

**Comprehensive Exam Write-up: High and Low
Fidelity Simulations of Compressible Supersonic
Flow around Complex Geometries**

by

Nurlybek Kasimov

B.S., Moscow Institute of Physics and Technology, 2008

M.S., Moscow Institute of Physics and Technology, 2010

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Mechanical Engineering

2014

This thesis entitled:
Comprehensive Exam Write-up: High and Low Fidelity Simulations of Compressible
Supersonic Flow around Complex Geometries
written by Nurlybek Kasimov
has been approved for the Department of Mechanical Engineering

Prof. Oleg V. Vasilyev

Prof. Peter Hamlington

Prof. Daven Henze

Prof. Alireza Doostan

Prof. Sedat Biringer

Prof. John A. Evans

Date _____

The final copy of this thesis has been examined by the signatories, and we find that
both the content and the form meet acceptable presentation standards of scholarly
work in the above mentioned discipline.

Kasimov, Nurlybek (Ph.D., Mechanical Engineering)

Comprehensive Exam Write-up: High and Low Fidelity Simulations of Compressible Supersonic Flow around Complex Geometries

Thesis directed by Prof. Oleg V. Vasilyev

This work consist of two parts — high and low fidelity simulations of supersonic flow around solid obstacles of arbitrary shapes, accordingly. Low fidelity simulations are performed by solving 3D Euler equations for supersonic flows in presence of obstacles using Roe solver as a shock capturing scheme and modification of Brinkman penalization method to handle obstacles through forcing terms. Particle-particle interactions are handled using 3D DEM solver and then coupled with Euler/Roe CFD solver. High fidelity simulations, on the other hand, are performed using Adaptive Wavelet Collocation Method for grid adaptation and adaptive differentiation, Artificial Viscosity Method as a shock capturing scheme and Characteristic Based Volume Penalization Method to handle solid obstacles through forcing terms.

Dedication

To most important person in my life — my son.

Acknowledgements

Oleg Vasilyev, Eric Brown-Dymkoski, Jonathan Regele, Alexei Vezolainen, Scott Reckinger, Alireza Nejadmalayeri, thank you.

Contents

Chapter

1	Introduction	1
1.1	Motivation and Objective	1
1.2	Methodology	2
1.2.1	Volume penalization	3
1.2.2	Adaptive Wavelet Collocation Method	5
1.2.3	Shock Capturing Scheme	5
1.2.4	Shock Capturing Scheme, Low Fidelity Simulations	6
1.2.5	CFD/DEM Coupling, Low Fidelity Simulations	6
1.3	Organization	6
2	Background	7
2.1	Brinkman Penalization	7
2.2	Adaptive Wavelet Collocation Method	9
2.3	Wavelet Based Artificial Viscosity as a Shock Capturing Scheme	12
2.4	Riemann Problem and Riemann Solver of Roe	13
3	New Volume Penalization Methods	17
3.1	Brinkman-like Penalization for Low Fidelity Simulations	17
3.2	Characteristic Based Volume Penalization	18

4	High Fidelity Simulation Results	21
4.1	Benchmark of using Characteristic Based Volume Penalization	21
4.1.1	Dirichlet Boundary Condition	21
4.1.2	Neumann Boundary Condition	22
4.1.3	Robin Boundary Condition	23
4.2	Simulation Results	24
4.2.1	Flow around the Wedge	24
4.2.2	Flow around Single and Multiple Stationary Cylinders	26
4.2.3	Flow around Single Stationary and Moving Cylinder	28
5	Low Fidelity Simulation Results	30
5.1	Flow around Single Stationary Sphere	31
5.2	Flow around Moving Sphere	32
5.3	Flow around Several Free Ellipsoids	33
5.4	Flow around “Disk” and “Needle” like obstacles	34
5.5	Flow around Stationary Ellipsoid, Negative Pressure Fix	35
6	Future Work	37
6.1	High Fidelity Simulations	37
6.2	Low Fidelity Simulations	38
Bibliography		39

