



Lehrstuhl für  
Strömungsmechanik  
Friedrich-Alexander-  
Universität  
Erlangen-Nürnberg



## MASTER THESIS

# **Dynamical calculation of under-relaxation factor in a partitioned, implicitly coupled fluid-structure interaction solver**

NARAYANAN ACHUTHAN

Erlangen, June 6, 2018

Examiner: [YOUR EXAMINER (probably Prof. Dr. Felix Freiling)]  
Advisor: [YOUR ADVISOR]



## **Eidesstattliche Erklärung / Statutory Declaration**

---

Hiermit versichere ich eidesstattlich, dass die vorliegende Arbeit von mir selbständig, ohne Hilfe Dritter und ausschließlich unter Verwendung der angegebenen Quellen angefertigt wurde. Alle Stellen, die wörtlich oder sinngemäß aus den Quellen entnommen sind, habe ich als solche kenntlich gemacht. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt.

I hereby declare formally that I have developed and written the enclosed thesis entirely by myself and have not used sources or means without declaration in the text. Any thoughts or quotations which were inferred from the sources are marked as such. This thesis was not submitted in the same or a substantially similar version to any other authority to achieve an academic grading.

---

Der Friedrich-Alexander-Universität, vertreten durch den Lehrstuhl für Informatik 1, wird für Zwecke der Forschung und Lehre ein einfaches, kostenloses, zeitlich und örtlich unbeschränktes Nutzungsrecht an den Arbeitsergebnissen der Arbeit einschließlich etwaiger Schutz- und Urheberrechte eingeräumt.

Erlangen, June 6, 2018

---

[YOUR NAME]



## Abstract

The presented numerical study focuses on dynamical calculation of the under-relaxation factor during each sub-iteration step of the *Fluid-Structure Interaction* solver using adaptive schemes. The adaptive schemes presented in the study are *Aitken's  $\Delta^2$  method* and *steepest descent method*. The mentioned schemes have been found to be efficient, yet easy to implement. The implemented schemes have been validated by a numerical simulation of flow around an elastically mounted circular cylinder at a Reynolds number of 200. (Cite Zhou)

The calculations were performed on a 2-D 0-type curvilinear orthogonal grids containing a total of 120x100 control volumes. The FSI simulations were performed using a *semi implicit predictor-corrector scheme* for fluid-structure coupling. The *semi implicit predictor-corrector scheme* is a strong coupling scheme between flow and structural solver, while also maintaining the explicit time marching schemes. The simulations were carried out for different reduced damping coefficients ( $Sg$ ) and for a mass ratio ( $M^*$ ) of 1. These cases were simulated with constant under-relaxation factor, and with dynamic under-relaxation factor using *Aitken's  $\Delta^2$  method* and *steepest descent methods*. The results were compared and validated with (Breuer and Muensch) and (zhou), the results were in good agreement with these established numerical data. Average time taken for sub-iterations within the time steps were calculated, *Aitken's  $\Delta^2$  method* was observed to be more efficient in accelerating the convergence.



# CONTENTS

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Task . . . . .	1
1.3	Related Work . . . . .	1
1.4	Results . . . . .	1
1.5	Outline . . . . .	2
1.6	Acknowledgments . . . . .	2
<b>2</b>	<b>Theoretical background</b>	<b>3</b>
2.1	Fluid-Structure Interaction . . . . .	3
2.1.1	Numerical simulation strategies of FSI problems . . . . .	4
2.1.2	Classification based on meshing strategies . . . . .	7
2.2	Governing equations of fluid flow and structural deformations . . . . .	8
2.2.1	Coupling of fluid and structural solvers . . . . .	9
2.3	Flow separation and vortex shedding . . . . .	12
<b>3</b>	<b>Implementation</b>	<b>15</b>
<b>4</b>	<b>Evaluation</b>	<b>17</b>
<b>5</b>	<b>Conclusion and Future Work</b>	<b>19</b>
	<b>Bibliography</b>	<b>21</b>





# INTRODUCTION

---

Some general information on the context and setting.

## 1.1 Motivation

Specific motivation for the problem at hand.

## 1.2 Task

Concrete task to be solved.

## 1.3 Related Work

Other relevant academic work and how it differs from this work, for example ? ] and ? ].  
Distinguish between “textual” citation, as shown in ? ], and “parenthesis” citation [? ].

## 1.4 Results

What has been achieved in this work?

## **1.5 Outline**

How is the thesis structured and why?

## **1.6 Acknowledgments**

A big thank you for the support to ...

