mathcal{D}$ even though we will not explicitly write this in the formulation. We require that the linear velocity of a given

reference point (e.g., the center of mass of the body) $U(t)$ and the angular velocity $\Omega(t)$ of the body are specified functions of time, and without loss of generality, we assume that the rigid body is at rest.[2] In addition to features of the fluid flow, typical quantities of interest are the total drag force $F(t)$ and total drag torque $T(t)$ between the fluid and the body. Another closely related problem to which IB methods can be extended is the case when the motion of the rigid body (i.e., $U(t)$ and $\Omega(t)$) is not known but the body is subject to specified external force $F(t)$ and torque $T(t)$. For example, in the sedimentation of rigid particles in suspension, the external force is gravity and the external torque is zero. Handling this *free kinematics* problem [14; 13] requires a nontrivial extension of our formulation and numerical algorithm.

In the immersed boundary (IB) method [64; 37; 35], the velocity field $v(r, t)$ is extended over the whole domain $\mathcal{D}$, *including* the body interior. The body is discretized using a collection of *markers*, which is a set of N points that cover the interior of the body and at which the interaction between the body and the fluid is localized. For example, the markers could be the nodes of a triangular ($d = 2$) or tetrahedral ($d = 3$) mesh used to discretize Ω; an illustration of such a *volume* grid of markers discretizing a rigid disk immersed in steady Stokes flow is shown in the left panel of Figure 1. In the case of Stokes flow, the specification of a no slip condition on the boundary of a rigid body is sufficient to ensure rigidity of the fluid inside the body [23]. Therefore, for Stokes flow, the grid of markers does not need to extend over the volume of the body and can instead be limited to the surface of the rigid body, thus substantially reducing the number of markers required to represent the body. In this case, the markers could be the nodes of a triangulation ($d = 3$) of the surface of the body; an illustration of such a *surface* grid of markers is shown in the right panel of Figure 1. We discuss the differences between a volume and a surface grid of markers in Section VII.

The traditional IB method is concerned with the motion of elastic (flexible) bodies in fluid flow, and the collection of markers can be viewed as a set of quadrature points used to discretize integrals over the moving body. The elastic body forces are most easily computed in a Lagrangian coordinate system attached to the deforming body, and the relative positions of the markers in the fixed Eulerian frame of reference generally vary in time. For a rigid body, however, the relative positions of the markers do not change, and it is not necessary to introduce two distinct coordinate frames. Instead, we use the same Cartesian coordinate system to describe points in the fluid domain and in the body; the positions of the N markers in this fixed frame of reference will be denoted with $R = \{R_1, \ldots, R_N\}$, where $R \subset \Omega$ for volume meshes or $R \subset \partial\Omega$ for surface meshes.

---

[2]The case of more general specified kinematics is a straightforward generalization and does not incur any additional mathematical or algorithmic complexity [14; 13].
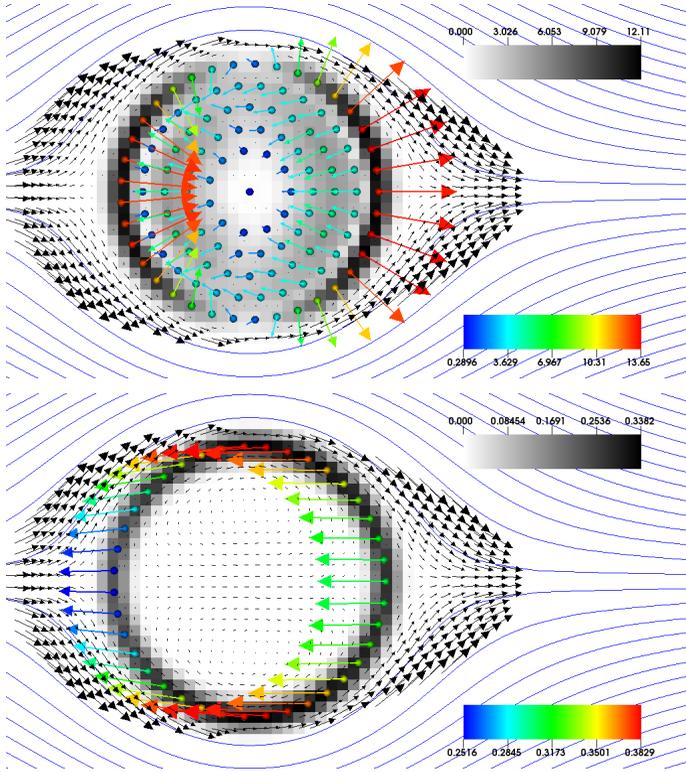
Figure 1. Two-dimensional steady Stokes flow past a periodic column of circular cylinders (disks) at zero Reynolds number obtained using our rigid-body IB method (the same setup is also studied at finite Reynolds numbers in Section F). The markers used to mediate the fluid-body interaction are shown as small colored circles. The Lagrangian constraint forces $\boldsymbol{\Lambda}$ that keep the markers at their fixed locations are shown as colored vectors; the color of the vectors and the corresponding marker i are based on the magnitude of the constraint force $\boldsymbol{\Lambda}_i$ (see color bar). The fluid velocity field is shown as a vector field (black arrows) in the vicinity and the interior of the body; further from the body, flow streamlines are shown as solid blue lines. The magnitude of the Eulerian constraint force $\mathcal{S}\boldsymbol{\Lambda}$ is shown as a gray color plot (see grayscale bar). Top panel: a volume marker grid of 121 markers is used to discretize the disk. The majority of the constraint forces are seen to act near the surface of the body, but nontrivial constraint forces are seen also in the interior of the body. Bottom panel: a surface grid of 39 markers is used to discretize the disk, which strictly localizes the constraint forces to the surface of the body.

In the standard IB method for flexible immersed bodies, elastic forces are computed in the Lagrangian frame and then *spread* to the fluid in the neighborhood of the markers using a regularized delta function $\delta_a(\boldsymbol{r})$ that integrates to unity and converges to a Dirac delta function as the regularization width $a \to 0$. The regularization length scale a is typically chosen to be on the order of the spacing

between the markers (as well as the lattice spacing of the grid used to discretize
the fluid equations), as we discuss in more detail later. In turn, the motion of the
markers is specified to follow the velocity of the fluid *interpolated* at the positions
of the markers.

   The key difference between an elastic and a rigid body is that, for a rigid object,
the motion of the markers is known (e.g, they are fixed in place or move with a spec-
ified velocity) and the body forces are unknown and must be determined within each
time step. To obtain the fluid-marker interaction forces $\boldsymbol{\Lambda}(t) = \{\boldsymbol{\Lambda}_i(t), \ldots, \boldsymbol{\Lambda}_N(t)\}$
that constrain the motion of the N markers, we solve for the Eulerian velocity field
$\boldsymbol{v}(\boldsymbol{r}, t)$, the Eulerian pressure field $\pi(\boldsymbol{r}, t)$, and the Lagrangian constraint forces
$\boldsymbol{\Lambda}_i(t)$ the system

$$\rho(\partial_t \boldsymbol{v} + \boldsymbol{v} \cdot \nabla \boldsymbol{v}) + \nabla \pi = \eta \nabla^2 \boldsymbol{v} + \sum_{i=1}^{N} \boldsymbol{\Lambda}_i \delta_a(\boldsymbol{R}_i - \boldsymbol{r}),$$

$$\nabla \cdot \boldsymbol{v} = 0, \tag{1}$$

$$\boldsymbol{V}_i = \int \delta_a(\boldsymbol{R}_i - \boldsymbol{r}) \boldsymbol{v}(\boldsymbol{r}, t) \, d\boldsymbol{r} = 0, \quad i = 1, \ldots, N,$$

along with suitable boundary conditions. In the case of steady Stokes flow, we set
$\rho = 0$. The first two equations are the incompressible Navier–Stokes equations with
an Eulerian constraint force

$$\boldsymbol{\lambda}(\boldsymbol{r}, t) = \sum_{i=1}^{N} \boldsymbol{\Lambda}_i \delta_a(\boldsymbol{R}_i - \boldsymbol{r}).$$

The last condition is the *rigidity constraint* that requires that the Eulerian velocity
averaged around the position of marker i must match the known marker velocity
$\boldsymbol{V}_i$. This constraint enforces a regularized no-slip condition at the locations of
the IB markers, which is a numerical approximation of the true no-slip condition
on the surface (or interior) of the body. Observe that flow may still penetrate the
body in-between the markers and this leads to a well-known small but nonzero
"leak" in the traditional Peskin IB method. This leak can be greatly reduced by
adopting a staggered-grid formulation [36], as done in the present work. Other
more specialized approaches to reducing spurious fluxes in the IB method have
been developed [65; 38], but will not be considered in this work.

   Notice that for zero Reynolds number, the semicontinuum formulation (1) is
closely related to the popular method of regularized Stokeslets, which solves a
similar system of equations for $\boldsymbol{\Lambda}$ [23; 21]. The key difference[3] is that in the
method of regularized Stokeslets, the fluid equations are eliminated using analytic

---

[3]Another important difference is that we follow Peskin and use the regularized delta function both
for spreading and interpolation (this ensures energy conservation in the formulation [64]), whereas in
the method of Regularized Stokeslets only the spreading uses a regularized delta function. Our choice

Green's functions; this necessitates that nontrivial precomputations of these Green's functions be performed for each type of boundary condition [2; 56].

In this work, we treat (1) as the primary continuum formulation of the problem. This is a semicontinuum formulation in which the rigid body is represented as a discrete collection of markers but the fluid description is kept as a continuum, which implies that different discretizations of the fluid equations are possible. One can, in principle, try to write a fully continuum formulation in which the discrete set of rigidity forces $\boldsymbol{\Lambda}$ are replaced by a continuum force density field $\boldsymbol{\lambda}(\boldsymbol{R} \in \Omega, t)$. The well-posedness and stability of such a fully continuum formulation is mathematically delicate, however, and there can be subtle differences between weak and strong interpretations of the equations. To appreciate this, observe that if each component of the velocity is discretized with N_f degrees of freedom, it cannot in general be possible to constrain the velocity strongly at more than N_f points (markers). By contrast, in our strong formulation (1), the velocity is infinite dimensional but it is only constrained in the vicinity of a finite number of markers. Therefore, the problem (1) is always well posed and is directly amenable to numerical discretization and solution, at least when it is well-conditioned. As we show in this work, the conditioning of the fully discrete problem is controlled by the relationship between the regularization length a and the marker spacing.

The physical interpretation of the constraint forces $\boldsymbol{\Lambda}_i$ depends on details of the marker grid and the type of the problem under consideration. For fully continuum formulations, in which the fluid-body interaction is represented solely as a surface force density, the force $\boldsymbol{\Lambda}_i$ can be interpreted as the integral of the traction (normal component of the fluid stress tensor) over a surface area associated with marker i. Such a formulation is appropriate, for example, for steady Stokes flow. In particular, for steady Stokes flow our method can be seen as a discretized and regularized first-kind integral formulation in which Green's functions are computed by the fluid solver. This approach is different from the method of regularized Stokeslets, in which regularized Green's functions must be computed analytically [21; 23].

For cases in which markers are placed on both the surface and the interior of a rigid body, the precise physical interpretation of the volume force density, and thus of $\boldsymbol{\Lambda}$, is delicate even for steady Stokes flow. Notably, observe that the splitting between a volume constraint force density and the gradient of the pressure is not unique because the pressure inside a rigid body cannot be determined uniquely. Specifically, only the component of the constraint force density projected onto the space of divergence-free vector fields is uniquely determined. In the presence of finite inertia and a density mismatch between the fluid and the *moving* rigid bodies, the inertial terms in (1) need to be modified in the interior of the body [9].

---

ensures that the linear system we solve is symmetric and positive semidefinite, which is crucial if one wishes to account for Brownian motion and thermal fluctuations.

Furthermore, sufficiently many markers in the interior of the body are required to prevent spurious angular momentum being generated by motions of the fluid inside the body [78]. We do not discuss these physical issues in this work because they do not affect the numerical algorithm, and because we restrict our numerical studies to flow past *stationary* rigid bodies, for which the fluid-body interaction force is localized to the surface of the body in the continuum limit.

III. Discrete formulation

The spatial discretization of the fluid equation uses a uniform Cartesian grid with grid spacing h and is based on a second-order accurate staggered-grid finite-difference discretization, in which vector-valued quantities, including velocities and forces, are represented on the faces of the Cartesian grid cells, and scalar-valued quantities, including the pressure, are represented at the centers of the grid cells [37; 35; 9; 13]. Our implicit-explicit temporal discretization of the Navier–Stokes equation is standard and summarized in prior work; see for example the work of Griffith [35]. The key features are that we treat advection explicitly using a predictor-corrector approach, and that we treat viscosity implicitly, using either the backward Euler or the implicit midpoint method. For steady Stokes flow, no temporal discretization required, although one can also think of this case as corresponding to a backward Euler discretization of the time-dependent problem with a very large time step size Δt. A key dimensionless quantity is the viscous CFL number $\beta = \nu \Delta t / h^2$, where the kinematic viscosity is $\nu = \eta / \rho$. If β is small, the pressure and velocity are weakly coupled, but for large β, and in particular for the steady Stokes limit $\beta \to \infty$, the coupling between the velocity and pressure equations is strong.

We do *not* use a fractional time-stepping scheme (i.e., a projection method) to split the pressure and velocity updates; instead, the pressure is treated as a Lagrange multiplier that enforces the incompressibility and must be determined together with the velocity at the end of the time step [34]; except in special cases, this is *necessary* for small Reynolds number flows. This approach also greatly aids with imposing stress boundary conditions [34]. The constraint force $\lambda(r, t)$ is treated analogously to the pressure, i.e., as a Lagrange multiplier. Whereas the role of the pressure is to enforce the incompressibility constraint, λ enforces the rigidity constraint. Like the pressure, λ is an unknown that must be solved for in this formulation.

A. *Force spreading and velocity interpolation.* In the fully discrete formulation of the fluid-body coupling, we replace spatial integrals by sums over fluid or body grid points in the semicontinuum formulation (1). The regularized delta function is discretized using a tensor product in d-dimensional space (see [9] for more details),

$$\delta_a(\boldsymbol{r}) = h^{-d} \prod_{\alpha=1}^{d} \phi_a(r_\alpha),$$

where h^d is the volume of a grid cell. The one-dimensional kernel function ϕ_a is chosen based on numerical considerations of efficiency and maximized approximate translational invariance [64]. In this work, for reasons that will become clear in Section IV, we prefer to use a kernel that maximizes translational and rotational invariance (i.e., improves grid-invariance). We therefore use the smooth (three-times differentiable) six-point kernel recently described by Bao et al. [10]. This kernel is more expensive than the traditional four-point kernel [64] because it increases the support of the kernel to $6^2 = 36$ grid points in two dimensions and $6^3 = 216$ grid points in three dimensions; however, this cost is justified because the new six-point kernel improves the translational invariance by orders of magnitude compared to other standard IB kernel functions [10].

The interaction between the fluid and the rigid body is mediated through two crucial operations. The discrete velocity-interpolation operator $\mathcal{J}$ averages velocities on the staggered grid in the neighborhood of marker i via

$$(\mathcal{J}v)_i^\alpha = \sum_k v_k^\alpha \, \phi_a(\boldsymbol{R}_i - \boldsymbol{r}_k^\alpha),$$

where the sum is taken over faces k of the grid, α indexes coordinate directions (x, y, z) as a superscript, and r_k^α is the position of the center of the grid face k in the direction α. The discrete force-spreading operator $\mathcal{S}$ spreads forces from the markers to the faces of the staggered grid via

$$(\mathcal{S}\Lambda)_k^\alpha = h^{-d} \sum_i \Lambda_i^\alpha \, \phi_a(\boldsymbol{R}_i - \boldsymbol{r}_k^\alpha), \tag{2}$$

where now the sum is over the markers that define the configuration of the rigid body. These operators are adjoint with respect to a suitably defined inner product, $\mathcal{J} = \mathcal{S}^\star = h^d \, \mathcal{S}^T$, which ensures conservation of energy [64]. Extensions of the basic interpolation and spreading operators to account for the presence of physical boundary conditions are described in Appendix D.

B. *Rigidly constrained Stokes problem.*

At every stage of the temporal integrator, we need to solve a linear system of the form

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} & -\mathcal{S} \\ -\mathcal{D} & 0 & 0 \\ -\mathcal{J} & 0 & 0 \end{bmatrix} \begin{bmatrix} v \\ \pi \\ \Lambda \end{bmatrix} = \begin{bmatrix} g \\ h = 0 \\ W = 0 \end{bmatrix}, \tag{3}$$

which is the focus of this work. The right-hand side g includes all remaining fluid forcing terms, explicit contributions from previous time steps or stages, boundary conditions, etc. Here, $\mathcal{G}$ is the discrete gradient operator, $\mathcal{D} = -\mathcal{G}^T$ is the discrete divergence operator, and $\mathcal{A}$ is the vector equivalent of the familiar screened Poisson

(or Helmholtz) operator

$$\mathcal{A} = \frac{\rho}{\Delta t}\mathcal{I} - \frac{\kappa \eta}{h^2}\mathcal{L}_v,$$

with $\kappa = 1$ for the backward Euler method or for steady Stokes, and $\kappa = 1/2$ for the implicit midpoint rule. Here $\mathcal{L}_v$ is the dimensionless vector Laplacian operator, which takes into account boundary conditions for velocity such as no-slip boundaries. Since the viscosity appears multiplied by the coefficient κ, we will henceforth absorb this coefficient into the viscosity, $\eta \leftarrow \kappa \eta$, which allows us to assume, without loss of generality, that $\kappa=1$ and to write the fluid operator in the form

$$\mathcal{A} = \eta h^{-2}(\beta^{-1}\mathcal{I} - \mathcal{L}_v). \tag{4}$$

We remark that making the $(3, 3)$ block in the matrix in (3) nonzero (i.e., regularizing the saddle-point system) is closely related to solving the Brinkman equations [17] for flow through a permeable or porous body suspended in fluid [84]. In particular, by making the $(3, 3)$ block a diagonal matrix with suitable diagonal elements, one can consistently discretize the Brinkman equations. Such regularization greatly simplifies the numerical linear algebra except, of course, when the permeability of the body is so small that it effectively acts as an impermeable body. In this work, we focus on developing a solver for (3) that is effective even when there is no regularization (permeability), and even when the matrix $\mathcal{A}$ is the discretization of an elliptic operator, as is the case in the steady Stokes regime. This is the hardest case to consider, and a solver that is robust in this case will be able to handle the easier cases of finite Reynolds number or permeable bodies with ease.

It is worth noticing the structure of the linear system (3). First, observe that the system is symmetric, at least if only simple boundary conditions such as periodic or no-slip boundaries are present [34]. In the top 1×1 block, $\mathcal{A} \succsim \mathbf{0}$ is a symmetric positive-semidefinite (SPD) matrix. The top left 2×2 block represents the familiar saddle-point problem arising when solving the Navier–Stokes or Stokes equations in the absence of a rigid body [34]. The whole system is a saddle-point problem for the fluid variables and for $\mathbf{\Lambda}$, in which the top-left block is the Stokes saddle-point matrix.

C. *Mobility matrix.* We can formally solve (3) through a Schur complement approach, as described in more detail in Section V. For increased generality, which will be useful when discussing preconditioners, we allow the right hand side to be general and, in particular, do not assume that $\mathbf{h}$ and $\mathbf{W}$ are zero.

First, we solve the unconstrained fluid equation for pressure and velocity

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} \\ -\mathcal{D} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \pi \end{bmatrix} = \begin{bmatrix} \mathcal{S}\mathbf{\Lambda} + \mathbf{g} \\ \mathbf{h} \end{bmatrix}, \tag{5}$$

where we recall that $\mathcal{A} = \eta h^{-2}(\beta^{-1}\mathcal{I} - \mathcal{L}_v)$. The solution can be written as $\boldsymbol{v} = \mathcal{L}^{-1}(\boldsymbol{\mathcal{S}}\boldsymbol{\Lambda} + \boldsymbol{g}) + \mathcal{L}_p^{-1}\boldsymbol{h}$, where $\mathcal{L}^{-1}$ is the standard Stokes solution operator for divergence-free flow ($\boldsymbol{h} = 0$), given by

$$\mathcal{L}^{-1} = \mathcal{A}^{-1} - \mathcal{A}^{-1}\mathcal{G}(\mathcal{D}\mathcal{A}^{-1}\mathcal{G})^{-1}\mathcal{D}\mathcal{A}^{-1}, \tag{6}$$

where we have assumed for now that $\mathcal{A}^{-1}$ is invertible. For a periodic system, the discrete operators commute, and we can write

$$\mathcal{L}^{-1} = \mathcal{P}\mathcal{A}^{-1} = (\mathcal{I} - \mathcal{G}(\mathcal{D}\mathcal{G})^{-1}\mathcal{D})\mathcal{A}^{-1}, \tag{7}$$

where $\mathcal{P}$ is the Helmholtz projection onto the space of divergence-free vector fields. We never explicitly compute or form $\mathcal{L}^{-1}$; rather, we solve the Stokes velocity-pressure subsystems using the projection-method based preconditioner developed by Griffith [34]. Let us define $\tilde{\boldsymbol{v}} = \mathcal{L}^{-1}\boldsymbol{f} + \mathcal{L}_p^{-1}\boldsymbol{h}$ to be the solution of the *unconstrained* Stokes problem

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} \\ -\mathcal{D} & 0 \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{v}} \\ \tilde{\pi} \end{bmatrix} = \begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{h} \end{bmatrix}, \tag{8}$$

giving $\boldsymbol{v} = \tilde{\boldsymbol{v}} + \mathcal{L}^{-1}\boldsymbol{\mathcal{S}}\boldsymbol{\Lambda}$.

