

Comparison of mesh lifting operators for Fluid-Structure Interaction

Andreas Slyngstad

Master's Thesis, Autumn 2017



This master's thesis is submitted under the master's programme *Computational Science and Engineering*, with programme option *Mechanics*, at the Department of Mathematics, University of Oslo. The scope of the thesis is 60 credits.

The front page depicts a section of the root system of the exceptional Lie group E_8 , projected into the plane. Lie groups were invented by the Norwegian mathematician Sophus Lie (1842–1899) to express symmetries in differential equations and today they play a central role in various parts of mathematics.

Acknowledgements

First off, I would like to my supervisors Dr. Kristian Valen-Sendstad and Aslak Wigdahl Bergersen. Kristian, your encouragement .. Aslak, numerical insight and problem solving methodology have been mostly a

I would also like to thank Professor Mikael Mortensen and Miroslav Kuchta at the Department of Mathematics at the University of Oslo. Your technical insight in the FEniCS project have been essential for completing this thesis.

Finally I would like to thank my partner Charlotte and daughter Linde Olivia. Your love and support have kept me motivated during this thesis, would not and for giving me something to look

Contents

1	Governing equations	3
1.1	Continuum Mechanics	3
1.2	The Lagrangian and Eulerian description of motion	4
1.3	The Solid problem	6
1.4	The Fluid problem	9
	Appendices	11
A	The deformation gradient	13

Numerical Experiments

1.1 Comparison of mesh lifting operators

Mesh lifting operators are essential for numerical stability of fluid-structure interaction. If the fluid mesh doesn't conform with the solid deformation, mesh entanglement increases with the possibility of numerical instabilities. In general, mesh models have shown to be either robust concerning mesh entanglements at the cost of computational time, or computationally efficient with less robustness [20]. However, computational efficiency has proven not only to be dependent of the complexity of model, but also the regularity of the fluid mesh, reducing Newton iterations needed per time step [41]. In this section I compare the mesh moving models from section 3.1.4. for the FSI-3 benchmark. The linear elastic model was found not applicable in section 4.2.3. Therefore, only the Laplace and biharmonic lifting operators will be considered, comparing vertical displacement of the elastic flag, regularity of the fluid mesh, and number of Newton iterations per time step. To evaluate the regularity of the fluid mesh, the minimum value of the Jacobian of the deformation gradient have been considered in [41]. The Jacobian serves as a measure of mesh entanglement, meaning if $J_f \geq 0$, there are no crossing cells in the fluid mesh.

$$J_f = \det(\hat{\mathbf{F}}_f) = \det(I + \hat{\nabla} \hat{\mathbf{u}}_f)$$

where I is the identity matrix and $\hat{\mathbf{u}}_f$ is the fluid mesh deformation.

Results

Figure ?? shows a comparison the mesh moving models at the time interval $t = [0, 5]$, when a stable periodic solution is obtained. As shown in the middle figure all models shows a minima of mesh regularity at $3.8s < t < 4.2$, which is expected due to the largest deformation of the elastic flag. A larger number of Newton iterations for all models shows are needed at each time step, when the elastic flag starts oscillating for $t > 3s$. The biharmonic models is superior in terms of number of iterations need per time step, and mesh regularity in comparison with the Laplace model. Further, the biharmonic 2 model shows better mesh regularity than biharmonic 1, but shows equal behavior in terms of Newton iterations. For all models, no distinct difference in deformation of the y-component is found.