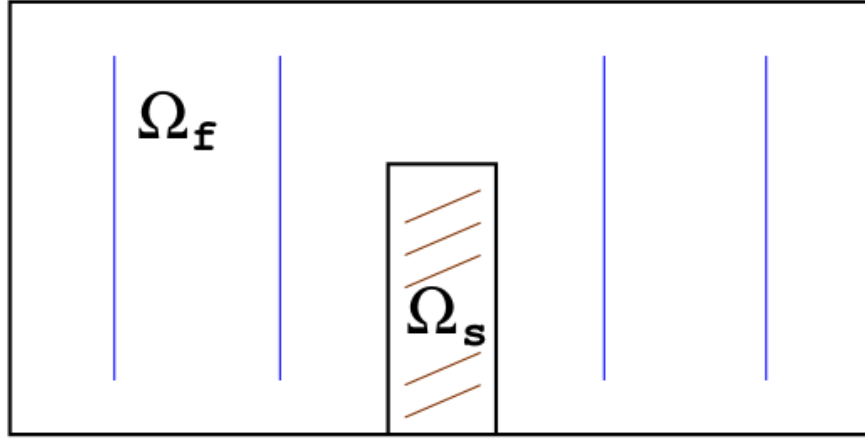
# 2

## THEORETICAL BACKGROUND

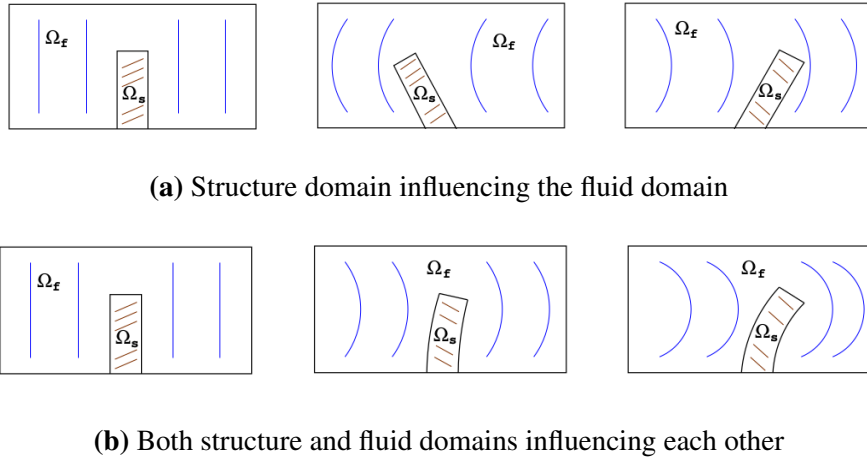
---

### 2.1 Fluid-Structure Interaction

Fluid-Structure Interaction (FSI) studies the interaction between a structure (solid) and a fluid flow (liquid or gas) around it. It is a multi-physics problem which has large interest in diversified fields such as mechanical engineering (e.g. airfoils), civil engineering (e.g. towers) or medicine technique (e.g. artificial heart valves). Based on the response of structure and fluid fields, it is classified as one-way or two-way fluid-structure interaction problem. If the structural displacement/deformation does not influence the flow fluid or vice versa, then the FSI system is termed to be one-way fluid-structure interaction system. Conversely if the fluid flow and the displacement or deformation of the structure have significant influence on each other then the FSI system is termed as two-way fluid-structure interaction problem. Figure 2.1 represents a typical domain of a fluid-structure interaction problem,  $\Omega$  refers to the common domain,  $\Omega_f$  the fluid domain and  $\Omega_s$  the structure domain. Figure 2.2a and 2.2b represents one-way and two-way FSI fluid-structure interaction problems respectively.



**Figure 2.1:** Representational figure of Fluid-Structure Interaction domain,taken from Richter [15]



**Figure 2.2:** Representational figure of one-way 2.2a, and two-way 2.2b fluid-structure interaction domains,taken from Richter [15]

### 2.1.1 Numerical simulation strategies of FSI problems

Numerical simulation of FSI involves solving set of differential equations and corresponding boundary conditions for fluid and structural fields respectively. Suitable interface conditions needs to be defined so as the structural and fluid domains are well distinguished. There are different methodologies implemented to solve the FSI problem and are well documented in the literature. An overview of the numerical solution procedure for FSI problems are presented below.

The FSI problem is classified based on solution approaches and on the treatment of mesh

handling techniques as represented below. A brief overview of these methods are presented subsequently.

- Classification based on numerical solution approaches
  - Monolithic solver
  - Partitioned solver
    - \* Explicit coupling
    - \* Implicit coupling
- Classification based on meshing strategies
  - Conforming mesh
  - Non-conforming mesh

### Classification based on numerical solution approaches

This classification is based on how the fluid and structural fields are getting solved. In *monolithic solvers* both the fluid and structural domains are expressed by single set of equations, whereas in *partitioned solver approach* structural fields and fluid fields are solved separately and additionally requires coupling of the distinct solvers. A brief introduction to these methods are represented below.

