WAIT: Workshop on the Advances in Information Technology, Budapest, 2018

Aortic Root Simulation Using Smoothed Particle Hydrodynamics

Márton Tóth[1], Tamás Umenhoffer[1], László Szécsi[1], Ágota Kacsó[1] and Balázs Benyó[1]

[1] Department of Control Engineering and Information Technology,
Budapest University of Technology and Economics, Budapest, Hungary

Abstract

This paper presents a simulation engine that can model multiple physical phenomena including fluid dynamics, rigid body movement and elastic transformations. All of these behaviors are modeled in a single framework using the Smoothed Particle Hydrodynamics method. The purposed modeling task is the simulation of the aortic root including the aortic valve. Such simulation task usually requires the usage of different modeling tools but the presented framework offers an integrated solution. The capabilities of the presented software is illustrated by simple test cases and finally the simulation of the integrated aortic root model is presented.

Categories and Subject Descriptors (according to ACM CCS): I.6.3 [Simulation and Modeling]: Applications

1. Introduction

In case of aortic root enlargement or aortic root aneurysm, the surgical replacement of the aortic root can be necessary. This replacement usually means the replacement of the aortic valve with a synthetic one, but in cases when the patient's own valves are not diseased the original valves can be sewed back in the graft. During the surgery the valves are sewed back in predefined positions which are not necessary the same as the patients original valve positions.

Recently special tools have been developed to measure the patient's original valve positions and it has became possible to sew back the valves to their original location. However, there is no data available how the valves operate in the graft if not the predefined positions are used.

Our long-range goal is to define a realistic model of the aortic root and the aortic valve which can be used to validate the new sewing idea. This simulation task includes the modeling of several physical phenomena which are hard to implement in a single framework. However, the Smoothed Particle Hydrodynamics (SPH) offers a common platform to evaluate the necessary calculations.

2. Previous Work

Modeling of the cardiac flow using computational fluid dynamics (CFD) methods is an active field of research. Most of the applied techniques can be categorized into two main groups, mesh-based and mesh-free methods. Most of the current simulations[2, 3] based on the conventional mesh-based approaches[4]. These methods have demonstrated their abilities to model cardiovascular flows but they have their limitations. Complex geometries, large scale deformations of the boundaries or moving interfaces still remain major challenges for mesh-based approaches[1].

To overcome these limitations, mesh-free methods have been developed and adapted to this kind of problems. Mesh-free methods are based on the Lagrangian approach, therefore the continuous medium is discretized as a set of particles over the modeled spatial domain.

Smoothed Particle Hydrodynamics[5, 6], as a completely mesh-free method, has been introduced for astrophysical calculations and later has been extended to model various physical phenomena.

As SPH can handle complex geometric motions, it is an attractive tool for modeling fluid-structure interaction (FSI). This property of the method has been already utilized in several works, investigating the blood flow in the human heart [1, 4, 7].

In this paper we further investigate the capabilities of the SPH approach and we introduce a complex framework for

the simulation of the blood flow in the aortic root, including the modeling of the aortic valve.

3. The SPH formalism

3.1. Fluid dynamics

The incompressible, isothermal Navier-Stokes equations in Lagrangian coordinates can be expressed as:

$$\frac{d\rho_i}{dt} = 0, \tag{1}$$

$$\frac{d\mathbf{v}_i}{dt} = -\frac{1}{\rho_i}\nabla p_i + \nu\nabla^2\mathbf{v}_i + \frac{\mathbf{F}_i^{sum}}{m_i}, \tag{2}$$

where $\mathbf{v}_i$ is the velocity of the particle, ρ_i is its density, p_i is the pressure, m_i is the mass. Beside of these properties a particle also has a position $\mathbf{x}_i$ and a volume that can be expressed as $V_i = m_i/\rho_i$.

