A PHASE-FIELD METHOD APPLIED TO INTERFACE TRACKING FOR BLOOD CLOT FORMATION

Marek Čapek, Praha

Received March 10, 2019. Published online July 9, 2020.

Abstract. The high shear rate thrombus formation was only recently recognized as another way of thrombosis. Models proposed in Weller (2008), (2010) take into account this type of thrombosis. This work uses the phase-field method to model these evolving interface problems. A loosely coupled iterative procedure is introduced to solve the coupled system of equations. Convergence behavior on two levels of refinement of perfusion chamber geometry and cylinder geometry is then studied. The perfusion chamber simulations show good agreement with the original results of Weller. The code is implemented in FEM-library deal.ii Alzeta et al. (2018), which enables distribution of computations to large number of processing units. A scalability and numerical performance study of the loosely coupled iterative procedure is performed, combined with several preconditioners for the linear subproblems.

Keywords: thrombus growth; free boundary problem; fluid dynamics; phase field method; finite element method; scalability; high shear rate thrombosis

MSC 2020: 76D05, 76M10, 76T99

1. Introduction of high shear rate thrombus formation

Blood coagulation is a natural process whose aim is to stop blood loss in the case of the vessel injury. It is so potent that it saves life not only in the case of a minor vessel injury, but also during serious injuries in the larger arteries and veins.

However, this process can be triggered by other circumstances in the human body as well. The most well known condition of blood coagulation is the stasis of blood in the vessel. It is one of the key elements of Virchow's triad, which is known since the 19th century [14], [20], [13]. This state is mostly present in the smaller vessels.

This work was supported by The Ministry of Education, Youth and Sports from the Large Infrastructures for Research, Experimental Development and Innovations project "IT4Innovations National Supercomputing Center—LM2015070".

For a long time, Virchow's triad was accepted as the only possible way of creation of thrombi in the blood flow.

Nevertheless, current research shows that thrombosis can be started even under conditions where the stasis of blood is by far not present. To be precise, the thrombosis may appear under conditions of high shear rate in the blood flow even in larger arteries. For details, see [11], [12], and [26].

We will not delve into the thrombosis, which originates from the fulfilment of the stasis condition of Virchow's triad, as that is not aim of our study. We can however observe from experiments that thrombi in the larger arteries are platelet rich, in contrast with the thrombi in the smaller vessels.

Recent studies have provided an explanation for this fact. The main precondition of both types of clotting is the same—a reactive surface. This thrombogenic surface can be subendothelial collagen exposed by injury to the vessel wall or to artificial material of a prosthetic device, like polyester or stainless steel. On the surface, platelets and adhesive proteins like fibrinogen and the von Willebrand factor (vWF) are caught. It is obvious that the coverage of the surface by platelets or proteins is determined by the concentration of platelets and proteins in the vicinity of the thrombogenic surface.

Concentrations of platelets and proteins in the blood under static conditions are minute. However, once the blood is under shear, red blood cells push the other blood species outside the center of the vessel. The increased diffusivity causes margination of platelets and proteins. Therefore, they are at the reactive surface in increased concentration (see [1]).

Although vWF is in the blood present in a low concentration, its chemical composition makes it the key actor in the high shear type of blood coagulation. It is contained in blood plasma, platelet α-granules, and the extracellular matrix of blood vessels. It creates chemical bonds mainly with itself, fibrillar collagen, platelet receptor glycoprotein (GP) Ib or platelet integrin $\alpha_{\mathrm{IIb}}\beta_3$ (GPIIb/IIIa).

The molecule of vWF has vWF-A1 binding sites on its surface. The availability of the sites depends on the conformation of the molecule. Namely, in the globular form, some part of these sites is hidden from the surrounding environment. On the other hand, in the transition of vWF into the stretched conformation the binding sites are more exposed. The changeover from the globular to the elongated form is caused by the increase of shear rate in the blood flow and it is reversible. The stretched molecules of vWF form a network in which the majority of the platelets in the vicinity of the vessel wall get caught. Even the fast moving platelets are captured by this net.

It is important to note that the platelets do not need to be activated in order to be captured. Namely, vWF-A1 binding sites associate with GPIb receptors on the surface of platelets, which are always prepared to bind [12].

It can be argued that the prevalence of this process in the case of high shear rate thrombogenesis follows from the fact that the time of passing of platelets through stenosis is too short for platelets to activate. So, as the platelets actually bind, one can conclude that the overwhelming factor of the capture of platelets into the thrombus is the affinity of the A1 binding sites of vWF and the GPIb receptors of platelets in the blood flow.

After this, captured mural platelets get activated due to exposition to high shear rates for a longer time. The time necessary for adsorption and activation of a platelet is called the *lagtime*. It depends on the shear rate—it decreases with higher shear rate. This dependence can have two reasons. Firstly, proteins and platelets are marginated in elevated measure due to high values of shear rate, as noted above. Secondly, high shear rate reduces the time of mural platelet activation.

The newly activated platelets set free platelet granule stores of vWF and other procoagulant species, which either stabilize the thrombus (such as platelet integrin $\alpha_{\mathrm{IIb}}\beta_3$) or are carried away by the flow. The newly released vWF is captured into already existing nets and enhances further platelet capture. This positive feedback leads to tremendous growth of the thrombus. The growing thrombus changes rheological conditions in the blood vessel, as its diameter is narrowing. This causes on increase of the shear rate, which again stimulates the growth of the thrombus. The initially captured platelets do not need to be stable, which can lead to occurrence of rotating platelet aggregates at high shear rates, which do not have to be activated. Either platelet integrin $\alpha_{\mathrm{IIb}}\beta_3$ irreversibly bounded to vWF or fibrinogen converted to fibrin can stabilize the high shear rate thrombus. If the bond between platelet integrin $\alpha_{\mathrm{IIb}}\beta_3$ and vWF is too weak, the high shear stress can tear off parts of the thrombus and form emboli [11], [12].

The high shear rate thrombus (white clot) can be seen as a precursor of the red clot under certain circumstances. Namely, the stasis condition created by the white clot is necessary for the set up of red clot (one of conditions of Virchow's triad).

The process of high shear thrombus formation is summed up in Table 1. Figure 1 shows a graphical representation of catching of platelets (red ones) into the vWF nets (green ones). See [18] for details.

We note also that the activation of platelets could be dependent not only on the magnitude of shear rate, the form of vWF molecule and platelet sensitivity, but also on the size of the vWF molecule. In [35], a *platelet activation risk index* (PARI) is introduced, which depends on vWFs size, shear rate and platelet sensitivity. This shows that there is ongoing research on the dependence of platelet activation on the mentioned factors.

(1)	In the stenosed region, high values of wall shear rate appear.
(2)	To walls composed of collagen or artificial surface molecules of vWF bind; due to the high values of wall shear rate they uncoil.
(3)	Blood platelets are marginated to the walls due to the shear enhanced diffusivity caused by pushing red blood cells in the center of the vessel.
(4)	Unactivated platelets near the wall are caught by the vWF, which is already bound to the vessel wall.
(5)	Already bound platelets release vWF in large amounts, which activate $\alpha_{\mathrm{IIb}}\beta_3$ on the platelets. This causes firm adhesion.
(6)	Due to the capture of platelets, nets created from vWF are created on the surface of the emerging thrombus.
(7)	The networks of vWF catch many platelets, quickly which create a large thrombus. The clot can either occlude the vessel or embolize.

Table 1. Process of high shear rate thrombus formation.

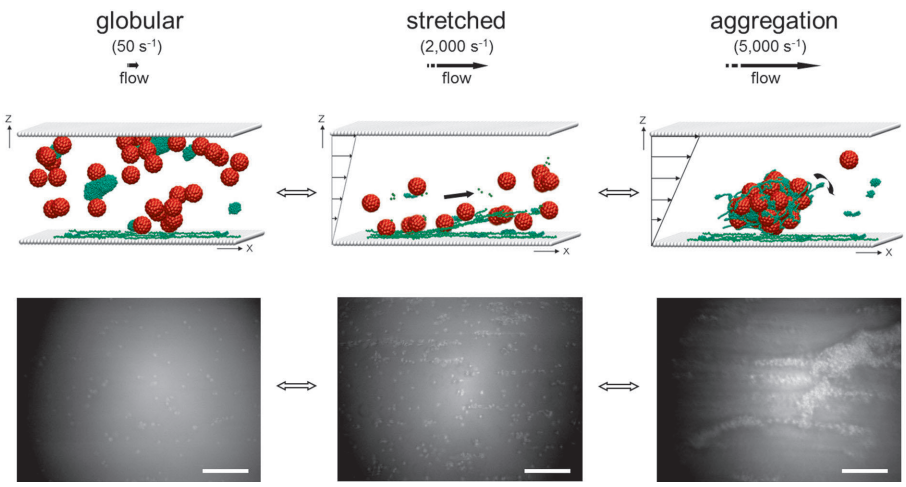


Figure 1. VWF networks transition from globular to stretched to aggregation.

2. Mathematical modelling of blood coagulation

The complexity of blood coagulation attracted scientists to formulating various mathematical models. They differ one from another in the extent of incorporated chemical and rheological processes, which take part in the model of blood coagulation process.

It is therefore questionable how to categorize different models of blood coagulation. In [9] a two dimensional classification based on two criteria is proposed—the scale of the processes which the model is able to capture, and the extent of the feature that the model is able to incorporate.

Initially there were two groups of models—a group of rheological models with simplified biochemistry and a group of biochemical models without any connection to the rheology. The former focused on the modelling of blood flow, almost entirely ignoring the chemical reactions occurring in the vessel. There it is supposed that the blood coagulation occurs in the steady state of enclosing blood plasma. Therefore, the development of species concentration was modelled by ordinary differential equations. The latter approach concentrated on the formulation of equations for the reacting species in the flowing blood. This approach resulted in partial differential equations for the species, which contained reactive, diffusive and convective parts. However, gradually it has become clear that biologically relevant models must encompass both approaches. The models, which resulted from this recognition, differ again in the level of detail.

From the rheological point of view, the blood could have been taken as a Newtonian or a non-Newtonian fluid. The non-Newtonian fluid can be generally defined by several features, like shear-rate dependent viscosity or viscoelasticity.

The treatment of the biochemical part of coagulation in the more encompassing models differed mainly in the number of chemical species and number of types of bodies (e.g. platelets), for which equations were formulated. Contrary to the model of coagulation in the stationary state, the formulated equations contained, beside reaction terms and time derivative terms, also diffusive terms and convective terms. The distribution of chemical species influenced the rheological properties.

Such new comprehensive models, where the biochemistry is coupled with rheology, gave rise to a full stack of new problems and challenges. However, some of these models show better correspondence with simplified blood clotting simulations than previous ones that neglected the interdependence of physical and biochemical processes occurring in the blood clotting.

It is usually impossible to get an analytical solution with a closed form for these coupled models. The solution must be found using numerical experiments. The onset of high performance computers enabled the solution of large nonlinear systems of equations, which are part of these models.

