Computer Methods in Applied Mechanics and Engineering 426 (2024) 116979

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

# Eulerian formulation of the tensor-based morphology equations for strain-based blood damage modeling

Nico Dirkes [a,*], Fabian Key [b], Marek Behr [a]

[a] Chair for Computational Analysis of Technical Systems RWTH Aachen University, Schinkelstr. 2, 52062 Aachen, Germany
[b] Institute of Lightweight Design and Structural Biomechanics TU Wien, Gumpendorfer Str. 7, A-1060 Vienna, Austria

ARTICLE INFO

*Keywords:*
Computational hemodynamics
Finite element method
Ventricular assist device
Hemolysis
CFD
Red blood cell

ABSTRACT

The development of blood-handling medical devices, such as ventricular assist devices, requires the analysis of their biocompatibility. Among other aspects, this includes *hemolysis*, i.e., red blood cell damage. For this purpose, computational fluid dynamics (CFD) methods are employed to predict blood flow in prototypes. The most basic hemolysis models directly estimate red blood cell damage from fluid stress in the resulting flow field. More advanced models explicitly resolve cell deformation. On the downside, these models are typically written in a Lagrangian formulation, i.e., they require pathline tracking. We present a new Eulerian description of cell deformation, enabling the evaluation of the solution across the whole domain. The resulting hemolysis model can be applied to any converged CFD simulation due to one-way coupling with the fluid velocity field. We discuss the efficient numerical treatment of the model equations in a stabilized finite element context. We verify the model by comparison to the original Lagrangian formulation in selected benchmark flows. Two more complex test cases demonstrate the method's capabilities in real-world applications. The results highlight the advantages over previous hemolysis models. In conclusion, the model holds great potential for the design process of future generations of medical devices.

## 1. Introduction

As the prevalence of heart failure continues to rise worldwide [1], the development of blood handling medical devices, such as ventricular assist devices (VADs), has become paramount in improving patient outcomes and extending life expectancy. Nowadays, the development process is supported by CFD. It allows for a comprehensive understanding of how the design and operating parameters of a VAD influence the device's *hydraulic performance*. Additionally, prototypes need to be assessed for *biocompatibility*. Among other aspects, this involves the quantification of *hemolysis*, i.e., red blood cell (RBC) damage.

In the context of blood-handling medical devices, hemolysis occurs when fluid stresses induce excessive RBC deformation, damaging or even rupturing the cell membrane. Modeling hemolysis has been subject of intense research over the past 40 years [2–7]. Despite these efforts, there is still no universally accepted approach. In fact, a recent comparison has shown that computational predictions are still frequently inaccurate [8]. This uncertainty, in combination with new findings on the clinical significance of hemolysis [9], calls for the development of more reliable hemolysis models. Existing models can be categorized as *stress-based* models and *strain-based* models.

Stress-based hemolysis models apply an empirical correlation for hemoglobin release directly to the instantaneous fluid stress field. This approach has two downsides. First, such models assume implicitly that cells immediately deform to their steady state

* Corresponding author.
*E-mail address:* dirkes@aices.rwth-aachen.de (N. Dirkes).

https://doi.org/10.1016/j.cma.2024.116979
Received 17 January 2024; Received in revised form 29 March 2024; Accepted 30 March 2024
Available online 11 April 2024

when encountering stress. This is because empirical hemolysis correlations are generally measured at constant stress, i.e., the cell has time to adapt to this level of stress and reach a steady state. When encountering changing levels of stress, however, the cell membrane exhibits viscoelastic behavior [10]. This means that short, high levels of stress may not lead to significant hemolysis if the membrane does not have sufficient time to stretch in response. Stress-based models assume that the full instantaneous fluid stress applies immediately. Second, it is unclear how the correlation should be applied to arbitrary three-dimensional stress states. The empirical correlations [3,11] generally only relate shear stress to hemoglobin release. One generalization to arbitrary stress states was proposed by Bludszuweit [4]. She defined a scalar stress that results from a weighted sum of the stress components. However, the reduction to a representative scalar loses some information of the three-dimensional stress state, as a cell experiences different loads depending on its alignment with the principal axes of stress. In particular, Goldsmith and Marlow [12] showed that cells tend to align more with the flow direction at higher shear rates. More recently, it was suggested [13,14] that for certain flows, extensional stresses have a larger effect on cell deformation than shear stresses. Overall, the proper weighting of the stress components is still unclear and may depend on the flow regimes at hand. For these reasons, a first-principles model that explicitly describes cell deformation as a result of flow forces is preferable.

Strain-based hemolysis models incorporate such cell deformation models into hemolysis prediction. The empirical correlations for hemoglobin release are then applied to the causative membrane strain itself. There are different degrees of complexity for the cell deformation model. On the simpler side, Chen and Sharp [15] proposed a scalar model for estimating membrane strain in the context of cell rupture. However, this model was not originally intended for predicting sublethal hemolysis [16] and failed to achieve satisfactory results for this purpose [17]. Arwatz and Smits [18] described a scalar viscoelastic model for cell deformation. Its real-world applicability is limited, though, as it does not account for the complex three-dimensional flow forces and it was formulated only for constant shear flow. Thus, simple strain-based models tend to be valid only for specific applications.

On the more complex side, Ezzeldin et al. [19] and Sohrabi and Liu [20] developed models that resolve the cell membrane's three-dimensional deformation. Porcaro and Saeedipour [21] employed a reduced-order model for the cell structure and modeled the cells' interaction with the flow and with each other. Moreover, there exists a wide array of cell deformation models that were not specifically intended for hemolysis prediction but could be used in this context [22–27]. These complex approaches all have in common that their computational cost is too high to simulate large-scale VADs. Modern devices support flow rates of up to $10\,\frac{\mathrm{L}}{\mathrm{min}}$ [28], corresponding to almost 1 trillion RBCs per second. This means that only a small fraction of all cells are chosen at the inlet of the device and tracked until the outlet. Hemolysis is then averaged over these cells. This approach has two disadvantages. First, it is unclear how to pick the cells to achieve a representative average. Second, it gives a global index for hemolysis, but does not highlight critical regions inside the domain. In particular, it cannot be guaranteed that the selected cells will penetrate every part of the domain, e.g., boundary layers and recirculation areas. In conclusion, such complex cell deformation models are not well suited to aid the design process of VADs.

The aim of the present work is to develop a new model that combines the advantages of the different classes of models described above: the efficiency of stress-based and simple strain-based models, the characteristic cell response to three-dimensional stress of more complex strain-based models, and an Eulerian description, enabling evaluation across the entire domain.

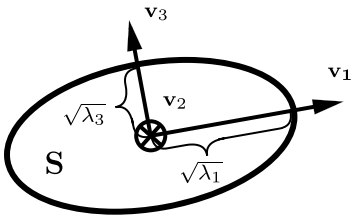
For this purpose, the strain-based model by Arora et al. [5] serves as basis, as it provides a compromise between these degrees of complexity, approximating RBCs as three-dimensional ellipsoids. However, it is based on a Lagrangian description of particle motion. In consequence, its evaluation requires particle tracking and comes with the same drawbacks as the more complex cell deformation models described above. Pauli et al. [6] introduced an Eulerian formulation. In the process, they neglected part of the model to achieve this formulation. As we will show in the present work, this may generally not be an admissible simplification in many situations. Instead, we will derive a full-order Eulerian formulation that is analytically equivalent to the original model. In addition, we will present a modification to improve robustness and efficiency, thus making the model suitable for simulations of realistic VAD configurations.

The paper is structured as follows: In Section 2, we define the constitutive equations for our strain-based hemolysis model. We introduce the original Lagrangian model formulation and derive the new full-order Eulerian formulation, contrasting it with the previous Eulerian formulation. In addition, we present a novel model with improved efficiency and robustness. In Section 3, we discuss how we treat the model equations numerically. In Section 4, we show results for verification and benchmarking. Finally, we summarize our work, discuss the model's limitations and give an outlook on future research in Section 5.

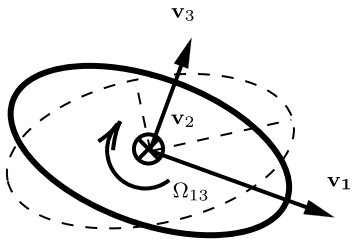
## 2. Model equations

At rest, red blood cells are known to aggregate into stacks, so-called rouleaux. Under shear, these structures break up and the cells move through the flow in a tumbling motion. At even higher shear rates, cells stop tumbling and start to assume a fixed orientation. In this state, their shape resembles elongated ellipsoids, with their major axis aligned with the flow direction [12,29]. This motion has been termed tank-treading, as the cell membrane rotates around the cell contents like the treads of a tank [30]. This regime of high shear rates is of primary interest for hemolysis modeling.

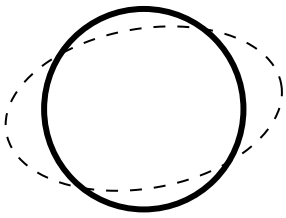
In this section, we present different models to describe RBCs in this regime. In Section 2.1, we introduce the original Lagrangian model formulation that constitutes the basis for all further discussion. In Section 2.2, we derive an equivalent Eulerian formulation. In Section 2.3, we present a new model that effectively replicates the original model behavior with better numerical performance. Finally, we contrast this with an older Eulerian model formulation in Section 2.4. In strain-based hemolysis modeling, any of these RBC models may be used to compute a scalar parameter that serves as input to empirical hemolysis correlations. This process is described in Section 2.5.



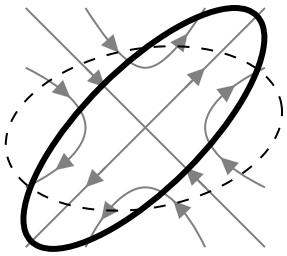

(a) Morphology $\mathbf{S}$ with eigenvalues $\lambda_i$ and eigenvectors $\mathbf{v}_i$



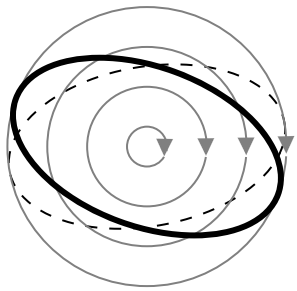

(b) Eigenvector rotation $\mathbf{\Omega}$



(c) Recovery source term



(d) Strain source term



(e) Vorticity source term

**Fig. 1.** Ellipsoidal cell representation in Arora morphology model.

### 2.1. Lagrangian model formulation

Based on the fluid droplet model by Maffettone and Minale [31], Arora et al. [5] derived a Lagrangian cell deformation model. Thereby, RBCs are assumed as neutrally bouyant ellipsoidal droplets carried with the flow. The limitations associated with this assumption are discussed in Section 5. Ellipsoids can be described mathematically by a symmetric positive definite morphology tensor $\mathbf{S}$, whose eigenvalues $\lambda_i$ represent the squared lengths of the ellipsoid's semi-axes. In this work, they are assumed to be in descending order, i.e., $\lambda_1 \geq \lambda_2 \geq \lambda_3$. Due to volume conservation, their product has to remain constant. The eigenvectors $\mathbf{v}_i$ represent the orientation of the ellipsoid's semi-axes. They can be understood as a rotation matrix $\mathbf{Q} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]$ that rotates the global inert coordinate system to the local rotating coordinate system of the cell. As derived in Appendix A, the rotation rate of the local frame with respect to the inertial frame is quantified by the rotation tensor

$$\mathbf{\Omega} = \frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t}\mathbf{Q}^{\mathbf{T}}. \tag{1}$$

The cells in the Arora model [5] do not interact with one another and do not influence the flow field. The evolution equation for the morphology of a single cell may be written along its pathline as follows:

$$\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t} - [\mathbf{\Omega}, \mathbf{S}] = -f_1\left(\mathbf{S} - g\left(\mathbf{S}\right)\mathbf{I}\right) + f_2\left(\mathbf{E}, \mathbf{S}\right) + f_3\left[\mathbf{W} - \mathbf{\Omega}, \mathbf{S}\right], \tag{2}$$

with tensor operations $[\mathbf{\Omega}, \mathbf{S}] := \mathbf{\Omega S} - \mathbf{S\Omega}$ and $(\mathbf{E}, \mathbf{S}) := \mathbf{ES} + \mathbf{SE}$, and coefficients

$$f_1 = 5.0\,\mathrm{s}^{-1}, \qquad f_2 = f_3 = 4.2298 \cdot 10^{-4}. \tag{3}$$

For details on the derivation of these coefficients, see [5]. The first term on the right-hand side describes shape recovery with the scalar function

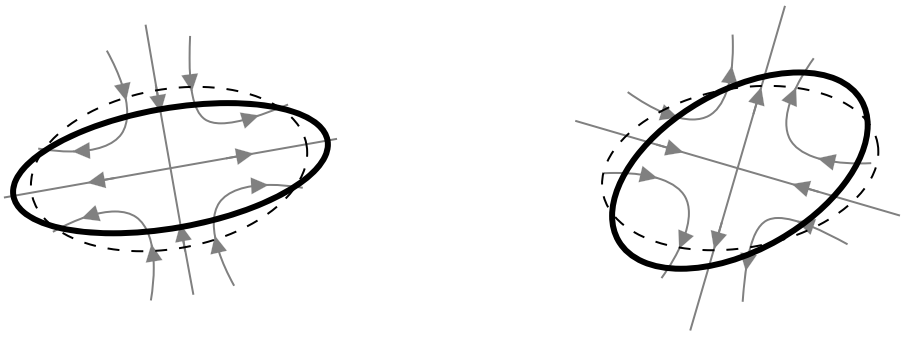
$$g(\mathbf{S}) = \frac{6\,\mathrm{det}\mathbf{S}}{\mathrm{tr}(\mathbf{S})^2 - \mathrm{tr}\left(\mathbf{S}^2\right)}.$$

The second term on the right-hand side describes the effect of fluid strain

$$\mathbf{E} = \frac{1}{2}\left(\left(\mathbf{\nabla u}\right) + \left(\mathbf{\nabla u}\right)^{\mathbf{T}}\right),$$

which causes the cell to align with the principal axes of strain and to deform along those axes. The third term on the right-hand side describes the effect of fluid vorticity

$$\mathbf{W} = \frac{1}{2}\left(\left(\mathbf{\nabla u}\right) - \left(\mathbf{\nabla u}\right)^{\mathbf{T}}\right),$$



(a) Deformational strain $\hat{\mathbf{E}}$ (b) Rotational strain $\mathbf{E} - \hat{\mathbf{E}}$

**Fig. 2.** Strain terms in full-order Eulerian formulation.

which rotates the cell. These three effects are combined as linear superposition. The left-hand side of Eq. (2) represents a Jaumann derivative that defines a co-rotating reference frame. Hence, the model employs a Lagrangian description of cell deformation, so its evaluation requires pathline tracking. The components of the model are visualized in Fig. 1.