As SPH is a mesh-free approach, the continuous fluid is discretized by a set of particles. Each particle has its own physical properties that can be interpolated at any location using the properties of the neighboring particles. To interpolate a physical property A at the position $\mathbf{x}$ in the domain, the following equation shall be evaluated:

$$A(\mathbf{x}) = \int A(\mathbf{x}')W(\mathbf{x}-\mathbf{x}',h)d\mathbf{x}, \tag{3}$$

where h is the smoothing length, which determines the size of the neighborhood domain, and $W(\mathbf{x}-\mathbf{x}',h)$ is a weighting function also called as interpolating kernel.

Integral (3) is approximated by a summation over the particles:

$$A_i = \sum_j V_j A_j W_{ij} = \sum_j \frac{m_j}{\rho_j} A_j W_{ij}, \tag{4}$$

where W_{ij} is a shorter form of $W(\mathbf{x}_i - \mathbf{x}_j, h)$. In this work a cubic kernel function was used:

$$W(q) = \alpha \cdot \begin{cases} \frac{2}{3} - q^2 + \frac{1}{2}q^3 & \text{if } 0 \le q < 1 \\ \frac{1}{6}(2-q)^3 & \text{if } 1 \le q < 2 \\ 0 & \text{if } 2 \le q \end{cases}, \tag{5}$$

where $q = \left\|\mathbf{x}_i - \mathbf{x}_j\right\|/h$ and α is a normalization parameter, in 3D $\alpha = 3/(2\pi h^3)$. This kernel should also satisfy the following requirements:

$$\int_\Omega W(\mathbf{x}-\mathbf{x}',h)d\mathbf{x}' = 1, \tag{6}$$

$$\int_\Omega W(\mathbf{x}-\mathbf{x}',h)d\mathbf{x}' = \int_\Omega W(\mathbf{x}'-\mathbf{x},h)d\mathbf{x}', \tag{7}$$

$$\int_\Omega W(\mathbf{x}-\mathbf{x}',h)d\mathbf{x}' \ge 0, \tag{8}$$

Equation (6) states that the integral of the kernel should be 1, (7) is the symmetric property and by (8) the compact kernel support is ensured. Beside of these properties the kernel function should be Dirac-delta like function and shall be differentiable.

In this work derivatives and second derivatives are approximated as:

$$\nabla A_i = \rho_i \sum_j m_j \left(\frac{A_i}{\rho_i^2} + \frac{A_j}{\rho_j^2}\right)\nabla\mathbf{W}_{ij}, \tag{9}$$

$$\nabla^2 A_i = 2\sum_j \frac{m_j}{\rho_j} A_{ij} \frac{\mathbf{x}_{ij}\cdot\nabla\mathbf{W}_{ij}}{\mathbf{x}_{ij}\cdot\mathbf{x}_{ij} + 0.01h^2}, \tag{10}$$

where $\nabla\mathbf{W}_{ij} = \partial W(q)/\partial q \cdot \mathbf{x}_{ij}/(\left\|\mathbf{x}_{ij}\right\| h)$ is the gradient of the kernel function.

To express the density of a particle, equation (4) can be used as:

$$\rho_i = \sum_j m_j W_{ij}. \tag{11}$$

However, this approach introduces numerical errors near to the fluid boundaries and therefore the usage of an other solution is advised:

$$\frac{d\rho_i}{dt} = -\sum_j m_j \mathbf{v}_{ij}\nabla\mathbf{W}_{ij}. \tag{12}$$

As the density of a particle is known, the pressure at the particles position can be calculated according to the weakly compressible SPH method:

$$p_i = k\left(\left(\frac{\rho_i}{\rho_0}\right)^7 - 1\right), \tag{13}$$

With these equations in mind, an iterative SPH simulation of fluids can be calculated as stated in Algorithm 1:

Algorithm 1 SPH algorithm

for all particle i **do**
  find neighbors j
end for
for all particle i **do**
  $\rho_i = \sum_j m_j W_{ij}$
  compute p_i from ρ_i
end for
for all particle i **do**
  $F_i^{pressure} = -\frac{m_i}{\rho_i}\nabla p_i$
  $F_i^{viscosity} = m_i\nu\nabla^2\mathbf{v}_i$
  $F_i^{external} = m_i\mathbf{g}$
  $F_i(t) = F_i^{pressure} + F_i^{viscosity} + F_i^{external}$
end for
for all particle i **do**
  $\mathbf{v}_i(t+\Delta t) = \mathbf{v}_i(t) + \Delta t\mathbf{F}_i(t)/m_i$
  $\mathbf{x}_i(t+\Delta t) = \mathbf{x}_i(t) + \Delta t\mathbf{v}_i(t+\Delta t)$
end for

3.2. Flexible materials

In the original SPH method the kernel, that defines the interaction neighborhood travels with the particles. Thus, the interaction neighborhood is redefined for every new position the particle attains, so particles can move through the kernel domain. For fluid simulations this effect is desired, but in case of solid mechanics it leads to the tensile instability[8]. Therefore, formulations were developed, which defines the neighborhood in a fixed reference configuration. Usually the initial particle configuration defines the neighborhood. This particular flavor of SPH is called as the Total Lagrangian SPH (TL-SPH).

To calculate the internal forces of a solid object, subject to deformation, one should determine the deformation gradient then calculate the stress tensor as a function of the deformation gradient and finally the stress tensor shall be transformed into nodal forces, that directly act on the particles[9].

The deformation gradient can be expressed as:

$$\mathbf{F}_i = \sum_j V_j^0 (\mathbf{u}_j - \mathbf{u}_i) \otimes \nabla W(\mathbf{X}_i - \mathbf{X}_j, h) + \mathbf{I}, \tag{14}$$

where $\mathbf{X}_i$ is the particles position in the reference configuration, $\mathbf{u}_i = \mathbf{x}_i - \mathbf{X}_i$.

In the TL-SPH the natural strain measure is the Green-Lagrange strain,

$$\mathbf{E} = \frac{1}{2}(\mathbf{C} - \mathbf{I}), \tag{15}$$

where $\mathbf{C} = \mathbf{F}^T\mathbf{F}$ is the right Cauchy-Green deformation tensor which describes the change of a line element in the reference configuration. The proper stress measure is the Second Piola-Kirchhoff stress $\mathbf{S}$. In the TL-SPH model the linear elasticity model is given in terms of the Lamé parameters λ and μ as

$$\mathbf{S} = \lambda Tr(\mathbf{E}) + 2\mu\mathbf{E}. \tag{16}$$

The Second Piola-Kirchhoff stress tensor applies to the reference configuration, however, nodal forces act in the current configuration. Therefore, a stress measure is required to map the stress from the reference configuration to the current configuration and this is the First Piola-Kirchhoff stress:

$$\mathbf{P} = \mathbf{FS}. \tag{17}$$

The nodal forces are obtained from the SPH approximation of the stress divergence as:

$$\mathbf{f}_i = \sum_j V_i^0 V_j^0 (\mathbf{P}_j + \mathbf{P}_i) \nabla W_i(X_{ij}). \tag{18}$$

3.3. Rigid bodies

In this work the rigid bodies were modeled as a composition of particles. These particles can interact with the other particles as the SPH method requires, but their movement is determined not only by the individual forces, as the particles of a rigid body have to move in a coupled way.

A description about particle based rigid body movement simulation can be found in the literature[10 11] and can be summarized in equations (19) - (22).

$$\frac{dP}{dt} = \sum_i F_i, \tag{19}$$

$$\frac{dL}{dt} = \sum_i (r_i - r_{cog}) \times F_i, \tag{20}$$

$$\omega = I(t)^{-1} L, \tag{21}$$

$$I(t)^{-1} = R(t) I(0)^{-1} R(t)^T. \tag{22}$$

In Eequation (19), P stands for the linear momentum of the rigid body. Its first derivative can be calculated as the sum of the forces exerted on the particles composing the rigid body. Based on the change of the linear momentum, the new velocity and position of the body can be calculated. This part is very similar to the motion of a single particle.

