

On Efficiency of Parallel Solvers for the Blood Flow through Aortic Valve

Vadimas Starikovičius^a, Arnas Kačeniauskas^a, Algirdas Maknickas^a, Eugenius Stupak^a, Ruslan Pacevič^a, Miglė Staškūnienė^a and Giedrius Davidavičius^b

^a *Vilnius Gediminas Technical University*

Saulėtekio al. 11, LT-10223 Vilnius, Lithuania

^b *Centre of Cardiology and Angiology, Vilnius University Hospital “Santaros Klinikos”*

Santariškių Str. 2, LT-08406, Vilnius, Lithuania

E-mail(*corresp.*): vadimas.starikovicius@vgtu.lt

E-mail: arnas.kaceniauskas@vgtu.lt

E-mail: algirdas.maknickas@vgtu.lt

E-mail: eugenius.stupak@vgtu.lt

E-mail: ruslan.pacevic@vgtu.lt

E-mail: migle.staskuniene@vgtu.lt

E-mail: giedrius.davidavicius@santa.lt

Received January 7, 2017; revised May 26, 2017; published online September 15, 2017

Abstract. Mathematical modelling of cardiac haemodynamics presents a great challenge to the computational scientists due to numerous numerical issues and required computational resources. In this paper, we study the parallel performance of 3D simulation software for the blood flow through the aortic valve. The fluid flow problem with the open aortic valve leaflets is formulated and solved in parallel. The choice between the segregated and coupled numerical schemes is discussed and investigated. We present and compare the parallel performance results of both types of parallel solvers. We investigate their strong and weak scalability.

Keywords: parallel efficiency, parallel algorithms, Reynolds-averaged Navier–Stokes equations, segregated and coupled schemes, flow through aortic valve.

AMS Subject Classification: 76Z05; 35Q30; 65Y05.

1 Introduction

Nowadays, application of parallel computing technologies in the numerical solution of any computationally challenging problem is essential. However, fast and efficient parallel computations require application of the appropriate parallel

algorithms. The permanent development of the parallel computing technologies with all their diversity requires a constant attention, which must be paid to a development and selection of proper parallel algorithms for the solution of various problems.

For the problems described by the systems of partial differential equations (PDEs), one needs to select the efficient combination of the domain decomposition method, numerical discretization scheme with the appropriate mesh generator, numerical solution method for the systems of linear equations with scalable preconditioner [4]. Unsuccessful choices of these methods or even their parameters often lead to the degraded performance of the resulting parallel solution algorithm. Here, we remind the well-known fact that combination of methods producing the quickest sequential solution time usually does not show the best possible parallel scalability and efficiency [36].

Obviously, the selection of the best suitable numerical methods and construction of the efficient parallel solution algorithm are highly dependable on the specifics of the considered problem [16,34,37]. Studies of parallel scalability of different applications are regularly appearing in the scientific literature [9,26]. Continuous attention to this field is caused by the constant development of parallel computing technologies, appearance of the multicore processors, specialized GPUs (Graphics Processing Units), accelerators and coprocessors [1,27]. New algorithms and software tools are developed seeking to meet the demands of petascale and exascale computing [10,33].

In this paper, we consider the application of parallel computing for mathematical modeling of blood flow through the aortic valve [23,32]. This problem causes great computational challenges due to the complex flow in complicated 3D geometries [24]. Its practical importance is obvious. In spite of continuous advancements in medicine, cardiovascular diseases are one of the leading causes of death in the world [24]. Therefore, great scientific effort is put into understanding of heart structure and function. The heart is a complex system governed by haemodynamics [20,28], structural dynamics and electromagnetics [23,39]. In spite of significant advances in imaging modalities for studying cardiac haemodynamics, present-day in vivo measurement techniques can only resolve the large scale blood flow features [22,24].

Computational fluid dynamics (CFD) can be used as a simulation tool to augment the information obtained by clinicians from the traditional diagnostic modalities, including echocardiography, computed tomography (CT) and magnetic resonance imaging (MRI). In theory, using a CFD, a computational representation of a vascular region prone to disease can be built for each patient. However, in practice, obtaining the aortic valve flow patterns at physiologic conditions and scales sufficiently fine for patient-specific haemodynamics is still a major research challenge [22]. Nevertheless, in recent years, there have been growing efforts and successes in 3D CFD simulations and high resolution 3D measurements of the blood flow and the heart motion [23,25]. Various techniques have been devised to obtain the detailed information on time-dependent aortic valve flow patterns [32]. For example, a framework for the simulation and visualization of blood flow through the aortic valve was developed by Kulp et al. [20]. Healthy and diseased aortic valves were simulated by using the

contrast-enhanced computed tomography, the full R-R animated model and the highly-accurate Navier-Stokes solver on a staggered grids. However, to the best of our knowledge, in such studies the issues of parallel performance of numerical solution algorithms were not considered.

In this work, we study the parallel performance of 3D simulation software for the blood flow through the aortic valve [18]. It is built upon the ANSYS Fluent package [2]. ANSYS Fluent is one of the most-powerful CFD software tools available. The choice of this popular commercial CFD software tool for the present study is also motivated by the fact that the parallel scalability of ANSYS Fluent based solvers is seldom reported in the scientific literature [13, 31] compared to the solvers based on the open source alternatives, such as OpenFOAM [9, 36], DUNE [10, 26], FEniCS [33].