### 2.2. Full-order Eulerian morphology model

A direct reformulation of Eq. (2) towards an Eulerian description is challenging due to the implicit dependency between the morphology tensor $\mathbf{S}$ and the rotation rate of its eigenvectors $\boldsymbol{\Omega}$. In order to resolve this dependency, we derive an expression for $\boldsymbol{\Omega}$ that does not involve derivatives of $\mathbf{Q}$. For this purpose, the spectral decomposition of the morphology tensor

$$\mathbf{S} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathbf{T}}, \qquad \boldsymbol{\Lambda} = \mathbf{diag}(\lambda_1, \lambda_2, \lambda_3), \qquad \mathbf{Q} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3], \tag{4}$$

is employed. In the following, the model equation (2) is transformed to the eigenbasis of the morphology tensor by multiplying with $\mathbf{Q}^{\mathbf{T}}$ from the left and with $\mathbf{Q}$ from the right. The transformed quantities are then defined as

$$\tilde{\mathbf{E}} = \mathbf{Q}^{\mathbf{T}}\mathbf{E}\mathbf{Q}, \qquad \tilde{\mathbf{W}} = \mathbf{Q}^{\mathbf{T}}\mathbf{W}\mathbf{Q}, \qquad \tilde{\boldsymbol{\Omega}} = \mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}\mathbf{Q}. \tag{5}$$

First, the derivative of the shape tensor $\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t}$ becomes

$$\begin{aligned}\mathbf{Q}^{\mathbf{T}}\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t}\mathbf{Q} = \mathbf{Q}^{\mathbf{T}}\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathbf{T}}\right)\mathbf{Q} = \mathbf{Q}^{\mathbf{T}}\left(\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t}\boldsymbol{\Lambda}\mathbf{Q}^{\mathbf{T}} + \mathbf{Q}\frac{\mathrm{d}\boldsymbol{\Lambda}}{\mathrm{d}t}\mathbf{Q}^{\mathbf{T}} + \mathbf{Q}\boldsymbol{\Lambda}\frac{\mathrm{d}\mathbf{Q}^{\mathbf{T}}}{\mathrm{d}t}\right)\mathbf{Q} \overset{(1)}{=} \mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}\mathbf{Q}\boldsymbol{\Lambda} + \frac{\mathrm{d}\boldsymbol{\Lambda}}{\mathrm{d}t} - \boldsymbol{\Lambda}\mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}\mathbf{Q} \\ = \frac{\mathrm{d}\boldsymbol{\Lambda}}{\mathrm{d}t} + \left[\tilde{\boldsymbol{\Omega}}, \boldsymbol{\Lambda}\right].\end{aligned}$$

Second, the rotation term $[\boldsymbol{\Omega}, \mathbf{S}]$ becomes:

$$\begin{aligned}\mathbf{Q}^{\mathbf{T}}\left[\boldsymbol{\Omega}, \mathbf{S}\right]\mathbf{Q} = \mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}\mathbf{S}\mathbf{Q} - \mathbf{Q}^{\mathbf{T}}\mathbf{S}\boldsymbol{\Omega}\mathbf{Q} = \mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}\mathbf{Q}\mathbf{Q}^{\mathbf{T}}\mathbf{S}\mathbf{Q} - \mathbf{Q}^{\mathbf{T}}\mathbf{S}\mathbf{Q}\mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}\mathbf{Q} = \tilde{\boldsymbol{\Omega}}\boldsymbol{\Lambda} - \boldsymbol{\Lambda}\tilde{\boldsymbol{\Omega}} \\ = \left[\tilde{\boldsymbol{\Omega}}, \boldsymbol{\Lambda}\right].\end{aligned}$$

Third, the same transformation is applied to the right-hand side of Eq. (2) in analogous fashion to obtain the full transformed model:

$$\frac{\mathrm{d}\boldsymbol{\Lambda}}{\mathrm{d}t} = -f_1\left(\boldsymbol{\Lambda} - g\left(\boldsymbol{\Lambda}\right)\mathbf{I}\right) + f_2\left(\tilde{\mathbf{E}}, \boldsymbol{\Lambda}\right) + f_3\left[\tilde{\mathbf{W}} - \tilde{\boldsymbol{\Omega}}, \boldsymbol{\Lambda}\right]. \tag{6}$$

Because $\tilde{\boldsymbol{\Omega}}$ and $\tilde{\mathbf{W}}$ are antisymmetric and $\boldsymbol{\Lambda}$ is diagonal, the last term on the right-hand side contains no elements on the diagonal. This lets us rewrite the equation only for the off-diagonal elements as follows:

$$\left[\tilde{\boldsymbol{\Omega}}, \boldsymbol{\Lambda}\right] = \frac{f_2}{f_3}\left(\tilde{\mathbf{E}}, \boldsymbol{\Lambda}\right) - \frac{f_2}{f_3}\mathbf{diag}\left(\tilde{\mathbf{E}}, \boldsymbol{\Lambda}\right) + \left[\tilde{\mathbf{W}}, \boldsymbol{\Lambda}\right].$$

Here, the **diag** operator leaves only the diagonal entries and sets all off-diagonal entries to zero. Applying the inverse coordinate transformation yields:

$$\left[\boldsymbol{\Omega}, \mathbf{S}\right] = \frac{f_2}{f_3}\left(\mathbf{E} - \hat{\mathbf{E}}, \mathbf{S}\right) + \left[\mathbf{W}, \mathbf{S}\right], \quad \hat{\mathbf{E}} = \mathbf{Q}\,\mathbf{diag}\left(\mathbf{Q}^{\mathbf{T}}\mathbf{E}\mathbf{Q}\right)\mathbf{Q}^{\mathbf{T}}. \tag{7}$$

Finally, we obtain the full Eulerian morphology model by treating the morphology tensor $\mathbf{S}$ as a field variable, hence understanding the Lagrangian derivative as a material derivative. The expression (7) then allows us to rewrite the original model (2) as follows:

$$\frac{\partial\mathbf{S}}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{S} = -f_1\left(\mathbf{S} - g\left(\mathbf{S}\right)\mathbf{I}\right) + f_2\left(\hat{\mathbf{E}}, \mathbf{S}\right) + \frac{f_2}{f_3}\left(\mathbf{E} - \hat{\mathbf{E}}, \mathbf{S}\right) + \left[\mathbf{W}, \mathbf{S}\right]. \tag{8}$$

This constitutes an analytically equivalent Eulerian formulation of the original model. Intuitively, $\hat{\mathbf{E}}$ represents the component of $\mathbf{E}$ that solely acts to deform the cell, computed as a projection on the morphology eigenvectors $\mathbf{Q}$. Consequently, $\mathbf{E} - \hat{\mathbf{E}}$ represents the component of $\mathbf{E}$ that solely acts to rotate the cell towards the principal axes of strain. The two different strain components are visualized in Fig. 2.

We remark that there is still an implicit relationship between the morphology tensor $\mathbf{S}$ and the deformational strain $\hat{\mathbf{E}}$, as the latter is a function of the morphology eigenvectors $\mathbf{Q}$. In contrast to the original formulation, however, this formulation does not involve the derivatives of these eigenvectors. The dependency is purely algebraic and can be resolved in the framework of the numerical method, e.g., by means of Newton iterations.

### 2.3. Tank-treading cell deformation model

An approach to resolve this dependency analytically is to rewrite the model explicitly in terms of the eigenvalues and eigenvectors of $\mathbf{S}$. For this purpose, we consider the diagonal terms and the off-diagonal terms of Eq. (6) separately and use the definitions (1) and (5):

$$\frac{\partial \lambda_i}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\lambda_i = -f_1\left(\lambda_i - g(\boldsymbol{\Lambda})\right) + 2f_2\lambda_i\tilde{E}_{ii}, \tag{9a}$$

$$\frac{\partial \mathbf{Q}}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{Q} = \mathbf{Q}\,\tilde{\boldsymbol{\Omega}}(\mathbf{Q}), \qquad \tilde{\Omega}_{ij}(\mathbf{Q}) = \frac{f_2}{f_3}\tilde{E}_{ij}\frac{\lambda_j+\lambda_i}{\lambda_j-\lambda_i} + \tilde{W}_{ij}. \tag{9b}$$

Equations (9a) and (9b) describe the deformation and rotation of cells, respectively. In particular, the deformation equation (9a) contains the effects of the recovery term (see Fig. 1(c)) and the deformational strain (see Fig. 2(a)). The terms are transformed to the eigensystem of the morphology tensor to act directly on the eigenvalues $\lambda_i$, which represent the ellipsoid's semi-axes. Similarly, the rotation equation (9b) describes the effects of the rotational strain (see Fig. 2(b)) and the vorticity (see Fig. 1(e)) on the eigenvectors $\mathbf{Q} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]$, which represent the ellipsoid's orientation.

The rotation equation (9b) exhibits a singularity at $\lambda_i = \lambda_j$. This corresponds to a circular cross-section of the ellipsoid, which makes the eigenvectors immediately align with the principal axes of strain in that plane. The eigenvectors may thus be computed as solution to $\tilde{E}_{ij} = 0$ in this degenerate state.

The formulation (9) is challenging to solve numerically due to the rotation equation (9b). On the one hand, the rotational source term (9b) is three orders of magnitude larger than the deformation source term (9a) and can become arbitrarily large due to the singularity at $\lambda_i = \lambda_j$ This creates a discrepancy of timescales and requires prohibitively small time steps. On the other hand, the orthogonal matrix $\mathbf{Q}$ is impractical to handle numerically as a field variable, as it contains 9 entries and its columns need to remain orthonormal. These issues lead to large computational effort and a high number of numerical constraints, making direct numerical simulation unfeasible for realistic geometries.

To alleviate these issues, we derive a model for the behavior of the rotation equation (9b) on the slower timescale of the deformation problem. For this purpose, we assume that rotation happens infinitely fast compared to deformation and distinguish between two states: *tank-treading* and *tumbling*.

A *tank-treading* cell is oriented such that the moments of strain and vorticity are in equilibrium, i.e., $\tilde{\boldsymbol{\Omega}} = \mathbf{0}$ in Eq. (9b). The orientation that satisfies this condition is termed $\mathbf{Q}_\star$ and depends on the cell deformation $\boldsymbol{\Lambda}$ and the flow quantities $\mathbf{E}$ and $\mathbf{W}$. It varies on the same scale as the flow quantities.

A *tumbling* cell is too deformed to assume an equilibrium orientation for the local flow state. In consequence, it will keep rotating and experience tensile and compressional stresses to approximately equal amounts. The source term $\tilde{\mathbf{E}}$ in the deformation equation (9a) thus tends to zero in the long timescale. This can be ensured by setting $\mathbf{Q} = \mathbf{0}$.

In sum, the tank-treading morphology model (TTM) becomes:

$$\frac{\partial \lambda_i}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\lambda_i = -f_1\left(\lambda_i - g(\boldsymbol{\Lambda})\right) + 2f_2\lambda_i\tilde{E}_{ii}, \qquad \tilde{\mathbf{E}} = \mathbf{Q}^{\mathbf{T}}\mathbf{E}\mathbf{Q}, \tag{10a}$$

$$\mathbf{Q} = \begin{cases} \mathbf{Q}_\star(\boldsymbol{\Lambda}, \mathbf{E}, \mathbf{W}), & \text{tank-treading}, \\ \mathbf{0}, & \text{tumbling}. \end{cases} \tag{10b}$$

This formulation reduces the rotation equation (9b) to an algebraic equation (10b), avoiding the associated time step limitations. The approach to solve the algebraic equation is detailed in Section 2.3.2. Moreover, the deformation equations can be simplified as well; due to volume conservation, the product of the eigenvalues has to remain constant and can be set to unity without loss of generality. Thus, we only need to solve Eq. (10a) for $\lambda_1$ and $\lambda_3$ and can compute $\lambda_2 = \frac{1}{\lambda_1\lambda_3}$. This reduces the number of differential variables from six in the full-order model (8) to just two in the TTM (10). As demonstrated in Section 4.4, this results in significant improvements in efficiency and robustness.

