This preprint has not undergone peer review or any post-submission improvements or corrections. The Version of Record of this article is published in *Computational Particle Mechanics*, and is available online at https://doi.org/10.1007/s40571-021-00454-6

Smoothed Particle Hydrodynamics for blood flow analysis: development of particle lifecycle algorithm

Marko Topalovic[a*], Aleksandar Nikolic[a], Vladimir Milovanovic[b], Snezana Vulovic[a], Milos Ivanovic[c]

[a] *Institute for Information Technologies, University of Kragujevac, Jovana Cvijića bb, 34000 Kragujevac, Serbia*

[b] *Faculty of Engineering, University of Kragujevac, Sestre Janjic 6, 34000 Kragujevac, Serbia*

[c] *Faculty of Science, University of Kragujevac, Radoja Domanovića 12, 34000 Kragujevac, Serbia*

* Corresponding author: Marko Topalovic

Institute for Information Technologies, University of Kragujevac, Jovana Cvijića bb, 34000 Kragujevac, Serbia

Contact tel. +381606011884

E-mail address: topalovic@kg.ac.rs

ORCID: 0000-0001-6101-755X

Acknowledgements

This study was funded by grants from Ministry of Education, Science and Technological Development of the Republic of Serbia (Projects Numbers TR32036 and 451-03-68/2020-14/200378).

Abstract

The aim of this research was to facilitate the application of Smoothed Particle Hydrodynamics (SPH) method to Computational Fluid Dynamics analysis of turbulent flow through complex geometry blood vessels, and to compare it with the state-of-the-art Finite Element Method (FEM). SPH offers the possibility to observe motion of fluid fragment or particle inclusion within the Lagrangian material framework, giving researchers greater insight into the Fluid-Structure interaction such as transportation and distribution of medical particles, or buildup of plaque in atherosclerosis. In order to generate the fluid flow in SPH, the particles are created at inlet, and destroyed at the outlet. In this paper we present a novel lifecycle algorithm for generation and destruction of the particles using particle types, which is more flexible and suitable for the complex geometry models in comparison to the current state of the art commercial solutions, which use boundary planes. Our algorithm features mother and new-born particle types located at inlets used for the particle flow generation, and at the outlets, we have dying and killer particle types, which are used for deletion of particles. Based upon the current neighbors, the type of the particle is updated within the nearest neighbor search method, which is invoked in each time step. The capabilities of the new algorithm are demonstrated using a benchmark example and a realistic patient specific geometry, showing similar results, but the SPH advantages of particle tracking are yet to be utilized in our future work.

Keywords: Smoothed Particle Hydrodynamics; particle generation; particle destruction; inlet; outlet; Lagrangian fluid flow; fluid-structure interaction; blood vessels; Finite Element Method; k-ω turbulent model

1. Introduction

Smoothed Particle Hydrodynamics (SPH) is mesh-free numerical method which was originally designed for solving astrophysical problems [1,2]. Its purpose was later extended to Computational Fluid Dynamics (CFD) problems, governed by the Navier–Stokes equations [3], and finally solid mechanics, by adding the strength of materials into SPH equilibrium equations [4]. Nowadays, SPH is mostly used to simulate high velocity impacts such as bird impacting plane or bullet impacting the target [5,6,7]. It is also often used to analyse aircraft ditching into water [8] or boat-water interaction [9,10]. These problems involve large deformations [11] and breakdown of impacting or impacted body, and are often performed using coupling of SPH and Finite Element Method (FEM), increasing accuracy and limiting the computational load [12]. FEM is one of the most versatile and proliferated numerical methods [13], used both for analysis in various industrial [14,15,16] and bioengineering problems [17]. However, large

deformations cause mesh distortions [15] leading to inaccurate results, and that's where mesh-free [16] nature of the SPH method gives it a clear advantage. Because of this, SPH is often used in the bioengineering [17] to analyse problems like pulsatile flow inside left heart cavity [18] and shear stress accumulation in blood components behind heart valves [19].

Further modelling advancement involves patient-specific 3D geometry [20] and realistic cardiac mechanics that include inflation and active contraction of a ventricle [21]. The most comprehensive multi-physics model now includes cardiac electrophysiology, i.e. electromechanical coupling analysis of heart excitation and contraction [22].

SPH is also one of the most suitable numerical methods for particle based simulations of Red Blood Cells (RBC), along with the Dissipative Particle Dynamics (DPD) and Lattice Boltzmann Method (LBM) [23].

For a more accurate representation of RBC, SPH is coupled with Discrete Element Method (DEM) which is used to model RBC membrane, while SPH represents haemoglobin in RBC and plasma in the blood [24]. However, there is another novel approach in which shell-fluid analysis is done purely based on the SPH method [25]. It features extrapolation of pressure and velocity fields through the wall particles [25]. Deformation of RBC enables them to squeeze through narrow capillaries with stenosis, however, there is a critical diameter [24,25] after which the flow is blocked.

Another similar problem is the analysis of the interaction of the blood components and plaque in a stenotic coronary artery [26]. This model features FEM-SPH coupling in which SPH is used to model blood plasma, RBC as well as White Blood Cells (WBC). Modelling blood with SPH requires use of a large number of particles in order to avoid deviation of the velocity contours, as is pointed out in [27]. SPH can model not only blood-plaque interaction [26], but thrombus formation as well [28,29]. Elasticity of the wall can influence flow distribution after the bifurcation [30], but in order to obtain the realistic behaviour, comprehensive model with the surrounding tissue is required in order to avoid unrealistic deformations shown in [30].

The most recent, detailed review of the Fluid Structure Interaction (FSI) in SPH is given in [31] and [32], while a great example of FSI in SPH used for cardio-vascular analysis is shown in [33]. The review [31] gives general overview of

grid-based methods in comparison to the mesh-free SPH, as well as their coupling, to achieve flexible and computationally efficient FSI models. One of the problems that is pointed out in this review regarding FSI in SPH is the need for particle regularization and adaptive particle resolution [34,35] which is needed to achieve balance between thoroughly detailed model and computer efficiency. From review [32], one can get a concise summary of the state of the art in Weakly compressible SPH, Incompressible SPH and Arbitrary Lagrangian–Eulerian approach. While reviews [31] and [32] are focused primarily on the positive aspects of the SPH method and its various forms, another recent review [36] points out disadvantages and problems regarding the SPH method.

Namely, SPH still has a few grand challenges left to be solved [36], which are grouped into the following categories:

1) convergence, consistency and stability,
2) boundary conditions,
3) adaptivity,
4) coupling to other models,
5) applicability to industry.

Boundary Conditions (BC), which are the grand challenge No. 2 [36], sprung out many different solutions, developed by various research groups, but there are still some limitations that we will address in this paper.

The paper is organized as follows: in Section 2 we reflect on the basic features of the SPH method, i.e., kernel and the particle approximations, and stress calculations for elastic material model and viscous fluid. In Section 3 we discuss turbulence in SPH, and compare it with the implementation of Navier–Stokes equations in k-ω turbulent model in FEM. In Section 4 we show a detailed depiction of life-cycle algorithm for generation and destruction of SPH particles. In Section 5 we first compare results of SPH and FEM analysis for the benchmark example, and afterwards briefly describe the methodology used for analysis of patient specific models, followed by the results of the simulations for both SPH and FEM for the model of the real patient-specific carotid bifurcation. Finally, in the conclusion, we discuss the advances introduced to the SPH method with our novel particle life-cycle algorithm, and how SPH offers a great potential for analysis of the interaction of real particles within a fluid flow, that will be the focus of our future work.

2. SPH Approximations, Materials and Fluid Flow

Each pseudo-particle of SPH method [37] is similar to an element in FEM method [13], but since there are no common nodes between them, they can have different neighbours throughout the simulation, which enables better handling of the large deformations [11], but requires the search for the neighbours in every time step. Same as FEM method, SPH is based on continuum mechanics [38] approach which means that every SPH pseudo-particle [37] describes a certain section of material (fluid or solid), and not individual real particles. But, in the rest of the paper, we will use term "particles" instead of "pseudo-particles" for concise expression, although we still consider continuum sections [38], and not the real particles. SPH is also based on the Lagrangian material framework, meaning that the motion of the particular material section is observed [39]. On the other hand, when used for CFD analysis, FEM implements Eulerian spatial formulation which observes a fixed volume through which the fluid flows [39]. Eulerian formulation can predict pressure and fluid velocity with great accuracy, as well as forces exerted by fluid on the surrounding vessel, but, if we want to observe the movements and interactions of particles within the fluid flow [40,23,26] we would still need a Lagrangian formulation. The difference between Lagrangian and Eulerian formulation is shown in the Fig. 1.

Fig. 1 a) Lagrangian b) Eulerian formulation

As can be seen from the SPH review [32] and detailed investigation [39,41], Lagrangian–Eulerian coupling promises advantages of the both formulations within one numerical solver and the same model. In our models the considered volumes of fluid re are not large enough to make this coupling feasible, so our model is fully Lagrangian. Although fluid structure interaction is commonly studied using coupled SPH-FEM with fluid modelled with SPH particles and thin-walled structure (aircraft fuselage or ship hull) modelled with FEM shell elements [42], for bioengineering analysis [17], this coupling is often too excessive as both solid and fluid matter can be properly represented with SPH particles alone [25]. SPH uses kernel approximation and particle approximation [37] to model continuum mater (both solid and fluid) which we briefly explain in this section.

2.1 Kernel Approximation

The conservation laws of continuum mechanics [38] are expressed in the form of partial differential equations which are transformed into integral equations by interpolation function that gives "kernel estimate" of the field variables at point [17]. The exact value of the function $f(\mathbf{x})$ in integral form is given with (1):

$$f(\mathbf{x}) = \int_{\Omega} f(\mathbf{x}')\delta(\mathbf{x}-\mathbf{x}')d\mathbf{x}' \tag{1}$$

where $f(\mathbf{x})$ is a function of position vector $\mathbf{x}$ defined in the domain Ω and $\delta(\mathbf{x}-\mathbf{x}') = \begin{cases} 1 & \mathbf{x}=\mathbf{x}' \\ 0 & \mathbf{x}\neq\mathbf{x}' \end{cases}$ is the Dirac delta measure [37,40]. Replacing $\delta(\mathbf{x}-\mathbf{x}')$ with bell-shaped kernel function $W(|\mathbf{x}-\mathbf{x}'|,h)$ where h is the smoothing length (bell base radius), gives us a kernel approximation [37] of function $f(\mathbf{x})$:

$$\langle f(\mathbf{x})\rangle = \int_{\Omega} f(\mathbf{x}')W(|\mathbf{x}-\mathbf{x}'|,h)d\mathbf{x}'. \tag{2}$$

The integral form given in Eq. (2) is not practical for numerical implementation because analysed continuum is divided into a finite number of particles, which carry individual mass and occupy individual space.

2.2 Particle Approximation

Continuous integral representations given with Eq. (2) are converted to discrete forms of summation over all particles within the support domain [37]. The infinitesimal volume $d\mathbf{x}'$ is replaced by finite volume of the particle $\Delta V_\beta = m_\beta/\rho_\beta$ where m_β and ρ_β are particle mass and particle density [37]. With the summation of all particles within the support domain implemented in Eq. (2), we get particle approximation of a function $f(\mathbf{x})$ for particle α:

$$\langle f(\mathbf{x}_\alpha)\rangle \cong \sum_{\beta=1}^{NNP} f(\mathbf{x}_\beta)W(|\mathbf{x}_\alpha-\mathbf{x}_\beta|,h)dV_\beta = \sum_{\beta=1}^{NNP}\frac{m_\beta}{\rho_\beta}f(\mathbf{x}_\beta)W(|\mathbf{x}_\alpha-\mathbf{x}_\beta|,h) \tag{3}$$

where *NNP* is the number of nearest neighbouring particles [37].