The main goal of the present research is to investigate the parallel efficiency of two main numerical approaches commonly used for the solution of CFD problems, specifically for the problem of flow through the open and fixed aortic valve. In this work, we study the scalability of parallel algorithms based on coupled [5, 8] and segregated [14, 30] numerical schemes. In the segregated approach, discretized PDEs of all independent variables are decoupled and solved separately. In the coupled approach, discretized PDEs of velocity and pressure variables are solved in one system of linear equations. For relatively simple CFD benchmarks, coupled schemes are reported to be not only more robust, but also significantly faster in CPU time [5, 8]. In this work, we study and compare the performance of both schemes, solving specific computationally challenging problem. Moreover, both types of schemes have different parallelization properties. Their comparative parallel performance is not obvious for each specific problem.

Other parts of the paper are organized as follows. Section 2 presents the mathematical model and defines the problem of flow through the aortic valve, which is used in this work for our study. In Section 3, first, we briefly present the numerical schemes that are applied to solve the considered problem. Next, we show and discuss their convergence. In Section 4, we discuss the parallelization of two employed solvers. We present and analyze the parallel performance of both solvers, measured in case of the considered problem. Finally, some conclusions are presented.

2 Mathematical model and problem formulation

To describe the turbulent blood flow through the aortic valve, we use the Reynolds-averaged Navier-Stokes (RANS) equations for the transient flow of incompressible Newtonian fluid [21]:

$$\rho \frac{\partial \bar{u}_i}{\partial t} + \rho \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j} \left(\mu \frac{\partial \bar{u}_i}{\partial x_j} - \rho \overline{u'_i u'_j} \right), \quad i = 1, 2, 3, \quad (2.1)$$

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0, \quad (2.2)$$

where \bar{u}_i is the mean velocity component ($i = 1, 2, 3$), \bar{p} is the mean pressure, μ is the viscosity of blood, ρ is the density of blood, $\overline{\rho u'_i u'_j}$ are the Reynolds stresses ($i, j = 1, 2, 3$). RANS equations are derived from the Navier-Stokes equations by using the Reynolds decomposition to decompose the flow variable (e.g. velocity u_i) into the mean (time-averaged) component \bar{u}_i and the fluctuating component u'_i , i.e. $u_i = \bar{u}_i + u'_i$. Note that we use the Einstein summation convention in equations (2.1)–(2.2).

The Newtonian fluid model was considered, since the blood in the aorta and large arteries behaves accordingly [23]. The non-Newtonian blood behaviour becomes relevant only in the cases of the slow flows in the small arteries and veins. Thus, the blood was modelled as an incompressible Newtonian fluid with the dynamic viscosity μ set to 0.004028 kg/(m s). The density of the blood ρ was set to 1060 kg/m³.

The need for turbulence modelling arises because of the high Reynolds numbers of the blood flow past the aortic valve, following the peak systole [3, 23]. Turbulence models based on RANS approach are commonly used in the numerical studies of the flow through the aortic valve [15, 23]. There are also attempts to employ the Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS) approaches to model the pulsatile flow through the bileaflet mechanical heart valves (see the references in [23]). As expected, the latter methods show the ability to capture the finer details of the flow field [12]. However, RANS approach is often preferred for the balance of turbulence resolution and computational cost. It also shows a good agreement with the more detailed turbulence models [12] as well as experimental data [28, 35].

In this work, the Boussinesq hypothesis is applied to model the Reynolds stress terms $\overline{\rho u'_i u'_j}$ in equation (2.1):

$$\overline{\rho u'_i u'_j} = \mu_t \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} \delta_{ij} \rho k, \quad (2.3)$$

where δ_{ij} is the Kronecker delta, μ_t is the turbulent viscosity, k is the turbulent kinetic energy. In this work, we use the two-equation k - ε model [21] to find the turbulent viscosity and turbulent kinetic energy. We have performed the study on performance of various turbulence models in another our work [38].

A critical prerequisite for the patient-specific analysis of the aortic valve is the integration of the state-of-the-art clinical imaging with CFD computations. To obtain the problem domain for this study, 3D images of the aortic valve were acquired from a human subject by using the computer tomography equipment GE LightSpeed VCT (GE Healthcare, USA). The MITK (Medical Imaging Interaction Toolkit) software [29] was employed to extract the geometric parameters of the aortic valve from the acquired images in DICOM (Digital Imaging and Communications in Medicine) format.

A 3D geometric model of the open aortic valve was constructed from the parametric curves according to the obtained patient-specific geometric parameters. The surface of sinuses was represented by the extrusion of curtate epicycloid. The analytical surface model of the valve leaflets is based on the hypocycloidal type surface of extrusion. The details of the geometric model can be found in [18, 38]. Figure 1a shows the final geometry of the aortic valve used

in our tests. Finally, the geometric model was imported into the ANSYS DesignModeler software for the mesh generation. The generated computational mesh with 834534 cells is shown in Figure 1b.

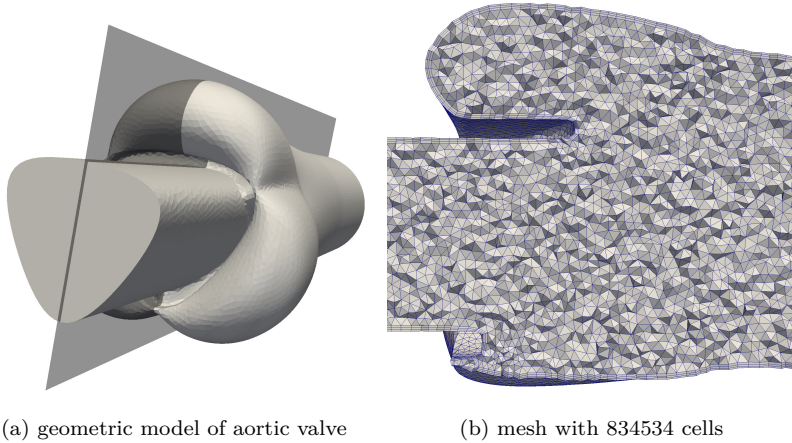


Figure 1. Problem geometry of open and fixed aortic valve and computational mesh

It is important to note here that the present research is focused on the computational challenges of flow resolution in the peak systole phase at maximum velocities. Obviously, a full cardiac cycle simulation requires to taking into account the movement of aortic valve leaflets. There are different approaches for doing that. An approach based on the Fluid-Structure Interaction (FSI) looks very promising in recent studies [22, 23, 24]. However, FSI computations require the profound knowledge of the complex tissue rheology, the electrical properties or even the external load due to the surrounding organs. It is hardly possible to obtain the required data in a non-invasive way today.

