PROJECTION BASED MODEL REDUCTION FOR THE IMMERSED BOUNDARY METHOD

A Preprint

Yushuang Luo
Department of Mathematics
The Pennsylvania State University, University Park, PA 16802, USA
yz155@psu.edu

Xiantao Li
Department of Mathematics,
The Pennsylvania State University, University Park, PA 16802, USA
xx112@psu.edu

Wenrui Hao
Department of Mathematics
The Pennsylvania State University, University Park, PA 16802, USA
wxh64@psu.edu

October 13, 2021

ABSTRACT

Fluid-structure interactions are central to many bio-molecular processes, and they impose a great challenge for computational and modeling methods. In this paper, we consider the immersed boundary method (IBM) for biofluid systems, and to alleviate the computational cost, we apply reduced-order techniques to eliminate the degrees of freedom associated with the large number of fluid variables. We show how reduced models can be derived using Petrov-Galerkin projection and subspaces that maintain the incompressibility condition. More importantly, the reduced-order model is shown to preserve the Lyapunov stability. We also address the practical issue of computing coefficient matrices in the reduced-order model using an interpolation technique. The efficiency and robustness of the proposed formulation are examined with test examples from various applications.

Keywords model reduction, fluid-structure interaction, immersed boundary method

1 Introduction

Biofluid dynamics, the study of cellular movement in biological fluid flow, is essential for understanding how the cellular behavior changes within living tissues [1]. With the rapid development of scientific computing algorithms, mathematical modeling and numerical simulations have become an indispensable approach for studying biofluid dynamics. Specifically, the interaction between cell structures and the surrounding fluid flow is of the utmost importance. Mathematically speaking, this belongs to a large class of problems known as the fluid-structure interactions (FSI), often described by coupling the incompressible Navier–Stokes equations with solid equations. A variety of computational and modeling techniques have been developed for FSIs, and they have been successfully implemented in studying (among many other applications) biology and biomedical diseases [2, 3, 4, 5, 6, 7, 8, 9]. In order to numerically solve FSI problems, several numerical methods have been developed to represent/track the interface movement explicitly, such as the boundary element method (BEM) [10, 11, 12], the IBM [13, 14, 15, 16], the immersed interface method (IIM) [17], the fictitious domain method (FDM) [18, 19], and the front tracking method (FTM) [20, 21]. Another alternative approach to solve the FSI problem is to capture the interface dynamics implicitly by evolving a
The Dirac delta function. The dynamics of the fluid is described in terms of the velocity defined by a set of differential equations involving a mixture of Eulerian and Lagrangian descriptions, linked by the IBM is intended for the computer simulation of FSI, especially in biological fluid dynamics. It is mathematically Full-order Model compare both full-order and reduced-order models in Section 4; then the conclusion is drawn in Section 5. The remaining part of the paper is organized as follows: in Section 2, we introduce the full-order model (FOM) in IBM setup; in Section 3, we formulate the reduced-order model (ROM); several numerical examples are used to To construct specific reduced models that do not involve the fluid dynamics explicitly, we first construct subspaces that preserve the incompressibility condition, followed by a Petrov-Galerkin projection. We show that the choice of the subspaces ensures certain interpolation conditions on the underlying transfer function. An important departure from standard reduced-order problems is that in IBM, the structure is also evolving continuously. As a result, the subspaces are varying in time. This poses some challenges as the coefficient matrices of the reduced models need to be updated frequently. To circumvent this issue, we observe the connection between the those matrices and the Green’s function of the Laplace equation. More specifically, the entries of those matrices are tied to the nodal points on the structure. When the two points are far apart, the corresponding entry can be well approximated by the Green’s function. On the other hand, for points that are within some cut-off distance, the computation can be done in advance, and then in the simulation, those entries can be computed by interpolation. We show that such a strategy avoids repetitive computation of those coefficient matrices and it can speed up the computation considerably.

The purpose of this paper is to explore an alternative to reduce the computational cost using reduced-order techniques, which have been applied to a wide variety of problems in science and engineering [26][27][28][29]. Reduced-order modeling is concerned with large-dimensional dynamical systems with low-dimensional input and output, and the main objective is to construct reduced models that can approximate the mapping from the input directly to the output. The present FSI problem will be formulated as a reduced-order problem, where the input is the force exerted from the structure and the output is the local velocity of the structure. As a proof-of-concept, we utilize the conventional IBM model [13]. Specifically, incompressible unsteady Stokes flows are considered, together with the no-slip interface condition enforced on the immersed structure. But it is also important to point out that there have been many extensions of the original IBM framework with different treatments for the Lagrangian equations of motion or the fluid dynamics [30][31][32], and reduced-order modeling can be considered in those settings as well. Our starting point is a semi-discrete representation of the IBM model, so that the dynamics of fluid and structure motion can be expressed as coupled ODEs, which can then be placed in the reduced-order modeling framework. Then we derive the effective mapping from the structure force to the local velocity, which completely eliminates the fluid variables.

The present FSI problem will be formulated as a reduced-order problem, where the input is the force exerted from the structure and the output is the local velocity of the structure. As a proof-of-concept, we utilize the conventional IBM model [13]. Specifically, incompressible unsteady Stokes flows are considered, together with the no-slip interface condition enforced on the immersed structure. But it is also important to point out that there have been many extensions of the original IBM framework with different treatments for the Lagrangian equations of motion or the fluid dynamics [30][31][32], and reduced-order modeling can be considered in those settings as well. Our starting point is a semi-discrete representation of the IBM model, so that the dynamics of fluid and structure motion can be expressed as coupled ODEs, which can then be placed in the reduced-order modeling framework. Then we derive the effective mapping from the structure force to the local velocity, which completely eliminates the fluid variables.

The remaining part of the paper is organized as follows: in Section 2, we introduce the full-order model (FOM) in the IBM setup; in Section 3, we formulate the reduced-order model (ROM); several numerical examples are used to compare both full-order and reduced-order models in Section 4; then the conclusion is drawn in Section 5.

2 Full-order Model

In this section we briefly review the mathematical formulation of the IBM and derive its semi-discrete representation, which will serve as the full-order model (FOM).