Rigid bodies, however, have orientation and angular momentum as well. Equation (20) expresses the change of the angular momentum L according to the forces exerted on the body. The angular velocity ω can be derived from L as equation (21) describes. Here I stand for the inertia tensor at time t. The current value of the inertia tensor can be calculated form the initial inertia tensor and the current rotation matrix.

To have interactions between rigid bodies a collision detection method has to be applied. To find colliding particle pairs the neighborhood search algorithm of SPH method can be used. To solve the collision reaction the following equations (23)-(25) are used. This model has been used by Mishra[12] for simulating granular materials.

$$F_{i,s} = -k(d - |r_{ij}|) \frac{r_{ij}}{|r_{ij}|}, \tag{23}$$

$$F_{i,d} = \eta v_{ij}, \tag{24}$$

$$F_{i,s} = k v_{ij,t}. \tag{25}$$

Equation (23) represents a repulsive force modeled by a linear spring. Equation (24) is a damping force, which dissipates energy between particles and equation (25) is shear force proportional to the relative tangential velocity.

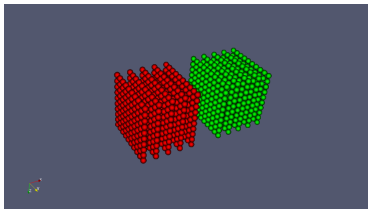
With the equations above one can calculate the movements of a rigid body composed by particles. This model also enables the interaction between fluid and rigid body particles.

4. Results

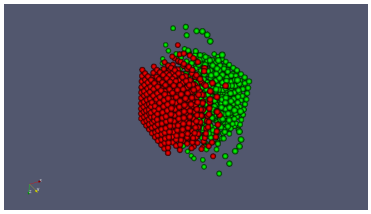
As it can be seen from the equations defined in Section 3, the particle model can handle the simulation of fluids, rigid

bodies and flexible materials in a common platform. Most of the required calculations can be performed independently on a per particle basis. This can be utilized in the implementation of the simulation engine. As the number of particles is high, it worth designing the algorithm on a massively parallel architecture, like the GPU[13].

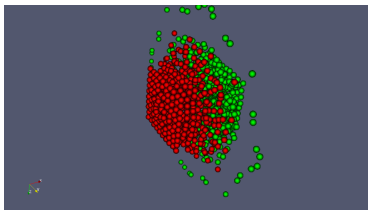
To validate the implemented system simple test cases were defined. In these cases the interaction between different types of materials were examined. In the first test two fluid cube instances were collided. The simulation results can be seen in Figure 1.



(a) Initial position.



(b) Begin of the collision.



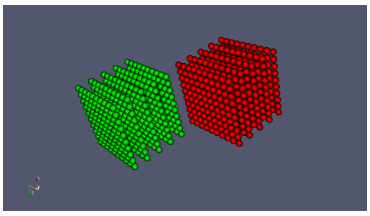
(c) End of the simulation.

Figure 1: *Collision of fluid cubes.*

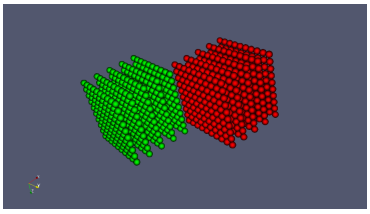
The second test case included two rigid bodies that were moving toward each other. The collision happened off-axis and therefore torque acted on the bodies. The results are shown in Figure 2.

The third test case defined a rigid and a fluid entity. The two entities were moving toward each other. The simulation illustrated as the fluid particles splashed into the rigid object. The results are illustrated in Figure 3.

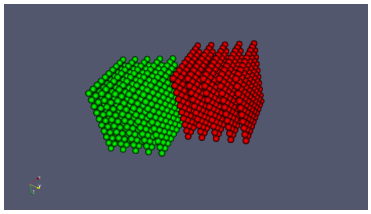
The fourth test included a flexible pole and a rigid object. One end of the pole was fixed. As the rigid body hit the other end, a wave started to spread through the pole and it reflected at the fixed end. The steps of the simulation can be seen in Figure 4.