2.3 Elastic Material Model in SPH

Blood vessels and solid wall in the benchmark example are modelled using the elastic material model [38]. However, deformations might be limited by the SPH particle constrains, as we will explain later in the algorithm and model descriptions in sections 4.3, 5.1 and 5.2. In elastic material model total stress tensor σ_{ij} is calculated as a sum of hydrostatic pressure p and stress deviator tensor S_{ij}:

$$\sigma_{ij} = -p\delta_{ij} + S_{ij}, \tag{4}$$

where δ_{ij} represent Kronecker delta [38]. Stress deviator tensor [38] in the next time step is calculated by adding its increment to the current step value:

$$^{t+\Delta t}S_{ij} = {}^{t}S_{ij} + \Delta t \cdot \frac{dS_{ij}}{dt}. \tag{5}$$

Stress deviator tensor rate is calculated using the shear modulus *G* [38]:

$$\frac{dS_{ij}}{dt} = G\left(\dot{\varepsilon}_{ij} - \frac{1}{3}\delta_{ij}\dot{\varepsilon}_{mm}\right) + S_{ik}\Omega_{jk} + S_{kj}\Omega_{ik}, \tag{6}$$

where strain rate tensor [43] is given with:

$$\dot{\varepsilon}_{ij}^{\alpha} = \frac{1}{2}\sum_{\beta=1}^{NNP}\frac{m^{\beta}}{\rho^{\beta}}\left(v_i^{\beta} - v_i^{\alpha}\right)\frac{\partial W^{\alpha\beta}}{\partial x_j^{\alpha}} + \frac{1}{2}\sum_{\beta=1}^{NNP}\frac{m^{\beta}}{\rho^{\beta}}\left(v_j^{\beta} - v_j^{\alpha}\right)\frac{\partial W^{\alpha\beta}}{\partial x_i^{\alpha}}, \tag{7}$$

while the rotation tensor [43] is given with:

$$\Omega_{ij} = \frac{1}{2}\sum_{\beta=1}^{NNP}\frac{m^{\beta}}{\rho^{\beta}}\left(v_i^{\beta} - v_i^{\alpha}\right)\frac{\partial W^{\alpha\beta}}{\partial x_j^{\alpha}} - \frac{1}{2}\sum_{\beta=1}^{NNP}\frac{m^{\beta}}{\rho^{\beta}}\left(v_j^{\beta} - v_j^{\alpha}\right)\frac{\partial W^{\alpha\beta}}{\partial x_i^{\alpha}}. \tag{8}$$

The Jaumann stress rate given by Eq. 6. is an objective stress rate, i.e. it does not depend on the frame of reference that is used, because SPH is based on Lagrangian material formulation, which was already explained at the beginning of section 2, and shown in Fig. 1. a).

2.4 Viscous Fluid in SPH

Similar to solids, total stress tensor σ_{ij} in the viscous fluid [44] consists of the hydrostatic pressure p and viscous stress ${}^{visc}\tau_{ij}$, which is the result of the fluid flow:

$$\sigma_{ij} = -p\delta_{ij} + {}^{visc}\tau_{ij}\,. \tag{9}$$

Viscous stress [37] can be calculated as:

$${}^{visc}\tau_{ij} = \mu\left(\partial_i v_j + \partial_i v_j - \frac{2}{3}\partial_k v_k \delta_{ij}\right) = \mu\varepsilon_{ij}\,, \tag{10}$$

where μ is the coefficient of dynamic viscosity and ε_{ij} is the strain rate tensor [37] which can be calculated as:

$$\varepsilon_{ij}^{\alpha} = \sum_{\beta=1}^{NNP} \frac{m_\beta}{\rho_\beta} v_j^{\beta\alpha} \frac{\partial W^{\alpha\beta}}{\partial x_i^\alpha} + \sum_{\beta=1}^{NNP} \frac{m_\beta}{\rho_\beta} v_i^{\beta\alpha} \frac{\partial W^{\alpha\beta}}{\partial x_j^\alpha} - \left(\frac{2}{3}\sum_{\beta=1}^{NNP} \frac{m_\beta}{\rho_\beta} \mathbf{v}^{\beta\alpha} \cdot \nabla_\alpha W^{\alpha\beta}\right)\delta_{ij}\,. \tag{11}$$

Next, we will compare SPH with FEM for turbulent flow analysis.

3. Turbulent Flow of Blood in SPH and FEM

In this section, we first discuss some SPH features that make this method convenient for turbulent flow analysis. Next, we briefly present the most important features of $k-\omega$ turbulent model which is implemented in FEM [45], and which we used for comparison with SPH model and verification of the results.

3.1 Turbulence in SPH

Due to the mesh-free, Lagrangian nature [39], the SPH method is naturally suited for modelling the turbulent flow. However, we must point out that some authors [46,47] propose extending the SPH equilibrium equations with sub-particle scale (SPS) turbulence stress. We, on the other hand, choose to rely on the reduction of particle size in order to observe the formation of eddies within the fluid flow [48]. The SPH is a collocation method [37], with the particles representing collocation points in which all the variables are calculated. Each particle has a bell-shaped kernel function (Gaussian, 3rd or 5th order polynomial) which is used to interpolate values between particles [37], and is somewhat similar to finite element size [13]. These kernel functions overlap, which causes a certain smoothing of variables,

hence the name of the method. Since smoothing could potentially reduce turbulence, the particle size should be significantly smaller than expected eddies. However, the increased number of particles consequently leads to impractical computational time [27], therefore, some intermediate, optimal size of the particles should be chosen, corresponding to the model geometry [48]. However, unlike FEM, which requires elements to have common nodes [13], and the transition between large and small elements must be gradual, SPH can have a mixture of larger and smaller particles [37], as shown in the Fig. 2.

Fig. 2 SPH Turbulent fluid flow

In the Fig. 2. the velocity of the observed particle is a consequence of the interaction with the neighbouring particles [37]. Closer particles with larger radius of the bell-shaped kernel function have a greater influence. However, if the particle size is not uniform, accuracy can be reduced significantly, due to the violation of consistency conditions. Dynamic particle refinement and splitting scheme [34,35] is one of the proposed answers to adaptivity grand challenge [36]. But, in this paper, we will use particles with the same mass and radius, because this is more practical if we want to use particle generation to create fluid flow. Therefore, in both the verification example and the real patient carotid bifurcation case, at the beginning of the analysis vessels are empty, and are subsequently filled by generated particles.

3.2 Turbulence in FEM, Reynolds equations and k-ω turbulent model

Due to their unstable and divergent nature, the turbulent problems are treated through statistical rather than deterministic methods [49]. In this paper, two-equation statistical model that calculates values in the viscous sublayer is used. This approach involves approximation of the velocity of the fluid that can be determined as the sum of the average values of velocity and velocity fluctuations around that value [40]. The Reynolds Averaged Navier-Stokes (RANS) equations are based on the Navier-Stokes equation, and the equation of continuity [50]. Navier-Stokes equation for incompressible viscous fluid flow [51] in tensor notation is given with:

$$\rho\left(\frac{\partial v_i}{\partial t}+v_j\frac{\partial v_i}{\partial x_j}\right)=-\frac{\partial p}{\partial x_i}+\mu\frac{\partial}{\partial x_j}\left(\frac{\partial v_i}{\partial x_j}+\frac{\partial v_j}{\partial x_i}\right), \tag{12}$$

where ρ is fluid density, p is pressure in fluid, t is time, v_i, v_j are velocity components of the fluid and μ is dynamic viscosity. The equation of continuity [13], for the incompressible fluid, can be written in the following form:

$$\frac{\partial v_i}{\partial x_i}=0\,. \tag{13}$$

If the fluid flow is observed by statistically averaged parameters, each variable can be represented as the sum of the time averaged value for that parameter and the fluctuation around this value [49]:

$$v_i\left(x_i,t\right)=\overline{v}_i\left(x_i\right)+v_i^{'}\left(x_i,t\right), \tag{14}$$

where $v_i^{'}$ denotes velocity fluctuations around a mean (averaged) value [49] of velocity $\overline{v}_i\left(x_i\right)$ which is calculated as:

$$\overline{v}_i\left(x_i\right)=\lim_{T\to\infty}\frac{1}{T}\int_0^T v_i\left(x_i,t\right)dt\,. \tag{15}$$

Substitution of the previous equations into Eq. 12. with some approximations and simplifications [35] gives us the following equation:

$$\rho\left[\frac{\partial\overline{v}_i}{\partial t}+\overline{v}_j\frac{\partial\overline{v}_i}{\partial x_j}\right]=-\frac{\partial\overline{p}}{\partial x_i}+\frac{\partial}{\partial x_j}\left[\left(\mu_{eff}\right)\left(\frac{\partial\overline{v}_i}{\partial x_j}+\frac{\partial\overline{v}_j}{\partial x_i}\right)\right], \tag{16}$$

where μ_{eff} is the effective dynamic viscosity represented by the sum of the dynamic viscosity and turbulent dynamic viscosity [49]:

$$\mu_{eff}=\mu+\mu_T\,. \tag{17}$$

The turbulent dynamic viscosity is calculated as the ratio of the kinetic energy of turbulence k and specific dissipation of the kinetic energy of turbulence ω [50]:

$$\mu_T=\alpha^*\rho\frac{k}{\omega}\,. \tag{18}$$

The kinetic energy of the turbulence k is calculated from the following equation:

$$\rho\left(\frac{\partial k}{\partial t}+\overline{v}_j\frac{\partial k}{\partial x_j}\right)=\frac{\partial}{\partial x_j}\left[\left(\mu+\sigma^*\mu_T\right)\frac{\partial k}{\partial x_j}\right]+P_k-\beta_k\rho k\omega, \tag{19}$$

where P_k represents the effect of kinetic energy of turbulence [50] calculated as:

$$P_k = \mu_T \left(\frac{\partial \overline{v_i}}{\partial x_j} + \frac{\partial \overline{v_j}}{\partial x_i} \right) \frac{\partial \overline{v_i}}{\partial x_j} . \tag{20}$$

Specific dissipation of the kinetic energy of turbulence ω is variable characterizing the scale of turbulence [50] and is calculated using the following equation:

$$\rho \left(\frac{\partial \omega}{\partial t} + \overline{v}_j \frac{\partial \omega}{\partial x_j} \right) = \frac{\partial}{\partial x_j} \left[(\mu + \sigma \mu_T) \frac{\partial \omega}{\partial x_j} \right] + \alpha_k \frac{\omega}{k} P_k - \beta_\omega \rho \omega^2 . \tag{21}$$

The constants α^*, α, β_ω, β_k, σ and σ^* from Eq. 18-21. have the following values:

$$\alpha^* = 1, \alpha_k = \frac{5}{9}, \beta_\omega = \frac{3}{40}, \beta_k = \frac{9}{100}, \sigma = \frac{1}{2}, \sigma^* = \frac{1}{2} . \tag{22}$$

After the modification of Navier-Stokes equation (Eq. 12.) with an averaged value of velocity (Eq. 15.) the same is done with the equation of continuity (Eq. 13.):

$$\frac{\partial \overline{v_i}}{\partial x_i} = 0 . \tag{23}$$

Eq. 16. and Eq. 23. represent the Reynolds averaged Navier-Stokes equations [52]. Using the Galerkin method on Reynolds equations [52,53] we get a linear system of equations in the matrix form:

$$\begin{bmatrix} \mathbf{M} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\mathbf{V}} \\ \dot{\mathbf{P}} \end{bmatrix} + \begin{bmatrix} \mathbf{K}_{vv} + \mathbf{K}_{\mu vt} & \mathbf{K}_{vp} \\ \mathbf{K}_{vp}^{\ T} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{V} \\ \mathbf{P} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_v + \mathbf{F}_s \\ 0 \end{bmatrix} . \tag{24}$$

Using the same method [52,53] applied to Eq. 19. and Eq. 21. we get the following system of equations in matrix form [45]:

$$\begin{aligned} [\mathbf{M}][\dot{\mathbf{k}}] + [\mathbf{K}_{VK} + \mathbf{K}_{MK} + \mathbf{K}_{\beta k}][\mathbf{k}] - [\mathbf{K}_{VV1}][\mathbf{V}] &= \mathbf{F}_{SK2} \\ [\mathbf{M}][\dot{\boldsymbol{\omega}}] + [\mathbf{K}_{VK} + \mathbf{K}_{M\omega} + \mathbf{K}_{\beta\omega}][\boldsymbol{\omega}] - [\mathbf{K}_{VV2}][\mathbf{V}] &= \mathbf{F}_{S\omega} \end{aligned} . \tag{25}$$