2.1 Mathematical formulation of the IBM

The IBM is intended for the computer simulation of FSI, especially in biological fluid dynamics. It is mathematically defined by a set of differential equations involving a mixture of Eulerian and Lagrangian descriptions, linked by the Dirac delta function. The dynamics of the fluid is described in terms of the velocity \( u(x, t) \) and the pressure \( p(x, t) \) on an Eulerian coordinate for \( x \in \Omega \), where \( \Omega \subset \mathbb{R}^d \), \( d = 2 \) or \( 3 \), represents the fluid domain. The immersed structures, on the other hand, are handled in a Lagrangian coordinate as a parametric curve or surface \( X(s, t) \). Specifically, \( X(s, t) \) represents the position at time \( t \) in Cartesian coordinates of the structure point labeled by \( s \in \Gamma \), where \( \Gamma \subset \mathbb{R}^{d-1} \) is the parameter space. In this work, we focus on two-dimensional flows where the structure is described as a parametric curve. In this case \( x \in \Omega \subset \mathbb{R}^2 \) and \( s \) is a scalar parameter. The formulation is mostly algebraic. Therefore the extension to high dimensional cases is straightforward. Assuming constant density, the time-dependent Stokes equation is used to model the incompressible flow

\[
\rho \frac{\partial u}{\partial t} = -\nabla p + \mu \nabla^2 u + f, \quad (1)
\]

\[
\nabla \cdot u = 0, \quad (2)
\]
where \( \rho \) and \( \mu \) are the fluid density and viscosity, respectively. The body force \( f \) exerted by the structure on the fluid is defined as

\[
f(x,t) = \int \Gamma F(s,t) \delta(x - X(s,t)) \, ds,
\]

where \( \delta(x) \) is the Dirac delta function. In addition, \( F(s,t) \) denotes the force density on the immersed structure, defined as

\[
F(s,t) = F[X(s,t)],
\]

where \( F \) is a functional of the IBM configuration. Spring forces, bending resistance or any other type of behavior (area and volume conservation constraints) can be built into this functional to embody the physics of the immersed structure under different circumstances [19, 33]. We give detailed description of the force density in section 4 in the numerical examples for various systems.

Assuming an over-damped structure, the immersed boundary must move with the local fluid velocity:

\[
\frac{\partial X(s,t)}{\partial t} = u(X(s,t),t) = \int_\Omega u(x,t) \delta(x - X(s,t)) \, dx.
\]

This last equation is nothing other than the no-slip condition written as a delta function convolution.

### 2.2 The semi-discrete equations

To derive a semi-discrete representation of the IBM, we use the finite difference discretization [13]. Other numerical methods can also be applied to discretize the IBM, e.g., the finite element method [34] and the finite volume method [35], in which the state space consists of nodal values.

In this work, fluid variables are discretized on a uniform staggered Eulerian grid, denoted \( \Gamma_h \); and the structures are discretized on an independent Lagrangian grid, denoted by \( \Gamma_h \) (Fig 1). The Eulerian grid points are of the form \( x = jh \), where \( j = (j_1, j_2) \) is a two-dimensional vector with integer components and \( h \) is the Eulerian grid size. The Lagrangian grid is a set of \( s \) of the form \( k\Delta s \), where \( k \) has integer components. The following restriction is imposed to avoid leak [13],

\[
|X(s + \Delta s,t) - X(s,t)| < \frac{h}{2},
\]

for all \( s \). First, the semi-discrete equations for (1)-(2) form a system of linear differential-algebraic equations (DAEs)

\[
\rho u_h(t) = -N p_h(t) + \mu G u_h(t) + f_h(t),
\]

\[
0 = M u_h(t),
\]

Figure 1: The fluid field variables are defined on a regular staggered grid. The structure variables are defined on an Lagrangian grid.
where
\[
\mathbf{u}_h(t) = \begin{bmatrix}
\vdots \\
\vdots \\
\vdots
\end{bmatrix} \in \mathbb{R}^{n_v}, \quad \mathbf{p}_h(t) = \begin{bmatrix}
\vdots \\
\vdots \\
\vdots
\end{bmatrix} \in \mathbb{R}^{n_p}, \quad \mathbf{f}_h(t) = \begin{bmatrix}
\vdots \\
\vdots \\
\vdots
\end{bmatrix} \in \mathbb{R}^{n_u}, \quad \forall x \in \Omega_h,
\]
are vectors of discrete velocity field, pressure and body force, respectively. \(G \in \mathbb{R}^{n_v \times n_u}\) is the discrete Laplace operator. Matrices \(N \in \mathbb{R}^{n_u \times n_p}\) and \(M \in \mathbb{R}^{n_p \times n_u}\) are the discrete gradient and divergence operators, respectively.

Second, the integrals in (3) and (5) are replaced by the following sums over the appropriate grid points,
\[
f_h(x, t) = \sum_{s \in \Gamma_h} F_h(s, t) \delta_r(x - X_h(s, t)) \Delta s \quad \forall x \in \Omega_h, \tag{7}
\]
\[
\dot{X}_h(s, t) = \sum_{x \in \Omega_h} u_h(x, t) \delta_r(x - X_h(s, t)) h^2 \quad \forall s \in \Gamma_h, \tag{8}
\]
where \(F_h(s, t)\) is the discrete Lagrangian force density associated with the structure point labeled \(s\), obtained by discretizing (4). In addition, a function \(\delta_r(x)\) that is nonsingular for each \(r\) but approaches \(\delta(x)\) as \(r \to 0\) is needed. There are many ways to construct such \(\delta_r\). We choose a radially symmetric function with compact support as follows [36],
\[
\delta_r(x) = \begin{cases}
C_r \left[1 + \frac{|x|^2}{4} (2|x|^2 - 3r) \right] & |x| \leq r, \\
0 & |x| > r,
\end{cases}
\]
where the normalizing constant \(C_r = \frac{10}{9r} r^{-2}\) depends on \(r\) and the space dimension (\(C_r = \frac{12}{5r} r^{-3}\) in 3D). For computational efficiency, we choose \(r = 2h\) in all our numerical experiments, as suggested for IBM [13].

Meanwhile, equations (7) and (8) can be put into matrix-vector form:
\[
f_h(t) = B F_h(t), \tag{9}
\]
\[
\dot{X}_h(t) = h^2 B^T u_h(t),
\]
where
\[
F_h(t) = \begin{bmatrix}
\vdots \\
\vdots \\
\vdots
\end{bmatrix} \in \mathbb{R}^{n_v}, \quad X_h(t) = \begin{bmatrix}
\vdots \\
\vdots \\
\vdots
\end{bmatrix} \in \mathbb{R}^{n_v}, \quad \forall s \in \Gamma_h,
\]
are the discrete representations of the structure position and the Lagrangian force density. Using natural arrangement of the fluid variables, \(B \in \mathbb{R}^{n_u \times n_v}\) can be constructed as a block matrix
\[
B = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} \begin{bmatrix} T^1 & 0 \\ 0 & T^2 \end{bmatrix} \begin{bmatrix} B_1 & 0 & 0 \\ 0 & B_2 & 0 \end{bmatrix} \quad \text{in 3D}.
\]

A column of each \(B_k\) consists of evaluations of \(\delta_r(x - X(s, t))\) for a fixed \(X(s, t)\) on grid points \(x\) that store one component of the fluid velocity variables. For example, the \(i, j\)-entry of \(B_1\) is \(\delta_r(x_i - X(s_j, t))\), where \(x_i\) is the grid point that stores the \(i\)th fluid velocity in the \(x\)-direction. Note that \(B_k\)'s are not identical because they correspond to different Eulerian grid points \(x\). For example in Fig[1] \(x\) in \(B_1\) are the points marked by filled triangles, while \(x\) in \(B_2\) are marked by stars. We also point out that \(B\) is time dependent due to its dependence on \(X_h(t)\).

