DEVELOPMENT OF A THERMAL FLOW SOLVER USING REMESHED VORTEX METHODS

BY

NIKHIL ANTO VALLUVAN

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Aerospace Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2018

Urbana, Illinois

Adviser:

Professor Mattia Gazzola

Abstract

Fluid structure interaction problems involving heat transfer are ubiquitous in engineering applications. Computational techniques are often required to obtain usable information about these problems. A Remeshed Vortex Method with immersed boundaries is presented capable of handling Fluid Structure Interaction problems along with moving boundaries. A novel solver is developed based on these principles in C++ to solve the Navier Stokes equations for incompressible flows. The Boussinesq approximation is used to model the thermal behaviour of the flow. This solver is then validated by solving for the flow past an isothermal heated cylinder. Different test cases where free, forced and mixed convection dominate are studied. The results obtained are compared to those in literature to validate the code. The ability of the code to capture moving boundaries is also verified. Following this, potential applications and scope for future work is identified.

To appa amma

Acknowledgements

I would like to take this opportunity to express my gratitude to my adviser Professor Mattia Gazzola for introducing me to the field of Immersed Boundary Methods and Particle Techniques. I enjoyed the opportunity to explore this vast and rich field which kept me engaged and stimulated over the duration of my Masters degree. I am also grateful for the discussions and guidance provided which helped me maneuver the complicated subject matter whenever I hit a roadblock.

I would also like to thank my lab mates Tejaswin Parthasarathy and Fan Kiat Chan for their help in getting me started quickly on the path towards results in my chosen problem. I would not have been able to reach my end goal within the 12 months I had available to me without their help in navigating the multi disciplinary topics involved in CFD.

Lastly I wish to thank all the friends I made during my stay here at UIUC for all the support that they provided. I would be remiss if I didn't acknowledge my family who were a constant source of encouragement (Almost to a fault!).

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Many important engineering problems involve the interplay between solid and fluid domains. Some examples include the design of aircraft engines and structures, bridges, ships, wind turbines and so on. Problems of this nature are often grouped into a varied class called Fluid Structure Interaction (FSI) problems. Since the nature and the dynamics of these problems are so rich and complex, a computational approach to these problems is usually adopted, often due to a lack of choice. When modeling the fluid domain typical computational approaches, like the Finite Volume approach which is the most common implementation found in commercial Computer Aided Engineering(CAE) packages, are often complicated or are rendered invalid in many common FSI problems of interest. This occurs especially in situations that involve moving solid boundaries. The response to this is to often use complicated algorithms, to preserve the validity of the generated mesh, that are not very efficient from a computational standpoint. These conventional approaches are usually referred to as body fitted techniques and cover most Finite Volume and Finite Difference techniques. An additional problem is often presented when a complicated geometry is to be modeled in addition to these moving boundaries. These complex geometries are often tackled using unstructured mesh approaches in Finite Volume and Finite Element formulations. However, the problem of dealing with moving boundaries still remains for the most part. Moving boundaries often require the mesh to be regenerated at regular time intervals to preserve the accuracy and stability of the solver.

1.1 Immersed Boundary Methods

One potential solution to this problem that has been explored is the idea of having grids that do not conform to the body of the solid. There are many different ideas that have been suggested and implemented with varying success. This wide class of methods are often called Immersed Boundary Methods(IBM). The first instance of an IBM is often credited to Peskin [21] with many different additions and innovations added since. The reader is referred to [12,18] for a review of Immersed boundary methods.

1.1.1 Thermal flows

Immersed boundary methods to solve thermal flows are seldom seem in the Literature and rarer still are those that rely on particle methods [6,14,27,28]. Lagrangian approaches for such problems involving thermal flows or the transport of other scalars have often adopted techniques such as the Lattice Boltzman method or vortex element method/transport element method (particle methods). The disadvantage with most of these implementations is the sheer complexity of the algorithm and the codes required to fully capture the flow dynamics. In many implementations of particle methods it is seen that it can be problematic to capture diffusion [8, 25]. Often times this results in code bases that cannot be taken advantage of without access to massive computational resources well outside the reach of personal computers. The main advantage of the method adopted in this work, the Remeshed Vortex Method (RVM), is the simplicity and elegance of the underlying principles. The basic idea behind particle methods as a whole is that the avection operator in the flow equations can be simulated by means of particles that transport the flow quantities of interest. Also, diffusion can be modeled using a simple finite difference scheme of required order on a fixed grid. There is no documented work at the time of writing this thesis that I am aware of which utilizes a similar RVM approach to study thermal flows. The development of such an approach will be the focus of this thesis.

It is also worth pointing out that while this code is geared towards dealing with two way coupled Fluid Structure Interaction problems, only the flow physics will be studied in this work. The results from this solver is not coupled to any Finite Element solvers that model the physics in the solid domain. As such, the solid may be considered to be rigid and non-deformable in all the test cases presented in this work. The solver is written entirely in C++ with largely pre-existing building blocks and tools. Flow visualization and post processing is achieved using open source code libraries and software.

The focus of this work, will be the development of the tools and techniques and putting them together. This is followed by the validation of the method. While there are many potential applications for such a technique, none will be explored or discussed in any significant level of detail.

1.2 Structure of this Thesis

In the second chapter of this document, the basic mathematical model used to capture the physics of the flow domain are presented. The salient features of the model and its shortcomings are highlighted and the Remeshed Vortex Method (RVM) is elaborated. Finally the governing equations in their final form are presented that must then be solved computationally.

In Chapter 3, the algorithm to solve the governing equations obtained in the previous chapter is presented.

Some details regarding the code utilized and tools developed are also highlighted. Some issues encountered with manipulation of the flow field variables is also discussed and a solution presented.

In Chapter 4, the developed code is used to tackle a few canonical test problems that capture the physics associated with the the flow. The results obtained for these test cases are presented. They are then compared with experimental and numerical results available in literature to test the accuracy and validity of the method. In addition to this,the ability of the code to deal with moving solid bodaries is also tested.

In Chapter 5, potential applications of the developed code are highlighted and avenues for further work are identified.

Chapter 2

Mathematical formulation

2.1 Governing equations

The governing equations for all the problems presented in this thesis are the incompressible viscous Navier Stokes equations in two dimensions with the Boussinesq approximation:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u} = -\frac{1}{\rho}\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{g}\beta(T - T_\infty) \tag{2.1}$$

Here, $\mathbf{u}$ is the velocity vector field, t is the time variable, ρ is the fluid density, p is the pressure field, ν is the viscosity of the fluid (momentum diffusivity), $\mathbf{g}$ is the gravity vector, β is the coefficient of thermal expansion, T is the temperature scalar field and T_∞ is the temperature of the bulk fluid.

$$\nabla \cdot \mathbf{u} = 0 \tag{2.2}$$

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \alpha \nabla^2 T \tag{2.3}$$

Where α is the thermal diffusivity of the fluid.

Equations 2.1 and 2.2 may be combined and represented in the velocity-vorticity formulation as follows:

$$\frac{\partial \boldsymbol{\omega}}{\partial t} + \nabla \cdot (\mathbf{u} : \boldsymbol{\omega}) = (\boldsymbol{\omega} \cdot \nabla)\mathbf{u} + \nu \nabla^2 \boldsymbol{\omega} + \beta \nabla (T - T_\infty) \times \mathbf{g} \tag{2.4}$$

Where $\boldsymbol{\omega}$ is the vorticity vector field. In the case of two dimensional flow it reduces to a scalar field(A vector with the only component being normal to the plane of the flow). In recasting the governing equations in the vorticity-stream function form, we have eliminated the pressure variable from the equations. This greatly reduces the complexity of the solver. The tricky pressure poisson equation that one typically encounters in incompressible flow solvers is replaced instead with a much simpler Poisson equation for the stream function which can be solved by means of an efficient Fast Fourier Transform on multipoles(FFTM) solver.

The first term in 2.4 represents the unsteady term that captures the transients. The second term is the advection of vorticity. The third term is the stretching term which vanishes in two dimensions and the fourth represents the diffusion of vorticity.

Also, it must be noted that since the focus of this study is restricted to the study of adopting remeshed vortex methods to thermal flows, only neutrally buoyant solids are considered in this study. For this reason the baroclinic term is omitted.

Then we may simply use the definition of the stream function (ψ) given by 2.5 to recoup the velocity by solving the Poisson equation 2.6 for the stream function with suitable boundary conditions.

$$\mathbf{u} = \nabla \times \boldsymbol{\psi} \tag{2.5}$$

$$\nabla^2 \boldsymbol{\psi} = -\boldsymbol{\omega} \tag{2.6}$$

Note that we have also eliminated the divergence condition from the velocity-pressure formulation of the governing equations. Instead, this divergence free condition is implicitly satisfied by the definition of the stream function. The numerical method employed to computationally solve these equations is the remeshed vortex method as mentioned previously. This technique is further elaborated in the next section.