(a) Initial position.



(b) Begin of the collision.



(c) End of the simulation.

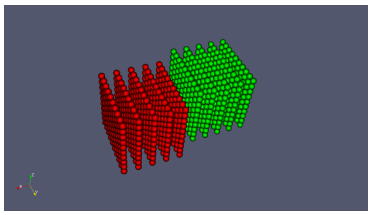
Figure 2: *Collision of two rigid bodies.*

The last test case show in this paper includes all the previously described materials and models the aortic root, including the aortic valve. In this case the aortic root was filled with fluid. The particles below the aortic valve were configured to have initial velocity to move through the valve. The valve itself was defined as a flexible object and the wall of the aortic root was modeled as fix, non-moving object. The flow is illustrated in Figure 5 and the movements of the aortic valve are highlighted in Figure 6.

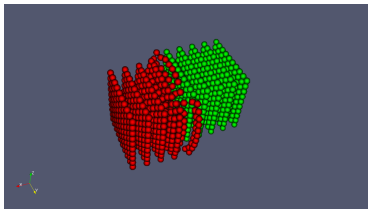
5. Conclusions

As shown in Section 4 the implemented system can model multiple material types and can handle complex physical behavior. The test cases showed that the simulation behaves as it is expected, however, it was not quantitatively tested. In the future we plan to add numerical validation to the tests.

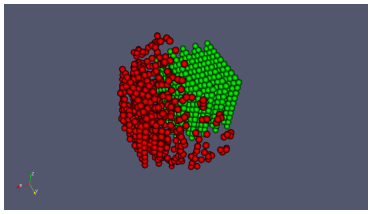
In this scenario, it was shown that the SPH method has the ability, on its own, to model the required physical phenomena to simulate the aortic root. It does not require multiple platforms and methods therefore it offers an effective standalone solution for the modeling of complex physical systems.



(a) Initial position.

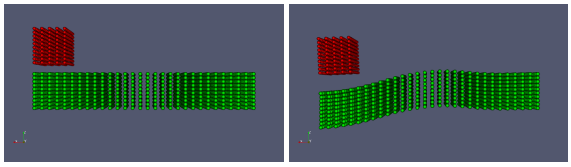


(b) Begin of the collision.

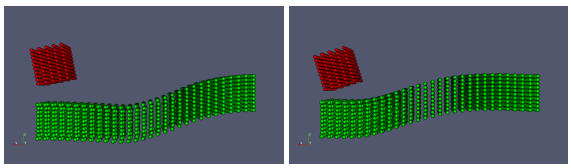


(c) End of the simulation.

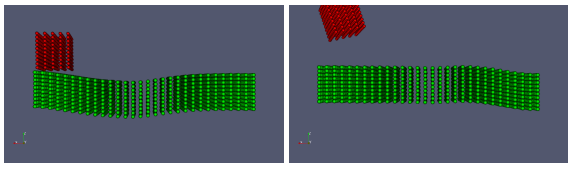
Figure 3: *Collision of fluid and rigid entities.*



(a) (b)

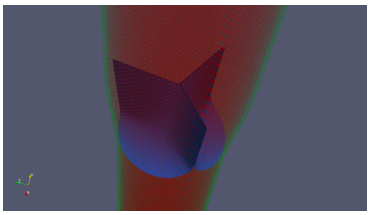


(c) (d)

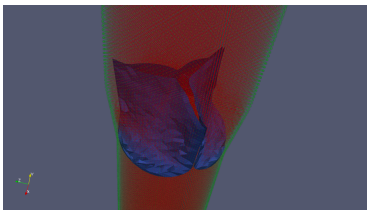


(e) (f)

