Modeling the flow of dense suspensions of deformable particles in three dimensions

Michael M. Dupin*

Steele Laboratory, Massachusetts General Hospital and Harvard Medical School, Boston, Massachusetts 02114, USA

Ian Halliday,† Chris M. Care,‡ and Lyuba Alboul

Materials and Engineering Research Institute, Sheffield Hallam University, Sheffield S1 1WB, United Kingdom

Lance L. Munn§

Steele Laboratory, Massachusetts General Hospital and Harvard Medical School, Boston, Massachusetts 02114, USA

We describe here a rigorous and accurate model for the simulation of three-dimensional deformable particles (DPs). The method is very versatile, easily simulating various types of deformable particles such as vesicles, capsules, and biological cells. Each DP is resolved explicitly and advects within the surrounding Newtonian fluid. The DPs have a preferred rest shape (e.g., spherical for vesicles, or biconcave for red blood cells). The model uses a classic hybrid system: an Eulerian approach is used for the Navier-Stokes solver (the lattice Boltzmann method) and a Lagrangian approach for the evolution of the DP mesh. Coupling is accomplished through the lattice Boltzmann velocity field, which transmits force to the membranes of the DPs. The novelty of this method resides in its ability (by design) to simulate a large number of DPs within the bounds of current computational limitations: our simple and efficient approach is to (i) use the lattice Boltzmann method because of its acknowledged efficiency at low Reynolds number and its ease of parallelization, and (ii) model the DP dynamics using a coarse mesh (approximately 500 nodes) and a spring model constraining (if necessary) local area, total area, cell volume, local curvature, and local primary stresses. We show that this approach is comparable to the more common—yet numerically expensive—approach of membrane potential function, through a series of quantitative comparisons. To demonstrate the capabilities of the model, we simulate the flow of 200 densely packed red blood cells—a computationally challenging task. The model is very efficient, requiring of the order of minutes for a single DP in a 50 μm $\times$ 40 μm $\times$ 40 μm simulation domain and only hours for 200 DPs in 80 μm $\times$ 30 μm $\times$ 30 μm. Moreover, the model is highly scalable and efficient compared to other models of blood cells in flow, making it an ideal and unique tool for studying blood flow in microvessels or vesicle or capsule flow (or a mixture of different particles). In addition to directly predicting fluid dynamics in complex suspension in any geometry, the model allows determination of accurate, empirical rules which may improve existing macroscopic, continuum models.

PACS number(s): 47.11.Qr, 83.10.Rs

I. INTRODUCTION

A. General observation

The flow of dense particle suspensions has been studied extensively experimentally. Two major areas of research have emerged naturally: solid particle suspensions such as colloids, which are relevant in shallow water sediment resuspension, river bed dynamics, etc., and deformable particle suspensions, which are important in the petroleum industry, pharmaceutical production, blood flow, emulsions, etc.

Within these two categories, models and experiments can address various length scales, namely, *mesoscopic* which focuses on length scales comparable to the size of the particles, and *macroscopic* which includes length scales at least two orders of magnitude larger than the size of the particles. Numerous computational models now exist for simulating the flow of solid particle suspensions, both at the mesoscopic and macroscopic levels.

The current paradigm is that, while many models can accurately approximate macroscopic flow of solid and deformable particles, mesoscopic simulation of deformable particles (DPs) remains a challenge. It is the latter domain, which contains complex and interesting fluid dynamics, in which particle size is similar to the conduit size, that we have focused our model of deformable particle. The flow of blood through microvessels is an important example.

Decades of experimental observations have shown that the two phase nature of blood is responsible for the flow anomalies observed in microvessels (i.e., the Fahraeus effect, Fahraeus-Lindqvist effect, and blood cells interactions; see [1–5], for example), and several computational approaches have led to major advances in the simulation of these systems. However, most of these models either simulate acute deformation of a single red blood cell (e.g. [6–10]), or model multiple red blood cells (RBCs), but with significant restrictions on, for example, deformability and shape (e.g., [11,12]) or the number of spatial dimensions (e.g., [11,13]). Other than these examples, numerical studies generally ignore the particulate nature of blood, and models of mesoscale blood

---

*Electronic address: mdupin@steele.mgh.harvard.edu

†Electronic address: i.halliday@shu.ac.uk

‡Electronic address: c.m.care@shu.ac.uk

§Electronic address: lance@steele.mgh.harvard.edu

dynamics which explicitly account for blood cells are lagging far behind the experimental data.

From the discussions above, we suggest that a useful, relevant DP model must (i) contain a physiologically relevant number of three-dimensional, deformable, and interacting RBCs as possible, all of which advect in flow, deform in a physically correct way, and exert a reciprocal effect on the flow (i.e., exhibit two-way coupling) and (ii) facilitate adaptable and efficient computation, so that results are obtained in a reasonable amount of time.

B. Rational

To date, no model has been reported with the ability to model $O(10^2)$ three-dimensional deformable particles. The limitation has been due to a lack of sufficient CPU capacity and memory to reproduce physically correct flow. For any model to be useful it must be designed around the current hardware limitations. Computational efficiency is therefore crucial; it may be attained through prudent choice of model assumptions, which must, however, be validated through rigorous testing. This paper demonstrates our model's efficiency and accuracy, validating our simplifying assumptions. The model was designed to run in parallel using multiple processors and to require a *reasonable* computational power and time. We define a model as *reasonable* if a simulation is complete within approximately 10 hours on 10 to 30 CPU's.

II. METHOD OVERVIEW

A. A brief overview of current DP models

The majority of current models for deformable cells are designated front-tracking (FT); i.e., the manifold surface of the cell is represented by a set of points embedded in the membrane, forming a boundary mesh which is superimposed over the separate, Eulerian lattice of the fluid domain (if the fluid is actually represented explicitly). FT methods are generally categorized as Lagrangian and provide very narrow interfaces. Alternatively, in the front capturing (FC) approach one can define a scalar field, χ, everywhere over the Eulerian mesh of the fluid. The χ field as a whole then carries information about the membrane location; a closed contour of a particular value of χ represents the position of the interface.

While FC interfaces suffer from being broader, they are usually preferred in the simulation of fluid interfaces as FT methods do not handle large interface distortions and interface breakup easily. The broader interface of the FC method is generally due to numerical diffusion and typically has a width of three to five times the underlying lattice spacing.

In the majority of the models incorporating flow effects, the governing equations for the fluid flow are determined by an Eulerian Navier-Stokes (NS) solver, a notable exception being low Reynolds Number (Re) so-called Stokesian dynamics solvers (e.g., [14]), or models limited to a fixed, simple geometry, where the flow perturbation of the cell can be added to the known viscous flow through the condition of Stokes regime (e.g., [15]).

1. Existing approaches

Sun *et al.* [16] simulated $O(10^2)$ solid RBCs in two dimensions and were able to recover the Fahraeus and Fahraeus-Lindqvist effects. Dao *et al.* ([17]) used a finite-element modeling (FEM) model to simulate the deformation of a single RBC stretched due to shear flow. The same authors also published a meshless representation of the membrane [9]. This model is, however, too computationally expensive to be applied to our goal of $O(10^2)$ RBC's. Dzwinel *et al.* developed a dissipative particle dynamics (DPD)-like method for both the RBC as a whole (inner skeleton and membrane) and the plasma [12]. As a consequence of modeling assumptions, the membrane shape was too rough to be considered in this approach. Nogushi *et al.* showed impressive tank treading behavior with their model [10]. However this method was, again, not designed to handle many RBC's. We finally note that Buxton *et al.* [18] have developed a model of three-dimensional deformable capsules using a Lagrangian and/or lattice Boltzmann method sightly different which we describe in the remainder of this section, and very recently, Alexeev *et al.* [19] used this approach to model two capsules in flow. This method would not be practical or sufficiently efficient to simulate the thin membrane of RBC's. Finally, following the work of Peskin on the immersed boundary method (IBM) for deformable boundaries (see, e.g., [20]), Feng and Michaelides ([21]) combined lattice Boltzman (LB) and IBM to simulate very stiff solid particle in flow. This approach could be applied to the current application, but computational efficiency is likely to be a limiting factor.

2. Our approach

Our model is based on a "bottom-up" approach, with the following assumptions: (i) the macroscopic deformation of each cell can be obtained by averaging the microscopic and/or molecular deformations over the cell surface, with different types of particles having different sets of microscopic and/or molecular parameters, and (ii) recovering the overall deformation of individual particles accurately will ensure correct simulation of flow containing many particles. It is assumed that the hydrodynamics provided by the LB method support the correct fluid-particle interactions.

In order to model a sufficient number of three dimensional DPs $[O(2)]$ to recover the required flow regimes, we (i) model the DP shape with a relatively coarse mesh, (ii) simulate the elastic properties of the DP membrane as a network of springs, without reference to a membrane potential function or its spatial gradients, and (iii) use an efficient fluid solver.

B. Fluid dynamics: lattice Boltzmann (LB)

1. Why lattice Boltzmann?

Clearly, the simulation of complex flows containing advected, Lagrangian particles is of interest in a variety of very important applications. However, for many, such simulations have been prohibitively demanding, from both computational

and algorithmic points of view. The advent of the lattice Boltzmann (LB) method has improved matters [22].

In fact, LB has well-recognized efficiency and algorithmic advantages over conventional computational fluid dynamics methods only for certain applications: canonically complex, time-dependent, multicomponent, and particulate flows. However, in these applications (which are highly relevant to the current work), LB methods can (i) achieve efficiency improvements approaching two orders of magnitude under the right circumstances and (ii) facilitate intuitive manipulation of the underlying flow physics. In this work advantage (i) underpins our central motivation—namely the provision of an accessible, ambitiously upscalable method. Advantage (ii) is manifest in the direct methods we use explicitly to couple advected red blood cells to flow modes; it arises as follows.

A LB flow solver works essentially at the kinetic equation scale [22], at which scale it is possible to incorporate the important interactions with Lagrangian particles (DPs). For this reason and also for simplicity of coding, adaptability, and parallelizability, we chose to use LB as a full Navier-Stokes solver. We note that LB may be adapted to Stokes flow, in which the fluid inertia is ignored in a regime of very small Re [15], but its simplest and most effective form lies in approximating the full Navier-Stokes equations. LB has previously been used to simulate blood flow (i) at the macroscopic, arterial level [23–25], (ii) at the microscopic level [16,26], and (iii) to simulate many deformable droplets in flow [27].

2. Description of lattice Boltzmann

For a detailed derivation of this and related lattice Boltzmann (LB) models, we refer the reader to Hou *et al.* [28]; for wider background details to Succi's review [22] and more recently, Sukop and Thorne's book [29].

The primary quantity in any LB method is a single particle distribution function, known as the momentum distribution function, f_i, which may be thought of as the mesoscopic density of molecules flowing in the direction of the velocity vector c_i. We chose to use the three-dimensional lattice with 19 allowed velocities, designated D3Q19 originally developed by Qian and d'Humières [30]. Each node is organized on a regular cubic lattice and connects to its nearest and next-nearest neighbors (a total of 18, see [22] for more details and illustration). The corresponding velocity vectors c_i's are defined as follows: $c_0 = (0,0,0)$, $c_{1,2} = (\pm 1,0,0)$, $c_{3,4} = (0,\pm 1,0)$, $c_{5,6} = (0,0,\pm 1)$, $c_{7,10} = (\pm 1,\pm 1,0)$, $c_{11,14} = (0,\pm 1,\pm 1)$, $c_{15,18} = (\pm 1,0,\pm 1)$.

The LB method is a kinetic equation method. Equations such as Eq. (1) (below) govern the evolution of the single-particle distribution function, and continuum length scale observables emerge from appropriate moments of the f_is. Thus, LB is not a discretization of the Navier-Stokes and continuity equations. The efficiency of the LB method comes from its "local" nature and its natural kinetic length scale. The LB method actually yields a weakly compressible form of the incompressible Navier-Stokes equations; the compressibility error is, however, very small at low Re.