Now to obtain a unique solution for the problem, we impose boundary conditions that accurately model the physical problem under study. In this study, these are usually unbounded domains that represent an infinite domain of fluid stretching out in all directions. An additional advantage of using the vorticity-stream function formulation is the compact support of vorticity in capturing the flow physics in an infinite domain. It is possible to impose periodic boundary conditions as well if required. One potential drawback of this technique however is the difficulty associated with modeling bounded domains with no outflow.

2.1.1 Boussinesq approximation

It is worth noting at this point that the Boussinesq approximation is employed in modeling the problem which introduces a few assumptions and restrictions of its own. The main idea behind this simplification is that the coupling between the temperature and the momentum conservation equations is only through the buoyancy term. That is the effect of density changes in the fluid due to increase in temperature are ignored except for the change it causes in the specific weight and the resulting buoyant force that acts on a fluid volume. These density variations are also modeled as a linear function of temperature with the coefficient of expansion β dictating the fluid property. One must keep in mind however, that this model breaks down when very high temperature gradients and phase changes are encountered [26]. As seen in 2.1, the resulting force is given by

$$\mathbf{F}_{buoyancy} = \frac{\partial \mathbf{u}}{\partial t} = \mathbf{g}\beta(T - T_{\infty}) \tag{2.7}$$

The curl of this term is then taken to obtain the resulting force in the velocity-vorticity formulation. This is given by the following term as seen in 2.4

$$\frac{\partial \boldsymbol{\omega}}{\partial t} = \beta\nabla(T - T_{\infty}) \times \mathbf{g} \tag{2.8}$$

Note that β is assumed to be a constant (does not vary with temperature) here.

2.2 Remeshed vortex method

The remeshed vortex method in the form that is used in this work is largely based on the work in [7]. It is similar to other particles solvers in that advection is modeled using the motion of pseudo particles that transport flow properties with the fluid velocity at that region. It is worth noting that these particles do not have any mass or momentum of their own. In most implementations of a Lagrangian solver these particles are allowed to evolve freely with the flow and new particles are generated as required. The problem with this method is that often an enormous number of particles are generated that it is no longer viable to keep track of all these particles. Another issue that arises is these particles tend to concentrate in regions of higher vorticity (In the case of vorticity being the quantity that is transported). This can result in the particles moving away from each other violating the overlapping condition. This can result in the solver becoming unstable as explained in later sections. In order to overcome this, in the remeshed vortex method, the particles are moved back onto a regular grid at the end of each time step. Put another way, we generate and delete an equal number of particles at the beginning of each time step. In this way we no longer generate new particles that could potentially grow to huge numbers and as a result unwieldy memory requirements. It also results in a less restrictive time step criterion for the advection term as we will see later. Typically, with Vortex Element Methods and Transport Element Methods one also finds that diffusion is quite difficult to model, with most implementations resorting to techniques like random walks or Monte Carlo simulation techniques. Of course, the disadvantage with doing that is the large number of particles that need to be simulated to accurately capture the underlying randomness of diffusion. There have been a few solutions proposed that involve complicated particle strength exchanges to capture diffusion but the algorithms to accomplish this are not very conducive to parallelization. In the case of RVM, the existence of a regular underlying grid allows us to split the operators of the governing equation and solve for diffusion using Finite differences [1,8,16,25].

2.2.1 Advection

As mentioned in the previous section the fluid domain is broken down into component pseudo particles. Each of these particles carry information about the flow quantity of interest. The flow can then be reconstructed from these particles using the following formulation. For instance in the case of vorticity, the field can be constructed at any given location as follows:

$$\boldsymbol{\omega}_\epsilon^h(\mathbf{x}) = \sum_p \Gamma_p \zeta_\epsilon(\mathbf{x} - \mathbf{x}_p) \tag{2.9}$$

Where $\Gamma_p = \int_{V_p} \boldsymbol{\omega} d\mathbf{x}$ is the strength of the vortex, ζ_ϵ is the interpolation kernel and ϵ is the particle core size.

Similarly the temperature field at a location can be reconstructed as follows:

$$T_\epsilon^h(\mathbf{x}) = \sum_p T_p \zeta_\epsilon(\mathbf{x} - \mathbf{x}_p) \tag{2.10}$$

Now from the governing equations given by 2.4, we can formulate the following equations that describe the evolution of the positions of the particles($\mathbf{x_p}$) and the vorticity carried by these particles. Note that these are ODEs that can be solved by an appropriate numerical scheme.

$$\begin{aligned}
\frac{d\mathbf{x_p}}{dt} &= \mathbf{u}(\mathbf{x_p}) \\
\frac{D\boldsymbol{\omega}}{Dt} &= (\boldsymbol{\omega} \cdot \nabla)\mathbf{u} + \nu\nabla^2\boldsymbol{\omega} \\
\frac{DT}{Dt} &= \nu\nabla^2 T
\end{aligned} \tag{2.11}$$

The equations given by 2.11, 2.9 and 2.10 are utilized in some form or another in most vortex particle methods [17,23]. The disadvantage however in using this form is that it may lead to the particles grouping together in specific regions depending on the governing physics and thus distorting the flow. It also violates what's called the overlapping condition, to be discussed later, rendering the solver unstable. In order to overcome this a remeshing strategy has been adopted. This forms the basis of the remeshed vortex method.

2.2.2 Remeshing

Once the particles have been advected in a manner dictated by the governing physics 2.11 for the duration of one time step, we then remesh the particles back to the underlying regular grid at the end of each time step.

This process of remeshing is nothing more than the interpolation of the transported quantity back to the grid nodes. In effect we create new particles at the nodes to represent the flow field while at the same time discarding the previous particles that are distorted to some degree. This is done using the third order MP4 interpolation kernel [19]. Higher order interpolation kernels can also be derived systematically as needed.

$$M_4'(x) = \begin{cases} \frac{1}{2}(|x|-1)(3|x|^2-2|x|-2) & |x| < 1 \\ -\frac{1}{2}(|x|-1)(|x|-2)^2 & 1 \leq |x| < 2 \\ 0 & 2 \leq |x| \end{cases} \tag{2.12}$$

Another advantage of using this regular grid is that it allows for an efficient Fast Fourier Transform solver to be used to solve the Poisson equation 2.6 for the stream function.

2.2.3 Stability: Overlapping condition

Since we do not discretize the advection term in a conventional manner, we are no longer bound by the linear CFL (Courant Friedrichs Levy) time step restrictions. However, as mentioned in the previous section, the particle field representation introduces the overlapping condition that must be satisfied. This may be thought of as a Lagrangian CFL criteria. The requirement is as follows: In a remeshed vortex method the particles are required to not move apart from each other by no more than a constant distance C in a given time step. This constant is dictated by the interpolation kernel chosen.

$$\Delta t|\dot{h}| \leq C \tag{2.13}$$

Where $\dot{h}$ is the rate of growth of the distance between particles and is given by $|\dot{h}| = h|\nabla u|$. Note that at the start of each time step, h simply corresponds to the grid spacing. Now we may express C as a function of h as follows,

$$\Delta t \leq C_1||\nabla \otimes \mathbf{u}||_\infty^{-1} \approx C_2||\boldsymbol{\omega}||_\infty^{-1} \tag{2.14}$$

Here, C_1 and C_2 depend on the problem and the mesh. As mentioned earlier this may be recast as a Lagrangian CFL as follows:

$$LCFL = \Delta t||\nabla \otimes \mathbf{u}||_\infty \tag{2.15}$$

It is usually the case that 2.15 results in a less restrictive time step than the typical CFL criterion for the advection operator. Results verifying this are presented int later sections of the document.

2.2.4 Geometry representation

The geometry is represented by constructing the characteristic function (χ_s). This is a function that contains unit values for grid nodes that contain the solid and zero for all other grid nodes. The surface of the characteristic function is also mollified. This is simply the smoothing out of the values at the surface to avoid any adverse aliasing effects from immersion in a regular grid like stair stepping. The length scale over which this process is carried out is chosen at the discretion of the user depending on the nature of the problem. The mollified characteristic function is constructed as follows,

$$\chi_s = \begin{cases} 0 & d < -\epsilon \\ \frac{1}{2}(1 + \frac{d}{\epsilon} + \frac{1}{\pi} sin(\pi \frac{d}{\epsilon}) & |d| \leq \epsilon \\ 1 & d > \epsilon \end{cases} \tag{2.16}$$

Here, d is the distance between the surface of the body and the point under consideration. It is negative outside and positive inside the surface of the solid. ϵ is the mollification length which is usually a small fraction of the characteristic length of the problem. It essentially represents the length scale of the smallest features that can be captured in the solid.