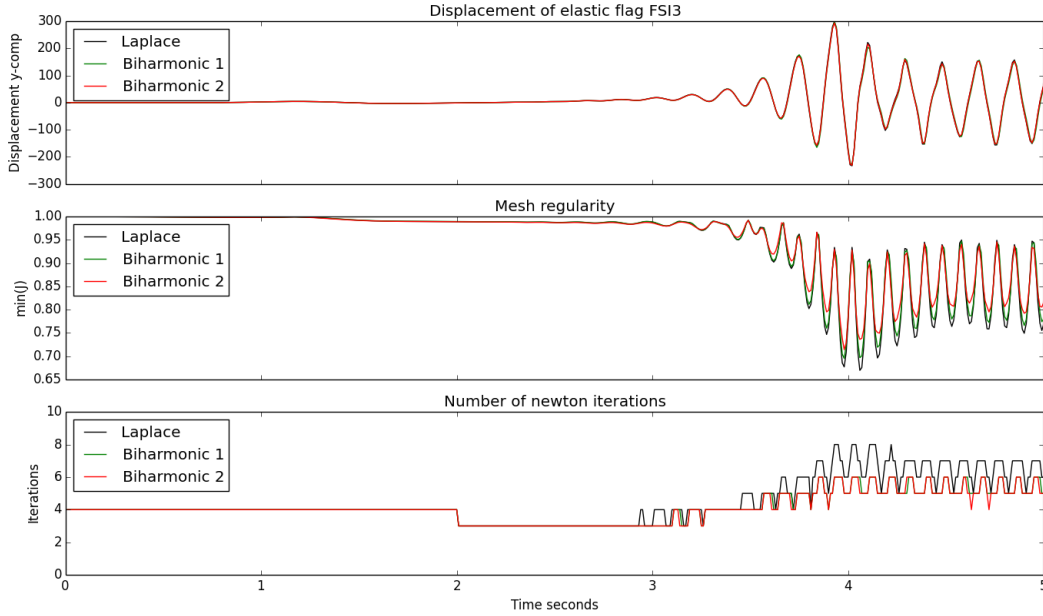


Figure 1.1: Investigation of mesh moving models for the FSI3 benchmark in the time interval $t \in [0, 5]$, comparing vertical displacement of elastic flag, mesh regularity, and number of Newton iterations.

Discussion

The numerical results confirms biharmonic models produce a better regularity of the fluid mesh cells, which in turn reduces the number of Newton iterations needed per time step. However, better evolution of mesh cells is by no means necessary for solving the FSI-3 problem. Therefore, the Laplace model remains a good choice, and its simplicity is preferable in terms of computational time (a topic to be discussed in section ?? ??).

1.2 Investigation of long term temporal stability

One of the main challenges for constructing time-stepping schemes for ALE-methods, is the additional non-linearity introduced by the domain-velocity term in the fluid problem [9]:

$$\hat{\mathbf{J}}_f(\hat{\mathbf{F}}_f^{-1}(\hat{\mathbf{v}}_f - \frac{\partial \hat{\mathbf{T}}_f}{\partial t}) \cdot \hat{\nabla})\hat{\mathbf{v}}_f \quad (1.1)$$

Closer inspection of the convection term reveal spatial and temporal differential operators depending non-linearly on one another. These differential operators often appear separated, making discretization of a general time-stepping scheme not directly intuitive. The domain velocity $\frac{\partial \hat{\mathbf{T}}_f}{\partial t}$ have proven to effect the stability of first and second-order accurate time stepping schemes on fixed grids, but to what extent remains unclear [9, 8]. The second order Crank-Nicolson used in this thesis, have also shown to suffer from temporal stability for long-term simulations of fluid problems, on fixed-grids [40]. The unconditionally stable Crank-Nicolson scheme is restricted by the condition [40]:

$$k \leq ch^{\frac{2}{3}} \quad (1.2)$$

Where c is a constant, while k and h is the time-step and a mesh-size parameter. While for the stability of the time derivative of the ALE-mapping, no accurate restriction is obtained (but thorough explored in [9]). As a result, time step restriction is necessary to ensure that numerical stability [9]. The temporal stability for the implicit Crank-Nicolson scheme, for the validation benchmark chosen in this thesis, was studied in [26]. The criteria for the numerical experiments was to obtain a stable solution in the time interval of 10 seconds, by temporal and spatial refinement studies. Following the ideas of [26], a second order scheme based on the Crank-Nicolson yields two possibilities:

Discretization 1.1. *Crank-Nicolson secant method*

$$\left[\frac{\hat{\mathbf{J}}(\hat{\mathbf{u}}^n)\hat{\nabla}\hat{\mathbf{v}}^n\hat{\mathbf{F}}_W^{-1}}{2} + \frac{\hat{\mathbf{J}}(\hat{\mathbf{u}}^{n-1})\hat{\nabla}\hat{\mathbf{v}}^{n-1}\hat{\mathbf{F}}_W^{-1}}{2} \right] \frac{\hat{\mathbf{u}}^n - \hat{\mathbf{u}}^{n-1}}{k}$$

Discretization 1.2. *Crank-Nicolson midpoint-tangent method*

$$\left[\frac{\hat{\mathbf{J}}(\hat{\mathbf{u}}_{cn})\hat{\nabla}\hat{\mathbf{v}}_{cn}\hat{\mathbf{F}}_W^{-1}}{2} \right] \frac{\hat{\mathbf{u}}^n - \hat{\mathbf{u}}^{n-1}}{k} \quad \hat{\mathbf{u}}_{cn} = \frac{\hat{\mathbf{u}}^n + \hat{\mathbf{u}}^{n-1}}{2} \quad \hat{\mathbf{v}}_{cn} = \frac{\hat{\mathbf{v}}^n + \hat{\mathbf{v}}^{n-1}}{2}$$