Lastly, by substituting (9) into (6) for \(f_h(t)\), we get the following DAE system which we shall refer to as the FOM
\[
\rho \dot{u}_h(t) = - N p_h(t) + \mu G u_h(t) + B F_h(t), \tag{10}
\]
\[
0 = M u_h(t), \tag{11}
\]
\[
\dot{X}_h(t) = h^2 B^T u_h(t). \tag{12}
\]
In general, the number of structure variables is much less than the number of fluid variables, i.e., \(n_s \ll n_v\). In fact, \(\delta_r\) having compact support means only a small fraction of the Eulerian grid points are directly interacting with the structure. If one is only interested in the motion of the structure, i.e. \(X_h(t)\), solving the system (10)-(12) becomes a reduced-order problem [37], where \(F_h(t)\) is the low-dimensional input and \(X_h(t)\) is the low-dimensional output.
3 Reduced-order Model

To construct our ROM, we start by transforming the DAE system \((10)-(12)\) to a coupled ODE system. Multiplying \((10)\) by \(M\) to the left and using \((11)\), we rewrite \((10)\) as:

\[
0 = -MNp_h(t) + \mu MGu_h(t) + MBF_h(t).
\]

Assuming \(MN\) is nonsingular, it follows that

\[
p_h(t) = (MN)^{-1} (\mu MGu_h(t) + MBF_h(t)).
\]

Substituting \((13)\) into \((10)\) for \(p_h(t)\), one gets

\[
\rho \dot{u}_h(t) = \mu QG u_h(t) + QB F_h(t),
\]

where

\[
Q = I - N(MN)^{-1} M
\]

is an oblique projection. It is worth emphasizing here that the discrete gradient operator \(N\) and the discrete divergence operator \(M\) are adjoint of each other with different dimensions.

An ODE system is then obtained from \((10)\).

\[
\dot{u}_h(t) = \frac{\mu}{\rho} QG u_h(t) + \frac{1}{\rho} QB F_h(t),
\]

\[
\dot{X}_h(t) = h^2 B^T u_h(t).
\]

We assume \(u_h(0) = 0\) in the rest of this section. Nonzero initial values can be handled by linear superposition

\[
u_h(t) = u_h^{(0)}(t) + u_h^{(1)}(t),
\]

in which \(u_h^{(0)}(0) = 0\) and \(u_h^{(1)}(0) = u_h(0)\). Then one can decompose \((14)-(15)\) to

\[
\dot{u}_h^{(0)}(t) = \frac{\mu}{\rho} QG u_h^{(0)}(t) + \frac{1}{\rho} QB F_h(t),
\]

\[
\dot{u}_h^{(1)}(t) = \frac{\mu}{\rho} QG u_h^{(1)}(t),
\]

\[
\dot{X}_h(t) = h^2 B^T (u_h^{(0)}(t) + u_h^{(1)}(t)).
\]

The dynamics of \(u_h^{(1)}(t)\), which has nonzero initial value, is described by a first-order linear ODE, without interactions with the immersed structure. It can be solved separately in advance, or in some cases, it can be resolved analytically.

We consider a general Galerkin projection of \((14)\), motivated by its success in reduced-order problems \([37, 26]\). More specifically, we seek \(\tilde{u}_h(t)\) in a subspace, spanned by the columns of a tall matrix \(V\), as an approximation for \(u_h(t)\), such that for any \(w(t)\) in a test space, spanned by the columns of a tall matrix \(W\), we have

\[
\left( \dot{\tilde{u}}_h(t) - \frac{\mu}{\rho} QG \tilde{u}_h(t) - \frac{1}{\rho} QB F_h(t), w(t) \right) = 0.
\]

Note that the subspaces are not necessarily fixed, which means that the matrices \(V\) and \(W\) are generally time-dependent. This point will be addressed in Section 3.2.

In a matrix-vector form, the approximate solution is written as

\[
\tilde{u}_h(t) = Vz(t).
\]

Then the Galerkin projection yields a reduced-order equation

\[
W^T (\dot{V} z)(t) = \frac{\mu}{\rho} W^T QGV z(t) + \frac{1}{\rho} W^T QB F_h(t).
\]

Thus we obtain an ROM of \((14)-(15)\):

\[
\dot{z}(t) = M_0^{-1} M_1 z(t) + M_0^{-1} M_2 F_h(t),
\]

\[
\dot{X}_h(t) \approx h^2 B^T V z(t),
\]

where the matrices are given by

\[
M_0 = W^T V, \quad M_1 = \frac{\mu}{\rho} W^T QGV - W^T V, \quad M_2 = \frac{1}{\rho} W^T QB,
\]

assuming \(M_0\) is nonsingular. The computation of the coefficient matrices \(M_0, M_1\) and \(M_2\) depends on \(V\) and \(W\). In the rest of this section we first discuss our choice for the subspaces \(V, W\) and their properties. Then we demonstrate how an interpolation procedure can help accelerate the computation of the coefficients by exploiting the connection between the matrix entries and the Green’s function.
3.1 Subspace Selection

We propose the following choice of \( V \) and \( W \),
\[
V = QB, \quad W = B. \tag{18}
\]
For later reference, note that both subspaces vary in time. The resulting coefficient matrices are given by
\[
M_0 = B^T QB, \quad M_1 = \frac{\mu}{\rho} B^T QGQB - B^T Q\dot{B}, \quad M_2 = \frac{1}{\rho} B^T QB = \frac{1}{\rho} M_0. \tag{19}
\]
In principle, one can use higher dimensional Krylov subspaces (\( V \) and \( W \) with more columns), followed by Lanczos orthogonalization algorithms \([26, 37, 38]\), to improve the accuracy of the ROM. Specifically, we shall see in the following discussion that our choice satisfies two interpolation conditions. Higher dimensional Krylov subspaces are able to interpolate the transfer function more accurately by enforcing more interpolation conditions, but at the expense of a reduced computational speedup. From the numerical tests, our observation is that the current subspaces achieve a good balance between accuracy and efficiency.