2.2.5 Penalization

We can then apply the required boundary conditions by using a suitable penalization term. This is simply a forcing term introduced in the governing equation that injects the required momentum and heat into the domain to cause the boundary conditions of no slip and isothermal heating to be satisfied [9,13]. It must be noted that non-isothermal boundaries may also be modeled using this technique. However, for the purposes of this article, only isothermal boundaries are considered. The penalization term for the momentum and heat equations take the following form in two dimensions:

$$\frac{\partial \omega}{\partial t} = \lambda \nabla \times \chi_S (\mathbf{u}_S - \mathbf{u}) \tag{2.17}$$

$$\frac{\partial T}{\partial t} = \lambda_T \chi_S (T_s - T) \tag{2.18}$$

As seen from the form of the terms, the penalization is only active in the solid regions and can be thought of a feedback term that forces the flow quantity being transported to the required value dictated by the boundary conditions. As previously mentioned, this technique is not limited to Dirichlet boundary conditions. However, all models constructed in this article will be restricted to Dirichlet boundaries which takes the form presented in 2.17 and 2.18.

A fictitious fluid also extends to inside the solid region where the penalization term is active. Here, the penalization factor(λ) can be thought of as modeling the porosity of the solid in question and higher values correspond to a greater accuracy in the penalized solution. This however also incurs a greater computational expense and it is often chosen taking into consideration the Reynolds number of the flow in question. A value of 10^4 is used for all the problems presented in this study (For both λ and λ_T). This value was found to strike an acceptable balance between computational expense and accuracy.

2.3 Final form of equations

Thus, the final governing equations along with the penalization terms included take the following form for two dimensions. Note that the stretching term from 2.4 vanishes in two dimensions and vorticity reduces to a scalar.

$$\frac{\partial \omega}{\partial t} + \nabla \cdot (\mathbf{u}\omega) = \nu \nabla^2 \omega + \lambda \nabla \times \chi_s(\mathbf{u}_s - \mathbf{u}) \tag{2.19}$$

$$\frac{\partial T}{\partial t} + \mathbf{u} \cdot \nabla T = \alpha \nabla^2 T + \lambda_T \chi_s (T_s - T) \tag{2.20}$$

The velocity of the solid is decomposed into its constituent translational($\mathbf{u}_T$), rotational($\mathbf{u}_R$) and deformation($\mathbf{u}_{DEF}$) velocities as shown below

$$\mathbf{u}_s = \mathbf{u}_T + \mathbf{u}_R + \mathbf{u}_{DEF} \tag{2.21}$$

These velocities, depending on the problem under study, maybe pre defined or a result of the momentum and hydrodynamic forces acting on the solid. In the case of a two way coupled FSI problem, a projection approach is adopted to determine these velocities, where the flow is evolved freely for half a time step. The deformation velocity is only required when the solid deforms in such a way that volume is not preserved. This situation will not be encountered in any of the test cases presented in this document. The algorithm for solving these equations computationally is presented in the following chapter.

Chapter 3

Structure of Code

As mentioned in the previous chapter 2.19 and 2.20 represents the final form of the equation that is to be solved. The methodology adopted can be summarized into the following steps that represent all the computations performed in one complete time step. These steps are all repeated for each time step of course as we march in time.

3.1 Algorithm

3.1.1 Generation of characteristic function

First the characteristic function for the solid under study is generated. This is done using 2.16

3.1.2 Solution of Poisson equation for stream function

The Poisson equation is then solved using a Fast Fourier Transform Solver to obtain a solution for the Poisson equation given by 2.6

3.1.3 Calculate velocity from stream function

The stream function obtained by solving 2.6 is then used to calculate the velocity field using 2.5. It must be noted here that in the case of deforming bodies, the divergence of the velocity field inside the body may need to be computed to ensure a consistent velocity field. However, we will not be looking at any problems where this is the case.

3.1.4 Determine the size of the time step Δt

The size of the time step is determined to ensure stability of the solver. The criteria that are usually looked at is the overlapping condition which inform the LCFL given by 2.15. The vorticity field from the previous time step is used for this purpose. In addition to this the stability criteria for the diffusion operator split is also ensured for both vorticity and temperature diffusion. The condition is given by:

$$\Delta t \leq \frac{\Delta x^2}{4 \times \mathcal{D}} \tag{3.1}$$

Where $\mathcal{D}$ from the above expression corresponds to the diffusivity. This may of course be thermal or momentum diffusivity depending on the equation being considered. The smaller value for the time step is chosen from all three considerations. Also, it is worth noting that in the case of the initial few time steps, a very small value is chosen to overcome any potential instabilities arising from artifacts due to modeling an impulsive start with infinite acceleration. The number and size of these time steps are arbitrary and chosen purely at the discretion of the user based on the specifics of the geometry.

3.1.5 Calculate solid boundary velocity

The velocity of the solid boundary is calculated from the decomposed velocities as given by 2.21. In the case of a two way coupled FSI problem the translational velocity $\mathbf{u}_T$ is calculated by a projection approach by allowing the fluid to evolve freely for a part of the time step. This is given by the relation:

$$\mathbf{u}_T^n = \frac{1}{M_s} \int_\Sigma \rho^n \chi_s^n \mathbf{u}^n d\mathbf{u} \tag{3.2}$$

A similar approach is adopted for the rotational velocity with one minor change. The angular velocity of rotation is determined first before the linear analog can be determined.

$$\dot{\theta}^n = \frac{1}{J_s^n} \int_\Sigma \rho^n \chi_s^n (\mathbf{x} - \mathbf{u}_{cm}^n) \times \mathbf{u}^n d\mathbf{x} \tag{3.3}$$

$$\mathbf{u}_R^n = \dot{\theta}^n \times (\mathbf{x} - \mathbf{u}_{cm}^n) \tag{3.4}$$

The deformation velocity is usually prescribed if required by the problem. This situation is typically encountered when simulating bio-inspired flows like swimming fish.

3.1.6 Penalize the velocity

Once the solid boundary velocity is known from the previous step, the entire velocity field is penalized using an implicit first order euler scheme. The final equation for this scheme is given by:

$$\mathbf{u}_\lambda^n = \frac{\mathbf{u}^n + \lambda \Delta t \chi_s^n (\mathbf{u}_T + \mathbf{u}_R + \mathbf{u}_{DEF})}{1 + \lambda \Delta t \chi_s^n} \tag{3.5}$$

3.1.7 Penalize the temperature

In a manner similar to the previous section, we penalize the entire temperature field given the isothermal wall temperature. The same first order Euler scheme is used as before resulting in the relation:

$$T_\lambda^n = \frac{T^n + \lambda \Delta t \chi_s^n T_s^n}{1 + \lambda \Delta t \chi_s^n} \tag{3.6}$$

3.1.8 Update vorticity given new penalized velocity field

The vorticity field is then updated using the definition of vorticity from the new penalized velocity field. The equation for this is given by:

$$\omega_\lambda^n = \nabla \times \mathbf{u}_\lambda^n \tag{3.7}$$

3.1.9 Update vorticity for the diffusion split

The updated vorticity field from the previous step is then used to compute the following equation that represents the diffusion split for the vorticity evolution equation

$$\frac{\partial \omega_\lambda^n}{\partial t} = \nu \nabla^2 \omega_\lambda^n \tag{3.8}$$

Here, a second order Runge Kutta(RK2) scheme is used to march in time. Explicit second order stencils are used for the second derivatives in space.

In the case of a two way coupled Boussinesq problem, the buoyancy term is also included in this split. Once again, the RK2 scheme is used to march in time. In this case the equation that is solved numerically becomes:

$$\frac{\partial \omega_\lambda^n}{\partial t} = \nu \nabla^2 \omega_\lambda^n + \beta \nabla (T - T_\infty) \times \mathbf{g} \tag{3.9}$$

3.1.10 Update temperature for the diffusion split

The diffusion split for the temperature field is performed in a manner identical to vorticity. The equation is given as follows:

$$\frac{\partial T_\lambda^n}{\partial t} = \alpha \nabla^2 T_\lambda^n \tag{3.10}$$

Similar to the vorticity split, a second order Runge Kutta scheme is used to march in time and second order spatial discretization is used resulting in an explicit finite difference scheme.

3.1.11 Advection split

The advection time step forms the core of the Remeshed Vortex Method. Here, the particles at the grid points are advected with the fluid velocity carrying the temperature and vorticity information at that point. The equations that govern this is given by,

$$\frac{\partial \omega_{\lambda}^{n}}{\partial t} + \nabla \cdot (\mathbf{u}_{\lambda}^{n} \omega_{\lambda}^{n}) \tag{3.11}$$

$$\frac{\partial T_{\lambda}^{n}}{\partial t} + \nabla \cdot (\mathbf{u}_{\lambda}^{n} T_{\lambda}^{n}) \tag{3.12}$$

It is important to note that this advection is carried out for only half a time step. Once this is done the MP4 interpolation kernel given by 2.12 is used to interpolate the values of the velocity to the new position of the particles. The particles are then advected again for the remaining time step with the new velocities from their intermediate position. It is worth noting that the values of the transported quantities in each particle, namely vorticity and temperature, do not change over the course of their movement from the grid to their final position. This is because of the governing equations which dictate that the flow quantities do not change along the particle pathlines.