In [4] and [23] the activation of platelets can proceed in two ways—biochemical and rheological. The rheological activation depends on the so-called activation number, which is assigned to each platelet. The activation number depends on the length of exposure of a platelet to elevated shear stress.

Shear rate dependent adhesion of activated platelets was introduced into the multiscale model of [34]. The platelets become activated when the concentration of relevant chemical species attains a specific threshold. The macro description of the blood flow is done by the Navier-Stokes equations, whereas micro level is modelled by stochastic Cellular Potts Model.

In the model [28], the adhesion of platelets to thrombus is taken as a shear rate and concentration of erythrocytes dependent process. However, the activation of platelets occurs only due to certain species circulating in the blood. The article [29] presents yet another approach. The platelets are activated either due to elevated levels of chemical agonists or due to reaching of certain threshold of shear stress.

We will now turn our attention to the model of Weller [31], [33]. The occurrence of platelet aggregation without action of any chemical species could not be satisfactorily explained at that time, as the unfolding of globular vWF caused by the high shear rate was a controversial idea. Nevertheless, Weller's model introduced the idea that the aggregation rate of platelets is dependent on shear rate. The actual uncoiling of vWF is hidden in an aggregation rate function, as tracing the concentration and the conformation (globular or stretched) of vWF would be a difficult task.

In the following sections, we introduce Weller's free boundary problem. The problem was solved numerically using the level-set method for tracking interface between blood and clot. In contrast with Weller, we employ a phase field, which serves in our system as an indicator. Namely, the phase field tells us in which part of the computational domain blood flows and in which part the growing clot resides.

We use the library deal.ii [5], which allows us to distribute the computations across many processing units on modern clusters. We use the message passing communication pattern (i.e. not the shared programming using threads). Each core inside of a node on the cluster is taken as an independent processing unit, which communicates with other cores using the message passing interface (MPI). The other cores can be either on the same node or on some other node of the cluster. Hence, in the following, the term 'core' denotes the independent processing unit exchanging messages with other cores, i.e. other processing units, by passing of messages.

Further development of Weller's model was already performed in [29], where a model comprising more biochemistry was formulated. In [29], the level-set method was used for tracing of the development of thrombus as well.

The main contribution of this article is twofold. Firstly, we formulate the free boundary problem originating from [31] using the phase-field formalism. The original model [31] is reviewed in Section 3, whereas the reformulation is outlined in Section 4. On the one hand, the phase field method formulation adds to the complexity of the equations to be solved, but on the other hand it appears to us that it has some favourable properties. Secondly, we perform tests of our implementation and study

some numerical properties of the proposed iterative scheme. Afterwards we study whether our code scales from the point of view of both weak and strong scaling.

3. Review of the model of Weller

3.1. Formulation of the free boundary problem. Let us introduce the basic setting of the free boundary problem, see Fig. 2.

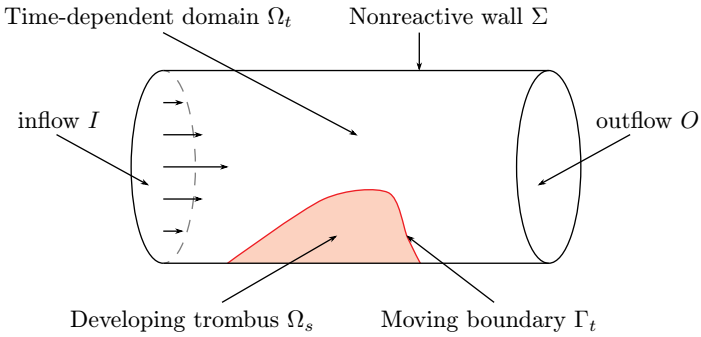


Figure 2. Computational domain.

In this picture, we have the time independent parts, namely inflow I, non-reactive wall Σ and outflow O. The blood flowing from the inflow I carries new platelets, some of which get incorporated on the moving boundary Γ_t into the developing thrombus Ω_s. As the thrombus grows, the area of flowing blood Ω_t reduces. Hence, the growing clot affects the flow.

Firstly we define the domains, with T as the end time of simulation:

- ▷ inflow boundary: $I_T := I \times (0, T]$
- ▷ outflow boundary: $O_T := O \times (0, T]$
- ▷ non-reactive wall $\Sigma_T := \Sigma \times (0, T]$
- ▷ time dependent area of blod flow: $\Omega_T := \bigcup\limits_{0<t\leqslant T} \Omega_t \times \{t\}$
- ▷ moving boundary: $\Gamma_T := \bigcup\limits_{0<t\leqslant T} \Gamma_t \times \{t\}$

Relevant equations are then formulated in these domains. The whole computational domain Ω^* is defined as $\Omega^* = \Omega_t(t) \cup \Gamma_t(t) \cup \Omega_s(t)$. The model contains Navier-Stokes equations for the blood, as blood is taken as a Newtonian fluid with constant viscosity ν. The blood platelets are distributed through the blood and obey a transport equation.

The free boundary model then consists of the Navier-Stokes part:

$$
\begin{aligned}
\frac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p &= \mathbf{f} && \text{in } \Omega_T, \\
\nabla \cdot \mathbf{u} &= 0 && \text{in } \Omega_T, \\
\mathbf{u} &= 0 && \text{on } \Gamma_T \cup \Sigma_T, \\
\mathbf{u} &= \mathbf{u}_D && \text{on } I_T, \\
\mathbf{o} \cdot \nabla \mathbf{u} &= p\mathbf{o} && \text{on } O_T, \\
\mathbf{u}(t=0) &= \mathbf{u}_0 && \text{in } \Omega_0,
\end{aligned} \tag{3.1}
$$

where $\mathbf{u}$ is the velocity of blood, p is the pressure, $\mathbf{f}$ is the external force field, like gravity, $\mathbf{o}$ is the unit outer normal to the outflow boundary O. Furthermore, $\mathbf{u}_D$ is the Dirichlet boundary condition imposed on velocity on the inflow I, $\mathbf{u}_0$ is the initial condition on the velocity, and ν is the kinematic viscosity.

The transport equation for the platelet concentration w has the following form:

$$
\frac{\partial w}{\partial t} - D \Delta w + \mathbf{u} \cdot \nabla w = 0 \qquad \text{in } \Omega_T, \tag{3.2}
$$

$$
\begin{aligned}
D\mathbf{n} \cdot \nabla w &= kw && \text{on } \Gamma_T, \\
\mathbf{n} \cdot \nabla w &= 0 && \text{on } \Sigma_T \cup O_T, \\
w &= w_D && \text{on } I_T, \\
w(t=0) &= w_0 && \text{in } \Omega_0,
\end{aligned} \tag{3.3}
$$

where D is the constant diffusivity, $\mathbf{u}$ is the velocity field obtained from the solution of the Navier-Stokes equations (3.1), $\mathbf{n}$ is the unit outer normal to the reactive interface Γ_T (see Fig. 2), w_D is the Dirichlet boundary condition on platelets, and w_0 is the initial condition on platelets.

The parameter k is the adhesion rate, which depends on shear stress s, to be defined later on:

$$
k := \kappa_1 + \kappa_2 s. \tag{3.4}
$$

We assume that the interface clot-blood grows in the opposite direction to the influx of platelets on the surface of the clot. Hence, the growth velocity of the interface $\mathbf{u_i}$ is supposed to be in the same direction as platelet gradient, i.e., the proportionality constant α is chosen as positive:

$$
\mathbf{u_i} = \alpha \nabla w \quad \text{on } \Gamma_T. \tag{3.5}
$$

We now use the main idea of the level set method (for details see e.g. [27] and [24])—the moving interface Γ_T can be represented implicitly as the zero level set of a function ϕ, whose sign can serve to distinguish the domains of *fluid blood* $\Omega_f(t) := \Omega_t$ and emerging thrombus $\Omega_s(t)$. To be more specific, the different parts of the domain are discerned by the function ϕ in the following manner:

$$\Omega_t(t) = \{x \in \Omega^* \colon \phi(x,t) > 0\}, \quad \Gamma_t(t) = \{x \in \Omega^* \colon \phi(x,t) = 0\}, \quad \Omega_s(t) = \{x \in \Omega^* \colon \phi(x,t) < 0\}. \tag{3.6}$$

It follows from the definition of ϕ, that the transport equation

$$\frac{\partial \phi}{\partial t} + \mathbf{u_i} \cdot \nabla \phi = 0 \tag{3.7}$$

has to be fulfilled along the moving interface. The natural extension to the whole domain Ω^* can be formulated as

$$\frac{\partial \phi}{\partial t} + \alpha k D^{-1} w |\nabla \phi| = 0. \tag{3.8}$$

At time $t = 0$, the level set function is initialized to the signed distance function, that is

$$\phi(x,0) := \begin{cases} \operatorname{dist}(x, \Gamma_0), & x \in \Omega_t \cup \Gamma_0, \\ -\operatorname{dist}(x, \Gamma_0), & x \in \Omega_s(0). \end{cases} \tag{3.9}$$

Weller [31] substituted the following term, obtained by integration by parts, over the interface Γ_t

$$\int_{\Gamma_t} \left(-\nu \frac{\partial \mathbf{u}}{\partial n} + p\mathbf{n} \right) \cdot \boldsymbol{\psi} \, \mathrm{d}S \tag{3.10}$$

by the following expression

$$\beta(h) \int_{\Gamma_t} \mathbf{u} \cdot \boldsymbol{\psi} \, \mathrm{d}S, \tag{3.11}$$

where β is a function depending on the cell size h so that $\beta(h) \to \infty$ as $h \to 0$. The $\boldsymbol{\psi}$ term is a vector test function. The velocity field is extended to Ω_s to fulfill $\mathbf{u} = \mathbf{0}$ in Ω_s, which leads to the additional term $\int_{\Omega_s} \mathbf{u} \cdot \boldsymbol{\psi} \, \mathrm{d}x$. The pressure is harmonically extended into the solid. On the part of the boundary $\partial\Omega_s(t) \setminus \Gamma_t$, the homogeneous Dirichlet boundary condition on $\mathbf{u}$ is imposed.

4. Realization of Weller's model using diffuse interface method

The sharp interface method, for example the level-set method as introduced in the previous section, provides an elegant way to compute useful quantities, like the normal vector to the interface, and the characteristic function of the interface, the Heaviside function of the flow domain. However, for the numerical computations one needs a smooth approximation of these quantities. The phase field method circumvents this issue by implicitly assuming that a smooth transition region around the interface is present.

In the following, we introduce the phase-field method as it is used for our purposes. In Section 4.1, we describe the movement of the interface blood-clot described using the phase-field function. In Section 4.2, we adjust the transport equation of platelets to our purposes using the phase-field function.