#### 2.3.1. Logarithmic formulation of tank-treading morphology model

For additional robustness, we employ a logarithmic transformation as presented by Haßler et al. [32] to ensure positive eigenvalues in the correct order. We define the following transformation:

$$\lambda_1 = 1 + e^{\hat{\lambda}_1} \in (1, \infty), \qquad \lambda_3 = \frac{1}{1 + e^{\hat{\lambda}_3}} \in (0, 1). \tag{11}$$

We rewrite the material derivative of the deformation model (10a) in terms of the transformed eigenvalues:

$$\frac{\partial \hat{\lambda}_1}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\hat{\lambda}_1 = \frac{\mathcal{F}_1(\boldsymbol{\Lambda})}{\lambda_1 - 1}, \qquad \frac{\partial \hat{\lambda}_3}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\hat{\lambda}_3 = \frac{\mathcal{F}_3(\boldsymbol{\Lambda})}{\lambda_3^2 - \lambda_3}, \qquad \mathcal{F}_i(\boldsymbol{\Lambda}) = -f_1\left(\lambda_i - g(\boldsymbol{\Lambda})\right) + 2f_2\lambda_i\tilde{E}_{ii}. \tag{12}$$

This is the logarithmic formulation of the tank-treading morphology model (TTLM). The model is thus solved for the transformed eigenvalues $\hat{\lambda}_1$ and $\hat{\lambda}_3$. The original eigenvalues $\lambda_1$ and $\lambda_3$ are obtained from the transformation (11). The right-hand side source terms are computed using these original eigenvalues.

### 2.3.2. Determining the equilibrium orientation of the cell

The algebraic equation (10b) requires us to solve the equation $\tilde{\boldsymbol{\Omega}}(\mathbf{Q}_\star) = \mathbf{0}$. We will derive the solution analytically in 2D first and then generalize it numerically to 3D. In 2D, we assume without loss of generality that any rotation happens in the plane that contains the ellipsoid's first two axes $\mathbf{v}_1, \mathbf{v}_2$ and that $\lambda_1 \neq \lambda_2$ (otherwise the cross-section is circular and the orientation is arbitrary). Then we only need to consider $\mathbf{E}, \mathbf{W}, \boldsymbol{\Lambda}, \mathbf{Q}, \boldsymbol{\Omega} \in \mathbb{R}^{2\times 2}$. In particular, the two-dimensional orientation $\mathbf{Q} = [\mathbf{v}_1, \mathbf{v}_2]$ can be expressed as a rotation $\mathbf{R}$ of the reference orientation $\mathbf{I}$ by an angle $\theta$:

$$\mathbf{Q} = \mathbf{R}(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}. \tag{13}$$

This allows us to write the rotation rate (9b) as a function of this scalar $\theta$:

$$\tilde{\Omega}_{21}(\theta; \boldsymbol{\Lambda}, \mathbf{E}, \mathbf{W}) = \frac{f_2}{f_3}\frac{\lambda_1 + \lambda_2}{\lambda_1 - \lambda_2}\left(E_{12}\cos(2\theta) + \frac{1}{2}(E_{22} - E_{11})\sin(2\theta)\right) - W_{12}. \tag{14}$$

For ease of notation, we define the coefficients

$$a(\boldsymbol{\Lambda}, \mathbf{E}) := \frac{f_2}{f_3}\frac{\lambda_1 + \lambda_2}{\lambda_1 - \lambda_2}E_{12}, \qquad b(\boldsymbol{\Lambda}, \mathbf{E}) := \frac{f_2}{2f_3}\frac{\lambda_1 + \lambda_2}{\lambda_1 - \lambda_2}(E_{22} - E_{11}), \qquad c(\mathbf{W}) := W_{12}. \tag{15}$$

The equilibrium orientation angle $\theta_\star$ is a solution to the equation $\tilde{\Omega}_{21}(\theta_\star) = 0$. Mathematical analysis of Eq. (14) reveals that such a solution exists only if

$$a^2 + b^2 \geq c^2. \tag{16}$$

This condition weighs the aligning effects of strain against the rotational effects of vorticity. If the vorticity is too strong, the strain cannot keep the cell at equilibrium and the cell tumbles instead. If an equilibrium orientation exists, there are always two solutions. We demand that the equilibrium is stable, i.e., the cell returns to equilibrium after small perturbations. This corresponds to the condition $\frac{\partial \tilde{\Omega}_{21}}{\partial \theta} < 0$. Analysis of Eq. (14) shows that the only solution that satisfies this condition is

$$\theta_\star(a, b, c) = \arctan\left(\frac{a + \sqrt{a^2 + b^2 - c^2}}{b + c}\right). \tag{17}$$

This corresponds to the solution where the longest semi-axis of the cell experiences tension and the shortest semi-axis experiences compression. The equilibrium orientation may then be computed from Eq. (13) as $\mathbf{Q}_\star = \mathbf{R}(\theta_\star)$. This is illustrated in Fig. 3(a).

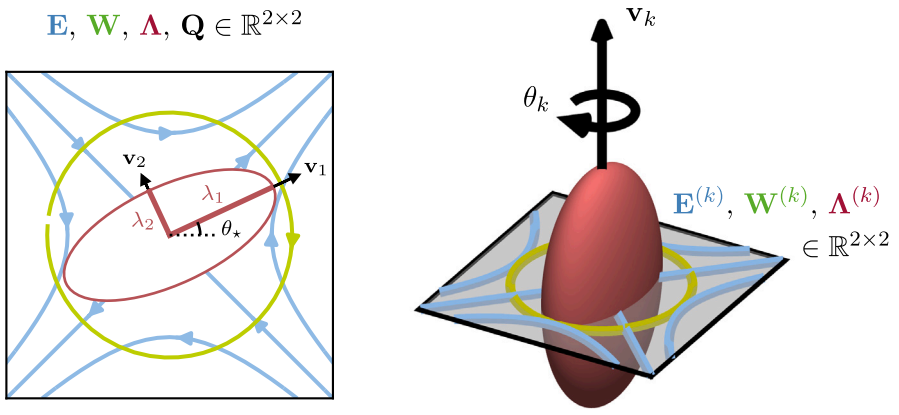
Next, we will present an approach to apply this analytical two-dimensional solution in a three-dimensional setting, i.e., $\mathbf{E}, \mathbf{W}, \boldsymbol{\Lambda}, \mathbf{Q}, \boldsymbol{\Omega} \in \mathbb{R}^{3\times 3}$. For this purpose, we define the *projection* operator

$$\boldsymbol{\mathcal{P}}_k : \mathbb{R}^{3\times 3} \to \mathbb{R}^{2\times 2}, \qquad \boldsymbol{\mathcal{P}}_k(\mathbf{A}) = (A_{ij})_{i\neq k, j\neq k}.$$

It projects a three-dimensional state to the plane normal to the $k$th coordinate axis by deleting the $k$th row and the $k$th column of the tensor $\mathbf{A}$. For each axis $k$, we obtain two-dimensional flow quantities $\mathbf{E}^{(k)} = \boldsymbol{\mathcal{P}}_k(\mathbf{E})$, $\mathbf{W}^{(k)} = \boldsymbol{\mathcal{P}}_k(\mathbf{W})$ and a two-dimensional cell deformation $\boldsymbol{\Lambda}^{(k)} = \boldsymbol{\mathcal{P}}_k(\boldsymbol{\Lambda})$ for the respective cross-section of the ellipsoid. We can then treat each cross-section according to the two-dimensional findings from above. This allows us to determine the equilibrium angle $\theta_k$ in that cross-section $k$. This is illustrated in Fig. 3(b).

All required steps are listed in Algorithm 1. We iteratively approximate the equilibrium orientation $\mathbf{Q}_\star$ by successively applying rotations $\theta_k$ around the ellipsoid's axes $\mathbf{v}_k$. The current approximation is stored in $\mathbf{Q} = [\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]$. Each iteration works as follows: First, we use the current orientation $\mathbf{Q}$ to transform the fluid strain and vorticity tensors to the coordinate system of the ellipsoid according to Eq. (5) (see line 5). The transformation ensures that the subsequent projection (see line 6) gives us information in a plane that contains two of the ellipsoid's axes $\mathbf{v}_i$, $i \neq k$. This is important because the deformation $\boldsymbol{\Lambda}^{(k)} = \boldsymbol{\mathcal{P}}_k(\boldsymbol{\Lambda})$ is given in this plane. Next, we treat this cross-section according to the two-dimensional results; we compute the coefficients $a_k$, $b_k$, $c_k$ using Eq. (15). Based on the condition (16), we determine if an equilibrium orientation exists. If it does, we compute the equilibrium angle $\theta_k$ according to Eq. (17) (see line 9). Analogous to Eq. (13), we define an elementary rotation matrix $\mathbf{R}_k(\theta) \in \mathbb{R}^{3x3}$ to perform a rotation around the $k$th axis. We apply the rotation to the current orientation $\mathbf{Q}$ by multiplying from the right (see line 10). This is because we want to rotate around the $k$th axis of the ellipsoid $\mathbf{v}_k$, rather than the $k$th axis of the reference coordinate system. If the equilibrium orientation does not exist, the cell is tumbling. According to Eq. (10b), we set $\mathbf{Q} = \mathbf{0}$ (see line 13).

In case of unstable convergence behavior, an underrelaxation factor may be added to the update of the orientation in line 10. In the underlying study, Algorithm1 converged for over 99.9% of quadrature points, mostly within fewer than 10 iterations of the outer loop. The algorithm converges towards a state where the longest axis of the cell experiences tension and the shortest axis experiences




(a) Equilibrium orientation angle in 2D (b) Projection to $k$-th normal plane in 3D

**Fig. 3.** Determining equilibrium orientation in 2D and 3D.

**Algorithm 1:** Equilibrium orientation algorithm in 3D

**Input:** A cell deformation $\mathbf{\Lambda}$, fluid strain $\mathbf{E}$ and fluid vorticity $\mathbf{W}$
**Output:** A cell orientation $\mathbf{Q}$ for the TTM within an accuracy of $\varepsilon$.

$\mathbf{Q} \leftarrow \mathbf{I}$ ;
**do**
  $converged \leftarrow true$ ;
  **for** $k \leftarrow 1$ **to** 3 **do**
    $\tilde{\mathbf{E}} \leftarrow \mathbf{Q}^{\mathrm{T}}\mathbf{E}\mathbf{Q}, \quad \tilde{\mathbf{W}} \leftarrow \mathbf{Q}^{\mathrm{T}}\mathbf{W}\mathbf{Q}$ ;
    $\mathbf{E}^{(k)} \leftarrow \boldsymbol{\mathcal{P}}_k(\tilde{\mathbf{E}}), \quad \mathbf{W}^{(k)} \leftarrow \boldsymbol{\mathcal{P}}_k(\tilde{\mathbf{W}}), \quad \mathbf{\Lambda}^{(k)} \leftarrow \boldsymbol{\mathcal{P}}_k(\mathbf{\Lambda})$ ;
    $a_k \leftarrow a(\mathbf{\Lambda}^{(k)}, \mathbf{E}^{(k)}), \; b_k \leftarrow b(\mathbf{\Lambda}^{(k)}, \mathbf{E}^{(k)}), \; c_k \leftarrow c(\mathbf{W}^{(k)})$ ;
    **if** $a_k^2 + b_k^2 \geq c_k^2$ **then** // cell is tank-treading
      $\theta_k \leftarrow \theta_\star(a_k, b_k, c_k)$ ;
      $\mathbf{Q} \leftarrow \mathbf{Q}\mathbf{R}_k(\theta_k)$ ;
      $converged \leftarrow (converged \wedge |\theta_k| \leq \varepsilon)$ ;
    **else** // cell is tumbling
      $\mathbf{Q} \leftarrow \mathbf{0}$ ;
      return ;
    **end**
  **end**
**while** *not converged*;

compression, i.e., $\tilde{E}_{11} \geq \tilde{E}_{22} \geq \tilde{E}_{33}$. This is in agreement with the two-dimensional results. Overall, we find that Algorithm 1 reliably and efficiently produces a unique solution.

An alternative is numerically integrating the transient equation $\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t} = \mathbf{Q}\tilde{\mathbf{\Omega}}$ up until steady state. This is computationally inefficient, as it requires on the order of 100,000 time steps and does not allow for the detection of tumbling. It is used only as a fallback solution in case Algorithm 1 does not converge.