Remeshing

At the end of the time step, once they have been advected to a distance away from the initial grid position as dictated by the local velocity, the MP4 kernel from 2.12 is used again to interpolate the values for temperature and vorticity back to the grid node locations. Once this is done, we regain the flow field values regularly distributed on the underlying grid that we started with. This marks the end of the advection split for the remeshed vortex method.

3.1.12 Update vorticity and temperature

The values for vorticity and temperature are now known at the grid locations after remeshing. They represent the new values to be used at the start of the next time step. They are updated as required in all the data structures:

$$\omega^{n+1} = \omega_{\lambda}^{n} \tag{3.13}$$

$$T^{n+1} = T_{\lambda}^{n} \tag{3.14}$$

3.1.13 Update solid position

Now the position of the solid is also updated based on the imposed or calculated velocities as follows:

$$\mathbf{x}_{cm}^{n+1} = \mathbf{x}_{cm}^{n} + \mathbf{u}_{T}^{n}\Delta t^{n} \tag{3.15}$$

$$\theta^{n+1} = \theta^{n} + \dot{\theta}^{n}\Delta t^{n} \tag{3.16}$$

This results in the new position of the solid that is used to generate the characteristic function for the subsequent time step.

3.2 Calculation of Nusselt number

The Nusselt number is often the quantity of interest in these heat transfer studies. It is a dimensionless quantity that represents the ratio of convective heat transfer to conductive heat transfer from a solid. It is defined by the following expression:

$$Nu = \frac{h}{k/L} \tag{3.17}$$

where, h is the average heat transfer coefficient, k the thermal conductivity of the fluid and L a characteristic length scale of the problem. In a more general sense it can be shown that the average Nusselt number is the local Nusselt number(Nu_x) at the point of interest integrated over the corresponding range. Typically this range is the surface of the solid over which the heat transfer is taking place. i.e.

$$\overline{Nu} = \frac{1}{H}\int Nu_x(x) \cdot dx \tag{3.18}$$

It can be shown from the governing equations and first principles that the Nusselt number is the line integral of the surface normal gradient over the solid surface when non dimensionalized i.e.

$$Nu = \frac{1}{L((T_s - T_\infty)}\oint \nabla(T - T_s) \cdot \hat{n} d\theta \tag{3.19}$$

The main disadvantage of this method is the fact that surface is not very well defined in the immersed boundary method utilized in this work. A typical alternative often adopted in literature [20] is the use of the divergence theorem to transform this surface integral into a volume integral which results in the following relation:

$$Nu = \frac{1}{L(T_s - T_\infty)}\int\int \nabla^2(T - T_s) dA \tag{3.20}$$

Once again, this is not an ideal approach to compute the Nusselt number as it involves the calculation of the Laplacian which only has a contribution at or near the mollified the surface of the solid. In this sense, it is quite similar to the previous method involving surface normal gradients. Figure 3.1 shows the Laplacian of the temperature for the case of an isothermally heated cylinder by penalization. Note that only the Laplacian of the temperature in the solid region is shown in the image as only these values are utilized to compute the Nusselt number in 3.20. It is clear from the image that the value for the Laplacian is very spiky(large gradients) and often spread across only a few grid nodes. As one might expect, this affects the accuracy of the solution adversely. A potential work around is to use multi resolution techniques to locally refine the surface of the cylinder, in effect adding more grid nodes to capture these steep gradients. However, this is still not ideal due to the discontinuous nature of the characteristic function.

The solution is to simply compute the amount of heat lost to the surrounding fluid in unit time from the penalization term instead. This is accomplished by integrating the penalization term over the entire volume of the solid. The resulting term is exactly equal to the heat lost by the solid per unit time which can then be non dimensionalized to compute the Nusselt number. This results in a relation for the Nusselt number of the following form:

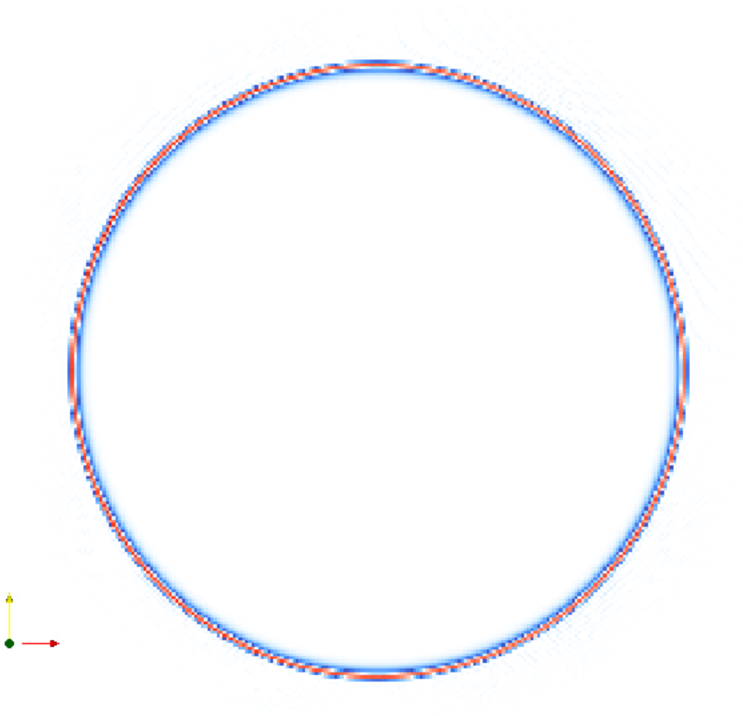


Figure 3.1: Laplacian of temperature in the solid domain

$$Nu = \frac{1}{A(T_s - T_\infty)} \int\int\int \lambda \chi_s (T_s - T) dV \tag{3.21}$$

Note that this formulation does not involve the calculation of any derivatives of the temperature. As a result we no longer need to compute any quantity concentrated at the mollified surface. i.e. There are no steep gradients that need to be resolved accurately. A here corresponds to the surface area over which heat transfer occurs and the volume integral in two dimensions devolves into an integral over the entire area of the solid.

Chapter 4

Validation

Since the focus of this code is studying heat transfer, a simple problem is chosen to validate the code. Namely, the heat transfer from a heated cylinder immersed in a fluid. As mentioned previously, only the isothermal (Dirichlet) boundary where the cylinder is maintained at a constant temperature is considered in this study. There are three primary modes of heat transfer by convection:

- Natural convection or Free convection
- Forced convection
- Mixed convection

Natural convection the situation where there is no imposed motion on the fluid. Instead all the heat transfer is through motion imposed by the heating of the fluid by the solid in addition to of course conduction. Usually, this is through the action of buoyant forces which arise from the action of a gravity field on the fluid. In the case of forced convection, a certain motion is imposed on the fluid through external forces and it carries away the heat from the solid. Mixed convection as the name suggests is a mixture of both mechanisms. The distinction between the three modes is made quantitatively using a set of dimensionless numbers that is explained in the next section.

These three mechanisms are chosen as they encompass the mechanism through which the physics of the heat equation enters the problem. The physics of the Fluid structure interaction problem and that encompassed by the momentum equation has already been validated thoroughly in [7]. As a result, we will not focus too much on this aspect except for verifying that moving boundaries are satisfactorily captured.

4.1 Dimensionless numbers

The advantage of using dimensionless numbers is that it allows us to distinguish the mechanisms at play quantitatively and while also providing a valid measure of comparison different setups. One of the dimensionless measures has already been introduced in the previous section, the Nusselt number, which represents a measure of the "output"(Amount of heat transferred from the solid) of the problem. The next three dimensionless numbers represent the inputs/setup of the problem.

4.1.1 Reynolds number

The Reynolds number is probably the most well known and useful dimensionless measure to emerge from the field of fluid mechanics [26]. It is a measure of the relative importance of the inertial forces in the fluid to the viscous forces. At higher Reynolds number we find the flow to transition from a steady laminar to an unsteady flow all the way up to turbulent flows with increasing Reynolds numbers. Note that we have not incorporated a model for turbulence in this code so we will restrict ourselves to low to moderate Reynolds numbers in this study. Also by definition, at higher Reynolds numbers the advection term of 2.1 becomes more important compared to the viscous term. The relation for the Reynolds number is given by:

$$Re = \frac{Dv}{\nu} \tag{4.1}$$

Where, D is a characteristic length of the problem (The diameter of the cylinder in this case), v is the imposed velocity of the fluid and ν is the kinematic viscosity of the fluid.