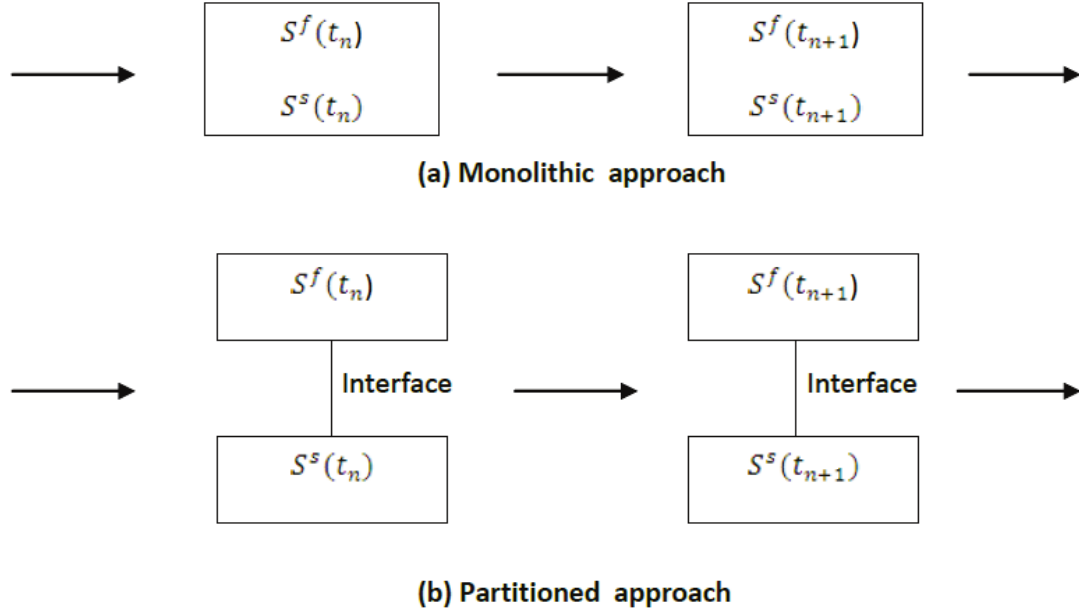
#### Monolithic solver approach

The equations governing the fluid flow and the displacement/deformation of the structure are represented by single set of equations which are then solved simultaneously by an unified algorithm, within a single solver framework Becker et al. [1], Hübner et al. [7], Michler et al. [10], Richter [15]. The central idea of monolithic solvers is to represent the interface by an homogeneous discretization, thus maintaining the conservation properties at the interface Michler et al. [10] Van Brummelen et al. [18]. Although monolithic approach gives strong coupling between the fields, it is commonly considered to be impractical for real world applications. It also demands enormous computational power for solving such a large system of equations since it has to incorporate the behavior of both fluid flow and solid structure. Figure 2.3 represents a schematic of the monolithic and partitioned solver approaches in solving FSI problems.

#### Partitioned solver approach

Another approach to fluid-structure interaction is to use two distinct solvers to model both fluid and solid domains. This technique allows the coupling of the fluid and solid solution by maintaining suitable coupling conditions. The interface conditions are used explicitly to communicate information between the fluid and structure solutions.

Some coupling algorithms were suggested by various studies (Farhat and Lesoinne [4], Felippa et al. [5], Piperno et al. [14]) which allows for reuse of existing codes that



**Figure 2.3:** Schematic representation of Monolithic and Partitioned solution approaches.  $S^f$  represents the fluid solver and  $S^s$  represents the structural solver for two successive time steps current ( $t_n$ ) and next ( $t_{n+1}$ ). Figure taken from Hou et al. [6]

have been developed for each field. This approach is very robust and can be used for wide variety of applications. The major disadvantage of this approach is that, the interface location that divides the fluid and the structure domains is not known a-priori and usually changes in time. Thus, the partitioned approach requires tracking of the new interface location and its related quantities, which can be cumbersome and error-prone. The interface coupling conditions that are as stated below, have to be satisfied in order to have a stable solution.

- Kinematic coupling condition: The displacements, velocities and accelerations of the sub-zones have to be equal at the interface at any point in time.

$$\psi_{\Gamma}^{CFD}(t) = \psi_{\Gamma}^{CSD}(t), \dot{\psi}_{\Gamma}^{CFD}(t) = \dot{\psi}_{\Gamma}^{CSD}(t), \ddot{\psi}_{\Gamma}^{CFD}(t) = \ddot{\psi}_{\Gamma}^{CSD}(t)$$

- Dynamic coupling condition: Conservation of the dynamic equilibrium of all forces at the interface needs to be satisfied. (Action and reaction forces must cancel out each other)

$$f_{\Gamma}^{CFD}(t) = -f_{\Gamma}^{CSD}(t)$$

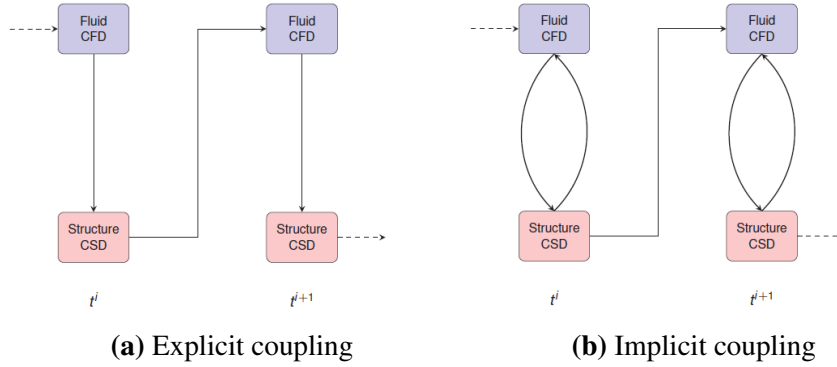
Depending on the influence of the structure movement on the fluid field the FSI problems are further subdivided into two following categories:

- Explicit/Weakly coupled: In an explicitly coupled algorithm, the equations of fluid mechanics, structural mechanics and the relative mesh movement are solved sequentially. Initially the governing equation of fluid is solved with the velocity

boundary condition derived from the structural displacement. The structural mechanics equations are solved next with the forces obtained from the fluid solver. Finally the grid is adapted based on the structural displacement. This coupling scheme is termed as weak coupling, as there is no sub iteration between the two sub zones.