Figures

Figure

1.1	IED explosion in Iraq, hitting U.S. military convoy	2
1.2	Body fitted grid examples	3
2.1	Mesh points for IB methods	7
2.2	Wavelet coefficients of a function with two sharp transitions	11
2.3	Localized nature of wavelet function	11
2.4	Safety region with possible significant wavelets after one time step	12
2.5	Localized nature of wavelet function	13
3.1	Riemann problem set up	18
3.2	Normal reflection from the wall (top) and BP (bottom)	18
3.3	SOD1 test problem with CBVP	20
4.1	Temperature profile at the end of simulation. Dirichlet boundary condition	22
4.2	Temperature profile at the end of simulation. Neumann boundary condition	23
4.3	Temperature profile at the end of simulation. Robin boundary condition	24
4.4	Schematic of the flow around the wedge	25
4.5	$\theta - \beta - M$ relation plot	25
4.6	Flow around the wedge results	26

4.7	Single and multiple cylinders case schematics	27
4.8	Supersonic flow around single cylinder	27
4.9	Supersonic flow around multiple cylinders	28
4.10	Supersonic flow around stationary and moving cylinders	28
5.1	CFD/DEM coupling schematics	30
5.2	Low Fidelity simulations set up	31
5.3	Flow around single stationary sphere	32
5.4	Flow around single free sphere	32
5.5	Grid cells occupied partially by the particle calculated more accurately	33
5.6	Flow around two ellipsoidal particles	34
5.7	Flow around 1k ellipsoidal particles	34
5.8	Disk shape (left) and needle shape (right) particles	35
5.9	Dynamics of the flow around disk (left) and needle (right)	35
5.10	Flow around single ellipsoid with porosity fix and energy fix enabled . .	36

Chapter 1

Introduction

1.1 Motivation and Objective

This project is supported by ONR MURI on Soil Blast Modeling. It is a part of a bigger multiscale/multiphysics project on research and simulations of explosions of buried explosive devices [1]. In Figure 1.1 one can see ejecta under asphalt pavement after improvised explosive device (IED) explosion hitting U.S. military convoy, which was the starting point of the project. Main motivation of the project is to overcome wide variety of uncertainties that occur at the moment of explosion (blast wave propagation) due to the soil type, moisture, density; landmark peculiarities, explosive type, and etc., so one can better predict harmful effect of the explosion and perform a proper defensive strategies such as armoring.

Existing methodologies allow to do simulations on each phenomenon consisting this complex phenomenon separately but not for all of them simultaneously using coupled equations. The main focus of this project is to provide a low fidelity computation model of fluid-structure interaction of compressible supersonic flow around multiple solid particles. The model is based on high fidelity DNS simulations involving several orders of magnitude less number of particles and properly extrapolated and/or averaged in order to couple with other approaches. For that purpose new volume penalization method is developed for high fidelity simulations [2]. For low fidelity simulations volume penalization method roughly based on Brinkman method is developed, which mimicked



Figure 1.1: IED explosion in Iraq, hitting U.S. military convoy

boundary conditions on fluid-particle interface but which also took into the account extremely rough grid, i.e. boundary layer is not resolved.

1.2 Methodology

In this project several numerical techniques are used to address different parts of the framework. High fidelity simulations utilized Characteristic Based Volume Penalization (CBVP) method developed by O. V. Vasilyev, E. Brown-Dymkoski and N. Kasimov. To address discontinuities in flow parameters (shock waves) specific artificial viscosity method based on wavelets is used [3]. Local grid refinement (data compression) and corresponding differentiation on adaptive grid is achieved using Adaptive Wavelet Collocation Method (AWCM) [4]. Low fidelity simulations are performed using Briknman-like momentum penalization of Euler equations (inviscid flow) in order to match drag force. Riemann solver of Roe is used as a shock capturing scheme [5]. Particle-particle interaction is calculated using DEM/FEM method and later coupled with CFD part of the solver.

1.2.1 Volume penalization

In order to handle flows around solid obstacles numerically, several approaches are used. One can classify them into two main categories — Body Fitted Grid (BFG) methods and Immersed Boundary methods (IB) [6]. BFG methods allow to construct typically curvilinear grid in such a way that obstacle boundary is aligned with grid lines (See Figure. 1.2a), so no special treatment of the boundary conditions at the obstacle is needed.