4.1.2 Grashof number

The Grashof number is a measure of the relative importance of the buoyant forces to the viscous forces in a fluid. [11] It is analogous to the Reynolds number in situations where natural convection is important as opposed to forced convection with an imposed fluid velocity. It is given by the relation:

$$Gr = \frac{g\beta(T_s - T_\infty)D^3}{\nu^2} \tag{4.2}$$

Rayleigh number

The Rayleigh number is often used interchangeably with the Grashof number, particularly in situations with pure natural convection with no imposed fluid motion. It is simply the product of the Grashof number and the Prandtl number: [11]

$$Ra = GrPr = \frac{g\beta}{\nu\alpha}(T_s - T_\infty)D^3 \tag{4.3}$$

4.1.3 Prandtl number

The Prandlt number is purely a measure of the fluid property and does not depend on the geometry of the problem [11]. It is a measure of the relative importance of the the momentum diffusion to thermal diffusion. It is given by the relation

$$Pr = \frac{\nu}{\alpha} \tag{4.4}$$

For all the test cases presented in this study we will look at air as the operating fluid which has a value of $Pr = 0.71$.

4.1.4 Distinguishing between the regimes

We can distinguish between the three different regimes of convection quantitatively as follows: [11]

$$\frac{Gr}{Re^2} >> 1 \text{ Natural convection}$$

$$\frac{Gr}{Re^2} \approx 1 \text{ Mixed convection}$$

$$\frac{Gr}{Re^2} << 1 \text{ Forced convection}$$

4.2 Basic setup of problem

In this study for validation purposes we will restrict ourselves to looking at a cylinder immersed in a fluid. The problem is completely defined by the boundary conditions and the dimensionless numbers associated with the setup in addition to the geometry. Note that the code is not restricted to the square geometry depicted in the figure. It is simpy for convenience and simplicity that a square domain is chosen for all the test cases presented in this work. A schematic of the setup is given below.

This setup of a cylinder immersed in an unbounded fluid is chosen for two reasons. One, it is a very simple setup that allows for clear uncomplicated analysis while still capturing much of the relevant complexity of the governing physics. Secondly, it is a well studied problem in the literature and there is an abundance of high quality benchmark results both from experiments and numerical studies. Note that the results presented for Nusselt number are all the mean value over the entire surface rather than the local value at a specific position.

In all the test cases, the cylinder diameter is set to D = 0.05 and it is placed at the center of the domain as show in 4.1. The values for the penalization factor λ and λ_T are set at 10^4. The mollification factor ϵ is set based on the grid resolution. The grid of size 1x1 unit (non dimensional) is discretized into a regular

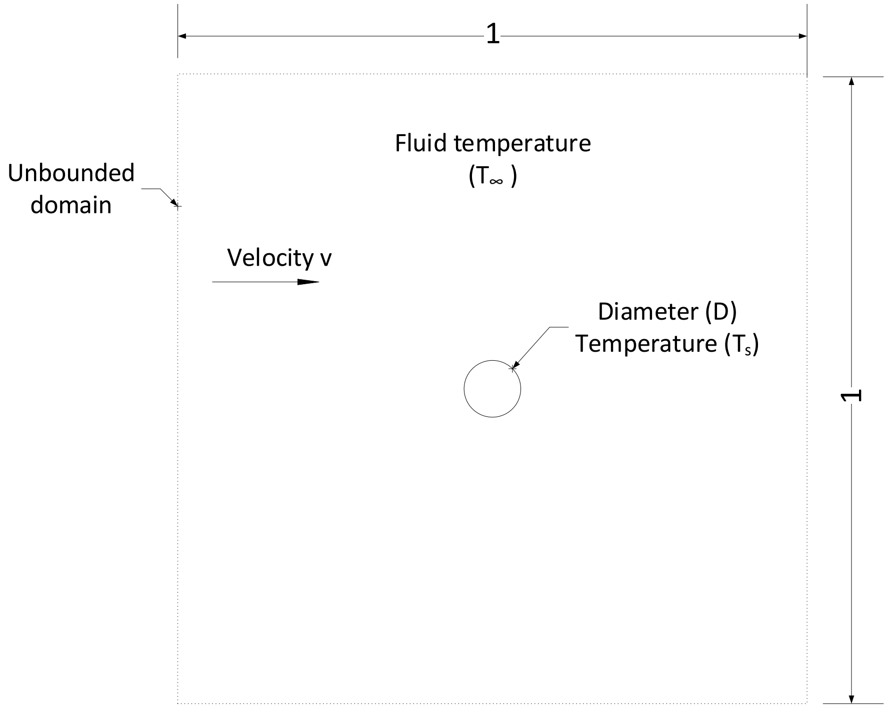


Figure 4.1: Basic setup

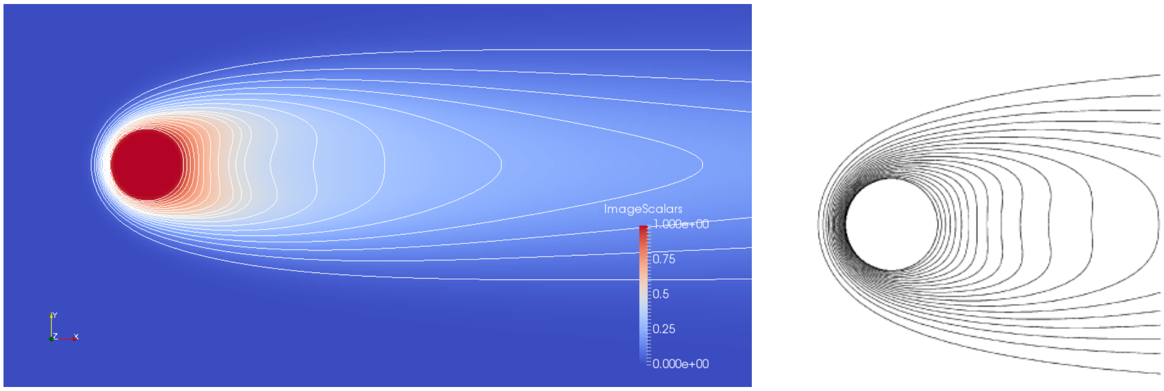
evenly spaced domain of 2048x2048 points. All other flow parameters and properties are chosen in such a way that the dimensionless parameters of the specific test case are satisfied. Also, to mitigate any aliasing effects from the impulsive start the time steps are restricted to a small arbitrary value initially depending on the problem.

4.3 Forced convection

As mentioned previously, the forced convection regime is where the fluid is forced over the heated solid to enable heat transfer. In this regime, the advection portion of the governing physics given by 2.20 plays a major role. Thus the results in this test case are primarily a measure of the ability of the remeshed vortex method to capture the advection term of the energy equation given by 2.20. We look at two test cases in this setup, one with a Reynolds number of 40 and one with a Reynolds number of 100. In both cases the Prandtl number is set to 0.71. The fluid is forced with a velocity of 0.05 units per second from left to right. Since we are interested in purely forced convection here, we neglect the effect of gravity and the buoyancy force term in 2.4 drops out completely. This also means that the Grashof number is zero(value of $\mathbf{g}$ is zero). Thus, this setup represents the idealized case of forced convection in a sense as there is no action of the buoyant forces and it is purely advection that causes the fluid flow and any resulting heat transfer. The results from this case are presented in table 4.1. The results are compared to those from [20]. The values for coefficient of lift for the Re = 40 case are missing as the flow is symmetric about the line through center of the cylinder in the x direction. It is also steady. As a result there are no vertical forces in the upward(y) direction. In the Re = 100 case however, the flow is periodic and the vertical forces oscillate and change directions as the subsequent vortices are shed.