The numerical experiments showed very similar performance for Discretization ??, ??, and significant differences of temporal accuracy was not found. However, spatial and temporal refinement showed the implicit Crank-Nicolson scheme gave stability problems for certain time steps. Choosing the time step $k = [0.005, 0.003]$, the FSI-3 problem (Section ??) suffered from numerical instabilities. Interestingly, the instabilities occurred earlier in simulation time for increasing mesh refinement. A similar experiment in [42], showed reducing the time step $k = 0.001$ yield stable long-time simulation for both Discretization ?? and ??

To overcome the numerical instabilities two approaches have been suggested in the literature, the *shifted Crank-Nicolson* and the *frac-step method* [26, 42, 40]. In this thesis the shifted Crank-Nicolson scheme was considered, introducing stability to the overall system by shifting the centered scheme by $\theta = \frac{1}{2} + k$. If the shift is dependent of the time-step k such that $\frac{1}{2} \leq \theta \leq \frac{1}{2} + k$, the scheme will be of second order [26].

1.2.1 Results

A numerical investigation of long term numerical stability is shown in Figure ??, ??, where the shifted Crank-Nicolson scheme $\theta = \frac{1}{2} + \Delta t$, is compared the original Crank-Nicolson $\theta = \frac{1}{2}$. The shifted version surpasses the original Crank-Nicolson scheme in terms of stability, for all time steps. Choosing $\Delta t = 0.01$, the shifted Crank-Nicolson scheme retain long-time temporal stability, while capturing the physics of the benchmark. While for the ordinary Crank-Nicolson scheme, numerical experiments showed choosing $\Delta t = 0.001$ was necessarily to ensure stability, confirming the results found in [42].

Figure ?? shows choosing $\Delta t \in [0.2, 0.1]$ results in a steady-state solution, which can be explained by influence the solid problem.. A centered Crank-Nicolson scheme $\theta = \frac{1}{2}$ is energy preserving, meaning no energy is dissipated from the system. Since the shifted Crank-Nicolson scheme is not centered, the amount of dissipated of energy is related to the time step. If the time step is sufficiently high such as $\Delta t \in [0.2, 0.1]$, the scheme will dissipate energy at a higher rate. Therefore, no periodic oscillation of the elastic flag is obtained. The validation of the solid solver shows the difference of energy dissipation for a centered and backward numerical scheme. Given the same solid parameters, a steady-state solution is obtained for CSM-1 ($\theta = 1.0$), while CSM-3 yields a periodic solution CSM-3 ($\theta = \frac{1}{2}$), shown in Figure ??.

For $\Delta t \in [0.05, 0.02]$, the shifted Crank-Nicolson scheme is close to centered, meaning energy is nearly preserved in the structure. However, the Newton-solver diverges before full time interval of 10 seconds is finished. Surprisingly, a finer time step $\Delta t = 0.02$ fails at an earlier stage than $\Delta t = 0.05$. The breakdown is not due to mesh entanglement of the ALE-mapping, but divergence of Newton method itself[26]. It is believed the divergence of the Newton solver is linked to the influence of the domain velocity, by the research found in [9], but no clear time step restriction is obvious. Hence, several publications indicates choosing time step for a shifted Crank-Nicolson scheme is based on trial and error [42, 40].