Grouping together Eq. 24. and Eq. 25. we get a single large matrix equation system [45]:

$$\begin{bmatrix} \mathbf{M}_V & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{M}_K & 0 \\ 0 & 0 & 0 & \mathbf{M}_K \end{bmatrix} \begin{bmatrix} \dot{\mathbf{V}} \\ \dot{\mathbf{P}} \\ \dot{\mathbf{k}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{vp} & 0 & 0 \\ \mathbf{K}_{vp}^{\ T} & 0 & 0 & 0 \\ -\mathbf{K}_{VV1} & 0 & \mathbf{K}_{33} & 0 \\ -\mathbf{K}_{VV2} & 0 & 0 & \mathbf{K}_{44} \end{bmatrix} \begin{bmatrix} \mathbf{V} \\ \mathbf{P} \\ \mathbf{k} \\ \boldsymbol{\omega} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_1 \\ 0 \\ \mathbf{F}_{SK2} \\ \mathbf{F}_{S\omega} \end{bmatrix} , \tag{26}$$

where substitutions for shorter notation are given with:

$$\begin{aligned} \mathbf{K}_{11} &= \mathbf{K}_{vv} + \mathbf{K}_{\mu vt}, \quad \mathbf{K}_{33} = \mathbf{K}_{\mathbf{VK}} + \mathbf{K}_{\mathbf{MK}} + \mathbf{K}_{\beta k} \\ \mathbf{K}_{44} &= \mathbf{K}_{\mathbf{VK}} + \mathbf{K}_{\mathbf{M}\omega} + \mathbf{K}_{\beta\omega}, \quad \mathbf{F}_1 = \mathbf{F}_v + \mathbf{F}_s \end{aligned} . \tag{27}$$

Matrix equation (Eq. 26.) can be solved using standard FEM incremental-iterative procedure [13]. The vector of kinetic energy of turbulence $\mathbf{k}$ [52,53], the specific dissipation of turbulent kinetic energy $\boldsymbol{\omega}$ [52,53] and the velocity of fluid at the end of the time step $\mathbf{V}$ can be calculated as the values from the previous iteration and the increment from the current iteration [13]. Matrix $\mathbf{M}$ is mass matrix [13], matrices $\mathbf{K}_{vv}, \mathbf{K}_{\mathbf{VK}}, \mathbf{K}_{\mathbf{VV1}}, \mathbf{K}_{\mathbf{VV2}}$ are matrices with convective members, matrices $\mathbf{K}_{\mu vt}, \mathbf{K}_{\mathbf{MK}}, \mathbf{K}_{\mathbf{M}\omega}$ are matrices with viscous members and $\mathbf{F}_v, \mathbf{F}_s, \mathbf{F}_{\mathbf{SK2}}, \mathbf{F}_{\mathbf{S}\omega}$ are the volumetric and surface external forces, as explained in detail in [13,45,52,53].

4. SPH particle lifecycle algorithm for particle generation and destruction (inflow/outflow)

4.1 Initial state of the art in the commercial software and scientific solutions

As said before, one of the first applications of the SPH method was in CFD [3], but that was mostly widespread in the aerospace and naval industry for impact or free surface analysis [6-10]. Although there are numerous examples of SPH used in bioengineering and biomechanics [18-33], there are still significant challenges that make this numerical method less than ideal solution. One of these grand challenges [36] is the prescription of realistic boundary conditions, needed for the simulation of fluid flow in the Lagrangian formulation, which we will be addressing in this paper.

One frequently used approach in the scientific community was to develop a periodic boundary condition [37,54,55,56] which creates replica layers that connect the inlet and the outlet of the cylindrical vessel [37,56]. However, this solution shown in the Fig. 3 is feasible only in the mechanical and civil engineering for e.g. pipeline analysis because the inlet and outlet must have the same cross-section.

Fig. 3 Periodic boundary conditions

On the other hand, commercial approach to boundary conditions is not so sophisticated and involves setting up activation and deactivation boundary planes [57,58,59] which is implemented in the commercial LS-Dyna solver (LSTC, Ansys, Inc., Canonsburg, Pennsylvania, US) using BOUNDARY_SPH_FLOW and CONTROL_SPH (BOXID) keywords [60,61], shown in the Fig. 4.

Fig. 4 LS-Dyna SPH flow generation

Activation plane is defined using a particular boundary node and a vector [61]. SPH nodes used for flow generation are grouped by a node set or a part set [61], and at time t=0 they are deactivated. Once they pass through the activation plane, they are included in the calculations [60].

This solution could provide us with a generation of SPH particles flow into the irregular shaped inlet [59], and deletion of particles when they reach a predefined box, but, in that case, the outlet must lie on the box face. This however means that in the case of multiple outlets they all must be parallel, or perpendicular, which is inconvenient for the realistic bioengineering analysis. Also, using commercial SPH solvers requires extending the modelled reservoir tubes long enough to accommodate the inflow and outflow of particles for the duration of analysed cardiac cycles [20].

In comparison to the commercial solution [60,61], the scientific community has enhanced inflow/outflow permeable boundary conditions [62] with generation of particles which greatly reduces starting model size. Our initial implementation of inlet/outlet BC is similar to the state of the art scientific solution [63] with comparison and capabilities given in the next section.

4.2 Initial particle generation/destruction algorithm

In order to take advantage of SPH inherit capability to handle turbulence [49], and use it in bioengineering [17], we must somehow generate flow through blood vessels with complex geometry [26]. The first iteration of our solution was to define boundary planes [62] for the generation and deletion of particles as is illustrated in the Fig. 5.

Fig. 5 Initial SPH generation and deletion solution

Terminology that we used for the specification of components and features of particle generation/destruction algorithm is adopted in order to serve as an intuitive explanation of the function they perform, which will be particularly meaningful later, as we describe our improved complex solution. For example, a *boundary planes* term which is used in [62] is replaced with *birth plane* and *death plane*, which more precisely define their role in generation and destruction of particles.

Unlike LS-DYNA implementation [60,61] once inactive particle passes through activation (*birth*) plane, it get copied upstream [62] and thus removing the need for modelling the whole volume of moving fluid [20]. Inflow/outflow BC presented in [62] features the inflow zone. The length of this zone in the flow direction is equal to or greater than the smoothing length [62]. The particle that passes the boundary plane is copied to the beginning of the inflow zone, with prescribed velocity, while the pressure is extrapolated from the interior domain [62]. Similar in terminology and function is the solution presented in [64], which instead of inflow and outflow zones has two buffer zones and a particle management algorithm which transform open boundary particles into regular fluid particles and vice versa. The methods presented in [62] and [64] require the boundary to be planar and the flow to be normal to the boundary, which is one of the major drawbacks of this method that we will correct in our algorithm.

At the beginning of the analysis, we define an initial position for every particle, as shown with blue colour in the Fig. 5, and let the particle move from its initial position in the direction of the flow. To visualize this motion in the Fig. 5. we use different colour (blue to pink) representing discrete time steps. When the particle coordinates are updated, we check if the particle has passed the predefined *birth plane*. Once this condition is satisfied (pink position in the Fig. 5.), the particle is copied back to its initial position (marked with black arrow in the Fig. 5.). But in the next time step, this condition is still satisfied, and the particle is copied once more (red arrow in the Fig. 5.). This would create two copies, created in the two consecutive time steps, with very small inter-particle distance, which would create high repulsive forces [66] and instability [67] of the model. Therefore, it was necessary to introduce a limitation in the implementation that each particle can be copied only once. Also, from this analysis it was evident that the initial particle

should be placed closer to the *birth plane* to avoid discontinuities in the particle flow. The adequate initial position is coloured yellow in the Fig. 5.

Afterwards, we added a so-called *life-status* property to all particles. This property plays the same role as set used in [68] to group inlet, fluid and outlet particles, however the whole particle array from [68] is too cumbersome especially block rearrangement after some particles change set. Likewise, inflow threshold and outflow threshold from [68] correspond to our *birth* and *death* planes. The initial particle has *life-status new-born* (=0). Once it passes through the *birth plane*, it copies itself to *copy1* and it gets a *life-status adult* (=1). The coordinates of the *adult* particles are not checked against *birth plane*, and hence they are never copied again. *Copy1* particle is now the new *new-born*, and once it too passes through *birth plane*, it is copied to *copy2* particle, while *copy1* becomes another *adult*. This process repeats from the moment the initial particle passes through the *birth plane*, until the end of the simulation.

On the other side (outlet), we have the *death plane*. Once the *adult* particle passes through, it gets deleted (grey).

In the Fig. 5. the process of creation and deletion of the particles is demonstrated on 1D example, but the algorithm is identical for real-life 3D problems. First, we calculate the distance from a point i.e. SPH particle, to the *birth* or *death* plane. Second, we check if its within prescribed tolerance, if yes then we update particle *life-status* accordingly.

This is slightly different from the solution described in [62] where the particle is copied to the beginning of the inflow zone. Similar is the entry boundary condition [65] that has entry layer and three replicated layers of image particles with prescribed velocities. Once the entry layer particle [65] moves beyond the prescribed boundary it becomes interior particle [65] and another image particle is injected in the last replicated layer [65]. The difference between our solution and methods presented in [62] and [65] is that in our solution the distance between the initial particle position and the *birth plane* determines the frequency of the new particle generation (which is in accordance with the prescribed velocity), while in [62] it's the width of the inflow zone, and in [65] it contains three fixed layers of replicated particles. Either way, the boundary deficiency problem is solved. In order to generate new particles at desired inter-particle distance, our solution

needed enhanced if condition that checks whether the particle has passed through birth plane, and if that's the case the particle is copied to its initial position. Unlike LS-DYNA Deactivation box [60,61] shown in the Fig. 3. which practically has six *death planes*, our initial solution has only one, to avoid unnecessary checks in every time step [37].

This solution is suited for the engineering applications [4], such as pipeline analysis [57], modelling of mould filling [69] or filling of liquid containers [70], but for the bioengineering application [5] it is inadequate, because model outlets [55] need to be on the same *death plane*.

Another issue that is present in LS-DYNA is the sudden drop of pressure and lack of consistency [37,67] when the particles are deleted. The solution to the consistency issue is adding ghost particles downstream [57]. These particles have the pressure set to zero, and prescribed velocity of water front [58]. To avoid a sudden drop of pressure, outlet ghost particles can have frozen physical variables [68].

Inflow/outflow boundary conditions can be used to indirectly prescribe pressure of boundary particles using time variation of water level in the reservoir [57,68]. Current state of the art pressure inlet and outlet boundary conditions [71] enables generalised constant pressure on open boundaries to be specified. This is done by controlling incoming or outgoing particle velocity to assure a target pressure and density is achieved at the theoretical boundary interface [71]. Therefore, it is still velocity prescribed problem, but with pressure as a controlled measure.

The most advanced academic/scientific SPH solver DualSPHysics [63,72,73] has buffer zones [64] with recommended five layers [65] of buffer particles. DualSPHysics has free surface elevation prescription [57,68] as well.

However, when it comes to using described inlet/outlet algorithms in bioengineering applications, there are some disadvantages that could be solved with the solution proposed within this work. First, in DualSPHysics the buffer layers are created by defining simple geometries like a rectangle or a circle [73], therefore complex 3D geometry found in inlets/outlets of cardio-vascular models cannot be accurately represented. Second, prescribed velocity must be perpendicular to the buffer threshold which means that the inlet must be placed so that prescribed velocity is collinear with the predicted blood flow. Third, prescribed velocity profile in DualSPHysics can be fixed constant, fixed linear, or

fixed parabolic [73], but elaborate velocity profiles like Womersley profile which characterise pulsatile flow cannot be specified.
Our initial methodology for particle generation and destruction is virtually the same as DualSPHysics inlet/outlet algorithm [63,72,73], and algorithms presented in [57,62,64,65,68]. It has the same functionality with different terminology, but also the same shortcomings when it comes to bioengineering applications. We fixed these drawbacks in the next iteration of our lifecycle algorithm.

4.3 Novel flexible generation/destruction algorithm

The most important improvement of the initial solution was to replace the *birth plane* which is equivalent to buffer threshold from DualSPHysics with *mother SPH particles* (*life-status* = -1) and the *death plane* which is equivalent to deactivation boundary plane [57,58,59] implemented in LS-Dyna with *killer SPH particles* with *life-status* = -2. In the contrast to the *new-born* particles, which correspond to open boundary particles in inlet buffer [64] we also added *dying particles* with *life-status* = 2. These *dying particles* are located at the outlet [55,64], and their properties are unchanged, except for coordinates [68], which prevents a sudden drop of pressure and consistency deficiency [37,67]. In the following text, we will explain the functionality of the new life-cycle algorithm shown in the Fig. 6.