Table 4.1: Comparison of results for forced convection

Reynolds	**Coefficient of drag (C_d)**		**Coefficient of lift (C_l)**		**Nusselt number (Nu)**	
number (Re)	**Literature**	**Current**	**Literature**	**Current**	**Literature**	**Current**
40	1.52	1.593	-	-	3.36	3.3618
100	1.33	1.343	±0.32	±0.346	5.21	5.2093



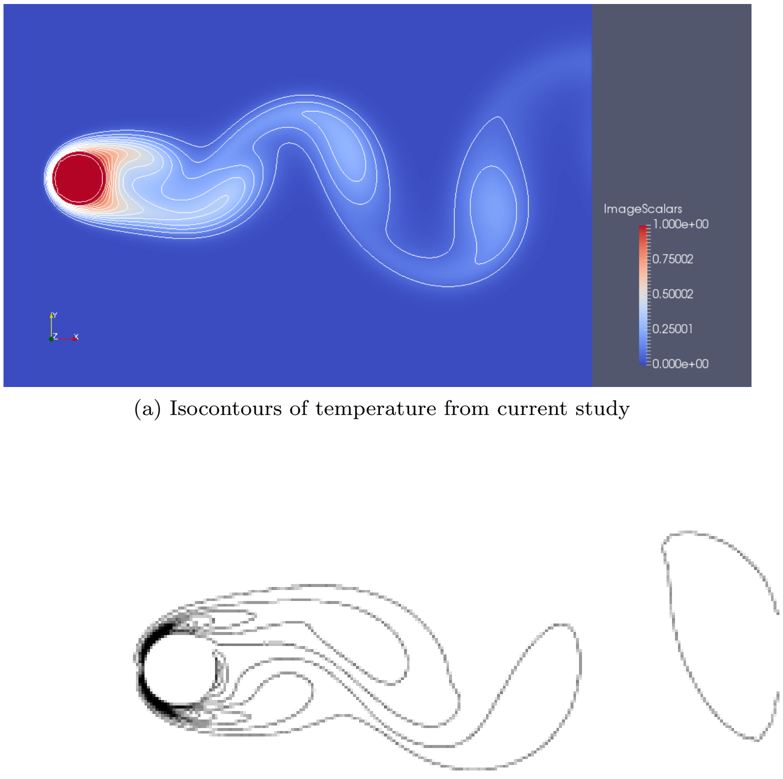
(a) Isocontours of temperature from current study

(b) Isocontours of temperature from literature [20]

Figure 4.2: Comparison of temperature contours for Re = 40

We can see from 4.2 that the code captures the qualitative aspects of the temperature field satisfactorily. We see the lines of constant temperature(temperature contours) in 4.2(a) and the temperature values range from 0 to 1 as seen in the scale. This is to be expected as these temperatures correspond to that of the fluid(coldest) and solid(hottest) respectively. It must be noted that the temperatures are non dimensionalized with respect to the temperature of the hot body and and value of 0 for the temperature has no physical significance here.

In addition to this, in the Re = 100 case we see unsteady but periodic vortex shedding as well which is captured by the vorticity contours. The temperature field also follows the vorticity field in a qualitative sense and as a result the shed vortices are observed in the temperature field as well. A qualitative comparison of the temperature contours is shown in figure 4.3.



(a) Isocontours of temperature from current study

(b) Isocontours of temperature from literature [20]

Figure 4.3: Comparison of temperature contours for Re = 100

4.3.1 Cylinder moving at constant velocity

In addition to the case with the fixed cylinder that we saw in the previous section, we may also perform a change in the frame of reference to study the problem. i.e. Instead of having the fluid flow over the cylinder at a constant speed, we may force the cylinder to move through the fluid at a constant velocity. These two situations are identical in terms of the flow physics and the heat transfer involved. However, such a change in the frame of reference allows us to test the ability of the solver in capturing moving boundaries.

Figure 4.4 shows the temperature field as the cylinder moves through the fluid at a constant velocity. These snapshots are presented at different instants of time. The cylinder is of the same diameter as before i.e.D = 0.05 units. The difference here is that the cylinder moves through the fluid downwards at a velocity of 0.05 units per second instead of the fluid moving. Gravity is once again neglected so buoyant forces do not play any role and the cylinder is neutrally buoyant which means that no baroclinicity is involved. The images in 4.4 correspond to the case where Re = 40. Comparing the images to 4.2 we can see that they are qualitatively identical for the fully developed wake, save for the change in orientation. The value for the Nusselt number is the similar as well (Nu = 3.369). This confirms that the solver deals with moving boundaries accurately at no additional computational cost. Note that we did not have to perform any remeshing of the grid, even for the diffusion operator. Instead we simply keep track of the characteristic function for the solid. This is akin to solving for an additional flow field variable.

4.4 Natural convection

In the natural convection regime, there is no imposed flow on the fluid, rather it remains stationary initially. Once the heat from the solid diffuses into the fluid heating it up, buoyant forces kick in due to a change in the specific weight of the fluid resulting in a net motion of the fluid in a direction opposed to that of the gravity vector. This results of this test case are a good measure of the coupling between 2.20 and 2.19 through the Boussinesq approximation in addition to the ability of the code to capture the advection physics of 2.20 that we tested in the previous section. Gravity is considered to act in the downward direction. Three test cases are considered with Raleigh numbers of 10^3, 10^4 and 10^5. These represent both steady and unsteady cases. The numerical results are presented in table 4.2.

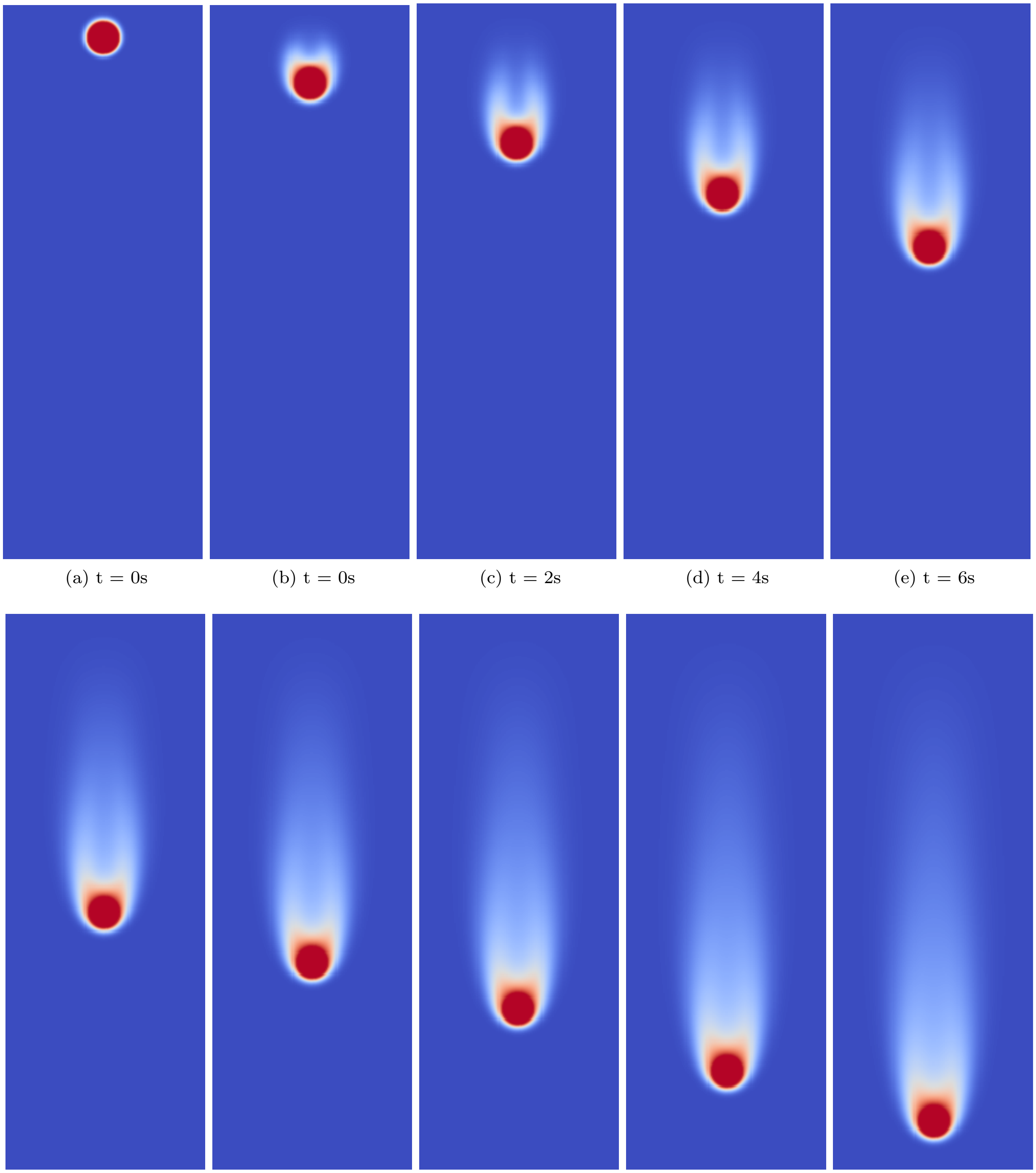
The qualitative comparison of the temperature and vorticity contours from the Ra = 10^4 case are presented in figure 4.5. The formation of buoyant rising plumes of fluid is one of the interesting features of this test case and at higher Rayleigh numbers they turn unsteady and at still higher Rayleigh numbers they turn turbulent.