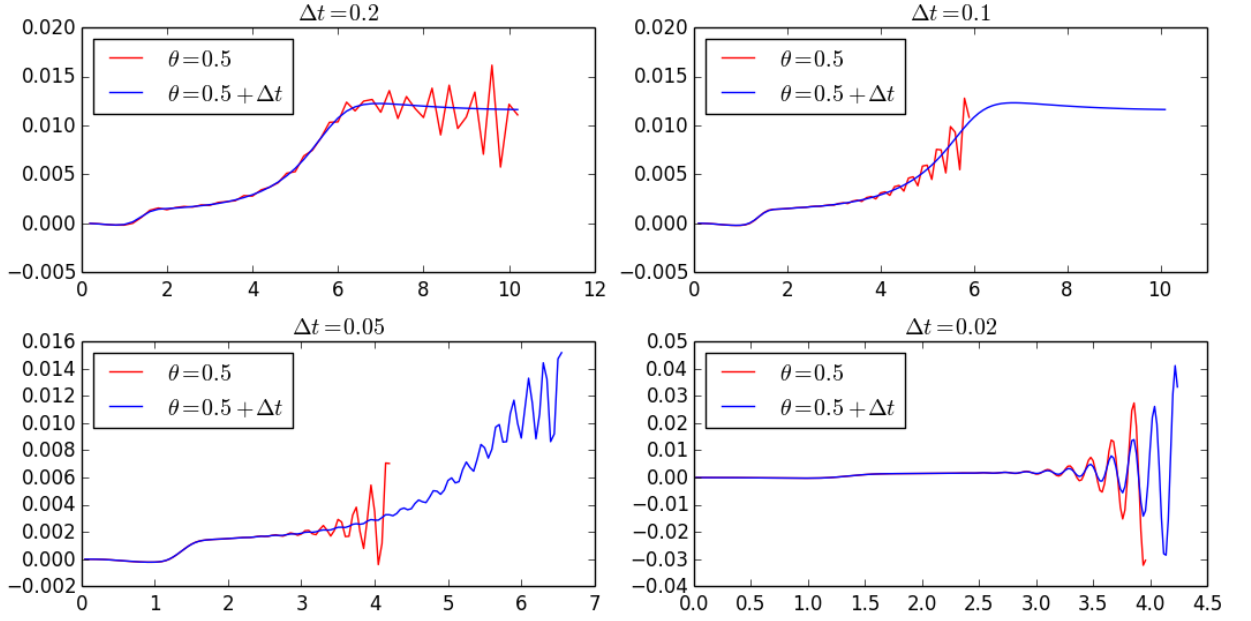


Figure 1.2: Investigation of long term numerical stability for the FSI3 benchmark in the time interval $t \in [0, 10]$, comparing the shifted and centered Crank-Nicolson scheme.

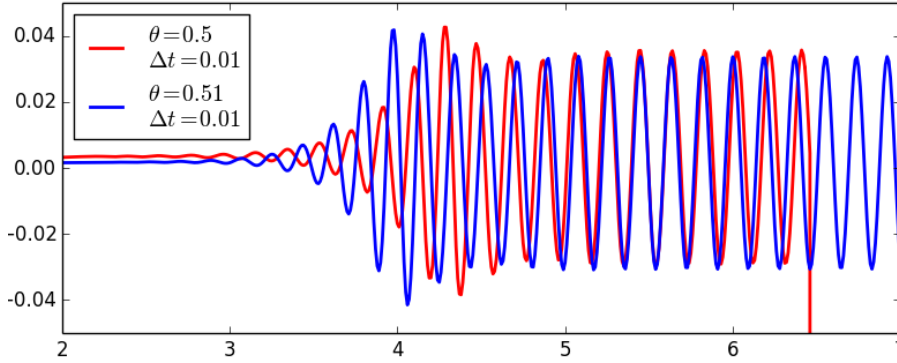


Figure 1.3: Investigation of long term numerical stability for the FSI3 benchmark in the time interval $t \in [0, 10]$, comparing the shifted and centered Crank-Nicolson scheme.

1.3 Optimization of the Newton solver

Software profiling is a dynamic program analysis, with the purpose of finding software sections which can be optimized. In scientific computing, software profiling often focuses on minimizing memory usage or executing speed, by collecting performance data to identify bottlenecks. A *bottleneck* is a term used when the total performance of a complete implementation is limited to small code fragments, accounting for the primary consumption of computer resources. For many applications, one can often assume the usage of resources follows the *The Pareto principle*. Meaning that for different types of events, roughly 80% of the effects come from 20% of the causes. An analogy to computational sciences it that 80% of the compu-