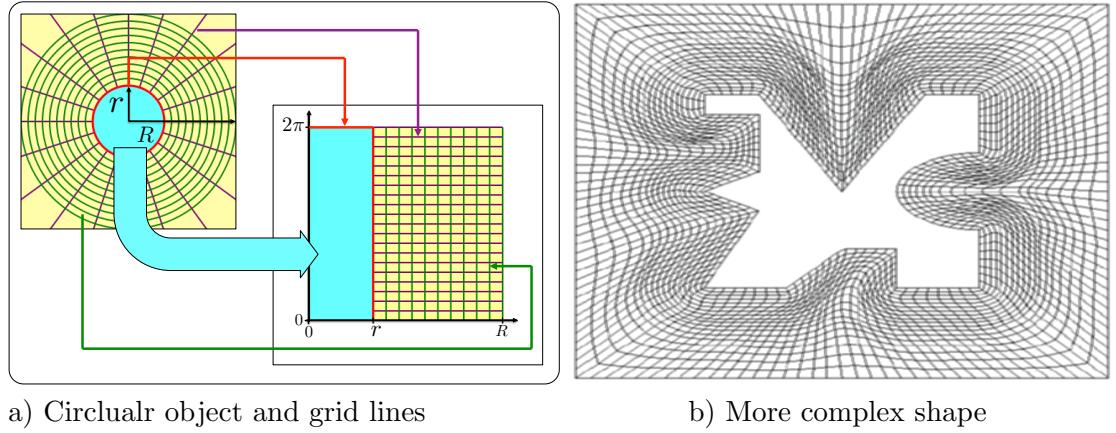


Figure 1.2: Body fitted grid examples

In order to get rectilinear system as in Figure 1.2(a) one need to perform radial transformation

$$x = x_0 + r \cos \varphi$$

$$y = y_0 + r \sin \varphi$$

and change all equations accordingly. In most cases obstacles have more complex shapes (see Figure 1.2b) and finding corresponding transformation of the coordinates might be problematic. Additional difficulties arise when obstacle is allowed to move and deform due to the interaction with the flow — new transformation should be found at each time step.

IB methods, Volume Penalization (VP) methods in particular, in contrary, do

not change existing grid, but rather change governing equations, i.e. they are intrusive methods. Equations are modified through artificial forcing terms that are added to the governing equations so that solution at the boundary tends to the exact solution. The methodology underlying this project is a VPM.

1.2.1.1 Brinkman Penalization Method

The first VPM that is used at early stage of the project is Brinkman Penalization method (BPM), originally proposed by Arquis and Caltagirone [7] for incompressible flow around solid obstacles through momentum penalization. This method is successfully extended to the subsonic compressible flow [8]. The main advantages of the method are systematic error control and relative simplicity of numerical implementation. The very basic interpretation of the method can be achieved if one represents solid obstacle as a porous media with controlled permeability and porosity.

1.2.1.2 Characteristic Based Volume Penalization

BPM applied “as it is” to supersonic flows led to numerous errors, such as inaccurate acoustic reflection, pressure seepage, phase lag, etc. Also the fact that BPM cannot be used to model boundary conditions other than Dirichlet and homogeneous Neumann made it difficult for some type of simulations with Neumann and Robin type boundary conditions. In order to handle these deficiencies new VPM is developed — Characteristic Based Volume Penalization. To mimic Dirichlet type boundary conditions CBVP still utilizes BP with some modifications to enhance numerical stability. For Neumann and Robin type boundary conditions the solution is propagated inside of the obstacle along inward directed normal vector, hence the name of the method.

1.2.1.3 Brikman-like Volume Penalization, Low Fidelity Simulations

For Low Fidelity simulations supersonic flow is modeled using extremely coarse grid. The latter restricted us to resolve boundary layer, so Euler equation are used instead. To mimic drag force caused by the flow as in case of the viscous flow, we introduced penalization of the momentum equation. Also, due to the coarse grid, in case of many obstacles amount of the flow penetrating the obstacle could lead to significant numerical errors. To fix it, another penalization terms related to the porosity is introduced, which is also inspired by BPM.