Another important moment in the problem setup, which cannot be underestimated, is the specification of the appropriate boundary conditions. The systolic phase of the cardiac cycle was simulated by applying a time-dependent velocity of the plug flow as the inflow boundary condition:

$$u_3(\vec{x}, t) = U(t), \quad \vec{x} \in S_{inlet}, \quad t \in [0, T], \quad (2.4)$$

where u_3 is the velocity component normal to the inlet surface S_{inlet} . Note that we have returned to the usual notation of velocity. Tangential to the inlet S_{inlet} velocity components were set to zero:

$$u_1(\vec{x}, t) = 0, \quad u_2(\vec{x}, t) = 0, \quad \vec{x} \in S_{inlet}, \quad t \in [0, T]. \quad (2.5)$$

The normal velocity component $U(t)$ was determined by using the clinical Doppler measurements of the considered patient performed by the Philips iE33 ultrasonographic system (see Figure 2b). Measured data were approximated by the smooth curve to avoid known convergence problems. Obtained waveform

of the inlet velocity (2.4) is shown in Figure 2a. The measured velocity reached maximal value of 1.44 m/s at $t = 0.13$ s.

In this work, we have simulated the flow through the aortic valve with leaflets fixed at the open position in time interval $[0, T]$, where $T = 0.2$ s. All simulations were started from a zero initial condition.

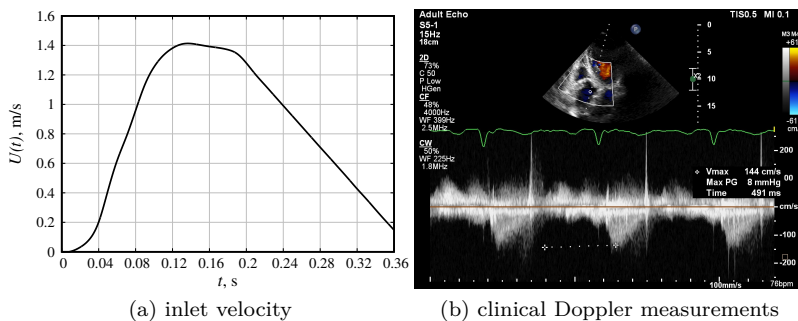


Figure 2. The analytic approximation a) of the measured blood velocity b) serving as the inlet boundary condition (2.4)

The no-slip boundary conditions were prescribed for the velocity on the aorta walls and fixed rigid leaflet surfaces. So called outflow boundary conditions are used to model flow exits, where the details of the flow velocity and pressure are not known prior to solving the flow problem. The required information is extrapolated from the interior of the solution domain. The outflow boundary conditions are approached physically in fully developed flows. The applied outflow boundary conditions have the smallest impact on the upstream flow solution and allow vortices to exit the computational domain with minimal disturbances. However, the outflow boundary conditions usually lead to solution divergence in the case of backflows [11]. This problem is going to be investigated in another our study.

3 Numerical solution algorithms

To solve the formulated problem of flow through the aortic valve, nonlinear Reynolds-averaged Navier-Stokes equations (2.1)–(2.2) are discretized with a finite volume method on generated collocated grids. Besides the choice of the grid, the way of coupling between the velocity and pressure is an essential part of any numerical scheme for the solution of Navier-Stokes equations. There are two main strategies to perform the velocity-pressure coupling, either a segregated or a coupled approach.

In the segregated approach, equations for all variables in the system are decoupled by using the fixed known values from the last iteration of the other independent variables. The linear systems obtained after the discretization can now be solved separately for all variables. A well-known representative of the segregated approach is the SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) algorithm [30].

This approach has the advantage of yielding small storage requirements and systems amenable to solution by classical iterative methods because of the standard structure and properties of system matrices. Unfortunately, suppressed interlinkage between the PDE equations results in serious drawbacks, such as convergence deterioration and the need for underrelaxation. Over the years many ways have been proposed to improve the convergence of the SIMPLE algorithm staying within segregated approach. Among the most important ones is the PISO (Pressure-Implicit with Splitting of Operators) algorithm [14]. It performs additional neighbour and skewness corrections to reduce the number of outer iterations and speed up the convergence.