Next, we plug the velocity $\boldsymbol{v}$ into the rigidity constraint, $\mathcal{J}\boldsymbol{v} = -\boldsymbol{W}$, to obtain

$$\mathcal{M}\boldsymbol{\Lambda} = -(\boldsymbol{W} + \mathcal{J}\tilde{\boldsymbol{v}}), \tag{9}$$

where the Schur complement or *marker mobility matrix* is

$$\mathcal{M} = \mathcal{J}\mathcal{L}^{-1}\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{S}}^{\star}\mathcal{L}^{-1}\boldsymbol{\mathcal{S}}. \tag{10}$$

The mobility matrix $\mathcal{M} \succsim \boldsymbol{0}$ is SPD and has dimensions $dN \times dN$, and the $d \times d$ block $\mathcal{M}_{ij}$ relates the force applied at marker j to the velocity induced at marker i. Our approach to obtaining an efficient algorithm for the constrained fluid-solid system is to develop a method for approximating the marker mobility matrix $\mathcal{M}$ in a simple and efficient way that leads to robust preconditioners for solving the mobility subproblem (9); see Section IV.

Observe that the conditioning of the saddle-point system (3) is controlled by the conditioning of $\mathcal{M}$. In particular, if the (nonnegative) eigenvalues of $\mathcal{M}$ are bounded away from zero, then there will be a unique solution to the saddle-point system. If this bound is uniform as the grid is refined, then the problem is well-posed and will satisfy a stability criterion similar to the well-known Ladyzhenskaya–Babuska–Brezzi (LBB) condition for the Stokes saddle-point problem (8). We investigate the spectrum of the marker mobility matrix numerically in Section V. In practice, there may be some nearly zero eigenvalues of the matrix $\mathcal{M}$ corresponding to physical (rather than numerical) null modes. An example is a sphere discretized

with markers on the surface: we know that a uniform compression of the sphere will not cause any effect because of the incompressibility of the fluid filling the sphere. This compression mode corresponds to a null-vector for the constraint forces $\mathbf{\Lambda}$; it poses no difficulties in principle because the right-hand side in (9) is always in the range of $\mathcal{M}$. Of course, when a discrete set of markers is placed on the sphere, the rotational symmetry will be broken and the corresponding mode will have a small but nonzero eigenvalue, which can lead to numerical difficulties if not handled with care.

D. *Periodic steady Stokes flow.* In the time-dependent context, β is finite, and it is easy to see that $\mathcal{A} \succ \mathbf{0}$ is invertible. The same happens even for steady Stokes flow if at least one of the boundaries is a no-slip boundary. In the case of periodic steady Stokes flow, however, $\mathcal{A} = -\eta h^{-2} \mathcal{L}_v$ has in its range vectors that sum to zero, because no nonzero total force can be applied on a periodic domain. This means that a solvability condition is

$$\langle \mathcal{S}\mathbf{\Lambda} + \mathbf{g} \rangle = \mathrm{vol}^{-1} \sum_{i=1}^{N} \mathbf{\Lambda}_i + \langle \mathbf{g} \rangle = \mathrm{vol}^{-1} \mathbf{1}^T \mathbf{\Lambda} + \langle \mathbf{g} \rangle = 0,$$

where $\langle \; \rangle$ denotes an average over the whole system, $\mathbf{1}$ is a vector of ones, and vol is the volume of the domain. This is an additional constraint that must be added to the constrained Stokes system (3) for a periodic domain the steady Stokes case. In this approach, the solution has an indeterminate mean velocity $\langle \mathbf{v} \rangle$ because momentum is not conserved. This sort of approach is followed for a scalar (reaction-diffusion) equivalent of (3) in [62, Appendix], for the traditional Peskin IB method in [77], and for a higher-order IB method in [70].

Here, we instead impose the mean velocity $\langle \mathbf{v} \rangle = \mathbf{0}$ and ensure that the total force applied to the fluid sums to zero, i.e., we enforce momentum conservation. Specifically, for the special case of periodic steady Stokes, we solve the system

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} & -(\mathcal{S} - \mathrm{vol}^{-1} \mathbf{1}^T) \\ -\mathcal{D} & \mathbf{0} & \mathbf{0} \\ -\mathcal{J} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \pi \\ \mathbf{\Lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{g} \\ h \\ W \end{bmatrix}, \tag{11}$$

together with the constraint $\langle \mathbf{v} \rangle = \mathbf{0}$, where we assume that $\langle \mathbf{g} \rangle = \mathbf{0}$ for consistency. This change amounts to simply redefining the spreading operator to subtract the total applied force on the markers as a uniform force density, $\mathcal{S} \leftarrow \mathcal{S} - \mathrm{vol}^{-1} \mathbf{1}^T$. This can be justified by considering the unit cell to be part of an infinite periodic system in which there is an externally applied constant pressure gradient, which is balanced by the drag forces on the bodies so as to ensure that the domain as a whole is in force balance [45; 52; 54].

IV. **Approximating the mobility matrix**

A key element in the preconditioned Krylov solver for (3) that we describe in Section V is an approximate solver for the mobility subproblem (9). The success of this approximate solver, i.e., the accuracy with which we can approximate the Schur complement of the saddle-point problem (3), is crucial to an effective linear solver and one of the key contributions of this work.

Because it involves the inverse Stokes operator $\mathcal{L}^{-1}$, the actual Schur complement $\mathcal{M} = \mathcal{S}^{\star}\mathcal{L}^{-1}\mathcal{S}$ cannot be formed efficiently. Instead of forming the true mobility matrix, we instead approximate $\mathcal{M} \approx \widetilde{\mathcal{M}}$ by a *dense* but low-rank *approximate mobility matrix* $\widetilde{\mathcal{M}}$ given by simple *analytical* approximations. To achieve this, we use two key ideas:

(1) We ignore the specifics of the boundary conditions and assume that the structure is immersed in an infinite domain at rest at infinity (in three dimensions) or in a finite periodic domain (in two dimensions). This implies that the Krylov solver for Equation (3) must handle the boundary conditions.

(2) We assume that the IB spatial discretization is translationally and rotationally invariant; that is, $\mathcal{M}$ does not depend on the exact position and orientation of the body relative to the underlying fluid grid. This implies that the Krylov solver must handle any grid-dependence in the solution.

The first idea, to ignore the boundary conditions in the preconditioner, has worked well in the context of solving the Stokes system (8). Namely, a simple but effective approximation of the inverse of the Schur complement for (8), $(\mathcal{D}\mathcal{A}^{-1}\mathcal{G})^{-1}$, can be constructed by assuming that the domain is periodic so that the finite difference operators commute, and thus the Schur complement degenerates to a diagonal or nearly diagonal mass matrix [29; 34; 18]. The second idea, to make use of the near grid invariance of Peskin's regularized kernel functions, has previously been used successfully in implicit immersed-boundary methods by Ceniceros et al. [19]. Note that for certain choices of the kernel function, the assumption of grid invariance can be a very good approximation to reality; here, we rely on the recently developed six-point kernel [10], which has excellent grid invariance and relatively compact support.

In the remainder of this section, we explain how we compute the entries in $\widetilde{\mathcal{M}}$ in three dimensions, assuming an unbounded fluid at rest at infinity. The details for two dimensions are given in Appendix B and are similar in nature, except for complications for two-dimensional steady Stokes flow resulting from the well-known Stokes paradox.

The mobility matrix $\mathcal{M}$ is a symmetric block matrix built from $N \times N$ blocks of size $d \times d$. The block $\mathcal{M}_{ij}$ corresponding to markers i and j relates a force applied at marker j to the velocity induced at marker i. Our basic assumption is that $\mathcal{M}_{ij}$

does not depend on the actual position of the markers relative to the fluid grid, but rather only depends on the distance between the two markers and on the viscous CFL number β in the form

$$\widetilde{\mathcal{M}}_{ij} = f_\beta(r_{ij})\mathcal{I} + g_\beta(r_{ij})\hat{\boldsymbol{r}}_{ij} \otimes \hat{\boldsymbol{r}}_{ij}, \tag{12}$$

where $\boldsymbol{r}_{ij} = \boldsymbol{R}_i - \boldsymbol{R}_j$ and r_{ij} is the distance between the two markers, and hat denotes a unit vector. The functions of distance $f_\beta(r)$ and $g_\beta(r)$ depend on the specific kernel chosen, the specific discretization of the fluid equations (in our case the staggered-grid scheme), and the viscous CFL number β. To obtain a specific form for these two functions, we empirically fit numerical data with functions with the proper asymptotic behavior at short and large distances between the markers. For this purpose, we first discuss the asymptotic properties of $f_\beta(r)$ and $g_\beta(r)$ from a physical perspective.

It is important to note that the true mobility matrix $\mathcal{M}$ is guaranteed to be SPD because of its structure and the adjointness of the spreading and interpolation operators. This can be ensured for the approximation $\widetilde{\mathcal{M}}$ by placing positivity constraints on suitable linear combinations of the Fourier transforms of $f_\beta(r)$ and $g_\beta(r)$, which ensure that the kernel $\underline{\mathcal{M}}(\boldsymbol{r}_i, \boldsymbol{r}_j)$ given by (12) is SPD in the sense of integral operators. It is, however, very difficult to place such constraints on empirical fits in practice, and in this work, we do not attempt to ensure $\widetilde{\mathcal{M}}$ is SPD for all marker configurations.

A. *Physical constraints*. Let us temporarily focus on the semicontinuum formulation (1) and ignore Eulerian discretization artifacts. The pairwise mobility between markers i and j for a continuum fluid is

$$\mathcal{M}_{ij} = \eta^{-1} \int \delta_a(\boldsymbol{R}_i - \boldsymbol{r}'')\mathbb{G}(\boldsymbol{r}'', \boldsymbol{r}')\delta_a(\boldsymbol{R}_j - \boldsymbol{r}') \, d\boldsymbol{r}'' d\boldsymbol{r}', \tag{13}$$

where $\mathbb{G}(\boldsymbol{r}, \boldsymbol{r}')$ is the Green's function for the fluid equation, i.e., $\boldsymbol{v}(\boldsymbol{r}) = (\mathcal{L}^{-1}\boldsymbol{f})(\boldsymbol{r})$ $= \int \mathbb{G}(\boldsymbol{r}, \boldsymbol{r}')\boldsymbol{f}(\boldsymbol{r}')d\boldsymbol{r}'$, where

$$\frac{\rho}{\Delta t}\boldsymbol{v} + \nabla\pi - \eta\nabla^2\boldsymbol{v} = \boldsymbol{f} \quad \text{and} \quad \nabla \cdot \boldsymbol{v} = 0. \tag{14}$$

It is well-known that $\mathbb{G}$ has the same form as (12),

$$\mathbb{G}(\boldsymbol{R}_1, \boldsymbol{R}_2) = f\,(r_{12})\,\mathcal{I} + g\,(r_{12})\,\hat{\boldsymbol{r}}_{12} \otimes \hat{\boldsymbol{r}}_{12}.$$

For steady Stokes flow ($\beta \to \infty$), $\mathbb{G} \equiv \mathcal{O}$ is the well-known Oseen tensor or Stokeslet,[4] and corresponds to $f_S(r) = g_S(r) \approx (8\pi\eta r)^{-1}$. For inviscid flow, $\beta = 0$, and we have that $\mathcal{A} = (\rho/\Delta t)\mathcal{I}$ and (7) applies, and therefore $\mathcal{L}^{-1} = (\Delta t/\rho)\mathcal{P}$ is

---

[4]The regularized Stokeslet of Cortez [23] is similar to (13) but contains only one regularized delta function in the integrand; this makes the resulting mobility matrix asymmetric, which is unphysical.

a multiple of the projection operator. For finite nonzero values of β, we can obtain $\mathbb{G}$ from the solution of the screened Stokes (i.e., Brinkman) equations (14) [17; 28; 22], and corresponds to the "Brinkmanlet" [28; 22]

$$f_B(r) = \frac{e^{-\alpha r}}{4\pi \eta r}\left(\left(\frac{1}{\alpha r}\right)^2 + \frac{1}{\alpha r} + 1\right) - \frac{1}{4\pi \eta \alpha^2 r^3},$$

$$g_B(r) = -\frac{e^{-\alpha r}}{4\pi \eta r}\left(3\left(\frac{1}{\alpha r}\right)^2 + \frac{3}{\alpha r} + 1\right) + \frac{3}{4\pi \eta \alpha^2 r^3}, \tag{15}$$

where $\alpha^2 = \rho/(\eta \Delta t) = (\beta h^2)^{-1}$. Note that in the steady Stokes limit, $\alpha \to 0$ and the Brinkmanlet becomes the Stokeslet.

We can use (15) to construct $\widetilde{\mathcal{M}}_{ij}$ when the markers are far apart. Namely, if $r_{ij} \gg h$, then we may approximate the IB kernel function by a true delta function, and thus $f_\beta(r)$ and $g_\beta(r)$ are well-approximated by (15). For steady Stokes flow, the interaction between markers decays like r^{-1}. For finite β, however, the viscous contribution decays exponentially fast as $\exp(-r/(h\sqrt{\beta}))$, which is consistent with the fact that markers interact via viscous forces only if they are at a distance not much larger than $h\sqrt{\beta} = \sqrt{\nu \Delta t}$, the typical distance that momentum diffuses during a time step. For nonzero Reynolds numbers, the leading order asymptotic r^{-3} decay of $f_\beta(r)$ and $g_\beta(r)$ is given by the last terms on the right hand side of (15) and corresponds to the electric field of an electric dipole; its physical origin is in the incompressibility constraint, which instantaneously propagates hydrodynamic information between the markers.[5]

For steady Stokes flow, we can say even more about the approximate form of $f_\beta(r)$ and $g_\beta(r)$. As discussed in more detail by Delong et al. [27], for distances between the markers that are not too small compared to the regularization length a, we can approximate (13) with (12) using the well-known Rotne–Prager–Yamakawa (RPY) [68; 57; 80] tensor for the functions $f_\beta(r)$ and $g_\beta(r)$,

$$f_{RPY}(r) = \frac{1}{6\pi \eta a}\begin{cases}\frac{3a}{4r} + \frac{a^3}{2r^3}, & r > 2a, \\[2mm] 1 - \frac{9r}{32a}, & r \le 2a,\end{cases}$$

$$g_{RPY}(r) = \frac{1}{6\pi \eta a}\begin{cases}\frac{3a}{4r} - \frac{3a^3}{2r^3}, & r > 2a, \\[2mm] \frac{3r}{32a}, & r \le 2a,\end{cases} \tag{16}$$

where a is the effective *hydrodynamic radius* of the specific kernel δ_a, defined by $(6\pi a)^{-1} = \int \delta_a(\mathbf{r}'') \mathcal{O}(\mathbf{r}'', \mathbf{r}') \delta_a(\mathbf{r}')\, d\mathbf{r}'' d\mathbf{r}'$. Note that for $r \gg a$ the RPY tensor

---

[5]In reality, of course, this information is propagated via fast sound waves and not instantaneously.
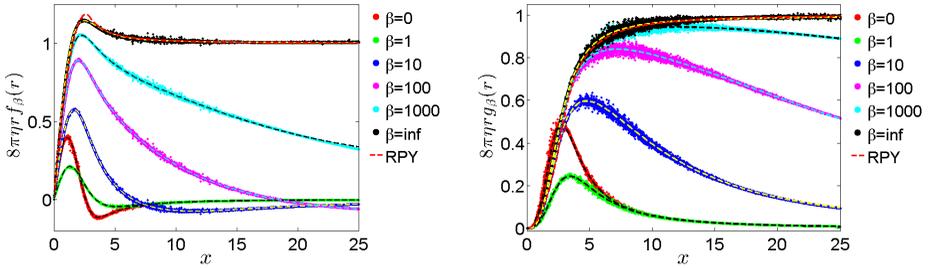
Figure 2. Normalized mobility functions $\tilde{f}(x)$ (left) and $\tilde{g}(x)$ (right) defined similarly to (A.1) as a function of marker-marker distance $x = r/h$, in three dimensions for the six-point kernel of Bao et al. [10], over a range of viscous CFL numbers (different colors, see legend). Numerical data is shown with symbols and obtained using a 256^3 periodic fluid grid, while dashed lines show our empirical fit of the form (A.2) for steady Stokes ($\beta \to \infty$) and (A.6) for finite β. For steady Stokes flow, the numerical data is in reasonable agreement with the RPY tensor (16) (dashed red line).

approaches the Oseen tensor and decays like r^{-1}. A key advantage of the RPY tensor is that it *guarantees* that the mobility matrix (12) is SPD for *all* configurations of the markers, which is a rather nontrivial requirement [80]. The actual discrete pairwise mobility $\mathcal{M}_{ij}$ obtained from the spatially discrete IB method is well-described by the RPY tensor [27] (see Figure 2). The only fitting parameter in the RPY approximation is the effective hydrodynamic radius a averaged over many positions of the marker relative to the underlying grid [9; 27]; for the six-point kernel used here,[6] $a = 1.47\,h$. For the Brinkman equation, the equivalent of the RPY tensor can be computed for $r \geq 2a$ by applying a Faxen-like operator from the left and right on the Brinkmanlet (see [28, Equation (26)]); the resulting analytical expressions are complex and are not used in our empirical fitting.

B. *Empirical fits.* In this work, we use empirical fits to approximate the mobility. This is because the analytical approximations, such as those offered by the RPY tensor, are most appropriate for unbounded domains and assume the markers are far apart compared to the width of the regularized delta function. In numerical computations, we use a finite periodic domain, and this requires corrections to the analytic expressions that are difficult to model. For example, for finite β, we find that the periodic corrections to the inviscid (dipole) r^{-3} contribution dominate over the exponentially decaying viscous contribution, which makes the precise form of the viscous terms in (15) irrelevant in practice. For $r \gg h$, only the asymptotically dominant far-field terms survive, and we make an effort to preserve those in our fitting because the numerical results are obtained using finite systems and thus not

---

[6]As summarized in [9; 27], $a \approx 1.25h$ for the widely used four-point kernel [64], and $a \approx 0.91h$ for the three-point kernel [67].

reliable at large marker distances. At shorter distances, however, the discrete nature of the fluid solver and the IB kernel functions becomes important, and empirical fitting seems to be a simple yet flexible alternative to analytical computations. At the same time, we feel that is important to *constrain* the empirical fits based on known behavior at short and large distances.

Firstly, for $r \ll h$, the pairwise mobility can be well-approximated by the self-mobility ($r = 0$, corresponding to the diagonal elements $\widetilde{\mathcal{M}}_{ii}$), for which we know the following facts:

- For the steady Stokes regime ($\beta \to \infty$), the diagonal elements are given by Stokes's drag formula, yielding

$$f_\infty(0) = (6\pi \eta a)^{-1} \sim 1/\eta h \quad \text{and} \quad g_\infty(0) = 0,$$

  where we recall that a is the effective hydrodynamic radius of a marker for the particular spatial discretization (kernel and fluid solver).

- For the inviscid case ($\beta = 0$), it is not hard to show (see [9]) that

$$f_0(0) = \frac{d-1}{d} \frac{\Delta t}{\rho} V_m^{-1} \sim \beta/\eta h \quad \text{and} \quad g_0(0) = 0, \tag{17}$$

  where $d = 3$ is the dimensionality, and $V_m = c_V h^3$ is the "volume" of the marker, where the constant c_V is straightforward to calculate.

- The above indicates that $f_\beta(0)$ goes from $\sim \beta/(\eta h)$ for small β to $\sim 1/(\eta h)$ for large β. At intermediate viscous CFL numbers β, we can set

$$f_\beta(0) = \frac{C(\beta)}{\eta h} \quad \text{and} \quad g_\beta(0) = 0, \tag{18}$$

  where $C(\beta \ll 1) \approx 2\beta/(3c_V)$ is linear for small β and then becomes $O(1)$ for large β. We will obtain the actual form of $C(\beta)$ from empirical fitting.

Secondly, for $r \gg h$, we know the asymptotic decay of the hydrodynamic interactions from (15):

- For the steady Stokes regime ($\beta \to \infty$), we have the Oseen tensor given by

$$f_\infty(r \gg h) \approx g_\infty(r \gg h) \approx (8\pi \eta r)^{-1}. \tag{19}$$

- For the inviscid case ($\beta = 0$), we get the electric field of an electric dipole,

$$f_0(r \gg h) \approx -\frac{\Delta t}{4\pi \rho r^3} \quad \text{and} \quad g_0(r \gg h) \approx \frac{3\Delta t}{4\pi \rho r^3}, \tag{20}$$

  which is also the asymptotic decay for $\beta > 0$ for $r \gg h\sqrt{\beta}$.

We obtain the actual form of the functions $f_\beta(r)$ and $g_\beta(r)$ empirically by fitting

numerical data for the parallel and perpendicular mobilities

$$\mu_{ij}^{\parallel} = \hat{\boldsymbol{r}}_{ij}^{T}\widetilde{\mathcal{M}}_{ij}\hat{\boldsymbol{r}}_{ij} \approx f_{\beta}(r_{ij}) + g_{\beta}(r_{ij}) \quad \text{and} \quad \mu_{ij}^{\perp} = (\hat{\boldsymbol{r}}_{ij}^{\perp})^{T}\widetilde{\mathcal{M}}_{ij}\hat{\boldsymbol{r}}_{ij}^{\perp} \approx f_{\beta}(r_{ij}),$$

where $\hat{\boldsymbol{r}}_{ij}^{\perp} \cdot \hat{\boldsymbol{r}}_{ij} = 0$. To do so, we placed a large number of markers N in a cube of length $l/8$ inside a periodic domain of length l. For each marker i, we applied a unit force $\boldsymbol{\Lambda}_i$ with random direction while leaving $\boldsymbol{\Lambda}_j = 0$ for $j \neq i$, solved (8), and then interpolated the fluid velocity $\boldsymbol{v}$ at the position of each of the markers. The resulting parallel and perpendicular relative velocity for each of the $N(N-1)/2$ pairs of particles allows us to estimate $f_{\beta}(r_{ij})$ and $g_{\beta}(r_{ij})$. By making the number of markers N sufficiently large, we sample the mobility over essentially all relative positions of the pair of markers. For the self-mobility $\widetilde{\mathcal{M}}_{ii}$ ($r_{ii} = 0$), we take $g_{\beta}(0) = 0$ and compute $f_{\beta}(0)$ from the numerical data.