1.2.2 Adaptive Wavelet Collocation Method

In order to achieve data compression, i.e. to use fewer grid points than if one utilizes uniform rectangular grid, AWCM is used. The main advantage of the method is systematic a priori error control while maintaining structured rectilinear grid. Second generation interpolation wavelets are used as basis functions. Grid adaptation algorithm underlying the method allows to both accurately track regions of “important physics” and easily program it using existing numerical platforms. Tracking can be performed for dynamically changing phenomena without significant overhead — in many cases extra calculations are overcompensated by high data compression.

1.2.3 Shock Capturing Scheme

To handle discontinuities in High Fidelity simulations, shock capturing scheme based on AWCM is used. It is based on the solution of hyperbolic equation near the discontinuities/shocks in order to smooth them with desired width. This procedure introduces non-linear artificial viscosity. The main reason to incorporate this method with AWCM is that latter serves as a good indicator of shock location through wavelet coefficients. Method exhibits good localization property, avoiding unnecessary smoothing in the regions where it is not required.

1.2.4 Shock Capturing Scheme, Low Fidelity Simulations

To handle discontinuities in Low Fidelity simulations Riemann solver of Roe is used. This method is equivalent to using artificial viscosity, although it cannot be controlled and viscosity value is defined by the flow. Main procedure underlying the method is performing certain type of averaging of flow parameters, so called Roe average. After the averaging, Euler equations become pseudo linear hyperbolic equations. Roe solver handles transonic and subsonic regions of the solution worse than discontinuities, although there are ways to overcome this issue, such as so called Harten-Hyman Entropy Fix.

1.2.5 CFD/DEM Coupling, Low Fidelity Simulations

In Low Fidelity simulations, due to the coarse CFD grid, it is possible to model supersonic flow around large number of particles that might interact with both flow and other particles. CFD-DEM coupling provides particle-particle and fluid-particle interactions, while CBVP provides particle-fluid interactions, closing the system. The only restriction applied to the system is to have same global time step for both CFD and DEM solvers, which in some cases might be inefficient.

1.3 Organization

The rest of this paper is organized as follows. In Chapter 2, the necessary background of Volume Penalization techniques, shock capturing schemes, Adaptive Wavelet transforms and basics of CFD-DEM coupling will be covered. Testing different numerical techniques with proper algorithm justification of chosen or developed algorithms is discussed in Chapter 3. In Chapters 4 and 5 one can find results of High Fidelity and Low Fidelity simulations accordingly. Future work is presented in Chapter 6.

Chapter 2

Background

2.1 Brinkman Penalization

There are three main algorithmic steps involved in IB methods:

- (1) Splitting mesh points into three classes (See Figure 2.1): Field points — points outside of the obstacle; Band points — field points closest to the obstacle surface; and Interior points — points inside of the obstacle.