4.1. Interface tracking using the phase-field method. The most important advantages of the phase-field approach [15] in comparison with the level-set method are:

- ▷ no necessity of artificial smearing of the characteristic and Heaviside functions for numerical computations, as the phase field is implicitly smeared by the phase field equation (to be specified below)
- ▷ the phase-field function does not require the reinitalization procedure, which is necessary for the signed distance function of the level-set method

At the heart of our phase-field formulation is the Cahn-Hilliard equation, wherein the velocity $\tilde{\mathbf{u}}$ is given as the sum of the external part $\mathbf{u_e}$, originating from flow of the fluid, and internal component $\mathbf{u_i}$, stemming from the growth of interface due to the influx of platelets, i.e., it holds that $\tilde{\mathbf{u}} = \mathbf{u_e} + \mathbf{u_i}$. We take the growth velocity $\mathbf{u_i}$ as in (3.5)

$$\mathbf{u_i} = \alpha \nabla w, \tag{4.1}$$

as we will need this form in the derivation of modified phase-field equations. In order to treat the Cahn-Hilliard equation appropriately, it is reformulated as two coupled equations:

$$\frac{\partial c}{\partial t} + \tilde{\mathbf{u}} \cdot \nabla c - \nabla \cdot M \nabla \mu = 0 \quad \text{in } \Omega^*, \tag{4.2}$$

$$\mu - \frac{1}{\varepsilon^2} W'(c) + \Delta c = 0 \quad \text{in } \Omega^*, \tag{4.3}$$

where c is the phase field, with values between -1 and 1, μ is the chemical potential, $W'(c)$ is the derivative of the double-well potential $W(c)$, M is the mobility, and ε

is the interface thickness. The system of equations (4.2) and (4.3) is derived in [15], based on thermodynamic reasoning.

We use the Cahn-Hilliard system only for the tracking of the interface blood-clot without any consideration of thermodynamics. In other works, e.g. [15], the phase-field is taken as a variable reflecting thermodynamic state of the system.

Therefore, we solve a system of two equations, which is by one more than in the case of the level-set method. The Cahn-Hilliard equations are to be solved in the whole computational domain Ω^*.

Further we assume that the blood flow does not exhibit mechanical influence on the emerging clot. The blood affects the clot only by carrying blood platelets into the vicinity of the clot surface. This assumption was taken in Weller's original model [31]. Hence, we neglect the term $\mathbf{u_e}$, i.e. $\tilde{\mathbf{u}} = \mathbf{u_i}$ in (4.2). We manipulate the equation (4.2) in the following manner:

$$
\begin{aligned}
\frac{\partial c}{\partial t} + \mathbf{u_i} \cdot \nabla c - \nabla \cdot M \nabla \mu &= \frac{\partial c}{\partial t} + \mathbf{u_i} \cdot \nabla c - \nabla \cdot M \nabla \mu \\
&= \frac{\partial c}{\partial t} + \alpha \nabla w \cdot \nabla c - \nabla \cdot M \nabla \mu \\
&= \frac{\partial c}{\partial t} + \alpha \nabla w \cdot \frac{\nabla c}{|\nabla c|} |\nabla c| - \nabla \cdot M \nabla \mu \\
&= \frac{\partial c}{\partial t} + \alpha \nabla w \cdot \mathbf{n} |\nabla c| - \nabla \cdot M \nabla \mu \\
&= \frac{\partial c}{\partial t} + \alpha D^{-1} k w |\nabla c| - \nabla \cdot M \nabla \mu = 0,
\end{aligned} \tag{4.4}
$$

where we used (3.3) and the fact that the following relation holds for the phase-field function c:

$$
\mathbf{n} = \frac{\nabla c}{|\nabla c|}. \tag{4.5}
$$

The term $|\nabla c|$ is due to its delta function-like behaviour substituted by a different expression. Namely, we approximate the delta function by

$$
|\nabla \varphi_\varepsilon| = \frac{(\varphi_\varepsilon)^2}{\varepsilon} \mathrm{e}^{c/\varepsilon},
$$

where

$$
\varphi_\varepsilon(c) = \frac{1}{1 + \mathrm{e}^{c/\varepsilon}},
$$

with ε being a small constant parameter, k is the adhesion rate (3.4).

Finally, we get the following equations:

$$\frac{\partial c}{\partial t} - \nabla \cdot M \nabla \mu = \alpha D^{-1} k w |\nabla \varphi_\varepsilon(c)| \quad \text{in } \Omega^*, \tag{4.6}$$

$$\mu - \frac{1}{\varepsilon^2} W'(c) + \Delta c = 0 \quad \text{in } \Omega^*. \tag{4.7}$$

The viscous traction vector which is exerted by the wall on the fluid is defined as

$$\boldsymbol{\tau} = 2\nu \mathbf{D}(\mathbf{u})\mathbf{n}, \tag{4.8}$$

where $\mathbf{D}(\mathbf{u})$ is the symmetric velocity gradient, i.e. $\mathbf{D}(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + (\nabla \mathbf{u}^T))$. The normal component of the traction vector is obtained as

$$\tau_n = \boldsymbol{\tau} \cdot \mathbf{n}, \tag{4.9}$$

whereas the shear part of the viscous traction vector is obtained as

$$\tau_t = \boldsymbol{\tau} \cdot \mathbf{t}, \tag{4.10}$$

where $\mathbf{t}$ is the unit tangent, which is oriented parallel to the fluid velocity near the wall. The viscous traction vector can be then written as

$$\boldsymbol{\tau} = \boldsymbol{\tau}_n + \boldsymbol{\tau}_t = \tau_n \mathbf{n} + \tau_t \mathbf{t}. \tag{4.11}$$

The wall shear stress s is computed as l^2-norm of the shear traction vector $\boldsymbol{\tau}_t$:

$$s = \|\boldsymbol{\tau}_t\|_{l^2} = \|\boldsymbol{\tau} - \boldsymbol{\tau}_n\|_{l^2} = \|\boldsymbol{\tau} - (\boldsymbol{\tau} \cdot \mathbf{n})\mathbf{n}\|_{l^2}. \tag{4.12}$$

4.2. Transport of platelets using the phase-field framework. We have two areas in our computational domain, which have to be described by the transport equation of platelets. It is the fluid area Ω_t and the solid area Ω_s of the clot. For simplicity, we do not track the further development of the platelets, which are incorporated into the solid, the thrombus. Therefore, we suppose, that the density of platelets in the solid is constant. Our starting point is the transport equation formulated by Weller, i.e. equation (3.2).

On the interface between the solid Ω_s and the fluid Ω_t, we impose the Neumann-boundary condition using the specific procedure developed in [22]. The approach [22] was developed to approximate the boundary condition imposed on the boundaries of domains on which elliptic equations were defined. The procedure enables time development of domains in space; hence, it suits our purposes as our clot is slowly growing.

Let us have the following equation

$$\Delta u = f \quad \text{in } \Omega_t \tag{4.13}$$

equipped with the Robin boundary condition

$$\nabla u \cdot \mathbf{n} = k(u - g) \quad \text{on } \Gamma_t, \tag{4.14}$$

where g is a scalar function defined on the interface Γ_t, i.e. $g\colon \Gamma_t \to \mathbb{R}$, and k is a scalar constant, i.e. $k \in \mathbb{R}$. Li et al. in [22] developed the following approximation for this problem using the phase field method (for details see [22] Section 2.5, Approximation 2):

$$\nabla \cdot (\phi \nabla u) + \frac{1}{\varepsilon} B(\phi) k(u - g) = \phi f \quad \text{in } \Omega^*, \tag{4.15}$$

where $B(\phi) = \phi^2(1-\phi)^2$. The variable ϕ is interrelated with our phase-field variable c from above using the following formula:

$$\phi(c) = \frac{c+1}{2} \begin{cases} 1, & c = 1, \\ 0, & c = -1. \end{cases} \tag{4.16}$$

In [22] the behaviour of the approximation (4.15) is analyzed in detail. In the work [22] the possibility of application of this treatment of boundary conditions to other types of equations, i.e., not only to elliptic but also to parabolic equations, is introduced.

We will reformulate the transport equation in the following manner:

$$\frac{\partial(\phi w)}{\partial t} - D\nabla \cdot (\phi \nabla w) + \nabla \cdot (\phi w \mathbf{u}) + \frac{1}{\varepsilon} B(\phi) k w = 0 \quad \text{in } \Omega^*. \tag{4.17}$$

The equation is derived in the Appendix of [22]. To sum up, the equation (4.17) is obtained by formal multiplication of (3.2) by a test function, integrating in time and space (over the current domain of blood Ω_t) and introduction of the characteristic function of Ω_t. The characteristic function of Ω_t is then approximated by ϕ.

4.3. The whole system of equations. Our system contains the equations of conservation of linear momentum and mass for the fluid

$$\begin{gathered} \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\nabla p + \nabla \cdot (\nu \mathbf{D}(\mathbf{u})) + \mathbf{f} \quad \text{in } \Omega^*, \\ \nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega^*, \end{gathered} \tag{4.18}$$

and it contains the system of two equations originating from the Cahn-Hilliard of equation, derived in Section 4.1, which tracks the development of the fluid-solid interface (clot)

$$\frac{\partial c}{\partial t} - \nabla \cdot M \nabla \mu = \alpha D^{-1} k w |\nabla \varphi_\varepsilon| \quad \text{in } \Omega^*, \tag{4.19}$$
$$\mu - \frac{1}{\varepsilon^2} W'(c) + \Delta c = 0 \quad \text{in } \Omega^*$$

and it contains the transport equation for the platelets

$$\frac{\partial(\phi w)}{\partial t} - D\nabla \cdot (\phi \nabla w) + \nabla \cdot (\phi w \mathbf{u}) + \frac{1}{\varepsilon} B(\phi) k w = 0 \quad \text{in } \Omega^*, \tag{4.20}$$

where ϕ is related to the concentration c using (4.16).

To make the equation (4.20) well defined in the whole domain, we need to assign some values to the platelet field in the area of the solid part. Therefore, we add to the last equation the additional term $-(1-\phi)\Delta w$, i.e., we extend harmonically the platelet field in the area of the growing thrombus. The equation (4.20) will then have the following form:

$$\frac{\partial w}{\partial t} - D\nabla \cdot (\phi \nabla w) + \nabla \cdot (\phi w \mathbf{u}) + \frac{1}{\varepsilon} B(\phi) k w - (1-\phi)\Delta w = 0 \quad \text{in } \Omega^*. \tag{4.21}$$

5. Numerical treatment of the rephrased Weller's model

5.1. A splitting method for the linear momentum equations and continuum equations. The incompressible Navier-Stokes equations are a saddle point problem. After discretization by finite element method, the corresponding matrix to be solved by linear solvers is indefinite.

There are some methods which can be used to tackle this problem. We decided to solve this issue by using a projection method.