- **Implicit / Strongly coupled:** In a strongly/implicitly coupled system, sub iterations between both the fluid and structural solvers are carried out at every time step till convergence is achieved. The convergence is determined by maintaining a small enough structural residuum. Detailed explanation on this, is presented in the subsequent section. This approach is implicit in nature and more robust, however requires additional computation time.

Figure 2.4 represents the schematic representation of explicit and implicitly coupled, partitioned approach to solving fluid structure interaction. In the presented study *semi-implicit predictor-corrector* coupling scheme is used.



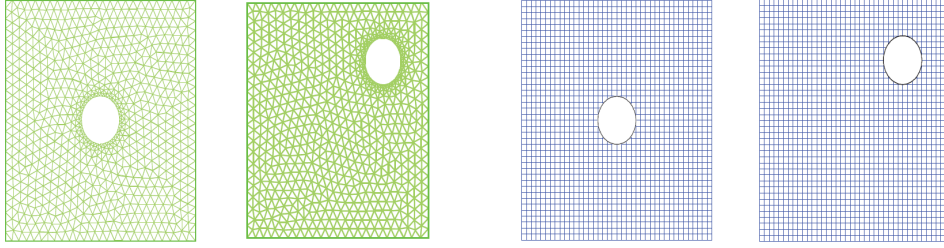
**Figure 2.4:** Representational sketch of explicit coupling 2.4a, and implicit coupling 2.4b approaches to solving partitioned, fluid-structure interaction problems, taken from Münsch [13]

### 2.1.2 Classification based on meshing strategies

The FSI problem is further classified based on the meshing strategies implemented to couple the fluid and structure domains. They are classified into two methods: *conforming mesh* and *non-conforming mesh*

- **Conforming mesh method:** In conforming mesh methods, the interface governing the fluid and structural domains is resolved such that the node connectivity is maintained between the domains. In this method, the interface conditions are considered as physical boundary conditions, which treat the interface location as part of the solution. Owing to the movement and/or deformation of the solid structure, re-meshing (or mesh-update) is required.

- Non-conforming mesh method: The non-conforming mesh methods treat the interface location as constraints imposed on the model equations so that non-conforming meshes (node-connectivity need not be maintained between the domains) can be employed. As a result, the fluid and solid equations can be conveniently solved independently from each other with their respective grids, and re-meshing is not necessary. The distinction between these two types of meshes can be observed in figure 2.5, where a solid body (a sphere) is moving in a fluid domain.



(a) Conforming mesh method representation at time  $t_n$  and  $t_{n+1}$  (b) Non-conforming mesh method representation at time  $t_n$  and  $t_{n+1}$

**Figure 2.5:** Examples of Conforming mesh 2.5a and Non-conforming mesh 2.5b methods, taken from Hou et al. [6].

## 2.2 Governing equations of fluid flow and structural deformations

In Fluid-Structure Interaction problems, the solid moves through the fluid domain due to different forces or excitation. The computation domain changes with time and must be considered during the simulation. An approach which aims to solve this particular problem is Arbitrary Eulerian Lagrangian formulation, popularly referred to ALE formulation. The governing equation for conservation of mass and momentum are reformulated for a moving grid which are time dependent control volume and surface integrals. The governing equations for a Newtonian incompressible fluid in ALE formulation is given as follows.

$$\frac{\partial u_j}{\partial x_i} = 0 \quad (2.1)$$

$$\frac{\partial u_i}{\partial t} + (u_i - v_{g,i}) \frac{\partial u_i}{\partial x_j} = \frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} \quad (2.2)$$

$$\frac{1}{J} \frac{\partial J}{\partial t} - \frac{\partial v_{g,i}}{\partial x_i} = 0 \quad (2.3)$$

The equations presented are *conservation of mass, conservation of momentum and Space*



*Conservation Law* (proposed by Demirdžić and Perić [3]) in ALE formulation in order to account for the moving grid representation. Here  $v_g$  is the grid cell velocity and  $J$  is the determinant of the metric tensor. Numerical treatment of these equations are presented in refer to the relevant section.

### 2.2.1 Coupling of fluid and structural solvers

The FSI problem in this study is solved by a *semi-implicit predictor corrector* coupling scheme, which is a strong coupling method between fluid and structural solver. Details about this scheme is explained in Breuer et al. [2], summary of this scheme is presented in brief as follows:

- (a) At the start of the time step, the displacement of the structure is predicted based on values from the previous time step.
- (b) A predictor corrector approach is implemented to solve for conservation of momentum and mass (velocity  $u_i$  and pressure  $p$ ). This is explained in detail in section refer to relevant section
- (c) The calculated forces are then tranferred to the Computational Structural Dynamics (CSD) solver. Generalized- $\alpha$  method is used to solve the conservation equations of the structural solver. This is explained in detail in section refer to relevant section.
- (d) The FSI-subiterations are performed until desired structural and fluid convergence is achieved for a particular time step. Dynamic calculation of the under-relaxation factor is used to enhance the convergencerefer to relevant section. The corrector step takes the predicted velocity  $\star u_i$  as approximation to obtain the new corrected velocity  $u_i$  and pressure  $p$ .
- (e) Mesh adaptation for each FSI-subiteration is performed, based on transfinite interpolation by Thompson et al. [17]. A comprehensive implementation can be found in Münsch and Breuer [12].