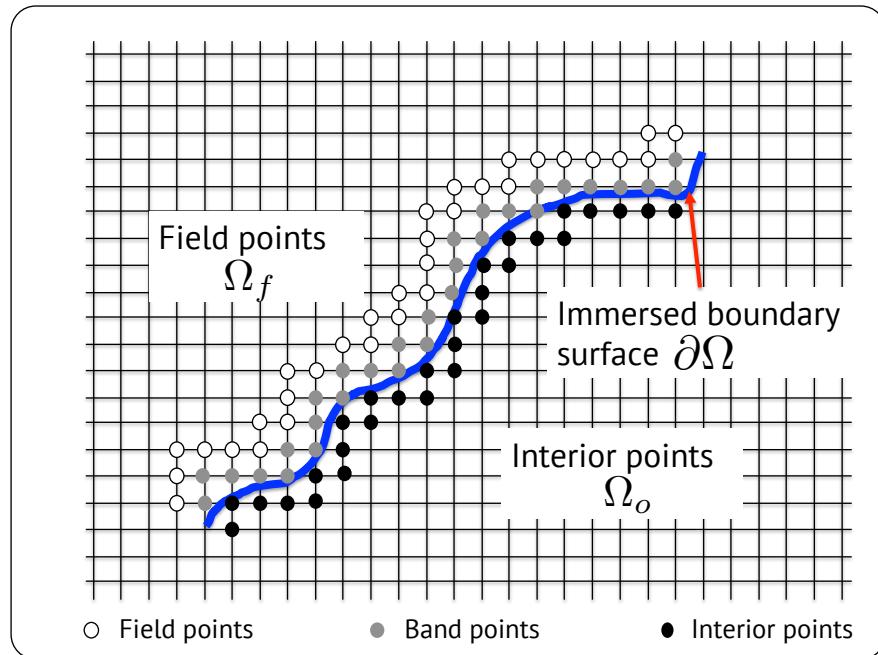


Figure 2.1: Mesh points for IB methods

- (2) Introducing so-called Mask function, typically denoted as $\chi(\mathbf{x})$, which is nothing but indicator function

$$\chi(\mathbf{x}) = \begin{cases} 1, & \text{if } \mathbf{x} \in \Omega_o \\ 0, & \text{else.} \end{cases}$$

- (3) Altering governing equations by adding forcing terms, so that boundary conditions at the obstacle surface are approximately satisfied, involving the $\chi(\mathbf{x})$ above, so that equations in field points do not change.

Additional forcing terms can be added either before the governing equations are discretized or after. This order determines the type of methods in IB methods family. Volume penalization (VP) is a subclass of IB methods where equations are altered before the discretization, therefore conceptually does not depend on numerical approach to be used.

Consider the following Navier-Stokes equations

$$\begin{cases} \frac{\partial \rho}{\partial t} + \frac{\partial \rho u_j}{\partial x_j} = 0 \\ \frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} + \frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} = 0 \\ \frac{\partial \rho e}{\partial t} + \frac{\partial u_j (\rho e + p)}{\partial x_j} + \frac{\partial u_i \tau_{ij}}{\partial x_j} = 0, \end{cases} \quad (2.1)$$

along with constitutive equations for ideal gas that form a closed system.

Throughout entire writeup numerically more convenient form of the equations as described below will be used:

$$\frac{\partial \varphi}{\partial t} = RHS, \quad (2.2)$$

where RHS is different for each equation.

The idea behind the Brinkman penalization is to alter momentum equation, i.e. penalize it, at every grid point.

$$\frac{\partial \rho u_i}{\partial t} = RHS - \frac{\chi}{\eta} \rho (u_i - U_i^o), \quad (2.3)$$

where η is numerical permeability of the obstacle. One needs to keep in mind that it is artificial. This equation is unchanged outside of the obstacle due to the masking function and it mimics no slip boundary condition $u_i|_{\partial\Omega} = U_i^o$ at the timescale η .

In order to reach proper acoustic reflection from the obstacle one can also introduce numerical porosity ϕ in continuity equation to mimic porous media in full extent.

$$\frac{\partial \rho}{\partial t} = -\frac{1}{\phi} \frac{\partial \rho u_j}{\partial x_j},$$

which can be translated to the form (2.2) as

$$\frac{\partial \rho}{\partial t} = RHS - \chi \left(\frac{1}{\phi} - 1 \right) \frac{\partial \rho u_j}{\partial x_j}. \quad (2.4)$$

One can also penalize temperature as well, in case such boundary condition is meant to be satisfied at the obstacle boundary.

$$\frac{\partial T}{\partial t} = RHS - \frac{\chi}{\eta} (T - T^o), \quad (2.5)$$

which mimics isothermal boundary condition $T|_{\partial\Omega} = T^o$ and can be incorporated to the energy equation through the full energy relation for ideal gas,

$$\rho e = \frac{\rho u_j u_j}{2} + \rho c_V T = \frac{\rho u_j u_j}{2} + \frac{p}{\gamma - 1}. \quad (2.6)$$

Results of BPM implementation to supersonic inviscid flow are shown in Chapter 3 as a motivation for new VPM.