If the spatial discretization were perfectly translationally and rotationally invariant and the domain were infinite, all of the numerical data points for $f_{\beta}(r)$ and $g_{\beta}(r)$ would lie on a smooth curve and would not depend on the actual position of the pair of markers relative to the underlying grid. In reality, it is not possible to achieve perfect translational invariance with a kernel of finite support [64], and so we expect some (hopefully small) scatter of the points around a smooth fit. Normalized numerical data for $f_{\beta}(r)$ and $g_{\beta}(r)$ are shown in Figure 2, and we indeed see that the data can be fit well by smooth functions over the whole range of distances. To maximize the quality of the fit, we perform separate fits for $\beta \to \infty$ (steady Stokes flow) and finite β. We also make an effort to make the fits change smoothly as β grows towards infinity, as we explain in more detail in Appendix A. Code to evaluate the empirical fits described in Appendices A and B is publicly available to others for a number of kernels constructed by Peskin and coworkers (three-, four-, and six-point) in both two and three dimensions at cims.nyu.edu/~donev/src/MobilityFunctions.c.

V. Linear solver

To solve the constrained Stokes problem (3), we use the preconditioned flexible GMRES (FGMRES) method, which is a Krylov solver. We will refer to this as the "outer" Krylov solver, as it must be distinguished from "inner" Krylov solvers used in the preconditioner. Because we use Krylov solvers in our preconditioner and because Krylov solvers generally cannot be expressed as linear operators, it is crucial to use a flexible Krylov method such as FGMRES for the outer solver. The overall method is implemented in the open-source immersed-boundary adaptive mesh refinement (IBAMR) software infrastructure [37]; in this work we focus on uniform grids and do not use the AMR capabilities of IBAMR (but see [13; 14]). IBAMR uses Krylov solvers that are provided by the PETSc library [7].

A. *Preconditioner for the constrained Stokes system.* In the preconditioner used by the outer Krylov solver, we want to *approximately* solve the nested saddle-point linear system

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} & -\mathcal{S} \\ -\mathcal{D} & \mathbf{0} & 0 \\ -\mathcal{J} & \mathbf{0} & 0 \end{bmatrix} \begin{bmatrix} v \\ \pi \\ \Lambda \end{bmatrix} = \begin{bmatrix} cg \\ h \\ W \end{bmatrix},$$

where we recall that $\mathcal{A} = (\rho/\Delta t)\,\mathcal{I} - \eta h^{-2}\mathcal{L}_v$. Let us set $\alpha = 1$ if $\mathcal{A}$ has a null-space, (e.g., for a fully periodic domain for steady Stokes flow) and we set $\alpha = 0$ if $\mathcal{A}$ is invertible. When $\alpha = 1$, let us define the restricted inverse $\mathcal{A}^{-1}$ to only act on vectors of mean value zero, and to return a vector of mean zero.

Applying our Schur complement based preconditioner for solving (3) consists of the following steps:

(1) Solve the (unconstrained) fluid subproblem,

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} \\ -\mathcal{D} & \mathbf{0} \end{bmatrix} \begin{bmatrix} v \\ \pi \end{bmatrix} = \begin{bmatrix} g \\ h \end{bmatrix}.$$

   To control the accuracy of the solution one can either use a relative tolerance based stopping criterion or fix the number of iterations N_s in the inner solver.

(2) Calculate the slip velocity on the set of markers, $\Delta V = -(\mathcal{J}v + W)$.

(3) Approximately solve the Schur complement system,

$$\widetilde{\mathcal{M}}\Lambda = \Delta V, \tag{21}$$

   where the mobility approximation $\widetilde{\mathcal{M}}$ is constructed as described in Section IV.

(4) Optionally, re-solve the corrected fluid subproblem,

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} \\ -\mathcal{D} & \mathbf{0} \end{bmatrix} \begin{bmatrix} v \\ \pi \end{bmatrix} = \begin{bmatrix} g + \mathcal{S}\Lambda - \alpha\,\mathrm{vol}^{-1}\,\mathbf{1}^T\Lambda \\ h \end{bmatrix}.$$

All linear solvers used in the preconditioner can be approximate, and this is in fact the key to the efficiency of the overall solver approach. Notably, the inner Krylov solvers used to solve the unconstrained Stokes subproblems in steps 1 and 4 above can be done by using a small number N_s of iterations using a method briefly described in the next section. If the fluid subproblem is approximately solved in both steps 1 and 4, which we term the *full* Schur complement preconditioner, each application of the preconditioner requires $2N_s$ applications of the Stokes preconditioner (22). It is also possible to omit step 4 above to obtain a block *lower triangular* Schur preconditioner [30], which requires only N_s applications of the unconstrained Stokes preconditioner (22). We will numerically compare these two preconditioners and study the effect of N_s on the convergence of the FGMRES outer solver in Section A.

B. *Unconstrained fluid solver.* A key component we rely on is an approximate solver for the unconstrained Stokes subproblem,

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} \\ -\mathcal{D} & 0 \end{bmatrix} \begin{bmatrix} v \\ \pi \end{bmatrix} = \begin{bmatrix} g \\ h \end{bmatrix},$$

for which a number of techniques have been developed in the finite-element context [30]. To solve this system, we use GMRES with a preconditioner $\mathcal{P}_S^{-1}$ based on the projection method, as proposed by Griffith [34] and improved to some extent by Cai et al. [18]. Specifically, the preconditioner for the Stokes system that we use in this work is

$$\mathcal{P}_S^{-1} = \begin{pmatrix} \mathcal{I} & h^2 \mathcal{G} \widetilde{\mathcal{L}}_p^{-1} \\ 0 & \widetilde{\mathcal{B}}^{-1} \end{pmatrix} \begin{pmatrix} \mathcal{I} & 0 \\ -\mathcal{D} & -\mathcal{I} \end{pmatrix} \begin{pmatrix} \widetilde{\mathcal{A}}^{-1} & 0 \\ 0 & \mathcal{I} \end{pmatrix}, \tag{22}$$

where $\mathcal{L}_p = h^2(\mathcal{D}\mathcal{G})$ is the dimensionless pressure (scalar) Laplacian, and $\widetilde{\mathcal{A}}^{-1}$ and $\widetilde{\mathcal{L}}_p^{-1}$ denote approximate solvers obtained by a *single* V-cycle of a geometric multigrid solver for the vector Helmholtz and scalar Poisson problems, respectively. In the time-dependent case, the approximate Schur complement for the unconstrained Stokes subproblem is

$$\widetilde{\mathcal{B}}^{-1} = -\frac{\rho h^2}{\Delta t} \widetilde{\mathcal{L}}_p^{-1} + \eta \mathcal{I},$$

and for steady Stokes flow, $\widetilde{\mathcal{B}}^{-1} = \eta \mathcal{I}$. Further discussion of the relation of these preconditioners to the those described in the book [30] can be found in [34].

Observe that one application of $\mathcal{P}_S^{-1}$ is relatively inexpensive and involves only a few scalar multigrid V-cycles. Indeed, solving the Stokes system using GMRES with this preconditioner is only a few times more expensive than solving a scalar Poisson problem, even in the steady Stokes regime [18]. Note that it is possible to omit the upper right off-diagonal block in the first matrix on the right hand side of (22) to obtain a block lower triangular preconditioner that is also effective, and may in fact be preferred at zero Reynolds number since it allows one to skip a sweep of the pressure multigrid solver [18]. We empirically find that including the Poisson solve (velocity projection) improves the overall performance of the outer solver.

C. *Mobility solver.* From a computational perspective, one of the most challenging steps in our preconditioner is solving the mobility subproblem (21). Since this is done inside a preconditioner, and because $\widetilde{\mathcal{M}}$ is itself an approximation of the true mobility matrix $\mathcal{M}$, it is not necessary to solve (21) exactly. In the majority of the examples presented herein, we solve (21) using direct solvers provided by LAPACK. This is feasible on present hardware for up to around 10^5 markers and allows us to focus on the design of the approximation $\widetilde{\mathcal{M}}$ and to study the accuracy of the overall method.

Let us denote with s the smallest marker-marker spacing. For well-spaced markers, $s/h \gtrsim 2$, our approximate mobility $\widetilde{\mathcal{M}}$ is typically SPD even for large numbers of markers, and in these cases, we can use the Cholesky factorization to solve (21). In some cases, however, there may be a few small or even negative eigenvalues of $\widetilde{\mathcal{M}}$ that have to be handled with care. We have found that the most robust (albeit expensive) alternative is to perform an SVD of $\widetilde{\mathcal{M}}$, and to use a pseudoinverse of $\widetilde{\mathcal{M}}$ (keeping only eigenvalues larger than some tolerance $\epsilon_{SVD} > 0$) to solve (21). This effectively filters out the spuriously small or negative eigenvalues. The factorization of $\widetilde{\mathcal{M}}$ needs to be performed only once per constrained Stokes solve since the body is kept fixed during a time step. In cases where there is a single body, the factorization needs to be performed only once per simulation and can be reused; if the body is translating or rotating, one ought to perform appropriate rotations of the right hand side and solution of (21). In some cases of practical interest where the number of markers is not too large, it is possible to precompute the true mobility $\mathcal{M}_0$ with periodic boundary conditions (for a large enough domain) and to store its factorization. Even if the structure moves relative to the underlying grid, such a precomputed (reference) mobility $\mathcal{M}_0$ is typically a much better approximation to the true mobility than our empirical approximation $\widetilde{\mathcal{M}}$, and can effectively be used in the preconditioner. Determining effective approaches to solving the mobility subproblem in the presence of multiple moving rigid bodies remains future work, as discussed further in the Conclusions.

VI. Conditioning of the mobility matrix

The conditioning of the constrained Stokes problem (3) is directly related to the conditioning of the Schur complement mobility matrix $\mathcal{M} = \mathcal{J}\mathcal{L}^{-1}\mathcal{S}$, which is intimately connected to the relation between the fluid solver grid spacing h and the smallest intermarker spacing s. Firstly, it is obvious that if two markers i and j are very close to each other, then the fluid solver cannot really distinguish between Λ_i and Λ_j and will instead effectively see only their sum. We also know that using too many markers for a fixed fluid grid will ultimately lead to a rank-deficient $\mathcal{M}$, because it is not possible to constrain a finite-dimensional discrete fluid velocity at too many points. This physical intuition tells us that the condition number of $\mathcal{M}$ should increase as the marker spacing becomes small compared to the grid spacing. This well-known intuition, however, does not tell us how closely the markers can or must be placed in practice. Standard wisdom for the immersed boundary method, which is based on the behavior of models of elastic bodies, is to make the marker spacing on the order of half a grid spacing. As we show, this leads to extremely ill-conditioned mobility matrices for rigid bodies. The specific results depend on the dimensionality, the details of the fluid solver, and the specific kernel used; however, the qualitative features we report appear to be rather general.
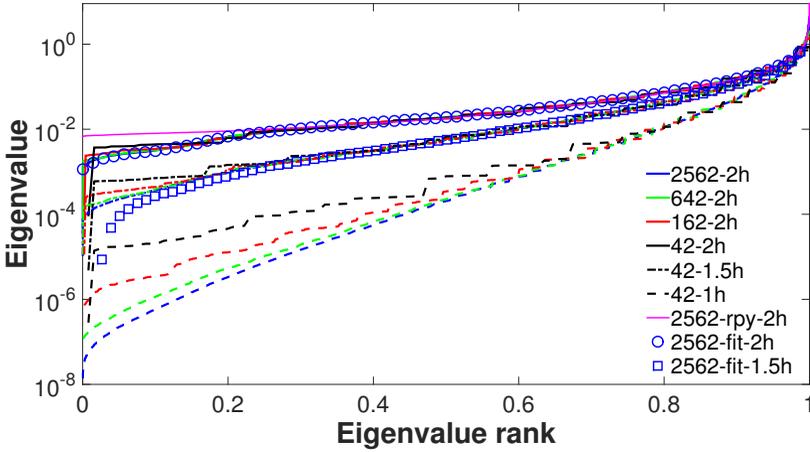
Figure 3. Eigenvalue spectrum of the mobility matrix for steady Stokes flow around a spherical shell covered with different numbers of markers (42, 162, 642, or 2562, see legend) embedded in a periodic domain. Solid lines are for marker spacing of $s \approx 2h$, dashed-dotted lines for spacing of $s \approx 1.5\,h$, and dashed lines for spacing of $s \approx 1h$. The marker spacing is $s \approx 1$ in all cases; for $s \approx 2h$, the fluid grid size is 128^3 for 2562 markers and 64^3 for smaller number of markers, and scaled accordingly for other spacings. For comparison, we show the spectrum of $\widetilde{\mathcal{M}}_{RPY}$ for the most resolved model ($N = 2562$ markers) at $s/h \approx 2$. Also shown is the spectrum of the empirical (fit) approximation to the mobility $\widetilde{\mathcal{M}}$ for the two larger spacings; for $s \approx h$ our empirical approximation is very poor and includes many spurious negative eigenvalues (not shown).

To determine the condition number of the mobility matrix, we consider "open" and "filled" sphere models. We discretize the surface of a sphere as a shell of markers constructed by a recursive procedure suggested to us by Charles Peskin (private communication). We start with 12 markers placed at the vertices of an icosahedron, which gives a uniform triangulation of a sphere by 20 triangular faces. Then, we place a new marker at the center of each edge and recursively subdivide each triangle into four smaller triangles, projecting the vertices back to the surface of the sphere along the way. Each subdivision approximately quadruples the number of vertices, with the k-th subdivision producing a model with $10 \cdot 4^{k-1} + 2$ markers. To create filled sphere models, we place additional markers at the vertices of a tetrahedral grid filling the sphere that is constructed using the TetGen library, starting from the surface triangulation described above. The constructed tetrahedral grids are close to uniform, but it is not possible to control the precise marker distances in the resulting irregular grid of markers. We use models with approximately equal edges (distances between nearest-neighbor markers) of length $\approx s$, which we take as a measure of the typical marker spacing. We numerically computed the mobility matrix $\mathcal{M}$ for an isolated spherical shell in a large periodic domain

for various numbers of markers N. Here we keep the ratio s/h fixed and keep the marker spacing fixed at $s \approx 1$; one can alternatively keep the radius of the sphere fixed.[7] In Figure 3, we show the spectrum of $\mathcal{M}$ for varying levels of resolution for three different spacings of the markers, $s/h = 1$, $s/h = 3/2$, and $s/h = 2$. Similar spectra, but with somewhat improved condition number (i.e., fewer smaller eigenvalues), are seen for nonzero Reynolds numbers (finite β).

The results in Figure 3 strongly suggest that as the number of markers increases, the low-lying (small eigenvalue) spectrum of the mobility matrix approaches a limiting shape. Therefore, the nontrivial eigenvalues remain bounded away from zero even as the resolution is increased, which implies that for $s/h \gtrsim 1$ the system (3) is uniformly solvable or "stable" under grid refinement. Note that in the case of a sphere, there is a trivial zero eigenvalue in the continuum limit, which corresponds to uniform compression of the sphere; this is reflected in the existence of one eigenvalue much smaller than the rest in the discrete models. Ignoring the trivial eigenvalue, the condition number of $\mathcal{M}$ is $O(N)$ for this example because the largest eigenvalue in this case increases like the number of markers N, in agreement with the fact that the Stokes drag on a sphere scales linearly with its radius. This is as close to optimal as possible, because for the continuum equations for Stokes flow around a sphere, the eigenvalues corresponding to spherical harmonic modes scale like the index of the spherical harmonic. However, what we are concerned here is not so much how the condition number scales with N, but with the size of the prefactor, which is determined by the smallest nontrivial eigenvalues of $\mathcal{M}$.

Figure 3 clearly shows that the number of very small eigenvalues increases as we bring the markers closer to each other, as expected. The increase in the conditioning number is quite rapid, and the condition number becomes $O(10^6 - 10^7)$ for marker spacings of about one per fluid grid cell. For the conventional choice $s \approx h/2$, the mobility matrix is so poorly conditioned that we cannot solve the constrained Stokes problem in double-precision floating point arithmetic. Of course, if the markers are too far apart then fluid will leak through the wall of the structure. We have performed a number of heuristic studies of leak through flat and curved rigid walls and concluded that $s/h \approx 2$ yields both small leak and a good conditioning of the mobility, at least for the six-point kernel used here [10]. Therefore, unless indicated otherwise, in the remainder of this work, we keep the markers about *two grid cells apart* in both two and three dimensions. It is important to emphasize that this is just a heuristic recommendation and not a precise estimate. We remark that Taira and Colonius, who solve a different Schur complement "modified Poisson

---

[7]The scaling used here, keeping $s = 1$ fixed, is more natural for examining the small eigenvalues of $\mathcal{M}$, which are dominated by discretization effects, as opposed to the large eigenvalues, which correspond to physical modes of the Stokes problem posed on a sphere and are insensitive to the discretization details.

equation", recommend $s/h \approx 1$ to "achieve a reasonable condition number and to prevent penetration of streamlines caused by a lack of Lagrangian points."

It is important to observe that putting the markers further than the traditional wisdom will increase the "leak" between the markers. For rigid structures, the exact positioning of the markers can be controlled since they do not move relative to one another as they do for an elastic bodies; this freedom can be used to reduce penetration of the flow into the body by a careful construction of the marker grid. In Section VIII, we discuss alternatives to the traditional marker-based IB method [38] that can be used to control the conditioning number of the Schur complement and allow for more tightly spaced markers.

It is worthwhile to examine the underlying cause of the ill-conditioning as the markers are brought close together. One source of ill-conditioning comes from the *discrete* (finite-dimensional) nature of the fluid solver, which necessarily limits the rank of the mobility matrix. But another contributor to the worsening of the conditioning is the *regularization* of the delta function. Observe that for a true delta function ($a \to 0$) in Stokes flow, the pairwise mobility is the length-scale-free Oseen tensor $\sim r^{-1}$, and the shape of the spectrum of the mobility matrix has to be *independent* of the spacing among the markers. In the standard immersed boundary method, $a \sim h$, so the fluid grid scale h and the regularization scale a are difficult to distinguish.

To try to separate h from a, we can take a continuum model of the fluid, but keep the discrete marker representation of the body; see (1). In this case the pairwise mobility would be given by (13), which leads to the RPY tensor (16) for a kernel that is a surface delta function over a sphere of radius a (see [80, Equation (4.1)]). In Figure 3 we compare the spectra of the discrete mobility $\mathcal{M}$ with those of the analytical mobility approximation $\widetilde{\mathcal{M}}_{RPY}$ constructed by using (16) for the pairwise mobility. We observe that the two are very similar for $s \approx 2h$, however, for smaller spacings $\widetilde{\mathcal{M}}_{RPY}$ does not have very small eigenvalues and is much better conditioned than $\mathcal{M}$ (data not shown). In Figure 3 we also show the spectrum of our approximate mobility $\widetilde{\mathcal{M}}$ constructed using the empirical fits described in Section IV. The resulting spectra show a worsening conditioning for spacing $s \approx 1.5h$ consistent with the spectrum of $\mathcal{M}$. These observations suggest that both the regularization of the kernel and the discretization artifacts contribute to the ill-conditioning, and suggest that it is worthwhile to explore alternative discrete delta function kernels in the context of rigid-body IB methods.

We also note that we see a severe worsening of the conditioning of $\mathcal{M}$, independent of β, when we switch from a spherical shell to a filled sphere model. Some of this may be due to the fact that the tetrahedral volume mesh used to construct the marker mesh is not as uniform as the surface triangular mesh. We suspect, however, that this ill-conditioning is primarily *physical* rather than numerical, and comes

from the fact that the present marker model cannot properly distinguish between surface tractions and body (volume) stresses. Therefore, $\boldsymbol{\Lambda}$ remains physically ill-defined even if one gets rid of all discretization artifacts.

Lastly, it is important to emphasize that in the presence of ill-conditioning, what matters in practice are not only the smallest eigenvalues but also their associated eigenvectors. Specifically, we expect to see signatures of these eigenvectors (modes) in $\boldsymbol{\Lambda}$, since they will appear with large coefficients in the solution of (9) if the right hand side has a nonzero projection onto the corresponding mode. As expected, the small-eigenvalue eigenvectors of the mobility correspond to high-frequency (in the spatial sense) modes for the forces $\boldsymbol{\Lambda}$. Therefore, if the markers are too closely spaced the solutions for the forces $\boldsymbol{\Lambda}$ will develop unphysical high-frequency oscillations or jitter, even for smooth flows, especially in time-dependent flows, as observed in practice [82]. We have observed that for *smooth* flows (i.e., smooth right hand-side of (9)), the improved translational invariance of the 6-point kernel reduces the magnitude of this jitter compared to the traditional Peskin four-point kernel.

VII. Numerical tests

In this section we apply our rigid-body IB method to a number of benchmark problems. We first present tests of the preconditioned FGMRES solver, and then demonstrate the advantage of our method over splitting-based direct forcing methods. We further consider a simple test problem at zero Reynolds number, involving the flow around a fixed sphere, and study the accuracy of both the fluid (Eulerian) variables $\boldsymbol{v}$ and π, as well as of the body (Lagrangian) surface tractions represented by $\boldsymbol{\Lambda}$, as a function of the grid resolution. We finally study flows around arrays of cylinders in two dimensions and spheres in three dimensions over a range of Reynolds numbers, and compare our results to those obtained by Ladd using the lattice Boltzmann method [54; 52; 44].