### 2.4. Simplified Eulerian formulation

Previously, another formulation was obtained by Pauli et al. [6] by neglecting the eigenvector rotation $\mathbf{\Omega}$ in Eq. (2). This enables a more direct Eulerian description:

$$\frac{\partial \mathbf{S}}{\partial t} + (\mathbf{u} \cdot \boldsymbol{\nabla})\mathbf{S} = -f_1(\mathbf{S} - g(\mathbf{S})\mathbf{I}) + f_2(\mathbf{E}, \mathbf{S}) + f_3[\mathbf{W}, \mathbf{S}]. \tag{18}$$

Applying the same transformation as in Section 2.2 yields

$$[\mathbf{\Omega}, \mathbf{S}] = f_2\left(\mathbf{E} - \hat{\mathbf{E}}, \mathbf{S}\right) + f_3[\mathbf{W}, \mathbf{S}]. \tag{19}$$

The eigenvector rotation rate $\boldsymbol{\Omega}$ is three orders of magnitude smaller than that in the full-order model (7). The simplified model (18) is thus not equivalent to the original formulation. Furthermore, Eq. (19) shows that the simplified model still contains eigenvector rotation, despite initially neglecting the associated terms. These findings indicate that the simplified Eulerian formulation is inconsistent. The effects of this inconsistency will be discussed in Section 4.

### 2.5. Strain-based hemolysis modeling

Following Arora et al. [5], any of the cell deformation models from above may be employed to compute the cell distortion $D$ and the resulting effective shear rate $G_{\text{eff}}$:

$$D = \frac{\sqrt{\lambda_1} - \sqrt{\lambda_3}}{\sqrt{\lambda_1} + \sqrt{\lambda_3}}, \qquad G_{\text{eff}} = \frac{2Df_1}{(1-D^2)f_2}. \tag{20}$$

This effective shear rate may then serve as input to an empirical correlation relating shear stress to hemoglobin release [3,11]. The release rate can be interpreted as source term for an advection–diffusion equation that describes the free hemoglobin concentration in the blood. This has been described in detail in previous works [6,33]. The focus of the present study will be the cell morphology model itself, as it constitutes the defining feature of strain-based blood damage models.

It should be noted that despite its name, the effective shear rate $G_{\text{eff}}$ accounts for both shear and extensional fluid stress, as it is computed directly from the instantaneous cell distortion $D$. In fact, it provides a natural way to incorporate experimental observations on the significance of extensional stress [13,14]. In simple shear flow, RBCs assume a tank-treading orientation that is generally not aligned with the principal axis of strain [12]. Thus, the cells experience only a fraction of the total fluid stress. In extensional flow through a contraction, RBCs align with the principal axis of extension [14]. As a result, the cells experience the full stress. The predicted distortion $D$ and the resulting effective shear rate $G_{\text{eff}}$ are thus higher.

## 3. Numerical method

Our simulations consist of two components: the flow problem and the morphology problem. The flow problem is solved first and information about the flow field is then transferred to the morphology problem in a one-way coupling. In this section, we first present the mathematical definitions of these two problems in Sections 3.1 and 3.2. The computational frameworks that implement these problems are described in Section 3.3.

### 3.1. Flow problem

Whole blood is assumed to be an incompressible, Newtonian fluid, which is generally valid for blood at high shear rates [34]. High shear regions are the most significant in this context, since this is where the majority of hemolysis occurs. Fluid flow is thus governed by the incompressible Navier–Stokes equations:

$$\begin{aligned}
\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\mathbf{u}\right) &= -\boldsymbol{\nabla}\cdot\boldsymbol{\sigma} && \text{in } \mathfrak{D},\\
\boldsymbol{\nabla}\cdot\mathbf{u} &= 0 && \text{in } \mathfrak{D},\\
\boldsymbol{\sigma} &= -p\mathbf{I} + 2\mu\mathbf{E} && \text{in } \mathfrak{D},\\
\mathbf{u} &= \mathbf{u}_{\text{in}} && \text{on } \Gamma_{\text{in}},\\
\mathbf{u} &= \mathbf{u}_{\text{wall}} && \text{on } \Gamma_{\text{wall}},\\
\boldsymbol{\sigma}\cdot\mathbf{n} &= \mathbf{0} && \text{on } \Gamma_{\text{out}},
\end{aligned}$$

for the fluid domain $\mathfrak{D}$. The domain boundary $\partial\mathfrak{D}$ is partitioned into inflow $\Gamma_{\text{in}}$, no-slip walls $\Gamma_{\text{wall}}$ and stress-free outflow $\Gamma_{\text{out}}$. As material parameters for blood, we select a density of $\rho = 1.054\,\text{g}\,\text{cm}^{-3}$ and a viscosity of $3.5\,\text{cP}$. The problem is discretized with stabilized finite elements. For details on the stabilization terms, we refer to Pauli and Behr [35]. To simulate rotating systems at steady state, we employ the moving reference frame (MRF) approach [36]. For our applications involving an impeller, we define a rotating inner domain $\mathfrak{D}^{\text{rot}}$ and a fixed outer domain $\mathfrak{D} \setminus \mathfrak{D}^{\text{rot}}$. The inner domain rotates along with the impeller at an angular velocity $\boldsymbol{\Omega}^{\text{imp}}$.

### 3.2. Morphology problem

Fundamentally, the morphology models (8), (10) and (18) represent advection-reaction equations. Aggregating the respective differential degrees of freedom in a vector $\boldsymbol{\phi}$ , each of them can be written in the following form:

$$\mathcal{R}_{\text{M}}(\boldsymbol{\phi};\mathbf{u},\boldsymbol{\nabla}\mathbf{u}) := \frac{\partial\boldsymbol{\phi}}{\partial t} + (\mathbf{u}\cdot\boldsymbol{\nabla})\boldsymbol{\phi} - \mathbf{f}(\boldsymbol{\phi},\boldsymbol{\nabla}\mathbf{u}) = \mathbf{0} \qquad \text{in } \mathfrak{D}, \tag{21a}$$

$$\boldsymbol{\phi} = \boldsymbol{\phi}_{\text{in}} \qquad \text{on } \Gamma_{\text{in}}, \tag{21b}$$

$$\boldsymbol{\nabla}\boldsymbol{\phi}\cdot\mathbf{n} = \mathbf{0} \qquad \text{on } \Gamma_{\text{wall}} \cup \Gamma_{\text{out}}. \tag{21c}$$

At the inlet $\Gamma_{\text{in}}$, we impose some cell morphology, e.g., an undeformed cell. At walls and at the outflow, there is no component of advection velocity from the boundary into the computational domain. Consequently, we apply no-flux boundary conditions at all remaining boundaries $\Gamma_{\text{wall}} \cup \Gamma_{\text{out}} = \partial\mathfrak{D} \setminus \Gamma_{\text{in}}$.

Using the MRF approach, the governing equations (21a) are transformed to the same reference frames as the flow problem. In particular, RBCs are advected by the relative velocity field $\tilde{\mathbf{u}} = \mathbf{u} - \boldsymbol{\Omega}^{\text{imp}} \times \mathbf{r}$ in the rotating frame. The flow forces acting on the cells, i.e., the velocity gradients, are transformed accordingly:

$$\boldsymbol{\mathcal{R}}_{\mathsf{M}}(\boldsymbol{\phi}; \tilde{\mathbf{u}}, \boldsymbol{\nabla}\mathbf{u} - \boldsymbol{\Omega}^{\text{imp}}) = \mathbf{0} \qquad \text{in } \mathfrak{D}^{\text{rot}}.$$

This formulation holds true for model formulations that are *objective*, i.e., invariant under rotation. This is the case for models (8) and (10). The simplified model (18) is not objective, however, as cells do not rotate with the full fluid vorticity $\mathbf{W}$. This particular model thus requires the addition of another term that accounts for the rotating reference frame:

$$\frac{\partial \mathbf{S}}{\partial t} + (\tilde{\mathbf{u}} \cdot \boldsymbol{\nabla})\mathbf{S} = -f_1(\mathbf{S} - g(\mathbf{S})\mathbf{I}) + f_2(\mathbf{E}, \mathbf{S}) + f_3[\mathbf{W}, \mathbf{S}] - \left[\boldsymbol{\Omega}^{\text{imp}}, \mathbf{S}\right] \qquad \text{in } \mathfrak{D}^{\text{rot}}.$$

For the finite element discretization, we consider the stabilized weak Galerkin formulation of the problem: Find $\boldsymbol{\phi}^h \in \mathcal{S}^h \subset [H^1(\mathfrak{D})]^{n_{\text{dof}}}$ such that:

$$D_{\mathsf{M}}(\boldsymbol{\psi}^h, \boldsymbol{\phi}^h) + E_{\mathsf{M}}(\boldsymbol{\psi}^h, \boldsymbol{\phi}^h) + J_{\mathsf{M}}(\boldsymbol{\psi}^h, \boldsymbol{\phi}^h) = 0 \qquad \forall \boldsymbol{\psi}^h \in \mathcal{V}^h \subset [H_0^1(\mathfrak{D})]^{n_{\text{dof}}}, \tag{22}$$

where $H^1(\mathfrak{D})$ denotes the set of functions with square-integrable first derivatives on $\mathfrak{D}$ and $H_0^1(\mathfrak{D})$ is a subset of those functions satisfying a homogeneous Dirichlet boundary condition on $\Gamma_{\text{in}}$. The first term represents the constitutive model equations:

$$D_{\mathsf{M}}(\boldsymbol{\psi}^h, \boldsymbol{\phi}^h) = \int_{\mathfrak{D}\setminus\mathfrak{D}^{\text{rot}}} \boldsymbol{\psi}^h \cdot \boldsymbol{\mathcal{R}}_{\mathsf{M}}(\boldsymbol{\phi}^h, \mathbf{u}, \boldsymbol{\nabla}\mathbf{u})\,\mathrm{d}\mathbf{x} + \int_{\mathfrak{D}^{\text{rot}}} \boldsymbol{\psi}^h \cdot \boldsymbol{\mathcal{R}}_{\mathsf{M}}(\boldsymbol{\phi}^h, \tilde{\mathbf{u}}, \boldsymbol{\nabla}\mathbf{u} - \boldsymbol{\Omega}^{\text{imp}})\,\mathrm{d}\mathbf{x}.$$

The second term $E_{\mathsf{M}}$ represents stabilization. We are using Galerkin/least-squares (GLS) stabilization [37]. For details on the formulation for multi-dimensional variables, we refer to Pauli [34]. The third term represents $YZ\beta$ discontinuity capturing [38], computed element by element:

$$J_{\mathsf{M}}(\boldsymbol{\psi}^h, \boldsymbol{\phi}^h) = \sum_{e=1}^{n_{\text{el}}} \int_{\mathfrak{D}_e} \nu_{\text{DC}} \boldsymbol{\nabla}\boldsymbol{\psi}^h : \boldsymbol{\nabla}\boldsymbol{\phi}^h \,\mathrm{d}\mathbf{x}.$$

### 3.3. Implementation

The Eulerian problems presented in Sections 3.1 and 3.2 are solved using our in-house multiphysics finite element code `XNS`. Details on the implementation can be found in Appendix B.

In order to verify the Eulerian model formulations and their numerical discretization, we compare them to their Lagrangian counterparts. This is achieved in five steps, all of which are available as part of our open-source Python package HemTracer.[1] First, we use the same Eulerian flow field determined by the flow problem (see Section 3.1) to compute pathlines, i.e., trajectories of fluid particles. They are obtained by integrating the following ordinary differential equation (ODE) starting from a given initial position $\mathbf{X}_0$:

$$\frac{\mathrm{d}\mathbf{X}}{\mathrm{d}t} = \begin{cases} \mathbf{u}(\mathbf{X}, t), & \mathbf{X}(t) \in \mathfrak{D} \setminus \mathfrak{D}^{\text{rot}}, \\ \tilde{\mathbf{u}}(\mathbf{X}, t), & \mathbf{X}(t) \in \mathfrak{D}^{\text{rot}}, \end{cases} \qquad \mathbf{X}(0) = \mathbf{X}_0, \tag{23}$$

Second, we compute the velocity gradients of the flow field and interpolate them to the pathlines $\mathbf{X}(t)$ to obtain $\boldsymbol{\nabla}\mathbf{u}(t)$. Third, we rewrite the morphology problem in the Lagrangian frame by substituting the derivatives with respect to space and time with the material derivatives along the pathline, i.e., the morphology problem (21) becomes:

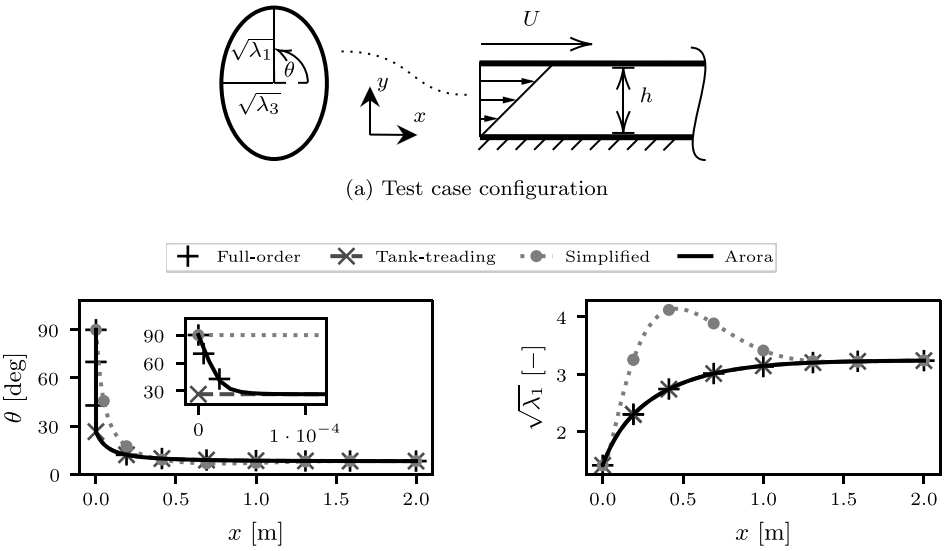
$$\frac{\mathrm{d}\boldsymbol{\phi}}{\mathrm{d}t} = \begin{cases} \mathbf{f}(\boldsymbol{\phi}, \boldsymbol{\nabla}\mathbf{u}(t)), & \mathbf{X}(t) \in \mathfrak{D} \setminus \mathfrak{D}^{\text{rot}}, \\ \mathbf{f}(\boldsymbol{\phi}, \boldsymbol{\nabla}\mathbf{u}(t) - \boldsymbol{\Omega}^{\text{imp}}), & \mathbf{X}(t) \in \mathfrak{D}^{\text{rot}}, \end{cases} \qquad \boldsymbol{\phi}(0) = \boldsymbol{\phi}_0.$$

We match the initial condition $\boldsymbol{\phi}_0$ to the Eulerian morphology field at the initial position $\mathbf{X}_0$. Fourth, we solve the Lagrangian problem numerically by integrating the above ODE in time. We use the `scipy.integrate.solve_ivp` routine with an adaptive Runge–Kutta 45 scheme. Fifth, we interpolate the Eulerian morphology results to the same pathline and compare them to the Lagrangian results.