Fig. 6 Improved SPH generation and deletion

In the Fig. 6. the brown particles represent blood vessel wall made of elastic material, while the pink particles represent viscous fluid (blood) *adult* particles. Blue are *mother* particles and black are *killer* particles, which both have fixed coordinates. Green are *new-born* particles which have prescribed velocity and other properties, same as DualSPHysics buffer particles [73], and yellow are *dying* particles, whose properties remain unchanged, except coordinates [68]. Grey are particles that are being deleted.
Unlike *birth plane* from the first solution, the *mother* particles (blue) are behind *new-born* particles (green), and instead of checking the distance between *new-born* particles and *birth plane*, we now check *new-born* neighbours. This makes prescription of *birth plane* side, which is in DualSPHysics defined using keyword

direction in input file [73], unnecessary and obsolete. *New-born* particles flow away from *mother* particles, and once they are no longer in contact, they copy themselves to the initial position, marked with an arrow in the Fig.6. The newly copied *new-born* particles then start to flow with prescribed velocity, while the old, original *new-born* particles now become *adult* (pink). The solution presented in this paper can be modified and upgraded for prescription of pressure boundary conditions needed for simulation of pressure gradient driven flow [65], which will be evaluated in the discussion. Currently, however, velocity or body force must be prescribed at the inlet to the *new-born* particles and pressure must be prescribed to the *dying* particles.

The velocities of the adult particles are calculated according to [37] in every time step. *Adult* fluid particles continue to flow within the boundaries set by solid elastic wall particles (brown), until they come in contact with *killer* particles (black). Then they become *dying* particles (yellow), which means that their properties are no longer updated in every time step, except for coordinates, same as in [68]. Due to the velocity they had when their life-status became *dying*, they continue moving in a straight line, as long as they stay in contact with the *killer* particles. Once *dying* particles are no longer in contact with *killer* particles, they are deleted (grey).

Killer particles have very small smoothing length compared to the regular *adult* fluid particles, so they form more linear barrier.

Smoothing length of *mother* particles is used to fine tune intervals at which new particles are born, which is much more practical than moving them, since they don't need to belong to the plane which is perpendicular to the Cartesian axis.

Fluid *adult* particles are unconstrained and are free to move within the volume confined by wall. Boundary condition prescription for the wall particles is a particular challenge. If these particles remain unconstrained, they would flow with the fluid making the model pointless. If all of the wall particles are constrained, elasticity and deformation would be eliminated. Fixed constrain imposed to inlet and outlet wall particles would produce unrealistic deformations because the effect of blood vessel before and after analysed section is removed, and the influence of the surrounding tissue is neglected [30]. The most realistic model would imply blood vessel and enclosing flesh represented by the several layers of SPH particles with only the outer layer fully constrained, leaving the inner layers deformable. We

tried this on other models, not shown in this paper, and the results were on average within 3% of velocity values for one layer, fully constrained wall. Hence, in this paper, we modelled wall with one layer fully constrained particles, except a few chosen particles which were left unconstrained that we used for plotting physical values of interest.

Using *mother* and *killer* particles instead of buffer threshold [73], gives us the possibility to prescribe irregular shaped inlet/outlet boundaries. Since the initial particle position is used as a reference for particle copy and not the distance from the inflow zone [62], prescribed velocity does not need to be perpendicular to the boundary. Also, instead of particle management algorithm [64], the *mother* and *killer* particles allow linked list neighbour search algorithm to be reused for changing the *life-status* of particles, thus improving SPH efficiency, which will be explained in the next section.

4.4 Linked list neighbour search

The most time-consuming process in the SPH analysis is the search for neighbouring particles [37]. As particles move around the model, their neighbours change, so this action must be performed at each time step [37].

The simplest, but infeasible solution is the direct search [74], which implies that for every particle, we must cycle through all of the other particles, calculate inter-particle distance r_{ij}, and check if that distance is less than smoothing length h of the bell-shaped kernel function W. If this condition is satisfied, the tested particle is added to the nearest neighbour list [74] and used in particle approximation [37], as explained in the Section 2.

The more practical solution is to divide the entire model into subdomains [75], whose dimensions correspond to the smoothing length (as illustrated in the Fig. 7.).

Fig. 7 Linked list subdomains

These subdomains form linked lists [74,75] as depicted in the Fig. 7. Based on its coordinates, each particle is assigned to the specific cell [75]. When we search for the neighbours of a specific particle, we check all the particles belonging to the same cell as the observed particle (red cell), and the connected (blue) cells [75].

We used linked list neighbour search algorithm [74,75] to determine the life status of a particular particle as well. When a certain particle is within the smoothing length [37] of the observed particle, we check if the certain particle is a *mother* or a *killer*, and change the life status of the observed particle accordingly.

5. Results and Discussion

For the simulations, in-house developed software is used. The FEM simulations are performed using PAK-F software [76] that was already successfully applied to model several problems in bioengineering [17]. The SPH simulations are carried out using SPH07 software [77] that is adapted for a broader application in bioengineering, using the approach described in this paper.

5.1 Benchmark: fluid flow in backward facing step channel

This benchmark model is often used for validation of the numerical fluid flow simulation, with experimental results published in [78] and the problem geometry [79] presented in the Fig. 8.

Fig. 8 Backward facing step problem geometry

The turbulence in this model is caused by a sudden increase of a vessel cross-section (after the step) with eddy forming right behind it as is illustrated in the Fig. 8. The step has a relative height *H*, and its relative length is 4*H* [79]. The width of the channel is 5*H* and its total length is 20*H* [79]. The important measure is the distance after which persistent fluid flow touches the vessel wall, denoted as *Xr* in the Fig. 8. which is experimentally determined as 6*H* [78]. In the case of the SPH, the wall is modelled as elastic material with density $\rho = 7.83e^{-3} g/mm^3$, Young's modulus $E = 2.07e^{11} Pa$, and Poisson's ratio $\nu = 0.3$. Inner wall particles are unconstrained, while the outer layer is fully constrained. One of the inner, unconstrained particles is SPH particle 467 (denoted in Fig. 14) which we used to plot the values of interest. In the FEM simulation we do not model F-S interaction. Instead, we prescribe the boundary condition that ensures that fluid velocity on the wall surface is equal to zero. Corresponding FEM node 4819 is used to obtain fluid pressure over time. This node is denoted in Fig. 13.

Inside the vessel there is a Newtonian fluid with density $\rho = 1e^{-3} g/mm^3$, and dynamic viscosity $\mu = 1e^{-2} g/mms$. With the averaged inlet velocity of $v_{inlet} = 16666 mm/s$ we get $\mathrm{Re} = 5000$ [78]. The FEM model consists of 7600 elements, while SPH model has on average 7820 particles. The FEM analysis was performed within the time frame of 100 seconds using 200 time steps with fixed duration of 0.5 s. In SPH, we added 10 seconds for the model to be filled with particles, while the time step length is determined dynamically using Courant-Friedrichs-Lewy condition. The results in the Fig. 9. and Fig. 10. show velocity in the flow direction for $k-\omega$ model implemented in solver PAK-F and corresponding results from the SPH07 solver at the end of the analysis (t=100s).

Fig. 9 FEM results (VX velocity) from PAK-F using *k-ω* model

Fig. 10 SPH results (VX velocity) from SPH07

The streamlines visualize eddy formation determined by PAK-F and SHP07 at the beginning of the analysis (t=1s) in the Fig. 11. and Fig. 12. respectively.

Fig. 11 FEM results (streamlines) from PAK-F using *k-ω* model

Fig. 12 SPH results (streamlines) from SPH07

Differential pressure field occurs due to changes of channel cross-section and resulting eddy formation and the subsequent turbulent flow. This field, obtained from PAK-F and SHP07, at the beginning of the analysis (t=1s), is shown in the Fig. 13. and Fig. 14.

Fig. 13 FEM results (differential pressure) from PAK-F using *k-ω* model

Fig. 14 SPH results (differential pressure) from SPH07

Differential pressure value at step edge during the whole analysis is shown in the Fig. 15.

Fig. 15 Differential pressure at step edge

Comparing Fig. 9. to Fig. 15. one can notice great correspondence between FEM results using $k-\omega$ model and the SPH results. In the SPH analysis, we can observe small velocity and pressure fluctuations between neighbouring particles, due to coarse particle size and non-uniform particle distributions [65,68], while the velocity transitions in the FEM analysis are significantly smoother.

5.2 Real life example: patient-specific carotid bifurcation

This geometry is obtained from the CT scan of an anonymous patient shown in the Fig. 16.

Fig. 16 Scanned carotid geometry

The bifurcation section of the right carotid artery is extracted and converted into FEM mesh depicted in the Fig. 17. using STL2FEM software [80] and FEM nodes are consecutively converted into SPH particles [81].

Fig. 17 FEM mesh of carotid bifurcation obtained from STL2FEM software

The FEM model of the fluid (blood) within carotid bifurcation analysed volume consists of 20237 3D elements. Like in the previous benchmark example, we do not model F-S interaction with FEM and instead we prescribe the fluid velocity on the wall surface to be equal to zero.

SPH patient-specific modelling of human organs and the cardiovascular system is not new [82-84], however, since FEM is much more widespread in comparison to the SPH method, and is compatible with significantly more pre-processing and post-processing software solutions, the most common practice for generation of SPH particles is to first create FEM mesh, and then to convert FEM nodes, or element centres, into the SPH particles [81]. However, if we observe the Fig. 17. we can see that the size of the elements at the outlet branches is significantly smaller than the size at the common section surface. Using SPH particles with variable size would be impractically difficult, because if we would assume that

every particle has the same mass and smoothing length, then we would get sections with excessive overlapping and sections with gaps. Thus, for the SPH analysis, we took only vessel shell elements, and divided larger elements into smaller ones using commercial software FEMAP (Siemens PLM Software, Plano, Texas, US) until all the elements were approximately the same size. Thereafter, we used LS-PrePost (LSTC, Ansys, Inc., Canonsburg, Pennsylvania, US) to create SPH particles corresponding to the element centres. That way we got vessel wall made of elastic material, and *new-born* and *killer* particles at the inlet and the outlet respectively. *Mother* particles are obtained from the small shell section which we offset in FEMAP. Unlike FEM analysis, in the case of SPH, we first fill the empty vessel (bifurcation) with particles, and then compare the behaviour with FEM. Initial SPH model shown in the Fig. 18. has 7732 particles, 6499 belonging to the wall, while the rest are either *new-born*, *mother* or *killer* particles.

Fig. 18 SPH model of carotid bifurcation: empty vessel with mother, newborn and killer particles

The FEM analysis was carried out during the period of 0.8s, i.e. a single heartbeat. The FEM analysis consists of 40 time steps of 0.02s, while the SPH solver determines the time step automatically. SPH time step varies during the analysis, but the output files are written regularly at 0.02s to facilitate comparison with the FEM.

Inside the carotid vessel, there is blood modelled as a Newtonian fluid with density $\rho = 1.05e^{-3} g/mm^3$, and dynamic viscosity $\mu = 3.675e^{-3} g/mms$. The wall acts as an elastic material with density $\rho = 1.1e^{-3} g/mm^3$, Young's modulus $E = 3.61e^5 Pa$, and Poisson's ratio $\nu = 0.49$. In the carotid model presented in this paper, wall is represented as a single layer with fully constrained particles, except for particles used for physical values measurement as explained in the Section 4.3. According to [85], Lennard-Jones repulsive force [3] is applied to the fluid particles approaching the wall. As can be seen from the Fig. 18., the initial velocity profile is flattened parabola with Womersley number α=4.7 according to Ponzini [86]. At the inlet, a parabolic velocity profile is prescribed, with the averaged inlet velocity of $v_{avg} = 169mm/s$ and the maximum velocity during systole of $v_{max} = 327mm/s$.

Velocity fields during systole (0.1s) for PAK-F and SPH07 are shown in the Fig. 19. and Fig. 20. respectively.

Fig. 19 FEM results: velocity magnitude during systole

Fig. 20 SPH results: velocity magnitude during systole

Same as in the case of backward facing step, we can see velocity fluctuations between SPH particles due to coarse particle size and non-uniform particle distributions [65,68], while the FEM results show smoother transitions between layers with different velocities. The corresponding streamlines are shown in the Fig. 21. and Fig.22.