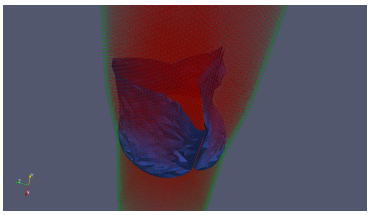
Figure 4: *Flexible object.*



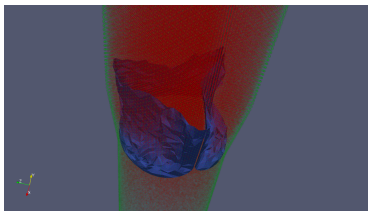
(a)



(b)

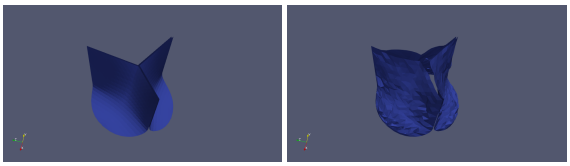


(c)

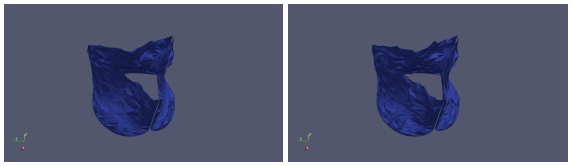


(d)

Figure 5: *Simulation of the aortic valve.*



(a) (b)



(c) (d)

Figure 6: *Movements of the aortic valve.*

Acknowledgements

This work has been supported by VKSZ-14 SCOPIA project.

References

1. A. Caballero et al. Modeling Left Ventricular Blood Flow Using Smoothed Particle Hydrodynamics *Cardiovascular Engineering and Technology*, **8**(4):465–479, 2017.
2. Chnafa, C. and Mendez, S. and Nicoud, F. Image-based large-eddy simulation in a realistic left heart *Computers and Fluids*, **94**:173–187, 2014.
3. Khalafvand, Seyed Saeid and Ng, Eddie Yin-Kwee and Zhong, Liang and Hung, Tin-Kan Three-dimensional diastolic blood flow in the left ventricle *Journal of Biomechanics*, **50**:71–76, 2017.
4. Shahriari, S. and Kaden, L- and Rogers, B. D. and Hassan, I. Smoothed particle hydrodynamics method applied to pulsatile flow inside a rigid two-dimensional model of left heart *International Journal for Numerical Methods in Biomedical Engineering*, **28**(1):1121–1143, 2012.
5. Monaghan J. J. Smoothed Particle Hydrodynamics *Annual Review of Astronomy and Astrophysics*, **30**:543–574, 1992.
6. Damien Violeau Fluid Mechanics and the SPH Method: Theory and Applications *Oxford University Press*, ISBN: 9780199655526, 2012.
7. Kulp, Scott and Gao, Mingchen and Zhang, Shaoting and Qian, Zhen and Voros, Szilard and Metaxas, Dimitris and Axel, Leon Practical patient-specific cardiac blood flow simulations using SPH *Proceedings - International Symposium on Biomedical Imaging*:832–835, 2013.
8. Swegle, J.W. and Hicks, D.L. and Attaway, S.W. Smoothed Particle Hydrodynamics Stability Analysis *Journal of Computational Physics*, **116**(1):123–134, 1995.
9. Ganzenmüller, Georg C. An hourglass control algorithm for Lagrangian Smooth Particle Hydrodynamics *Computer Methods in Applied Mechanics and Engineering*, **286**:87–106, 2015.
10. Nguyen, Hubert (Editor) GPU Gems 3 Publisher: Addison-Wesley Professional, 2007
11. Szirmay-Kalos, László and Antal, György and Csonka, Ferenc Háromdimenziós grafika, animáció és játékfejlesztés , ComputerBooks, Budapest, 2003.
12. B. Mishra A review of computer simulation of tumbling mills by the discrete element method: Part I-contact mechanics *International Journal of Mineral Processing*, **4** (71), p. 73 - 93, 2003.
13. L. Szirmay-Kalos and L. Szécsi General Purpose Computing on Graphics Processing Units *Algorithms of Informatics*, p. 1451–1495, 2010