## 4. Numerical results

In order to compare the models presented above, we apply them to a selection of benchmark problems. In Sections 4.1 and 4.2, we consider two simple flows to verify the new Eulerian model formulations (8) and (10) and show the deficiencies of the simplified model (18). We discuss two more realistic configurations in Sections 4.3 and 4.4 to show practical advantages of the tank-treading morphology model (TTM) (10) over the full-order model (8).

[1] https://github.com/nicodirkes/HemTracer




(a) Test case configuration

(b) Angle of major axis to flow direction.

(c) Major axis length.

**Fig. 4.** Comparison of morphology models under simple shear.

### 4.1. Simple shear

First, we consider planar two-dimensional Couette flow. This configuration is visualized in Fig. 4(a). The channel height is set to $h = 2.5 \cdot 10^{-5}\,\mathrm{m}$ and the top wall moves at $U = 1\,\mathrm{m\,s^{-1}}$. The channel has a streamwise length of 2 m. The computational domain is discretized with 10,000 rectangular finite elements. At the inlet, the elements have a streamwise length of $5 \cdot 10^{-6}\,\mathrm{m}$ to capture cell behavior in the entry region. The length progressively increases downstream. At the inlet, red blood cells are introduced with an initial shape of $(\lambda_1, \lambda_2, \lambda_3) = (2, 1, 0.5)$. The angle between the flow direction and the major axis is defined as $\theta$. At the inlet, the orientation is imposed such that the major axis is orthogonal to the flow direction, i.e., $\theta_{\mathrm{in}} = 90°$. The solutions for the Eulerian models (8), (10) and (18) are obtained by solving the steady state morphology problem (22).

From a Lagrangian point-of-view, planar Couette flow produces simple shear, i.e., unidirectional flow with a constant gradient perpendicular to the flow direction. Fluid strain and vorticity are of equal strength in simple shear. Their magnitude is defined by the fluid shear rate $G_{\mathrm{f}} = 40{,}000\,\mathrm{s^{-1}}$. If we align the $x$-axis with the flow direction and the $y$-axis with the gradient direction, the flow gradients can be written as follows:

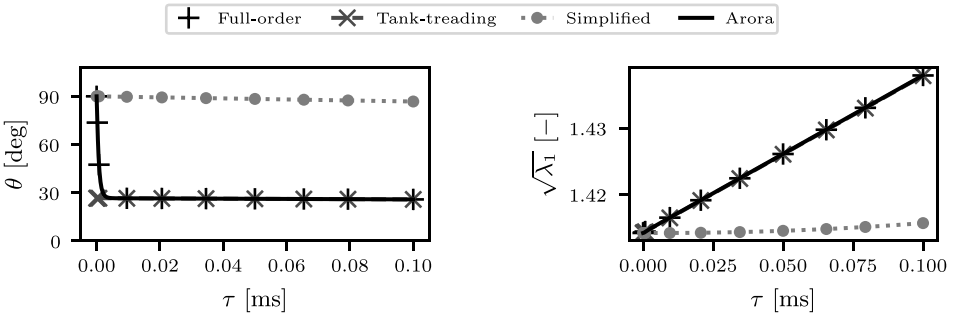
$$\nabla \mathbf{u} = \begin{bmatrix} 0 & G_{\mathrm{f}} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

For this flow field, pathlines are straight lines with $y = \mathrm{const}$. We evaluate the Lagrangian Arora model along these pathlines as described in Section 3.3. The initial condition is equivalent to the inlet boundary condition then. If we choose the pathline at the top, i.e., $y = h$ with $U = 1\,\mathrm{m\,s^{-1}}$, the time coordinate $t$ in the Lagrangian formulation corresponds one-to-one to the spatial coordinate $x$ in the Eulerian formulation. The solution of the Lagrangian model can thus be compared directly to the Eulerian solution.

Fig. 4(b) shows the evolution of the major axis angle $\theta$. Starting from the imposed initial angle of 90°, the cell rapidly assumes an orientation $\theta = 26.6°$. As visualized in the inset in Fig. 4(b), the initial alignment happens within the first 0.1 ms according to the Arora model (2). The resulting angle corresponds precisely to the equilibrium orientation for the initial deformation and the given flow state, i.e., $\theta_\star = 26.6°$ according to Eq. (17). Downstream, the cell remains at equilibrium orientation $\theta_\star$, which decreases asymptotically towards 0 with increasing cell deformation. This steady orientation represents a tank-treading state. The alignment of the major axis with the flow direction at high shear rates agrees with experimental observations [12,29].

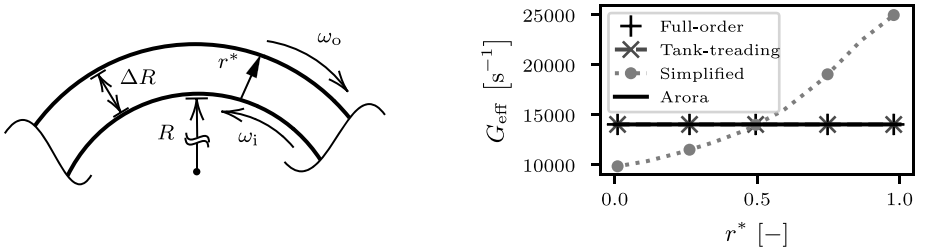
The Eulerian models differ in their ability to predict the initial rapid alignment. As shown in the inset, the full-order model (8) predicts the alignment perfectly. The TTM does not resolve the initial alignment, as the cell is assumed to be at equilibrium permanently. However, it predicts the orientation accurately after the initial alignment, i.e., $x > 1 \cdot 10^{-4}\,\mathrm{m}$. In contrast, the simplified model (18) predicts a slower alignment with the tank-treading orientation. This is due to the lower rotation rate of cells (cf. Section 2.4). The final orientation is predicted accurately by the simplified model. This is because it represents an equilibrium between the rotational effects of strain and vorticity. The strength of these effects relative to one another is the same in the simplified model as in the full-order model. The absolute rotation rate is lower, however, causing slower transient behavior.

Fig. 4(c) shows the evolution of the major axis length $\sqrt{\lambda_1}$. Starting from the initial deformation, the cell deforms towards a steady state deformation of $(\lambda_1, \lambda_2, \lambda_3) = (10.5, 0.43, 0.22)$. According to Eq. (20), this corresponds to an effective shear rate of




(a) Angle of major axis to flow direction. (b) Major axis length.

**Fig. 5.** Comparison of morphology models under short exposure times $\tau = x/U$.




(a) Test case configuration (b) Radial distribution of effective shear rate

**Fig. 6.** Comparison of morphology models under rotating shear.

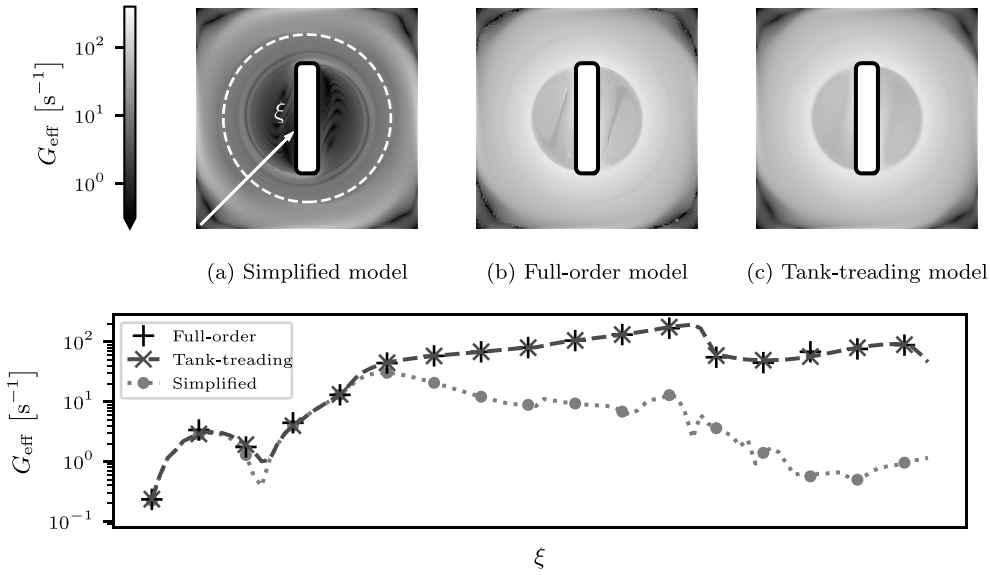
$G_{\text{eff}} = 40{,}000\,\text{s}^{-1} = G_{\text{f}}$. This is expected behavior, as the effective shear rate is constructed precisely to predict an equivalent simple shear flow that induces the instantaneous cell deformation at steady state. The Eulerian models agree in the prediction of this final deformation, but they differ in their prediction of transient behavior. The full-order model predicts the evolution of deformation in full agreement with the Arora model. The TTM provides an equivalent approximation. The discrepancy with respect to the initial cell orientation is limited to such small timescales that it does not affect deformation, which occurs over longer timescales. The simplified model, on the other hand, significantly overpredicts intermediate deformation. This is due to the slower alignment with the tank-treading state, which causes the cell to be more aligned with the direction of shear for longer. The steady deformation is predicted correctly due to the correct prediction of the final orientation.

To investigate the behavior under shorter exposure times, we modify the configuration in Fig. 4(a) by increasing the top wall velocity to $U = 200\,\text{m}\,\text{s}^{-1}$ and increasing the bottom wall velocity to $U_{\text{bot}} = 100\,\text{m}\,\text{s}^{-1}$. With a gap width of $h = 1 \cdot 10^{-4}\,\text{m}$, this results in a fluid shear rate of $G_{\text{f}} = 1{,}000{,}000\,\text{s}^{-1}$. We compare the models' predictions in Fig. 5 over the exposure time $\tau = x/U$ of cells moving along the top wall. Because of the short exposure time, the total deformation in Fig. 5(b) is lower than in the previous configuration. This is a feature of strain-based hemolysis models, which take into account the viscoelastic behavior of the cell membrane. Due to the higher levels of shear stress, the initial alignment of the cell with the tank-treading orientation in Fig. 5(a) happens faster than in the previous configuration. In particular, it still takes up only a miniscule fraction of the total exposure time. As a result, the TTM is again able to predict orientation and deformation accurately in the relevant timescales, as evidenced by Fig. 5(b). In contrast, the alignment with the tank-treading state is much slower in the simplified mode. As a result, this model fails to predict any significant deformation over this exposure time.

Overall, this test case verifies the full-order Eulerian reformulation (8), which agrees perfectly with the Lagrangian model (2). The TTM (10) only deviates slightly for the initial orientation, but agrees for the deformation. Since deformation is the primary quantity of interest, the TTM is a good approximation for the full-order model. The simplified model (18), on the other hand, deviates significantly for the transient cell deformation. This demonstrates the effects of the lower rotation rate.