Fig. 21 FEM streamlines: velocity magnitude during systole

Fig. 22 SPH streamlines: velocity magnitude during systole

The differential pressure field during systole is shown in the Fig. 23. and Fig.24.

Fig. 23 Differential pressure field in FEM during systole

Fig. 24 Differential pressure field in SPH during systole

Changes of the total pressure at bifurcation point, which is measured at FEM node 16438 marked in Fig. 23. and SPH particle 747 marked in Fig 24. are shown in the Fig.25.

Fig. 25 Total pressure at bifurcation

The results presented for both benchmark example, and the real patient's carotid model show that SPH gives more chaotic behaviour with pressure variations caused by density fluctuations [68] and non-uniform particle distributions [65] in both industrial (experimental) and bioengineering applications of turbulent fluid flow analysis. FEM is advantageous over SPH in terms of model (mesh)

generation, and execution time, but that can change with proliferation of the SPH method and its great parallelization potential [63,87].

5.3 Pressure calculation and prescription issues

In this section we will discuss pressure computation problems and possibility of using our life cycle algorithm for pressure gradient driven analysis.
The SPH method uses a weakly compressible form with an equation of state that defines pressure as a function of density [68]:

$$p = c_0^2 (\rho - \rho_0), \tag{28}$$

where c_0 is initial sound speed and ρ_0 is initial density. A large speed of sound $c_0 > 10 \cdot \max(|\mathbf{v}|)$ is used to maintain density variation lower than 1% ρ_0 , but this also leads to smaller time step [88]. The greater time step can generate large and non-physical fluctuations of pressure, which may cause numerical instability [65] and corrupt the results.
The current algorithm implies prescribed velocity to *new-born* particles and zero pressure prescribed to *dying* particles. In order to have a pressure gradient driven problem [71] a few modifications to life-cycle algorithm must be performed. First to match calculation point [71] which is used to measure pressure in order to adjust inlet velocity, we must add another layer of particles which would have special *life-status* for example let's call them *child* particles. Child particles would be between *new-born* and *adult* particles, and their average pressure could be compared to the prescribed pressure in order to calculate a scale factor for *new-born* particle velocities in the next step. If we would want to prescribe pressure directly from a time-varying function to *new-born* particles, we would not need *child* particles, but instead a few layers of ghost particles should be added behind *mother* particles, and around *new-born* in order to have neighbouring consistency, and to prevent backflow.
Pressure prescription however, would create one big issue: velocity profile. Pulsatile blood flow in the cardiovascular system is characterised by complex Womersley profile [86], and pressure prescription would generate fixed linear [73] profile at the inlet. This could be corrected by extending inlet and outlet zones according to [89], giving enough space for Womersley profile to be formed due to the interaction between blood and vessel, but this however would

significantly increase the model size, and the reduction of the model size was one of the main motives in the development of our *life-cycle* algorithm.

6. Conclusion

In this paper, we presented an improvement to inflow/outflow boundary conditions that we named *life-cycle* algorithm. The main difference in comparison to the traditional implementation is that we utilize special sets of particles that we named *mother* and *killer* particles instead of buffer threshold [73], to separate analysed fluid domain from the inflow and outflow regions. The main advantage of this implementation is that inflow/outflow boundary can have multiple complex shapes and positions often found in bioengineering, and the second one is that prescribed velocity does not need to be perpendicular to the boundary. This proposed solution can be used to augment existing algorithms like the ones presented in the literature [57,62,64,65,68,71,73].

In order to verify functionality of the new algorithm, we compared the results of the state of the art FEM method with Eulerian formulation and k-ω turbulence model with SPH mesh-free Lagrangian numerical method. The results indicate that SPH can be used to obtain similar results to that of a traditional and well-established FEM solver (PAK-F). SPH results are realistic with some velocity fluctuations between neighbouring particles and differential pressure field. Also, the SPH method has an advantage, which is presented in paper [48] when it comes to the fluid-solid interaction analysis due to the Lagrangian formulation, which had so far predominantly industrial utilization.

Bioengineering application of this method was limited by the need for generation and destruction of particles in order to create fluid flow, which is the issue that we addressed in this paper. The novel particle lifecycle algorithm has the advantage over commercial solutions employing traditional activation plane/deactivation box in terms of flexibility and robustness. Using this new approach, in our future work we will study fluid-solid interaction, such as tracking of cholesterol deposition or distribution of drugs within the fluid flow, bleeding and coagulation, vessel deformations and stent deployment.

Compliance with Ethical Standards

Conflict of Interest: On behalf of all authors, the corresponding author states that there is no conflict of interest.

Funding: This study was funded by grants from Ministry of Education, Science and Technological Development of the Republic of Serbia (Projects Numbers TR32036, 451-03-9/2021-14/200378.)

References

[1] Gingold R A, Monaghan J J (1997) Smoothed particle hydrodynamics: theory and application to non-spherical stars. Mon Notices Royal Astron Soc 181: 375-389

[2] Lucy L B (1997) A numerical approach to the testing of fusion process. Astron J 88: 1013-1024

[3] Monaghan, J J, Pongracic H (1985) Artificial viscosity for particle methods. Appl Numer Math 1(3): 187-194

[4] Libersky L D, Petschek A G (2005) Smooth particle hydrodynamics with strength of materials, Lecture Notes in Physics. Springer Link, Berlin: 248-257

[5] Xiao Y, Wu H, Ping X (2020) On the Simulation of Fragmentation During the Process of Ceramic Tile Impacted by Blunt Projectile with SPH Method in LS-DYNA. Comput Model Eng Sci 122(3): 923-954

[6] Grimaldia A, Solloa A, Guidab M, Marulo F (2013) Parametric study of a SPH high velocity impact analysis – A birdstrike windshield application. Compos Struct 96: 616–630

[7] Parshikova A N, Medina S A, Loukashenkob I I, Milekhin V A (2000) Improvements in SPH method by means of interparticle contact algorithm and analysis of perforation tests at moderate projectile velocities. Int J Impact Eng 24(8): 779–796

[8] Groenenboom P H, Campbell J, Benítez Montañés L, Siemann M H (2014) Innovative SPH methods for aircraft ditching. Proc of 11th WCCM/5th ECCM, Barcelona, Spain

[9] Patel H M, Vignjevic R Campbell J (2009) An SPH technique for evaluating the behaviour of ships in extreme oceean waves. Int J Marit Eng 151: 39-47

[10] Kawamura K, Hashimoto H, Matsuda A, Terada D (2016) SPH simulation of ship behaviour in severe water-shipping situations. Ocean Eng 120: 220-229

[11] Lin J, Naceur H, Coutellier D, Laksimi A (2014) Efficient meshless SPH method for the numerical modeling of thick shell structures undergoing large deformations. Int J Nonlin Mech 65: 1-13

[12] De Vuyst T, Vignjevic R, Campbell J (2005) Coupling between meshless and finite element methods. Int J Impact Eng 31: 1054–1064.

[13] Bathe K J (2007). Finite Element Procedures, Klaus-Jurgen Bathe

[14] Tran V-X, Samuel Geniaut S (2012) Development and industrial applications of X-FEM axisymmetric model for fracture mechanics. Eng Fract Mech 82: 135-157

[15] Uomoto T, Satoh K, Okada H, Yusa Y (2017) Mesh-independent data point finite element method (MDP-FEM) for large deformation elastic-plastic problems - An application to the problems of diffused necking. Finite Elem Anal Des 136: 18-36

[16] Gui-Rong L (2002) Mesh free methods: moving beyond the finite element method. CRC press

[17] Kojic M, Filipovic N, Stojanovic B, Kojic N (2009) Computer Modeling in Bioengineering: Theoretical Background, Examples and Software. John Wiley & Sons
[18] Shahriari S, Kadem L, Rogers B D, Hassan I (2012) Smoothed particle hydrodynamics method applied to pulsatile flow inside a rigid two-dimensional model of left heart cavity. Int J Numer Meth Biomed Engng 28(11): 1121-1143
[19] Shahriari S, Maleki H, Hassan I, Kadem L (2012) Evaluation of shear stress accumulation on blood components in normal and dysfunctional bileaflet mechanical heart valves using smoothed particle hydrodynamics. J Biomech 45(15): 2637-44
[20] Caballero A, Mao W, Liang L, Oshinski J, Primiano C, McKay R, Kodali S, Sun W (2017) Modeling Left Ventricular Blood Flow Using Smoothed Particle Hydrodynamics. Cardiovasc Eng Technol 8(4): 465-479
[21] Lluch È, De Craene M, Bijnens B, Sermesant M, Noailly J, Camara O, Morales HG (2019) Breaking the state of the heart: meshless model for cardiac mechanics. Biomech Model Mechanobiol 18(6): 1549-1561
[22] Zhang C, Wang J, Rezavand M, Wu D, Hu X (2021) An integrative smoothed particle hydrodynamics method for modeling cardiac function. Comput Methods Appl Mech Eng 381: 113847
[23] Ye T, Phan-Thien N, Lim C T (2016) Particle-based simulations of red blood cells-A review. J Biomech. 49(11): 2255-2266
[24] Polwaththe-Gallage H-N, Saha, S C, Sauret E Flower R, Senadeera W, Gu Y (2016) SPH-DEM approach to numerically simulate the deformation of three-dimensional RBCs in non-uniform capillaries. Biomed Eng Online 15(2): 161
[25] Soleimani M, Sahraee S, Wriggers P (2019) Red blood cell simulation using a coupled shell–fluid analysis purely based on the SPH method. Biomech Model Mechanobiol 18: 347–359
[26] Karimi A, Razaghi R (2018) Interaction of the blood components and plaque in a stenotic coronary artery. Artery Res 24: 47-61
[27] Qin Y, Wu J, Hu Q, Ghista D N, Wong K K (2017) Computational evaluation of smoothed particle hydrodynamics for implementing blood flow modelling through CT reconstructed arteries. J Xray Sci Technol 25(2): 213-232
[28] Al-Saad M, Suarez C A, Obeidat A, Bordas P S, Kulasegaram S (2020) Application of Smooth Particle Hydrodynamics Method for Modelling Blood Flow with Thrombus Formation. Comput Model Eng Sci 122(3): 831–862
[29] Wang F, Xu S, Jiang D, Zhao B, Dong X, Zhou T, Luo X (2021) Particle hydrodynamic simulation of thrombus formation using velocity decay factor. Comput Methods Programs Biomed 207: 106173
[30] De Carvalho A S V J, Bíscaro H H (2019) Blood Flow SPH Simulation with Elastic Deformation of Blood Vessels. Proceedings of IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE) 532-538
[31] Liu M, Zhang Z (2019) Smoothed particle hydrodynamics (SPH) for modeling fluid-structure interactions. Sci China Phys Mech Astron 62: 984701