A. *Empirical convergence of GMRES.* Here we consider the model problem of flow past a sphere in a cubic domain that is either periodic or with no-slip boundaries. Except for the largest resolutions studied here, the number of markers is relatively small, and dense linear algebra can be used to solve the mobility subproblem (21) robustly and efficiently, so that the cost of the solver is dominated by the fluid solver. We therefore use the number of total applications of the Stokes preconditioner (22) as a proxy for the CPU effort, instead of relying on elapsed time, which is both hardware and software dependent. A key parameter in our preconditioner is the number of iterations N_s used in the iterative unconstrained Stokes solver. We recall that in the full preconditioner, there are two unconstrained inexact Stokes solves per iteration, giving a total of $2N_s$ applications of $\mathcal{P}_S^{-1}$ per outer FGMRES iteration. If
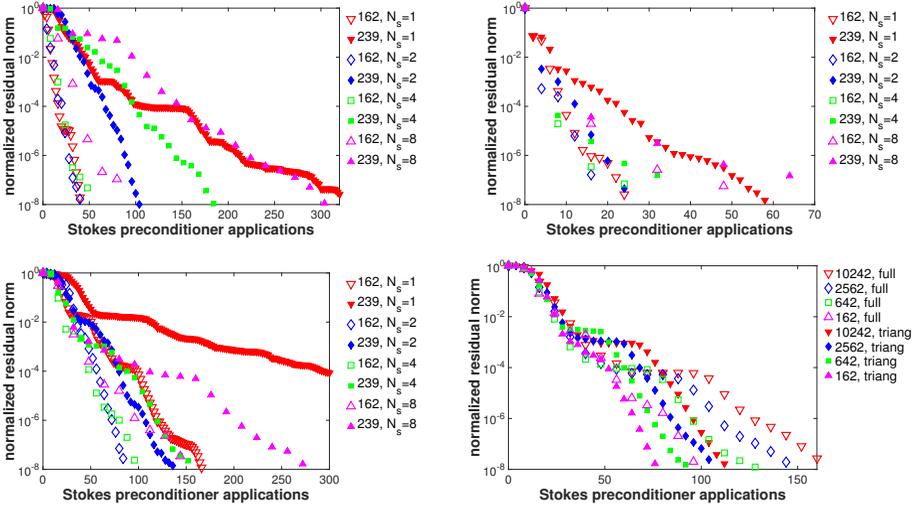
Figure 4. FGMRES convergence for the constrained problem (3) for different numbers of iterations N_s in the unconstrained Stokes solver used in the preconditioner. The specific problem is a rigid sphere of hydrodynamic radius R moving through a stationary domain of length $L \approx 4.35\,R$, with the marker spacing fixed at $s/h \approx 2$ and the GMRES restart frequency set to 100 iterations. Top left panel: steady Stokes ($Re = 0$) flow for an empty 162-marker shell and a filled 239-marker sphere moving through a periodic domain of 32^3 fluid grid cells. Top right panel: same as top left but for $Re \approx 10$. Bottom left panel: as top left panel but now in domain with no slip boundary conditions applied on all sides of the domain. Bottom right panel: a spherical shell moving in a nonperiodic domain (as in bottom left panel) for different resolutions of the shell (162, 642, 2562, and 10242 markers, respectively) and the fluid solver grid (32^3, 64^3, 128^3, and 256^3 grid cells, respectively), fixing $N_s = 4$, for both the full Schur complement preconditioner and the lower triangular approximate Schur complement preconditioner.

the lower triangular preconditioner is used, then the second inexact Stokes solve is omitted, and we perform only N_s applications of $\mathcal{P}_S^{-1}$ per outer FGMRES iteration.

In the first set of experiments, we use the full preconditioner and periodic boundary conditions. We represent the sphere by a spherical shell of markers that is either empty (162 markers) or is filled with additional markers in the interior (239 markers). The top panels of Figure 4 show the relative FGMRES residual as a function of the total number of applications of $\mathcal{P}_S^{-1}$ for several different choices of N_s, for both steady Stokes flow (left panel) and a flow at Reynolds number Re = 10 (right panel). We see that for spherical shells with well-conditioned $\mathcal{M}$ and $\widetilde{\mathcal{M}}$, the exact value of N_s does not have a large effect on solver performance. However, making N_s very large leads to wasted computational effort by "oversolving" the Stokes system. This degrades the overall performance, especially for tight solver tolerance. For the ill-conditioned case of a filled sphere model in steady Stokes

flow, the exact value of N_s strongly affects the performance, and the optimal value is empirically determined to be $N_s = 2$. As expected, the linear system (3) is substantially easier to solve at higher Reynolds numbers, especially for the filled-sphere models.

In the bottom left panel of Figure 4 we show the FGMRES convergence for a nonperiodic system. In this case, we know that the Stokes preconditioner $\mathcal{P}_S^{-1}$ itself does not perform as well as in the periodic case [34; 18], and we expect slower overall convergence. In this case, we see that $N_s = 2$ and $N_s = 4$ are good choices. Investigations (data not shown) show that $N_s = 4$ is more robust for problems with a larger number of markers. Also, note that increasing N_s decreases the total number of FGMRES iterations for a fixed number of applications of $\mathcal{P}_S^{-1}$, and therefore reduces the overall memory usage and the number of times the mobility subproblem (21) needs to be solved; however, note that each of these solves is just a backward/forward substitution if a direct factorization of $\widetilde{\mathcal{M}}$ has been precomputed.

The bottom right panel of Figure 4 shows the FGMRES convergence for a nonperiodic system as the resolution of the grid and the spherical shell is refined in unison, keeping $N_s = 4$. The results in Figure 4 demonstrate that our linear solver is able to cope with the increased number of degrees of freedom under refinement relatively robustly, although a slow increase of the total number of FGMRES iterations is observed. Comparing the full preconditioner with the lower triangular preconditioner, we see that the latter is computationally more efficient overall; this is in agreement with experience for the unconstrained Stokes system [18]. In some sense, what this shows is that it is best to let the FGMRES solver correct the initial unconstrained solution for the velocity and pressure in the *next* FGMRES iteration, rather than to re-solve the fluid problem in the preconditioner itself. However, if very tight solver tolerance is required, we find that it is necessary to perform some corrections of the velocity and pressure inside the preconditioner. In principle, the second unconstrained Stokes solve in the preconditioner can use a different number of iterations N_s' from the first, but we do not explore this option further here. Moreover, if $\widetilde{\mathcal{M}} \approx \mathcal{M}$ (for example, if $\widetilde{\mathcal{M}}$ was computed numerically rather than approximated), then the full Schur complement preconditioner will converge in one or two iterations and there is no advantage to using the lower triangular preconditioner.

B. *Flow through a nozzle.* In this section we demonstrate the strengths of our method on a test problem involving steady-state flow through a nozzle in two dimensions. We compare the steady state flow through the nozzle obtained using our rigid-body IB method to the flow obtained by using a splitting-based direct forcing approach [78; 13]. Specifically, we contrast our monolithic fluid-solid solver to a split solver based on performing the following operations at time step n:Solve

the fluid subproblem as if the body were not present,

$$\begin{bmatrix} \mathcal{A} & \mathcal{G} \\ -\mathcal{D} & 0 \end{bmatrix} \begin{bmatrix} \tilde{\boldsymbol{v}}^{n+1} \\ \tilde{\pi}^{n+\frac{1}{2}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{g}^{n+\frac{1}{2}} \\ 0 \end{bmatrix}.$$

(1) Calculate the slip velocity on the set of markers, $\Delta \boldsymbol{V} = -(\mathcal{J}^{n+\frac{1}{2}} \tilde{\boldsymbol{v}}^{n+1} + \boldsymbol{W}^{n+\frac{1}{2}})$, giving the fluid-solid force estimate $\boldsymbol{\Lambda}^{n+\frac{1}{2}} = (\rho/\Delta t) \Delta \boldsymbol{V}$.

(2) Correct the fluid velocity to approximately enforce the no-slip condition,

$$\boldsymbol{v}^{n+1} = \tilde{\boldsymbol{v}}^{n+1} + S^{n+\frac{1}{2}} \Delta \boldsymbol{V}.$$

Note that in the original method of [13] in the last step the fluid velocity is projected onto the space of divergence-free vector fields by re-solving the fluid problem with the approximation $\mathcal{A} \approx (\rho/\Delta t) \, \boldsymbol{I}$ (i.e., ignoring viscosity). We simplify this step here because we have found the projection to make a small difference in practice for steady state flows, since the same projection is carried out in the subsequent time step.

We discretize a nozzle constriction in a slit channel using IB marker points about 2 grid spacings apart. The geometry of the problem is illustrated in the top panel of Figure 5; parameters are $\rho = 1$, grid spacing $\Delta x = 0.5$, nozzle length $l = 55.5$, nozzle opening width $d \approx 2.9$, and η variable (other parameters are given in the figure caption). No slip boundary conditions are specified on the top and bottom channel walls, and on the side walls the tangential velocity is set to zero and the normal stress is specified to give a desired pressure jump across the channel of $\Delta \pi = 2$. The domain is discretized using a grid of 256×128 cells and the problem evolved for some time until the flow becomes essentially steady. The Reynolds number is estimated based on the maximum velocity through the nozzle opening and the width of the opening.

In the bottom four panels in Figure 5 we compare the flow computed using our method (left panels) to that obtained using the splitting-based direct forcing algorithm summarized above (right panels). Our method is considerably slower (by at least an order of magnitude) for this specific example because the GM-RES convergence is slow for this challenging choice of boundary conditions at small Reynolds numbers in two dimensional (recall that steady Stokes flow in two dimensions has a diverging Green's function). To make the comparison fairer, we use a considerably smaller time step size for the splitting method, so that we approximately matched the total execution time between the two methods. Note that for steady-state problems like this one with fixed boundaries, it is much more efficient to precompute the *actual* mobility matrix (Schur complement) once at the beginning, instead of approximating it with our empirical fits. However, for a more fair and general comparison we instead use our preconditioner to solve
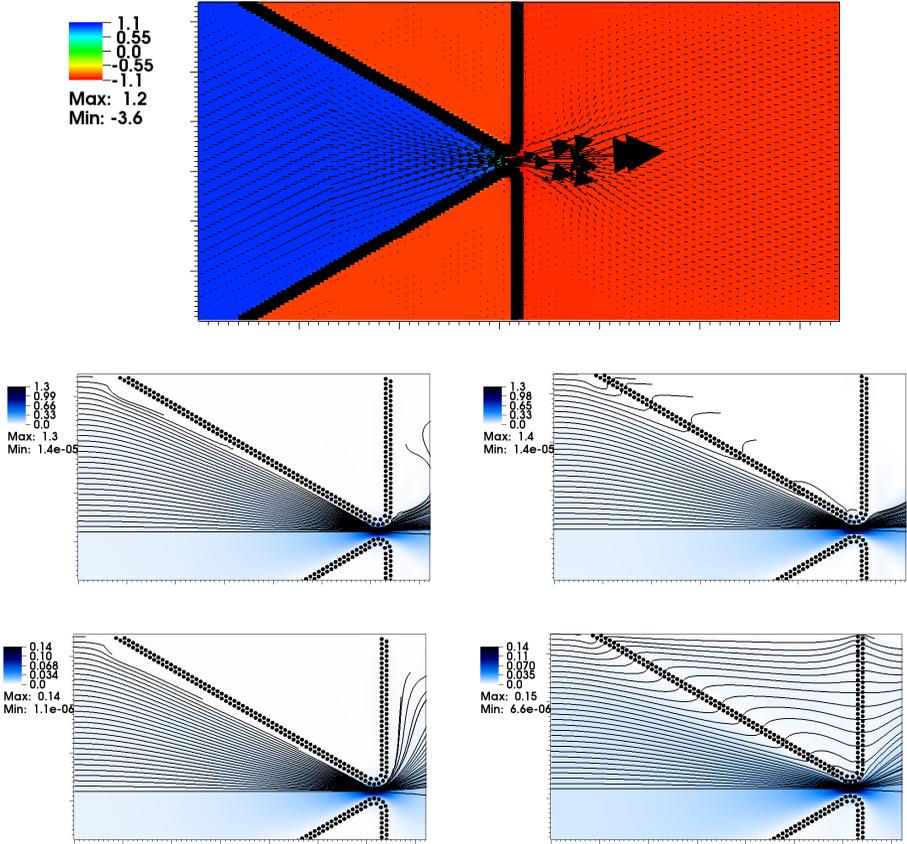
Figure 5. Two-dimensional flow through a nozzle (top panel) in a slit channel computed by our rigid IB method (left panels) and a simplified version of the splitting-based method of Bhalla et al. [13] (right panels). Top panel: the geometry of the channel along with the (approximately) steady state flow at Re≈19, as obtained using our method. The color plot shows the pressure and the velocity is shown as a vector field. Middle panels: flow at Re ≈ 19, computed at time $T = 10^2$. For our method (left) we use a time step size of $\Delta t = 5 \cdot 10^{-2}$ (corresponding to advective CFL number of $U_{\max} \Delta t / \Delta x \approx 0.13$), while for the splitting method (right) we use $\Delta t = 10^{-3}$. The streamlines are traced from the entrance to the channel for a time of $T_s = 7 \cdot 10^3$ and shown as black lines. Bottom panels: same as the middle row but now for Re ≈ 0.2, final time $T = 10$ and streamlines followed up to $T_s = 4 \cdot 10^4$, with $\Delta t = 0.125$ for our method (left), and $\Delta t = 10^{-3}$ for the splitting method (right).

the constrained fluid problem in each time step anew to a tight GMRES tolerance of 10^{-9}. For this test we use $N_s = 2$ iterations in the fluid solves inside our preconditioner.

The visual results in the right panels of Figure 5 clearly show that the splitting errors in the enforcement of the no-slip boundary condition lead to a notable "leak"

through the boundary, especially at small Reynolds numbers. To quantify the amount of leak we compute the ratio of the total flow through the opening of the nozzle to the total inflow; if there is no leak this ratio should be unity. Indeed, this ratio is larger than 0.99 for our method at all Reynolds numbers, as seen in the lack of penetration of the flow inside the body in the left panels in Figure 5. For $Re \approx 19$, we find that even after reducing the time step by a factor of 50, the splitting method gives a ratio of 0.935 (i.e., 6.5% leak), which can be seen as a mild penetration of the flow into the body in the middle right panel in Figure 5. For $Re \approx 0.2$, we find that we need to reduce Δt by a factor of 1250 to get a flow ratio of 0.94 for the splitting method; for a time step reduced by a factor of 125 there is a strong penetration of the flow through the nozzle, as seen in the bottom right panel of Figure 5.

C. *Stokes flow between two concentric shells.* Steady Stokes flow around a fixed sphere of radius R_1 in an unbounded domain (with fluid at rest at infinity) is one of the fundamental problems in fluid mechanics, and analytical solutions are well known. Our numerical method uses a regular grid for the fluid solver, however, and thus requires a finite truncation of the domain. Inspired by the work of Balboa Usabiaga et al. [8], we enclose the sphere inside a rigid spherical shell of radius $R_2 = 4R_1$. This naturally provides a truncation of the domain because the flow exterior to the outer shell does not affect the flow inside the shell. Analytical solutions remain simple to compute and are given in Appendix C.

We discretize the inner sphere using a spherical shell of markers, since for steady Stokes flow imposing a rigid body motion on the surface of the inner sphere
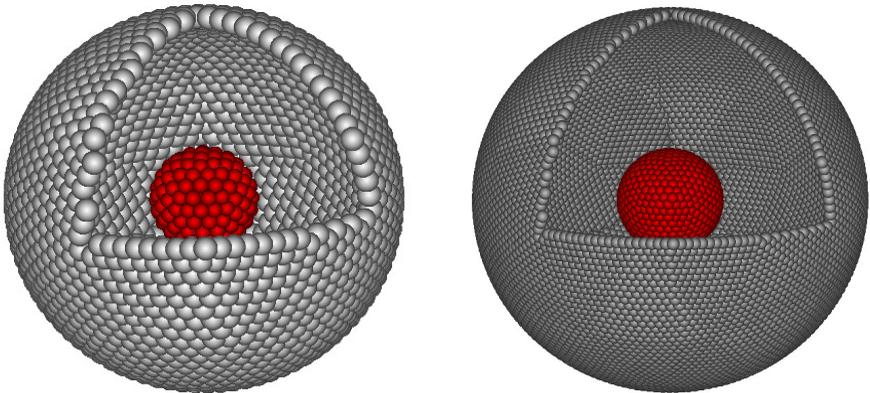
Figure 6. Marker configuration for computing Stokes flow between two concentric spherical shells. Markers are shown as spheres with size on the order of their effective hydrodynamic radius. The inner shell of markers is shown in red, and the outer shell of markers is shown in gray. Left: intermediate resolution, inner shell of 162 markers and outside shell of 2562 markers. Right: highest resolution studied here, inner shell of 642 markers and outside shell of 10242 markers.

guarantees a stress-free rigid body motion for the fluid filling the inner sphere [23]. We use the same recursive triangulation of the sphere, described in Section V, to construct the marker grid for both the inner and outer shells, as illustrated in Figure 6. The ratio of the number of markers on the outer and inner spheres is approximately 16 (i.e., there are two levels of recursive refinement between the inner and outer shells), consistent with keeping the marker spacing similar for the two shells and a fixed ratio $R_2/R_1 = 4$. The fluid grid size is set to keep the markers about two grid cells apart, $s \approx 2h$. The rigid-body velocity is set to $V = (1, 0, 0)$ for all markers on the outer shell, and to $V = 0$ on all markers on the inner shell. The outer sphere is placed in a cubic box of length $l = 4.15R_2$ with specified velocity $v = (1, 0, 0)$ on all of the boundaries; this choice ensures that the flow outside of the outer shell is nearly uniform and equal to $v = (1, 0, 0)$. In the continuum setting, this exterior flow does not affect the flow of interest (which is the flow in-between the two shells), but this is not the case for the IB discretization since the regularized delta function extends a few grid cells on *both* sides of the spherical shell.

A spherical shell of *geometric* radius R_g covered by markers acts hydrodynamically as a rigid sphere of effective *hydrodynamic* radius $R_h \approx R_g + a$ [79], where a is the hydrodynamic radius of a single marker [16; 9; 27] (we recall that for the six-point kernel used here, $a \approx 1.47\,h$). A similar effect appears in the lattice Boltzmann simulations of Ladd, with a being related to the lattice spacing [54; 79; 45]. When comparing to theoretical expressions, we use the effective hydrodynamic radii of the spherical shells (computed as explained below) and *not* the geometric radii. Of course, the enhancement of the effective hydrodynamic radius over the geometric one is a numerical discretization artifact, and one could choose not to correct the geometric radius. However, this comparison makes immersed boundary models of steady Stokes flow appear much less accurate than they actually are in practice. For example, one should not treat a line of markers as a zero-thickness object of zero geometric radius; rather, such a line of rigidly connected markers should be considered to model a rigid cylinder with finite thickness proportional to a [16].

We can measure the effective hydrodynamic radius R_h of a spherical shell of markers from the drag force on a periodic cubic lattice of such objects moving with velocity V. Specifically, we place a single shell of N markers in a triply periodic domain with cubic unit cell of length l, set $V = (1, 0, 0)$ on all markers, solve (3), and measure the total drag force as $F = \sum_{i=1}^{N} \Lambda_i$. The periodic correction to the Stokes drag formula is well-known [41]:

$$\frac{F}{\eta V} = \frac{6\pi R_h}{1 - 2.8373(R_h/l) + 4.19(R_h/l)^3 - 27.4(R_h/l)^6 + \text{h.o.t.}}, \quad (23)$$

and allows us to obtain a very accurate estimate of R_h from the drag for $l \gg R_h$. The results are given in the left half of Table 1 in the form of the dimensionless ratio

resolution grid size	number markers inner shell	R_h/R_g	number markers outer shell	R_h/R_g
30^3	12	1.48	162	0.93
60^3	42	1.22	642	0.96
120^3	162	1.09	2562	0.98
240^3	642	1.04	10242	0.99

Table 1. Ratio of the effective hydrodynamic and geometric radii of the inner (left half) and outer (right half) spherical shells for simulations of steady Stokes flow around a fixed sphere embedded within a moving spherical cavity, at different resolutions.

R_h/R_g; we see that as the resolution is increased $R_h \to R_g$ with an approximately linear rate of convergence, as expected. Since this computation refers to flow outside of the shell of markers, we can call the computed R_h the effective *outer* hydrodynamic radius and use it to set the value of R_1 in the theory. We use a similar procedure to measure an effective *inner* hydrodynamic radius R_2 for the outer spherical shell. Specifically, we obtain R_2 from the drag on the inner sphere based on the theoretical formula (C.1), where we use the previously determined value of R_1 for the effective radius of the inner sphere. The results are given in the right half of Table 1 and again show that as the grid is refined the hydrodynamic radii converge to the geometric ones.

1. *Convergence of fluid flow (pressure and velocity).* The top panel of Figure 7 shows a slice through the middle of the nested spherical shells along with the fluid velocity v. Recall that the flow inside the inner sphere should vanish, implying that the pressure inside the inner shell should be constant (set to zero here), and the flow outside of the outer sphere should be uniform. The bottom right panel of the figure zooms in around the inner sphere to reveal that there is some spurious pressure gradient and an associated counterrotating vortex flow generated inside the inner sphere. The bottom right panel shows the error in the computed fluid flow (v, π), that is, the difference between the computed flow and the theoretical solution given in Appendix C. It is clear that the majority of the error is localized in the vicinity of the inner shell and in the interior of the inner sphere. Note that these errors would be much larger if the theory had used the geometric radii instead of the hydrodynamic radii for the shells.