For the work presented in this paper, we adopt a lattice Boltzmann Bhatnagar-Gross-Crook (LBGK) [31] scheme with a source term added to account for the systematic influence of body forces on the fluid:

$$f_i(\mathbf{x} + \mathbf{c}_i \delta_t, t + \delta_t) = f_i(\mathbf{x},t) + \frac{1}{\tau}\big[f_i^{(0)}(\rho,\mathbf{v}) - f_i(\mathbf{x},t)\big]. \quad (1)$$

The parameter τ determines the relaxation rate to a local equilibrium distribution function $f_i^{(0)}(\rho,\mathbf{v})$, and τ may be related to the fluid kinematic viscosity ν through the equation

$$\nu = c_s^2(2\tau - 1), \quad (2)$$

where c_s is the speed of sound. This result [Eq. (2)] can be recovered by a Chapman Enskog expansion similar to that used in kinetic theory [32].

The first and second discrete **c** moments of the f_i's provide the hydrodynamic, continuum length scale, observables of density $\rho(\mathbf{x},t)$, and momentum $\rho(\mathbf{x},t)\mathbf{v}(\mathbf{x},t)$:

$$\rho(\mathbf{x},t) = \sum_i f_i(\mathbf{x},t),$$

$$\rho(\mathbf{x},t)\mathbf{v}(\mathbf{x},t) = \sum_i f_i(\mathbf{x},t)\mathbf{c}_i + \sum_i \phi_i \mathbf{c}_i. \quad (3)$$

The relationship between the source term, ϕ_i, and the target fluid body force is strictly given by Eq. (20) of Ref. [33]; in the rest of the present article we shall, for simplicity and to make best use of physical insight, concentrate on this macroscopic body force (total force applied to produce the RBC reciprocal fluid motion and interactions).

The equilibrium distribution function $f_i^{(0)}$ is determined solely by these observables and takes the form of a truncated, low velocity expansion of a translating Maxwell-Boltzmann distribution,

$$f_i^{(0)}(\rho,\mathbf{v}) = t_i \rho\left[1 + \frac{\mathbf{v}\cdot\mathbf{c}_i}{c_s^2} - \frac{v^2}{2c_s^2} + \frac{(\mathbf{v}\cdot\mathbf{c}_i)^2}{2c_s^4}\right], \quad (4)$$

where the t_i's are constants, termed weights, depending only on the velocity index i and the geometry of the lattice [28,30]. For the D3Q19 lattice used in this work, the weights are given by $t_0 = 1/3$, $t_{1\ldots6} = 1/18$, $t_{7\ldots18} = 1/36$. The results in Eqs. (3) and (4) are similar to those obtained in classical kinetic theory, as explained in Ref. [32].

The term ϕ_i in Eq. (3) is a source term to impress force on the lattice fluid at the macroscopic, Navier-Stokes level. Using methods directly adapted from kinetic theory [22,32], it can be shown ([33,34]) that Eq. (1) leads to the following weakly compressible, forced Navier-Stokes evolution:

$$-c_s^2\frac{\partial}{\partial t}\rho v_\alpha + \frac{\partial}{\partial x_\beta}\rho v_\beta v_\alpha = -\frac{\partial}{\partial x_\alpha}\rho + \frac{\partial}{\partial x_\beta}(2\rho\nu S_{\alpha\beta}) + \sum_i \phi_i c_{i\alpha},$$

$$(5)$$

which includes the source forcing term explicitly. Here c_s is the speed of sound ($c_s^2 = 1/3$) and $S_{\alpha\beta} = 1/2(\partial_\alpha v_\beta + \partial_\beta v_\alpha)$ is the macroscopic, continuum scale rate of strain tensor.

3. Lattice Boltzmann boundary conditions

Solutions of the incompressible Navier-Stokes and continuity equations for the velocity and pressure, $\{\mathbf{v}(\mathbf{r},t), P(\mathbf{r},t)\}$, are most straightforwardly closed by Dirichlet boundary conditions on fluid velocity (only):

$$\mathbf{v}(\mathbf{x},t)\big|_{\mathbf{x}'t} = \mathbf{u}_0(\mathbf{x}',t), \qquad (6)$$

where $\mathbf{x}'$ denotes a position on the boundary of the flow domain and $\mathbf{u}_0(\mathbf{x}',t)$ is a specified boundary velocity distribution. The special case in which the fluid adjacent to the boundary is at rest, $\mathbf{u}_0 = \mathbf{0}$, is referred to as the no-slip condition.

Several simple and efficient techniques for imposing the no-slip condition exist for LB simulation. We use the popular midlink bounceback method which has been carefully evaluated by a number of workers (e.g., [35–37]). The f_i's on a boundary node at position $\mathbf{x}_w$ are subject to a specular reflection, taken to occur at the midpoint $\mathbf{x}_w + \frac{1}{2}\mathbf{c}_i$ of the appropriate lattice link $\mathbf{c}_i$,

$$f_{-i}(\mathbf{r}_w, t+1) = f_i(\mathbf{r}_w, t), \quad \mathbf{c}_{-i} \equiv -\mathbf{c}_i. \qquad (7)$$

This simple rule was generalized by Ladd [38] to moving boundaries with velocity $\mathbf{u}_0$ by replacing the evolution expressed in Eq. (1) with

$$f_{-i}(\mathbf{r}_w, t+1) = f_i(\mathbf{r}_w, t) + \frac{2\rho t_i}{c_s^2}(\mathbf{u}_0 \cdot \mathbf{c}_1). \qquad (8)$$

Moving walls are implemented using this approach.

Inlet and outlet fluxes and an outlet pressure (density) distribution are specified at every time step, using an appropriate equilibrium $f_i^{(0)}(\rho, \mathbf{u})$ [22]. The inlet pressure distribution is allowed to develop, while the outlet flux is taken to be a square profile matching the inlet flux. In this way, the unknown inlet density develops to a "fully developed," uniform pressure gradient condition [26,39].

C. Deformable particle dynamics

1. Different types of deformable particles

We consider in this manuscript three categories of DPs: capsules, vesicles, and red blood cells.

We define *capsules* as spheroids filled with liquid, with an unbreakable, Hermetic membrane of finite thickness (approximately 1/10th to 1/5th of their radius). This membrane provides resistance to shear and change of volume but allows relatively small perturbations in surface area.

We define *vesicles* as spheroids filled with liquid, with a membrane that is unbreakable, hermetic, and very thin compared to their radius, typically 1/1000th. Vesicles exhibit a strong resistance to any volume and total surface area change, but are insensitive to strain and only weakly sensitive to local bending change.

Red blood cells (RBCs) are biconcave disks with a rest shape conferred by two structural membrane components: the plasma membrane and the cytoskeleton (see Fig. 1). The plasma membrane is similar to the vesicle membrane, and is therefore responsible for a resistance to surface area dilata-
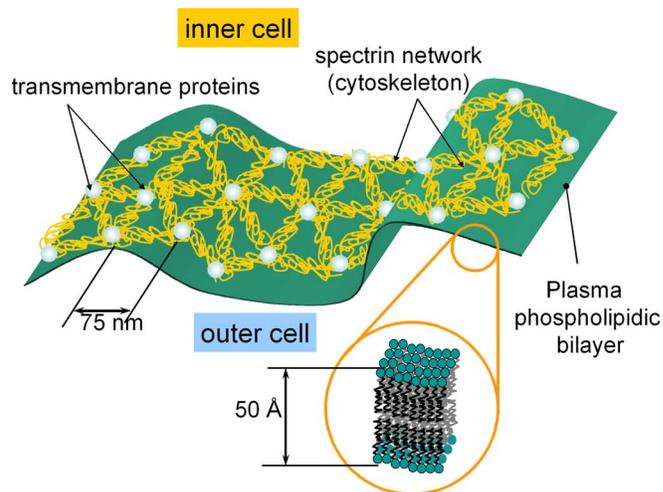
FIG. 1. (Color online) Artist's view of a RBC membrane. The cytoskeletal triangular mesh is embedded in the plasma phospholipid bilayer by the transmembrane proteins. The plasma layer (bilayer of amphifilic molecules) is believed to be responsible for the constraints of constant area and resistance to buckling. It can rearrange itself very easily, and is often referred to as a fluid membrane. The network of proteins attached underneath the plasma layer is responsible for the shear resistance (through molecular links between the proteins) and bending. The whole membrane also contains ionic pumps maintaining the inner volume of the RBC constant.

tion. The cytoskeleton, on the other hand, is a triangular protein network, that acts like a triangular network of springs, providing viscoelastic resistance to bending of the membrane (similarly to capsules, see [40–42]). Similar to capsules and vesicles, red blood cells strongly resist changes in volume. Fortunately for our modeling efforts, experimental and theoretical studies by a number of authors have analyzed RBC membrane dynamics in detail; Mukhopadhyay *et al.* [6], Kuzman *et al.* [7], Pozrikidis [8], Li *et al.* [9], and Noguchi *et al.* [10].

2. The importance of different constraints in a DP

There are four parameters that affect the mechanical energy, and therefore shape, of these particles: (1) inner volume, (2) surface area, (3) hydrodynamic shear, and (4) membrane bending.

Capsules are only subject to two constraints: volume and strain. Their volume is most often modeled as constant and their stretching dependence varies with their capsule chemistry and composition [18].

Vesicles have three constraints: strong resistance to changes in volume and surface area, and a weaker resistance to bending, which is essentially the only remaining free parameter [15].

The case of RBCs is more complicated. The characteristic biconcave shape arises because the plasma membrane does not allow expansion or contraction of the membrane (just like any other vesicle) and the cytoskeleton restricts its strain and bending, through the three-dimensional mesh it forms. We note that buckling is believed to be restricted by the

FIG. 2. (Color online) Meshes used for the RBCs, capsules, and vesicles, consisting of only approximately 400 nodes (800 faces approximately) only, compared to $\sim 10^4$ nodes for comparable methods. This relatively coarse mesh reduces computational requirements while accurately reproducing flow and cell deformation.

FIG. 3. (Color online) Diagram defining DP mesh face metrics for area, stretching, and bending.

plasma bilayer [7]. These properties are biologically crucial, for they allow RBCs to squeeze through narrow capillaries and to withstand large shear forces in the blood stream.

The RBC elastic modulus for volume compressibility K^V is of the order of 0.033 μN s cm^{-1}, the elastic modulus for area compressibility K^A is of the order of 5×10^5 pN μm^{-1}, the elastic modulus for shear K^S is of the order 5 pN μm^{-1}, and the bending elastic modulus K^B is of the order of 10^{-1} pN μm (parameters from Dao *et al.* [17] and Bagchi *et al.* [13]). Ghosh *et al.* [43] noted that, in the range of displacements we are modeling (a few micrometers), the elastic energies of bending, shearing, and dilatation are in the ratios 1:50:10^6; therefore we can write

$$E_{bend} \ll E_{shear} \ll E_{area} \ll E_{volume}. \qquad (9)$$

These values reflect the energy cost of certain mechanical deformations about a preferred value which constrains the DP dynamics. To model vesicles, we use similar volume, bending and area constraints (see the quantitative measures by Kroll *et al.*, for example). Capsules are taken to have, quantitatively, the same bending modulus and volume constraints as RBCs, but a strain constraint which can vary significantly.

3. A coarser mesh for essential gains in efficiency

Our compromise is to use a coarse mesh of approximately 400 nodes for the RBC surface, defining approximately 800 triangular faces, compared to $O(10^4)$ nodes usually involved in comparable models (e.g., in [44], [8], [45], or [9]). Each vertex is defined (on average) by six faces, also referred to as an *order 6* mesh; see Fig. 2. This resolution was arbitrarily chosen to fit both the spatial DP constraint (expressing smooth biconcavity in the case of a RBC), and computational constraints (not too fine a mesh). Once the resolution was chosen, we obtained the mesh by triangular tessellation from the equation of a sphere in the case of vesicles and, in the case of RBCs, from the empirical solution of Evans and Fung [46],

$$Z = \pm 0.5 R_0 \left[1 - \frac{X^2 + Y^2}{(R_0)^2} \right]^{0.5}$$

$$\times \left[C_0 + C_1 \frac{X^2 + Y^2}{(R_0)^2} + C_2 \left(\frac{X^2 + Y^2}{(R_0)^2} \right)^2 \right], \quad R_0 = 3.91 \ \mu\text{m},$$

$$C_0 = 0.207, \quad C_1 = 2.0 \quad C_2 = -1.123.$$

We find that this approximation allows up scaling to hundreds of DPs, while maintaining correct membrane dynamics for accurate emergent blood rheology.