[32] Lind S J, Benedict R D Peter S K (2020) Review of smoothed particle hydrodynamics: towards converged Lagrangian flow modelling. Proc R Soc A 476: 20190801
[33] Mao W, Li K, Sun W (2016) Fluid-Structure Interaction Study of Transcatheter Aortic Valve Dynamics Using Smoothed Particle Hydrodynamics. Cardiovasc Eng Technol. 7(4): 374-388
[34] Feldman J, Bonet J (2007) Dynamic refinement and boundary contact forces in SPH with applications in fluid flow problems. Int J Numer Meth Engng 72: 295-324
[35] Vacondio R, Rogers B D, Stansby P K, Mignosa P, Feldman J (2013) Variable resolution for SPH: A dynamic particle coalescing and splitting scheme, Comput Methods Appl Mech Eng 256: 132–148
[36] Vacondio R, Altomare C, De Leffe M, Hu X, Le Touzé D, Lind S, Marongiu J-C, Marrone S, Rogers B D, Souto-Iglesias A (2021) Grand challenges for Smoothed Particle Hydrodynamics numerical schemes. Comp Part Mech 8: 575–588
[37] Liu G R, Liu M B (2003) Smoothed Particle Hydrodyndmics a meshfree particle method, World Scientific Publishing, Singapore
[38] Holzapfel G (2001) Nonlinear Solid Mechanics. A Continuum Approach for Engineering. John Wiley & Sons
[39] Fourtakas G, Stansby P K, Rogers B D, Lind S J (2008) An Eulerian–Lagrangian incompressible SPH formulation (ELI-SPH) connected with a sharp interface. Comput Methods Appl Mech Eng 329: 532-552
[40] Djukic T, Saveljic I, Filipovic N (2019) Numerical modeling of the motion of otoconia particles in the patient-specific semicircular canal. Comp Part Mech 6: 767–780
[41] Lind S, Stansby, P K (2016) High-Order Eulerian Incompressible Smoothed Particle Hydrodynamics with Transition to Lagrangian Free-Surface Motion. J Comput Phys 326: 290–311
[42] Campbell J C, Vignjevic R (2012) Simulating structural response to water impact. Int J Impact Eng, 49: 1-10
[43] Liu M B, Liu G R, Lam K Y (2006) Adaptive smoothed particle hydrodynamics for high strain hydrodynamics with material strength. Shock Waves 15: 21–29
[44] Liu M B, Liu GR (2010) Smoothed Particle Hydrodynamics (SPH): an Overview and Recent Developments. Arch Computat Methods Eng 17: 25–76
[45] Nikolic A (2018) Numerical analysis of laminar and turbulent flow in real models of arterial bifurcations with stenosis. Ph.D. dissertation, Faculty of Engineering, University of Kragujevac, Serbia
[46] Lo E Y M, Shao S (2002) Simulation of near-shore solitary wave mechanics by an incompressible SPH method. Appl Ocean Res 24(5): 275-286
[47] Liu X, Lin P, Shao S (2014) An ISPH simulation of coupled structure interaction with free surface flows, J Fluid Struct 48: 46-61
[48] Fragassa C, Topalovic M, Pavlovic A, Vulovic S (2019) Dealing with the Effect of Air in Fluid Structure Interaction by Coupled SPH-FEM Methods Materials 12(7):1162
[49] McDonough J M (2007) Introductory Lectures on Turbulence, Physics, Mathematics and Modeling, Departments of Mechanical Engineering and Mathematics, University of Kentucky, USA

[50] Wilcox D C (1988) Reassessment of the scale-determining equation for advanced turbulence models. AIAA J 26(11): 1299-1310
[51] Wilcox D C (2006) Turbulence Modeling for CFD, 3rd edition, La Canada, CA: DCW Industries
[52] Bassi F, Crivellini A, Rebay S, Savini M (2005) Discontinuous Galerkin solution of the Reynolds-averaged Navier–Stokes and k–ω turbulence model equations. Comput Fluids 34(4–5): 507-540
[53] Bassi F, Ghidoni A, Perbellini A, Rebay S, Crivellini A, Franchina N, Savini M (2014) A high-order Discontinuous Galerkin solver for the incompressible RANS and k–ω turbulence model equations. Comput Fluids 98: 54-68
[54] Morris J P, Fox P J, Zhu Y (1997) Modeling low Reynolds number incompressible flows using SPH. J Comput Phys 136: 214-226
[55] Liang C, Huang J, Shi W (2014) A New Treatment for Boundary of Laminar Flow Inlet or Outlet in SPH. J Softw Eng 8: 321-327
[56] Jonsson P, Andreasson P, Gunnar J, Hellström I, Jonsén P, Lundström T S (2016) Smoothed Particle Hydrodynamic simulation of hydraulic jump using periodic open boundaries. Appl Math Model 40(19–20): 8391-8405
[57] Hou Q, Zhang L X, Tijsseling A S, Kruisbrink A C H (2012) Rapid filling of pipelines with the SPH particle method. Procedia Eng 31: 38-43
[58] Jonsson P, Jonsén P, Andreasson P, Hellström J G I, Lundström T S (2011) Smoothed Particle Hydrodynamics Modeling of Hydraulic Jumps. Proceedings of Particle-Based Methods II—Fundamentals and Applications, Barcelona, 26-28 October, 490-501
[59] Hamid MS (2018) Numerical Simulation Transcatheter Aortic Valve Implantation and Mechanics of Valve Function. Proceedings of 15th International LS-DYNA® Conference & Users Meeting, Dearborn, Michigan, USA June 10-14, 1-8
[60] Hallquist J O (2006) LS-DYNA Theory Manual. Livermore Software Technology Corporation
[61] LSTC (2010) LS-DYNA Keyword User´s Manual Version 971 Rev 5. Livermore: Livermore Software Technology Corporation (LSTC)
[62] Lastiwka M, Basa M Quinlan N J (2009) Permeable and non-reflecting boundary conditions in SPH. Int J Numer Meth Fluids 61: 709-724
[63] Domínguez J M, Fourtakas G, Altomare C, Canelas R B, Tafuni A, García-Feal O, Martínez-Estévez I, Mokos A, Vacondio R, Crespo A J C, Rogers B D, Stansby P K, Gómez-Gesteira M (2021) DualSPHysics: from fluid dynamics to multiphysics problems. Comput Part Mech
[64] Vacondio R, Rogers B, D Stansby P K, Mignosa P (2012) SPH Modeling of Shallow Flow with Open Boundaries for Practical Flood Simulation. J Hydraul Eng 138(6): 530-541
[65] Hosseini S M, Feng J J (2011) Pressure boundary conditions for computing incompressible flows with SPH. J Comput Phys 230(19): 7473-7487
[66] Vignjevic R, De Vuyst T, Campbell J, (2002) The Use of a Homogeneous Repulsive Force for Contact Treatment in SPH. Proceedings of Fifth World Congress on Computational Mechanics WCCM V, Vienna, Austria, July 7-12

[67] Vignjevic R, Campbell J (2009) Review of Development of the Smooth Particle Hydrodynamics (SPH) Method. In: Hiermaier S. (eds) Predictive Modeling of Dynamic Processes. Springer, Boston, USA
[68] Federico I, Marrone S, Colagrossi A, Aristodemo F, Antuono M (2012) Simulating 2D open-channel flows through an SPH model. Eur J Mech B Fluids 34: 35-46
[69] Cao W J, Yang D Z, Lu X W, He Y, Zhou Z Y (2013) Numerical Simulation of Flow and Heat Transfer during Filling Process Based on SPH Method. Adv Mater Res 658: 276–280
[70] Jinlian R, Jie O, Binxin Y, Tao J, Hongyan M (2011) Simulation of container filling process with two inlets by improved smoothed particle hydrodynamics (SPH) method. Int J Comut Fluid Dyn 25(7): 365–386
[71] Holmes D W, Pivonka P (2021) Novel pressure inlet and outlet boundary conditions for Smoothed Particle Hydrodynamics, applied to real problems in porous media flow. J Comput Phys 429: 110029
[72] DualSPHysics team http://dual.sphysics.org. XML GUIDE FOR DUALSPHYSICS
[73] DualSPHysics team http://dual.sphysics.org. XML GUIDE FOR DUALSPHYSICS OPEN BOUNDARY CONDITIONS SPECIAL: INLET/OUTLET
[74] Fraga F C A D, Schuina L L, Porto B S (2020) An Investigation into Neighbouring Search Techniques in Meshfree Particle Methods: An Evaluation of the Neighbour Lists and the Direct Search. Arch Computat Methods Eng 27: 1093–1107
[75] Winkler D, Rezavand M, Rauch W (2018) Neighbour lists for smoothed particle hydrodynamics on GPUs. Comput Phys Commun 225: 140-148
[76] Filipovic N, Milasinovic D, Jagic N, Miloradovic V, Hetterich H, Rieber J (2011) Numerical simulation of the flow field and mass transport pattern within the coronary artery, Comput Methods Biomech Biomed Engin 14(4): 379-388
[77] Filipovic N, Ivanovic M, Kojic M (2009) A comparative numerical study between dissipative particle dynamics and smoothed particle hydrodynamics when applied to simple unsteady flows in microfluidics. Microfluid Nanofluid 7: 227–235
[78] Jovic S, Driver D M (1994) Backward-facing step measurements at low Reynolds number, Re (sub h)= 5000. Tech rep NASA
[79] Creech A, Jackson A, Maddison J, Percival J, Bruce T (2016) Efficient Large Eddy Simulation for the Discontinuous Galerkin Method. arXiv: Fluid Dynamics
[80] Blagojevic M, Nikolic A, Zivkovic M, Zivkovic M, Stankovic G (2014) A novel framework for fluid/structure interaction in rapid subject-specific simulations of blood flow in coronary artery bifurcation. Vojnosanit Pregl 71(3): 285-292
[81] Topalovic M, Blagojevic M, Nikolic A, Zivkovic M, Filipovic N (2015) Application of smoothed particle hydrodynamics in biomechanics: Advanced procedure for discretization of complex biological shapes into pseudo-particles. Proceedings of IEEE 15th International Conference on Bioinformatics and Bioengineering (BIBE), Belgrade, Serbia, November 2-4, 1-4
[82] Müller M, Schirm S, Teschner M (2004) Interactive blood simulation for virtual surgery based on smoothed particle hydrodynamics. Technol Health Care 12(1): 25-31

[83] Hieber SE, Walther JH, Koumoutsakos P (2004) Remeshed smoothed particle hydrodynamics simulation of the mechanical behavior of human organs. Technol Health Care 12(4): 305-14
[84] Tanaka N, Hayakawa Y, Masuzawa T (2006) Three-dimensional simulations of microscopic blood flow using SPH method. J Biomech 39(S1): S430
[85] Shahriari S, Kadem L (2018) Chapter 11 - Smoothed Particle Hydrodynamics Method and Its Applications to Cardiovascular Flow Modeling, Numerical Methods and Advanced Simulation in Biomechanics and Biological Processes, Academic Press, 203-219
[86] Ponzini R, Vergara C, Rizzo G, Veneziani A, Roghi A, Vanzulli A, Parodi O Redaell A (2010), Womersley Number-Based Estimates of Blood Flow Rate in Doppler Analysis: In Vivo Validation by Means of Phase-Contrast MRI. IEEE Trans Biomed Eng 57(7): 1807-1814
[87] Egorova M, Dyachkov S, Parshikov A, Zhakhovsky A (2019), Parallel SPH modeling using dynamic domain decomposition and load balancing displacement of Voronoi subdomains. Comput Phys Commun 234: 112-125
[88] Violeau D, Leroy A (2014) On the maximum time step in weakly compressible SPH. J Comput Phys 256: 388-415
[89] Moyle K R, Antiga L, Steinman D A (2006) Inlet conditions for image-based CFD models of the carotid bifurcation: is it reasonable to assume fully developed flow? J Biomech Eng 128(3): 371-379.

Figures:

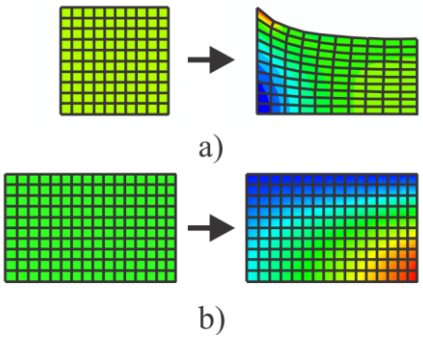

Fig. 1 a) Larangian b) Eulerian formulation

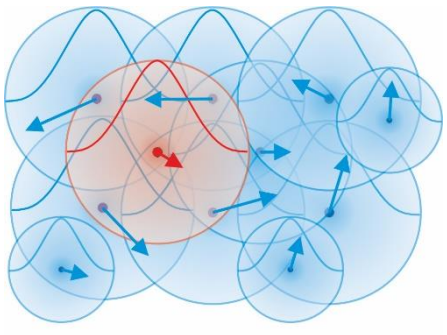

Fig. 2 SPH Turbulent fluid flow

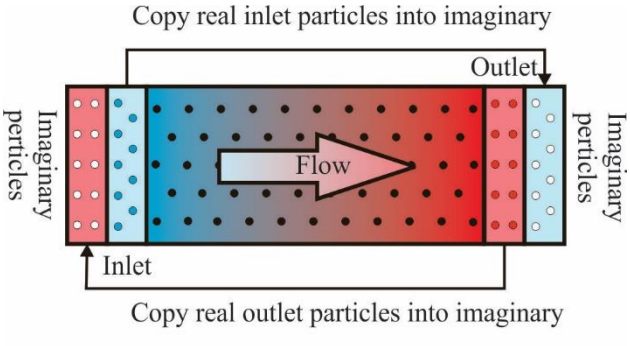

Fig. 3 Periodic boundary conditions

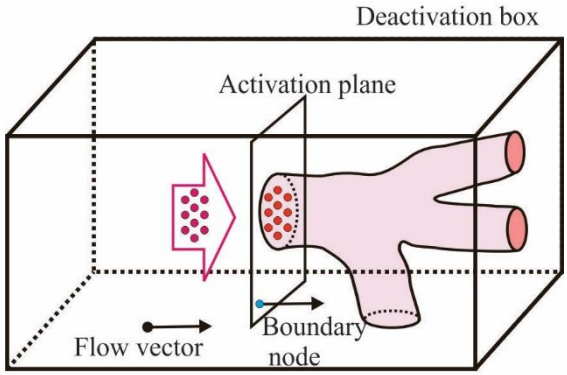

Fig. 4 LS-Dyna SPH flow generation

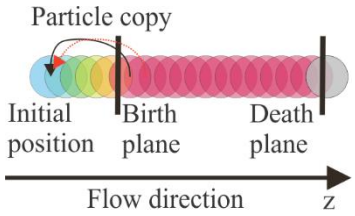

Fig. 5 Initial SPH generation and deletion solution

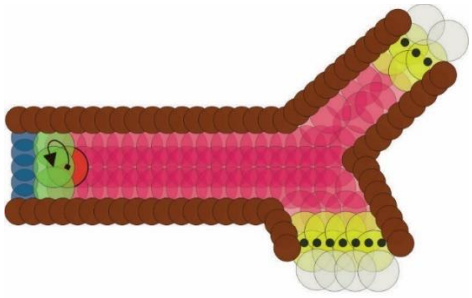

Fig. 6 Improved SPH generation and deletion

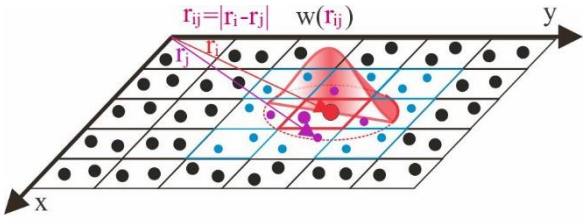

Fig. 7 Linked list subdomains

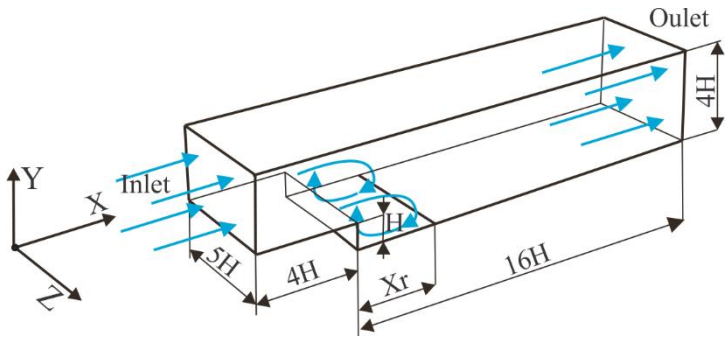

Fig. 8 Backward facing step problem geometry

Fig. 9 FEM results (VX velocity) from PAK-F using *k-ω* model

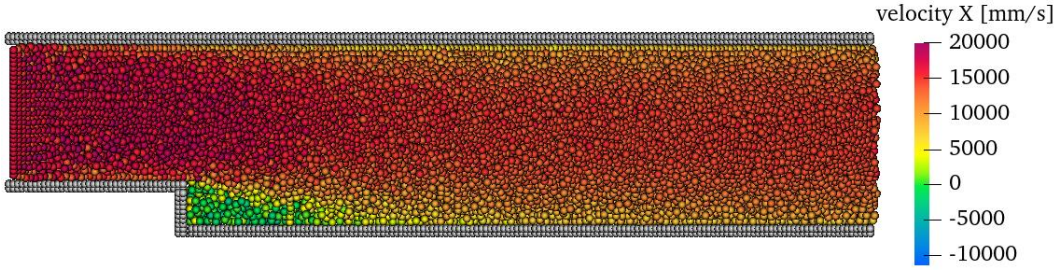

Fig. 10 SPH results (VX velocity) from SPH07

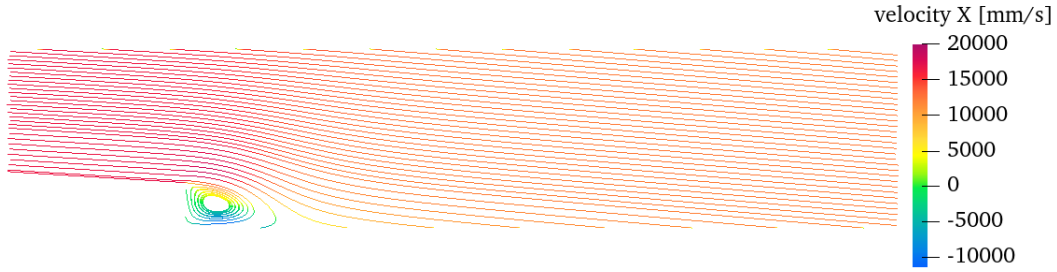

Fig. 11 FEM results (streamlines) from PAK-F using *k-ω* model

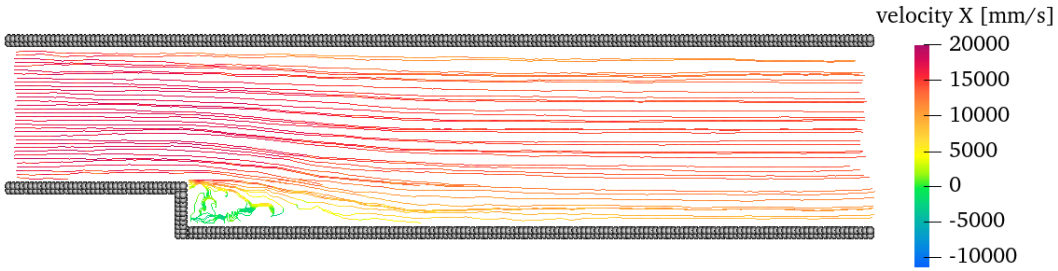

Fig. 12 SPH results (streamlines) from SPH07

Fig. 13 FEM results (differential pressure) from PAK-F using *k-ω* model

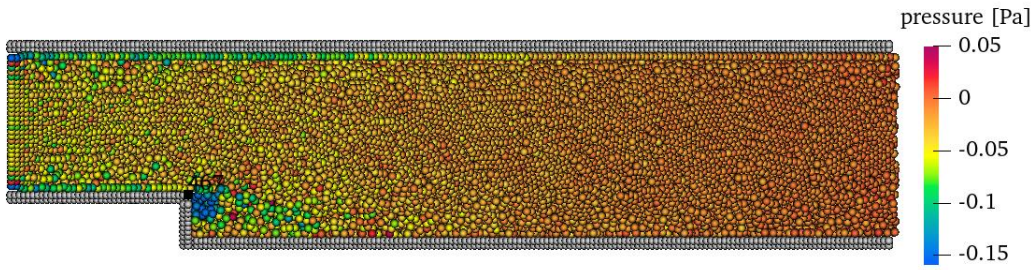