Table 2 shows the norms of the error in the computed flow field as a function of resolution. Asymptotically first-order convergence is observed in the L_1 and L_2 norms for both the velocity and the pressure. In the L_∞ norm, we expect the velocity to also converge linearly, but we do not expect to see convergence in the pressure, since the velocity is continuous across the interface but the pressure has a jump; this is consistent with the numerical data.
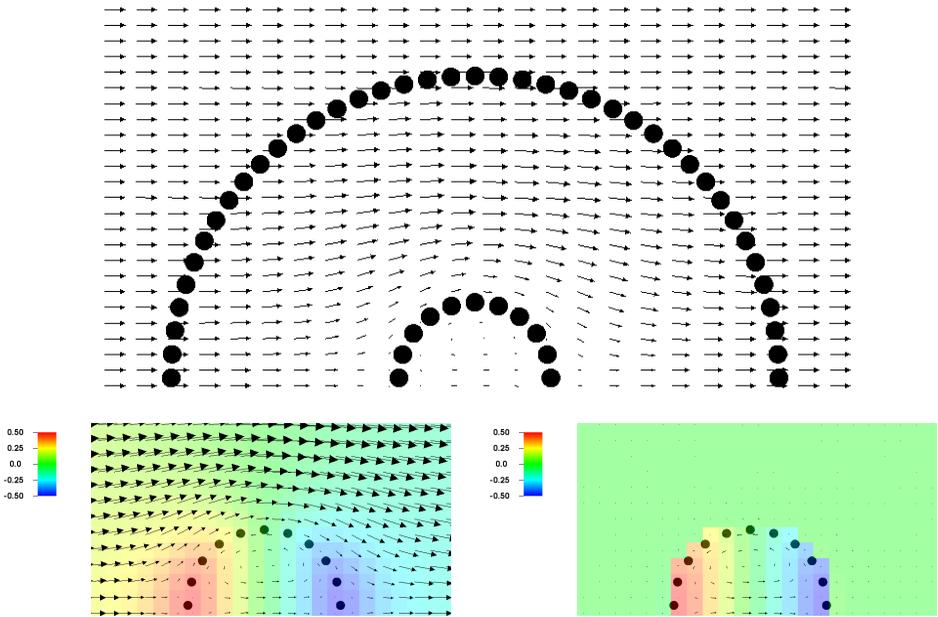
Figure 7. Flow field around a fixed sphere inside a moving spherical cavity. The outer shell is discretized using 2562 markers, while the inner one has 162 markers, shown as black circles. Top: velocity field. Bottom left: zoom of the velocity (vector field) and pressure (color plot) around the inner shell. Bottom right: same as bottom left but now showing the *error* in the velocity and pressure compared to the theoretical expressions.

markers	resol	$\|\Delta\boldsymbol{v}\|_1/\|\boldsymbol{v}\|_1$	rate	$\|\Delta\boldsymbol{v}\|_2/\|\boldsymbol{v}\|_2$	rate	$\|\Delta\boldsymbol{v}\|_\infty/\|\boldsymbol{v}\|_\infty$	rate
162-12	30^3	$4.08 \cdot 10^{-2}$		$6.39 \cdot 10^{-2}$		0.558	
642-42	60^3	$1.14 \cdot 10^{-2}$	1.83	$2.08 \cdot 10^{-2}$	1.62	0.322	0.79
2562-162	120^3	$4.61 \cdot 10^{-3}$	1.30	$8.74 \cdot 10^{-3}$	1.24	0.160	1.01
10242-642	240^3	$2.16 \cdot 10^{-3}$	1.09	$4.26 \cdot 10^{-3}$	1.04	0.091	0.82

markers	resol	$\|\Delta\pi\|_1/\|\pi\|_1$	rate	$\|\Delta\pi\|_2/\|\pi\|_2$	rate	$\|\Delta\pi\|_\infty/\|\pi\|_\infty$	rate
162-12	30^3	0.849		0.788		1.0	
642-42	60^3	0.567	0.58	0.486	0.70	1.0	0.0
2562-162	120^3	0.344	0.72	0.275	0.82	0.860	0.22
10242-642	240^3	0.196	0.81	0.164	0.75	0.704	0.29

Table 2. Normalized norms of the error in the computed velocity $\boldsymbol{v}$ (top) and the pressure π (bottom) for steady Stokes flow around a fixed sphere embedded within a moving spherical cavity, at different resolutions (see two leftmost columns, where "Resol" denotes resolution grid size). An estimated order of convergence based on successive refinements is indicated in the column to the right of the corresponding error norm.

2. *Convergence of Lagrangian forces (surface stresses).* The first-order conver-
gence of the pressure and velocity is expected and well-known in the immersed
boundary community. The convergence of the *tractions* ($\boldsymbol{\sigma} \cdot \boldsymbol{n}$) on the fluid-body
interface is much less well studied, however. This is in part because in penalty-
based or splitting methods, it is difficult to estimate tractions precisely (e.g., for
penalty methods using stiff springs, the spring tensions oscillate with time), and in
part because a large number of other studies have placed the markers too closely
to obtain a well-conditioned mobility matrix and thus to obtain accurate forces.
Furthermore, there are at least two ways to estimate surface tractions in IB methods,
as discussed in extensive detail in [81]. One method is to estimate fluid stress from
the fluid flow and extrapolate toward the boundary. Another method, which we use
here, is to use the computed surface forces $\boldsymbol{\Lambda}$ to estimate the tractions.

We obtain pointwise estimates of the tractions at the positions of the markers
from the relation $(\boldsymbol{\sigma} \cdot \boldsymbol{n})(\boldsymbol{R}_i) \approx \boldsymbol{\Lambda}_i / \Delta A_i$, where ΔA_i is the surface area associated
with marker i. We obtain ΔA_i from the surface triangulation used to construct the
marker of grids by assigning one third of the area of each triangle to each of its
nodes. In Figure 8 we show the computed normal and tangential components of
the traction in polar coordinates, with the z symmetry axes along the direction of
the flow. The theoretical prediction given in Appendix C is shown with a black line
and is based on the *geometric* radii.

The top row of Figure 8 shows the computed tractions for several resolutions with
marker spacing $s \approx 2h$. It is seen that as the grid is refined, the computed tractions
appear to converge *pointwise* to the correct values. However, the convergence is
very slow, and even for the large resolutions reported here, it is evident that the
asymptotic convergence regime has not been reached. Consequently, no precise
statement about the order of convergence can be made from these data. At lower
resolution, some of the results even show qualitatively wrong behavior. For example,
the normal traction $\sigma_n = \boldsymbol{n} \cdot \boldsymbol{\sigma} \cdot \boldsymbol{n}$ for a resolution of 42 inner and 642 outer markers
grows with θ, but the theoretical result decreases with θ. We also see scatter in the
values among individual markers, indicating that the geometrical and topological
nonuniformity of the marker grid affect the pointwise values.

Nonetheless, we remark that low-order *moments* of the surface tractions are much
more accurate than the pointwise tractions. For example, the total drag on the inner
sphere is much more accurate, as seen in Table 1. Other test problems not reported
here indicate that stresslets are also computed quite accurately, especially if one
accounts for the distinction between geometric and hydrodynamic radii. These
findings suggest that weak convergence of the tractions is more robust than strong
convergence. In fact, lower order moments can show reasonable behavior even if
the marker spacing is small and the pointwise forces are numerically unstable to
compute. Unsurprisingly, we find that the pointwise traction estimates are improved
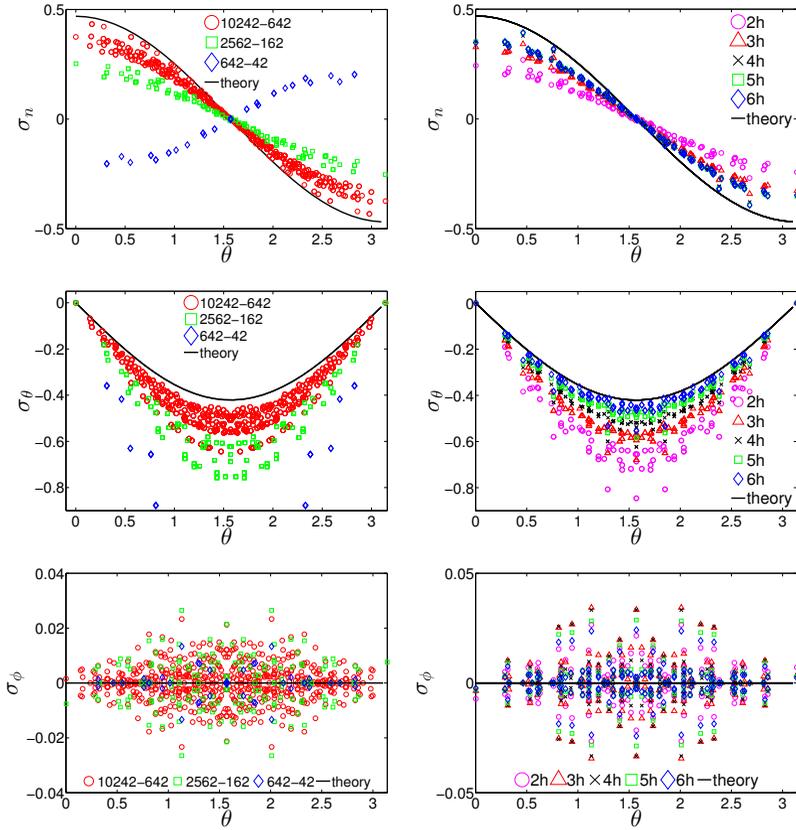
Figure 8. Convergence of surface stresses to their theoretical values for the three different resolutions. Pointwise traction estimates are shown with symbols as a function of the angle θ relative to the direction of the flow, while the theory is shown with a solid black line. Top row: normal component of the traction $\sigma_n = \hat{r} \cdot \sigma \cdot n$. Center row: tangential component of traction in direction of flow, $\sigma_\theta = \hat{\theta} \cdot \sigma \cdot n$. Bottom row: tangential component in the direction perpendicular to the flow, $\sigma_\phi = \hat{\phi} \cdot \sigma \cdot n$, which should vanish by symmetry. Left column: different resolutions (see legend) for a fixed spacing $s \approx 2h$. Note that for the coarsest resolution of only 12 markers on the inner sphere, the computed tractions have values off the scale of this plot and are thus not shown. Right column: the most resolved case of 2562–162 markers for different spacing between markers, as indicated in the legend. Note that using $s \approx h$ leads to severe ill-conditioning and the computed tractions show random scatter well beyond the scale of the plot and are thus not shown.

as the spacing among the markers is increased; see the bottom row in Figure 8. The improvement is not only due to the reduction of the scatter, as expected from the improvement in conditioning number of the mobility matrix, but also due to global reduction of the error in the tractions; the observed global reduction may, however, be specific to steady Stokes flow. For widely spaced markers, however, the error in the computed flow field will increase because the flow will penetrate

the shell boundary. This once again demonstrates the delicate balance that is required in choosing the marker spacing for rigid bodies, as we discuss further in the Conclusions.

D. *Steady Stokes flow around sphere in a slit channel.* In this section we study a problem at zero Reynolds numbers with nontrivial boundary conditions, namely, steady Stokes flow around a sphere in a slit channel (flow between two parallel walls). It is well-known that computing flows in such geometries using Green's function based methods such as boundary-integral methods is highly nontrivial [51; 59; 73]. Specific methods for spheres in a channel have been developed [15] but these are not general, in particular, flow in a square channel requires a different method, and incorporating the periodicity in some of the dimensions is nontrivial [1; 59]. At the same time, we wish to point out that boundary-integral methods have some advantages over our IB method as well. Notably, they are *considerably* more accurate, and handling domains unbounded in one or more directions is possible by using the appropriately decaying Green's function.

Unlike the case of a single no-slip boundary, writing down an analytical solution for slit channels is complex and requires numerically evaluating the coefficients in certain series expansions [73]. For the component of the mobility $\mu = F/V$ of a sphere in an infinite slit channel, Faxen has obtained exact series expansions for the mobility at the half and quarter channel locations,

$$\mu_\parallel\left(H = \frac{d}{2}\right) = \frac{1}{6\pi\eta R_h}\left[1 - 1.004\frac{R_h}{H} + 0.418\frac{R_h^3}{H^3} + 0.21\frac{R_h^4}{H^4} - 0.169\frac{R_h^5}{H^5} + \cdots\right],$$

$$\tag{24}$$

$$\mu_\parallel\left(H = \frac{d}{4}\right) = \frac{1}{6\pi\eta R_h}\left[1 - 0.6526\frac{R_h}{H} + 0.1475\frac{R_h^3}{H^3} - 0.131\frac{R_h^4}{H^4} - 0.0644\frac{R_h^5}{H^5} + \cdots\right],$$

where R_h is the (hydrodynamic) radius of the sphere, H is the distance from the center of the particle to the nearest wall, and d is the distance between the walls.

To simulate a spherical particle in a slit channel we place a single spherical shell with different number of IB markers in a domain of size $L \times L \times d$, at either a quarter or half distance from the channel wall. No slip walls are placed at $z = 0$ and $z = L$, and periodic boundary conditions are applied in the x and y directions. For each L, we compute an effective hydrodynamic radius R_L by assuming (24) holds with R_h replaced by R_L. We know that as $L \to \infty$ we have $R_L \to R_h$, however, we are not aware of theoretical results for the dependence $R_L(L)$ at finite L. In Figure 9, we plot $R_L/R_h - 1$ versus R_h/L for $d \approx 8R_h$. Here the effective hydrodynamic radius of the shell R_h is estimated by using (23), as shown in Table 1 (see inner radius). We see that we have consistent data for $R_L(L)$ among different resolutions, and we obtain consistency in the limit $L \to \infty$. This indicates that even a low-resolution
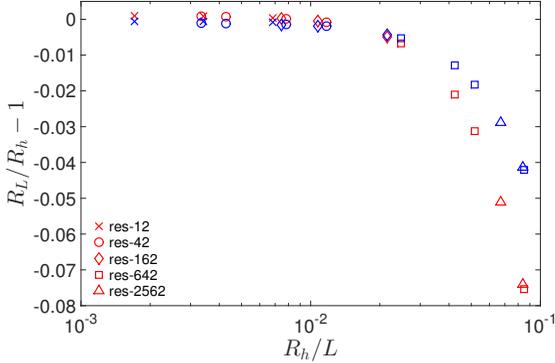
Figure 9. Effective hydrodynamic radius $R_L(L)$ of a sphere of hydrodynamic radius R_h, translating parallel to the walls of a slit channel of dimensions $L \times L \times d$. Red symbols are for the sphere at the midplane of the channel, $H/d = 0.5$, and blue symbols are for $H/d = 0.25$; these two give different dependence $R_L(L)$ as expected. The channel width is taken $d \approx 8R_H$ and different numbers of markers are used for the sphere (see legend), and the grid spacing is set to give a marker spacing s as close as possible to $a/s \approx 0.5$. Note that similar to the example of flow between concentric spheres the correct value of the drag is determined by the larger hydrodynamic and *not* by the geometric radius of the shell.

model with as few as 12 markers offers a reasonably accurate model of a sphere of effective radius R_h, *independent* of the boundary conditions.

E. *Steady Stokes flow around cylinders.* Here we study the drag force on a periodic square array of cylinders (i.e., disks in two dimensions) with lattice spacing l. The corresponding study in three dimensions is presented in Section G. The analog of (23) in two dimensions for *dilute* systems is [41; 52]

$$\frac{F}{\eta V} = \frac{4\pi}{-\ln(\sqrt{\phi}) - 0.738 + \phi - 0.887\phi^2 + 2.038\phi^3 + O\left(\phi^4\right)}, \qquad (25)$$

where $\phi = \pi R_h^2 / l^2$ is the packing fraction of the disks and R_h is the hydrodynamic radius of the cylinder, which is defined from (25). Observe that in two dimensions, there is no limit as $\phi \to 0$, in agreement with Stokes's paradox for flow around a single cylinder; one must account for inertial effects for very small volume fractions in order to obtain physically relevant results. Table 3 reports R_h for several different marker models of a cylinder, as estimated by computing the drag for a range of packing fractions and extrapolating to $\phi \ll 1$ using (25). As expected, the more resolved the cylinder, the closer R_h is to R_g. Filling the cylinder with markers both substantially enlarges the effective hydrodynamic radius and also degrades the conditioning of the mobility matrix, and is therefore not advised at zero Reynolds number.

number markers	surface markers	interior markers	R_h/R_g
39 shell	39	0	1.04
121 cylinder	37	84	1.15
100 shell	100	0	1.02
834 cylinder	100	734	1.03

Table 3. Hydrodynamic radii of several discretizations of a cylinder with different numbers of markers on the surface and the interior of the body, keeping $s/h \approx 2$. Two models have markers only on the surface of the cylinder (see Figure 12, right panel, for a 39-marker shell). The rest of the models are constructed from a regular polar grid of markers filling the interior of the cylinder (see Figure 12, left panel, for a 121-marker cylinder).

Another interesting limit for which there are theoretical results is the *dense* limit, in which the disks/cylinders almost touch, so that there is a lubrication flow between them. In this limit [52],

$$\frac{F}{\eta V} \approx \frac{9\pi}{2^{\frac{3}{2}}} \varepsilon^{-\frac{5}{2}}, \qquad (26)$$

where $\varepsilon = 1 - \sqrt{4\phi/\pi} = (l - 2R_h)/l$ is the relative gap between the particles. Note that because the number of hydrodynamic cells must be an integer, we cannot get an arbitrary gap between the cylinders for a given cylinder model and fixed s/h (i.e., a fixed R_h/R_g). Also note that when the gap between the cylinders is too small, the kernels from markers on two cylinders start to overlap, and the problem becomes ill-conditioned; we have been able to compute reliable results down to a relative gap of $\varepsilon \gtrsim 10^{-2}$ for the resolutions studied here.

Numerical results for the normalized drag over a broad range of volume fractions are shown in Figure 10 and compared to results obtained using an in-house two-dimensional version of the spectrally accurate boundary integral method proposed
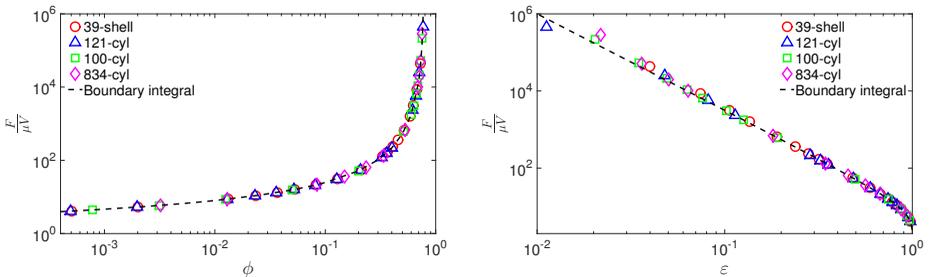
Figure 10. The drag coefficient for a periodic array of cylinders in steady Stokes flow for different resolutions (see Table 3). Left: as function of volume fraction, compared to the results of a highly accurate boundary-integral method. Right: zoom in for close-packed arrays with interparticle gap plotted on a log scale to show the asymptotic $\varepsilon^{-5/2}$ divergence of the lubrication force.

in [1]. We obtain very good agreement, similar to that observed using the lattice Boltzmann method [52], indicating that even moderately resolved cylinders are good representations so long as one uses their hydrodynamic rather than their geometric radius when computing the effective volume fraction. In particular, in the right panel, we obtain excellent agreement with the lubrication result (26), seeing an increase in the drag of over six orders of magnitude consistent with theory. Of course, the IB method results for the drag do not have a true divergence as $\epsilon \to 0$ because of the regularization of the singular kernel; one must use adaptively refined nonregularized boundary integral methods to truly resolve the divergence. In practice, however, effects not included in the theoretical model, such as surface roughness or partial slip, will mollify the unphysical divergence.

F. *Unsteady flow around cylinders.* Next, we examine the ability of our rigid-body IB method to model unsteady two-dimensional flow around cylinders (disks). We define Reynolds number by

$$\mathrm{Re} = \frac{\rho V R_h}{\eta} = \frac{V R_h}{\nu},$$

where R_h is the hydrodynamic radius of the cylinder measured at $\mathrm{Re} = 0$ (see Table 3), and V is the velocity of the incident flow. For small Reynolds numbers, the mean drag per unit length F is given [52] by

$$\frac{F}{\eta V} = k_0 + k_2 \mathrm{Re}^2,$$

where $k_0\,(\phi)$ and $k_2\,(\phi)$ are constants that depend on the packing fraction ϕ (defined using the hydrodynamic radius). In the range $\mathrm{Re} \sim 2-5$, the drag becomes quadratic in the flow rate [52], and for moderate Reynolds numbers, a drag coefficient is defined from the empirical relation

$$C_D = \frac{F}{\rho V^2 R_h}.$$

As the Reynolds number is increased, the flow becomes unsteady and vortex shedding occurs, and eventually there is a transition to three-dimensional flow. Here we focus on steady flow at $\mathrm{Re} \leq 100$.

A staggered-grid variant of the piecewise-parabolic Godunov method is used for spatial discretization of the advective terms, as explained in detail by Griffith [34]. In our tests, the time step size is determined by fixing the advective Courant number $V \Delta t / h = 0.1$; this value is well bellow the stability limit and ensures that the discretization errors coming from the (unconstrained) fluid solver are small. The Adams–Bashforth method is used to handle advection explicitly. The viscous terms are handled implicitly using the backward Euler method rather than the implicit

midpoint rule because we are interested in steady states and not transient phenomena. We initialize the simulations with the fluid moving at a uniform velocity but allow enough time for a steady-state to be reached.

1. *Drag on periodic array of disks.* The permeability of a periodic array of aligned cylinders is a well-studied problem and can be computed by placing a single cylinder in a periodic domain. To create flow through the periodic system, we follow Ladd et al. [45; 52] and apply a constant body force f throughout the domain (including in the interior of the body). We solve the constrained time-dependent problem to a steady state, keeping the cylinder at rest, and measure the average velocity in the domain, $\bar{v} = \text{vol}^{-1} \int_V v \, dr$. In two spatial dimensions, the dimensionless drag coefficient is defined by

$$k = \frac{F_x}{\eta \bar{v}_x},$$

where the force $F = \text{vol} \, f = -\mathbf{1}^T \Lambda$ is the total force applied to the fluid, which must also equal the negative of the total force exerted on the rigid body.