4. Definition of notation

Referring to Fig. 3, we now introduce the following quantities and notations:

Ψ is the index of the face.

$\mathbf{r}(\Psi)$ is the position of the face Ψ, taken to be its center of gravity.

Ψ_1, Ψ_2, Ψ_3 is the nodes defining the face Ψ.

$\mathbf{r}(\Psi_1), \mathbf{r}(\Psi_2), \mathbf{r}(\Psi_3)$ is the position of nodes Ψ_1, Ψ_2, Ψ_3.

$\mathbf{N}(\Psi)$ is the normal to the face Ψ, pointing outside the enclosed volume of the DP.

$A(\Psi)$ is the area of the face Ψ.

$L_{12}(\Psi)$, $L_{13}(\Psi)$, $L_{23}(\Psi)$ is the length between $\mathbf{r}(\Psi_1)$ and $\mathbf{r}(\Psi_2)$, $\mathbf{r}(\Psi_1)$ and $\mathbf{r}(\Psi_3)$, and $\mathbf{r}(\Psi_2)$ and $\mathbf{r}(\Psi_3)$, respectively (note that $L_{\alpha\beta}=L_{\beta\alpha}$).

$\theta_{\alpha\beta}(\mathbf{r})$ is the angle between face Ψ and the face that has node Ψ_α and Ψ_β in common (with $\{\alpha,\beta\}$ in $\{\{1,2\},\{1,3\},\{2,3\}\}$).

$\Psi_{\overline{\alpha\beta}}$ is the node of face Ψ other than Ψ_α and Ψ_β.

$\mathbf{w}_\alpha$ is the vector from centroid of face to node Ψ_α $[\mathbf{w}_\alpha = \mathbf{r}(\Psi_\alpha) - \mathbf{r}(\Psi)]$.

5. Constant particle volume constraint

The inner volume of the DPs is held essentially constant at all times:

$$V(t) = V_0. \qquad (10)$$

The incompressible fluid encapsulated within the DP maintains a uniform internal (hydraulic) fluid pressure on the inner membrane surface (see, e.g., Dao *et al.* [17]); there is therefore no fluid pressure drop across the membrane:

$$P(\mathbf{r},t)|_{outside} = P(\mathbf{r},t)|_{inside}. \qquad (11)$$

Consequently, we use a very stiff Hookean "spring" approach to describe how a DP reacts to a change of inner volume. Each of its faces, Ψ, is subjected to the force $\mathbf{F}^V(\Psi)$ as follows:

$$\mathbf{F}^V(\Psi) = -K^V \frac{V-V_0}{V_0} A(\Psi)\hat{\mathbf{N}}(\Psi), \qquad (12)$$

where we compute the inner volume by

$$V = \sum_\Psi A(\Psi)\hat{\mathbf{N}}(\Psi) \cdot \mathbf{h}(\Psi), \qquad (13)$$

where $\mathbf{h}(\Psi)$ is the vector from the center of the face $\mathbf{r}(\Psi)$ to any plane which does not cross the cell (we chose to use the x-y plane). The force $F^V(\Psi)$ is distributed equally onto each of Ψ's nodes, Ψ_α ($\alpha = 1\ldots3$),

$$F^V(\Psi_1) = F^V(\Psi_2) = F^V(\Psi_3) = \frac{F^V(\Psi)}{3}. \qquad (14)$$

We chose to add area dependence to Eq. (13) in order to account for possible local variation of area. We found that this restrains the inner volume to within $\pm 2\%$.

6. Particle membrane area constraint

We impose the area constraint through the force

$$\mathbf{F}^A(\Psi_\alpha) = -K^A \frac{\Delta A(\Psi)}{A(\Psi)} \hat{\mathbf{w}}_\alpha, \qquad (15)$$

where K^A is the elastic modulus of area dilatation, $\Delta A(\Psi)$ is the change in area of face Ψ, and $\hat{\mathbf{w}}_\alpha$ is the unit vector from the centroid of the face $\mathbf{r}(\Psi)$ to the node Ψ_α ($\alpha=\{1,2,3\}$). This force has the effect of (homocentrically) shrinking or expanding the face.

In the case of a phospholipid plasma membrane of the RBC or the envelope of a vesicle, only a relatively small amount of energy is required to reorganize the molecular distribution locally to accommodate a local change of area. This provides a certain "fluidity" to the membrane, which is constantly undergoing changes in shape. Local stresses resulting in a change of local area are easily accommodated by a fluid reorganization of the membrane [47]. This fluidity is modeled by modifying Eq. (15) to conserve global area strongly, but local area only weakly,

$$\mathbf{F}^A(\Psi_\alpha) = -\left(K_L^A \frac{\Delta A(\Psi)}{A(\Psi)} + K_T^A \frac{\Delta A_T}{A_T} \right)\hat{\mathbf{w}}_\alpha, \qquad (16)$$

where the subscript L denotes local and T the total area and $K_T^A \gg K_L^A$. We note that more complicated fluidization methods do exist (see Kroll [47], for example). We also note that Eggleton *et al.* report that the common Evans and Skalak membrane model used for RBCs ([41]) has a very high dilatation modulus, giving rise to instability in their model [45]. The same authors showed that a much lower dilatation modulus was equally valid, provided that total membrane area is conserved to within $\pm 1\%$ [45]. We calibrated our dilatation coefficient in a similar way: the local area of our DP was found to vary by less than 10% under normal conditions, and the total area varied by only 2% under the most severe conditions.

7. Particle surface strain constraint

The stretching force between nodes Ψ_α and Ψ_β, at positions $r(\Psi_\alpha)$ and $r(\Psi_\beta)$, respectively, can be written

$$\mathbf{F}^S(\Psi_\alpha, \Psi_\beta) = -\mathbf{F}^S(\Psi_\beta, \Psi_\alpha) = K^S(\lambda_{\alpha\beta})\frac{\Delta L_{\alpha\beta}}{L_{\alpha\beta}^0}\hat{\mathbf{L}}_{\alpha\beta}, \quad (17)$$

where K^S is the in-plane shear modulus [see Eq. (19)], $\hat{\mathbf{L}}_{\alpha\beta}$ is the unit vector from $\mathbf{r}(\Psi_\alpha)$ to $\mathbf{r}(\Psi_\beta)$, and $\lambda_{\alpha\beta}$ is the stretch value between Ψ_α and Ψ_β defined by

$$\lambda_{\alpha\beta} = \frac{L_{\alpha\beta}}{L_{\alpha\beta}^0}. \qquad (18)$$

It is should be noted that the stretching force due to the spring does not depend directly on the distance between the nodes. K_S has units of N/m (recalling that the membrane is considered to have no thickness, this length dependency accounts for the amount of material on the ring perpendicular to the direction of stretching, see Sec. III A). The neo-Hookean (hyperelastic) behavior of the RBCs membrane is imposed through the spring constant K^S following Evans and Skalak [17,41]:

$$K^S \rightarrow K^S(\lambda_{\alpha\beta}) = \frac{(\lambda_{\alpha\beta})^{0.5} + (\lambda_{\alpha\beta})^{-2.5}}{\lambda_{\alpha\beta} + (\lambda_{\alpha\beta})^{-3}}. \qquad (19)$$

Our model does not allow rupture of the membrane. Ultimately, this results in a lack of stability for extremely high membrane stresses.

8. Local curvature constraint: Bending

We imposed bending moments proportional in magnitude to the departure from preferred local curvature, but obtained unphysical cell behavior at high shear: the RBC buckled and folded when the radii of curvature became small, leading to instability. Eggleton *et al.* reported precisely the same limitation [45] using their similar approach. We therefore choose to impose a preferred angle between faces. This provides more spatial information for the (on average) six faces at typical vertices, especially on a coarse mesh. We compute the departure from rest angles θ^0 between faces from

$$\mathbf{F}^B(\Psi_{\overline{\alpha\beta}}) = \left(K^B \frac{\Delta\theta_{\alpha\beta}}{\theta^0_{\alpha\beta}} \right) \hat{\mathbf{N}}(\Psi), \qquad (20)$$

where K^B is the coefficient of bending stiffness. See Sec. II C 4 for a definition of the other symbols. An equal and opposite force is distributed to two nodes α and β,

$$\mathbf{F}^B(\Psi_\alpha) = \mathbf{F}^B(\Psi_\beta) = -\frac{1}{2}\mathbf{F}^B(\Psi_{\overline{\alpha\beta}}). \qquad (21)$$

This force ensures that the local and total stress on the membrane are unchanged by the constraint of bending, while permitting a local net rotation.

D. Coupling of DP and fluid dynamics

A range of strategies have been used to couple the Eulerian fluid mesh (position $\mathbf{x}$) and the Lagrangian particle mesh (position $\mathbf{r}$). Jadhav *et al.* [48] weight the Eulerian lattice fluid velocities by the function $\delta_h(\mathbf{r}-\mathbf{x})$, aiming to cut off the part of the fluid which is too distant to interact (typically more than two Eulerian lattice spacings) and provide a distance weight of cosine type. Our recurrent motive of simplicity favors the choice of a simple weighted average for the interpolation between the Lagrangian, DP surface nodes (r, at position $\mathbf{r}$) and the Eulerian, LB, fluid nodes (x at position $\mathbf{x}$). Essentially, each velocity and/or force component has a weight inversely proportional to the cuboidal volume defined by its position and the LB node to which it is closest,

$$\mathbf{v}(r) = \sum_{\mathbf{x} \in \Xi(r)} \mathbf{v}(\mathbf{x})\Gamma(\mathbf{r},\overline{\mathbf{x}}), \qquad (22)$$

where (see Fig. 4) $\overline{\mathbf{x}}$ represents the diagonally opposite vertex to $\mathbf{x}$, $\Gamma(r,\mathbf{x})$ is the volume of the cube defined by $\mathbf{x}$ and $\mathbf{r}$ which has all its segments parallel to an axis of the geometry, $\Xi(r)$ represents the vicinity of DP mesh node r, defined as the LB cube which contains it (see Fig. 4), and $\mathbf{v}(\mathbf{x})$ is the LB velocity at $\mathbf{x}$, defined by Eq. (3). It should be noted that we have the following relationship:

$$\sum_8 \Gamma(\mathbf{r},\mathbf{x}_i) = \sum_8 \Gamma(\mathbf{r},\overline{\mathbf{x}}_i) = 1; \qquad (23)$$

see Fig. 4.

Similarly, the force on a fluid node is given by

○ Eulerian LB lattice

● Lagrangian RBC node to which the velocity is interpolated
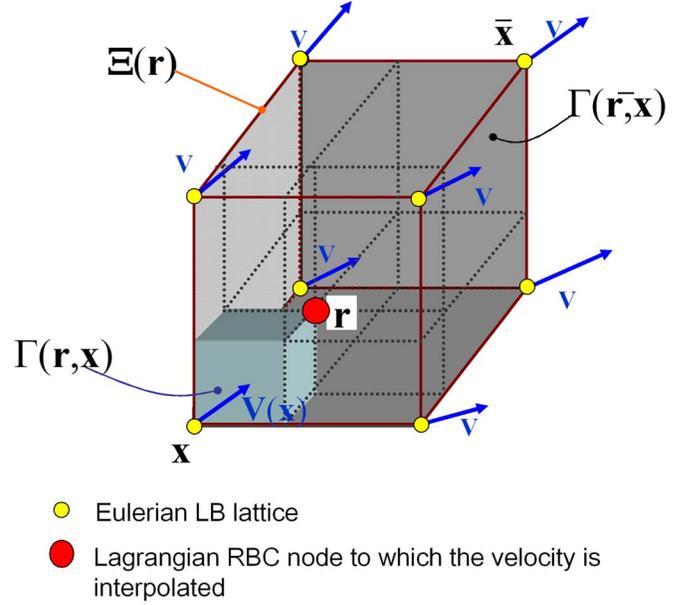
FIG. 4. (Color online) Interpolation of the velocity of the DP surface mesh node r from the eight surrounding LB fluid nodes (including $\mathbf{x}$), contained in the volume $\Xi(r)$. $\Gamma(\mathbf{r},\mathbf{x})$ is the volume as depicted and Ξ is the LB subdomain encapsulating $\mathbf{r}$.

$$\mathbf{F}(\mathbf{x}) = \sum_r \mathbf{F}(\mathbf{r})\Gamma(\mathbf{r},\mathbf{x})\delta(\mathbf{r},\mathbf{x}), \qquad (24)$$

where $\mathbf{F}(\mathbf{r})$ is the total force acting on the node (after computing the membrane stresses) and $\delta(\mathbf{r},\mathbf{x})$ is the function cutting off nodes too distant from $\mathbf{r}$ to be included in a computation of the average:

$$\delta(\mathbf{r},\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{r} \text{ is within } \Xi(\mathbf{x}), \\ 0 & \text{otherwise}, \end{cases}$$

where $\Xi(\mathbf{x})$ is the cube comprising the eight cubes which have $\mathbf{x}$ as a summit (see Fig. 5).