The drawbacks of segregated schemes and tremendous increase in the available computer memory have stimulated the search for more robust, coupled methods of solution [5, 8]. The idea is to solve discretized momentum and pressure-based continuity equations together in one system of linear equations:

$$\sum_{j \in N(i)} \mathbf{A}_{ij} \vec{X}_j = \vec{B}_i, \quad i \in \omega_h, \quad (3.1)$$

where i is the cell index from discrete grid ω_h , j are cells from the the discretization stencil $N(i)$ of that cell,

$$\mathbf{A}_{ij} = \begin{pmatrix} a_{ij}^{pp} & a_{ij}^{pu} & a_{ij}^{pv} & a_{ij}^{pw} \\ a_{ij}^{up} & a_{ij}^{uu} & a_{ij}^{uv} & a_{ij}^{uw} \\ a_{ij}^{vp} & a_{ij}^{vu} & a_{ij}^{vv} & a_{ij}^{vw} \\ a_{ij}^{wp} & a_{ij}^{wu} & a_{ij}^{wv} & a_{ij}^{ww} \end{pmatrix}, \quad \vec{X}_j = \begin{pmatrix} p_j \\ u_j \\ v_j \\ w_j \end{pmatrix}, \quad \vec{B}_i = \begin{pmatrix} b_i^p \\ b_i^u \\ b_i^v \\ b_i^w \end{pmatrix},$$

here \mathbf{A}_{ij} is the discretization coefficients matrix, \vec{X}_j and \vec{B}_i are vectors with discrete unknowns and right-hand side values. It is worth noting that for clarity we have used here a different notation for velocity components.

Retaining the coupling between the momentum and pressure equations promotes the stability and accelerates the convergence rates [5, 8]. Application of the coupled scheme is advised when the quality of the mesh is poor, non-linear iterations are very expensive due to the time-consuming physical models for constitutive relations, or if larger time steps need to be used.

However, for significantly reduced and stable number of outer iterations of coupled scheme, we pay a price with a solution of four times larger systems of linear equations (3.1) with non-standard matrices. Thus, the performance of the coupled scheme is critically dependent on the performance of an iterative solver. There is a danger that the advantage of the higher convergence rate will be countered by the increase in computational time incurred in the solution of the enlarged system of equations.

Now we discuss and compare the performance of the coupled and PISO schemes solving the flow problem formulated in Section 2. The problem under consideration is characterized by the complex 3D flow patterns. Figure 3 shows the numerical solution at the time moment $t = 0.158$ s.

The obtained 3D velocity field is visualized by using the streamlines technique and ParaView software [17]. Figure 3 shows the vortices that are developing in aortic sinuses between the wall of the aorta and each of the three aortic

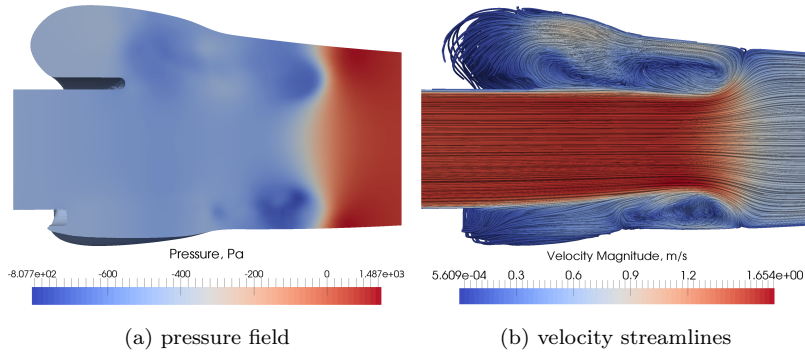


Figure 3. Numerical solution of flow through the aortic valve problem at time moment $t = 0.158$ s

cusps. The developed vortices detach and move downstream in the ascending aorta as the flow starts to decelerate after the peak systole.

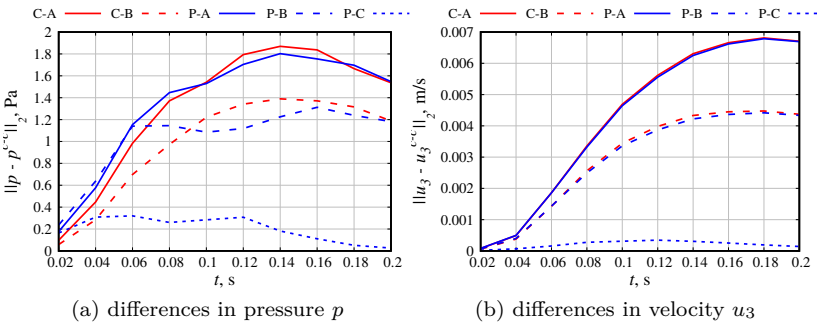


Figure 4. Time evolution of differences in pressure p and velocity u_3 computed in L_2 norm between the numerical solutions obtained with PISO (P-x) and coupled (C-x) schemes on three discrete meshes of increasing size (x-A, x-B, x-C)

In Figure 4, we compare the convergence of the coupled and PISO schemes applied to solve the formulated problem. It shows the time evolution of differences in pressure p and velocity u_3 between the various numerical solutions. The presented differences were computed in L_2 norm. We compare the numerical solutions obtained with PISO (P-x) and coupled (C-x) schemes on three different discrete meshes of increasing size. Figure 1b shows the mesh with 834534 cells, which is denoted by A in Figure 4. The mesh with 1679578 cells is denoted by B and that with 1941161 cells is represented by C. Numerical solution obtained with the coupled scheme on mesh C serves as the reference (exact) solution in this study.

In Figure 4, the curves P-A and C-A represent the differences in L_2 norm between the reference solution and solutions obtained on mesh A with PISO and coupled schemes, respectively. The curves P-B and C-B represent the

differences in L_2 norm between the reference solution and solutions obtained on mesh B. Finally, the curve P-C represents the time evolution of differences between solutions obtained with the PISO and coupled schemes on mesh C.

In all computational tests presented here, we have used a sufficiently small time step $\Delta t = 0.0002$ s, which is based on the time step independence study. We note here that Jahandardoost et al. [15] have reported using a time step $\Delta t = 0.001$ s for the RANS modelling of flow through the bileaflet mechanical valve with leaflets fixed in the open position and similar mesh size $2 \cdot 10^6$. It is important to note that we have a more complicated geometry in our test case. Le et al. [22] have used a time step $\Delta t = 1.074 \cdot 10^{-4}$ s in their FSI simulations of an aortic heart valve prosthesis. Thus, our time step is consistent with those used in the similar numerical studies [11, 15, 22].