The system is split into convection-diffusion equations for the velocity and the Poisson problem for the pressure. There are several ways to split the Navier-Stokes equations; here we have decided to use the incremental pressure correction scheme (IPCS) [16]. Let us define operators $L(\mathbf{u})$ and $N(\mathbf{u})$ as

$$L(\mathbf{u}) = 2\,\mathrm{div}(\nu \mathbf{D}(\mathbf{u})) = \mathrm{div}(\nu(\nabla \mathbf{u} + (\nabla \mathbf{u})^\top))$$

and

$$N(\mathbf{u}) = \mathbf{u} \cdot \nabla \mathbf{u}.$$

The IPCS scheme can be described as follows:

Having the velocity $\mathbf{u}^k$ and pressure p^k from the previous time step, we solve first for the so called *tentative velocity* $\mathbf{u}_*^{k+1}$:

$$\frac{\mathbf{u}_*^{k+1} - \mathbf{u}^k}{\delta t} + N(\mathbf{u}_*^{k+1}) + \nabla p^k - L(\mathbf{u}_*^{k+1}) = \mathbf{0}, \tag{5.1}$$

where δt is the timestep size. Afterwards we solve for the new pressure p^{k+1}:

$$\Delta(p^{k+1} - p^k) = \frac{1}{\delta t} \operatorname{div} \mathbf{u}_*^{k+1}. \tag{5.2}$$

Then we perform the correction of the velocity $\mathbf{u}_*^{k+1}$:

$$\mathbf{u}^{k+1} = \mathbf{u}_*^{k+1} - \delta t(\nabla p^{k+1} - \nabla p^k). \tag{5.3}$$

The currently introduced splitting scheme must be adapted for the system which we wish to solve. We want to enforce low values of velocity in the area of the thrombus. For that purpose we enlarge viscosity in the area of the thrombus and we add a Brinkman-like term in the area of thrombus. We set the velocity field to zero in the velocity correction step as well.

Hence, we solve the following form of the transport equation for the velocity field:

$$\frac{\mathbf{u}_*^{k+1} - \mathbf{u}^k}{\delta t} + N(\mathbf{u}_*^{k+1}) + \nabla p^k - L_\phi(\mathbf{u}_*^{k+1}) + (1-\phi)\mathbf{u}_*^{k+1} = 0, \tag{5.4}$$

where

$$L_\phi(\mathbf{u}) = \operatorname{div}(2\nu(\phi)\mathbf{D}(\mathbf{u})) = \operatorname{div}(\nu(\phi)(\nabla\mathbf{u} + (\nabla\mathbf{u})^T)), \tag{5.5}$$

i.e., the viscosity is dependent on the value of ϕ defined by (4.16).

As was said above, we set the velocity field in the thrombus area to zero value in the following manner:

$$\mathbf{u}^{k+1} = \phi(\mathbf{u}_*^{k+1} - \delta t(\nabla p^{k+1} - \nabla p^k)), \tag{5.6}$$

which could be seen as a projection of the velocity on the space of velocity with trivial values in the area of the thrombus.

The nonlinear term, i.e., $N(\mathbf{u})$ needs special treatment. There is a possibility to use an extrapolation from the previous time step for linearization; however we have decided to linearize the convective term using the Picard iteration. We use the norm of the residual of the equation (5.1) as the termination criterion of the Picard iteration.

5.2. An heuristic adaptive time stepping method. Suppose that we are about to solve dynamic equations, such as Navier-Stokes equations or transport equations of some species. We will now assume, for instance, that we have a transport equation for the quantity u.

We refer to the work [30] and [21] for further details on the shown procedure.

In the following, u denotes the unknown precise solution of the equations, whereas $u_{\Delta t}$ denotes an approximate solution obtained using time step size Δt. We want to reach some prescribed tolerance TOL in the solution process:

$$\|u - u_{\Delta t}\| \approx \text{TOL}. \tag{5.7}$$

For that purpose we perform an expansion to obtain the local truncation error after one step Δt and one step of length $m\Delta t$, where m is an appropriately chosen integer number.

We will assume that the error of the solution $e(u)$ is independent of the time step size. Then the expansions of the error can be written as:

$$u_{\Delta t} = u + (\Delta t)^2 e(u) + \mathcal{O}((\Delta t)^4) \tag{5.8}$$

and

$$u_{m\Delta t} = u + m^2(\Delta t)^2 e(u) + \mathcal{O}((\Delta t)^4). \tag{5.9}$$

From the previous two expressions we derive the equation for the error:

$$e(u) \approx \frac{u_{m\Delta t} - u_{\Delta t}}{(\Delta t)^2(m^2 - 1)}. \tag{5.10}$$

Suppose now that we want to get an estimation of the error after one properly chosen timestep Δt_*.

We have

$$u_{\Delta t_*} = u + (\Delta t_*)^2 e(u) + \mathcal{O}((\Delta t_*)^4). \tag{5.11}$$

We obtain from this equation, on rearrangement and substitution from (5.10), the following approximate identity:

$$\|u - u_{\Delta t_*}\| \approx \Big(\frac{\Delta t_*}{\Delta t}\Big)^2 \frac{\|u_{\Delta t} - u_{m\Delta t}\|}{m^2 - 1} = \text{TOL}. \tag{5.12}$$

The estimator of the adaptive time step is then

$$(\Delta t_*)^2 = \text{TOL}\, \frac{(\Delta t)^2(m^2 - 1)}{\|u_{\Delta t} - u_{m\Delta t}\|}. \tag{5.13}$$

We sum up the usage of the estimator in the algorithm scheme 1.

Algorithm 1: Algorithm for one adaptive time step.

Algorithm: Adaptive time step
Data: u^n
Result: u^{n+1}
Given the old solution u^n do:
begin

1. Make m small timesteps of size Δt to compute $u_{\Delta t}$
2. Make one large step of size of size $m\Delta t$ to compute $u_{m\Delta t}$
3. Evaluate the relative solution changes $\|u_{\Delta t} - u_{m\Delta t}\|$
4. Calculate the 'optimal' value Δt_* using (5.13) for the next time step
5. If $\Delta t_* \ll \Delta t$, reset the solution and go back to step 1, using Δt_* as new timestep
6. Set $u^{n+1} = u_{\Delta t}$

end

R e m a r k 5.1. We don't set the newly obtained timestep to Δt_*, as we want to avoid instability problems. Instead if

▷ $\Delta t_* \ll \Delta t$, then we set $\Delta t_{\text{new}} = \Delta t/k$,
▷ $\Delta t_* \gg \Delta t$, then we set $\Delta t_{\text{new}} = \Delta t * k$,

where k is some appropriately chosen integer number.

5.3. Resolution of different time scales and decoupling of the solved equations. We model processes with different time scales. Hence we should expect that the timesteps obtained using the previous adaptive timestepping algorithm will be different for each process.

After computation of each time iteration using the previous algorithm, we take the minimum of all proposed timesteps from (5.13), i.e.

$$\Delta t_{*\text{FUTURE}} = \min\{\Delta t_{*\text{NAVIERSTOKES}}, \Delta t_{*\text{PHASE}}, \Delta t_{*\text{TRANSPORT}}\}. \tag{5.14}$$

The equations for modelled processes are solved as decoupled. However, actually they are coupled.

We will reintroduce the coupling into our system of equations by subiteration. The ending criterion of the subiteration is based on the size of difference of solutions originating from last two subiterations.

5.4. Weak formulation.

5.4.1. Weak formulation of the equations for the split momentum equations and continuity equation. Assuming that all data are sufficiently regular, we obtain the weak formulations by formally multiplying the equations by a vector test function $\mathbf{v}$ or scalar function q from appropriated spaces defined later.

We will define the following notation for the inner products of tensor, vector and scalar functions:

$$(\mathbf{A},\mathbf{B}) = \int_{\Omega^*} \mathbf{A} : \mathbf{B}\,\mathrm{d}x, \quad (\mathbf{u},\mathbf{v}) = \int_{\Omega^*} \mathbf{u}\cdot\mathbf{v}\,\mathrm{d}x, \quad (p,q) = \int_{\Omega^*} pq\,\mathrm{d}x. \tag{5.15}$$

We multiply (5.4) by the vector function $\mathbf{v}$ obtaining

$$\begin{aligned}\Big(\frac{\mathbf{u}_*^{k+1}-\mathbf{u}^k}{\delta t} + \mathbf{u}_{\mathrm{nonlin}}\cdot\nabla\mathbf{u}_*^{k+1} + (1-\phi)\mathbf{u}_*^{k+1},\mathbf{v}\Big)\\ + (\nabla p^k,\mathbf{v}) - (\operatorname{div} 2\nu(\phi)\mathbf{D}(\mathbf{u}_*^{k+1}),\mathbf{v}) = \mathbf{0},\end{aligned} \tag{5.16}$$

where $\mathbf{u}_{\mathrm{nonlin}}$ is the velocity solution from the previous Picard iteration.

In order to redistribute derivations evenly between the solution and test functions, we formally perform integration by parts on the diffusion and pressure terms. This leads to

$$\begin{aligned}\Big(\frac{\mathbf{u}_*^{k+1}-\mathbf{u}^k}{\delta t} + \mathbf{u}_{\mathrm{nonlin}}\cdot\nabla\mathbf{u}_*^{k+1} + (1-\phi)\mathbf{u}_*^{k+1},\mathbf{v}\Big) - (p^k,\operatorname{div}\mathbf{v}) + (p^k\mathbf{n},\mathbf{v})_{\partial\Gamma}\\ + (2\nu(\phi)\mathbf{D}(\mathbf{u}_*^{k+1}),\nabla\mathbf{v}) - (2\nu(\phi)\mathbf{D}(\mathbf{u}_*^{k+1})\mathbf{n},\mathbf{v})_{\partial\Gamma} = \mathbf{0},\end{aligned} \tag{5.17}$$

where $(\cdot,\cdot)_\Gamma$ stands for integral over boundary Γ, defined simarly to (5.15). We have $\mathbf{v}$ equal to zero on Γ_{in} and Γ_{wall}, therefore, the boundary integrals on these parts of boundary vanish. We have to handle the boundary terms on Γ_{ouflow}.

We set on outflow

$$2\nu(\phi)\mathbf{D}(\mathbf{u}_*^{k+1})\mathbf{n} = \mathbf{0}, \tag{5.18}$$

hence, in the equation (5.17) only the boundary integral with pressure corresponding to the outflow part of boundary Γ_{ouflow} remains.

We will now turn our attention to the Poisson problem for pressure (5.2). We multiply the equation by a test function q and integrate formally by parts, getting

$$-(\nabla(p^{k+1}-p^k),\nabla q) + (\nabla(p^{k+1}-p^k)\cdot\mathbf{n},q)_{\partial\Omega} = \frac{1}{\delta t}(\operatorname{div}\mathbf{u}_*^{k+1},q). \tag{5.19}$$

In this case we have to deal with the implicit use of a boundary condition. The application of the homogeneous Neumann condition $\partial p/\partial\mathbf{n} = 0$ leads to a singular

matrix to be solved by linear solver. One of workarounds to tackle this issue is to prescribe an additional boundary condition on pressure $p = 0$ on Γ_{out}.