Fig. 14 SPH results (differential pressure) from SPH07

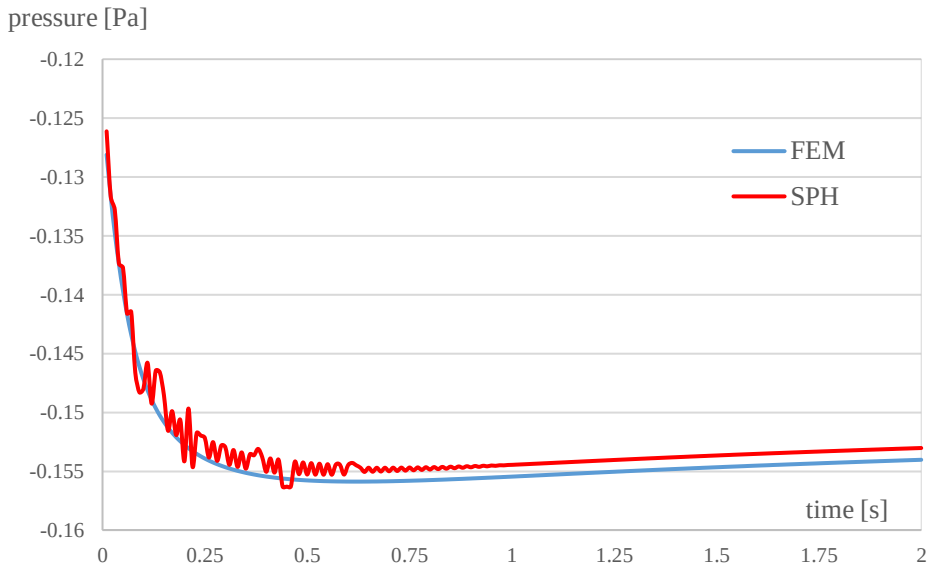

Fig. 15 Differential pressure at step edge

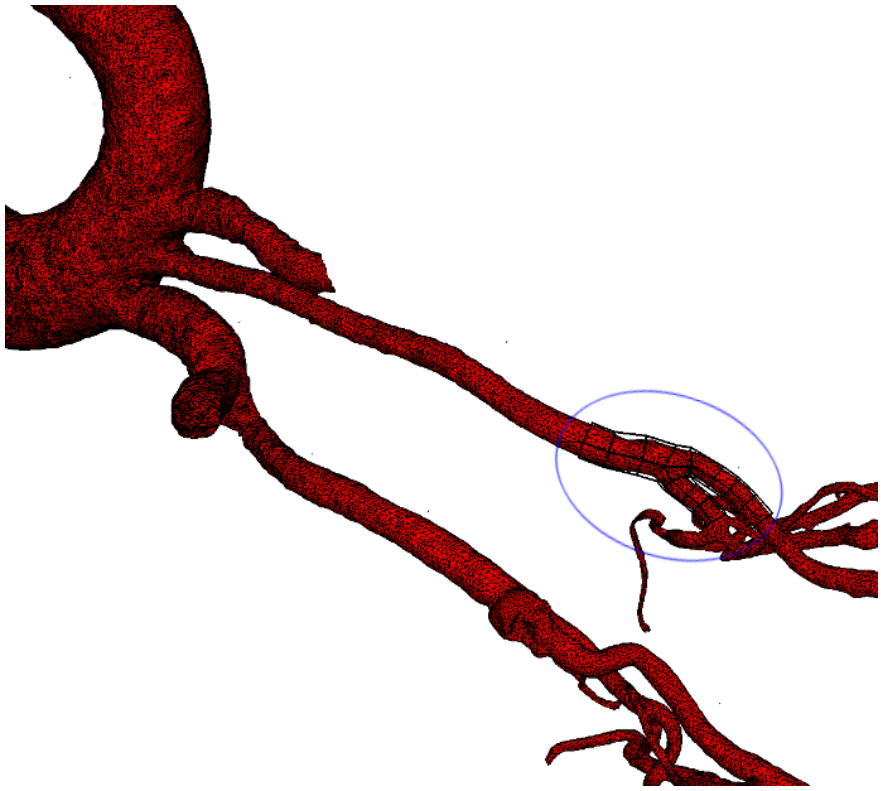

Fig. 16 Scanned carotid geometry

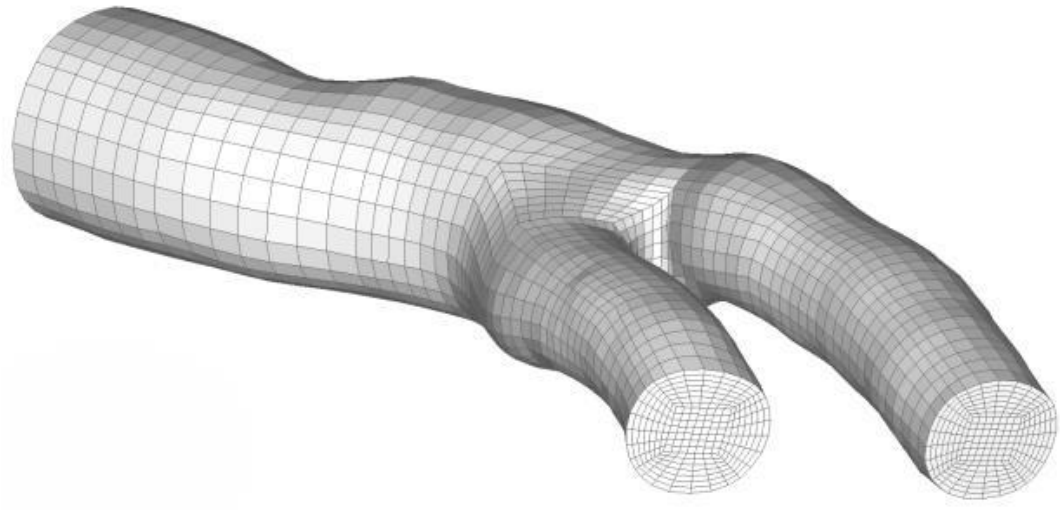

Fig. 17 FEM mesh of carotid bifurcation obtained from STL2FEM software

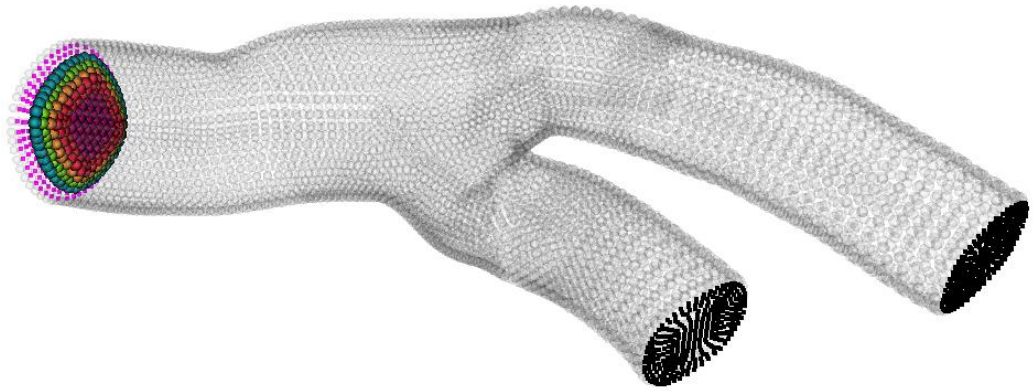

Fig. 18 SPH model of carotid bifurcation: empty vessel with mother, newborn and killer particles

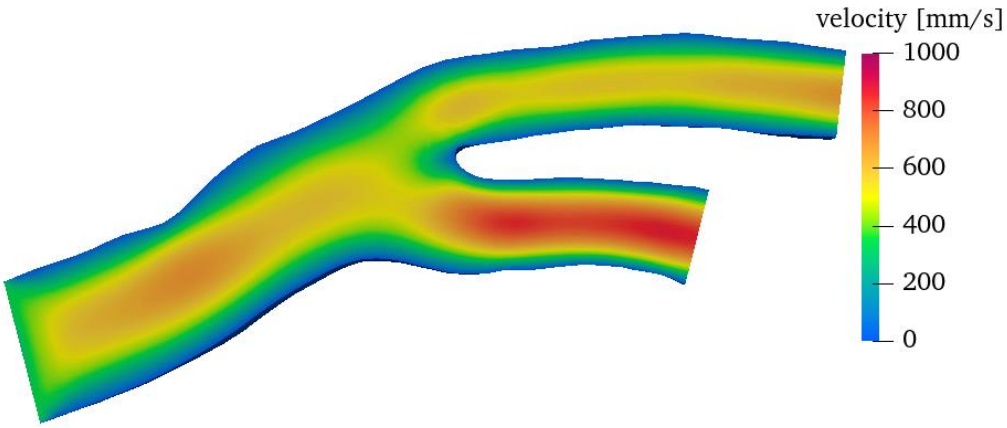

Fig. 19 FEM results: Velocity magnitude during systole

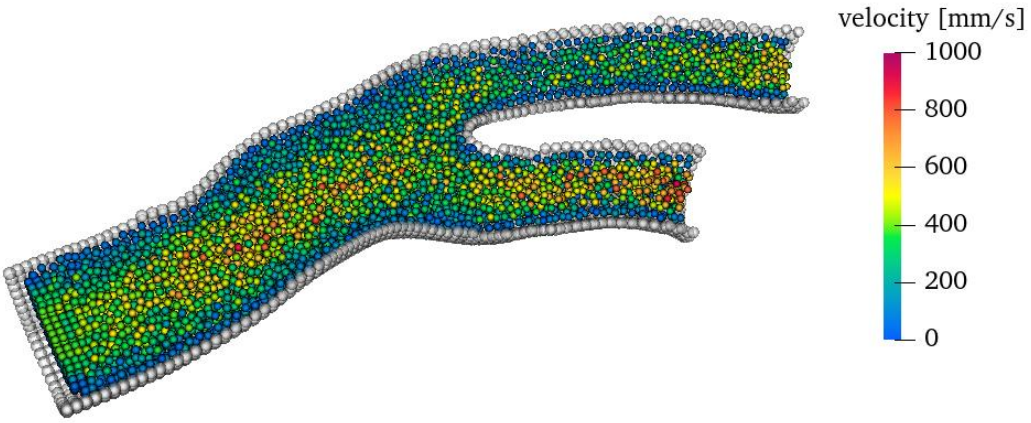

Fig. 20 SPH results: Velocity magnitude during systole

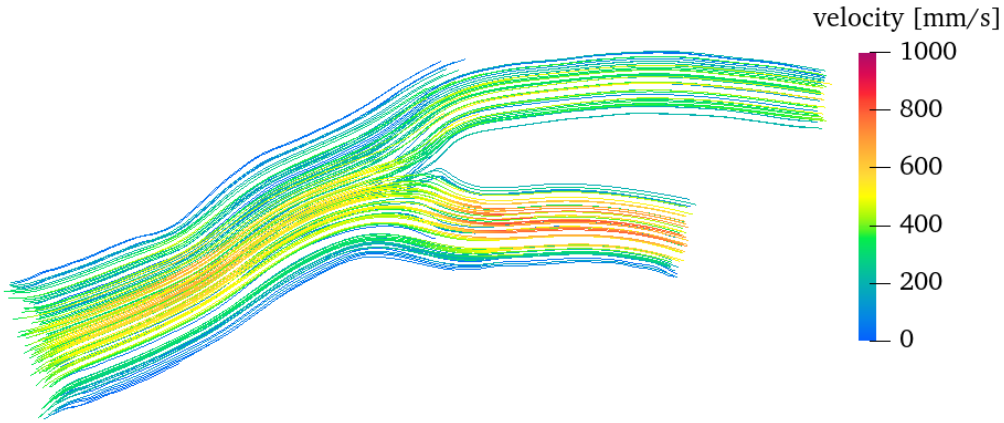

Fig. 21 FEM streamlines: Velocity magnitude during systole

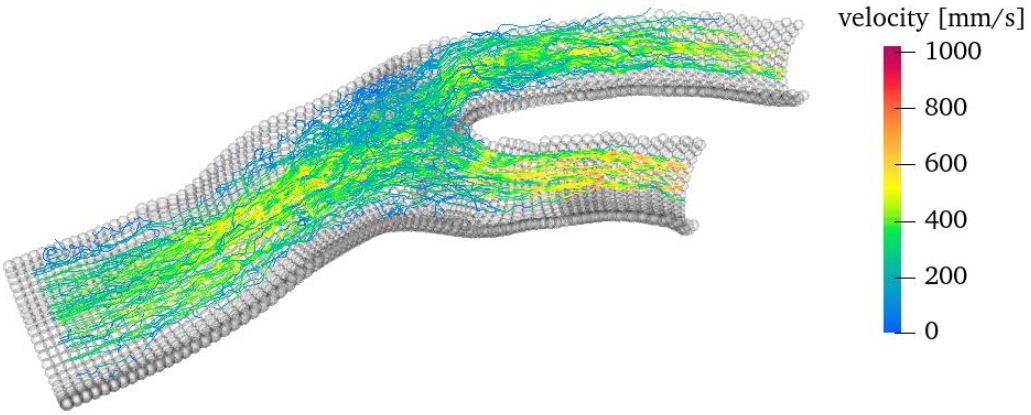

Fig. 22 SPH streamlines: Velocity magnitude during systole

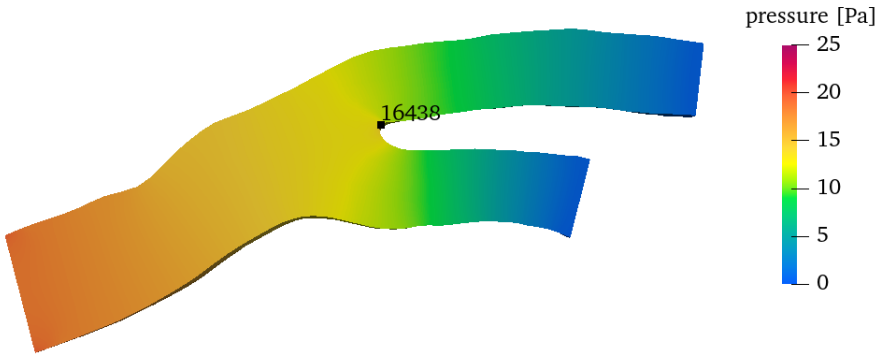

Fig. 23 Differential pressure field in FEM during systole

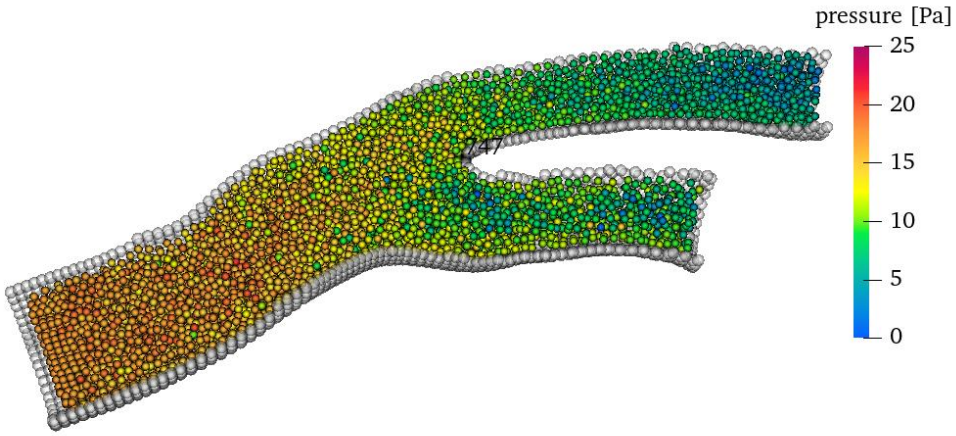

Fig. 24 Differential pressure field in SPH during systole

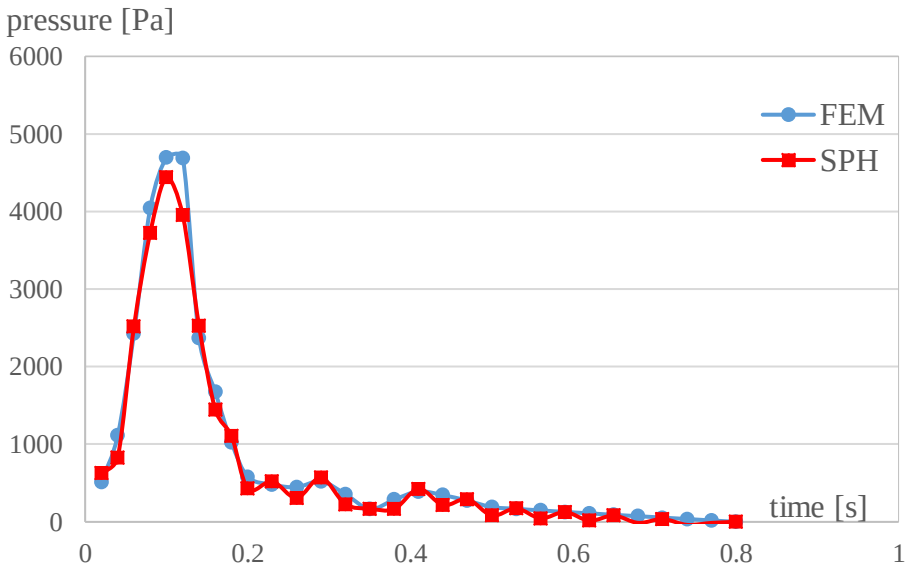

Fig. 25 Total pressure at bifurcation