3.1.1 Transfer function approximation

We first show the accuracy property of our choice of subspaces. This can be understood by solving the linear ODE (14) analytically for \( u_h \), which yields,
\[
u_h(t) = \frac{1}{\rho} \int_0^t \exp \left[ \frac{\mu}{\rho} (t - \tau)QG \right] QB F_h(\tau) d\tau. \tag{20}\]
Note that we assume zero initial condition as discussed before. Plugging (20) into (15) gives
\[
\dot{X}_h(t) = \int_0^t \phi(t - \tau) F_h(\tau) d\tau,
\]
where \( \phi(t) \) denotes the transfer function,
\[
\phi(t) = \frac{h^2}{\rho} B^T \exp \left[ \frac{\mu_t}{\rho} tQG \right] QB. \tag{21}\]
A similar calculation for the reduced-order model (16) - (17) shows that:
\[
\dot{X}_h(t) \approx \int_0^t \phi_{red}(t - \tau) F_h(\tau) d\tau,
\]
where the transfer function \( \phi_{red}(t) \) of the reduced-order model is given by,
\[
\phi_{red}(t) = h^2 B^T V \exp \left[ tM_0^{-1}M_1 \right] M_0^{-1} M_2,
\]
\[
= \frac{h^2}{\rho} B^T QB \exp \left[ t(B^T QB)^{-1} \left( \frac{\mu}{\rho} B^T QGQB - B^T Q\dot{B} \right) \right]. \tag{22}\]
\( \phi_{red}(t) \) is expected to approximate \( \phi(t) \) in the sense that,
\[
\phi_{red}(0) = \phi(0), \quad \phi_{red}(0) = \dot{\phi}(0). \tag{23}\]
The equality (23) follows immediately from evaluating (21) and (22) at \( t = 0 \). Differentiating (21) and (22) at \( t = 0 \) yields
\[
\dot{\phi}_{red}(0) = \frac{\mu h^2}{\rho^2} B^T QB (B^T QB)^{-1} B^T QGQB = \frac{\mu h^2}{\rho^2} B^T QGQB = \dot{\phi}(0).
\]
In the above calculation, we have treated \( B \) as a constant matrix. The reason is that we are only concerned with a small time interval \([0, t]\), typically with the size of one time step. In numerical simulations, the matrix \( B \) is usually treated as constant when advancing one time step.
3.1.2 Enforcing incompressibility

Another essential property of the full model is the incompressibility of the fluid. Recall that $M$ is the discrete divergence operator. The approximate fluid solution,

$$\hat{u}_h(t) = V z(t),$$

is incompressible if

$$M \hat{u}_h(t) = MV z(t) = 0.$$

A quick calculation verifies that our choice of $V = QB$ satisfies this constraint:

$$MV = MQB = M(I - N(MN)^{-1}M)B = (M - MN(MN)^{-1}M)B = 0.$$ 

Therefore, the incompressibility property is preserved in the ROM.

3.1.3 Lyapunov Stability

The ROM also preserves Lyapunov stability of the FOM with our choice of subspaces. We first show the stability of the FOM. We assume the discrete Lagrangian force density $F_h$ is given by an energy functional $W(X_h)$ of the structure configuration, i.e.,

$$F_h(t) = -\nabla_{X_h} W(X_h(t)).$$

We also assume the discrete gradient and divergence operators satisfy

$$M = N^T,$$

such that

$$Q = I - \Sigma = I - N(N^T N)^{-1}N^T$$

is an orthogonal projection. We now define the following Lyapunov functional for the FOM consisting of the kinetic and the elastic energy,

$$V(u_h(t), X_h(t)) = \frac{1}{2} u_h(t)^T Q u_h(t) + \frac{1}{\rho} W(X_h(t)).$$

We have $V(u_h(t), X_h(t)) \geq 0$ because $Q$, as a projection, is positive semidefinite with eigenvalues 0 or 1. In particular, notice that $Q^2 = Q$ and $Q^T = Q$. In addition, the divergence-free condition implies that $Q u_h(t) = u_h(t)$. A direct calculation shows that

$$\dot{V}(u_h(t), X_h(t)) = u_h(t)^T Q^2 \left( \frac{\mu}{\rho} G u_h(t) + \frac{1}{\rho} B F_h(t) \right) - \frac{1}{\rho} u_h(t)^T B F_h(t)$$

$$= \frac{\mu}{\rho} u_h(t)^T Q G u_h(t) + \frac{1}{\rho} u_h(t)^T Q B F_h(t) - \frac{1}{\rho} u_h(t)^T B F_h(t)$$

$$= \frac{\mu}{\rho} u_h(t)^T Q G u_h(t) \leq 0,$$

since the discrete Laplace operator $G$ is negative semidefinite. This implies the Lyapunov stability of the FOM.

The Lyapunov functional for the ROM is defined as follows

$$V_r(X_h(t)) = \frac{h^2}{\rho} W(X_h(t)) + \frac{1}{2} \dot{X}_h(t)^T (B^T Q B)^{-1} X_h(t).$$

It is now clear that $V_r(X_h(t)) \geq 0$ holds for all $t$ since $B^T Q B$ is positive semidefinite.

To prove $\dot{V}_r(X_h(t)) \leq 0$, we start by rewriting the ROM (16) - (17) as a second-order ODE of $X_h$. Note that (17) and (18) imply $z(t) = h^{-2}(B^T Q B)^{-1} X_h(t)$ in the ROM. Using the symmetry of $Q$, one has,

$$\dot{X}_h = h^2 [ (B^T Q B) z + B^T Q B \dot{z}(t)]$$

$$= h^2 (B^T Q B + \frac{\mu}{\rho} B^T Q G Q B) z + \frac{h^2}{\rho} B^T Q B F_h(X_h(t))$$

$$= (B^T Q B + \frac{\mu}{\rho} B^T Q G Q B) (B^T Q B)^{-1} X_h + \frac{h^2}{\rho} B^T Q B F_h(X_h(t)).$$
Then the following calculation shows that $V_r$ is nonincreasing,

$$
V_r(X_h(t)) = -\frac{\mu}{\rho} \dot{X}_h^T F_h + X_h (B^T Q B)^{-1} \dot{X}_h + \frac{1}{2} \dot{X}_h (B^T Q B)^{-1} \dot{X}_h \\
= -\frac{\mu}{\rho} \dot{X}_h^T (B^T Q B)^{-1} (B^T Q G Q B) (B^T Q B)^{-1} \dot{X}_h + \frac{h^2}{\rho} \dot{X}_h^T F_h + \frac{h^2}{\rho} \dot{X}_h^T F_h \\
= -\frac{\mu}{\rho} \dot{X}_h^T (B^T Q B)^{-1} (B^T Q B) (B^T Q B)^{-1} \dot{X}_h \\
= -\mu Y^T G Y \leq 0,
$$

where we have defined $Y := QB (B^T Q B)^{-1} X_h$. The last inequality holds because $G$ is negative semidefinite.