We define function spaces for tentative velocity and pressure in the following manner: Test space for tentative velocity:

$$\mathbf{V}' = \mathbf{H}_0^1(\Omega^*) = \{\mathbf{v} \in [H^1(\Omega^*)]^3,\ \mathbf{v}|_{\Gamma_{\text{in}} \cup \Gamma_{\text{wall}}} = \mathbf{0}\}. \tag{5.20}$$

Trial space for tentative velocity:

$$\mathbf{V} = \mathbf{H}_D^1(\Omega^*) = \{\mathbf{v} \in [H^1(\Omega^*)]^3,\ \mathbf{v}|_{\Gamma_{\text{in}}} = \mathbf{u}_D,\ \mathbf{v}|_{\Gamma_{\text{wall}}} = \mathbf{0}\}. \tag{5.21}$$

The test and trial space for the weak formulation (5.19) of Poisson problem (5.2) are identical:

$$Q' = Q = \{q \in H^1(\Omega),\ q|_{\Gamma_{\text{out}}} = 0\}. \tag{5.22}$$

In the case of the velocity correction we perform only the L_2-projection and it is not necessary to discuss the boundary conditions for this step.

5.4.2. Weak formulation of the phase-field and chemical potential equation. We multiply (4.19) by a tuple (ψ_c, ψ_μ) from the appropriate test spaces to be specified later:

$$\begin{aligned} \Big(\frac{c^{k+1} - c^k}{\delta t}, \psi_c\Big) - (\nabla \cdot M \nabla \mu^{k+1}, \psi_c) &= (\alpha D^{-1} k w |\nabla \varphi_\varepsilon|, \psi_c), \\ \Big(\mu^{k+1} - \frac{1}{\varepsilon^2} W'(c^{k+1}) + \Delta c^{k+1}, \psi_\mu\Big) &= 0. \end{aligned} \tag{5.23}$$

We impose the following boundary conditions on the system:

$$M \nabla \mu^{k+1} = 0 \quad \text{on } \Gamma \tag{5.24}$$

and

$$\nabla c^{k+1} \cdot \mathbf{n} = 0 \quad \text{on } \Gamma_{\text{wall}} \cup \Gamma_{\text{out}}, \tag{5.25}$$

where Γ is the boundary of the whole domain. In our case it is the union of the inflow boundary, the outflow boundary and walls. This leads to zero boundary terms after performing of formal integration of parts of (5.23):

$$\begin{aligned} \Big(\frac{c^{k+1} - c^k}{\delta t}, \psi_c\Big) + (M \nabla \mu^{k+1}, \nabla \psi_c) &= (\alpha D^{-1} k w |\nabla \varphi_\varepsilon|, \psi_c), \\ \Big(\mu^{k+1} - \frac{1}{\varepsilon^2} W'(c^{k+1}), \psi_\mu\Big) - (\nabla c^{k+1}, \nabla \psi_\mu) &= 0. \end{aligned} \tag{5.26}$$

Test space for the phase field c:

$$\Psi_c' = \{c \in H^1(\Omega^*), c|_{\Gamma_{\text{in}}} = 0\}. \tag{5.27}$$

Trial space for the phase field c:

$$\Psi_c = \{c \in H^1(\Omega^*), c|_{\Gamma_{\text{in}}} = c_D\}. \tag{5.28}$$

Test and trial space for the chemical potential

$$\Psi_\mu' = \Psi_\mu = H^1(\Omega^*). \tag{5.29}$$

5.4.3. Weak formulation of the transport equation for platelets. We multiply (4.21) by the test function Ψ_w from the appropriate space to be specified later:

$$\begin{aligned} &\Big(\frac{w^{k+1} - w^k}{\delta t} - D\text{div}(\phi\nabla w^{k+1}) + \text{div}(\phi w^{k+1}\mathbf{u}), \Psi_w\Big) \\ &\quad + \Big(\frac{1}{\varepsilon}B(\phi)kw^{k+1} - (1-\phi)\Delta w^{k+1}, \Psi_w\Big) = 0 \quad \text{in } \Omega^*. \end{aligned} \tag{5.30}$$

After integration by parts and application of the no-flux boundary condition, we obtain the following weak formulation:

$$\begin{aligned} &\Big(\frac{w^{k+1} - w^k}{\delta t} + \text{div}(\phi w^{k+1}\mathbf{u}) + \frac{1}{\varepsilon}B(\phi)kw^{k+1}, \Psi_w\Big) + ((1-\phi)\nabla w^{k+1}, \nabla\Psi_w) \\ &\quad - ((1-\phi)\nabla w^{k+1}\cdot\mathbf{n}, \Psi_w)_{\Gamma_{\text{outflow}}} + D(\phi\nabla w^{k+1}, \nabla\Psi_w) \\ &\quad - D(\phi\nabla w^{k+1}\cdot\mathbf{n}, \Psi_w)_{\Gamma_{\text{outflow}}} = 0 \quad \text{in } \Omega^*. \end{aligned} \tag{5.31}$$

Test space for the platelets field w:

$$\Psi_w' = \{w \in H^1(\Omega^*),\ w|_{\Gamma_{\text{in}}} = 0\}. \tag{5.32}$$

Trial space for the platelets field w:

$$\Psi_w = \{w \in H^1(\Omega^*),\ w|_{\Gamma_{\text{in}}} = w_D\}. \tag{5.33}$$

6. Finite element library deal.ii

6.1. Introduction of the library. The finite element library deal.ii [3], [7] is an open source library developed in C++. It uses features of C++ to allow mathematicians rapid finite element code development. It features wrapping of general parts of finite element codes such as grid creation and refinement, processing of degrees of freedom, output of results into different graphics formats, and output of meshes. For that purpose, deal.ii leverages the object orientation of C++, which allows us to write highly modular code.

Another feature of C++ used in deal.ii library design is the support of generic programming, i.e. template (meta)programming. One great example of using templates in deal.ii is so-called dimension independent programming. It allows us to change the spatial dimension of the solved problem from one to two or three spatial dimensions without principally great effort.

The deal.ii library supports different kinds of refinement—h-, p-, hp-. The refinement criterion is based on the local error indicator or error estimator. In deal.ii, refinement functions are implemented which take the indicator or estimator object as a function parameter. The function then refines, either in h-, p-, or hp- manner, based on the current solution and the object of indicator or estimator.

The deal.ii library allows to use the computational sources of multiple cores on one node or use multiple nodes in a network. The first case is implemented using the library Threading Building Blocks, [25], i.e., it uses a threading technology. On the other hand the communication among nodes in a network is implemented using the communication framework of MPI.

In deal.ii the backend for linear algebra operations is implemented. It contains support for sparse matrices, vectors, Krylov subspace methods, direct solvers, and blocked matrices. Deal.ii provides a rich interface with well established numerical packages Petsc [6] and Trilinos [17] as well, which are to be used mainly in the cases of computations distributed across multiple nodes in the cluster. The deal.ii library needs to solve the problem of partitioning of complicated large computational domains into small chunks, in the case of parallel or distributed computations. For that purpose, it uses the libraries METIS [19] and p4est [10].

6.2. Used features of the library. We ran our computations in the distributed setting, i.e. we computed on multiple cores of several nodes of a cluster. Hence, the communication between cores and nodes happens using MPI library interprocess calls. We generated grids for computations using the functions provided by the deal.ii library, i.e., we generated them at the beginning of our computations. In the case of the perfusion chamber geometry, we refined the grid at the bottom of the

computational area in order to be able to capture more precisely the development of the interface between the clot and blood flow. For the solution of our problems, we used the deal.ii interface to the well established bunch of libraries Trilinos [17]. This enables us to switch the linear solvers easily without profound changes in our code, as the interface is quite uniform for all linear solvers of Trilinos.

7. Results

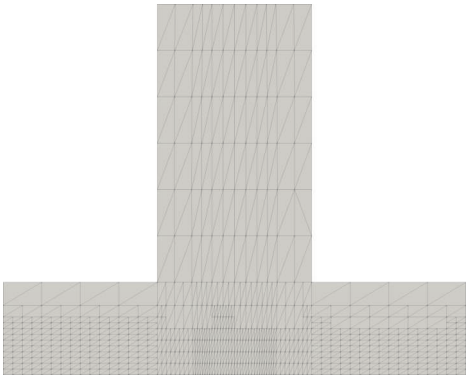
We have chosen two computational geometries for simulations. In Section 7.1, we outline the results achieved on the perfusion chamber geometry, whereas in Section 7.2 we show results computed on the cylinder geometry.

7.1. Perfusion chamber geometry. In our first simulation, we have chosen one of the configurations used in [31], [32] and [33]. This configuration originates from the experimental setting of Affeld [2]. In this experiment, platelet-rich plasma with pre-activated platelets flows from the inflow of the perfusion chamber onto a glass plane. The glass plane is orthogonal to the flow direction at the inflow. The key result of the experiment is that the deposition of platelets is minute at the point of stagnation on the glass plane, whereas the site with greatest concentration of deposited platelets proves to be at the place of elevated shear rate. This location appeared to be downstream of the axis of symmetry of the perfusion chamber, i.e. considerably far away from the stagnation point.

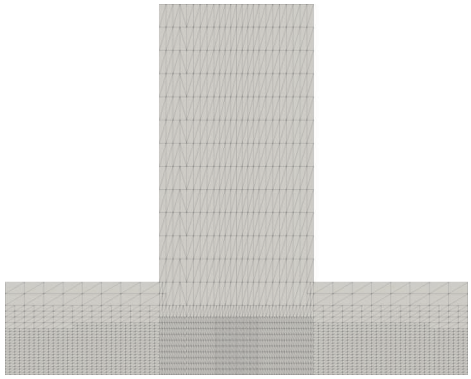
We solved our model on two differently refined meshes. The coarser one has 26422, whereas the finer mesh has 197376 cells. One can see the computational meshes of both refinements in Fig. 3. Initial values of the phase field are rendered in the 2D cut of the computational domain for the coarse mesh in Fig. 4(a) and for the fine mesh in Fig. 4(b).

We have performed computations on both meshes and we achieved similar results on both levels of refinement. In Figures 4(c) and 5(a) it could be seen that the clot has grown in the area where the the highest shear stress occurs. The same behavior can be observed for the fine mesh, see the Figures 4(d) and 5(b). In the pictures of computations on the finer mesh it is apparent that the shape of emerging clot is smoother, which corresponds to the better resolution of phase field due smaller mesh cells.