2.2 Adaptive Wavelet Collocation Method

For High Fidelity simulations to reduce number of used grid points, AWCM is used. Let us consider some target function $u(\mathbf{x})$ that can be sampled on some uniform grid of chosen highest level of resolution, although grid does not have to be uniform, as long as it is structured. In this project uniform grid is used as a first step in grid adaptation. Following Vasilyev et al. [4] one can decompose given function into wavelet

functions and scaling function at the lowest level of resolution

$$u(\mathbf{x}) = \sum_{k \in \mathcal{K}^0} c_k^0 \phi_k^0(\mathbf{x}) + \sum_{j=0}^{\infty} \sum_{\mu=1}^{2^n-1} \sum_{l \in \mathcal{L}^{\mu,j}} d_l^{\mu,j} \psi_l^{\mu,j}(\mathbf{x}), \quad (2.7)$$

where $\phi_k^0(\mathbf{x})$ is a scaling function at the coarsest level of resolution and $\psi_l^{\mu,j}(\mathbf{x})$ is a wavelet function at the level of resolution j , c_k^0 and $d_l^{\mu,j}$ are scaling and wavelet coefficients accordingly. Main advantage of the decomposition above is it can be split into two parts

$$u(\mathbf{x}) = u_{\geq}(\mathbf{x}) + u_{<}(\mathbf{x}), \quad (2.8)$$

where

$$\begin{aligned} u_{\geq}(\mathbf{x}) &= \sum_{k \in \mathcal{K}^0} c_k^0 \phi_k^0(\mathbf{x}) + \sum_{j=0}^{\infty} \sum_{\mu=1}^{2^n-1} \sum_{\substack{l \in \mathcal{L}^{\mu,j} \\ |d_l^{\mu,j}| \geq \epsilon \|u\|}} d_l^{\mu,j} \psi_l^{\mu,j}(\mathbf{x}), \\ u_{<}(\mathbf{x}) &= \sum_{j=0}^{\infty} \sum_{\mu=1}^{2^n-1} \sum_{\substack{l \in \mathcal{L}^{\mu,j} \\ |d_l^{\mu,j}| < \epsilon \|u\|}} d_l^{\mu,j} \psi_l^{\mu,j}(\mathbf{x}), \end{aligned} \quad (2.9)$$

based on some threshold value ϵ . Donoho [9] had shown that absolute error of neglecting $u_{<}(\mathbf{x})$ part of the function leads to controllable error on the function, namely

$$\|u - u_{\geq}\| \leq C\epsilon \|u\|, \quad (2.10)$$

which means that one can use only part of the grid points and retain desired accuracy. This property allows to construct an adaptive grid with significantly fewer number of points. Only the regions with steep gradients have large values of wavelet coefficients, which makes wavelets a good indicator of those gradients. For example, in Fig. 2.2 wavelet coefficients of the function with two sharp transitions are presented.

Wavelet basis functions possess other “nice” properties, such as localization in both space and time. In this project, along with all other projects conducted in Multiscale Modeling and Simulation Laboratory, interpolating wavelets are used (See