Viscoelastic properties of the DP membrane are not included in this implementation of the model because no models against which we benchtest our method account for it. Viscoelasticity may, however, be easily incorporated, following Bagchi's simple and elegant approach [13], by adding

$$F^{visc}(\Psi_\alpha) = \sum_\beta K^{visc} \frac{1}{\lambda_{\alpha\beta}} \frac{d\lambda_{\alpha\beta}}{dt} \qquad (25)$$

to the total force on the DP membrane.

III. MODEL VALIDATION

As discussed in Sec. II C 5 and II C 6, our model conserves both total DP surface area and volume to a good approximation. Thus, the remainder of this section deals with the validation of the two remaining DP constraints, namely stretching and bending.

A. Validation of stretching properties

Landau and Lifshitz [49] showed that, for a spherical shell with two opposite forces applied at its poles, the stress dis-
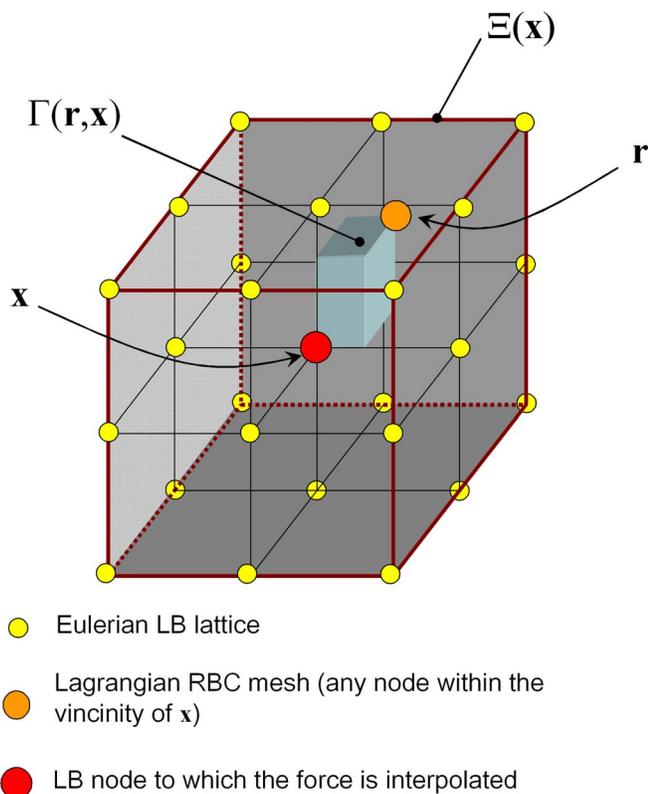
FIG. 5. (Color online) Interpolation of the force on the LB fluid node **x** from the DP surface mesh node **r**. $\Gamma(\mathbf{r},\mathbf{x})$ is the volume as depicted and $\Xi(\mathbf{x})$ is the LB subdomain encapsulating **x**.

tribution is unchanged along any latitude and is given by

$$T_x = \frac{F}{2\pi R(x)}, \qquad (26)$$

where $R(x)$ is the transverse radius of curvature at the longitudinal coordinate x ($x=0$ chosen to be the center of the spheroid) and F is the force applied to the sphere to stretch it.

We stretched the spherical mesh (see Fig. 2) by applying an equal and opposite force at each of its poles. The maximum strain was 120%, in order to remain in Landau and Lifshitz's small deformation regime. Longitudinal stress $T(x)$ was measured at steady state. It is important to note that the strain forces were the only forces applied to the membrane for this particular experiment; that is, $K_L^A = K_T^A = K^V = K^B = 0$. This represents a capsule with no bending dependence.

Figure 6 shows the expected inverse relationship between the local longitudinal stress T_x and the local circumference, $2\pi R(x)$, as shown experimentally by Hénon *et al.* [50] and studied analytically by Landau and Lifshitz [49]. The simulation size was $50 \times 20 \times 20$ lattice nodes, the LB relaxation parameter was equal to 1, the sphere had an initial radius of 4 lattice units and was stretched (in the x direction) at a rate of 0.05% per time step to limit viscous interactions. In this paper, all simulations ran in parallel on a Hewlett-Packard (HP) cluster DL145, running AMD-Opteron 32-64 bit processors with 4 GB random access memory (RAM) each. This particular simulation ran on 5 CPUs and required only
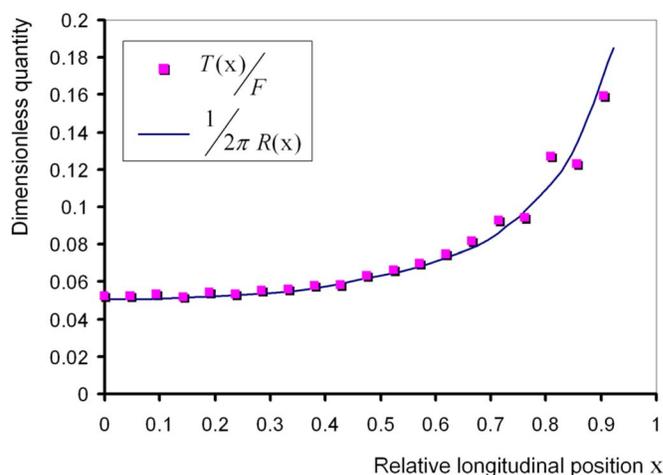
FIG. 6. (Color online) Longitudinal local stress T_x and lateral inverse radius $1/R(x)$ of a spheroidal vesicle stretched by two equal and opposite forces of magnitude F at its two poles. The abscissa is the normalized position on the hemisphere (0/1 corresponding to the *central* and/or *extreme* positions). The measurements were taken at steady state, with a total longitudinal stretch of 120%, in order to remain in the small deformation regime. The only force applied on the vesicle membrane was the strain force, $F^S(\Psi_\alpha, \Psi_\beta)$. T_x was normalized by F. The oscillations observed near $x \to 1$ (corresponding to the extremity of capsule) are most certainly due to the discretization of the mesh: the number of mesh nodes decreases significantly towards the extremity, hence the histogram bin used to calculate at x is subjected to more noise.

one minute to reach equilibrium (approximately 1000 time steps). T_x in Fig. 6 has been normalized by F and $R(x)$ by 2π. This validates the stretching behavior of our DP membrane.

B. Validation of bending properties

This section deals with the dynamics of a vesicle. Kraus *et al.* [15] showed analytically and computationally that the
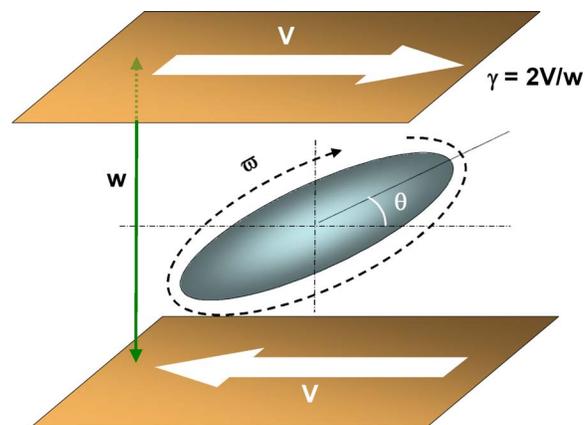
FIG. 7. (Color online) Diagram of the configuration used by Kraus *et al.* [15]. A neutrally buoyant vesicle is sheared horizontally at a shear rate γt and exhibits a tank-treading motion with, at steady state, an inclination θ and an average rotation velocity, ω.

TABLE I. Set of lengths for the major axes of an ellipsoid $\{a,b,c\}$ and corresponding reduced volume v. a, b, and c were chosen such that the ellipsoid was not significantly flat in any direction.

v	1	0.95	0.9	0.85	0.8	0.75	0.7
a	4	3.5	3	2.6	2.2	2	1.7
b	4	4	5.4	5.4	5.4	5.1	4.3
c	4	5.5	5.5	5.5	5.5	5.7	6

stationary state of a vesicle ($K_S=0$, $K_V>K_A^T>K_A^L>K_B>0$) in a linear shear flow is characterized by (i) its finite inclination angle, θ, subtended at the longest axis of inertia of the vesicle and the flow direction, and (ii) the revolution frequency, ω, of the "tank-treading" tangential motion of the membrane around the vesicle; see Fig. 7.

Both θ and ω depend strongly on the reduced volume $v=V/(4\pi R_0^3/3)$ of the vesicle, where V is the enclosed volume and R_0 is determined by $R_0=\sqrt{A/4\pi}$. Interestingly, θ and ω do not depend on the reduced shear rate $\chi \equiv \gamma \eta R_0^3 / K^B$. The vesicle mesh (Fig. 1) was inflated to an ellipsoidal shape with parameters $\{a,b,c\}$ (lengths of the three major axes of the ellipsoid). We found that interchanging a, b, and c did not affect θ or ω. Table I gives the sets of $\{a,b,c\}$ to control v in the range $0.7\ldots1$, with an incremental step of 0.05.

The simulation size was $50\times30\times30$ lattice nodes, the fluid relaxation parameter was $\tau=1$, the vesicle's parameters were taken from Table I, and the top and bottom shear velocities were 0.1 (shear rate of 0.003 lattice units). The simu-
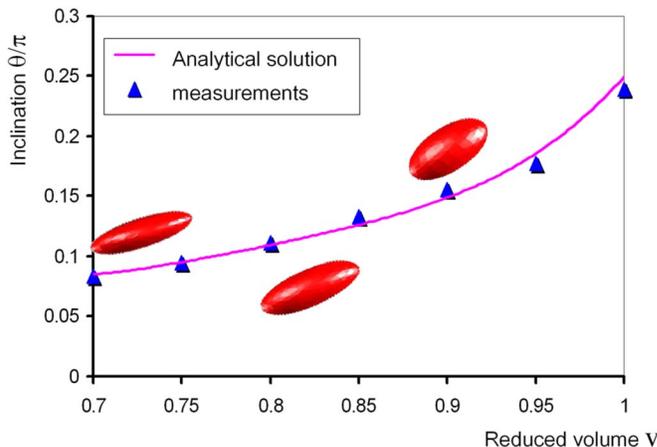
FIG. 9. (Color online) Inclination angle, θ, measured between the direction of shear and the longest axis of inertia, as a function of the reduced volume v. θ was normalized to π. The continuous solid line is the predicted analytical value in Ref. [15].

lation ran on 5 CPU's and required only 4 minutes to reach steady tank-treading equilibrium (approximately 2000 time steps). We obtained the steady state elliptical profiles predicted by Kraus *et al.* [15], see Fig. 8. The corresponding angle of inclination, θ, was calculated from an ellipsoidal fit on a profile crossing the ellipse at its center (observable on Fig. 8), parallel to the direction of the shear. It was found that θ follows the theoretical predictions given in [15], see Fig. 9. Finally, the revolution frequency ω of the sheared vesicle, defined by the average angular velocity of each node, follows the theoretical prediction given in [15], see Fig. 10.

(a) $v=0.7$, $\theta=15°$.　　(b) $v=0.75$, $\theta=17°$.　　(c) $v=0.8$, $\theta=21°$.　　(d) $v=0.85$, $\theta=24°$.