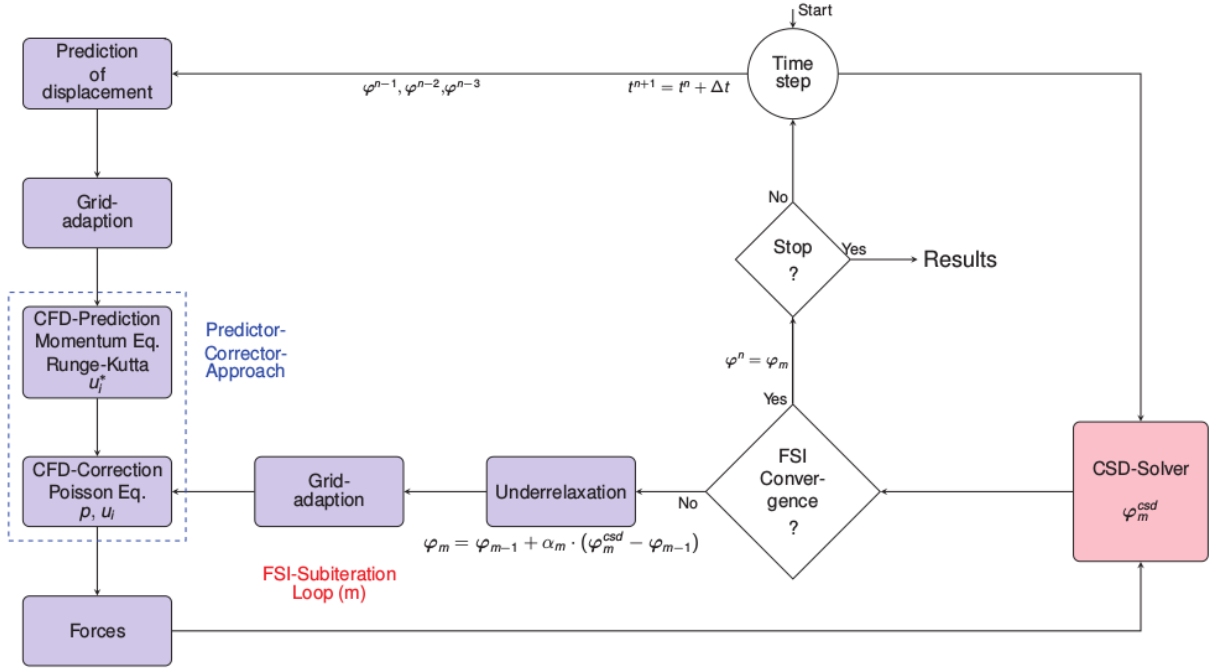
The predcitor-corrector algorithm for a strongly coupled, partitioned FSI solver is represented in the flowchart 2.6

### Convergence of the FSI sub-iterations

As explained in flowchart 2.6 for an implicitly coupled, partitioned FSI solver, the flow field is determined in the actual flow geometry. From this, the friction and pressure forces on the interacting walls are computed. These are boundary conditions to the structural solver. The structural solver computes the deformations, with which the fluid mesh is then modified. Afterwards the flow solver is started again.

The FSI sub-iteration loop is repeated until a convergence criteria  $\epsilon$  is satisfied, which is determined by the variation of the mean displacements 2.4.

## 2.2. GOVERNING EQUATIONS OF FLUID FLOW AND STRUCTURAL DEFORMATIONS



**Figure 2.6:** Flowchart representing the predictor-corrector coupling scheme to solve a strongly coupled, partitioned FSI solver Münsch and Breuer [12]

$$\frac{1}{N} \sum_{n=1}^k \frac{\|\mathbf{d}_i^{n+1} - \mathbf{d}_{i+1}^{n+1}\|^2}{\|\mathbf{d}_{i+1}^{n+1} - \mathbf{d}_{i+1}^n\|^2} < \epsilon \quad (2.4)$$

In order to accelerate the rate of convergence, a relaxation factor is added to the displacement and force values. The resulting displacement is computed from the current sub-iteration displacement value and displacement value from the last sub-iteration. Under-relaxation of the displacements are done as given by the equation 2.5, where  $\omega$  represents the under-relaxation factor.

$$\mathbf{d}_{i+1}^{n+1} = \mathbf{d}_i^{n+1} + \omega_i \left( \tilde{\mathbf{d}}_{i+1}^{n+1} - \mathbf{d}_i^{n+1} \right) \quad (2.5)$$

Relaxation of the interface displacements is synonymous to the line search step of the non-linear solver. There are different methods to calculate the under-relaxation parameter. The most simplest method is by using a *constant under-relaxation parameter* for all time-steps. There are various methods available now to determine the under-relaxation parameter dynamically, however the presented study focuses on *Aitken relaxation method* and relaxation via the method of *steepest descent*. An introduction to these methods is presented below.

- (i) Constant under-relaxation method: The most simplest and ineffective method is to choose a constant  $\omega$  for all time-steps. The relaxation parameter has to be small enough to keep the iteration from diverging, but as large as possible in order to use as much of the new solution as possible and to avoid unnecessary FSI iterations. The

optimal  $\omega$  value is problem specific and not known a priori. Furthermore even the optimal fixed value could lead to more iterations than a suitable dynamic relaxation parameter.