According to its definition, L_2 norm shows the averaged difference between the two numerical solutions inside the whole problem domain. In Figure 4, the presented results show that the PISO and coupled schemes produce very similar solutions on the same mesh. Analyzing results on the different meshes, we see that obtained differences, comparing to the reference solution, increase as the flow accelerates and then slightly decrease in the deceleration phase. Nevertheless, convergence in space can be clearly observed. Similar results were obtained for the convergence in time.

In Table 1, we demonstrate the computational performance of the coupled and PISO schemes applied to solve the formulated problem. We present the total simulation time (the CPU wall time in hours), the average number of iterations per time step and the total number of cycles of algebraic multigrid (AMG) solver used for the numerical solution of arising linear systems.

Table 1. Computational performance of the coupled (C) and PISO (P) schemes solving the aortic valve problem: total solution time in hours, average number of iterations per time step and total number of AMG cycles

Mesh size	Time, h			Iterations			AMG cycles		
	C	P	C/P	C	P	C/P	C	P	C/P
834534	31.30	9.66	3.24	7.44	2.72	2.74	738965	926041	0.80
1679578	53.95	19.40	2.78	7.81	2.96	2.64	745131	935608	0.80
1941161	82.70	28.04	2.95	9.41	3.50	2.69	756767	1156378	0.65

First, we note that the computational complexity of the formulated problem and RANS model, considered in this study, is clearly demonstrated by the total simulation times in Table 1. Moreover, these solution times were obtained solving the formulated problem in parallel with 32 processes on 8 computing nodes. For this study, all performance tests were performed on the computer cluster “Vilkas” (<http://vilkas.vgtu.lt>) at the Laboratory of Parallel Computing of Vilnius Gediminas Technical University. We have used up to eight nodes with Intel® Core™ i7 processors with 4 cores (@ 2.80 GHz) per node. Computational nodes of dedicated cluster are interconnected via Gigabit Ethernet Smart Switch.

In agreement with the theory, the coupled scheme appeared to be very

robust. No convergence problems were encountered for different meshes and time steps considered in this study. At the same time, the PISO scheme experienced significant convergence problems. However, after a careful selection of numerical parameters, we were able to obtain superior performance not only in the solution time, but even in the number of iterations. In each PISO iteration, we have performed 2 skewness corrections and 3 neighbour corrections. The under-relaxation parameters were tuned for each specific case (mesh and time step) and changed inside the simulation time interval $[0, 2]$.

In the case of considered aortic valve problem, we have obtained the opposite conclusion compared to the results reported in other numerical studies [5, 8]. This can be possibly explained by the fact that much more simple CFD benchmarks were used in those studies (mostly 2D problems). Our results suggest that the comparative performance of coupled and segregated schemes strongly depends on the complexity of the flow patterns and the quality of the computational meshes.

It is clear that the quality of mesh is critically important for the numerical solution of the considered problem. However, mesh generation is very challenging in the considered case. Some open issues with the currently used mesh generation algorithms can be seen in Table 1. The total solution times increase by 53% (coupled) and 45% (PISO) as the mesh size increases from 1679578 cells to 1941161 cells (16%). These issues are going to be investigated in the future work.

4 Parallel performance tests and analysis

Now we present the parallel performance tests and analysis of solvers based on the PISO and coupled schemes that were applied to solve the considered challenging problem of flow through the aortic valve. The parallel solvers in ANSYS Fluent package are obtained in the same way as in alternative open source packages, such as OpenFOAM, DUNE and others. Parallelization in ANSYS Fluent is implemented at a low level using the MPI library and does not require any parallel specific coding from the user.

ANSYS Fluent employs the domain decomposition method [4] for the parallelization of numerical solution algorithms of PDEs problems. The mesh and its associated fields are partitioned into sub-domains, which are allocated to different processes. Parallel computation of any implicit finite volume scheme requires two types of communication [37]. The first type of communication has the local nature. It is done between the neighbouring processes for the approximation of differential operators on the given stencil and for the matrix-vector multiplication in iterative linear solvers. The second type of communication is the global communication, which is performed between all processes for computation of scalar products in iterative linear solvers.

ANSYS Fluent supports two methods of domain decomposition, which decompose the data into non-overlapping sub-domains: bisection and Metis methods. For our problem, the mesh is partitioned by using the method from the well-known Metis library [19] for graph and mesh partitioning. No geometric input is required from the user. This decomposition method attempts to ensure

the load balance between the parallel processes and to minimize the number of boundary edges between the sub-domains. The latter factor reduces the amount of data transferred between the neighbouring processes.

In Figure 5, we compare the strong scalability of two parallel algorithms based on the coupled and PISO schemes, respectively, for the solution of the considered aortic valve problem. We present the parallel performance results for two problems of fixed size solved on meshes with 0.8 and 1.6 million of cells. We show the measured values of parallel algorithmic speedup $S_p = T_1/T_p$ and parallel efficiency $E_p = S_p/p$, increasing the number of parallel processes p . Here T_p is the total CPU wall time of the solution. Note that due the large memory requirements, we were not able to solve the problem with 1.6 million cells on a single node. Therefore parallel performance results for this problem size are presented starting from 2 nodes, i.e. $2 \times 4 = 8$ parallel processes.