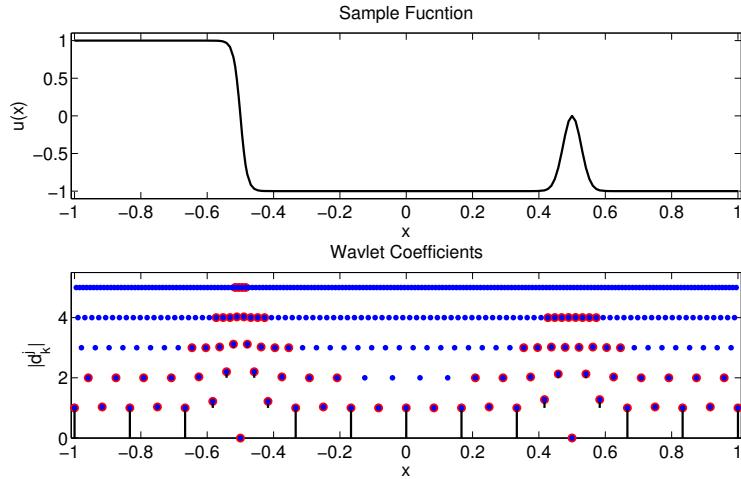


Figure 2.2: Wavelet coefficients of a function with two sharp transitions

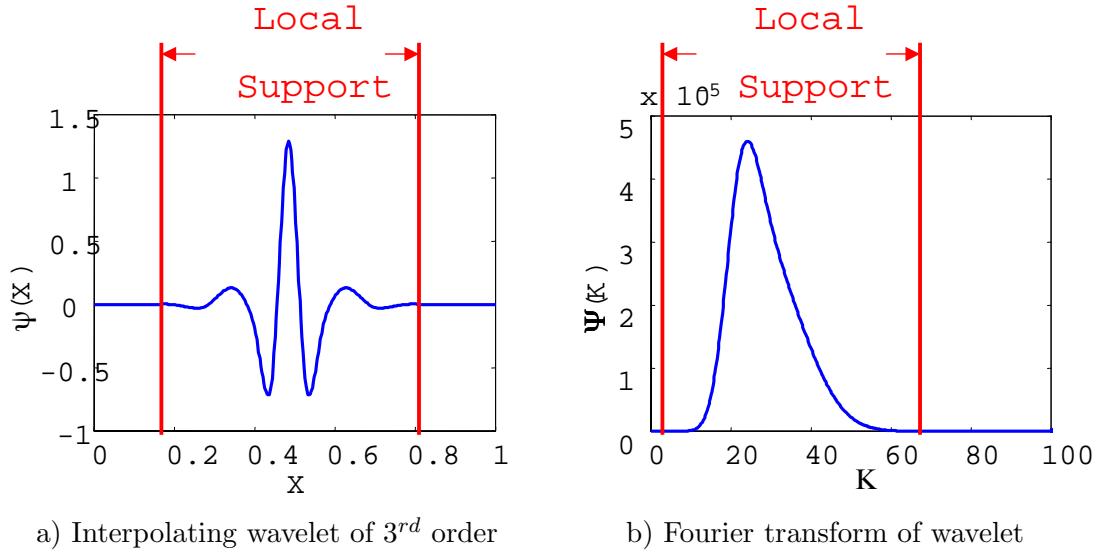


Figure 2.3: Localized nature of wavelet function

Fig. 2.3). Constructing wavelet functions is relatively simple and have well described recursive algorithm [10].

Adaptation procedure can be easily iterated over time evolution, which makes it also a dynamic, so one can assure that grid always represents current state of the physical phenomenon. In order to do this properly the concept of Adjacent Zone is introduced [11] (see Fig. 2.4).

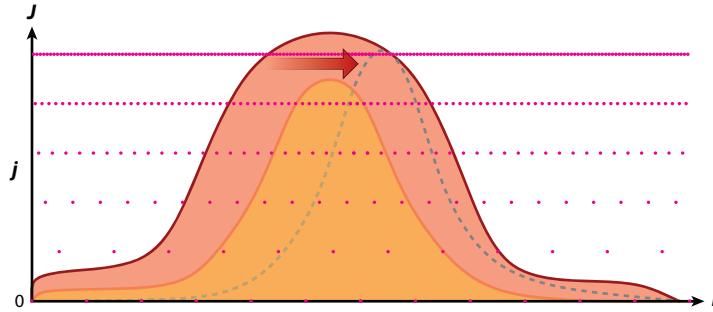


Figure 2.4: Safety region with possible significant wavelets after one time step [4]

One may consider it as a safety regions within which coefficients might become significant even after the time integration. In this case one can make sure that global accuracy of the simulations is still within the user defined and controlled threshold in evolution problems, and most importantly keep adapted grid in up to date state. Using adapted grid also saves memory required to store variable values at each grid point, since one needs fewer of them.