(e) $v=0.9$, $\theta=28°$.　　(f) $v=0.95$, $\theta=32°$.　　(g) $v=1.0$, $\theta=43°$.
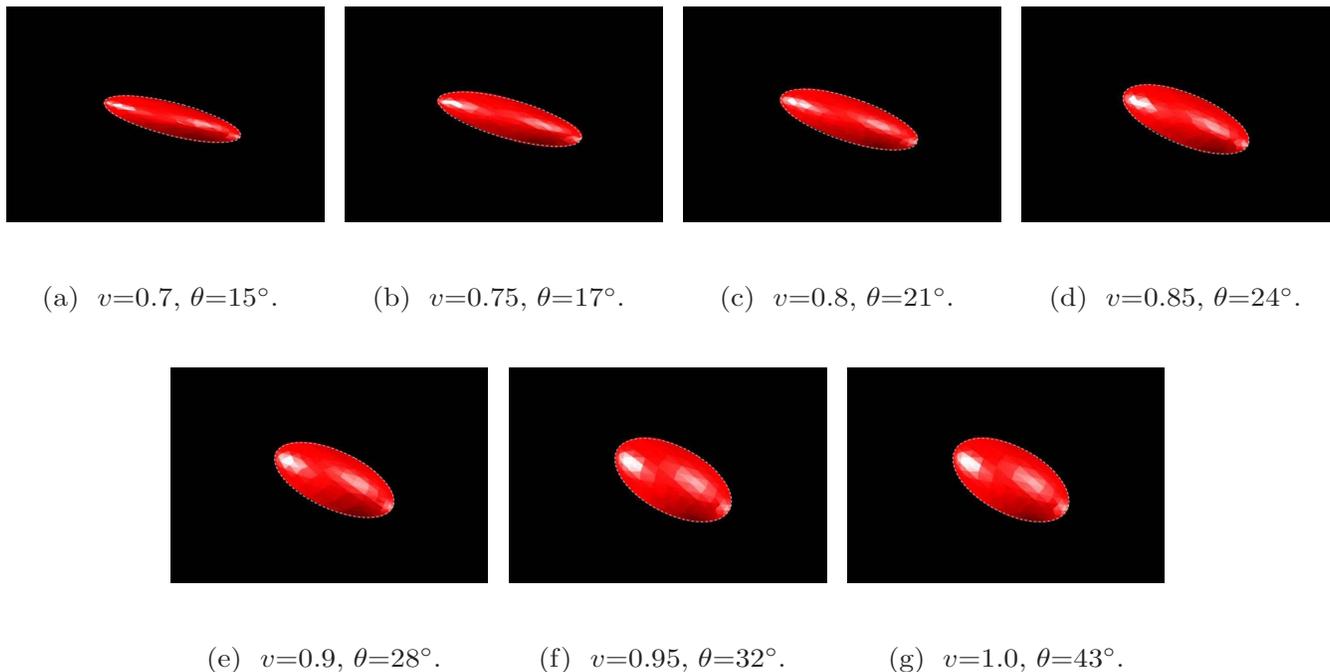
FIG. 8. (Color online) Snapshots showing the steady state (tank-treading) orientation and shape of an ellipsoidal vesicle for various reduced volume v. The shear rate did not vary between simulations. The angle of view of was chosen to show the elliptical profile of the vesicles predicted by [15]. The forces applied were $\mathbf{F}^A(\Psi)$, $\mathbf{F}_\alpha^B(\Psi)$, and $\mathbf{F}_\alpha^V(\Psi)$, area, bending, and volume, respectively. The fitted ellipsoidal profiles are shown with the dashed white line.
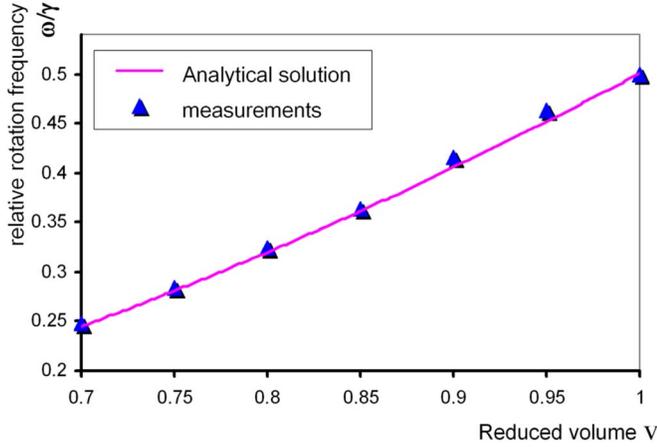
FIG. 10. (Color online) Measured angular velocity of the vesicle averaged over all nodes, ω, as a function of different reduced volumes v. ω was scaled to the shear rate γ. The continuous solid line is the predicted analytical value by [15].

Kraus *et al.* [15] reported good agreement with analytical predictions using a finite element model of a vesicle in shear flow, having only area and bending constraints. We obtained accuracy similar to that theoretical model with our simple method, validating the bending property of our membrane.

IV. MODEL CALIBRATION AND RBC BEHAVIOR

A. Calibration

We adopt the following notation:

$$\xi = \bar{\xi}\hat{\xi}, \tag{27}$$

where ξ represents the physical parameter of interest in *Systéme International* (SI), $\bar{\xi}$ is the conversion factor in SI units per lattice unit, and $\hat{\xi}$ represents the physical parameter of interest in lattice units.

The calibration of lattice Boltzmann starts with the calibration of length ([26,51]). Both a typical length of interest and the number of regularly spaced grid points allocated to represent it must be known (we take RBC dimensions). Therefore, $\bar{x} = x/\hat{x}$. Time calibration is derived from the (fixed) viscosity of the suspending fluid: $\nu = (\bar{x})^2/\bar{t}\hat{\nu}$, where $\hat{\nu} = c_s^2(2\tau - 1)$ [see Eq. (2)]. Similarly, mass is derived from the density of the fluid: $\rho = \bar{m}/\bar{x}^3\hat{\rho}$.

In the particular spatial resolution of the application of this paper (1 lattice spacing per μm, $\rho = 1000$ kg/m^3, and $\nu = 10^{-6}$ m^2/s), we have

$\bar{x} \equiv 10^{-6}$ m/l_s,

$\bar{t} \equiv 1/6 \times 10^{-6}$ s/l_t,

$\bar{m} \equiv 10^{-12}$ kg/l_m,

$\bar{F} \equiv 4 \times 10^{-5}$ kg m s^{-2}/$l_m l_s (l_t)^{-2}$,

$\bar{v} \equiv 6$ m s^{-1}/$l_s(l_t)^{-1}$, $\bar{\gamma} \equiv 6$ s^{-1}/$(l_t)^{-1}$,

where l_s is the lattice spacing, l_t is the lattice unit of time, and l_m is the lattice unit of mass. Note that we used the RBC stretching configuration of the next section to calibrate the force with the quantity ϕ_i of Eq. (1).
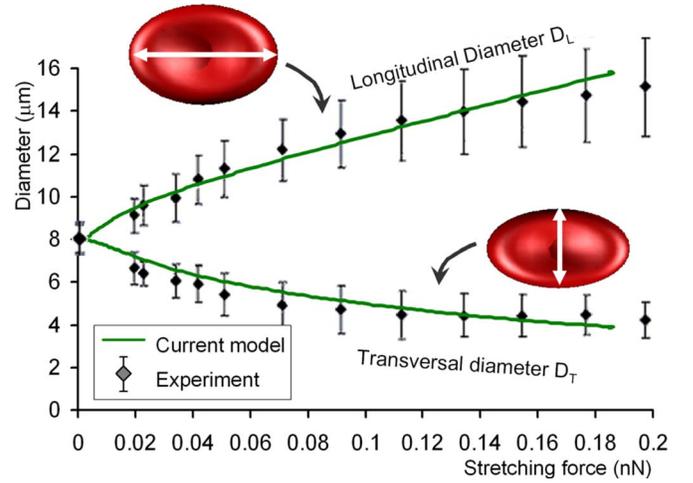
FIG. 11. (Color online) Longitudinal (D_L) and transverse (D_T) diameters of the stretched RBC as a function of the stretching force F. The black diamonds along with error bars are the experimental measurements by Dao *et al.* [17].

B. RBC deformation

Sections III A and III B showed a validation of the mechanics of a capsule and of a vesicle respectively; this section demonstrates a combination of the two: a RBC for which all four constraints apply.

Dao *et al.* [17] compared real RBC stretching experiments with a high resolution deformation model. The setup was to stretch a RBC by fixing two diametrally opposed clamps and measure the force required as a function of the stretch. Their strain-energy potential model involves a fine grid with an impressive mesh size comparable to the actual cell cytoskeleton's mesh size ($\Delta l \approx 75$ nm, see Fig. 1) in order to represent the RBC membrane; however, the model lacks fluid dynamics. The group could successfully compare their model with experiments, since the application of force in the experiment allowed the membrane to achieve a series of steady states. In this regime, a RBC does not induce significant flow by its deformation because the rate at which the RBC is stretched (≈ 10 μm per second) is too slow to induce any significant fluid stresses compared to the membrane stresses.

Our simulation size was $50 \times 20 \times 20$ lattice nodes, the fluid relaxation parameter was taken to be $\tau = 1$, the RBC had an initial radius of 4 lattice units and was stretched (in the x direction) at a rate of 0.05% per time step, to limit the influence of deformation on flow (for consistency with Dao's experiment and model). The simulation ran in parallel on 5 CPU's and required only 2 minutes to reach a deformation of 200% (2000 time steps).

Our results agreed well with those of Dao *et al.* [17] for both the maximum transversal diameter, D_T, and longitudinal diameter, D_L as the RBC was stretched by increasing force, F; see Fig. 11.

We stress that successful comparison of our model against this experiment (consisting of a series of equilibrium states) only validates the mesh mechanics. We address validation of our model's dynamics in Sec. IV D.

Mills *et al.* [52] used Dao's computational model [17] to provide the local maximum principal stretch ratio of the

Experimental picture   Current model   Dao's model (fine mesh)

F=0 pN

F=67 pN

F=130 pN

F=193 pN
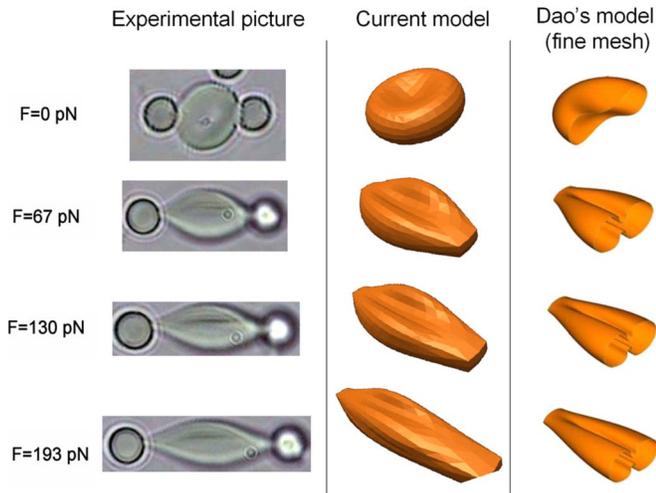
FIG. 12. (Color online) Deformation of a RBC stretched along its poles. The first column is experimental data from Mills *et al.* [52], the second column is from the present method, and the third column is from a finer mesh model by Dao *et al.* [17]. Our results in this series of stretching simulations agree closely with those from both experimental and numerical studies. The simulation configuration used for these data was identical to that of Fig. 11.

RBC under stretch. Figures 12 and 13 show that our model quantitatively recovers Dao's results very well. A remarkable point is that the distribution of stress is (on average) dependent on the axial radius of the RBC: the highest stress is near the poles (where the clamps are), where the axial radius is the smallest. The stress is the least in the middle of the RBC, where the radius is the greatest. This, indeed, correlates with the Landau and Lifshitz theoretical model (see previous section).