### 3.2 Computing the coefficients using interpolation

Because the coefficient matrices in the ROM are in principle time dependent, they should be updated frequently during simulation. Direct matrix multiplication for this purpose is time consuming since $Q$ is a dense matrix in $\mathbb{R}^{2n_u, 2n_u}$. For example, computing $M_1$ in [19] has complexity $O(\nu_u^2)$. In the rest of this section, we propose a computationally cheaper approach using interpolation to approximate the coefficients.

We first approximate $\dot{B}$ by

$$
\dot{B}(t) \approx \frac{1}{\Delta t} (B(t) - B(t - \Delta t)).
$$

One could consider a higher order discretizations for $\dot{B}$ so that a method of order higher than one in time can be used to solve the ROM. Ultimately, this is a trade-off between accuracy and the offline interpolation efficiency. From the numerical tests, we will see such first order approximation of $\dot{B}$ together with the forward Euler’s method provides acceptable accuracy for the numerical tests compared to the FOM.

The other observation is that the matrix $M_1$ is then approximated by

$$
M_1 \approx \frac{\mu}{\rho} B(t)^T Q G Q B(t) - \frac{1}{\Delta t} B(t)^T Q B(t) + \frac{1}{\Delta t} B(t)^T Q B(t - \Delta t).
$$

Together with $M_0 = B(t)^T Q B(t)$, the following three matrices are needed for building our ROM

$$
B(t)^T Q B(t), \quad B(t)^T Q G Q B(t), \quad B(t)^T Q B(t - \Delta t).
$$

Since $Q$ and $G$ are constant matrices, the $i, j$-entry of any of the above matrices at time $t$ is determined by the $i$th row of $B(t)^T$ and the $j$th column of $B(t)$ (or $B(t - \Delta t)$). Recall that each column of $B(t)$ (or row of $B(t)^T$) represents a smoothed delta function associated with a structure point. Suppose the $i$th column of $B(t)^T$ is associated with the Lagrangian grid point $X_l \in \mathbb{R}^d$ ($d = 2, 3$) and the $j$th column of $B(t)^T$ (or $B(t - \Delta t)$) is associated with $X_r \in \mathbb{R}^d$. Given the prescribed function $\delta$, and a fixed Eulerian grid, the $i, j$-entry of a coefficient matrix is uniquely determined by $X_l$ and $X_r$, which can be viewed as a function from $\mathbb{R}^{2d}$ to $\mathbb{R}$. It is then natural to sample such functions before the simulation starts. As the simulation runs, coefficient matrices are updated by interpolation using precomputed samples. In this work, linear interpolation is used. Because each entry of the $2n_s$-by-$2n_s$ coefficient matrix is obtained by evaluating a precomputed linear function, the complexity is typically $O(n_u^2)$, which is much smaller than the complexity of direct matrix multiplications $O(n_u^2)$, given $n_s \ll n_u$.

Next, motivated by our numerical experiments illustrated in Fig 2, we show that the interpolated $2d$-dimensional functions of $X_l$ and $X_r$ can be well approximated by $d$-dimensional functions of $X_l - X_r$, i.e., the relative position of the two points. Such low-dimensional approximation significantly reduces the number of samples needed for more accurate interpolations. Hence the sampling process can also be accelerated.

Here we provide justifications of this approach by making connections to the Green’s functions. Recall that $Q = I - N (MN)^{-1} M$, where $N$ and $M$ are discrete gradient and divergence operators. Therefore, each entry of the
Figure 2: Surface plots of entries of (a) $M_0$ and (b) $M_1$ against $X_l - X_r = (x_l - x_r, y_l - y_r)$ in 2D case. Both matrices are 2-by-2 block matrices where each block corresponds to a function from $\mathbb{R}^4$ to $\mathbb{R}$. Large numbers of $(X_l, X_r) = (x_l, y_l, x_r, y_r)$ pairs are sampled so that many of them correspond to the same difference $X_l - X_r$. Then corresponding entries of $M_0$ and $M_1$ blocks are plotted against $X_l - X_r$. In each plot, we observe a single surface, indicating no multiple values. Therefore these $\mathbb{R}^4$ functions can be considered as functions in $\mathbb{R}^2$ of $X_l - X_r$.

Figure 3: Scatter plots of entries of (a) $M_0$ and (b) $M_1$ against $|X_l - X_r|$ in 2D case. Multiple values exist, indicating these $\mathbb{R}^4$ functions may not be considered as functions in $\mathbb{R}$ of $|X_l - X_r|$. 

matrix $B(t)^T Q B(t)$ or $B(t)^T Q B(t - \Delta t)$ is a numerical approximation of the integral
\[
I_0 = \int_{\Omega} \delta_r(x - X_l)(\delta_{ij} - \delta_i \Delta^{-1} \delta_j) \delta_r(x - X_r) \, dx
\]
\[
= \delta_{ij} \int_{\Omega} \delta_r(x - X_l) \delta_r(x - X_r) \, dx - \int_{\Omega} \delta_r(x - X_l) \delta_i \Delta^{-1} \delta_j \delta_r(x - X_r) \, dx,
\]
where $i, j = 1, \cdots, d$ and $d = 2$ or $3$. $\delta_{ij}$ is the Kronecker delta function. $I_1$ only depends on $|X_l - X_r|$ due to our choice of $\delta_r$. For $I_2$, we assume $X_l$ and $X_r$ are far from the boundary of $\Omega$, so the Green’s function $G(x, y)$ can be applied. Considering the limiting case of $r \to 0$, i.e., $\delta_r \to \delta$, as $r \to 0$, and we arrive at,
\[
\lim_{r \to 0} I_2 = -\partial_{x_i} \partial_{y_j} G(x, y) \big|_{x=X_l, y=X_r},
\]
which depends only on $X_l - X_r$.