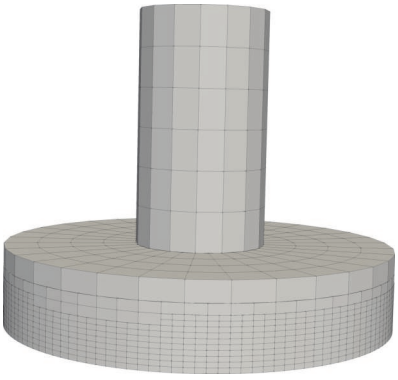
We have mentioned that the clot grows mostly in the area of high shear stress. This statement can be demonstrated by Fig. 5(c) for the coarse mesh and Fig. 5(d) for the fine mesh. In the figures there are rendered values of the velocity field, both as colour on the slice and arrows on the slice. However, the most interesting for us is the 3D contour of phase field of zero value. We have coloured the contour by values



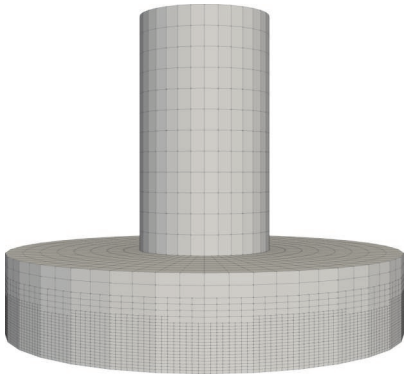
(a) 2D cut of the computational mesh—coarse mesh.



(b) 2D cut of the computational mesh—fine mesh.



(c) 3D view—coarse mesh.

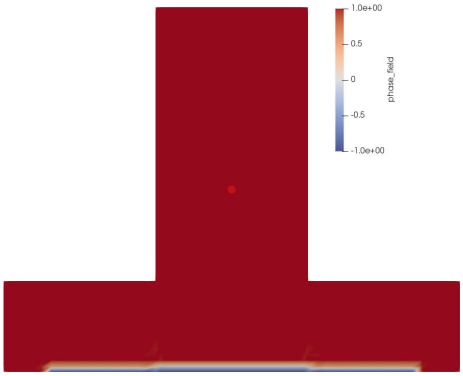


(d) 3D view—fine mesh.

Figure 3. Overview of computational meshes—perfusion chamber.

of the shear stress field. It is therefore apparent that the clot grows preferably in the area of high shear stress. This is in accordance with the assumptions of high shear stress thrombosis mentioned in Section 1.

We have reached similar location of growth of the clot as in [33]. We must however admit that our results are considerably not so fine as in [33]. Namely, the interface between flow and the clot is rather blurred. This originates from the very basic assumption of the phase-field—the interface is represented by a no-sharp function. The other reason for the not so sharp location of the interface is the fact, that we computed the problem in fully three dimensional setting. In [31], [32], and [33], the cylindrical symmetry of the problem was used to allow for finer spatial resolution. Saying this, we aspire however to be able to compute similar problems with similar granularity of computational mesh as was presented [31], [32], and [33].



(a) Initial phase field setting—coarse mesh.

(b) Initial phase field setting—fine mesh.

(c) Final state of the phase field—coarse mesh.

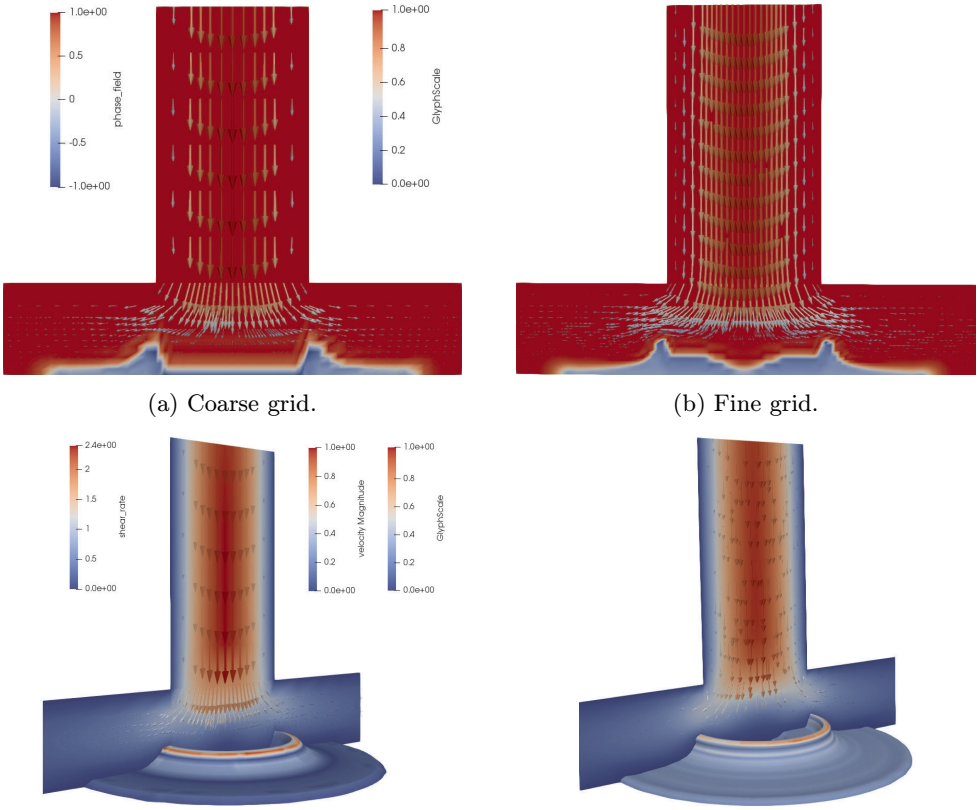
(d) Final state of the phase field—fine mesh.

Figure 4. Initial and final state of phase field c for coarse mesh ((a) and (c)) and fine mesh ((b) and (d)).

In Sections 8.2 and 8.3 on scaling of our fully three dimensional implementation of the model, we will discuss the possibility of scaling up of our code to a larger number of cells.

7.2. Cylinder geometry. In the second simulation, we solved our system in the cylinder geometry, where the reactive zone is located in the middle in a symmetric position at the wall. We let blood flow from right to left. As the platelets gradually arrive at the reactive zone, the process of coagulation is triggered off. The use case for this geometry could be a stented artery, as the artificial surface of the stent is reactive.

We solved our model on two differently refined meshes. The coarser one has 5120 cells, whereas the finer mesh has 40960 cells. You can see the computational meshes



(a) Coarse grid.

(b) Fine grid.

(c) Coarse meesh.

(d) Fine mesh

Figure 5. 2D view of velocity $\mathbf{u}$ and phase field c ((a) and (b)), 3D view of the velocity field u and the zero value phase field c isocontour colored by shear stress s ((c) and (d)).

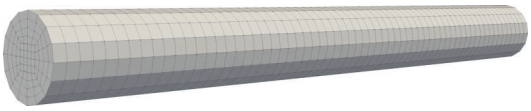
for the coarse grid and the fine grid in Fig. 6. Initial values of the phase field are shown in Fig. 7(a) for the coarse mesh and in Fig. 7(b) for the fine mesh. We have performed computations on both meshes and we have achieved corresponding results on both levels of refinement.

In Fig. 7(c) and 8(b) it could be again observed that the clot has grown the most in the area which experiences the highest shear stress. The same behaviour can be observed for the fine mesh; see Fig. 7(d) and 8(b). In the pictures of computations on the finer mesh it is apparent that the shape of emerging clot is smoother, which corresponds to the better resolution of phase field due smaller mesh cells.

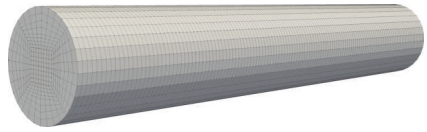
The Figures 8(c) for coarse mesh and 8(d) for the fine mesh demonstrate that the clot grows mostly in the area of high shear stress. In the picture, values of the velocity field are rendered both as colour on the slice and arrows on the slice.

(a) 2D cut of the computational mesh—coarse mesh.

(b) 2D cut of the computational mesh—fine mesh.

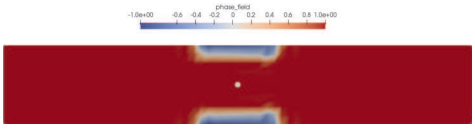


(c) 3D view—coarse mesh.



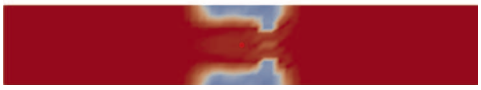
(d) 3D view—fine mesh.

Figure 6. Overview of computational meshes—cylinder.

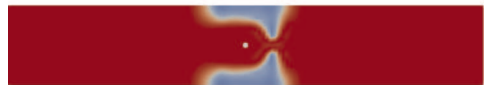


(a) Initial phase field setting—coarse mesh.

(b) Initial phase field setting-fine mesh.

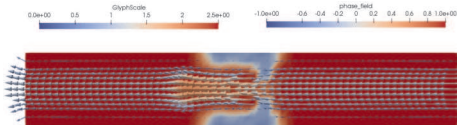


(c) Final state of the phase field—coarse mesh.

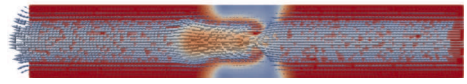


(d) Final state of the phase field—fine mesh.

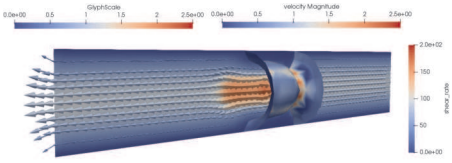
Figure 7. Initial ((a) and (b)) and final state ((c) and (d)) of phase field c.



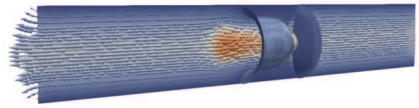
(a) 2D view of velocity field as glyphs and phase field values—coarse grid.



(b) 2D view of velocity field as glyphs and phase field values—fine grid.



(c) 3D view of the zero value phase field isocontour colored by shear stress and the velocity field-coarse mesh.

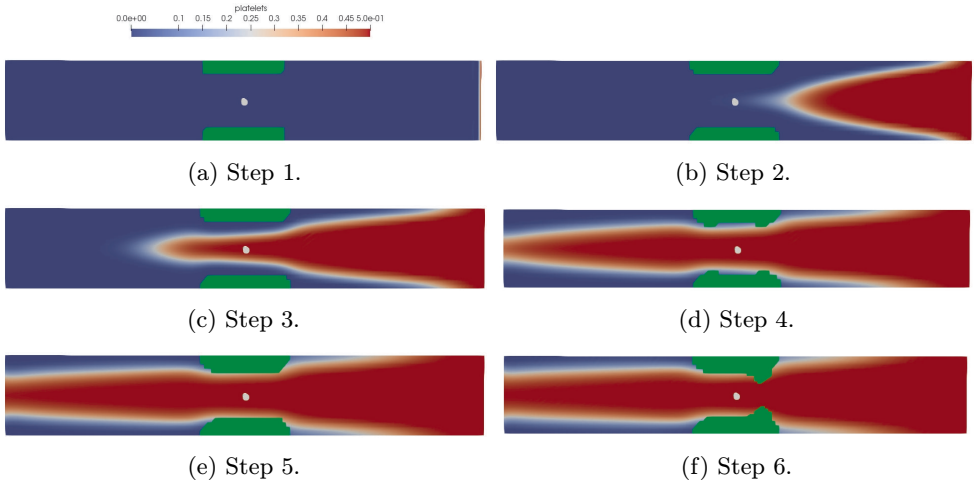


(d) 3D view of the zero value phase field isocontour colored by shear stress and the velocity field—fine mesh.