Current model   Dao's model (much finer mesh)

F=0 pN

F=67 pN

F=130 pN

F=193 pN

maximum principal strain

0%   20%   40%   60%   80%   100%   120%
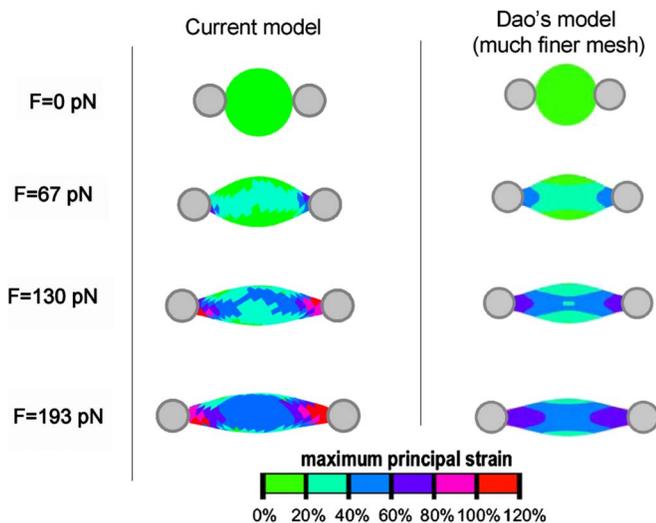
FIG. 13. (Color online) Local maximum principal strain of the stretching experiment, Fig. 12. The first column is from our model and the second column is from the finer model by Dao *et al.* [17]. The distribution of stress on the RBC is qualitatively similar in both methods; the difference in mesh coarseness in the two models is obvious here. The slight asymmetry of our measurements are due to the inherent slight asymmetry of our coarse RBC mesh.

RBC is released t_c = 320 lattice time steps
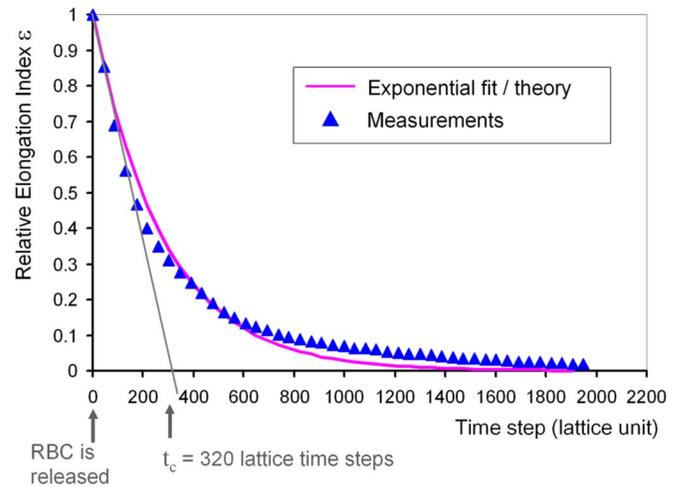
FIG. 14. (Color online) Evolution of the elongation index (ε) during the relaxation to rest, biconcave equilibrium shape of a RBC initially stretched by 200%. ε follows the expected exponential decay, providing a time constant t_c of 320 LB time steps.

C. RBC relaxation to equilibrium: Force parameter calibration

Following Dao *et al.* [17], when the stretching force, F, imposed on the poles of the RBC is released at the point of maximum deformation, the cell returns to its original biconcave shape with a characteristic time t_c. Hochmuth *et al.* [53] proposed the following relaxation dynamics:

$$\varepsilon(t) = \frac{(\lambda_1^2 - 1)(\lambda_{1,max}^2 + 1)}{(\lambda_1^2 + 1)(\lambda_{1_{max}}^2 - 1)} = \exp\left(-\frac{t}{t_c}\right), \quad (28)$$

where λ_1 is the total longitudinal stretch ratio, $\lambda_{1,max}$ is the maximum stretch (200% in this case) and ε is the elongation index. Figure 14 shows the results of our characteristic re-
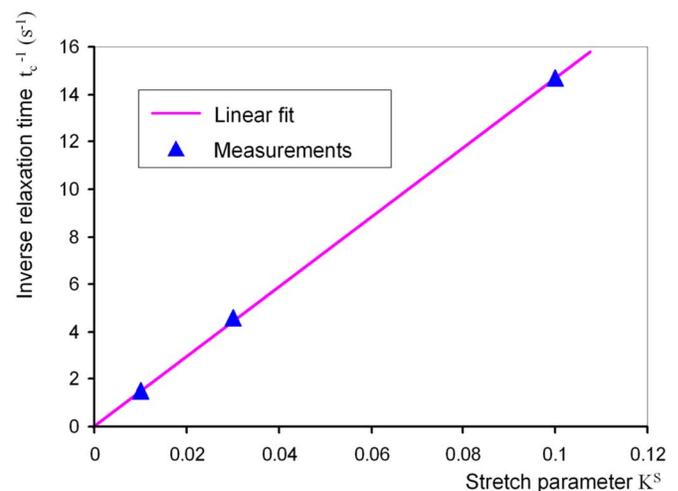
FIG. 15. (Color online) Dependence of the inverse characteristic relaxation time $(t_c)^{-1}$ with the stretch modulus of the membrane K^S. Dao *et al.* [17] reported an experimental relaxation time of t_c^{exp} =0.12±0.05 s, corresponding to $(t_c)^{-1}$=8.3 s^{-1}. This calibrates the stretching modulus k^S to k^S=0.03.
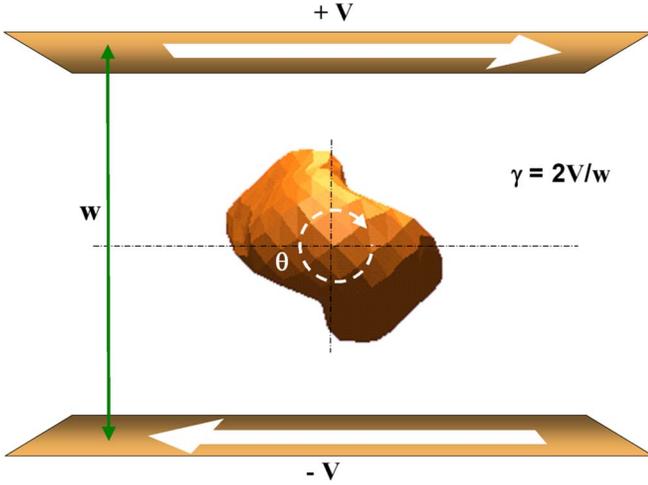
FIG. 16. (Color online) Tumbling RBC from linear shear impressed horizontally. A sheared RBC (1) tumbles with no significant deformation below a shear rate of approximately 10 s^{-1}, and (2) tank treads at higher shear rates (the membrane rotates around the highly deformed, flat ellipsoidal cell). We therefore chose to investigate the tumble and/or deformation of a RBC at this transition shear rate. See [54], for example, for more experimental details.

FIG. 17. (Color online) Profiles of a neutrally buoyant tumbling RBC from a linear shear impressed at top and bottom. The snap shots are at dimensionless time γt=0, 4, 8, and 11, respectively. The shear rate γ is 10 s^{-1}, approximately at the tumbling and/or tank-treading transition value found experimentally (see Skalak and Chien [54], for example).

laxation measurements along with the best exponential fit. We found that the characteristic time of relaxation t_c was 320 time steps. We also found that the inverse of the characteristic relaxation time, t_c, for relaxation to equilibrium shape for the RBC, is linearly dependent on the stretch parameter K^S, see Fig. 15. Dao *et al.* [17] found experimentally that t_c =0.19±0.06 s. With this information and the calibration of Fig. 15, we have K^S=0.03 for healthy RBC's. Furthermore, Dao *et al.* reported that a force of 140 pN is necessary to stretch a RBC longitudinally by 75% (see Fig. 11). This provides a calibration of the force applied on the RBC's pole, through the quantity ϕ_i of Eq. (1), see Fig. 15. We therefore have the following parameters for healthy human RBC's (in LB units), $\hat{K}^S$=0.03, $\hat{K}^A_L$=0.01, $\hat{K}^A_T$=0.1, $\hat{K}^V$=0.3, $\hat{K}^B$=0.006.

D. Deformable particle dynamics

Only the dynamic behavior of the DPs remains to be validated. To this end, we first compare the shear-induced tumbling of our RBC with that of a solid ellipse.

1. RBC dynamics

We use a conformation identical to that of Pozrikidis [40]; see Fig. 16.

Our simulation consists of a single, neutrally buoyant RBC in a linear shear flow. The long axis of symmetry of the RBC is perpendicular to the flow. Parameters were taken from the previous section, representing a healthy RBC.

The simulation size was $50 \times 30 \times 30$ lattice nodes, the LB fluid relaxation parameter was τ=1, the RBC had an initial radius of 4 lattice units. The shear rate was 10 s^{-1}. The simulation ran in parallel on 5 CPU's and required only 3 minutes for each tumble (4000 time steps). A shear rate of approximately 10 s^{-1} provides the most dramatic RBC defor-

mation. A RBC exposed to shear rates below this level tumbles without significantly deforming; above this level, the RBC tank treads with no apparent variation of its shape (similar to the sheared vesicle of Fig. 19).

We obtained good qualitative agreement with the simulations of Pozrikidis [40], see Fig. 17. Note that (i) Eggleton and Popel [44] reported significant stability issues with their model in this configuration, making it impossible to simulate a full revolution; and (ii) Pozrikidis [40], in 2003, required of the order of a week of CPU time for one tumble in linear shear. In our simulations, one tumble required only 3 minutes of processing time on 5 CPUs; moreover, there were no stability issues in this regime.

2. Ellipse dynamics

Following the example of Aidun *et al.* [55], we performed additional, more quantitative tests of our model's dynamics by simulating the motion of a solid ellipse at vanishing Re (see below), with one of the principal axes constrained parallel to the vorticity vector of the shear flow (the ellipsoid rotates around this axis), see Fig. 18. In this section, the Reynolds number is defined

$$\text{Re} = \frac{\gamma d^2}{\nu}, \qquad (29)$$

where γ is the shear rate, ν is the viscosity of the fluid, and d=2b where b is the length of the major axis (radius) of the ellipsoid.

Under present conditions, Jeffery ([56]) showed that the angle of rotation $\theta_{ellipse}$ and the angular rate of rotation $\dot{\theta}_{ellipse}$ are given by

$$\theta_{ellipse} = \tan^{-1}\left[\frac{b}{c}\tan\left(\frac{bc}{a^2+b^2}\gamma t\right)\right], \qquad (30)$$
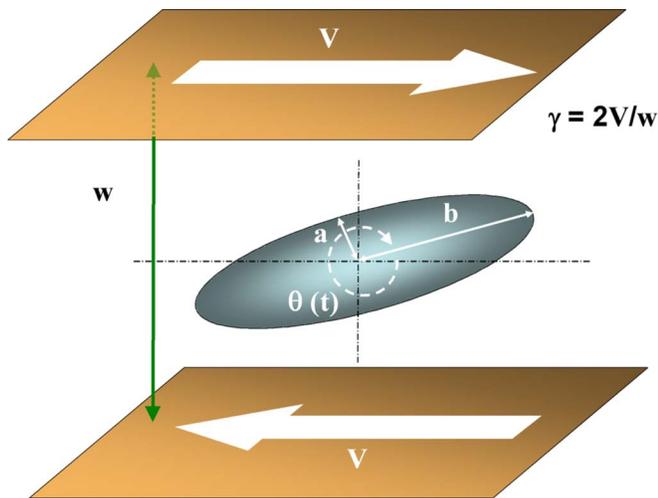
FIG. 18. (Color online) Tumbling of a solid ellipse in response to a linear shear impressed at top and bottom. The ellipse has an instantaneous orientation $\theta(t)$, rotation speed $\dot{\theta}(t) = d\theta/dt$, and $b(a)$ the major (minor) axis. The stability was compromised at the highest shear rates (corresponding to $\mathrm{Re} \approx 50$) due to the necessity for large forces applied to the membrane in order to simulate a "solid" ellipse (by design, our particles have deformable membranes). This stability problem is, however, not an issue in the range of Re and under the physiological conditions for which the model is intended (where the stresses on the cells are orders of magnitude lower).

$$\dot{\theta}_{ellipse} = \gamma \frac{1}{a^2 + b^2} (a^2 \cos^2 \theta + b^2 \sin^2 \theta), \qquad (31)$$

from which the motion of the ellipse has period

$$T_{ellipse} = 2\pi \frac{(a^2 + b^2)}{bc} \frac{1}{\gamma}. \qquad (32)$$

In turn, this means that (a) the rotation period scales linearly with the inverse of the shear rate γ and that (b) there is no difference between the trajectories of a two-dimensional and a three-dimensional ellipsoid, since Eqs. (30)–(32) do not include the parameter c, the remaining major axis of the ellipsoid, parallel to the axis of rotation. We define by $\varsigma(\mathrm{Re})$ the normalized product:

$$\varsigma(\mathrm{Re}) = \frac{T_{ellipse}}{T_{ellipse|low\ Re}} = T_{ellipse} \frac{ab}{2\pi(a^2 + b^2)} \frac{\mathrm{Re}\ \nu}{d^2}. \qquad (33)$$

Note that $\varsigma = 1$ under Jeffery's assumptions.