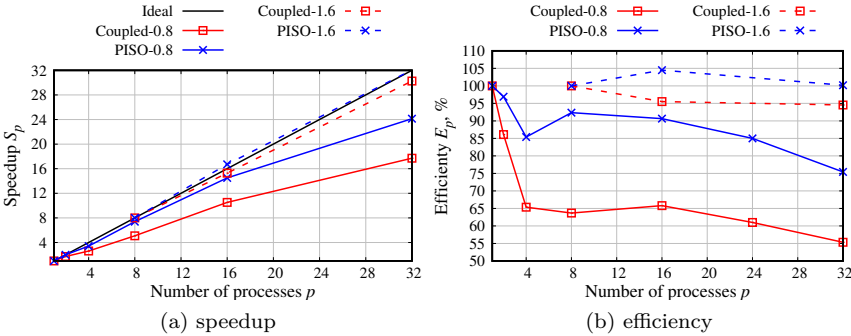


Figure 5. Strong scalability of parallel algorithms based on the coupled and PISO schemes solving the considered problem on meshes with 0.8 and 1.6 million cells

It can be observed from Figure 5 that the parallel algorithm based on the PISO numerical scheme demonstrates significantly better parallel scalability and efficiency than the parallel algorithm based on the coupled numerical scheme. As could be expected, the parallel performance metrics (speedup and efficiency) are greater for the bigger problem. This is in accordance with the theory. The parallel algorithm based on the PISO scheme shows even super-linear speedup. It is interesting to note that the measured parallel performance metrics in this study are significantly better than those obtained in our previous studies for OpenFOAM-based parallel solvers [36] and our own parallel solvers [6, 37].

In recent years, the common opinion of the parallel computing community is starting to shift in favor of a weak scalability studies. It is believed that in practice it is more important to solve the increasingly bigger problems without significant degradation in the performance (i.e. to have good weak scalability) than to efficiently reduce the solution time of a fixed-size problem (i.e. have good strong scalability).

Next, in Figure 6, we show and compare the weak scalability of two considered parallel algorithms based on the coupled and PISO schemes, respectively, solving the formulated problem of flow through the aortic valve. The parallel

efficiency of the weak scaling is defined as $E_N = T(0.8 \cdot 10^6; 8)/T(N; p(N))$. Here $T(N; p(N))$ is the total parallel solution time with $p(N)$ processes on mesh with N cells. In the weak scalability study, we solve the increasingly bigger problems with a proportionally increasing number of parallel processes $p(N)$ and measure $T(0.8 \cdot 10^6; 8)$, $T(1.6 \cdot 10^6; 16)$, $T(2.4 \cdot 10^6; 24)$, $T(3.2 \cdot 10^6; 32)$.

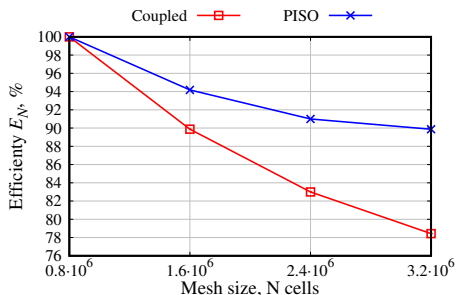


Figure 6. Weak scalability of parallel algorithms based on the coupled and PISO schemes solving the problem of flow through the aortic valve with increasing mesh size

The measured results of the parallel tests show that the weak scalability of the considered parallel solvers is even better than the strong scalability. Again, the parallel algorithm based on the PISO numerical scheme appears to be more advantageous for the parallel computations in spite of the less robust convergence.

Conclusions and future work

It is known that the choice between the coupled and segregated numerical schemes depends on the problem. We have shown that for the problems of challenging computational complexity significantly older segregated schemes like PISO can still be advantageous over the more modern and recently very popular coupled schemes. We believe that parallel efficiency of the developed numerical algorithms needs to be carefully studied for each important and complicated application.

In the studied case, the efficient application of parallel computing technologies makes the haemodynamic simulations more feasible. It enables the solution of more accurate models and complicated problems. However, employment of more accurate turbulence modelling approaches like LES and DNS would require simulation at even smaller time and space scales. Today, such simulations are only feasible on true supercomputers, but not on the middle-class clusters and workstations.

Parallel scalability of the OpenFOAM-based solvers will be studied for the considered problem of flow through the aortic valve in the future work. We will consider the performance of an important component of the parallel solution algorithm, namely, the scalability of the preconditioner for the parallel solution of large systems of linear equations.

Another important direction of the future work is a formulation of theoretical performance model for the considered parallel numerical algorithms. Accurate performance prediction model needs to properly describe the performance of linear system solver (including parallel scalability), convergence of PISO and coupled iterative methods, etc. Such theoretical performance model would need to be implemented in a special parallel performance prediction tool [7] and supplied with the properly obtained parameters. Many of these parameters will need to be test-specific. They will depend on the test (geometry, mesh, time step, flow parameters, etc.) and will need to be carefully reevaluated for each new test.

However, such parallel performance prediction tool would allow to assess the performance of whole parallel algorithm on various types of parallel computing infrastructure (e.g. GPU clusters) and choose suitable options and settings without real test runs, which can take up to several days and weeks (see Table 1).

Acknowledgements

The present research was supported by the Research Council of Lithuania in the framework of the project “Patient-specific simulation of the flow through the aortic valve” (MIP-052/2014).

Authors would like to thank the anonymous reviewers for their careful reading of manuscript and providing constructive comments and suggestions, which have helped to improve the quality of the paper.