### 4.2. Rotating shear

In order to explore transient cell behavior in more detail, we consider two-dimensional circular Couette flow. This test case is visualized in Fig. 6(a). We choose $R = 0.701\,\text{cm}$, $\Delta R = 0.0001\,\text{cm}$ and counter-rotating walls with $\omega_{\text{i}} = 1\,\text{rad}\,\text{s}^{-1}$, $\omega_{\text{o}} = -1\,\text{rad}\,\text{s}^{-1}$. This system produces a nearly constant shear rate of $G_{\text{f}} = 14{,}021\,\text{s}^{-1}$ across the domain. We define a non-dimensionalized radial coordinate $r^* = \frac{r-R}{\Delta R} \in [0, 1]$. The domain is discretized with 8 elements in radial direction and 3560 elements in azimuthal direction.




(a) Simplified model (b) Full-order model (c) Tank-treading model

(d) Comparison between morphology models along line from lower left corner to center (see Fig. 7a).

**Fig. 7.** Morphology results for square stirrer.

Due to the two-dimensional nature of the problem, the Taylor–Couette instability does not affect the flow. We simulate all Eulerian morphology models on this domain up to steady state.

The pathlines are closed circles with $r^* = \text{const}$. Along each pathline, a cell experiences simple shear flow with a rotating direction of shear:

$$\nabla \mathbf{u}(t) = \mathbf{R}_z(\omega t) \begin{bmatrix} 0 & G_{\text{f}} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{R}_z{}^{\mathbf{T}}(\omega t), \qquad \mathbf{R}_z(\varphi) = \begin{bmatrix} \cos\varphi & -\sin\varphi & 0 \\ \sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The angular frequency depends on the radius of the pathline:

$$\omega\left(r^*\right) = \omega_{\text{i}} + \left(\omega_{\text{o}} - \omega_{\text{i}}\right)\left(2r^* - 1\right).$$

We evaluate the Arora model for three radii $r^* \in \{0, 0.5, 1\}$ by integrating the Lagrangian morphology model along the circular pathlines up until steady state.

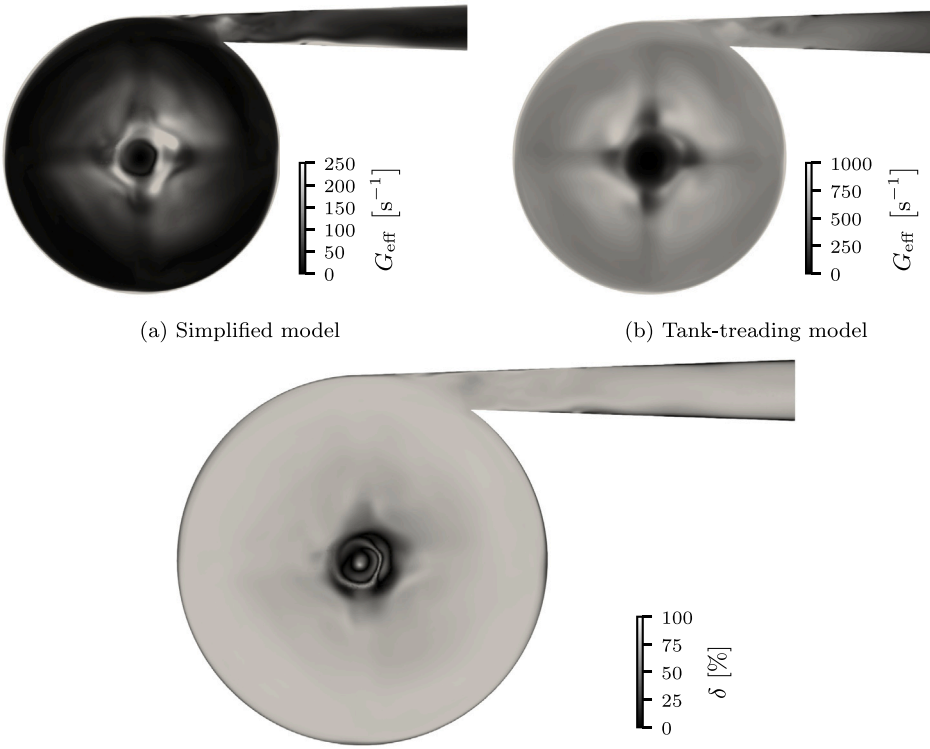
In Fig. 6(b), we compare the different morphology models. With the Lagrangian Arora model, the effective shear rate $G_{\text{eff}}$ is nearly constant across the range of $r^*$ and corresponds to fluid shear $G_{\text{f}}$. As cells travel along their pathlines, they constantly realign themselves with respect to the rotating direction of shear. In the Arora model, this realignment occurs so rapidly that cells practically experience only simple shear. For simple shear, $G_{\text{eff}} = G_{\text{f}}$ holds in steady state, as discussed in Section 4.1.

Out of the Eulerian models, the full-order model again agrees perfectly with the Lagrangian model. The TTM agrees very well with the full-order model for these parameters. The simplified model, on the other hand, deviates significantly from the full-order model. It underpredicts cell deformation at the inner wall and overpredicts it at the outer wall. It agrees with the full-order model only at the centerline $r^* = 0.5$. This is due to the lower rotation rate of cells in the simplified model, which causes a phase lag in their orientation as they travel along the circular pathline. At the inner wall, this phase lag causes the major axis to be less aligned with the principal axis of strain, leading to lower deformation. At the outer wall, the phase lag causes the major axis to be more aligned with the principal axis of strain, leading to higher deformation. At the centerline, cells are at rest, so they do not experience a rotating direction of shear, $\omega(r^* = 0.5) = 0$. This is equivalent to simple shear, where the simplified model agrees with the full-order model at steady state.

This test case serves as further verification for the Eulerian full-order model and the TTM, both of which agree with the Lagrangian Arora model. Furthermore, it demonstrates that the simplified model deviates from the full-order model not only in transient behavior (see Section 4.1), but also at steady state. In general, the simplified model is not suitable for any flows with non-constant gradients along its pathlines, as this requires cells to realign. This realignment incurs modeling errors due to the lower rotation rate in the simplified model. As this test case demonstrates, these errors can lead to both overprediction and underprediction of effective shear rate.

### *4.3. Square stirrer*

In order to test the morphology models' performance in more complex flows, we consider a two-dimensional square domain with a stirrer. The geometry and mesh are identical to those presented in previous works [32,34]. We set a stirrer frequency of




(a) Simplified model

(b) Tank-treading model

(c) Relative error of simplified model

**Fig. 8.** Effective shear rate in simplified three-dimensional blood pump in the plane 5 mm above the impeller blades.

**Table 1**
Execution times for the simple pump test case on 192 cores.

| Simulation | Time |
|---|---|
| CFD (MRF) | 20 h |
| Full-order morphology (est.) | 4000 h |
| Simplified morphology | 45 min |
| TTLM | 4 min |

$\omega_{\text{stir}} = 100\,\text{rad}\,\text{s}^{-1}$ and use the MRF approach for the rotating stirrer. The interface is visualized as a dashed line in Fig. 7(a). The computational domain is discretized with 92,262 triangular elements. The initial condition is an undeformed cell. We simulate the Eulerian morphology models up to steady state.

The model predictions are compared in Fig. 7. The solution in Fig. 7(a) matches previous results using the simplified model [32]. In comparison with the full-order model in Fig. 7(b), however, it deviates significantly. In particular, it underpredicts effective shear rate close to the stirrer by two orders of magnitude. The TTM in Fig. 7(c) agrees well with the full-order model, but lacks some of the finer details of the solution in the region close to the stirrer. Finally, the three models are compared along the profile line $\xi$ in Fig. 7(d). The results for the simplified model agree qualitatively with those presented by Haßler et al. [32]. However, this agrees with the full-order model only close to the outer walls. In that region, flow velocities are lower, so cells do not have to respond to changing velocity gradients as quickly. The simplified model is thus able to predict the effective shear rate accurately. Closer to the center, cells rotate more quickly, so the modeling error due to the lower rotation rate of the simplified model becomes more significant. This causes the underprediction in that region. Conversely, the TTM agrees well with the full-order model across the whole domain.

### *4.4. Simple blood pump*

The final test case involves a three-dimensional geometry. It was designed by Pauli et al. [6] to represent a simplified version of the FDA benchmark pump [8]. In particular, it is lacking the central cone element of the impeller. The domain is discretized with a total of 6,618,708 tetrahedral elements. We simulate fluid flow through the pump with the MRF approach at a flow rate of $0.5\,\text{L}\,\text{min}^{-1}$, with a rotation rate of 1000 rpm. The Reynolds numbers in the inflow tube and impeller region are $\text{Re}_{\text{in}} = \rho\bar{u}_{\text{in}}d_{\text{in}}/\mu = 320$

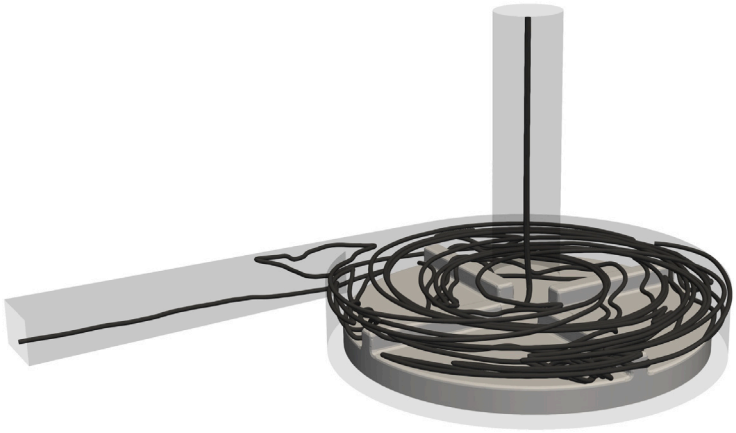


**Fig. 9.** Selected pathlines in the simple blood pump.

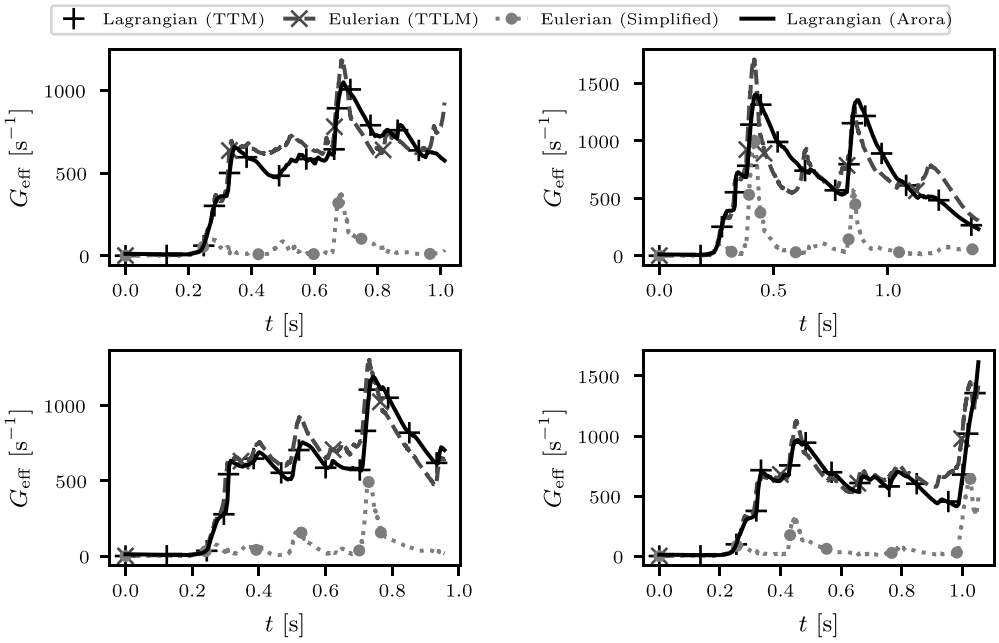



**Fig. 10.** Eulerian and Lagrangian model formulations compared along four sample pathlines in the simple blood pump.

and $\text{Re}_{\text{imp}} = \rho\omega_{\text{rot}} r_{\text{imp}}^2/\mu = 25{,}712$, respectively. Since this corresponds to a laminar flow regime [39], we do not employ a turbulence model. For the Eulerian morphology models, we set undeformed cells as initial condition and as boundary condition at the inlet. We employ the logarithmic formulation of the tank-treading morphology model (TTLM) for additional robustness. We perform unsteady simulations until we reach a quasi-steady state, i.e., the flow field and cell deformation do not change significantly over time.

The execution times to obtain the respective solutions are listed in Table 1. The CFD simulation requires 833 time steps of size $\Delta t = 5 \cdot 10^{-4}$ to produce a steady solution. Each time step takes 86 s on 192 cores. In this complex flow, cells exhibit unsteady tumbling behavior even in quasi-steady state. The two morphology models (8) and (18) require us to resolve this rotation. For the simplified morphology model, we require 60 time steps, which take 45 s each on 192 cores. As the rotation rate of cells in the full-order model is larger by a factor of $1/f_3$ (comp. Eqs. (7) and (19)), we would require at least 142,000 time steps for the full-order model. Each step takes more than 100 s due to the eigenvector operations, leading us to an estimated total execution time of at least 4000 h, two orders of magnitude larger than the CFD simulation. For this reason, we do not perform the Eulerian full-order morphology simulation for this test case. Finally, the TTLM does not need to resolve cell rotation and thus only requires 30 time steps. Additionally, each time step only takes 7 s due to the lower number of degrees of freedom.