Figure 8. 2D view of the phase field c and of the velocity field $\mathbf{u}$ ((a) and (b)), 3D view of the velocity field $\mathbf{u}$ and the zero value phase field isocontour colored by shear stress s ((c) and (d)).

Here we can also observe that narrowing of the lumen of the vessel by the emerging clot causes an increase of the velocity in the narrowed part of vessel. This leads to an increase of shear stress at the surface of the clot, which enhances the growth of the clot.

We would like also comment on the dependence of thrombus growth not only on shear stress, but also on the supply of blood platelets. Blood platelets, thrombocytes, are an integral part of the high shear rate thrombus. We take into account the dependence of growth rate on shear rate and thrombocytes inflow by the Robin boundary equation for the platelet field (3.3). The boundary condition depends linearly on the concentration of platelets and the adhesion rate, which grows linearly with shear stress. Therefore, we should expect that the growth of the thrombus starts only after the arrival of thrombocytes on the location of thrombus interface. We can witness this development in the series of steps 1–6 in Fig. 9.



(a) Step 1. (b) Step 2.

(c) Step 3. (d) Step 4.

(e) Step 5. (f) Step 6.

Figure 9. Development of the clot (green colour) and platelet field w from the initial state (Step 1) to the final state (Step 6).

We would like to add a note about the results of simulations in Fig. 9. It could seem that the development of the platelet field does not couple correctly with the development of the phase field (green colour in Fig. 9), i.e. the simulation results in nonzero values for platelet field in the area of the developing thrombus. We must however remind the reader that we have used harmonical extension of the platelet field in the area of thrombus; see the equation (4.21). Therefore, it could be said that in the area of the thrombus, the values of the platelet field are physically irrelevant for the model, as its values there are only supplied for the better solvability of our

problem. Hence, we do not assign any physical interpretation to the platelet field in the area of the developing thrombus.

8. Properties of numerical recoupling scheme, scaling of our implementation

In the following two sections, we want to show numerical and scaling properties of our implementation of the model. In Section 8.1 we study the numerical properties of our scheme from the point of view of recoupling. In Section 8.2, we study the efficiency of different combinations of linear solvers and preconditioners. Section 8.3 shows the weak and strong scaling properties of our implementation.

8.1. Recoupling of the decoupled equations. We have mentioned in Section 5.3 that recoupling iterations are needed for achievement of proper results. The norm of the difference of subsequent solutions played the role of the terminating criterion of recoupling subiterations.

We could naturally expect that if we decrease the initial timestep, then the norm of the difference of the solutions from different recoupling iterations would decrease faster. We performed a simple numerical experiment which confirms this expectation. We chose the cylindrical geometry with 5120 cells as our computational domain. The number of recoupling iterations in each timestep n was now fixed. We performed computations using our adaptive timestepping method until we arrived at a fixed time T. We denote the solution obtained using n recoupling iterations as $\mathbf{u}_n$ for velocity and w_n for platelets. We store the solution for $n = 4$, i.e. the solution at time T obtained using 4 recoupling iterations, to hard disk in order to be able to compare it with solutions for $n = 1, 2, 3$. The results of the computations are given in the Tables 2, 3, 4 and 5. From the tables it can be seen that the convergence to the finest solution is quicker for the smaller initial timestep 0.05 and that the convergence slows down for the larger timesteps 0.1 and 0.2 (see Table 3 and 4). In the case of convergence in Table 5 we can see that it is by several orders slower than in Table 2 for the platelet field. The differences in the velocity field in Table 5 decrease at the same pace as in 2, but the order of magnitude of the differences is approximately two orders lower. It is however questionable whether the solution with $n = 4$ for timestep 0.5, see Table 5, is a good candidate for comparing convergence of the solution $n = 1, 2, 3$, because it appears that the convergence in the platelet field w is very slow.

No. of couplings n	velocity $\|\mathbf{u}_4 - \mathbf{u}_n\|$	platelets $\|w_4 - w_n\|$
1	0.000108446	3.59149e–08
2	3.59149e–08	2.1103e–10
3	4.84701e–13	1.0633e–10

Table 2. Convergence table of the velocity and platelet solution for timestep Δt=0.05.

No. of couplings n	velocity $\|\mathbf{u}_4 - \mathbf{u}_n\|$	platelets $\|w_4 - w_n\|$
1	0.0223035	1.11685e–4
2	1.72131e–08	1.72131e–08
3	1.28085e–13	4.1479e–13

Table 3. Convergence table of the velocity and platelet solution for timestep $\Delta t = 0.1$.

No. of couplings n	velocity $\|\mathbf{u}_4 - \mathbf{u}_n\|$	platelets $\|w_4 - w_n\|$
1	0.236748	0.0989294
2	2.08167e–08	2.04242e–10
3	2.3034e–13	6.72133e–14

Table 4. Convergence table of the velocity and platelet solution for timestep $\Delta t = 0.2$.

No. of couplings n	velocity $\|\mathbf{u}_4 - \mathbf{u}_n\|$	platelets $\|w_4 - w_n\|$
1	0.101598	0.210646
2	1.03488e–06	0.10869
3	5.86096e–11	0.0986667

Table 5. Convergence table of the velocity and platelet solution for timestep $\Delta t = 0.5$.

8.2. Scaling of linear solvers. We have experimented with different combinations of linear solvers and preconditioners. The most demanding processing of linear systems were performed for the solution of the tentative velocity step (5.4) and for the solution of the projection step (5.2). The solution of the projection step results in a linear system with symmetric positive definitive matrix. We choose a conjugate gradient solver with either ILU or algebraic multigrid as preconditioner. We tried to increase the number of cores which were used for the solution of the system originating from cylindrical geometry with 5120 cells.

We can observe in the Table 6 that the duration of solution of the projection step increases as we add more cores for the solution of the system. This could be attributed to the fact that the system is quite small and communication gradually prevails with increasing of number of cores.

Discretization of the tentative velocity step does not end with a symmetric matrix. Therefore we have to use a more general solver for the solution of the linear

Preconditioner	24 Cores	48 Cores	96 Cores	192 Cores
ILU	0.229	0.411	0.904	1.49
AMG	0.581	0.994	1.53	2.98

Table 6. Solution times in seconds for CG with different preconditioners for the projection step (5.2).

system. We chose GMRES solver equipped with ILU, SOR or algebraic multigrid as preconditioners.

For almost all preconditioners we can observe in Table 7 a decrease of time of solution with increasing number of cores. The linear system is much larger than the pressure system, hence the scaling up is working. We can however see that for the algebraic multigrid there is a plateau in solution time for 96 and 192 cores.

Preconditioner	24 Cores	48 Cores	96 Cores	192
SOR	67.4	45	24.1	12
ILU	55.7	32.2	17.7	9.7
AMG	81.7	61	43.7	43.8

Table 7. Solution times in seconds for GMRES with different preconditioners for the tentative velocity step (5.4).

8.3. Strong and weak scaling of our implementation. In order to investigate parallel efficiency of our implementation, we will perform two sets of experiments with different settings—strong scaling tests and weak scaling tests. The former ones are underpinned by Amdahl's law, whereas the latter ones are related to Gustafson's law, see [8].

8.3.1. Strong scaling of our implementation. In the case of strong scaling, the problem size remains fixed but the number of cores deployed for the solution of the problem is increased. We have performed a series of computations on the cylinder and on the chamber mesh for two different levels of refinement. In each run we were computing until we reached a fixed time limit, i.e. until a fixed physical time of simulation was reached. The number of cores was increased by a factor 2 from 24 cores to 768 cores. We have achieved a considerable level of scaling of our computations; however a plateau has reached and the duration of computations of the problem started to grow (see Table 8 and the chart in Figure 10). This could be attributed to the fact that the communication overhead might start to dominate. That was the case for both refinement levels of chamber grid and of the cylinder geometry.

	Chamber geometry		Cylinder geometry	
Number of Cores	15040 cells	75954 cells	5120 cells	40960 cells
24	3.430	71.000	1.560	
48	1.670	31.700	822	
96	1.020	16.400	456	10.700
192	742	8.820	304	4.710
384	506	4.070	156	459
768	424	7.710	258	2.460

Table 8. Strong scaling—time of solution for number of cores.

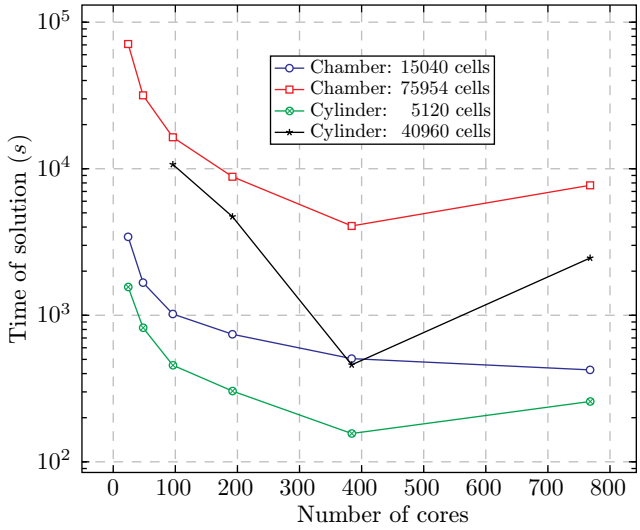


Figure 10. Strong scaling—time of solution for a number of cores.

8.3.2. Weak scaling of our implementation. In this case the workload (problem size) assigned to each core remains constant as additional cores are used to solve a larger problem.

In the case of weak scaling performance tests, we started with a low number of cells solved on certain number of cores. Then we increased the number of cells and the number of cores by a factor of 2 in each step in order to keep a fixed amount of work per core. If we had an ideal weak scaling property, we would witness that the function reflecting the dependence of solution time on the number of cores is constant. That expectation originates from the fact that the workload assigned to a core is the same for every number of deployed cores.

In the Table 9 and in the chart in the Figure 11 we can see that with increasing number of cells and cores we do not witness a constant level of solution time of our problem as the ideal scaling would dictate. As the solution time is almost

monotonously growing with number of cells/cores and jumps to one order higher value for the largest number of cells/processors, we suppose that the communication of our distributed code is becoming dominant. To improve this behaviour it would be necessary to refactor our current codebase.

No. of cells	No. of cores	time (s)
640	24	73.3
1280	48	104
2560	96	165
5120	192	204
10240	384	158
20480	768	1310

Table 9. Weak scaling—time of solution for number of cores.