References

- [1] A. AlOnazi. Design and optimization of OpenFOAM-based CFD applications for modern hybrid and heterogeneous HPC platforms. Master’s thesis, University College Dublin, Dublin, Ireland, 2013.
- [2] ANSYS Fluent. Scalability of ANSYS 16 applications and hardware selection - white paper. Technical report, Resource library, 2016.
- [3] D. Bluestein and S. Einav. Techniques in the stability analysis of pulsatile flow through heart valves. In Cornelius Leondes(Ed.), *Biomechanical Systems, Techniques and Applications, Volume II: Cardiovascular Techniques*. CRC Press, 2000. <https://doi.org/10.1201/9781420049534.ch-04>.
- [4] A.M. Bruaset and A. Tveito(Eds.). *Numerical solution of partial differential equations on parallel computers*. Lecture notes in computational science and engineering. Springer, Berlin, 2006.
- [5] Z.J. Chen and A.J. Przekwas. A coupled pressure-based computational method for incompressible/compressible flows. *Journal of Computational Physics*, **229**(24):9150–9165, 2010. <https://doi.org/10.1016/j.jcp.2010.08.029>.
- [6] R. Čiegis, A. Jakušev and V. Starikovičius. Parallel tool for solution of multiphase flow problems. In R. Wyrzykowski, J. Dongarra, N. Meyer and J. Waśniewski(Eds.), *Parallel Processing and Applied Mathematics: 6th International Conference, PPAM 2005, Poznań, Poland, September 11-14, 2005, Revised Selected Papers*, pp. 312–319. Springer Berlin Heidelberg, 2006. https://doi.org/10.1007/11752578_38.

- [7] R. Čiegis and V. Starikovičius. Realistic performance prediction tool for the parallel block LU factorization algorithm. *Informatica*, **14**(2):167–180, 2003.
- [8] M. Darwish, I. Sraj and F. Moukalled. A coupled finite volume solver for the solution of incompressible flows on unstructured grids. *Journal of Computational Physics*, **228**(1):180–201, 2009. <https://doi.org/10.1016/j.jcp.2008.08.027>.
- [9] A. Duran, M.S. Celebi, S. Piskin and M. Tuncel. Scalability of OpenFOAM for bio-medical flow simulations. *The Journal of Supercomputing*, **71**(3):938–951, 2015. <https://doi.org/10.1007/s11227-014-1344-1>.
- [10] C. Engwer and J. Fahlke. Scalable hybrid parallelization strategies for the DUNE grid interface. In A. Abdulle, S. Deparis, D. Kressner, F. Nobile and M. Picasso(Eds.), *Numerical Mathematics and Advanced Applications-ENUMATH 2013*, pp. 583–590. Springer, 2015. https://doi.org/10.1007/978-3-319-10705-9_57.
- [11] M. Esmaily Moghadam, Y. Bazilevs, T.Y. Hsia, I.E. Vignon-Clementel and A.L. Marsden. A comparison of outlet boundary treatments for prevention of backflow divergence with relevance to blood flow simulations. *Computational Mechanics*, **48**(3):277–291, 2011. <https://doi.org/10.1007/s00466-011-0599-0>.
- [12] L. Ge, H.L. Leo, F. Sotiropoulos and A.P. Yoganathan. Flow in a mechanical bileaflet heart valve at laminar and near-peak systole flow rates: CFD simulations and experiments. *Journal of Biomechanical Engineering*, **127**(5):782–797, 2005. <https://doi.org/10.1115/1.1993665>.
- [13] F. Greifzu, C. Kratzsch, T. Forger, F. Lindner and R. Schwarze. Assessment of particle-tracking models for dispersed particle-laden flows implemented in OpenFOAM and ANSYS Fluent. *Engineering Applications of Computational Fluid Mechanics*, **10**(1):30–43, 2016. <https://doi.org/10.1080/19942060.2015.1104266>.
- [14] R. Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of Computational Physics*, **62**(1):40–65, 1986. [https://doi.org/10.1016/0021-9991\(86\)90099-9](https://doi.org/10.1016/0021-9991(86)90099-9).
- [15] M. Jahandardoost, G. Fradet and H. Mohammadi. A novel computational model for the hemodynamics of bileaflet mechanical valves in the opening phase. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, **229**(3):232–244, 2015. <https://doi.org/10.1177/0954411915576944>.
- [16] A. Kačeniauskas. Development of efficient interface sharpening procedure for viscous incompressible flows. *Informatica*, **19**(4):487–504, 2008.
- [17] A. Kačeniauskas, R. Pacevič, A. Bugajev and T. Katkevičius. Efficient visualization by using ParaView software on BalticGrid. *Information Technology and Control*, **39**(2):108–115, 2010.
- [18] A. Kačeniauskas, R. Pacevič, V. Starikovičius, A. Maknickas, M. Staškūnienė and G. Davidavičius. Development of cloud services for patient-specific simulations of blood flows through aortic valves. *Advances in Engineering Software*, **103**:57–64, 2017. <https://doi.org/10.1016/j.advengsoft.2016.01.013>.
- [19] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing*, **20**(1):359–392, 1998. <https://doi.org/10.1137/S1064827595287997>.