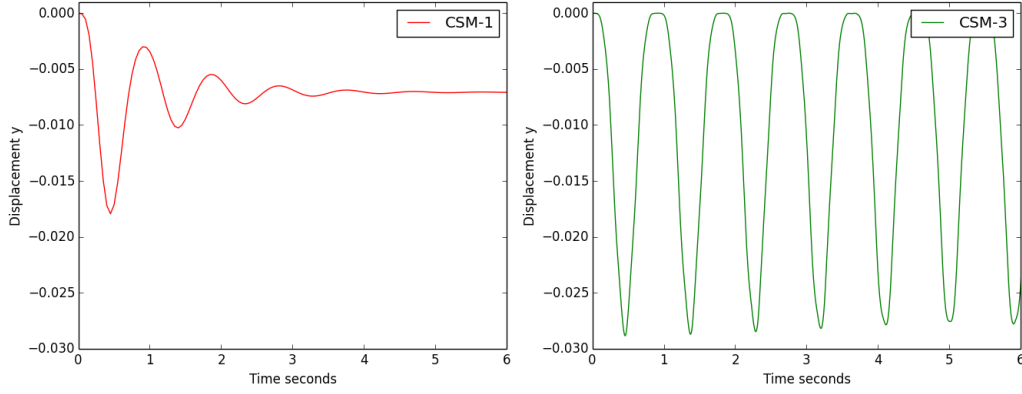


Figure 1.4: A comparison of a centered Crank-Nicolson scheme and backward Euler scheme, using the structure validation benchmark for the time interval $t \in [0, 6]$. Using the same structure parameters, a backward-Euler scheme $\theta = 1$ yields steady-state solution for CSM-1, while a periodic solution is observed for CSM-3 using a centered scheme $\theta = \frac{1}{2}$

tational demanding operations comes from 20% of the code. In this thesis, software profiling identified the bottleneck to be the Newton solver, using roughly $\sim 98\%$ of the total computational time for each individual numerical experiment. For a full simulation of the FSI-3 problem considered in Chapter ??, on a courses mesh with time step $k = 0.01$, the total computation time was initially 68 hours. With additional spatial and temporal refinement study of four mesh moving models, it became clear that the whole process would be too time consuming. Optimization of the Newton solver was therefore eminent to complete the initial goals set for this thesis.

Newtons method can be written as,

$$\nabla \mathbf{F}(\mathbf{x}_n)(\mathbf{x}_n - \mathbf{x}_0) = -\mathbf{F}(\mathbf{x}_n) \quad (1.3)$$

where \mathbf{F} is the residue of the variational formulation, \mathbf{x}_n , \mathbf{x}_0 is vector, and $\nabla \mathbf{F}$ is the Jacobian of the residue. The Newton method can be divided into two main computational demanding operations.

- **Jacobian assembly**

The construction of the Jacobian matrix $\nabla \mathbf{F}(\mathbf{x}_n)$ of the total residue $\mathbf{F}(\mathbf{x}_n)$

- **Direct solver**

LU factorization and solving the linear system $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} = \nabla \mathbf{F}(\mathbf{x}_n)$ and $\mathbf{b} = -\mathbf{F}(\mathbf{x}_n)$

The speed-ups methods explored in this thesis are divided into *consistent* and *inconsistent* methods. Consistent methods speeds-up the solution process by efficient assembly methods of the linear system ?. Inconsistent method involves simplifications of the linear system ?, often at the cost of poor convergence of Newtons method. As a consequence, additional Newton iterations are often necessary for convergence at each time step.

1.3.1 Consistent methods

Jacobi buffering

The residue of the FSI problem consists of both linear and non-linear terms, $\mathbf{F} = \mathbf{F}_{lin} + \mathbf{F}_{nonlin}$. For each time step the linear part \mathbf{F}_{lin} remains constant, which only need to be assembled on the first Newton iteration. The linear parts are then buffered, meaning saved in a separate matrix and added with the assembled non-linear \mathbf{F}_{nonlin} matrix for each Newton iteration at each time step.

1.3.2 Inconsistent methods

Reuse of Jacobian

Reusing the Jacobian of the residue $\nabla \mathbf{F}(\mathbf{x}_n)$ have two beneficial consequences. First, the Jacobian of the residue $\nabla \mathbf{F}(\mathbf{x}_n)$ is only assembled once at each time step. Second, the LU factorization of the linear system ?? is only needed at the first iteration, as the Jacobian remains constant through the whole time step. The first Newton iteration is therefore solved without simplifications, while the remaining iterations uses an inexact Jacobian which reduces the convergence of the method.

Quadrature reduce

The assemble time of the Jacobian greatly depends on the degree of polynomials used in the discretization of the total residual. The use of lower order polynomials reduces assemble time of the Jacobian at each Newton iteration due decreased size of the Jacobian matrix. The decreased size of the matrix also reduces the time needed for LU-factorization of the linear system ?. However, the quadrature reduce method leads to an inexact Jacobian which may results to additional iterations.