Table 4.2: Comparison of results for natural convection

Rayleigh number (Ra)	**Nusselt number (Nu)**	
	Literature	**Current**
10^3	3.058	3.06
10^4	4.820	4.86
10^5	8.199	8

4.5 Mixed convection

In the mixed convection regime, we impose a flow on the fluid in addition to the action of gravity(Once again in the downward direction). This causes the fluid to carry away heat from the body as it flows over it initially with the buoyant forces forcing the heated fluid upwards. It is interesting in the mixed flow regime to study two different test scenarios. One in which advection supplements the buoyant forces and aids the heat transfer. This is the cross flow regime where the fluid is forced to move from left to right with a velocity



(a) t = 0s

(b) t = 0s

(c) t = 2s

(d) t = 4s

(e) t = 6s

(f) t = 8s

(g) t = 10s

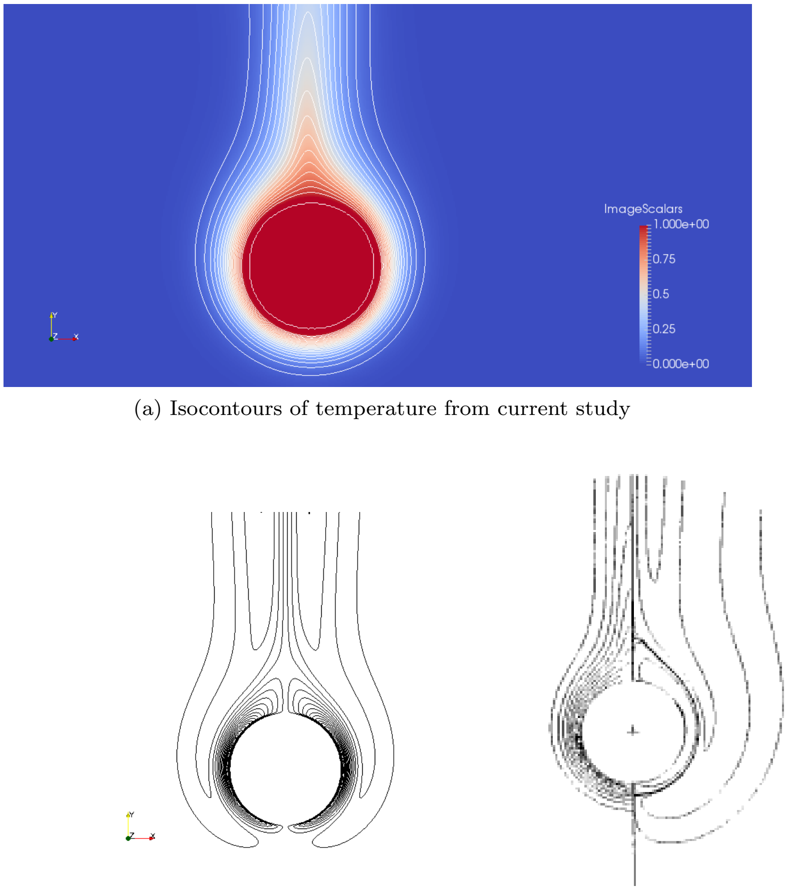
(h) t = 12s

(i) t = 14s

(j) t = 16s

Figure 4.4: Snapshots of temperature field at different times for Re = 40

of 0.05 units per second in a manner similar to the forced convection case. The next regime is the contra flow regime where the fluid advection negates the heat transfer due to buoyant forces. In this case gravity still acts in the downward direction forcing the fluid upward, however the fluid flow direction is now from



(a) Isocontours of temperature from current study

(b) Isocontours of vorticity from current study

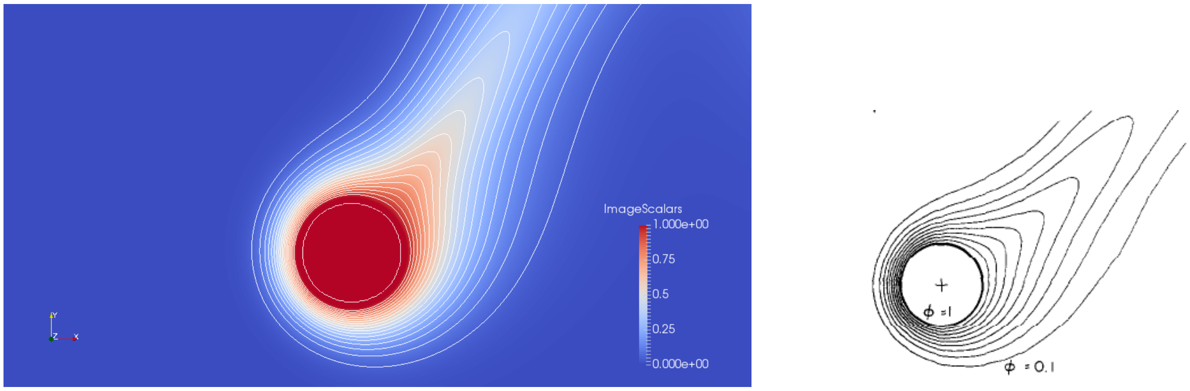
(c) Isocontours of temperature and vorticity from literature [24]

Figure 4.5: Comparison of temperature and vorticity contours for free convection Ra = 10^4

top to bottom with a velocity of 0.05 units per second. Thus we find that the imposed velocity of the fluid is in a direction opposed to that caused by the buoyant forces. This test case is a good measure of the overall performance of the thermal solver as all the modes of heat transfer are active. As mentioned previously the ratio Gr/Re^2 provides a measure of the relative importance of advection and buoyancy. The results are presented in table 4.3 and 4.4 below.

Table 4.3: Comparison of results for cross flow mixed convection

Reynolds number (Re)	**Grashof number** (Gr)	$\frac{Gr}{Re^2}$	**Nusselt number** (Nu)	
			Literature	**Current**
5	30	1.2	1.51	1.527
20	500	1.25	2.65	2.646
20	2000	5	3.1	3.064
40	6400	4	4.129	4.061



(a) Isocontours of temperature from current study

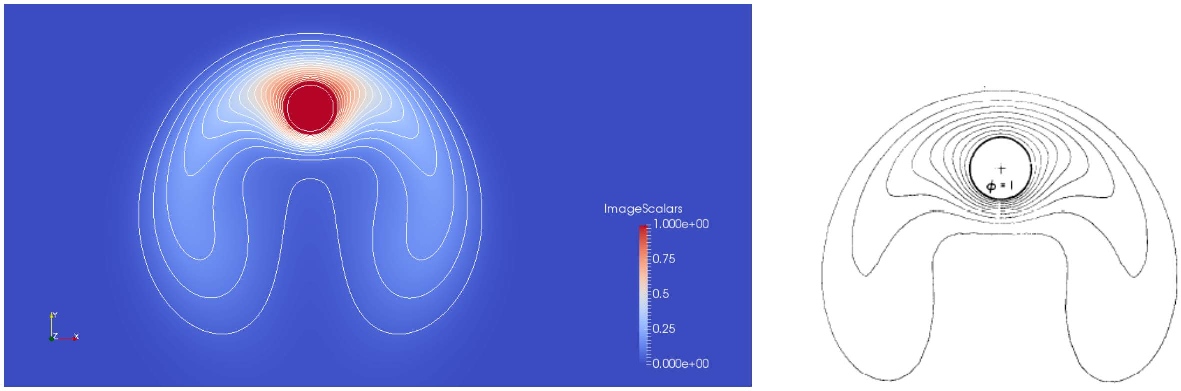
(b) Isocontours of temperature from literature [3]

Figure 4.6: Comparison of temperature contours for cross flow mixed convection Re = 20 Gr = 800

Table 4.4: Comparison of results for contra flow mixed convection

Reynolds number (Re)	**Grashof number** (Gr)	$\frac{Gr}{Re^2}$	**Nusselt number** (Nu)	
			Literature	**Current**
20	100	0.25	2.11	2.237
20	800	2	2.15	2.193
40	800	0.5	3.05	3.067
40	1600	1	2.7	2.770
40	3200	2	3.22	3.276

It is also interesting to note that in the results for Re = 40 in table 4.4 we see a drop in the overall heat transfer followed by an increase despite a continuously increasing value for Grashof number. This is of course because of the effect of the buoyant forces to oppose heat transfer due to the fluid advection. We see the subsequent increase in Nusselt number for the Gr = 3200 case where buoyancy dominates over advection as indicated by the value of Gr/Re^2.



(a) Isocontours of temperature from current study

(b) Isocontours of temperature from literature [4]

Figure 4.7: Comparison of temperature contours for contra flow mixed convection Re = 20 Gr = 800

It can be seen from figure 4.7 that the wake region somewhat elongated laterally due to the two opposing factors causing flow that are at play here. This test case is steady and it is interesting to note that the two limbs of the heated fluid in the wake do not attach.