- [20] S. Kulp, Z. Qian, M. Vannan, S. Rinehart and D. Metaxas. Patient-specific aortic valve blood flow simulations. In *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, pp. 939–942, 2014. <https://doi.org/10.1109/isbi.2014.6868026>.
- [21] B.E. Launder and D.B. Spalding. The numerical computation of turbulent flows. *Computer Methods in Applied Mechanics and Engineering*, **3**(2):269–289, 1974. [https://doi.org/10.1016/0045-7825\(74\)90029-2](https://doi.org/10.1016/0045-7825(74)90029-2).
- [22] T.B. Le and F. Sotiropoulos. Fluid-structure interaction of an aortic heart valve prosthesis driven by an animated anatomic left ventricle. *Journal of Computational Physics*, **244**:41–62, 2013. <https://doi.org/10.1016/j.jcp.2012.08.036>.
- [23] G. Marom. Numerical methods for fluid–structure interaction models of aortic valves. *Archives of Computational Methods in Engineering*, **22**(4):595–620, 2015. <https://doi.org/10.1007/s11831-014-9133-9>.
- [24] R. Mittal, J.H. Seo, V. Vedula, Y.J. Choi, H. Liu, H.H. Huang, S. Jain, L. Younes, T. Abraham and R.T. George. Computational modeling of cardiac hemodynamics: Current status and future outlook. *Journal of Computational Physics*, **305**:1065–1082, 2016. <https://doi.org/10.1016/j.jcp.2015.11.022>.
- [25] M.H. Moosavi, N. Fatouraei, H. Katoozian, A. Pashaei, O. Camara and A.F. Frangi. Numerical simulation of blood flow in the left ventricle and aortic sinus using magnetic resonance imaging and computational fluid dynamics. *Computer Methods in Biomechanics and Biomedical Engineering*, **17**(7):740–749, 2014. <https://doi.org/10.1080/10255842.2012.715638>.
- [26] E. Müller and R. Scheichl. Massively parallel solvers for elliptic PDEs in numerical weather- and climate prediction. *Quarterly Journal of the Royal Meteorological Society*, **140**(685):2608–2624, 2014. <https://doi.org/10.1002/qj.2327>.
- [27] E. Müller, R. Scheichl and E. Vainikko. Petascale solvers for anisotropic PDEs in atmospheric modelling on GPU clusters. *Parallel Computing*, **50**:53–69, 2015. <https://doi.org/10.1016/j.parco.2015.10.007>.
- [28] V.T. Nguyen, Y.H. Kuan, P.Y. Chen, L. Ge, F. Sotiropoulos, A.P. Yoganathan and H.L. Leo. Experimentally validated hemodynamics simulations of mechanical heart valves in three dimensions. *Cardiovascular Engineering and Technology*, **3**(1):88–100, 2012. <https://doi.org/10.1007/s13239-011-0077-z>.
- [29] M. Nolden, S. Zelzer, A. Seitel, D. Wald, M. Müller, A.M. Franz, D. Maleike, M. Fangerau, M. Baumhauer, L. Maier-Hein, K.H. Maier-Hein, H.P. Meinzer and I. Wolf. The Medical Imaging Interaction Toolkit: challenges and advances. *International Journal of Computer Assisted Radiology and Surgery*, **8**(4):607–620, 2013. <https://doi.org/10.1007/s11548-013-0840-8>.
- [30] S.V. Patankar and D.B. Spalding. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer*, **15**(10):1787–1806, 1972. [https://doi.org/10.1016/0017-9310\(72\)90054-3](https://doi.org/10.1016/0017-9310(72)90054-3).
- [31] S. Posey, B. Loewe and P. Calleja. Cluster scalability of ANSYS Fluent 12 for a large aerodynamics case on the Darwin supercomputer. In *4th European Automotive Simulation Conference*, pp. 1–6, 2009. Munich, Germany, 6–7 July
- [32] A.M. Pouch, H. Wang, M. Takabe, B.M. Jackson, C.M. Sehgal, J.H. Gorman, R.C. Gorman and P.A. Yushkevich. Automated segmentation and geometrical modeling of the tricuspid aortic valve in 3D echocardiographic images. In K. Mori, I. Sakuma, Y. Sato, C. Barillot and N. Navab(Eds.), *Medical Image*

- Computing and Computer-Assisted Intervention – MICCAI 2013: 16th International Conference, Nagoya, Japan, September 22–26, 2013, Proceedings, Part I*, pp. 485–492. Springer Berlin Heidelberg, 2013. https://doi.org/10.1007/978-3-642-40811-3_61.
- [33] C.N. Richardson and G.N. Wells. Expressive and scalable finite element simulation beyond 1000 cores. Technical report, University of Cambridge, 2013.
 - [34] D. Šešok, R. Belevičius, A. Kačeniauskas and J. Mockus. Application of GRID computing for optimization of grillages. *Mechanika*, **82**(2):63–69, 2010.
 - [35] E. Sirois, Q. Wang and W. Sun. Fluid simulation of a transcatheter aortic valve deployment into a patient-specific aortic root. *Cardiovascular Engineering and Technology*, **2**(3):186–195, 2011. <https://doi.org/10.1007/s13239-011-0037-7>.
 - [36] V. Starikovičius, R. Čiegis and A. Bugajev. On efficiency analysis of the OpenFOAM-based parallel solver for simulation of heat transfer in and around the electrical power cables. *Informatika*, **27**(1):161–178, 2016. <https://doi.org/10.15388/Informatica.2016.80>.
 - [37] V. Starikovičius, R. Čiegis, O. Iliev and Z. Lakdawala. A parallel solver for the 3D simulation of flows through oil filters. In *Parallel Scientific Computing and Optimization: Advances and Applications*, pp. 181–191. Springer New York, 2009.
 - [38] E. Stupak, R. Kačianauskas, A. Kačeniauskas, V. Starikovičius, A. Maknickas, R. Pacevič, M. Staškūnienė, G. Davidavičius and A. Aidietis. The geometric model-based patient-specific simulations of turbulent aortic valve flows. *Archives of Mechanics*, 2017. Submitted.
 - [39] L. Tumonis, R. Kačianauskas, A. Kačeniauskas and M. Schneider. The transient behavior of rails used in electromagnetic railguns: numerical investigations at constant loading velocities. *Journal of vibroengineering*, **9**(3):15–18, 2007.