Theory suggests that the correction to the drag scales as Re^2 for small Reynolds numbers due to the antisymmetry of the correction to the flow (relative to steady Stokes) of order Re [52], so that

$$k = k_0 + k_2 \text{Re}^2, \tag{27}$$

where the values $k_0(\phi)$ and $k_2(\phi)$ depend on the packing fraction ϕ. To obtain k_0, we move the body at a constant velocity and obtain the drag force $\mathbf{1}^T \Lambda$ from the solution of the constrained steady Stokes problem (11). Because marker-based models of rigid bodies do not have perfect symmetry, the force $f_0 = -\text{vol}^{-1} (\mathbf{1}^T \Lambda)$ has small nonzero components in the direction perpendicular to the flow. To ensure that in the limit $\text{Re} \to 0^+$ we have perfect consistency between the finite Re and zero Re computations, we use the force $f = (k/k_0) f_0$ to drive the flow at finite Re numbers. Note that it can take thousands of time steps for the steady state to be established for $\text{Re} \gtrsim 1$; to accelerate convergence, we initialize the computation for a given Re from the steady state for the closest smaller Re. Also note that the exact mobility matrix $\mathcal{M}$ and its factorization can be precomputed once at the beginning and used repeatedly for these steady-state calculations.

Figure 11 shows the dimensionless excess drag $k - k_0$ as a function of Re at packing fraction $\phi = 0.193$, which is close to the packing fraction $\phi = 0.2$ studied using the lattice Boltzmann method in [52]. We see very good agreement of the theoretical formula (27) with our results using the values of $k_0 = 49.2$ and $k_2 = 0.24$, which are in good agreement with the values of $k_0 = 51.2$ and $k_2 = 0.26$ given in Figure 1 of [52].
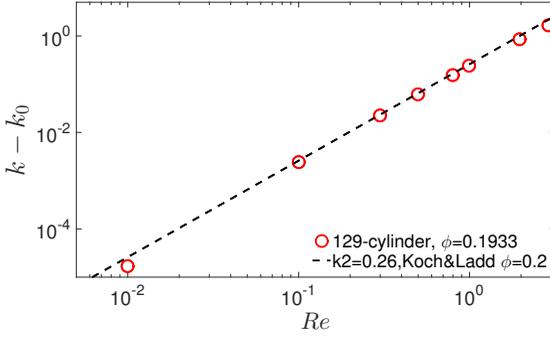
Figure 11. The dimensionless excess (over Stokes flow) drag coefficient for a square array of disks with packing fraction $\phi \approx 0.193$ (129-marker filled cylinder model, fluid grid of 64^2 cells). Comparison is made to known small-Re dependence of the form $k_0 + k_2 \mathrm{Re}^2$, with the coefficients k_0 and k_2 taken from the work of Koch and Ladd [52] at $\phi = 0.2$.

2. *Flow past a periodic column of cylinders.* Here we compute several solutions for flow past a column of cylinders at somewhat larger Reynolds numbers, mimicking the setup of Ladd [54]. The domain is a long narrow channel of 2048×128 grid cells[8] with grid spacing $h = 0.5$, keeping the markers at a distance $2h$. Periodic boundary conditions are used in the direction of the short side of the channel (y). The flow is driven by "uniform" inflow and outflow boundary conditions in the long direction (x). Specifically, we impose a specified normal velocity V and zero tangential velocity at both ends of the channel.[9] The center of the cylinder is fixed at a quarter channel length from the inlet. The cylinders in the periodic column are separated by approximately 10 hydrodynamic radii in the y direction (the separation is $9.958R_h$ for the 121-marker cylinder, and $9.875R_h$ for the 39-marker shell).

Representative flow fields are shown in Figure 12 for $\mathrm{Re} = 10$ for a filled cylinder model (left) and an empty shell model (right). Note that for the computation of total drag on a fixed cylinder either model can be used since the spurious flow seen inside the empty shell does not generate any overall acceleration of the fluid inside the body. Also note that the spurious counterrotating vortex pair inside the shell diminishes under refinement, at an approximately linear convergence rate, just as for steady Stokes flow. Computed drag coefficients k and wake lengths are shown in Table 4, and good agreement is seen with the results of lattice Boltzmann and finite difference schemes [54]. The wake length measures the distance from the

---

[8]For very elongated domains, our multigrid-based preconditioner converges much faster for grid sizes that are powers of two.

[9]An alternative is to use zero tangential stress on both boundaries, or zero normal and tangential stress on the outflow; such stress boundary conditions are supported in the fluid solver in the IBAMR library [34].
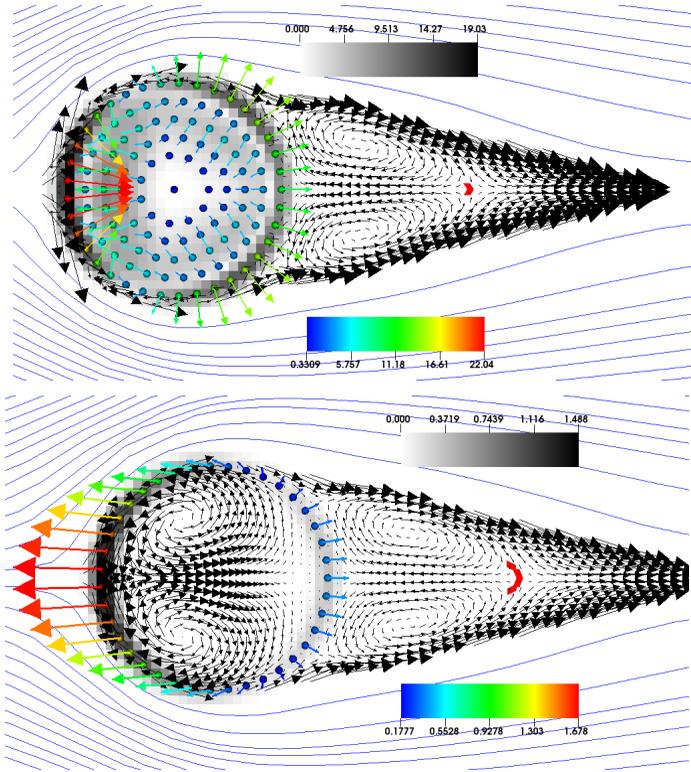
Figure 12. Steady incompressible flow at Re $= 10$ past a periodic column of cylinders represented as either filled disks of 121 markers (top) or a shell of 39 markers (bottom). We show the magnitude of the Eulerian constraint force $\mathcal{S}\mathbf{\Lambda}$ (gray color map), the streamlines outside of the wake (solid blue lines), the wake velocity field (black arrows), and the Lagrangian constraint forces associated with each marker (color arrows). The red arrow marks the stagnation point where $v_x = 0$, as used to determine the wake length.

cylinder center to the stagnation point, which is obtained by finding the largest x coordinate on the contour of zero horizontal velocity, $v_x = 0$.

G. *Flow past periodic arrays of spheres.* Finally, we study the drag on a cubic arrays of spheres of radius a at zero and finite Reynolds numbers, and compare our results to those of Hill et al. [45]. At small packing (volume) fractions ϕ and Reynolds numbers, according to Equations (1-2) in [44], $F - F_0 = 3\mathrm{Re}/8 + \text{h.o.t.}$ if $\sqrt{\phi} \ll \mathrm{Re} \ll 1$, or, more relevant to our study, $F - F_0 \sim \mathrm{Re}^2/\sqrt{\phi}$ if $\mathrm{Re} \ll \sqrt{\phi} \ll 1$. For small Re and at larger densities, the theoretical arguments in [45; 52] predict that the dimensionless drag is quadratic in Re because the linear term vanishes by symmetry, so that

$$k = \frac{F}{6\pi \eta a V} \approx k_0 + k_2 \mathrm{Re}^2.$$

Re	121 cyl	39 shell	LB	FD
5	4.31	4.35	4.21	4.32
10	2.96	2.99	2.91	2.98
20	2.16	2.19	2.17	2.19
50	1.55	1.58	1.67	1.61

Re	121 cyl	39 shell	LB	FD
5	1.52	1.40	1.5	1.49
10	2.55	2.59	2.6	2.65
20	4.50	4.61	4.7	4.74
50	9.96	9.91	10.7	10.3

Table 4. Numerical results for steady flow past a periodic column of cylinders at different Reynolds number, for two different models of the body (see Figure 12), either a filled cylinder or an empty shell of markers. For comparison we reproduce the results in Table 5 in [54], which are computed either using either a lattice Boltzmann (LB) or a finite difference (FD) method. Top half: mean drag coefficient. Bottom half: wake length in units of R_h.

For larger Re, the dependence is expected to switch to linear in Re.

Here we focus on close-packed cubic lattices of spheres with packing fraction $\phi = \pi/6 \approx 0.5236$. Note that unlike the case of two spatial dimensions, in three dimensions the flow does not need to squeeze in-between the (nearly) touching bodies, so the drag does not diverge even at close packing. The value of the steady Stokes drag k_0 is tabulated in Table 5 for several resolutions. Different resolutions are examined: an empty shell (see Table 1) of 162 (grid size is 16^3) or 642 markers (30^3 grid), as well as a filled sphere of 56 (42 on the surface, 10^3 grid) or 239 (162 on the surface, 16^3 grid) markers; the actual value of the packing fraction based on the effective hydrodynamic radius of the model is indicated in the table. A large difference is seen between the filled and empty shell models at this high packing fractions because the spheres are very close to each other and discretization artifacts

number of markers	ϕ	k_0
56 filled	0.5236	40.08
239 filled	0.5238	40.73
162 shell	0.5213	44.49
642 shell	0.5236	43.29

Table 5. Dimensionless drag force k_0 for steady Stokes flow (Re $= 0$) past a simple-cubic array of spheres at volume fraction $\phi \approx \pi/6$ (close packing). For the highest-resolution LB simulations in [54] the reported value is $k_0 = 42.8$.

become pronounced. We have also performed simulations at a lower (but still high) packing fraction of $\phi = 0.44$, and there we see much better agreement between the filled and empty sphere models; note that at small $\phi \ll 1$ the value of k_0 must match among resolutions since we *define* the packing fraction from R_h, which is itself determined from the value of k_0 at small ϕ using (25).

Numerical results for the dimensionless drag coefficient k near the close-packed density $\phi \approx 0.52$ are shown in Figure 13. Because our discrete models of spheres do not have the same symmetry as a perfect sphere, we numerically observe a small $O(\mathrm{Re})$ correction that can dominate the true correction $k_2\mathrm{Re}^2$ for $\mathrm{Re} \ll 1$; this is especially evident in the right panel of Figure 13 for coarsely resolved models (e.g., a 56-marker sphere).[10] Empirical fits to literature data for k_0, k_1, k_2, and the range of Re values over which the various fits are valid are tabulated in [12]. Figure 13

---

[10]In two dimensions, we can more easily make the discrete models symmetric and this is why Figure 11 does not show deviations from the expected quadratic behavior even at rather small Re.
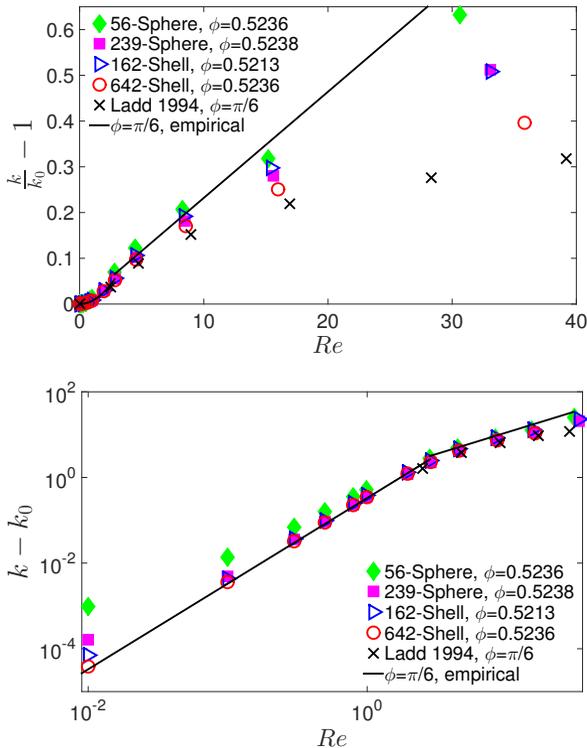
Figure 13. Numerical values (symbols) for the drag coefficient of a periodic array of spheres close packed in a cubic lattice of volume fraction $\phi = \pi/6 \approx 0.52$, for several resolutions (see legend), using a linear (top) or log scaling (bottom). Comparison is made to empirical formulas given in [12] (lines), as well as lattice Boltzmann results for close-packed cubic arrays given in Table 6 of [54] (crosses).

compares our results to these fits, as well as to reference results obtained using the lattice Boltzmann method [54] at close packing. We observe the expected switch from linear to quadratic dependence on Re and also a reasonable agreement with the literature data, and the agreement appears to improve with increasing resolution.

VIII. Conclusions

This paper develops an immersed boundary method that enforces strict rigidity of immersed bodies at both zero and finite Reynolds numbers. Unlike existing approaches, we do not rely on penalty or splitting approaches, and we instead directly solve a saddle-point system that couples the fluid velocity and pressure to the unknown rigidity forces. We developed a physics-inspired approximation $\widetilde{\mathcal{M}}$ to the Schur complement (mobility matrix) $\mathcal{M}$ of the constrained system, based on analytical considerations for a continuum fluid model, and demonstrated that this leads to a robust preconditioner so long as the immersed boundary markers are kept sufficiently far to ensure a well-conditioned mobility matrix. Contrary to common practice, we found that the markers should be kept approximately two fluid grid cells apart in rigid-body models in order to obtain accurate and stable pointwise estimates for the traction. We tested our method on a number of standard test problems in both two and three spatial dimensions, and at both zero and finite Reynolds number, and we observed good agreement with theory and literature values. Although in this work we focused on rigid bodies, our method can directly be applied to study fluid flow around bodies with specified kinematics. For example, it can be used to model the flow around a swimming body deforming with a specified gait. We have implemented the method described here in the open-source IBAMR software infrastructure [37] in the hope it will be useful to other users of the IB method.

Another challenge that we did not explore here is the efficient computation of the action of $\widetilde{\mathcal{M}}^{-1}$ when there are many markers present; there are many approximate solvers and emerging fast solvers we plan to explore in the future. Of course, using dense linear algebra to solve (21) is likely to be suboptimal, as these solves have $O(N^2)$ memory complexity and $O(N^3)$ time complexity. The problem of solving a linear systems similar in structure to (21) appears in many other methods for hydrodynamics of suspensions, including Brownian [43; 49] and Stokesian [69] dynamics, the method of regularized Stokeslets [21; 23], computations based on bead models of rigid bodies [46; 20; 61; 26], and first-kind boundary integral formulations of Stokes flow [66]. Similar matrices appear in static Poisson problems such as electrostatics or reaction-diffusion models [62], and there is a substantial ongoing work that can be applied to our problem. Notably, the approximate mobility matrix $\widetilde{\mathcal{M}}$ is dense but has a well-understood low-rank structure that can be exploited. Specifically, matrix-vector products $\widetilde{\mathcal{M}}\boldsymbol{\Lambda}$ can be

performed in almost linear time using the Fast Multipole Method [57]. If the condition number of $\widetilde{\mathcal{M}}$ is not too large, one can solve linear systems involving $\widetilde{\mathcal{M}}$ efficiently using an unpreconditioned Krylov solver. For poorly conditioned cases, however, a good preconditioner based on an approximate factorization of $\widetilde{\mathcal{M}}$ is required. In recent years, several approximate low-rank factorizations of matrices of this type have been developed [3; 47; 4], and can be used as preconditioners in Krylov methods. We have had reasonable success using a fast hierarchically off-diagonal low-rank (HODLR) factorization code developed by Ambikasaran and Darve [3], with significant improvement offered by a recently developed boundary distance low-rank approximation [4]. Preliminary results indicate great promise for the inverse fast multipole (iFMM) method [24]; we have been able to use iFMM to solve the system (21) for as many as $5 \cdot 10^5$ markers to a relative tolerance of 10^{-8}. These methods are, however, still under active development, and a significant amount of investigation is necessary to integrate them into the method described here. Notably, we only require an approximate solver for (21) and the impact of the inaccuracy in solving (21) on the overall convergence of the outer Krylov solver needs to be assessed.

The type of linear system we solve here is closely connected to those appearing in implicit immersed boundary methods [60; 39; 85]. It is in fact possible to recast the saddle-point problem we consider here into a form closely related to that appearing in implicit IB methods; the Schur complement for this system is in Eulerian rather than Lagrangian variables as it was for this work, and involves the matrix

$$\mathcal{L}_v + \kappa \mathcal{S} \left(\mathcal{J} \mathcal{S} \right)^{-1} \mathcal{J}, \tag{28}$$

for some constant κ that need *not* go to infinity. It may be that geometric multigrid methods [39] developed for implicit IB methods can be applied to the Eulerian Schur complement (28). At the same time, techniques developed herein may be useful in the development of more efficient implicit IB methods for nearly rigid bodies.

Our work is only the first step toward the ultimate goal of developing methods able to handle large numbers of rigid bodies in flow. Several computational challenges need to be tackled to realize this goal. Firstly, and most importantly, it is crucial to develop a preconditioner for the enlarged linear system that appears in the context of freely moving rigid bodies. An additional Schur complement appears when solving this saddle-point problem, and the challenge for future work is approximating the body mobility matrix $\mathcal{N} = (\mathcal{K}^\star \mathcal{M}^{-1} \mathcal{K})^{-1}$. Initial investigations have shown great promise in block-diagonal preconditioners with one block per body. In this approach, we neglect the hydrodynamic interactions between bodies, but use the mobility approximation developed in this work together with dense linear algebra for each body.

In the marker-based method described in this work, one must adjust the marker spacing to be "neither too small nor too large". The sensitivity of the solver performance and the numerical results to the exact spacing of the markers, which comes from the ill-conditioning of the mobility matrix, is one of the key deficiencies of the marker-based representation inherent to the traditional IB method. Recently, Griffith and Luo have proposed an alternative IB approach that models the deformations and stresses of immersed elastic body using a finite element (FE) representation [38]. In their IB/FE approach, the degrees of freedom associated with $\boldsymbol{\Lambda}$ are represented on an FE mesh that may be coarser than the fluid grid, and the interaction between the fluid grid and body mesh is handled by placing IB markers at the numerical quadrature points of the FE mesh. When such an approach is generalized to rigid bodies, the conditioning of the mobility becomes much less sensitive to the marker spacing. Using a finite-element basis to represent the unknown fluid-body interaction force amounts to applying a *filter* $\boldsymbol{\Psi}$ to the marker-based mobility matrix, which is a well-known and robust technique to regularize ill-conditioned systems. Specifically, in the context of the IB/FE approach, the mobility operator becomes

$$\mathcal{M}_{FE} = \boldsymbol{\Psi}(\mathcal{J}\mathcal{L}^{-1}\mathcal{S})\boldsymbol{\Psi}^T = \boldsymbol{\Psi}\mathcal{M}\boldsymbol{\Psi}^T,$$

where $\boldsymbol{\Psi}$ is a matrix that contains quadrature weights as well as geometric information about the relation between the nodes and quadrature points of the FE mesh. The FE mobility matrix $\mathcal{M}_{FE}$ is still symmetric, but now can be much smaller because the number of unknowns is equal to the number of FE degrees of freedom rather than the number of markers. Even if markers are closely spaced, the filtering of the high-frequency modes performed by representing forces in a smooth FE basis makes the mobility much better conditioned than for marker-based schemes. Furthermore, the mobility matrix, or approximations of it used for preconditioning, will be smaller and thus easier to fit in memory. We also expect the resulting method to be more accurate because the tractions are represented in a smoother basis. We will explore this promising extension of our rigid-body IB methods in future work.

Appendix A: Approximating the mobility in three dimensions

In this appendix we give the details of our empirical fits for the approximations to the functions $f_\beta(r)$ and $g_\beta(r)$ in (12) in three spatial dimensions, following the physics-based constraints discussed in Section IV. To maximize the quality of the fit, we perform separate fits for $\beta \to \infty$ (steady Stokes flow) and finite β. We also make an effort to make the fits change smoothly as β grows towards infinity.

1. *Steady Stokes flow.* Because our numerical computations are done in a periodic domain of length l rather than an unbounded domain, we need to apply a well-known

correction to the Oseen tensor [41; 9],

$$f_\infty(h \ll r \ll l) \approx (8\pi\eta r)^{-1} - 2.84(6\pi\eta l)^{-1}.$$

From the numerical data, we calculated the normalized functions

$$\tilde{f}(x) = (8\pi\eta r)(f_\infty(r) + 2.84/(6\pi\eta l)^{-1}),$$
$$\tilde{g}(x) = (8\pi\eta r)g_\infty(r),$$

(A.1)

where $x = r/h$ is the normalized distance between the markers. As explained previously, we know that $\tilde{f} \approx (8\pi\eta r)/(6\pi\eta a) = 4r/(3a)$ for $x \ll 1$ (in practice, markers are never too close to each other so we only need the self-mobility, i.e., $x = 0$), and that $\tilde{g}$ grows at least quadratically for small x (since $g(0) = 0$). We also know that $\tilde{f} \approx 1$ and $\tilde{g} \approx 1$ for large $r \gg h$. The numerical data for the normalized functions $\tilde{f}(x)$ and $\tilde{g}(x)$ are shown in Figure 2 along with fits to the following semiempirical rational functions,

$$\tilde{f}(x) = \begin{cases} \dfrac{x}{(3a)/(4h) + b_0 x^2} & \text{if } x < 0.8, \\ b_1 x e^{-b_2 x} + \dfrac{b_3 x^2 + x^4}{1 + b_4 x^2 + x^4} & \text{if } x \geq 0.8, \end{cases}$$

$$\tilde{g}(x) = \dfrac{x^3}{b_5 + b_6 x^2 + x^3}.$$

(A.2)

As the figure shows, the numerical data are well described by these formulas, and there is only small scatter of the numerical data around the fit, indicating approximate discrete translational and rotational invariance.[11] We also obtain a reasonable agreement with the RPY tensor (16) approximation; however, as expected, the empirical fits yield a better match to the data.