4.6 Comparison of time steps

Table 4.5 provides a comparison of the restrictions imposed on the solver by the advection operator due to stability. The two columns correspond to the LCFL criteria due to the overlapping condition and the restriction due to the CFL criteria. Of course, the CFL restriction is not applicable in this case but the figures presented serve to highlight the advantage of using a particle method based solver over an Eulerian solver. We see that in almost all the cases, there is a difference of almost an order of magnitude. It must be noted however that in many of these the condition for stability of the diffusion operator given by 3.1 may be more restrictive. In such cases of course the advantage of using a particle solver is lost as far as time step improvements are concerned.

Table 4.5: Comparison of time step limits due to Lagrangian and Eulerian CFL criteria

Test case	**Time step (Δt) limit**	
	LCFL	**CFL**
Forced convection		
Re = 40	9.704e-03	8.878e-04
Re = 100	5.432e-03	7.882e-04
Free convection		
Ra = 10^3	8.137e-02	3.947e-03
Ra = 10^5	3.930e-03	4.495e-04
Mixed convection		
Re = 5 Gr = 30	1.085e-01	4.759e-03
Re = 40 Gr = 6400	6.421e-03	5.105e-04

Chapter 5

Conclusions

A thermal solver capable of handling fluid structure interactions was developed using remeshed vortex methods. The problem of heat transfer by convection from a heated cylinder maintained at constant temperature in an unbounded domain was studied. The results of these simulations were found to compare favorably to previous experimental and numerical studies. The good agreement seen in the results is encouraging to focus further research efforts on this technique. It was also seen that the immersed boundary with the volume of fluid approach to capture the solid domain works well with moving boundaries.

There are plenty of applications of this type of technique primarily relating to heat transfer enhancement [5]. Since the solver is capable of capturing moving boundaries, one problem of interest is that of a phenomena called acoustic streaming or viscous streaming [22]. It is typically seen in viscous fluids involving oscillating solid boundaries that induce various flow patterns. There has also been a lot of interest in studying different methods of leveraging this phenomena to enhance heat transfer and cooling [10, 15].

5.1 Future work

The work presented in this thesis represents a novel method of simulating thermal flows with moving solid boundaries. However, there is scope for improvement:

- The behavior of Neumann and Robin boundary conditions using penalization techniques in similar remeshed vortex methods can be studied. This will enable access to a wider variety of thermal problems.
- Alternatives to penalization may also be investigated to enforce the boundary conditions. One such method is using level set tracking techniques to identify the surface of the solid domain to impose the required values in the flow field.
- The code may be coupled with a Finite element or other suitable solve to study the solid mechanics and account for the material properties. This will enable modeling of piezoelectric bi-morphs at a system level to explore active cooling strategies.

- The solver may be extended to three dimensions to enable the simulation of additional stretching and other ”out of plane” effects crucial to some problems. Multi resolutions techniques maybe be utilized to offset some of the computational costs associated with such a solver.

- A turbulence model may be incorporated into the code to study higher Reynolds number flows. This will allow the exploration of a wider range of problems such as gas turbine blade cooling and combustion chamber design using such techniques.

- There have been reports in the literature that solving for the evolution of the gradient of the transported scalar (temperature) rather than the scalar itself results in greater accuracy for particle methods [2]. Such a scheme may be implemented as required to take advantage of accuracy improvements if any.

Chapter 6

References

[1] Christopher R. Anderson. A vortex method for flows with slight density variations. *Journal of Computational Physics*, 61(3):417, 1985.

[2] C.R. Anderson. Vortex methods for flows of variable density., 1983.

[3] H.M. Badr. A theoretical study of laminar mixed convection from a horizontal cylinder in a cross stream. *International Journal of Heat and Mass Transfer*, 26(5):639–653, 1983. cited By 73.

[4] H.M. Badr. Laminar combined convection from a horizontal cylinder-parallel and contra flow regimes. *International Journal of Heat and Mass Transfer*, 27(1):15–27, 1984.

[5] Rafik Bouakkaz, Fouzi Salhi, Yacine Khelili, Mohamed Quazzazi, and Kamel Talbi. Unconfined laminar nanofluid flow and heat transfer around a rotating circular cylinder in the steady regime. *Archives of Thermodynamics*, 38(2):3 – 20, 2017.

[6] Zhi-Gang Feng and Efstathios E. Michaelides. Heat transfer in particulate flows with direct numerical simulation (dns). *International Journal of Heat and Mass Transfer*, 52(3):777 – 786, 2009.

[7] Mattia Gazzola, Philippe Chatelain, Wim M. van Rees, and Petros Koumoutsakos. Simulations of single and multiple swimmers with non-divergence free deforming geometries. *Journal of Computational Physics*, 230(19):7093 – 7114, 2011.

[8] A.F. Ghoniem, G. Heidarinejad, and A. Krishnan. Numerical simulation of a thermally stratified shear layer using the vortex element method. *Journal of Computational Physics*, 79(1):135–166, 1988.

[9] Mads Mlholm Hejlesen, Petros Koumoutsakos, Anthony Leonard, and Jens Honor Walther. Iterative brinkman penalization for remeshed vortex methods. *Journal of Computational Physics*, 280:547 – 562, 2015.

[10] Sinjae Hyun, Dong-Ryul Lee, and Byoung-Gook Loh. Investigation of convective heat transfer augmentation using acoustic streaming generated by ultrasonic vibrations. *International Journal of Heat and Mass Transfer*, 48(3):703 – 718, 2005.

[11] F.P. Incropera and D.P. DeWitt. *Fundamentals of heat and mass transfer*. Wiley, 1996.

[12] Nicolas James, Emmanuel Maitre, and Iraj Mortazavi. Immersed boundary methods for the numerical simulation of incompressible aerodynamics and fluid-structure interactions. *Annales Mathmatiques Blaise Pascal*, 20(1):139, 2013.

[13] Benjamin Kadoch, Dmitry Kolomenskiy, Philippe Angot, and Kai Schneider. A volume penalization method for incompressible flows and scalar advectiondiffusion with moving obstacles. *Journal of Computational Physics*, 231(12):4365 – 4383, 2012.

[14] B.S. Kim, D.S. Lee, M.Y. Ha, and H.S. Yoon. A numerical study of natural convection in a square enclosure with a circular cylinder at different vertical locations. *International Journal of Heat and Mass Transfer*, 51(7):1888 – 1906, 2008.

[15] M. Kimber, S.V. Garimella, and A. Raman. Local heat transfer coefficients induced by piezoelectrically actuated vibrating cantilevers. *Journal of Heat Transfer*, 129(9):1168–1176, 2007.

[16] A. Krishnan and A.F. Ghoniem. Simulation of rollup and mixing in rayleigh-taylor flow using the transport-element method. *Journal of Computational Physics*, 99(1):1–27, 1992.

[17] Eckart Meiburg. Vortex methods: Theory and practice georges-henri cottet petros koumoutsakos. *SIAM Review*, (3):502, 2002.

[18] Rajat Mittal and Gianluca Iaccarino. Immersed boundary methods. *Annual Review of Fluid Mechanics*, 37(1):239–261, 2005.

[19] J.J. Monaghan. Extrapolating b splines for interpolation. *Journal of Computational Physics*, 60(2):253–262, 1985.

[20] Dartzi Pan. An immersed boundary method on unstructured cartesian meshes for incompressible flows with heat transfer. *Numerical Heat Transfer, Part B: Fundamentals*, 49(3):277–297, 2006.

[21] Charles S Peskin. Flow patterns around heart valves: A numerical method. *Journal of Computational Physics*, 10(2):252 – 271, 1972.

[22] N Riley. Steady streaming. *Annual Review of Fluid Mechanics*, 33(1):43–65, 2001.

[23] Winckelmans G. S. *Vortex Methods*, chapter 5. American Cancer Society, 2004.

[24] K. Saitoh, T. Maruhara and T. Sajiki. Bench mark solutions to natural convection heat transfer problem around a horizontal circular cylinder. *International Journal of Heat and Mass Transfer*, 36(5):1251–1259, 1993.

[25] F. Schlegel, D. Wee, and A.F. Ghoniem. A fast 3d particle method for the simulation of buoyant flow. *Journal of Computational Physics*, 227(21):9063–9090, 2008.

[26] D.J. Tritton. Physical fluid dynamics. 2017.

[27] Zeli Wang, Jianren Fan, Kun Luo, and Kefa Cen. Immersed boundary method for the simulation of flows with heat transfer. *International Journal of Heat and Mass Transfer*, 52(19):4510 – 4518, 2009.

[28] H.S. Yoon, J.B. Lee, and H.H. Chun. A numerical study on the fluid flow and heat transfer around a circular cylinder near a moving wall. *International Journal of Heat and Mass Transfer*, 50(17):3507 – 3520, 2007.