Similarly, each entry of the matrix $B(t)^T Q G Q B(t)$ is a numerical approximation of the following integral
\[
J_0 = \int_{\Omega} \delta_r(x - X_l)(\delta_{ij} - \delta_i \Delta^{-1} \delta_j) \Delta(1 - \delta_j \Delta^{-1} \delta_j) \delta_r(x - X_r) \, dx
\]
\[
= \int_{\Omega} \delta_r(x - X_l) \delta_{ij} \Delta \delta_r(x - X_r) \, dx - \int_{\Omega} \delta_r(x - X_l) \delta_i \Delta^{-1} \delta_j \delta_r(x - X_r) \, dx,
\]
where $J_{11}$ and $J_{21}$ depend on $X_l - X_r$ due to our choice of $\delta_r$. For $J_{12}$ and $J_{22}$ we make the same assumptions as for $I_2$ and consider the limiting case. We obtain similar results
\[
\lim_{r \to 0} J_{12} = -\partial_{x_i} \partial_{y_j} G(x, y) \big|_{x=X_l, y=X_r},
\]
\[
\lim_{r \to 0} J_{22} = -\partial_{x_i} \Delta \partial_{y_j} G(x, y) \big|_{x=X_l, y=X_r}.
\]
So both terms depend only on $X_l - X_r$. However, the integrals $I_0$ and $J_0$ may not be further reduced to functions of $|X_l - X_r|$, as suggested by our numerical experiments, see Fig. 3.

4 Numerical results

In this section, we present three numerical examples to demonstrate the accuracy and speedup offered by our ROM. The finite difference method is used for both FOM and ROM. For the temporal discretization, we use the forward Euler method.

4.1 Oscillation of an elliptical membrane

We consider the oscillations of a pressurized fiber. Initially, the stretched elastic fiber resides in the center of a resting fluid. The semi-major and semi-minor axes of the fiber are 0.4 and 0.2 $\mu$m, respectively. The fluid domain is $4\mu m \times 4\mu m$ with periodic boundary conditions on all edges. Fluid density and viscosity are chosen so that the Reynolds number is 0.01. The body force in this example is generated by an elastic energy functional [13],
\[
E = \int_{\Gamma} \varepsilon \left( \left| \frac{\partial X}{\partial s} \right| \right) \, ds,
\]
where $\varepsilon$ is the local energy given by
\[
\varepsilon(x) = \frac{\sigma}{2} (x - L)^2,
\]
which corresponds to an elastic fiber having a “spring constant” $\sigma$ and an equilibrium state where the elastic strain $|\partial X / \partial s| \equiv L$. The force in (4) is then expressed as

$$F = -\nabla_X E = \sigma \frac{\partial^2}{\partial s^2} \left( \frac{\partial X}{\partial s} \left( 1 - \frac{L}{|\partial X / \partial s|} \right) \right).$$

Since the fluid in the interior of the membrane is confined, the membrane will oscillate and eventually settle into a circular state. Membrane configurations simulated by the FOM and ROM are compared at different times (Fig 4a & b). The ROM simulation captures almost the same equilibrium state as the FOM. In addition, the membrane configurations are approximated accurately during the oscillation. We demonstrate that the ROM preserves the incompressibility by comparing the evolution of mass flux with that of the FOM. The mass flux is calculated by integrating the velocity over the membrane surface using the trapezoidal rule. The mass flux of the ROM is in close agreement with the FOM. Both are very close to zero up to a numerical error which keeps decreasing as the grid becomes finer, as shown in Fig. 4c & d.

![Figure 4](image.png)

Figure 4: (a) Comparison between FOM and ROM of the elliptical membrane profiles at different times. (b) Relative difference between the x-coordinates of a reference structure point simulated by the FOM and ROM. (c, d) Evolution of mass flux across the membrane with fluid grid size $h = 1/8$ (c) and $h = 1/16$ (d).

The one-step computation time of our ROM simulations with various grid sizes is compared to the one-step FOM simulation time in Table 1. There is a clear increase in the speedup factor as the grid spacing decreases. With 2D flow, the time complexities are $O(h^{-2})$ and $O(h^{-1})$ for the FOM and ROM simulations, respectively. The effect of the additional sampling cost at the beginning of the simulation is reported in Table 2. This overhead is less than 20 time steps of the FOM simulation. In this example, the total number of time steps is 1000. Therefore the computational cost associated with the sampling process is negligible compared to the speedup during the simulation.

We show the perimeter of the immersed structure at final time for various choices of the grid size (Fig 5). As the grid spacing reduces, the perimeter approaches an asymptotic zero-grid spacing value. We determine the order of convergence of the ROM based on these results,

$$\ln \left( \frac{1.7759 - 1.7863}{1.7718 - 1.7759} \right) / \ln(2) = 1.3429.$$
4.2 Rotation of an elliptical particle in shear flow

We study the problem of the motion of a rigid elliptical particle freely suspended in a shear flow. The fluid domain is $8\mu m \times 8\mu m$. The semi-minor and semi-major axes of the ellipse are $S_1 = 0.2\mu m$ and $S_2 = 0.3\mu m$, respectively. Initially, the ellipse is immersed in the center of a shear flow with its semi-major axis positioned along the $y$-axis. The maximum fluid velocity of the shear flow, fluid density, viscosity are chosen so that the Reynolds number is 0.01. (Fig 6)
Figure 7: Ellipse rotation angles simulated by the full model and the reduced-order model compared with Jeffery’s orbit. The variation in the angle $\theta$ relative to the ellipse major axis is plotted as a function of time $t$.

It has been shown that the instantaneous inclination angle $\theta$ of the ellipse major axis with respect to the $y$-axis is

$$\tan(\theta) = \frac{S_2}{S_1} \tan\left(\frac{S_1 S_2 \gamma t}{S_1^2 + S_2^2}\right),$$

(24)

where $t$ is the time variable [39].

To preserve the elliptic shape of the rigid structure, the body force in this example is generated by a discrete bending energy [33]. Let $\theta_i^0$ be the initial angle between the adjacent edges with the $i$-th Lagrangian grid point and $\theta_i$ be the current angle. The bending energy is given by

$$E_b = \sigma_b \sum_{i=1}^{n_s} (1 - \cos(\theta_i - \theta_i^0)),$$

where $n_s$ is the number of Lagrangian grid points and $\sigma_b$ is the bending coefficient. In this example, we choose $\sigma_b = 2000$ to increase the stiffness. The bending force generated on each structure point is given by,

$$F_i = (F_{ix}, F_{iy}) = \left(-\frac{\partial E_b}{\partial x_i}, -\frac{\partial E_b}{\partial y_i}\right).$$

Fig[7] shows the simulated ellipse rotation rate and the analytical result (24). The rotation rate obtained by our ROM simulation is in close agreement with both the FOM simulation and the analytical solution. Table[3] shows the increase in the speedup factor as the grids become finer. Higher speedup factors are achieved for finer space grid.

4.3 Motion of two particles in laminar flow

In the last numerical test, we simulate the motion of two membranes in a $6\mu m \times 15\mu m$ channel. The fluid is initially at rest, with inlet velocity profile given by, as depicted in Fig[8]

$$U = U_0 \left[1 - \left(\frac{y}{D}\right)^2\right], \quad -D \leq y \leq D.$$

At the beginning, the two membranes of the same elliptic shape are placed with horizontal semi-major axes and the same distance $0.6\mu m$ from its center to the $x$-axis. The initial semi-major axis and semi-minor axis are $0.3\mu m$ and
Figure 8: Two membranes interacting in a laminar channel flow.