Fig. 8 shows the effective shear rate on a plane parallel to the impeller disk. The plane is located 0.5 mm above the blades. The simplified model in Fig. 8(a) again underpredicts the effective shear rate severely compared to the TTLM in Fig. 8(b). The relative

error $\delta = |G_{\text{eff}}^{\text{simpl}} - G_{\text{eff}}^{\text{TTLM}}|/G_{\text{eff}}^{\text{TTLM}}$ is shown in Fig. 8(c). Close to the outer wall and close to the inflow in the center, the two models agree. This can be explained by the same reasoning as in the square stirrer test case (cf. Section 4.3): The simplified model is able to predict the effective shear rate accurately in regions with low flow velocity, where cells do not have to respond to changing velocity gradients as quickly. Everywhere else, the simplified model underpredicts the effective shear rate by up to 98%.

For verification, we compare the Eulerian and Lagrangian model formulations as outlined in Section 3.3. Four particular pathlines are selected and visualized in Fig. 9. The pathlines provide good coverage of the area around the impeller. However, only one of the four pathlines exits the domain through the outlet. The other pathlines terminate at walls or at the impeller. This is an artifact of the discrete nature of the velocity field and the explicit time stepping scheme used to integrate the pathlines. It presents a principal limitation of the Lagrangian approach, as it is not possible to track every cell through the whole domain, especially in complex geometries, where cells can get trapped in recirculation zones. This limitation does not apply to the Eulerian models, as they are solved on the whole domain.

The model results along these particular pathlines are given in Fig. 10. The two Lagrangian solutions for Arora model and TTM are practically identical, underlining the validity of the TTM to approximate the full-order behavior. Out of the Eulerian models, the simplified model again underpredicts the effective shear rate significantly. For systems with higher rotation rates of the impeller, we expect this discrepancy to grow further, as cell rotation becomes more significant. The Eulerian formulation of the TTM agrees overall well with its Lagrangian formulation. The Eulerian formulation is more diffusive due to the finite element discretization in space. The Lagrangian formulation does not require any mesh, as its discretization is only governed by the integration step size along its pathline. This causes some local discrepancies between the solutions. However, if we integrate empirical hemolysis models along a larger set of 36 pathlines, the averaged results agree within 5%. Overall, this comparison demonstrates that the Eulerian formulation of the TTM is a valid approximation of the Lagrangian Arora model for the given application.

## 5. Conclusion and outlook

We have demonstrated the capabilities of our new Eulerian strain-based hemolysis model, the tank-treading morphology model (TTM), in a variety of benchmark flows. Presently, its largest limitation is the underlying assumption of ellipsoidal cell shape. More complex dynamics and morphologies have been shown to occur in various flow conditions [19,40–42]. We argue that the present model nevertheless provides practical predictions for two reasons. First, fully resolving small-scale cell behavior requires a more complex structural model for the cell membrane. Such models currently only allow for simulation of small ensembles of cells, many orders of magnitude away from the total number of cells in realistic configurations. They are thus not suitable for the design process of VADs. Second, scale-resolving simulations by Lanotte et al. [41] at physiological *hematocrit* (volume fraction of RBCs) and physiological plasma viscosities indicate that at high shear rates, elongated flattened cells with tank-treading dynamics dominate. Even though their shapes may feature irregularities compared with pure ellipsoids, orientation and characteristic deformation time should behave similarly. We thus expect the TTM to still describe the deformation of such cells adequately.

In the future, we will examine if the effects of the aforementioned morphology irregularities can be incorporated into the model phenomenologically, e.g., by adapting the model parameters $f_1, f_2, f_3$. Additionally, we are aiming to obtain experimental data for cell deformation. This will allow for model validation and provide a basis for the calibration of the model parameters, especially in light of new results for relaxation time [43] and methods of varying the parameters [44]. Finally, we intend to apply the TTM to more realistic VAD geometries and compare its hemolysis predictions with computational and empirical reference results, e.g., for the FDA benchmark blood pump [8]. In this context, we are also planning to investigate the effects of turbulence modeling and variable hematocrit on strain-based hemolysis predictions.

Overall, we are confident that the TTM represents an excellent compromise between accuracy and practicality. In contrast to Eulerian stress-based models, the TTM is able to capture the viscoelastic behavior of the cell membrane by resolving cell deformation time. Compared to Lagrangian strain-based models, the TTM is computationally more efficient and does not require tracking individual cells. Compared to a previous Eulerian strain-based model [6], the TTM more accurately captures the original Lagrangian model behavior. These qualities will make it a valuable tool for the design process of future generations of medical devices.

## CRediT authorship contribution statement

**Nico Dirkes:** Writing – review & editing, Writing – original draft, Visualization, Software, Methodology, Investigation, Formal analysis. **Fabian Key:** Writing – review & editing, Writing – original draft, Software. **Marek Behr:** Writing – review & editing, Supervision, Software, Project administration, Methodology, Funding acquisition, Conceptualization.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## Acknowledgments

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) through grant 333849990/GRK2379 (IRTG Modern Inverse Problems). The authors gratefully acknowledge the computing time granted by the JARA Vergabegremium and provided on the JARA Partition part of the supercomputer CLAIX at RWTH Aachen University and the computing time provided to them on the high-performance computer Lichtenberg at the NHR Centers NHR4CES at TU Darmstadt. This is funded by the Federal Ministry of Education and Research, and the state governments participating on the basis of the resolutions of the GWK for national high performance computing at universities (www.nhr-verein.de/unsere-partner).

## Appendix A. On the definition of the rotation term

The left-hand side of the Arora model (2) is meant to represent a Jaumann derivative. The defining feature of a Jaumann derivative is that it is objective under rotation, i.e., in a purely rotational flow and neglecting shape recovery, it has to hold that

$$\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t} - [\boldsymbol{\Omega}, \mathbf{S}] = \mathbf{0}.$$

Again employing the spectral decomposition (4) and noting that the eigenvalues are constant in the case of pure rotation, we find that

$$\begin{aligned}\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t} &= \frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t}\boldsymbol{\Lambda}\mathbf{Q}^{\mathbf{T}} + \mathbf{Q}\boldsymbol{\Lambda}\frac{\mathrm{d}\mathbf{Q}^{\mathbf{T}}}{\mathrm{d}t},\\ [\boldsymbol{\Omega}, \mathbf{S}] &= \boldsymbol{\Omega}\mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathbf{T}} - \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}.\end{aligned}$$

Thus, objectivity is ensured if

$$\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t} = \boldsymbol{\Omega}\mathbf{Q} \qquad \text{and} \qquad \frac{\mathrm{d}\mathbf{Q}^{\mathbf{T}}}{\mathrm{d}t} = \mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}.$$

These equations are only satisfied by the definition (1). In contrast, Arora et al. [5] employed the definition

$$\tilde{\boldsymbol{\Omega}} = -\frac{\mathrm{d}\mathbf{Q}^{\mathbf{T}}}{\mathrm{d}t}\mathbf{Q} = \mathbf{Q}^{\mathbf{T}}\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t} = \mathbf{Q}^{\mathbf{T}}\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t}\mathbf{Q}\mathbf{Q}^{\mathbf{T}} = \mathbf{Q}^{\mathbf{T}}\boldsymbol{\Omega}\mathbf{Q},$$

which corresponds to a change of basis to the coordinate system of the cell. In tensor notation, the two definitions are equivalent.

## Appendix B. Finite element code

Our in-house multiphysics finite element code `XNS` is written in `Fortran` and parallelized by the in-house `EWD` communication library using the Message Passing Interface (MPI). This allows us to exploit the potential of modern high performance computing (HPC) architectures, especially for large-scale problems. `XNS` supports semi-discrete as well as space–time finite element formulations [35] and entails a number of advanced methods for moving and deforming grids [45–48]. It also provides capabilities to perform parametric model order reduction [49,50] using the `RBniCS`[2] library [51].

The design of `XNS` adheres to the object-oriented programming (OOP) paradigm. Here, we employ encapsulation to structure the code in a modular fashion, which comes with increased data security and promotes a more intuitive understanding of the code's design. In particular, the code is organized to separate the numerical discretization aspects, including computational mesh and time discretization, from the formulation of physical field equations and the incorporation of features such as stabilization schemes or discontinuity capturing. Furthermore, the design also facilitates coupling, e.g., between multiple field problems, which requires exchanging the corresponding solution fields. Here, `XNS` provides functionality to couple field problems both weakly and strongly, according to one-way or mutual dependencies.

The OOP approach also enables the usage of inheritance and polymorphism. On the one hand, inheritance is leveraged throughout the code to create a hierarchy of classes, allowing the derived classes to inherit properties and behaviors from parent classes. Thereby, common functionality can be efficiently implemented by reusing and sharing existing code. For example, the distribution of temperature or concentration can be governed by the same scalar transport equation, but with different material parameters or source terms. Using the concept of inheritance, this unified equation structure allows us to encapsulate the common functionality while still addressing the specific variations.

On the other hand, polymorphism is implemented, enabling objects of different classes to be treated as objects of a common superclass. As an illustrative example, the use of polymorphism allows the assembly process in the finite element procedure to be implemented only once for different field problems derived from a common superclass. This design approach allows seamless interchangeability and coupling of different problem types and avoids code duplication.

The above concepts are also particularly useful in the context of this work. Due to the modular design, it was straightforward to create an additional module for the morphology problem and integrate it into the existing framework of field problems.

All field problems are derived from a common parent class `FieldProblemT`. The flow problem is implemented as a subclass `INSProblemT`. For the different Eulerian models from Section 2, we followed the concept of inheritance to reflect their common

[2] RBniCS — reduced order modeling in FEniCS, https://www.rbnicsproject.org/.

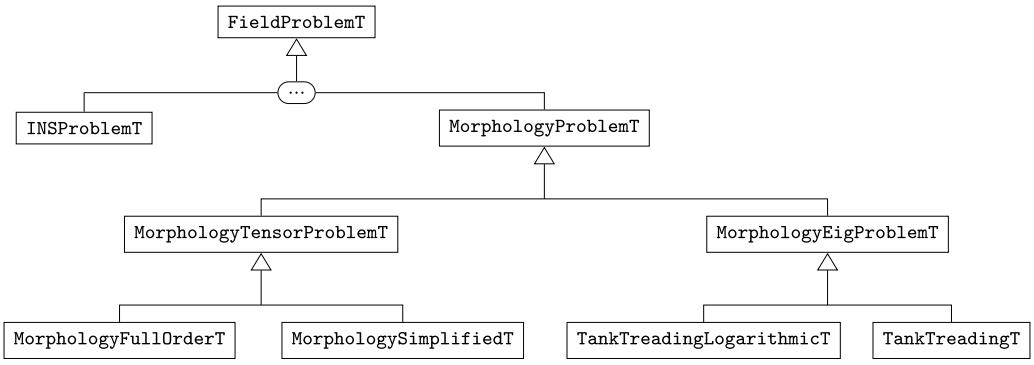



**Fig. B.11.** Inheritance structure of Eulerian field problems in XNS.

properties. We introduced a new parent class `MorphologyProblemT` related to the generic formulation (22). From this, we derived classes for the specific models that differ in number of degrees of freedom and in definition of the source term. `MorphologyTensorProblemT` is the parent class for the morphology models that involve the morphology tensor **S**. The full-order model (8) and the simplified model (18) are implemented as subclasses `MorphologyFullOrderT` and `MorphologySimplifiedT`, respectively. `MorphologyEigProblemT` implements the problems that operate directly on the eigenvalues $\lambda_i$ of **S**. The TTM from Eq. (10) and the TTLM from Eq. (12) are implemented as subclasses `TankTreadingT` and `TankTreadingLogarithmicT`, respectively. Fig. B.11 gives an overview of the classes and their inheritance.

As has been mentioned above, the flow and morphology problems are coupled in a weak manner. Thereby, the solution of the flow problem, i.e., the velocity field and its gradient, is used in the morphology problem.

## References

[1] T. Vos, S.S. Lim, et al., Global burden of 369 diseases and injuries in 204 countries and territories, 1990–2019: A systematic analysis for the global burden of disease study 2019, Lancet 396 (10258) (2020) 1204–1222, http://dx.doi.org/10.1016/S0140-6736(20)30925-9.

[2] G. Heuser, R. Opitz, A couette viscometer for short time shearing of blood, Biorheology 17 (1980) 17–24, http://dx.doi.org/10.3233/bir-1980-171-205.

[3] M. Giersiepen, L. Wurzinger, R. Opitz, H. Reul, Estimation of shear stress-related blood damage in heart valve prostheses—in vitro comparison of 25 aortic valves, Int. J. Artif. Organs. 13 (5) (1990) 300–306, http://dx.doi.org/10.1177/039139889001300507.

[4] C. Bludszuweit, Model for a general mechanical blood damage prediction, Artif. Organs 19 (7) (1995) 583–589, http://dx.doi.org/10.1111/j.1525-1594.1995.tb02385.x.

[5] D. Arora, M. Behr, M. Pasquali, A tensor-based measure for estimating blood damage, Artif. Organs 28 (11) (2004) 1002–1015, http://dx.doi.org/10.1111/j.1525-1594.2004.00072.x.