Combining quadrature reduce and Jacobian reuse

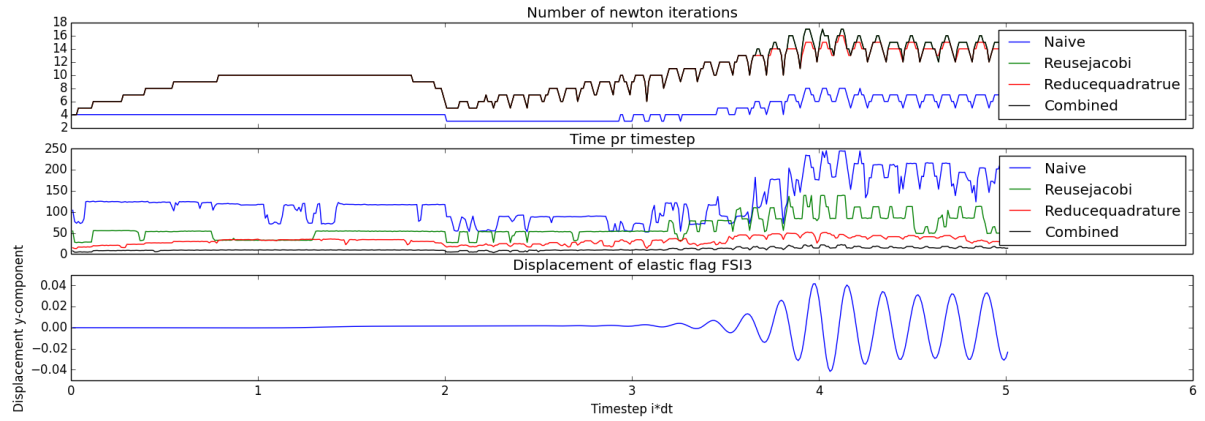
By using the previous inconsistent methods together, the first assembly of the Jacobian is done with lower order polynomials. This combination reduces assembly time of the Jacobian matrix at first iteration, and further speeds up LU-factorization of the linear system ?.

1.3.3 Comparison of speedup methods

In Figure ??, all speed-up techniques are compared on the time interval $t = [0, 5]$, for the Laplacian model. Numerical simulations shows how all inconsistent methods increase the number of iterations needed for convergence at each time step, contrary to a consistent naive method. However, the inconsistent methods clearly dominate the time used for a full Newton iteration at each time step, in comparison with the naive approach. The fastest method proved to be the combined method, using 1.5 hours for solving the full time interval. The naive approach used 17.1 hours, meaning the combined method resulted in a 91% speedup.

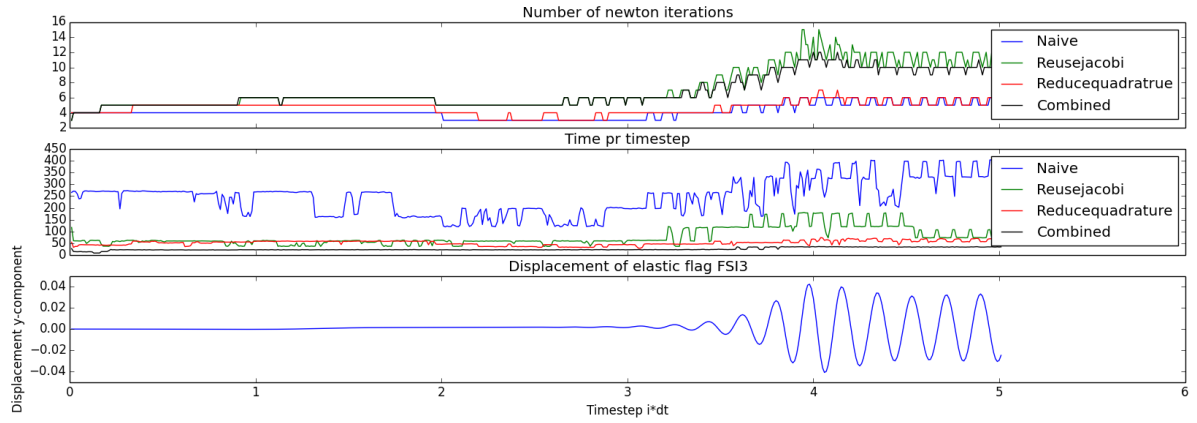
3. Optimization of the Newton solver

The biharmonic mesh model results in Figure ??, shows similar observation compared to the Laplacian model. However, the benefit of a better evolution of mesh cells is reflected in the reduced number of iterations needed at each times step, for all methods. The quadrature reduce method even compares to the naive method in terms of iterations. The naive approach computed the whole time interval in 33.87 hours, while the combined method 3.46 hours, a speedup of 89%.



Laplace					
Implementation	Naive	Buffering	Reusejacobi	Reducequadrature	Combined
Mean time pr timestep	123.10	159.85	61.31	31.43	11.11
Mean iteration	4.50	7.79	10.22	10.08	10.22
Speedup ratio	-	0.12	0.50	0.74	0.91

Figure 1.5: Comparison of speed-up techniques for the Laplace mesh model.