0.2\(\mu\)m, respectively. Fluid density, viscosity, \(U_0\) are chosen so that the Reynolds number is 0.01. Nonslip conditions are applied to the top and bottom boundaries.

The same bending force as in the previous example is applied to both membranes to prevent significant deformation. In addition, the two membranes interact with each other through a binding force and a repulsive force given respectively by

\[
F_{binding} = s(d - \lambda),
\]

\[
F_{repulsion} = ad + bd^3,
\]

where \(d\) is the distance between two Lagrangian nodes on different cells and \(a, b, s, \lambda\) are parameters. These forces are developed to model the biochemical interactions between flowing melanoma tumor cells and substrate adherent polymorphonuclear neutrophils [40]. The attraction and repulsion forces yield oscillatory trajectories for both membranes, shown in Fig. The other two parameter, \(b\) and \(\lambda\), are fixed for ease of comparison. (a) Snapshots of two membranes at different times. (b) Trajectories of membrane centers. (c) Relative error in x-coordinates of a reference structure point of the upper membrane, between the FOM and ROM.

4.4 Transport of circular capsule in a plain-Poiseuille flow

In this test case, the dynamics of a capsule within a plane-Poiseuille flow is considered. The setup of this example follows the test conducted by Coclite et al. [41]. Initially, the capsule has a diameter of 7 \(\mu\)m and is immersed in a 2D channel with a height \(H = 15\ \mu\)m and length equal 3\(H\), centered at 7.5 \(\mu\)m away from the bottom of the lower wall. The fluid is initially at rest, the plane-Poiseuille flow with \(u_{max} = 10\ \mu\)m/s is then established by posing a linear pressure drop. Simulation is run at \(Re = 0.01\), with \(\rho = 100\ \text{kg} \cdot \text{m}^{-3}\), \(\mu = 10^{-5}\ \text{Pa} \cdot \text{s}\). The body force on the capsule is the same as in Section 4.1, with three spring constant \(\sigma = 10^{-5}N \cdot \mu m^{-1}, 10^{-4}N \cdot \mu m^{-1}\) and \(10^{-3}N \cdot \mu m^{-1}\).
Following Cocli et al. [41], we compare the results between the FOM and the ROM in terms of the capsule perimeter variation with respect to its original configuration, $\delta p(t) = p(t) - p_0$ (Fig 10a), and of the swelling ratio, $Sw = \frac{A(t)}{p(t)^2/4\pi}$, where $A(t)$ is the area associated with a circle of perimeter $p(t)$ (Fig 10b). The snapshots of FOM and ROM are also compared (Fig 10c, d & e).

For $\sigma = 10^{-5}$, the ROM is a fair approximation of the FOM. As the force coefficient increases, the system becomes more stiff. Consequently, the ROM simulation does not approximate the FOM well. We emphasize that our result is not in full agreement with the data published in Cocli et al. [41] for two reasons. First, the time-dependent Stokes equations are considered in this work instead of Navier-Stokes equations. Secondly, the force we applied to the cell model is different.

![Figure 10](image-url)

Figure 10: Transport of a circular capsule in plane-Poiseuille flow. (a) Variation of the capsule relative perimeter over time as function of the spring constant $\sigma$. (b) Variation of the capsule swelling ratio over time as function of the spring constant $\sigma$. (c,d,e) Capsule snapshots for $\sigma = 10^{-5}$ (c), $\sigma = 10^{-4}$ (d), and $\sigma = 10^{-3}$ (e). (Solid curves for FOM and dashed curves for ROM)

## 5 Conclusion

In this paper, we develop a reduced-order modeling framework for FSI problems. Using the IBM as an example, we discussed the transfer function and its approximations. This proposed ROM formulation enforces the impressibility condition and also preserves the Lyapunov stability. An efficient interpolation technique is applied to efficiently update the time-dependent coefficient matrices. The proposed model reduction technique is applied to several biological applications involving linear incompressible Stokes flows, as demonstrated by the examples. Compared to other traditional methods, this new method has the following two advantages: 1) the fluid variables are the most time-consuming part in the traditional methods, such as IBM, IIM, and FDM. But they are not explicitly involved in our ROM; 2) the structure equation is derived explicitly. It does not require special discretization techniques, e.g., those for singular integrals used in the BEM. Recently, there have been growing interest in combining the reduced-order technique and data-driven methods. In this scenario, rather than the direct access to the FOM, one works with observations, e.g., structure conformations, in the form of time series. The problem is then reduced to inferring parameters in the ROM. This work is underway.
Acknowledgments

This work is supported by the National Science Foundation Grants DMS-1953120 (XL) and DMS-2052685 (WH).

Table 1: Membrane oscillation: Speedup of full order model and reduced-order model.

| h   | Model order | CPU time | Speedup factor |
|-----|-------------|----------|----------------|
|     | full | reduced | full | reduced |             |
| 1/6 | 1728 | 144 | .0118 | .0036 | 3.2778 |
| 1/8 | 3072 | 192 | .0309 | .0044 | 7.0227 |
| 1/12 | 6912 | 288 | .1391 | .007  | 19.871 |
| 1/16 | 12288 | 384 | .3940 | .0166 | 24.735 |
| 1/20 | 19200 | 480 | .9745 | .0275 | 35.436 |

Table 2: Membrane oscillation: Sampling cost and overall expected time saving (in seconds) for different numbers of total time steps.

| h   | Sampling time | N_T = 15 | N_T = 30 | N_T = 50 |
|-----|---------------|----------|----------|----------|
|     |               | FOM | ROM | Saving | FOM | ROM | Saving | FOM | ROM | Saving |
| 1/8 | .053 | .464 | .119 | .345 | .927 | .185 | .742 | 1.35 | .273 | 1.28 |
| 1/12 | .170 | 2.09 | .275 | 1.81 | 4.17 | .380 | 3.79 | 6.95 | .520 | 6.43 |
| 1/16 | .748 | 5.91 | .997 | 4.94 | 11.8 | 1.246 | 10.5 | 19.7 | 1.58 | 18.1 |

Table 3: Particle rotation: Speedup of full order model and reduced-order model.

| h   | Model order | CPU time | Speedup factor |
|-----|-------------|----------|----------------|
|     | full | reduced | full | reduced |             |
| 3/16 | 2048 | 32 | .0214 | .0017 | 12.5882 |
| 1/8  | 4608 | 48 | .1012 | .0029 | 34.8966 |
| 3/32 | 8192 | 64 | .3065 | .0057 | 53.7719 |

References

[1] Clement Kleinstreuer. Biofluid Dynamics: Principles and Selected Applications. CRC Press, 2006.