[6] L. Pauli, J. Nam, M. Pasquali, M. Behr, Transient stress-based and strain-based hemolysis estimation in a simplified blood pump, Int. J. Numer. Methods Biomed. Eng. 29 (2013) 1148–1160, http://dx.doi.org/10.1002/cnm.2576.

[7] M.M. Faghih, M.K. Sharp, Modeling and prediction of flow-induced hemolysis: A review, Biomech. Model. Mechanobiol. 18 (2019) 845–881, http://dx.doi.org/10.1007/s10237-019-01137-1.

[8] S.V. Ponnaluri, P. Hariharan, L.H. Herbertson, K.B. Manning, R.A. Malinauskas, B.A. Craven, Results of the interlaboratory computational fluid dynamics study of the FDA benchmark blood pump, Ann. Biomed. Eng. 51 (1) (2023) 253–269, http://dx.doi.org/10.1007/s10439-022-03105-w.

[9] J.N. Katz, B.C. Jensen, P.P. Chang, S.L. Myers, F.D. Pagani, J.K. Kirklin, A multicenter analysis of clinical hemolysis in patients supported with durable, long-term left ventricular assist device therapy, J. Heart Lung Transplant. 34 (5) (2015) 701–709, http://dx.doi.org/10.1016/j.healun.2014.10.002.

[10] M. Puig-de-Morales-Marinkovic, K.T. Turner, J.P. Butler, J.J. Fredberg, S. Suresh, Viscoelasticity of the human red blood cell, Am. J. Physiol. Cell Physiol. 293 (2) (2007) C597–C605, http://dx.doi.org/10.1152/ajpcell.00562.2006.

[11] T. Zhang, M.E. Taskin, H.-B. Fang, A. Pampori, R. Jarvik, B.P. Griffith, Z.J. Wu, Study of flow-induced hemolysis using novel Couette-type blood-shearing devices, Artif. Organs 35 (12) (2011) 1180–1186, http://dx.doi.org/10.1111/j.1525-1594.2011.01243.x.

[12] H.L. Goldsmith, J. Marlow, F.C. MacIntosh, Flow behaviour of erythrocytes – I. Rotation and deformation in dilute suspensions, Proc. R. Soc. Lond. [Biol.] 182 (1068) (1972) 351–384, http://dx.doi.org/10.1098/rspb.1972.0084.

[13] L.A. Down, D.V. Papavassiliou, E.A. O'Rear, Significance of extensional stresses to red blood cell lysis in a shearing flow, Ann. Biomed. Eng. 39 (6) (2011) 1632–1642, http://dx.doi.org/10.1007/s10439-011-0262-0.

[14] M.M. Faghih, M.K. Sharp, Deformation of human red blood cells in extensional flow through a hyperbolic contraction, Biomech. Model. Mechanobiol. 19 (1) (2020) 251–261, http://dx.doi.org/10.1007/s10237-019-01208-3.

[15] Y. Chen, M.K. Sharp, A strain-based flow-induced hemolysis prediction model calibrated by in vitro erythrocyte deformation measurements, Artif. Organs 35 (2) (2011) 145–156, http://dx.doi.org/10.1111/j.1525-1594.2010.01050.x.

[16] Y. Chen, T.L. Kent, M.K. Sharp, Testing of models of flow-induced hemolysis in blood flow through hypodermic needles, Artif. Organs 37 (3) (2013) 256–266, http://dx.doi.org/10.1111/j.1525-1594.2012.01569.x.

[17] H. Yu, S. Engel, G. Janiga, D. Thévenin, A review of hemolysis prediction models for computational fluid dynamics, Artif. Organs 41 (7) (2017) 603–621, http://dx.doi.org/10.1111/aor.12871.

[18] G. Arwatz, A.J. Smits, A viscoelastic model of shear-induced hemolysis in laminar flow, Biorheology 50 (1–2) (2013) 45–55, http://dx.doi.org/10.3233/BIR-130626.

[19] H.M. Ezzeldin, M.D. de Tullio, M. Vanella, S.D. Solares, E. Balaras, A strain-based model for mechanical hemolysis based on a coarse-grained red blood cell model, Ann. Biomed. Eng. 43 (6) (2015) 1398–1409, http://dx.doi.org/10.1007/s10439-015-1273-z.

[20] S. Sohrabi, Y. Liu, A cellular model of shear-induced hemolysis, Artif. Organs 41 (9) (2017) E80–E91, http://dx.doi.org/10.1111/aor.12832.

[21] C. Porcaro, M. Saeedipour, Hemolysis prediction in bio-microfluidic applications using resolved CFD-DEM simulations, Comput. Methods Programs Biomed. 231 (2023) 107400, http://dx.doi.org/10.1016/j.cmpb.2023.107400.
[22] D.A. Fedosov, B. Caswell, G.E. Karniadakis, A multiscale red blood cell model with accurate mechanics, rheology, and dynamics, Biophys. J. 98 (10) (2010) 2215–2225, http://dx.doi.org/10.1016/j.bpj.2010.02.002.
[23] T. Klöppel, W.A. Wall, A novel two-layer, coupled finite element approach for modeling the nonlinear elastic and viscoelastic behavior of human erythrocytes, Biomech. Model. Mechanobiol. 10 (4) (2011) 445–459, http://dx.doi.org/10.1007/s10237-010-0246-2.
[24] S. Mendez, E. Gibaud, F. Nicoud, An unstructured solver for simulations of deformable particles in flows at arbitrary Reynolds numbers, J. Comput. Phys. 256 (2014) 465–483, http://dx.doi.org/10.1016/j.jcp.2013.08.061.
[25] G. Závodszky, B. van Rooij, V. Azizi, A. Hoekstra, Cellular level in-silico modeling of blood rheology with an improved material model for red blood cells, Front. Physiol. 8 (2017) http://dx.doi.org/10.3389/fphys.2017.00563.
[26] C. Kotsalos, J. Latt, B. Chopard, Bridging the computational gap between mesoscopic and continuum modeling of red blood cells for fully resolved blood flow, J. Comput. Phys. 398 (2019) 108905, http://dx.doi.org/10.1016/j.jcp.2019.108905.
[27] F. Guglietta, M. Behr, L. Biferale, G. Falcucci, M. Sbragaglia, On the effects of membrane viscosity on transient red blood cell dynamics, Soft Matter 16 (2020) 6191–6205, http://dx.doi.org/10.1039/D0SM00587H.
[28] G. Foster, Third-generation ventricular assist devices, in: S.D. Gregory, M.C. Stevens, J.F. Fraser (Eds.), Mechanical Circulatory and Respiratory Support, Academic Press, 2018, pp. 151–186, http://dx.doi.org/10.1016/B978-0-12-810491-0.00005-9.
[29] H. Schmid-Schönbein, R. Wells, Fluid drop-like transition of erythrocytes under shear, Science 165 (3890) (1969) 288–291.
[30] T. Fischer, M. Stöhr-Liesen, H. Schmid-Schönbein, The red cell as a fluid droplet: Tank tread-like motion of the human erythrocyte membrane in shear flow, Science 202 (4370) (1978) 894–896, http://dx.doi.org/10.1126/science.715448.
[31] P. Maffettone, M. Minale, Equation of change for ellipsoidal drops in viscous flow, J. Non-Newton. Fluid Mech. 78 (1998) 227–241, http://dx.doi.org/10.1016/S0377-0257(98)00065-2.
[32] S. Haßler, L. Pauli, M. Behr, The variational multiscale formulation for the fully-implicit log-morphology equation as a tensor-based blood damage model, Int. J. Numer. Methods Biomed. Eng. 35 (12) (2019) e3262–1–e3262–19, http://dx.doi.org/10.1002/cnm.3262.
[33] L. Goubergrits, Numerical modeling of blood damage: current status, challenges and future prospects, Expert Rev. Med. Devices 3 (5) (2006) 527–531, http://dx.doi.org/10.1586/17434440.3.5.527.
[34] L. Pauli, Stabilized Finite Element Methods for Computational Design of Blood-Handling Devices (Ph.D. thesis), RWTH Aachen University, Aachen, Germany, 2016.
[35] L. Pauli, M. Behr, On stabilized space-time FEM for anisotropic meshes: incompressible Navier-Stokes equations and applications to blood flow in medical devices, Internat. J. Numer. Methods Fluids 85 (3) (2017) 189–209, http://dx.doi.org/10.1002/fld.4378.
[36] L. Pauli, J. Both, M. Behr, Stabilized finite element method for flows with multiple reference frames, Internat. J. Numer. Methods Fluids 78 (11) (2015) 657–669, http://dx.doi.org/10.1002/fld.4032.
[37] T. Hughes, M. Mallet, A new finite element formulation for computational fluid dynamics: III. The generalized streamline operator for multidimensional advective-diffusive systems, Comput. Methods Appl. Mech. Engrg. 58 (1986) 305–328, http://dx.doi.org/10.1016/0045-7825(86)90152-0.
[38] Y. Bazilevs, V.M. Calo, T.E. Tezduyar, T.J.R. Hughes, YZ$\beta$ discontinuity capturing for advection-dominated processes with application to arterial drug delivery, Internat. J. Numer. Methods Fluids 54 (6–8) (2007) 593–608, http://dx.doi.org/10.1002/fld.1484.
[39] K.H. Fraser, M.E. Taskin, B.P. Griffith, Z.J. Wu, The use of computational fluid dynamics in the development of ventricular assist devices, Med. Eng. Phys. 33 (2011) 263–280, http://dx.doi.org/10.1016/j.medengphy.2010.10.014.
[40] D.A. Fedosov, M. Peltomäki, G. Gompper, Deformation and dynamics of red blood cells in flow through cylindrical microchannels, Soft Matter 10 (24) (2014) 4258–4267, http://dx.doi.org/10.1039/C4SM00248B.
[41] L. Lanotte, J. Mauer, S. Mendez, D.A. Fedosov, J.-M. Fromental, V. Claveria, F. Nicoud, G. Gompper, M. Abkarian, Red cells' dynamic morphologies govern blood shear thinning under microcirculatory flow conditions, Proc. Natl. Acad. Sci. USA 113 (47) (2016) 13289–13294, http://dx.doi.org/10.1073/pnas.1608074113.
[42] J. Mauer, S. Mendez, L. Lanotte, F. Nicoud, M. Abkarian, G. Gompper, D.A. Fedosov, Flow-induced transitions of red blood cell shapes under shear, Phys. Rev. Lett. 121 (11) (2018) 118103, http://dx.doi.org/10.1103/PhysRevLett.121.118103.
[43] F. Guglietta, M. Behr, G. Falcucci, M. Sbragaglia, Loading and relaxation dynamics of a red blood cell, Soft Matter 17 (2021) 5978–5990, http://dx.doi.org/10.1039/D1SM00246E.
[44] D. Taglienti, F. Guglietta, M. Sbragaglia, Reduced model for droplet dynamics in shear flows at finite capillary numbers, Phys. Rev. Fluids 8 (1) (2023) 013603, http://dx.doi.org/10.1103/PhysRevFluids.8.013603.
[45] D. Hilger, N. Hosters, F. Key, S. Elgeti, M. Behr, A novel approach to fluid-structure interaction simulations involving large translation and contact, in: H. Van Brummelen, C. Vuik, M. Möller, C. Verhoosel, B. Simeon, B. Jüttler (Eds.), in: Isogeometric Analysis and Applications 2018, vol. 133, Springer International Publishing, Cham, 2021, pp. 39–56, http://dx.doi.org/10.1007/978-3-030-49836-8_3.
[46] J. Helmig, F. Key, M. Behr, S. Elgeti, Combining boundary-conforming finite element meshes on moving domains using a sliding mesh approach, Internat. J. Numer. Methods Fluids 93 (4) (2021) 1053–1073, http://dx.doi.org/10.1002/fld.4919.
[47] F.A. González, S. Elgeti, M. Behr, The surface-reconstruction virtual-region mesh update method for problems with topology changes, Internat. J. Numer. Methods Engrg. 124 (9) (2023) 2050–2067, http://dx.doi.org/10.1002/nme.7200.
[48] F. Key, L. Pauli, S. Elgeti, The virtual ring shear-slip mesh update method, Comput. & Fluids 172 (2018) 352–361, http://dx.doi.org/10.1016/j.compfluid.2018.04.006.
[49] F. Key, M. von Danwitz, F. Ballarin, G. Rozza, Model order reduction for deforming domain problems in a time-continuous space-time setting, Internat. J. Numer. Methods Engrg. 124 (23) (2023) 5125–5150, http://dx.doi.org/10.1002/nme.7342.
[50] F. Key, F. Ballarin, S. Eusterholz, S. Elgeti, G. Rozza, Reduced flow model for plastics melt inside an extrusion die, Proc. Appl. Math. Mech. 21 (1) (2021) e202100071, http://dx.doi.org/10.1002/pamm.202100071.
[51] J.S. Hesthaven, G. Rozza, B. Stamm, Certified reduced basis methods for parametrized partial differential equations, in: SpringerBriefs in Mathematics, Springer International Publishing, 2015, http://dx.doi.org/10.1007/978-3-319-22470-1.