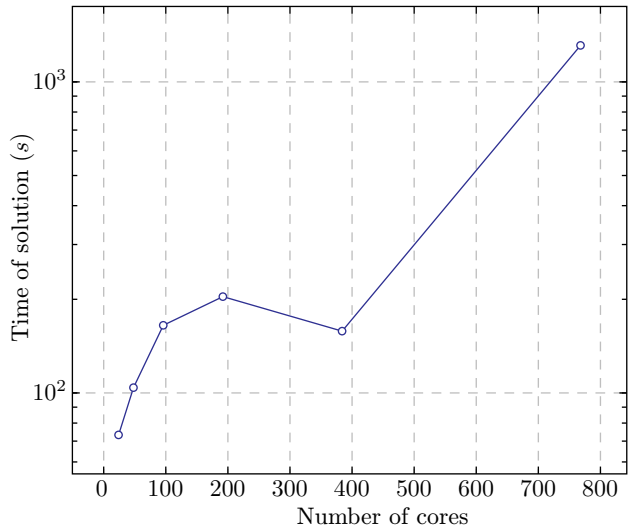


Figure 11. Weak scaling—time of solution for number of cores.

9. Conclusion

We have reviewed current developments in the area of high shear rate thrombosis research. It could be generally said that in the area of modelling of blood coagulation it is not possible to formulate an all-encompassing model. A model that would fit all situations easily becomes too complex to solve. Weller's model captures high shear rate thrombosis by the assumption that the adhesion rate of blood platelets

on the surface of the clot is proportional to the shear rate. The related biochemical processes, mentioned in Introduction, are incorporated in the shear stress dependent function, the adhesion rate (3.4), at the phenomenological level.

In Section Results we have shown that the clot structure is growing in the areas of clot with largest values of shear stress. This is in accordance both with the results of Weller in [31], [33], and [2].

We have commented on advantages of the phase-field formulation in Section 4. We are convinced that the tradeoff for the advantages (that is solution of a system of two equations, i.e. the Cahn-Hilliard system, instead of one transport equation for level-set function) is favourable. Namely, the phase-field is smeared implicitly by the Cahn-Hilliard system without the need for smearing of the sharp interface function, i.e. the level-set function.

We have performed a simple mesh convergence study both for the perfusion chamber geometry and the cylinder geometry. We have proven that the phase field in our simulations grows only when the blood platelets arrive, which agrees with our expectations based on the performed biochemical experiments or medical measurements.

In the part about scaling of our implementation we achieved a certain level of strong scaling. However, after further adding of computational power, i.e. cells, we arrived at a point where the performance deteriorates. Our study measures the times of computation of all parts of the solver. Hence, we do not know exactly what the bottleneck is, i.e. whether it is the assemble routine, the linear solver, on the communication overhead among components of our whole solver. Searching for the bottleneck will require both analysis of the source code and profiling of the application.

The proposed method could be applied not only to the initial stage of the development of the thrombus but also to the mature state of the thrombus. The modelling of this phenomenon is not only worth studying for academic reasons, but it could valuably contribute to the medical treatment of cardiac diseases.

References

[1] *P. A. Aarts, S. A. van den Broek, G. W. Prins, G. D. Kuiken, J. J. Sixma, R. M. Heethaar*: Blood platelets are concentrated near the wall and red blood cells, in the center in flowing blood. Arteriosclerosis *8* (1988), 819–824. doi

[2] *K. Affeld, A. J. Reininger, J. Gadischke, K. Grunert, S. Schmidt, F. Thiele*: Fluid mechanics of the stagnation point flow chamber and its platelet deposition. Artif. Organs. *19* (1995), 597–602. doi

[3] *G. Alzetta, D. Arndt, W. Bangerth, V. Boddu, B. Brands, D. Davydov, R. Gassmöller, T. Heister, L. Heltai, K. Kormann, M. Kronbichler, M. Maier, J.-P. Pelteret, B. Turcksin, D. Wells*: The deal.II library, version 9.0. J. Numer. Math. *26* (2018), 173–183. zbl MR doi

[4] *M. Anand, K. Rajagopal, K. R. Rajagopal*: A model incorporating some of the mechanical and biochemical factors underlying clot formation and dissolution in flowing blood. J. Theor. Med. *5* (2003), 183–218. zbl MR doi
[5] *D. Arndt, W. Bangerth, T. C. Clevenger, D. Davydov, M. Fehling, D. Garcia-Sanchez, G. Harper, T. Heister, L. Heltai, M. Kronbichler, R. M. Kynch, M. Maier, J.-P. Pelteret, B. Turcksin, D. Wells*: The deal.II library, version 9.1. J. Numer. Math. *27* (2019), 203–213. zbl MR doi
[6] *S Balay, W. D. Gropp, L. C. McInnes, B. F. Smith*: PETSc: Portable, Extensible Toolkit for Scientific Computation—Toolkit for Advanced Computation. Available at `http://www.mcs.anl.gov/petsc` (2018). sw
[7] *W. Bangerth, R. Hartmann, G. Kanschat*: deal.ii—a general-purpose object-oriented finite element library. ACM Trans. Math. Softw. *33* (2007), Article ID 24, 27 pages. zbl MR doi
[8] *G. Barlas*: Multicore and GPU Programming: An Integrated Approach. Morgan Kaufmann Publishers, San Francisco, 2015.
[9] *T. Bodnár, A. Fasano, A. Sequeira*: Mathematical models for blood coagulation. Fluid-Structure Interaction and Biomedical Applications (T. Bodnár et al., eds.). Advances in Mathematical Fluid Mechanics. Birkhäuser/Springer, Basel, 2014, pp. 483–569. zbl MR doi
[10] *C. Burstedde, L. C. Wilcox, O. Ghattas*: p4est: scalable algorithms for parallel adaptive mesh refinement on forests of octrees. SIAM J. Sci. Comput. *33* (2011), 1103–1133. zbl MR doi
[11] *L. D. C. Casa, D. H. Deaton, D. N. Ku*: Role of high shear rate in thrombosis. J. Vasc. Surg. *61* (2015), 1068–1080. doi
[12] *L. D. C. Casa, D N. Ku*: Thrombus formation at high shear rates. Ann. Rev. Biomed. Eng. *19* (2017), 415–433. doi
[13] *R. W. Colman, V. J. Marder, A. W. Clowes, J. N. George, S. Z. Goldhaber*: Hemostasis and Thrombosis: Basic Principles and Clinical Practice. Lippincott Williams & Wilkins, Philadelphia, 2005.
[14] *A. Fasano, A. Sequeira*: Hemorheology and hemodynamics. Hemomath. Modeling, Simulation and Applications 18. Springer, Cham, 2017, pp. 1–77. doi
[15] *S. Gross, A. Reusken*: Numerical Methods for Two-Phase Incompressible Flows. Springer Series in Computational Mathematics 40. Springer, Berlin, 2011. zbl MR doi
[16] *J. L. Guermond, P. Minev, J. Shen*: An overview of projection methods for incompressible flows. Comput. Methods Appl. Mech. Eng. *195* (2006), 6011–6045. zbl MR doi
[17] *M. A. Heroux, R. A. Bartlett, V. E. Howle, R. J. Hoekstra, J. J. Hu, T. G. Kolda, R. B. Lehoucq, K. R. Long, R. P. Pawlowski, E. T. Phipps, A. G. Salinger, H. Thornquist, R. S. Tuminaro, J. M. Willenbring, A. Williams, K. S. Stanley*: An overview of the Trilinos project. ACM Trans. Math. Softw. *31* (2005), 397–423. zbl MR doi
[18] *V. Huck et al.*: A2 - Research. Available at `http://www.shenc.de/A2-Huck-res.htm`.
[19] *G. Karypis, V. Kumar*: MeTis: Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 4.0. Available at `http://www.cs.umn.edu/~metis`, 2009. sw
[20] *N. S. Key, M. Makris, D. Lillicrap*, eds.: Practical Hemostasis and Thrombosis. John Wiley, Chichester, 2017. doi
[21] *D. Kuzmin*: Introduction to Computational Fluid Dynamics. Available at `http://www.mathematik.uni-dortmund.de/~kuzmin/cfdintro/lecture8.pdf`, 2010.
[22] *X. Li, J. Lowengrub, A. Rätz, A. Voigt*: Solving PDEs in complex geometries: a diffuse domain approach. Commun. Math. Sci. *7* (2009), 81–107. zbl MR doi
[23] *A. Mohan*: Modeling the Growth and Dissolution of Clots in Flowing Blood. PhD Thesis, Texas A&M University, College Station, 2005.
[24] *S. Osher, R. Fedkiw*: Level Set Methods and Dynamic Implicit Surfaces. Applied Mathematical Sciences 153. Springer, New York, 2003. zbl MR doi

[25] *J. Reinders*: Intel Threading Building Blocks: Outfitting C++ for Multi-Core Processor Parallelism. O'Reilly, Sebastopol, 2007.
[26] *K. S. Sakariassen, L. Orning, V. T. Turitto*: The impact of blood shear rate on arterial thrombus formation. Future Sci. OA *1* (2015), Article ID FSO30. doi
[27] *J. A. Sethian*: Level Set Methods and Fast Marching Methods. Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge Monographs on Applied and Computational Mathematics 3. Cambridge University Press, Cambridge, 1999. zbl MR
[28] *A. A. Tokarev, A. A. Butylin, F. I. Ataullakhanov*: Platelet adhesion from shear blood flow is controlled by near-wall rebounding collisions with erythrocytes. Biophys. J. *100* (2011), 799–808. doi
[29] *A. Tokarev, I. Sirakov, G. Panasenko, V. Volpert, E. Shnol, A. Butylin, F. Ataullakhanov*: Continuous mathematical model of platelet thrombus formation in blood flow. Russ. J. Numer. Anal. Math. Model. *27* (2012), 191–212. zbl MR doi
[30] *S. Turek*: Efficient Solvers for Incompressible Flow Problems. An Algorithmic and Computational Approach. Lecture Notes in Computational Science and Engineering 6. Springer, Berlin, 1999. zbl MR doi
[31] *F. Weller*: Modeling, Analysis, and Simulation of Thrombosis and Hemostasis: PhD Thesis. Ruprecht-Karls-Universität, Heidelberg, 2008. doi
[32] *F. F. Weller*: Platelet deposition in non-parallel flow: influence of shear stress and changes in surface reactivity. J. Math. Biol. *57* (2008), 333–359. zbl MR doi
[33] *F. F. Weller*: A free boundary problem modeling thrombus growth. Model development and numerical simulation using the level set method. J. Math. Biol. *61* (2010), 805–818. zbl MR doi
[34] *Z. Xu, N. Chen, S. C. Shadden, J. E. Marsden, M. M. Kamocka, E. D. Rosen, M. Alber*: Study of blood flow impact on growth of thrombi using a multiscale model. Soft Matter *5* (2009), 769–779. doi
[35] *K. E. Zlobina, G. Th. Guria*: Platelet activation risk index as a prognostic thrombosis indicator. Scientific Reports *6* (2016), Article ID 30508. doi

Author's address: *Marek Čapek*, Mathematical Institute, Charles University, Sokolovská 83, 186 75 Praha 8, Czech Republic, e-mail: `marek.capek@post.cz`.