- (ii) Aitken relaxation method: It is an effective and cheap method to calculate the under-relaxation parameter dynamically. The central idea of Aitken's  $\Delta^2$  method is to use values from two previous iterations to improve the current solution, proposed by Irons and Tuck [8]. The aitken factor  $\mu$  2.6 is calculated from the displacement values, and the under-relaxation factor  $\omega$  is calculated as given in the equation 2.9 Mok and Wall [11]. This method has been implemented and validated with many FSI applications Irons and Tuck [8], Küttler and Wall [9]. (Refer some more papers which uses aitken) The relaxation parameter calculation is valid in every FSI sub-iteration step. This method requires values from two previous sub-iterations, thus the relaxation parameter can be calculated after the first FSI sub-iteration. In the presented study the aitken factor from last time-step is used as an initial value for the current time-step ( $\mu_{i_{max}}^n = \mu_1^{n+1}$ ) as proposed by Irons and Tuck [8].

$$\mu_i^{n+1} = \mu_{i-1}^{n+1} + (\mu_{i-1}^{n+1} - 1) \frac{(\Delta \mathbf{d}_i^{n+1} - \Delta \mathbf{d}_{i+1}^{n+1})^T \Delta \mathbf{d}_{i+1}^{n+1}}{(\Delta \mathbf{d}_i^{n+1} - \Delta \mathbf{d}_{i+1}^{n+1})^T (\Delta \mathbf{d}_i^{n+1} - \Delta \mathbf{d}_{i+1}^{n+1})} \quad (2.6)$$

$$\mathbf{d}_i^{n+1} = \mathbf{d}_{i-1}^{n+1} - \mathbf{d}_i^{n+1} \quad (2.7)$$

$$\mathbf{d}_{i+1}^{n+1} = \mathbf{d}_i^{n+1} - \tilde{\mathbf{d}}_{i+1}^{n+1} \quad (2.8)$$

$$\alpha_i = 1 - \mu_i^{n+1} \quad (2.9)$$

- (iii) Steepest-Descent method: The best relaxation parameter possible is the one that finds the optimal step length in  $r_{i+1}^{n+1}$  direction. In order to obtain that, the existence of a merit function  $\varphi$  is assumed, that is minimal at the solution  $d_{i+1}^{n+1}$  and sufficiently smooth, such that the relaxation parameter  $\omega_i$  is given by equation 2.10. By Taylor's series expansion and assumption of connection between the merit function and interface residual, the final expression for the relaxation parameter is given by the equation 2.11. The steps in obtaining this expression is explained clearly in Küttler and Wall [9].

$$\omega_i = \arg \min_{\omega_i} \varphi(d_i^{n+1} + \omega_i r_{i+1}^{n+1}) \quad (2.10)$$

$$\omega_i = - \frac{(\mathbf{r}_{i+1}^{n+1})^T (\mathbf{r}_{i+1}^{n+1})}{(\mathbf{r}_{i+1}^{n+1})^T \mathbf{J} (\mathbf{r}_{i+1}^{n+1})} \quad (2.11)$$

In the equation 2.11,  $\mathbf{J}$  represents the Jacobian matrix of the interface residual ( $\mathbf{J} = \varphi''(\mathbf{d}_i^{n+1})$ ). This interface Jacobian is not readily available, there are methods to obtain the matrix vector product which can be performed with black-box solvers Küttler and Wall [9]. In the presented study, the matrix vector product is obtained by means of Finite-Difference method as expressed in equation 2.12.

$$\mathbf{J}\mathbf{r}_{i+1}^{n+1} = \frac{\mathbf{fsi}_{\text{sub}}(\mathbf{d}_i^{n+1} + \delta\mathbf{r}_{i+1}^{n+1}) - \mathbf{d}_i^{n+1} - \delta\mathbf{r}_{i+1}^{n+1} - \mathbf{r}_{i+1}^{n+1}}{\delta} \quad (2.12)$$

$$\delta = \lambda \left( \lambda + \frac{|\mathbf{d}_i^{n+1}|}{|\mathbf{r}_{i+1}^{n+1}|} \right) \quad (2.13)$$

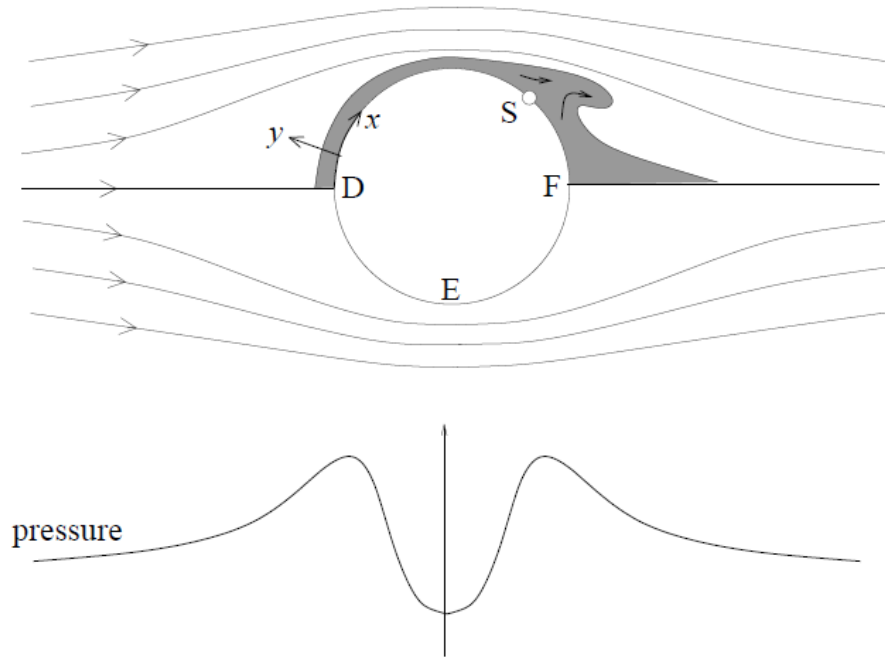
A small  $\lambda$  is assumed in the study, this method requires one more fsi-sub iteration cycle in order to evaluate the interface Jacobian. The computational costs for this method is slightly higher.