2. *Nonzero Reynolds numbers.* For finite β, we consider separately the case $r = 0$ (giving the diagonal elements $\widetilde{\mathcal{M}}_{ii}$) and $r > 0.1h$ (giving the off-diagonal elements). For $r = 0$ we use an empirical fit designed to conform to (18),

$$\varphi_0(\beta) = \frac{\eta h f_\beta(0)}{\beta} = \frac{1 + z_1\sqrt{\beta} + z_2\beta}{z_0 + z_3\beta + 6\pi(a/h)z_2\beta^2} \quad \text{and} \quad g_\beta(0) = 0,$$

(A.3)

where $z_1 - z_3$ are coefficients obtained by fitting the numerical data for the self mobility for different β. Note that $z_0 = 2h^3/(3V_m)$ is fixed by the inviscid condition (17). Also note that as $\beta \to \infty$, our fit obeys the correct Stokes limit,

$$\varphi_0(\beta \gg 1) \to \frac{1}{6\pi(a/h)} \cdot \frac{1}{\beta}.$$

[11]Most of the scatter comes from the finite size of the periodic box and can be explained using a known periodic correction to the RPY tensor [11].

We show the empirical fit for $\varphi_0(\beta)$ in Figure 15 in Appendix B.

For nonzero r, we introduce normalized functions $\tilde{f}_\beta$ and $\tilde{g}_\beta$ via

$$f_\beta(r) = -\frac{\beta}{\eta h} \cdot \frac{1}{4\pi x^3} \cdot \tilde{f}_\beta(x), \quad g_\beta(r) = \frac{\beta}{\eta h} \cdot \frac{3}{4\pi x^3} \cdot \tilde{g}_\beta(x), \qquad \text{(A.4)}$$

where $x = r/h$ is the normalized distance, and $\beta/\eta = \Delta t/\rho h^2$. For finite β, we know that $\tilde{f}_\beta(x \gg \sqrt{\beta}) \approx \tilde{g}_\beta(x \gg \sqrt{\beta}) \approx 1$ according to (20). As $\beta \to \infty$, we want to reach the Stokes limit

$$\tilde{f}_\infty(x \gg 1) \to -\frac{x^2}{2\beta}, \quad \tilde{g}_\infty(x \gg 1) \to \frac{x^2}{6\beta}, \qquad \text{(A.5)}$$

and for finite β, we want the viscous contribution to decay as $\exp(-x/(C\sqrt{\beta}))$ for some constant C that should be close to unity. Furthermore, we would like to ensure continuity near the origin with the fit for $r = 0$,

$$\tilde{f}_\beta(x \to 0) \to -4\pi x^3 \varphi_0(\beta).$$

A fitting formula that obeys these conditions that we find to work well for $r > 0.1h$ is

$$\tilde{f}_\beta(x) = \varphi_0(\beta) \frac{-4\pi x^3 + a_4[x^5 - x^7 e^{(-a_3 x/\sqrt{\beta})}/(2\beta)]}{1 + a_0 x + a_1 x^2 + a_2 x^3 + a_4 x^5 \varphi_0(\beta)} + \frac{a_5 x^4 e^{-a_6 x} + a_7 x^4}{1 + a_8 x^3 + a_9 x^5},$$

$$\text{(A.6)}$$

$$\tilde{g}_\beta(x) = \varphi_0(\beta) \frac{b_5[x^5 + x^7 e^{(-b_0 x/\sqrt{\beta})}/(6\beta)]}{1 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4 + b_5 \varphi_0(\beta) x^5},$$

where a_0-a_9 and b_0-b_5 are empirical coefficients. It is important to emphasize that (A.6) was chosen in large part based on empirical trial and error. Many other alternatives exist. For example, one could use the analytical Brinkmanlet (15) for sufficiently large distances and then add short-ranged corrections for nearby markers. Alternatively, one could first subtract the inviscid part $f_0(r)$ and $g_0(r)$ and then fit the viscous contribution only. As discussed above, ideally the fits would be constrained to guarantee an SPD approximate mobility matrix, but this seems difficult to accomplish in practice.

We computed the fitting coefficients in (A.6) for

$$\beta \in \{0, 0.1, 0.25, 0.5, 1, 10, 100, 1000\};$$

the coefficients for other values in the range $0 < \beta < 1000$ are interpolated using linear interpolation, and $\beta > 1000$ is treated using the steady Stokes fitting. We see a good match between the numerical data and our empirical fits in Figure 14, with good translational and rotational invariance (i.e., relatively small scatter of the numerical points around the fits).
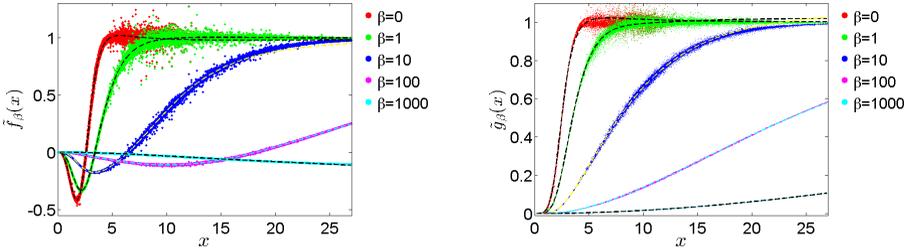
Figure 14. Normalized fitting functions $\tilde{f}_\beta(x)$ (left) and $\tilde{g}_\beta(x)$ (right) at finite β in three dimensions for the 6-point kernel, for different values of the viscous CFL number (see legend). Symbols are numerical data obtained by using a 256^3 periodic fluid grid, and dashed lines show the best fit of the form (A.6).

Appendix B: Approximating the mobility in two dimensions

To construct empirical approximations to the functions $f_\beta(r)$ and $g_\beta(r)$ in (12) in two spatial dimensions, we follow the same approach as we did for three dimensions in Appendix A. Specifically, we first discuss the known asymptotic behavior of these functions at short and large distances, and use this to guide the construction of empirical fitting formulas.

1. *Physical constraints.* In two dimensions, we need to modify (18) to agree with (17) for small β. For $d = 2$, $V_m = c'_V h^2$ and $f_0(0) \sim \beta/\eta$ so that we use the fit

$$f_\beta(0) = \frac{C(\beta)}{\eta} \quad \text{and} \quad g_\beta(0) = 0, \tag{B.1}$$

where $C(\beta)$ has the same asymptotic scaling as in three dimensions and is obtained from empirical fits (see Figure 15). A key difference exists between two and three spatial dimensions in the limit Re $\to 0$. For steady Stokes flow in a square two dimensional periodic domain, the Green's function diverges logarithmically with the system size l. Therefore, it is not possible to write a formula for the asymptotic
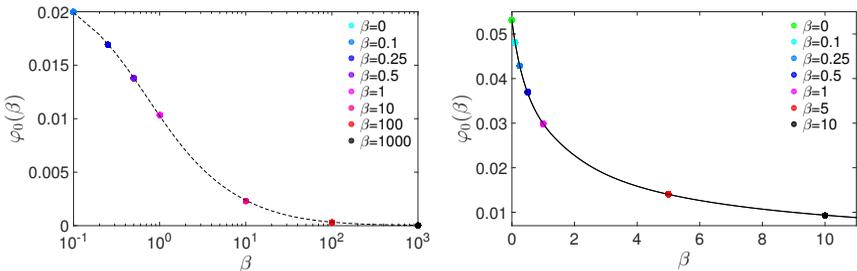
Figure 15. Empirical fit for $\varphi_0(\beta)$ as a function of β for different values of β. Left: three dimensions, 256^3 grid. Right: two dimensions, 512^2 grid.

behavior at large distances for an infinite system. Instead, we must subtract the divergent piece to get a well-defined answer. The standard Green's function for Stokes flow in two dimensions has logarithmic growth at infinity, which suggests that (19) should be replaced by

$$f_\infty(r \gg h) - f_\infty(r = 0) \approx -\frac{\ln(r/h)}{4\pi\eta} \quad \text{and} \quad g_\infty(r \gg h) \approx \frac{1}{4\pi\eta}. \qquad \text{(B.2)}$$

For inviscid flow we should replace (20) by the field of a dipole in two dimensions,

$$f_0(r \gg h) \approx -\frac{\Delta t}{2\pi\rho r^2} \quad \text{and} \quad g_0(r \gg h) \approx \frac{\Delta t}{\pi\rho r^2}. \qquad \text{(B.3)}$$

In two dimensions the solutions of the Brinkman equation (14) are analytically complicated and involve special functions. Even without solving these equations, however, physical scaling suggests that the same physical length scale $h\sqrt{\beta}$ should enter, in particular, the viscous corrections should decay to zero exponentially fast with $h\sqrt{\beta}$.

2. *Empirical fits.* We have used the analytical results above to construct empirical fitting formulas that have the correct asymptotic behavior, as we now explain in more detail.

a. *Steady Stokes flow.* In two dimensions, steady Stokes flow ($\beta \to \infty$) is not well behaved because the Green's function does not decay sufficiently rapidly (Stokes paradox). This makes the mobility an essentially dense matrix that is sensitive to boundary conditions and difficult to approximate. Nevertheless, we have used a periodic system to fit empirical data based on the theory (B.2). The diagonal value $f_\infty(0)$ diverges logarithmically with the system size L for periodic boundaries. Specifically, for a square unit cell of length $l \gg h$, it is known that [41]

$$f_\infty(0) = (4\pi\eta)^{-1} \ln\left(\frac{l}{3.708\, a}\right),$$

and this relation defines the effective hydrodynamic radius of a marker a (note that a/h is a universal value for a given spatial discretization, as it is in three dimensions). Since the precise form depends on boundary conditions and is not known in general, we treat $f_\infty(0)$ as an *input* parameter.

From the numerical data, we calculated the normalized functions

$$\tilde{f}(x) = -(4\pi\eta)(f_\infty(x) - f_\infty(0)) \quad \text{and} \quad \tilde{g}(x) = (4\pi\eta)g_\infty(x), \qquad \text{(B.4)}$$

where $x = r/h$ is the normalized distance between the markers. Observe that from (B.2) we know that $\tilde{f}(x \gg 1) \approx \ln x$ and $\tilde{g}(x \gg 1) \approx 1$. For the normalized
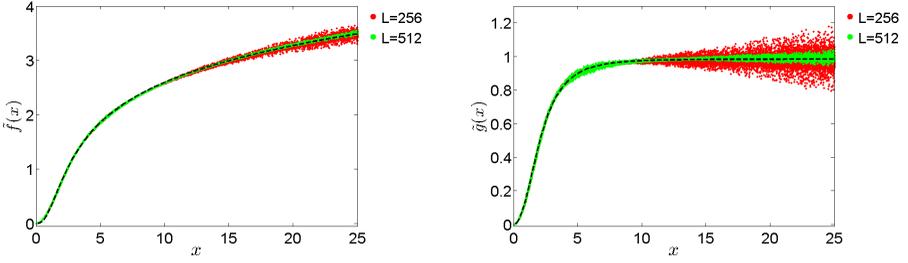
Figure 16. Empirical fits (lines) to numerical data (symbols) for $\tilde{f}(x)$ (left) and $\tilde{g}(x)$ (right), for the 6-point kernel in two dimensions, obtained using a periodic system of either 256^2 or 512^2 grid cells. Observe that both follow the correct asymptotic behavior at large distances, with scatter dominated by boundary effects.

functions, we use the fits

$$\tilde{f}(x) = \frac{a_0 x^2 + a_1 x^3 + a_2 x^3 \ln x}{1 + a_3 x + a_4 x^2 + a_2 x^3},$$

$$\tilde{g}(x) = \frac{b_0 x^2 + b_1 x^3}{1 + b_2 x + b_3 x^2 + b_1 x^3}.$$

(B.5)

Numerical results and empirical fits for $\tilde{f}(x)$ and $\tilde{g}(x)$ are shown in Figure 16. While the numerical data do conform to the theoretical asymptotic behavior, there is substantial scatter for larger distances because of the strong sensitivity to the boundaries.

b. *Nonzero Reynolds numbers.* For $r = 0$, we use a fitting formula in agreement with (B.1),

$$\varphi_0(\beta) = \frac{\eta f_\beta(0)}{\beta} = \frac{z_0 + z_1 \beta^3 \log(\beta)}{1 + z_2 \beta + z_3 \beta^2 + z_4 \beta^4} \quad \text{and} \quad g_\beta(0) = 0, \qquad (B.6)$$

where $z_1 - z_4$ are coefficients (obtained by fitting for each kernel data over a range of β's) and z_0 is fixed from the inviscid condition (17). The empirical fit for $\varphi_0(\beta)$ is shown in Figure 15. Note that for finite β, one must ensure that the system size used to tabulate the values of f_β and g_β is sufficiently large, $l \gg h\sqrt{\beta}$.

For $r > 0.1h$ we introduce normalized functions $\tilde{f}_\beta$ and $\tilde{g}_\beta$ via

$$f_\beta(r) = -\frac{\beta}{2\pi \eta x^2} \cdot \tilde{f}_\beta(x) \quad \text{and} \quad g_\beta(r) = \frac{\beta}{\pi \eta x^2} \cdot \tilde{g}_\beta(x), \qquad (B.7)$$

where in the inviscid case we take $\beta/\eta = \Delta t/\rho h^2$, and $x = r/h$ is the normalized distance. For finite β, we know that $\tilde{f}_\beta(x \gg \sqrt{\beta}) \approx \tilde{g}_\beta(x \gg \sqrt{\beta}) \approx 1$ according to
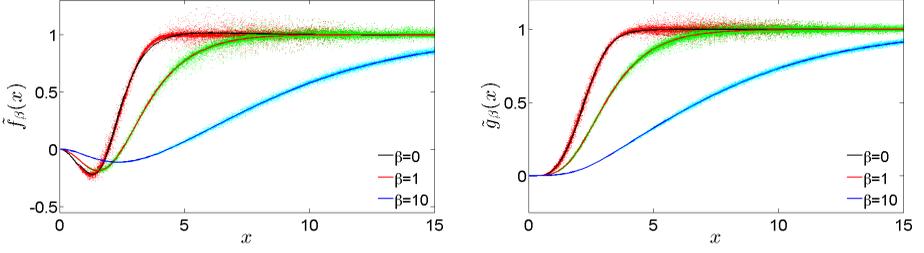
Figure 17. Empirical fitting of $\tilde{f}_\beta(r)$ and $\tilde{g}_\beta(r)$ in two dimensions for different values of β for the 6-point kernel, 512^2 grid.

(B.3). The numerical data is fitted with the empirical fitting functions

$$
\tilde{f}_\beta(x) = \frac{x^3 \ln(x)}{\beta(a_0 + 2x)} e^{-\frac{p_1 x}{\sqrt{\beta}}} + \frac{a_1 x^2 + a_2 x^3 + a_3 x^4}{1 + b_1 x^2 + b_2 x^3 + a_3 x^4},
$$

$$
\tilde{g}_\beta(x) = \frac{x^3}{\beta(c_0 + 4x)} e^{-\frac{p_2 x}{\sqrt{\beta}}} + \frac{x^3}{e^{-p_3 x}(c_1 + c_2 x + c_3 x^2) + x^3},
$$

(B.8)

as shown in Figure 17. Here $a_0 - a_3$, $p_1 - p_3$, $b_1 - b_3$, $c_0 - c_3$ are empirical coefficients, computed by fitting numerical data for β in $\{0, 0.1, 0.25, 0.5, 1.0, 5.0, 10.0\}$. Intermediate values in the range $0 < \beta < 10.0$ are interpolated using linear interpolation, and larger β's are handled using the steady Stokes fit (B.5).

Appendix C: Stokes flow between two concentric spheres

Consider steady Stokes flow around a rigid shell or sphere of radius a, placed in a centered position inside another spherical shell or cavity of radius $b = a/\lambda$. We consider the case when the outer shell is moving with velocity V and the inner shell is at rest, and for simplicity set the viscosity to unity, $\eta = 1$. Brenner [40] gives the drag force on the inner sphere for no slip boundary conditions as

$$
F = -6\pi a V K,
$$

(C.1)

where

$$
K = \frac{1 - \lambda^5}{\alpha} \quad \text{and} \quad \alpha = 1 - \frac{9}{4}\lambda + \frac{5}{2}\lambda^3 - \frac{9}{4}\lambda^5 + \lambda^6.
$$

Define the constants

$$
A = -\frac{15V}{4a^2} \cdot \frac{\lambda^3 - \lambda^5}{\alpha}, \qquad B = \frac{3Va}{2} \cdot \frac{1 - \lambda^5}{\alpha},
$$

$$
C = \frac{V}{2} \cdot \frac{1 + \frac{5}{4}\lambda^3 - \frac{9}{4}\lambda^5}{\alpha}, \qquad D = \frac{Va^3}{4} \cdot \frac{1 - \lambda^3}{\alpha}.
$$

The velocity in the region between the two spherical shells can be obtained from the expressions given by Brenner as

$$v_r = -\cos\theta\left(\frac{A}{5}r^2 - B\frac{1}{r} + 2C + 2D\frac{1}{r^3}\right), \quad v_\theta = \sin\theta\left(\frac{A}{5}r^2 - \frac{B}{2}\frac{1}{r} + 2C - D\frac{1}{r^3}\right),$$

$$v_\phi = 0,$$

and the pressure is

$$\pi = \pi_\infty + \mu B\frac{\cos\theta}{r^2} - 2\mu Ar\cos\theta,$$

where $\pi_\infty = 0$ since we impose that the pressure have mean zero to remove the null mode for pressure. In spherical coordinates, with the symmetry axes aligned with the direction of the flow, the traction on the surface of the inner sphere, which is the jump in the stress across the inner shell, is

$$\boldsymbol{\lambda} = \boldsymbol{\sigma} \cdot \boldsymbol{n} = \mu\cos\theta\left(2Ar - \frac{B}{r^2}\right)\hat{\boldsymbol{r}} + \mu\sin\theta\left(Ar + \frac{B}{r^2}\right)\hat{\boldsymbol{\theta}},$$

where

$$\hat{\boldsymbol{r}} = (\sin\theta\cos\phi,\ \sin\theta\sin\phi,\ \cos\theta),$$

$$\hat{\boldsymbol{\theta}} = (\cos\theta\cos\phi,\ \cos\theta\sin\phi,\ -\sin\theta).$$

Appendix D: Imposing physical boundary conditions

The local averaging and spreading operators have to be modified near physical boundaries, specifically, when the support of the kernel δ_a overlaps with a boundary. A proposal for how to do that has been developed by Yeo and Maxey [83], and an alternative proposal has been developed in the context of the immersed boundary method by Griffith et al. [33]. Here we have chosen to use the former approach because of its simplicity and the fact that it is independent of the kernel, as well as the fact that it ensures that the interpolated velocity strictly vanishes at a no-slip boundary; this ensures that the mobility of a marker is a monotonically decreasing function as it approaches a no-slip boundary. Since the description in [83] is limited to steady Stokes flow and a single no-slip boundary, we give here an algebraic formulation that extends to a variety of boundary conditions; this formulation is implemented in the IBAMR library and used in the examples in this paper in nonperiodic domains.

The basic idea in the handling spreading and interpolation near boundaries is to use the standard IB kernel functions in a domain extended with sufficiently many ghost cells so that the support of all kernels is strictly within the extended domain. For interpolation, we first fill ghost cells and then interpolate as usual using the ghost cell values. For spreading, we take the adjoint operator, which basically

means that we first spread to the extended domain including ghost cells in the usual manner, and then we accumulate the value spread to the ghost cell in the corresponding interior grid point, using the *same* weight (coefficient) that was used when filling ghost cells for the purposes of interpolation.

This process requires a consistent method for filling ghost cells, that is, for extending a (cell-centered or staggered) field $\boldsymbol{u}$ from the interior to the extended domain. In general, this will be an affine linear mapping of the form

$$\boldsymbol{u}_{\text{ext}} = \boldsymbol{E}\boldsymbol{u}_{\text{int}} + \boldsymbol{c},$$

where $\boldsymbol{E}$ is an extension matrix and $\boldsymbol{c}$ encodes inhomogeneous boundary conditions. Let $\mathcal{J}_0$ denote the standard IB interpolation operator that interpolates an *extended* field at a position inside the interior of the domain. The interpolated value in the presence of physical boundary conditions is then given by the affine linear mapping

$$\mathcal{J}_{BC}(\boldsymbol{u}_{\text{int}}) = \mathcal{J}_0 \boldsymbol{u}_{\text{ext}} = \mathcal{J}_0 \boldsymbol{E} \boldsymbol{u}_{\text{int}} + \mathcal{J}_0 \boldsymbol{c}.$$

The corresponding spreading operation is defined to be the adjoint of $\mathcal{J}_{BC}$ for homogeneous boundary conditions, as this ensures energy conservation in the absence of boundary forcing. Specifically, we use

$$\boldsymbol{\mathcal{S}}_{BC} = \boldsymbol{E}^T \mathcal{J}_0^\star = \boldsymbol{E}^T \boldsymbol{\mathcal{S}}_0.$$

The specific form of the extension operator $\boldsymbol{E}$ used in our implementation is based on linear extrapolation to a given ghost point based on the corresponding value in the interior *and* the values at the boundary as specified in the boundary conditions. Specifically, for homogeneous Neumann conditions we do a mirror image $u_{\text{ghost}} = -u_{\text{int}}$, while for homogeneous Dirichlet boundary conditions, such as no slip boundaries, we simply do a mirror inversion $u_{\text{ghost}} = -u_{\text{int}}$. This makes our implementation exactly identical to that proposed by Yeo and Maxey [83] in the context of the FCM method. One can think of this approach to no-slip boundaries as taking an inverted mirror image of the portion of the kernel outside of the domain [83]. Note that the same $\boldsymbol{E}$ is used to implement boundary conditions both in the fluid solver and when interpolating/spreading near boundaries; this greatly simplifies the implementation without lowering the second-order accuracy of the fluid solver [34]. In our implementation, we use simple transpose for spreading which is the adjoint $\boldsymbol{E}^\star$ with respect to the standard inner product.