Biharmonic Type 1					
Implementation	Naive	Buffering	Reusejacobi	Reducequad	Combined
Mean time pr timestep	243.39	307.67	76.77	51.64	24.87
Mean iteration	4.14	6.21	7.19	4.67	6.81
Speedup ration	-	-0.26	0.68	0.79	0.90

Figure 1.6: Comparison of speed-up techniques for the biharmonic type 1 mesh model.

Discussion

Of the speed-up methods considered in this thesis, the combined method proved to be fastest for both mesh models. With a computational time difference of nearly ~ 2 hours, the Laplace model is superior to the biharmonic model in terms of efficiency. As long as mesh regularity isn't critical for the simulation, the Laplacian model is the preferable mesh moving model. It is important to note that the introduced speed-up methods may not be applicable to other FSI problems, since inexact Jacobian methods are highly sensitive.

Bibliography

- [1] Robert T Biedron and Elizabeth M Lee-Rausch. Rotor Airloads Prediction Using Unstructured Meshes and Loose CFD/CSD Coupling.
- [2] P I Crumpton. Implicit time accurate solutions on unstructured dynamic grids. (95):1–23, 1995.
- [3] J Donea, A Huerta, J-Ph Ponthot, and A Rodríguez-Ferran. Arbitrary Lagrangian-Eulerian methods. (1969):1–38, 2004.
- [4] Richard P Dwight. Robust Mesh Deformation using the Linear Elasticity Equations.
- [5] Stéphane Étienne, D Tremblay, and Dominique Pelletier. Code Verification and the Method of Manufactured Solutions for Fluid-Structure Interaction Problems. *36th AIAA Fluid Dynamics Conference and Exhibit*, (June):1–11, 2006.
- [6] Miguel A Fernández and Jean-Frédéric Gerbeau. Algorithms for fluid-structure interaction problems. 2009.
- [7] Miguel A. Fernández, Jean Frederic Gerbeau, and Ceremade Grandmont. A projection semi-implicit scheme for the coupling of an elastic structure with an incompressible fluid. *International Journal for Numerical Methods in Engineering*, 69(4):794–821, 2007.
- [8] L Formaggia and F Nobile. A stability analysis for the arbitrary Lagrangian Eulerian formulation with finite elements. *East-west journal of numerical mathematics*, Vol. 7(No. 2):105–131, 1991.
- [9] Luca Formaggia and Fabio Nobile. Stability analysis of second-order time accurate schemes for ALE-FEM. *Computer Methods in Applied Mechanics and Engineering*, 193(39-41 SPEC. ISS.):4097–4116, 2004.
- [10] Christiane Förster, Wolfgang A. Wall, and Ekkehard Ramm. Artificial added mass instabilities in sequential staggered coupling of nonlinear structures and incompressible viscous flows. *Computer Methods in Applied Mechanics and Engineering*, 196(7):1278–1293, 2007.
- [11] Bernhard Gatzhammer. Efficient and Flexible Partitioned Simulation of Fluid-Structure Interactions. page 261, 2014.

- [12] Brian T. Helenbrook. Mesh deformation using the biharmonic operator. *International Journal for Numerical Methods in Engineering*, 2003.
- [13] Gerhard A. Holzapfel. *Nonlinear Solid Mechanics: A Continuum Approach for Engineering*. 2000.
- [14] Jaroslav Hron and Stefan Turek. Proposal for numerical benchmarking of fluid-structure interaction between an elastic object and laminar incompressible flow. *Fluid-Structure Interaction*, 53:371–385, 2006.
- [15] Su-Yuen Hsu, Chau-Lyan Chang, and Jamshid Samareh. A Simplified Mesh Deformation Method Using Commercial Structural Analysis Software.
- [16] Jay D. Humphrey. *Cardiovascular Solid Mechanics*. Springer New York, New York, NY, 2002.
- [17] Hrvoje Jasak and Željko Tuković. Automatic mesh motion for the unstructured Finite Volume Method. *Transactions of Fomena*, 30(2):1–20, 2006.
- [18] V V Meleshko. Bending of an Elastic Rectangular Clamped Plate: Exact Versus 'Engineering' Solutions. *Journal of Elasticity*, 48(1):1–50, 1997.
- [19] José Merodio and Giuseppe Saccomandi. Continuum Mechanics - Volume I. In *Volume 1*, chapter 3, pages 82–84. EOLSS, 2011.
- [20] Selim MM and Koomullil RP. Mesh Deformation Approaches – A Survey. *Journal of Physical Mathematics*, 7(2), 2016.
- [21] J Newman. *Marine Hydrodynamics*. 1977.
- [22] William L. Oberkampf and Christopher J. Roy. *Verification and Validation in Scientific Computing*. Cambridge University Press, Cambridge, 2010.
- [23] M Razzaq, Stefan Turek, Jaroslav Hron, J F Acker, F Weichert, I Grunwald, C Roth, M Wagner, and B Romeike. Numerical simulation and benchmarking of fluid-structure interaction with application to Hemodynamics. *Fundamental Trends in Fluid-Structure Interaction*, 1:171–199, 2010.
- [24] T. Richter and T. Wick. Finite elements for fluid-structure interaction in ALE and fully Eulerian coordinates. *Computer Methods in Applied Mechanics and Engineering*, 199(41-44):2633–2642, 2010.
- [25] Thomas Richter. Fluid Structure Interactions. 2016.
- [26] Thomas Richter and Thomas Wick. On Time Discretizations of Fluid-Structure Interactions. pages 377–400. 2015.
- [27] Patrick J. Roache. Code Verification by the Method of Manufactured Solutions. *Journal of Fluids Engineering*, 124(1):4, 2002.