## 2.3 Flow separation and vortex shedding

The presence of the fluid viscosity slows down the fluid particles very close to the solid surface and forms a thin slow-moving fluid layer called a boundary layer. The flow velocity is zero at the solid surface to satisfy the no-slip boundary condition. Inside the boundary layer, flow momentum is quite low since it experiences a strong viscous flow resistance. Therefore, the boundary layer flow is sensitive to the external pressure gradient (as the form of a pressure force acting upon fluid particles). If the pressure decreases in the direction of the flow, the pressure gradient is said to be favorable. In this case, the pressure force can assist the fluid movement and there is no flow retardation. However, if the pressure is increasing in the direction of the flow, an adverse pressure gradient condition exists. In addition to the presence of a strong viscous force, the fluid particles now have to move against the increasing pressure force. Therefore, the fluid particles could be stopped or reversed, causing the neighboring particles to move away from the surface. This phenomenon is called the boundary layer separation or flow separation.

In fig.2.7 there exists a pressure gradient  $\frac{\partial p}{\partial x} < 0$  to accelerate the fluid along the surface  $x$  from D to E and a gradient  $\frac{\partial p}{\partial x} > 0$  to decelerate the fluid between E and F. Potential energy (PE) in the pressure field is converted to kinetic energy between D and E, then fully back to PE between E and F, because there is no source of dissipation in an inviscid fluid.

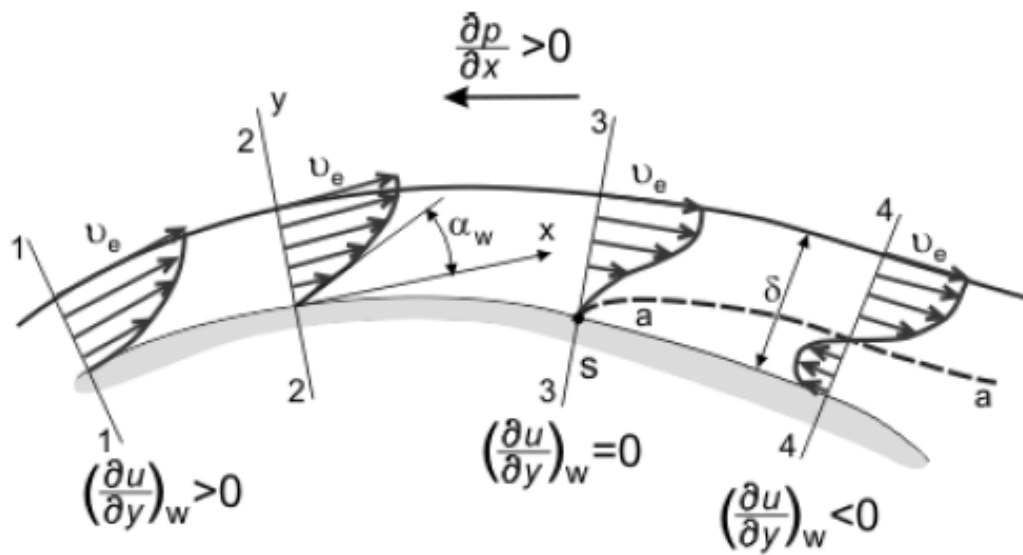
In the inviscid region, the fluid momentum is sufficient to overcome the adverse pressure gradient barrier and reach point F with the same kinetic energy as at point D. However, inside the boundary layer (near the surface of the cylinder) energy is dissipated by the viscous drag. As the momentum is smaller near the wall than in the regions closer to the free stream, a fluid particle on the boundary layer can not overcome the pressure barrier and comes to a standstill and eventually gets pushed backwards into motion by the pressure distribution of the outer flow. This continuous retardation of the flow brings the wall shear stress  $\tau_w$  to zero at the point S on the wall as shown in figure 2.8. The shear stress starts to become negative after the point S, which is the separation point. The boundary layer separation point is given by the mathematical relationship in equation



**Figure 2.7:** Pressure field and separation point for flow past a cylinder.

2.14.

$$\tau_w = \mu \left( \frac{\partial u}{\partial y} \right) \quad (2.14)$$



**Figure 2.8:** point of separation and Wall Shear stress,ref Schlichting et al. [16]

# 3

## IMPLEMENTATION

---

Some complex code is shown in Figure 3.1.