The simulation size was $50 \times 30 \times 30$ lattice nodes, and the fluid relaxation parameter $\tau = 1$; the ellipse had shape parameters of $a = 1.5$, $b = 4$, and $c = 1.5$. The top and bottom shear velocities were 0.01 (corresponding to a shear rate γ of 8×10^{-4}, and a Reynolds number Re of 0.3, consistent within Jeffrey's assumptions of low Re). The simulation ran in parallel on 5 CPUs and required only 6 minutes for one tumble of the ellipse (4000 time steps).

Figure 19 shows that our model recovers, very well, the analytical angular rate of rotation $\dot{\theta}_{ellipse}$ from Eq. (31). The "pointiness" of the profile we obtained compared to the analytical solution is probably due to tank treading of our ellipse
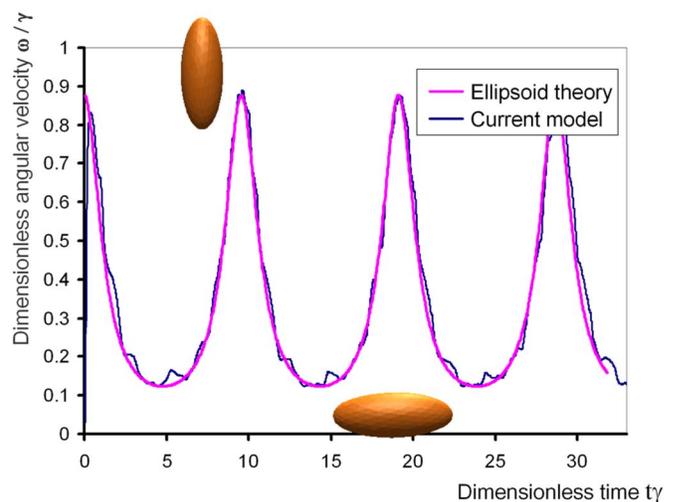
FIG. 19. (Color online) Angular velocity as a function of time for a solid tumbling ellipse in shear (solid line) and RBC (points). The disparity is probably due to some tank treading of the RBC when it is aligned with the direction of the shear.

when the major axis is parallel to the plane of shear (when $\dot{\theta}$ is at its minimum).

Equation (32) shows that, for low Re (Jeffery's assumptions), ς should remain constant and equal to unity, independent of Re. For nonzero Re, Aidun *et al.* [55] observed that ς diverges as $\mathrm{Re}^{-0.5}$, with a significant departure from unity (more than 10%) at Re=5. Figure 20 shows similar behavior, ς remain equal to unity for Re<3 and diverges for Re>3 (approximately). It is interesting to note that we found a log increase while Aidun found an inverse square root increase. The difference is probably due to the deformation and tank
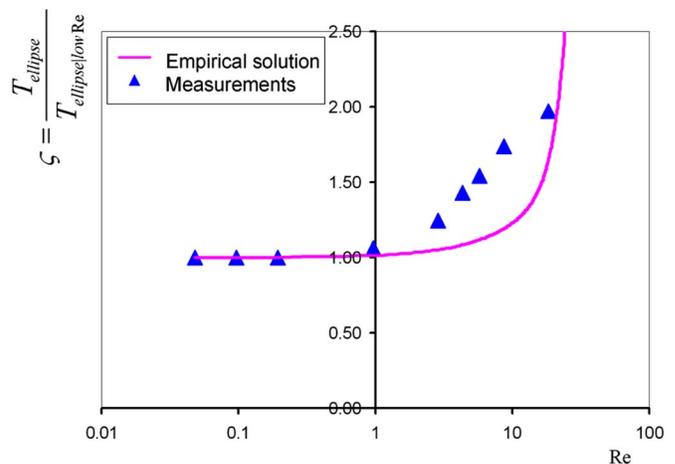
FIG. 20. (Color online) Dependance of ς as a function of Re. Jeffery [56] predicted analytically that $\varsigma = 1$ for vanishing Re, and Aidun *et al.* [55] observed with their simulation that ς remains constant until $\mathrm{Re} \approx 5$ and then diverges. Our model provides solutions consistent with both these regimes. The difference in divergence behavior is probably due to the fact that our model ellipse deforms and tank treads at high Re (our model's computational stability required lowering the stiffness of the ellipse as Re increased).

FIG. 22. (Color online) Snapshot of the evolution of 200 RBC's in flow, initialized on a regular lattice. The RBCs collide, interact, and deform in a physical manner. Qualitatively, the disorder and mixing of the RBC's reaches steady state after 0.45 s. The total computational time was only 10 hours on 15 CPUs.

FIG. 21. (Color online) Caption on Fig. 22.

treading of the RBC: the method we use models, by design, deformable objects and therefore is not efficient at maintaining solid shapes, especially at high stresses (the ellipse stiffness had to be reduced to comply with the stability constrains that the LB method imposes at high γ). These last series of results (tumbling of deformable RBC or solid ellipsoid) validate the overall dynamics of our model and illustrate its efficiency.

V. PROOF OF CAPABILITY: 200 RBCS IN CONFINED GEOMETRY

Thus far we have presented an efficient approach to simulating the rheology of a single DP. We next show that our model can reproduce flow of a dense solution of DPs, at physiological Re $[O(10^{-2}-10^{-3})]$. To the authors' best knowledge, this level of accuracy is the most complete, accurate, and computationally tractable at this date.

A. Snapshots

As a proof of capability, we simulated 200 RBCs in a rectangular channel. We chose RBCs over vesicles and capsules because the flow they produce is the most interesting. Their biconcavity provides them with a unique tumbling behavior at a moderate shear rate (≈ 10 s^{-1}), undeformed tumbling at lower shear, and tank treading at higher shear. The presence or absence of other cells locally increases or decreases the local shear rate, making the flow of concentrated RBCs highly complicated.

The simulation domain size was $\{length, width, depth\}$ $=\{100, 33, 27\}$. The 200 RBCs where initialized randomly on a rectangular lattice filling the simulation space (holding up to 210 RBCs), with 1 micron spacing between them (one lattice Boltzmann node). The volumetric cell concentration was 30%.

Figures 21 and 22 show the time evolution of the 200 RBCs. Figure 23 shows the velocity profiles averaged along the length of the conduit before steady state and at steady state.

The arrangement of the RBCs reached steady state after approximately 0.45 seconds in real units, or 3 hours of simulation. Running the simulation for three times longer did not produce any significant changes in RBC arrangement or axial velocity profile. This required only 10 hours of computation on 15 CPUs, demonstrating the high efficiency of this approach. The profiles of Fig. 23 exhibit the expected significant blunting occurring in suspension flows, as observed experimentally [57,58]. Varying cell concentration, deformability, viscosity, and other cell parameters would lead straightforwardly to empirically derived, accurate, local rules of the macroscopic flow of such a suspension (which might be used to increase the accuracy of continuum models solving similar flows at a larger length scale).

B. Detailed pressure field: Unprecedented accuracy

In more detail, Figs. 24 and 25 show the pressure fields corresponding to Figs. 21 and 22. The profiles show the pressure in the longitudinal plane bisecting the depth of the simulation, together with the profiles of RBCs cut by this plane. This shows an unexpected behavior: the pressure field varies
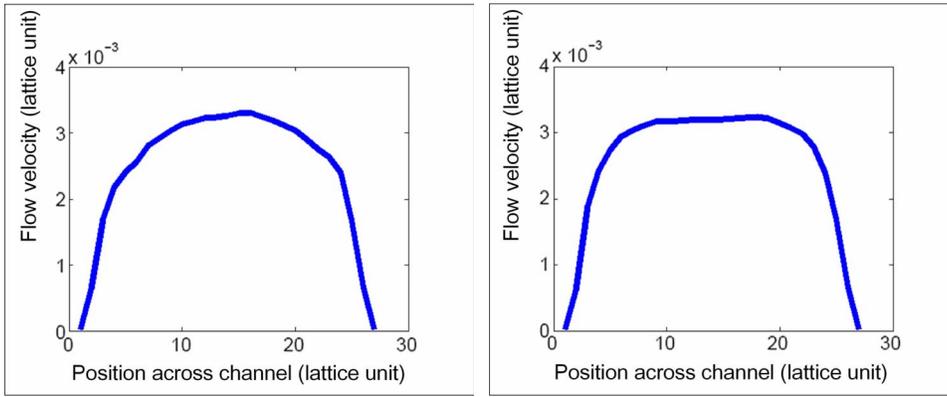
(a) After 0.113s   (b) After 0.450s

FIG. 23. (Color online) Axial in-length velocity profiles of the simulation described in Figs. 21 and 22. The velocity profile reaches steady state (b), approximately at the same time as the RBC's disorder described in Figs. 21 and 22 (after 0.45 s). (a) A transient velocity profile.

significantly—spatially and temporally; a fact that is normally ignored in biological models. A direct consequence could be that cells comprising the vessel wall would be exposed to widely varying pressures and shear forces, potentially affecting the gene expression of these cells.

This is the first time, to our knowledge, that the pressure field within a flowing suspension of deformable particles has been reported explicitly (computationally or experimentally). This kind of application, along with the model's ability to easily adapt to different conduit configurations and particle properties, should provide unprecedented insight into the fluid dynamics of RBCs in concentrated flow.

Finally, it is important to note that lubrication forces are not included in the present implementation of the model. This approximation relies on the fact that the cells were initialized with an explicit layer of fluid surrounding all cells;
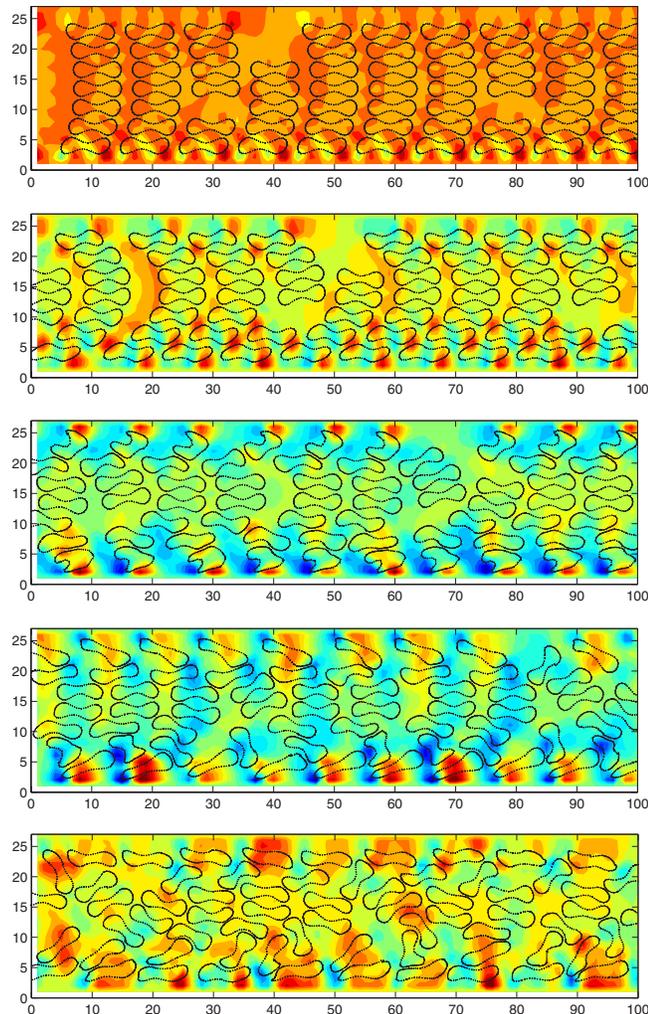
FIG. 24. (Color online) Caption on Fig. 25.
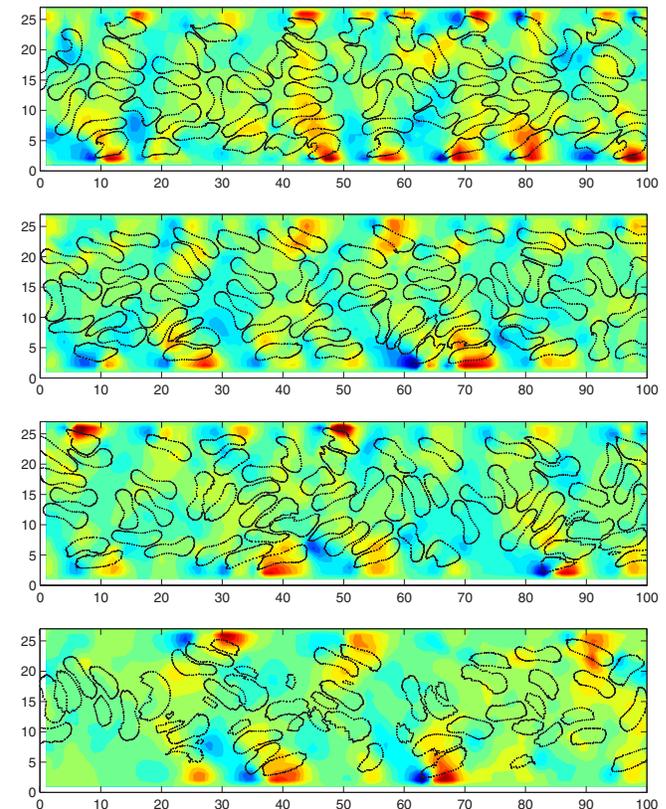
FIG. 25. (Color online) Cross sectional pressure and RBC profiles of 200 RBC's flowing at a concentration of 30%. The same color map has been used for all the pressure profiles in this figure. We found that the pressure varied significantly with time along the wall, which could have some influence on the behavior of the cells comprising it.