- [28] P.J. Roache. *Verification and Validation in Computational Science and Engineering*. Computing in Science Engineering, Hermosa Publishers, 1998, 8-9, 1998.
- [29] Edward J. Rykiel. Testing ecological models: The meaning of validation. *Ecological Modelling*, 90(3):229–244, 1996.
- [30] Kambiz Salari and Patrick Knupp. Code Verification by the Method of Manufactured Solution. Technical report, Sandia National Laboratories, 2000.
- [31] LE Schwer. Guide for verification and validation in computational solid mechanics. *American Society of Mechanical Engineers*, PTC 60(V&V 10):1–15, 2006.
- [32] Jason P Sheldon, Scott T Miller, and Jonathan S Pitt. Methodology for Comparing Coupling Algorithms for Fluid-Structure Interaction Problems. *World Journal of Mechanics*, 4(February):54–70, 2014.
- [33] J.C. Simo and F. Armero. Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier-Stokes and Euler equations. *Computer Methods in Applied Mechanics and Engineering*, 111(1-2):111–154, jan 1994.
- [34] Ian Sommerville. Verification and Validation. Technical Report February, 2006.
- [35] K Stein, T Tezduyar, and R Benney. Mesh Moving Techniques for Fluid-Structure Interactions With Large Displacements.
- [36] Stefan Turek, Jaroslav Hron, Mudassar Razzaq, and Hilmar Wobker. Numerical Benchmarking of Fluid-Structure Interaction : A comparison of different discretization and solution approaches.
- [37] A V, T Passerini, A Quaini, U Villa, A Veneziani, and S Canic. Numerical Analysis and Scientific Computing Preprint Series Validation of an open source framework for the simulation of blood flow in rigid and deformable vessels Preprint # 17 Department of Mathematics University of Houston. 2013.
- [38] Wolfgang A. Wall, Axel , Gerstenberger, Peter , Gamnitzer, Christiane , Förster, and Ekkehard , Ramm. Large Deformation Fluid-Structure Interaction – Advances in ALE Methods and New Fixed Grid Approaches. In *Fluid-Structure Interaction: Modelling, Simulation, Optimisation*, pages 195–232. Springer Berlin Heidelberg, 2006.
- [39] Frank White. *Viscous fluid flow*. McGraw-Hill, third edit edition.
- [40] T. Wick. Stability Estimates and Numerical Comparison of Second Order Time-Stepping Schemes for Fluid-Structure Interactions. In *Numerical Mathematics and Advanced Applications 2011*, pages 625–632. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

- [41] T Wick and Thomas Wick. Variational-monolithic ALE fluid-structure interaction: Comparison of computational cost and mesh regularity using different mesh motion techniques.
- [42] Thomas Wick. *Adaptive Finite Element Simulation of Fluid-Structure Interaction with Application to Heart-Valve*. PhD thesis, Heidelberg.
- [43] Thomas Wick. Solving Monolithic Fluid-Structure Interaction Problems in Arbitrary Lagrangian Eulerian Coordinates with the deal.II Library.
- [44] Thomas Wick. Fluid-structure interactions using different mesh motion techniques. *Computers and Structures*, 89(13-14):1456–1467, 2011.
- [45] Thomas Wick. Fully Eulerian fluid-structure interaction for time-dependent problems. *Computer Methods in Applied Mechanics and Engineering*, 255:14–26, 2013.
- [46] P. Wriggers. *Computational contact mechanics, second ed.*, Springer. 2006.