---

```

static int __init serpent_init(void)
{
    u64 xcr0;
    if (!cpu_has_avx || !cpu_has_osxsave) {
        printk(KERN_INFO "AVX instructions are not detected.\n");
        return -ENODEV;
    }
    xcr0 = xgetbv(XCR_XFEATURE_ENABLED_MASK);
    if ((xcr0 & (XSTATE_SSE | XSTATE_YMM)) != (XSTATE_SSE | XSTATE_YMM)) {
        printk(KERN_INFO "AVX detected but unusable.\n");
        return -ENODEV;
    }
    return crypto_register_algs(serpent_algs, ARRAY_SIZE(serpent_algs));
}

static void __exit serpent_exit(void)
{
    crypto_unregister_algs(serpent_algs, ARRAY_SIZE(serpent_algs));
}

module_init(serpent_init);
module_exit(serpent_exit);

MODULE_DESCRIPTION("Serpent Cipher Algorithm, AVX optimized");
MODULE_LICENSE("GPL");
MODULE_ALIAS("serpent");

```

**Figure 3.1:** Serpent AVX module initialization



# 4

## EVALUATION

---



# 5

## CONCLUSION AND FUTURE WORK

---

In this chapter we want to draw conclusions about the work, which has been done during this thesis.



## BIBLIOGRAPHY

---

- [1] P. Becker, S. R. Idelsohn, and E. Oñate. A unified monolithic approach for multi-fluid flows and fluid–structure interaction using the particle finite element method with fixed mesh. *Computational Mechanics*, 55(6):1091–1104, dec 2014. doi: 10.1007/s00466-014-1107-0.
- [2] Michael Breuer, Guillaume De Nayer, Manuel Münsch, Thomas Gallinger, and Roland Wüchner. Fluid–structure interaction using a partitioned semi-implicit predictor–corrector coupling scheme for the application of large-eddy simulation. *Journal of Fluids and Structures*, 29:107–130, 2012.
- [3] I Demirdžić and M Perić. Space conservation law in finite volume calculations of fluid flow. *International journal for numerical methods in fluids*, 8(9):1037–1050, 1988.
- [4] Charbel Farhat and Michael Lesoinne. Two efficient staggered algorithms for the serial and parallel solution of three-dimensional nonlinear transient aeroelastic problems. *Computer methods in applied mechanics and engineering*, 182(3-4):499–515, 2000.
- [5] Carlos A Felippa, KC Park, and Charbel Farhat. Partitioned analysis of coupled mechanical systems. *Computer methods in applied mechanics and engineering*, 190(24-25):3247–3270, 2001.
- [6] Gene Hou, Jin Wang, and Anita Layton. Numerical methods for fluid-structure

- interaction—a review. *Communications in Computational Physics*, 12(2):337–377, 2012.
- [7] Björn Hübner, Elmar Walhorn, and Dieter Dinkler. A monolithic approach to fluid–structure interaction using space–time finite elements. *Computer methods in applied mechanics and engineering*, 193(23-26):2087–2104, 2004.
- [8] Bruce M Irons and Robert C Tuck. A version of the aitken accelerator for computer iteration. *International Journal for Numerical Methods in Engineering*, 1(3):275–277, 1969.
- [9] Ulrich Küttler and Wolfgang A Wall. Fixed-point fluid–structure interaction solvers with dynamic relaxation. *Computational Mechanics*, 43(1):61–72, 2008.
- [10] C Michler, SJ Hulshoff, EH Van Brummelen, and René De Borst. A monolithic approach to fluid–structure interaction. *Computers & fluids*, 33(5-6):839–848, 2004.
- [11] Daniel P Mok and WA Wall. Partitioned analysis schemes for the transient interaction of incompressible flows and nonlinear flexible structures. *Trends in computational structural mechanics*, 49:689–698, 2001.
- [12] M Münsch and M Breuer. Numerical simulation of fluid–structure interaction using eddy–resolving schemes. In *Fluid Structure Interaction II*, pages 221–253. Springer, 2011.
- [13] Manuel Münsch. Entwicklung und anwendung eines semi-impliziten kopplungsverfahrens zur numerischen berechnung der fluid-struktur-wechselwirkung in turbulenten strömungen mittels large-eddy simulationen. 2015.
- [14] Serge Piperno, Charbel Farhat, and Bernard Larroutrou. Partitioned procedures for the transient solution of coupled aroelastic problems part i: Model problem, theory and two-dimensional application. *Computer methods in applied mechanics and engineering*, 124(1-2):79–112, 1995.
- [15] Thomas Richter. A monolithic multigrid solver for 3d fluid-structure interaction problems. 2010.
- [16] Hermann Schlichting, Klaus Gersten, Egon Krause, Herbert Oertel, and Katherine Mayes. *Boundary-layer theory*, volume 7. Springer, 1955.
- [17] Joe F Thompson, Zahir UA Warsi, and C Wayne Mastin. *Numerical grid generation: foundations and applications*, volume 45. North-holland Amsterdam, 1985.
- [18] EH Van Brummelen, SJ Hulshoff, and R De Borst. Energy conservation under incompatibility for fluid–structure interaction problems. *Computer Methods in Applied Mechanics and Engineering*, 192(25):2727–2748, 2003.