TABLE II. Scalability vs geometry size with constant size and/or CPU ratio and number of cells. The lower time requirements in the first column are certainly due to the effect of inter-CPU communication time.

1 RBC Size	$30 \times 30 \times 30$	$50 \times 30 \times 30$	$100 \times 30 \times 30$
No. of CPU	3	5	10
Δt	2.0	2.5	2.5

TABLE III. Simulation overall scalably: increasing geometry size while maintaining cell concentration constant. The plateau shows that the simulation is highly scalable.

Ht=0.1 Size	$30 \times 30 \times 30$	$50 \times 30 \times 30$	$100 \times 30 \times 30$
No. of CPU	4	6	11
Δt	5	7	7.5

because we operate at low Re, we expect that this layer should be maintained at all times. We did notice that the minimum distance separating any two DP nodes regularly crossed the half LB spacing limit (the limit below which the previous assumption no longer holds), probably due to the first order accuracy in the interpolation of the velocities of the DP nodes. We propose, in future work, to adapt and validate Nguyen and Ladd's model (Ref. [59]) for explicit lubrication in LB.

C. Efficiency and scalability of the model

Whether simulating blood flow, dense vesicle flow, or any suspension flow, the main question is how the model scales with the simulation size and number of particles. There are therefore two independent components to the model requirements. (1) The dependence on simulation size is usually linear with the volume of the simulation (total number of nodes). (2) The dependence on the number of particles in turn depends strongly on the essential fundamentals of the method, and the cell-cell interaction rules.

Tables II and III demonstrate the computational dependency of the model by (i) varying the size of the simulation and the number of CPU's accordingly, with only one DP, and (ii) varying the simulation size and the number of CPUs, maintaining the concentration of cells constant (important when the cell concentration is a fixed parameter, as in blood flow, for example). Δt represents the time needed for the simulation to reach 1000 time steps.

These results demonstrate that the model is appropriately linear in the number of cells and geometry, and therefore is highly scalable.

VI. CONCLUSION

In summary, we have developed and validated an approach for simulating the flow a dense suspension of deform-

able particles, with the primary design constraint of efficiency, rather than higher-order accuracy. We chose the lattice Boltzmann method to approximate the full Navier-Stokes equations efficiently and accurately, and a simple harmonic spring model to reproduce the membrane mechanics of the deformable cell. We validated the mechanical properties of each parameter of the model against several models of deformable particles, and the overall dynamics against known deformations. Finally, we demonstrated the power of this method by simulating 200 fully three-dimensional, deformable red blood cells in flow. Thanks to the simplifications made to this model, it is highly efficient and capable of addressing configurations unapproachable with current computationally expensive techniques.

Despite its efficiency, our approach was not designed for scales where the typical length is of the order of 100 cell diameters; in these cases, continuum models are more appropriate. Our model is, on the other hand, ideal for generating accurate rules for implementation in the latter continuum-based models.

Future improvements to the model will be the incorporation of reversible, dynamic RBC-RBC adhesion, and the inclusion of cytoskeletal reorganization, which can result in hysteresis of the stresses and temporary changes in the preferred cell shape. These changes, when applicable, will be easily linked to local properties of the fluid (e.g., a dynamic and local strain resistance also depending on local stress history would better approximate real RBC dynamics).

[1] P. G. H. L. Goldsmith, G. R. Cokelet, and R. Fhraeus, Am. J. Physiol. **257**, H1005 (1989).

[2] L. L. Munn, R. Melder, and R. Jain, Biophys. J. **71**, 466 (1996).

[3] C. Migliorini, Y. Qian, H. Chen, E. Brown, R. K. Jain, and L. L. Munn, Biophys. J. **83**, 1834 (2002).

[4] C. Sun, C. Migliorini, and L. L. Munn, Biophys. J. **85**, 208 (2003).

[5] G. W. Schmid-Schnbein, S. Usami, R. Skalak, and S. Chien, Microvasc. Res. **19**, 45 (1980).

[6] R. Mukhopadhyay, G. Lim, and M. Wortis, Biophys. J. **82**, 1756 (2002).

[7] D. Kuzman, S. Svetina, R. E. Waugh, and B. Žekš, Eur. Biophys. J. **33**, 1 (2004).

[8] C. Pozrikidis, Phys. Fluids **17**, 031503 (2005).

[9] J. Li, M. Dao, C. T. Lim, and S. Suresh, Biophys. J. **88**, 3707 (2005).

[10] H. Noguchi and G. Gompper, Phys. Rev. E **73**, 021903 (2006).

[11] C. H. Sun and L. L. Munn, Physica A **362**, 191 (2006).

[12] K. Boryczko, W. Dzwinel, and D. A. Yuen, J. Mol. Model. **9**, 16 (2003).

[13] P. Bagchi, P. C. Johnson, and A. S. Popel, J. Biomech. Eng. **127**, 1070 (2005).

[14] J. F. Brady and A. Sierou, J. Fluid Mech. **448**, 115 (2001).

[15] M. Kraus, W. Wintz, U. Seifert, and R. Lipowsky, Phys. Rev. Lett. **77**, 3685 (1996).

[16] C. Sun and L. L. Munn, Physica A **362**, 191 (2006).

[17] M. Dao, C. T. Lim, and S. Suresh, J. Mech. Phys. Solids **51**, 2259 (2003).

[18] G. A. Buxton, R. Verberg, D. Jasnow, and A. C. Balazs, Phys. Rev. E **71**, 056707 (2005).

[19] A. Alexeev, R. Verberg, and A. C. Balazs, Soft Matter **2**, 499 (2006).

[20] C. S. Peskin, Acta Numerica **11**, 479 (2002).

[21] Z.-G. Feng and E. E. Michaelides, J. Comput. Phys. **195**, 602 (2004).

[22] S. Succi, *The Lattice Boltzmann Equation for Fluid Mechanics and Beyond* (Clarendon Press, Oxford, 2001).

[23] A. M. Artoli, A. G. Hoekstra, and P. M. Sloot, J. Biomech. **39**, 873 (2006).

[24] M. Krafczyk, M. Cerrolaza, M. Schulz, and E. Rank, J. Biomech. **31**, 453 (1998).

[25] J. Boyd, J. Buick, J. A. Cosgrove, and P. Stansell, Phys. Med. Biol. **50**, 4783 (2005).

[26] M. M. Dupin, I. Halliday, and C. M. Care, Philos. Trans. R. Soc. London, Ser. B **362**, 1885 (2004).

[27] M. M. Dupin, I. Halliday, and C. M. Care, J. Phys. A **36**, 8517 (2003).

[28] S. Hou, S. C. Q. Zou, G. D. Doolen, and A. C. Cogley, J. Stat. Phys. **118**, 329 (1995).

[29] D. T. T. Michael and C. Sukop, *Lattice Boltzmann Modeling—An Introduction for Geoscientists and Engineers* (Springer, Berlin, 2006).

[30] Y. H. Qian, D. d'Humières, and P. Lallemand, Europhys. Lett. **17**, 479 (1992).

[31] P. E. Bhatnagar, P. E. Gross, and M. Krook, Phys. Rev. **94**, 511 (1954).

[32] R. C. Liboff, J. Herman, and R. L. Liboff, *Kinetic Theory*, 2nd ed. (Springer, Berlin, 2003).

[33] Z. Guo, C. Zheng, and B. Shi, Phys. Rev. E **65**, 046308 (2002).

[34] A. J. C. Ladd and R. Verberg, J. Stat. Phys. **104**, 1191 (2001).

[35] M. A. Gallivan, D. R. Noble, J. G. Georgiadis, and R. O. Buckius, Int. J. Numer. Methods Fluids **25**, 249 (1997).

[36] D. R. Noble, S. Chen, J. G. Georgiadis, and R. O. Buckius, Phys. Fluids **7**, 203 (1995).

[37] Q. Zou and X. Y. He, Phys. Fluids **9**, 1591 (1997).

[38] A. J. C. Ladd, J. Fluid Mech. **271**, 285 (1994).

[39] M. M. Dupin, I. Halliday, C. M. Care, and G. B. Jeffery, Phys. Rev. E **73**, 055701(R) (2006).

[40] C. Pozrikidis, Ann. Biomed. Eng. **31**, 1194 (2003).

[41] E. A. Evans and R. Shalak, *Mechanics and Thermodynamics of Biomembranes* (CRC, Bota Raton, FL, 1980).

[42] N. Mohandas and E. A. Evans, Annu. Rev. Biophys. Biomol. Struct. **23**, 787 (1994).

[43] A. Ghosh, S. Sinha, J. A. Dharmadhikari, S. Roy, A. K. Dharmidhikari, J. Samuel, S. Sharma, and D. Mathur, Phys. Biol. **3**, 67 (2006).

[44] C. D. Eggleton and A. S. Popel, Phys. Fluids **10**, 1843 (1998).

[45] C. D. Eggleton and A. S. Popel, Phys. Fluids **10**, 1834 (1998).

[46] E. A. Evans and Y. C. Fung, Microvasc. Res. **4**, 335 (1972).

[47] D. M. Kroll and G. Gompper, Science **255**, 968 (1992).

[48] S. Jadhav, C. D. Eggleton, and K. Konstantopoulos, Biophys. J. **88**, 96 (2005).

[49] L. D. Landau and E. M. Lifshitz, *Theory of Elasticity* (Pergamon Press, New York, 1959).

[50] S. Hénon, G. Lenormand, A. Richert, and F. Gallet, Biophys. J. **76**, 1145 (1999).

[51] R. S. Maier, D. M. Kroll, Y. E. Kutsovsky, and R. S. Bernard, Phys. Fluids **10**, 60 (1998).

[52] J. P. Mills, L. Qie, M. Dao, C. T. Lim, and S. Suresh, Mech. Chem. Biosyst. **1**, 169 (2004).

[53] R. M. Hochmuth, P. R. Worthy, and E. A. Evans, Biophys. J. **26**, 101 (1979).

[54] S. C. R. Skalak, *Handbook of BioEngineering* (McGraw Hill, New York, 1987).

[55] C. K. Aidun, Y. Lu, and E. J. Ding, J. Fluid Mech. **373**, 287 (1998).

[56] G. B. Jeffery, Proc. R. Soc. London, Ser. A **102**, 161 (1922).

[57] H. L. Goldsmith and V. T. Turitto, Thromb. Haemostasis **55**, 415 (1986).

[58] G. J. Tangelder, D. W. Slaaf, T. Arts, and R. S. Reneman, Am. J. Physiol. Heart Circ. Physiol. **254**, H1059 (1988).

[59] N.-Q. Nguyen and A. J. C. Ladd, Phys. Rev. E **66**, 046708 (2002).