Acknowledgements

References

[1] L. af Klinteberg and A.-K. Tornberg, *Fast Ewald summation for Stokesian particle suspensions*, Internat. J. Numer. Methods Fluids **76** (2014), no. 10, 669–698. MR 3276094

[2] J. Ainley, S. Durkin, R. Embid, P. Boindala, and R. Cortez, *The method of images for regularized Stokeslets*, J. Comput. Phys. **227** (2008), no. 9, 4600–4616. MR 2009e:76050

[3] S. Ambikasaran and E. Darve, *An O($N \log N$) fast direct solver for partial hierarchically semi-separable matrices: with application to radial basis function interpolation*, J. Sci. Comput. **57** (2013), no. 3, 477–501. MR 3123554 Zbl 1292.65030

[4] A. Aminfar, S. Ambikasaran, and E. Darve, *A fast block low-rank dense solver with applications to finite-element matrices*, J. Comput. Phys. **304** (2016), 170–188. MR 3422408

[5] A. M. Ardekani, S. Dabiri, and R. H. Rangel, *Collision of multi-particle and general shape objects in a viscous fluid*, J. Comput. Phys. **227** (2008), no. 24, 10094–10107. MR 2009k:76135 Zbl 1218.76039

[6] P. J. Atzberger, *Stochastic Eulerian Lagrangian methods for fluid-structure interactions with thermal fluctuations*, J. Comput. Phys. **230** (2011), no. 8, 2821–2837. MR 2012c:74040

[7] S. Balay, W. Gropp, L. McInnes, and B. Smith, *Efficient management of parallelism in object-oriented numerical software libraries*, Modern software tools for scientific computing (E. Arge, A. Bruaset, and H. Langtangen, eds.), Birkhäuser, Boston, MA, 1997, pp. 163–202. Zbl 0882.65154

[8] F. Balboa Usabiaga, I. Pagonabarraga, and R. Delgado-Buscalioni, *Inertial coupling for point particle fluctuating hydrodynamics*, J. Comput. Phys. **235** (2013), 701–722. MR 3017618

[9] F. Balboa Usabiaga, R. Delgado-Buscalioni, B. E. Griffith, and A. Donev, *Inertial coupling method for particles in an incompressible fluctuating fluid*, Comput. Methods Appl. Mech. Eng. **269** (2014), 139–172. MR 3144637 Zbl 1296.76167

[10] Y. Bao, J. Kaye, and C. Peskin, *A gaussian-like immersed boundary kernel with three continuous derivatives and improved translational invariance*, preprint, 2015. arXiv 1505.07529

[11] C. W. J. Beenakker, *Ewald sum of the Rotne–Prager tensor*, J. Chem. Phys. **85** (1986), no. 3, 1581–1582.

[12] S. Benyahia, M. Syamlal, and T. J. O'Brien, *Extension of Hill–Koch–Ladd drag correlation over all ranges of Reynolds number and solids volume fraction*, Powder Technol. **162** (2006), no. 2, 166–174.

[13] A. P. S. Bhalla, R. Bale, B. E. Griffith, and N. A. Patankar, *A unified mathematical framework and an adaptive numerical method for fluid-structure interaction and rigid, deforming, and elastic bodies*, J. Comput. Phys. **250** (2013), 446–476. MR 3079544

[14] _____ , *Fully resolved immersed electrohydrodynamics for particle motion, electrolocation, and self-propulsion*, J. Comput. Phys. **256** (2014), 88–108. MR 3117399

[15] S. Bhattacharya, J. Bławzdziewicz, and E. Wajnryb, *Hydrodynamic interactions of spherical particles in suspensions confined between two planar walls*, J. Fluid Mech. **541** (2005), 263–292. MR 2262651

[16] T. T. Bringley and C. S. Peskin, *Validation of a simple method for representing spheres and slender bodies in an immersed boundary method for Stokes flow on an unbounded domain*, J. Comput. Phys. **227** (2008), no. 11, 5397–5425. MR 2009j:76076 Zbl 1220.76026

[17] H. C. Brinkman, *A calculation of the viscous force exerted by a flowing fluid on a dense swarm of particles*, Appl. Sci. Res. **1** (1949), no. 1, 27–34. Zbl 0041.54204

[18] M. Cai, A. Nonaka, J. B. Bell, B. E. Griffith, and A. Donev, *Efficient variable-coefficient finite-volume Stokes solvers*, Commun. Comput. Phys. **16** (2014), no. 5, 1263–1297. MR 3256967

[19] H. D. Ceniceros and J. E. Fisher, *A fast, robust, and non-stiff immersed boundary method*, J. Comput. Phys. **230** (2011), no. 12, 5133–5153. MR 2012b:65105

[20] B. Cichocki and K. Hinsen, *Stokes drag on conglomerates of spheres*, Phys. Fluids **7** (1995), no. 2, 285–291. Zbl 0832.76017

[21] R. Cortez, *The method of regularized Stokeslets*, SIAM J. Sci. Comput. **23** (2001), no. 4, 1204–1225. MR 2002k:76102 Zbl 1064.76080

[22] R. Cortez, B. Cummins, K. Leiderman, and D. Varela, *Computation of three-dimensional Brinkman flows using regularized methods*, J. Comput. Phys. **229** (2010), no. 20, 7609–7624. MR 2011f:76127

[23] R. Cortez, L. Fauci, and A. Medovikov, *The method of regularized Stokeslets in three dimensions: analysis, validation, and application to helical swimming*, Phys. Fluids **17** (2005), no. 3, 031504. MR 2005k:76031 Zbl 1187.76105

[24] P. Coulier, H. Pouransari, and E. Darve, *The inverse fast multipole method: using a fast approximate direct solver as a preconditioner for dense linear systems*, preprint, 2015. arXiv 1508.01835

[25] O. M. Curet, I. K. AlAli, M. A. MacIver, and N. A. Patankar, *A versatile implicit iterative approach for fully resolved simulation of self-propulsion*, Comput. Methods Appl. Mech. Engng. **199** (2010), no. 37-40, 2417–2424. Zbl 1231.76176

[26] J. G. de la Torre, M. L. Huertas, and B. Carrasco, *Calculation of hydrodynamic properties of globular proteins from their atomic-level structure*, Biophys. J. **78** (2000), no. 2, 719–730.

[27] S. Delong, F. Balboa Usabiaga, R. Delgado-Buscalioni, B. E. Griffith, and A. Donev, *Brownian dynamics without Green's functions*, J. Chem. Phys. **140** (2014), no. 13, 134110.

[28] L. Durlofsky and J. F. Brady, *Analysis of the Brinkman equation as a model for flow in porous media*, Phys. Fluids **30** (1987), no. 11, 3329–3341. Zbl 0636.76098

[29] H. Elman, V. E. Howle, J. Shadid, R. Shuttleworth, and R. Tuminaro, *Block preconditioners based on approximate commutators*, SIAM J. Sci. Comput. **27** (2006), no. 5, 1651–1668. MR 2006j:65084 Zbl 1100.65042

[30] H. C. Elman, D. J. Silvester, and A. J. Wathen, *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics*, 2nd ed., Oxford University Press, 2014. MR 3235759 Zbl 1304.76002

[31] M. W. Gee, U. Küttler, and W. A. Wall, *Truly monolithic algebraic multigrid for fluid-structure interaction*, Int. J. Numer. Methods Eng. **85** (2011), no. 8, 987–1016. MR 2011m:65065 Zbl 1217.74121

[32] S. Ghose and R. Adhikari, *Irreducible representations of oscillatory and swirling flows in active soft matter*, Phys. Rev. Lett. **112** (2014), 118102.

[33] B. E. Griffith, X. Luo, D. M. McQueen, and C. S. Peskin, *Simulating the fluid dynamics of natural and prosthetic heart valves using the immersed boundary method*, Int. J. Appl. Mech. **01** (2009), no. 01, 137–177.

[34] B. E. Griffith, *An accurate and efficient method for the incompressible Navier–Stokes equations using the projection method as a preconditioner*, J. Comput. Phys. **228** (2009), no. 20, 7565–7595. MR 2561832

[35] ———, *Immersed boundary model of aortic heart valve dynamics with physiological driving and loading conditions*, Int. J. Numer. Methods Biomed. Eng. **28** (2012), no. 3, 317–345. MR 2910281 Zbl 1243.92017

[36] ———, *On the volume conservation of the immersed boundary method*, Commun. Comput. Phys. **12** (2012), no. 2, 401–432. MR 2897145

[37] B. E. Griffith, R. D. Hornung, D. M. McQueen, and C. S. Peskin, *An adaptive, formally second order accurate version of the immersed boundary method*, J. Comput. Phys. **223** (2007), no. 1, 10–49. MR 2008e:76133 Zbl 1163.76041

[38] B. E. Griffith and X. Luo, *Hybrid finite difference/finite element version of the immersed boundary method*, (2016), submitted to *Int. J. Numer. Methods Biomed. Eng.*

[39] R. D. Guy and B. Philip, *A multigrid method for a model of the implicit immersed boundary equations*, Commun. Comput. Phys. **12** (2012), no. 2, 378–400. MR 2897144

[40] J. Happel and H. Brenner, *Low Reynolds number hydrodynamics: with special applications to particulate media*, Mechanics of fluids and transport processes, no. 1, Springer, 1983. Zbl 0612.76032

[41] H. Hasimoto, *On the periodic fundamental solutions of the Stokes' equations and their application to viscous flow past a cubic array of spheres*, J. Fluid Mech. **5** (1959), 317–328. MR 21 #1078

[42] M. Heil, A. L. Hazel, and J. Boyle, *Solvers for large-displacement fluid–structure interaction problems: segregated versus monolithic approaches*, Comput. Mech. **43** (2008), no. 1, 91–101.

[43] J. P. Hernández-Ortiz, J. J. de Pablo, and M. D. Graham, *Fast computation of many-particle hydrodynamic and electrostatic interactions in a confined geometry*, Phys. Rev. Lett. **98** (2007), 140602.

[44] R. J. Hill, D. L. Koch, and A. J. C. Ladd, *The first effects of fluid inertia on flows in ordered and random arrays of spheres*, J. Fluid Mech. **448** (2001), 213–241. MR 2002i:76111 Zbl 1045.76036

[45] ———, *Moderate-Reynolds-number flows in ordered and random arrays of spheres*, J. Fluid Mech. **448** (2001), 243–278. MR 2002i:76112 Zbl 0997.76068

[46] K. Hinsen, *HYDROLIB: a library for the evaluation of hydrodynamic interactions in colloidal suspensions*, Comput. Phys. Commun. **88** (1995), no. 2-3, 327–340.

[47] K. L. Ho and L. Greengard, *A fast direct solver for structured linear systems by recursive skeletonization*, SIAM J. Sci. Comput. **34** (2012), no. 5, A2507–A2532. MR 3023714 Zbl 1259.65062

[48] R. M. Jendrejack, D. C. Schwartz, M. D. Graham, and J. J. de Pablo, *Effect of confinement on DNA dynamics in microfluidic devices*, J. Chem. Phys. **119** (2003), no. 2, 1165–1173.

[49] S. Jiang, Z. Liang, and J. Huang, *A fast algorithm for Brownian dynamics simulation with hydrodynamic interactions*, Math. Comput. **82** (2013), no. 283, 1631–1645. MR 3042579

[50] E. E. Keaveny, *Fluctuating force-coupling method for simulations of colloidal suspensions*, J. Comput. Phys. **269** (2014), 61–79. MR 3197680

[51] R. Kekre, J. Butler, and A. Ladd, *Comparison of lattice-boltzmann and brownian-dynamics simulations of polymer migration in confined flows*, Phys. Rev. E **82** (2010), 011802.

[52] D. L. Koch and A. J. C. Ladd, *Moderate Reynolds number flows through periodic and random arrays of aligned cylinders*, J. Fluid Mech. **349** (1997), 31–66. MR 98k:76030 Zbl 0912.76014

[53] D. L. Koch and G. Subramanian, *Collective hydrodynamics of swimming microorganisms: living fluids*, Annu. Rev. Fluid Mech. (2011), no. 43, 637–659. MR 2012f:76164 Zbl 1299.76320

[54] A. J. C. Ladd, *Numerical simulations of particulate suspensions via a discretized Boltzmann equation, II: Numerical results*, J. Fluid Mech. **271** (1994), 311–339. MR 95g:76040

[55] A. J. C. Ladd, R. Kekre, and J. E. Butler, *Comparison of the static and dynamic properties of a semiflexible polymer using lattice boltzmann and brownian-dynamics simulations*, Phys. Rev. E **80** (2009), 036704.

[56] K. Leiderman, E. L. Bouzarth, R. Cortez, and A. T. Layton, *A regularization method for the numerical solution of periodic Stokes flow*, J. Comput. Phys. **236** (2013), 187–202. MR 3020052 Zbl 1286.76043

[57] Z. Liang, Z. Gimbutas, L. Greengard, J. Huang, and S. Jiang, *A fast multipole method for the Rotne–Prager–Yamakawa tensor and its applications*, J. Comput. Phys. **234** (2013), 133–139. MR 2999771 Zbl 1284.76318

[58] S. Lomholt and M. R. Maxey, *Force-coupling method for particulate two-phase flow: Stokes flow*, J. Comput. Phys. **184** (2003), no. 2, 381–405. Zbl 1047.76100

[59] O. Marin, K. Gustavsson, and A.-K. Tornberg, *A highly accurate boundary treatment for confined Stokes flow*, Comput. Fluids **66** (2012), 215–230.

[60] E. P. Newren, A. L. Fogelson, R. D. Guy, and R. M. Kirby, *A comparison of implicit solvers for the immersed boundary equations*, Comput. Methods Appl. Mech. Eng. **197** (2008), no. 25-28, 2290–2304. MR 2009f:74031 Zbl 1158.76409

[61] A. Ortega, D. Amorós, and J. G. de La Torre, *Prediction of hydrodynamic and other solution properties of rigid proteins from atomic-and residue-level models*, Biophys. J. **101** (2011), no. 4, 892–898.

[62] A. Pal Singh Bhalla, B. Griffith, N. Patankar, and A. Donev, *A minimally-resolved immersed boundary model for reaction-diffusion problems*, J. Chem. Phys. **139** (2013), no. 21, 214112.

[63] S. Patankar, *Numerical heat transfer and fluid flow*, CRC Press, Boca Raton, FL, 1980.

[64] C. S. Peskin, *The immersed boundary method*, Acta Numer. **11** (2002), 479–517. MR 2004h: 74029 Zbl 1123.74309

[65] C. S. Peskin and B. F. Printz, *Improved volume conservation in the computation of flows with immersed elastic boundaries*, J. Comput. Phys. **105** (1993), no. 1, 33–46. MR 93k:76081 Zbl 0762.92011

[66] C. Pozrikidis, *Boundary integral and singularity methods for linearized viscous flow*, Cambridge Texts in Applied Mathematics, no. 8, Cambridge University Press, 1992. MR 93a:76027 Zbl 0772.76005

[67] A. M. Roma, C. S. Peskin, and M. J. Berger, *An adaptive version of the immersed boundary method*, J. Comput. Phys. **153** (1999), no. 2, 509–534. MR 2000e:76100 Zbl 0953.76069

[68] J. Rotne and S. Prager, *Variational treatment of hydrodynamic interaction in polymers*, J. Chem. Phys. **50** (1969), no. 11, 4831–4837.

[69] A. Sierou and J. F. Brady, *Accelerated stokesian dynamics simulations*, J. Fluid Mech. **448** (2001), 115–146.

[70] D. B. Stein, R. D. Guy, and B. Thomases, *Immersed boundary smooth extension: A high-order method for solving PDE on arbitrary smooth domains using Fourier spectral methods*, J. Comput. Phys. **304** (2016), 252–274. MR 3422411

[71] S.-W. Su, M.-C. Lai, and C.-A. Lin, *An immersed boundary technique for simulating complex flows with rigid boundary*, Comput. Fluids **36** (2007), no. 2, 313–324. Zbl 1177.76299

[72] J. W. Swan and J. F. Brady, *Simulation of hydrodynamically interacting particles near a no-slip boundary*, Phys. Fluids **19** (2007), no. 11, 113306. Zbl 1182.76735

[73] ———, *Particle motion between parallel walls: Hydrodynamics and simulation*, Phys. Fluids **22** (2010), no. 10, 103301.

[74] ———, *The hydrodynamics of confined dispersions*, J. Fluid Mech. **687** (2011), 254–299. MR 2855929 Zbl 1241.76406

[75] J. W. Swan, J. F. Brady, R. S. Moore, and ChE 174, *Modeling hydrodynamic self-propulsion with Stokesian dynamics, or teaching Stokesian dynamics to swim*, Phys. Fluids **23** (2011), no. 7, 071901.

[76] K. Taira and T. Colonius, *The immersed boundary method: a projection approach*, J. Comput. Phys. **225** (2007), no. 2, 2118–2137. MR 2009b:76036

[77] J. M. Teran and C. S. Peskin, *Tether force constraints in Stokes flow by the immersed boundary method on a periodic domain*, SIAM J. Sci. Comput. **31** (2009), no. 5, 3404–3416. MR 2010h:76056 Zbl 1197.76035

[78] M. Uhlmann, *An immersed boundary method with direct forcing for the simulation of particulate flows*, J. Comput. Phys. **209** (2005), no. 2, 448–476. MR 2151992 Zbl 1138.76398

[79] A. Vázquez-Quesada, F. Balboa Usabiaga, and R. Delgado-Buscalioni, *A multiblob approach to colloidal hydrodynamics with inherent lubrication*, J. Chem. Phys. **141** (2014), no. 20, 204102.

[80] E. Wajnryb, K. A. Mizerski, P. J. Zuk, and P. Szymczak, *Generalization of the Rotne–Prager–Yamakawa mobility and shear disturbance tensors*, J. Fluid Mech. **731** (2013), R3. MR 3138037 Zbl 1294.76262

[81] H. A. R. Williams, L. J. Fauci, and D. P. Gaver, III, *Evaluation of interfacial fluid dynamical stresses using the immersed boundary method*, Discrete Contin. Dyn. Syst. Ser. B **11** (2009), no. 2, 519–540. MR 2010g:65125 Zbl 1277.76132

[82] X. Yang, X. Zhang, Z. Li, and G.-W. He, *A smoothing technique for discrete delta functions with application to immersed boundary method in moving boundary simulations*, J. Comput. Phys. **228** (2009), no. 20, 7821–7836. MR 2010j:76099

[83] K. Yeo and M. R. Maxey, *Dynamics of concentrated suspensions of non-colloidal particles in Couette flow*, J. Fluid Mech. **649** (2010), 205–231. MR 2011a:76094 Zbl 1189.76677

[84] C. Zhang, R. D. Guy, B. Mulloney, Q. Zhang, and T. J. Lewis, *Neural mechanism of optimal limb coordination in crustacean swimming*, Proc. Nat. Acad. Sci. USA **111** (2014), no. 38, 13840–13845.

[85] Q. Zhang, R. D. Guy, and B. Philip, *A projection preconditioner for solving the implicit immersed boundary equations*, Numer. Math. Theory Methods Appl. **7** (2014), no. 4, 473–498. MR 3280968

[86] Y. Zhang, J. J. de Pablo, and M. D. Graham, *An immersed boundary method for Brownian dynamics simulation of polymers in complex geometries: Application to DNA flowing through a nanoslit with embedded nanopits*, J. Chem. Phys. **136** (2012), no. 1, 014901.

BAKYTZHAN KALLEMOV: kallemov@cims.nyu.edu
Courant Institute of Mathematical Sciences, New York University, New York, NY 10012,
United States

AMNEET PAL SINGH BHALLA: amneetb@live.unc.edu
Department of Mathematics, University of North Carolina, Chapel Hill, NC 27599, United States

BOYCE E. GRIFFITH: boyceg@email.unc.edu
Departments of Mathematics and Biomedical Engineering, University of North Carolina,
Chapel Hill, NC 27599, United States

ALEKSANDAR DONEV: donev@courant.nyu.edu
Courant Institute of Mathematical Sciences, New York University, New York, NY 10012,
United States

Guidelines for Authors

Authors may submit manuscripts in PDF format on-line at the Submission page at msp.org/camcos.

Originality. Submission of a manuscript acknowledges that the manuscript is original and and is not, in whole or in part, published or under consideration for publication elsewhere. It is understood also that the manuscript will not be submitted elsewhere while under consideration for publication in this journal.

Language. Articles in CAMCoS are usually in English, but articles written in other languages are welcome.

Required items. A brief abstract of about 150 words or less must be included. It should be self-contained and not make any reference to the bibliography. If the article is not in English, two versions of the abstract must be included, one in the language of the article and one in English. Also required are keywords and subject classifications for the article, and, for each author, postal address, affiliation (if appropriate), and email address.

Format. Authors are encouraged to use LaTeX but submissions in other varieties of TeX, and exceptionally in other formats, are acceptable. Initial uploads should be in PDF format; after the refereeing process we will ask you to submit all source material.

References. Bibliographical references should be complete, including article titles and page ranges. All references in the bibliography should be cited in the text. The use of BibTeX is preferred but not required. Tags will be converted to the house format, however, for submission you may use the format of your choice. Links will be provided to all literature with known web locations and authors are encouraged to provide their own links in addition to those supplied in the editorial process.

Figures. Figures must be of publication quality. After acceptance, you will need to submit the original source files in vector graphics format for all diagrams in your manuscript: vector EPS or vector PDF files are the most useful.

Most drawing and graphing packages (Mathematica, Adobe Illustrator, Corel Draw, MATLAB, etc.) allow the user to save files in one of these formats. Make sure that what you are saving is vector graphics and not a bitmap. If you need help, please write to graphics@msp.org with details about how your graphics were generated.

White space. Forced line breaks or page breaks should not be inserted in the document. There is no point in your trying to optimize line and page breaks in the original manuscript. The manuscript will be reformatted to use the journal's preferred fonts and layout.

Proofs. Page proofs will be made available to authors (or to the designated corresponding author) at a Web site in PDF format. Failure to acknowledge the receipt of proofs or to return corrections within the requested deadline may cause publication to be postponed.