[2] S. Canic. Blood flow through compliant vessels after endovascular repair: wall deformations induced by the discontinuous wall properties. Computing and Visualization in Science, 4(3):147–155, 2002.

[3] S. Canic and D. Mirkovic. A hyperbolic system of conservation laws in modeling endovascular treatment of abdominal aortic aneurysm. In Heinrich Freistühler and Gerald Warnecke, editors, Hyperbolic Problems: Theory, Numerics, Applications, volume 140, pages 227–236. Birkhäuser Basel, Basel, 2001.

[4] Huan Lei and George Em Karniadakis. Predicting the morphology of sickle red blood cells using coarse-grained models of intracellular aligned hemoglobin polymers. Soft Matter, 8(16):4507–4516, 2012.

[5] Xuejin Li, Zhangli Peng, Huan Lei, Ming Dao, and George Em Karniadakis. Probing red blood cell mechanics, rheology and dynamics with a two-component multi-scale model. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 372(2021):20130389, 2014.

[6] Q. Wang. A hydrodynamic theory for solutions of nonhomogeneous nematic liquid crystalline polymers of different configurations. The Journal of Chemical Physics, 116(20):9120–9136, 2002.

[7] X. Yang, G. Forest, W. Mullins, and Q. Wang. Dynamic defect morphology and hydrodynamics of sheared nematic polymers in two space dimensions. Journal of Rheology, 53(3):589–615, 2009.

[8] Yue Yu. Fluid-structure interaction modeling in 3d cerebral arteries and aneurysms. In Peter Wriggers and Thomas Lenarz, editors, Biomedical Technology: Modeling, Experiments and Simulation, pages 123–146. Springer, Cham, 2018.

[9] J. Zhao, X. Yang, J. Shen, and Q. Wang. A decoupled energy stable scheme for a hydrodynamic phase-field model of mixtures of nematic liquid crystals and viscous fluids. Journal of Computational Physics, 305:539–556, 2016.

[10] William S Hall. The Boundary Element Method, volume 27. Springer Netherlands, 1 edition, 1994.
Table 4: Two cells interaction: Speedup of full order model and reduced-order model.

| h   | Model order | CPU time | Speedup factor |
|-----|-------------|----------|----------------|
|     | full        | reduced  | full           | reduced       |               |
| 3/16| 4096        | 64       | .0834          | .0080         | 10.425        |
| 1/8 | 9216        | 96       | .4272          | .0148         | 28.8649       |
| 3/32| 16384       | 128      | 1.2492         | .0378         | 33.0476       |

[11] Shuwang Li and Xiaofan Li. A boundary integral method for computing the dynamics of an epitaxial island. *SIAM Journal on Scientific Computing*, 33(6):3282–3302, 2011.

[12] Gordon C Eversine and Francis M Henderson. Coupled finite element/boundary element approach for fluid–structure interaction. *The Journal of the Acoustical Society of America*, 87(5):1938–1947, 1990.

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

[14] Ming-Chih Lai and Charles S Peskin. An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *Journal of Computational Physics*, 160(2):705–719, 2000.

[15] Paul J. Atzberger, Peter R. Kramer, and Charles S. Peskin. A stochastic immersed boundary method for fluid-structure dynamics at microscopic length scales. *Journal of Computational Physics*, 224(2):1255–1292, June 2007.

[16] Fotis Sotiropoulos and Xiaolei Yang. Immersed boundary methods for simulating fluid–structure interaction. *Progress in Aerospace Sciences*, 65:1–21, 2014.

[28] S. Gugercin, A. Antoulas, and C. Beattie. *H2 Model Reduction for Large-Scale Linear Dynamical Systems*. *SIAM Journal on Matrix Analysis and Applications*, 30(2):609–638, 2008.

[29] Branimir Anic. *An interpolation-based approach to the weighted H2 model reduction problem*. PhD thesis, Virginia Polytechnic Institute and State University, Virginia, 2008.

[30] Yuanxun Bao, Aleksandar Donev, Boyce E Griffith, David M McQueen, and Charles S Peskin. An immersed boundary method with divergence-free velocity interpolation and force spreading. *Journal of Computational Physics*, 347:183–206, 2017.
[31] Alessandro Nitti, Josef Kiendl, Alessandro Reali, and Marco D de Tullio. An immersed-boundary/isogeometric method for fluid–structure interaction involving thin shells. *Computer Methods in Applied Mechanics and Engineering*, 364:112977, 2020.

[32] Anvar Gilmanov, Trung Bao Le, and Fotis Sotiropoulos. A numerical approach for simulating fluid structure interaction of flexible thin shells undergoing arbitrarily large deformations in complex domains. *Journal of Computational Physics*, 300:814–843, 2015.

[33] Igor V Pivkin and George Em Karniadakis. Accurate coarse-grained modeling of red blood cells. *Physical Review Letters*, 101(11):118105, 2008.

[34] Daniele Boffi and Lucia Gastaldi. A finite element approach for the immersed boundary method. *Computers & Structures*, 81(8-11):491–501, 2003.

[35] Jungwoo Kim, Dongjoo Kim, and Haecheon Choi. An immersed-boundary finite-volume method for simulations of flow in complex geometries. *Journal of Computational Physics*, 171(1):132–150, 2001.

[36] Jerry Zhijian Yang, Xiaojie Wu, and Xiantao Li. A generalized irving–kirkwood formula for the calculation of stress in molecular dynamics models. *The Journal of Chemical Physics*, 137(13):134104, 2012.

[37] Roland W Freund. Krylov-subspace methods for reduced-order modeling in circuit simulation. *Journal of Computational and Applied Mathematics*, 123(1-2):395–421, 2000.

[38] Lina Ma, Xiantao Li, and Chun Liu. Coarse-graining langevin dynamics using reduced-order techniques. *Journal of Computational Physics*, 380:170–190, 2019.

[39] George Barker Jeffery. The motion of ellipsoidal particles immersed in a viscous fluid. *Proceedings of the Royal Society of London. Series A*, 102(715):161–179, 1922.

[40] Julie Behr, Byron Gaskin, Changliang Fu, Cheng Dong, and Robert Kunz. Localized modeling of biochemical and flow interactions during cancer cell adhesion. *PloS One*, 10(9):e0136926, 2015.

[41] A. Coclite, S. Ranaldo, M.D. de Tullio, P. Decuzzi, and G. Pascazio. Kinematic and dynamic forcing strategies for predicting the transport of inertial capsules via a combined lattice boltzmann – immersed boundary method. *Computers and Fluids*, 180:41–53, 2019.