2.3 Wavelet Based Artificial Viscosity as a Shock Capturing Scheme

Using AWCM becomes even more attractive for High Fidelity simulations per discussion in previous section — Wavelet coefficients have large values at the regions of steep gradients and discontinuities are as steep as one can get. Numerically it means that there are coefficients on all levels of resolution. By Regele and Vasilyev [3] is developed a method to track and smooth those gradients. The core of the method is to add artificial viscosity to the desired hyperbolic equation with dynamic artificial viscosity, when latter depends on “shock locator” — specifically cooked flux limiting function. Let us consider simple one dimensional conservation equation with added artificial diffusion term,

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} = \frac{\partial}{\partial x} \left(\nu(\Phi) \frac{\partial u}{\partial x} \right), \quad (2.11)$$

where Φ can be found as

$$\Phi = \begin{cases} \frac{|d_k^{j_{max}}|}{\|u\|}, & \text{if } 0 < |d_k^{j_{max}}| \leq \epsilon_d \|u\| \\ 1, & \text{if } |d_k^{j_{max}}| > \epsilon_d \|u\|, \end{cases} \quad (2.12)$$

where $|d_k^{j_{max}}|$ is a wavelet coefficient at finest level of resolution and ϵ_d is also a chosen thresholding parameter to limit Φ , apart from the wavelet threshold parameter ϵ .

In Fig. 2.5 one may find viscosity field and corresponding Schlieren image of the density field for 2D supersonic flow around the solid circle. Problem is solved using aforementioned artificial viscosity. Per methods design, viscosity is non-zero at shock locations, based on the Schlieren image, and vanishes everywhere else.

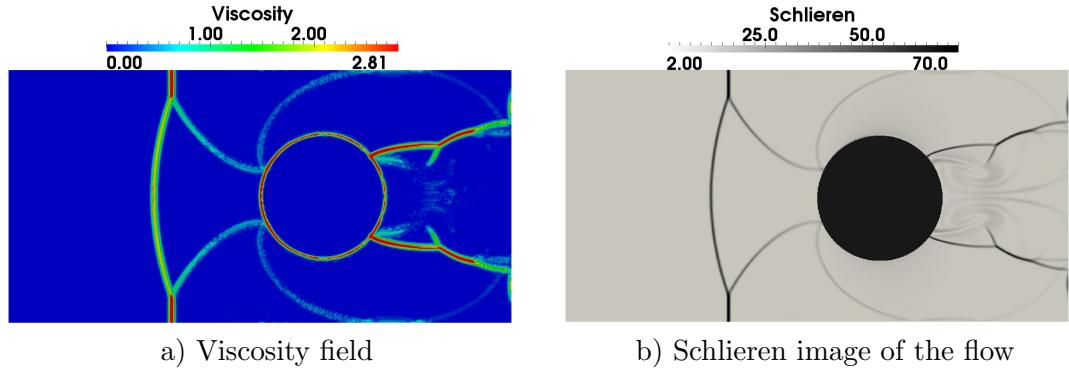


Figure 2.5: Localized nature of wavelet function

2.4 Riemann Problem and Riemann Solver of Roe

A Riemann problem consists of a conservation laws together with piecewise constant data having a single discontinuity, or in other words, one or multiple hyperbolic equations with discontinuous initial conditions. As such system one can consider Euler or Navier-Stokes equations. Either numerical or analytical technique to solve such a problem is called Riemann solver. For non-linear problems, like Euler and Navier-Stokes, there are number of numerical Riemann solvers, such as Godunov Scheme, Flux Vector Splitting, Roe solver [5], etc. For Low Fidelity simulations of this project latter is used.

For the sake of simplicity, let us consider pseudo one dimensional Euler equations for ideal gas with discontinuous ICs.

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{u})}{\partial x} = 0$$

$$\mathbf{u} = \begin{cases} \mathbf{u}_L, & \text{if } x \leq x_S \\ \mathbf{u}_R, & \text{if } x > x_S, \end{cases} \quad (2.13)$$

where

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho e \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ u(\rho e + p) \end{bmatrix}, \quad (2.14)$$

and constitutive equation (2.6) to close the system. It can be rewritten in pseudo linear form

$$\frac{\partial \mathbf{u}}{\partial t} + \mathcal{A}(\mathbf{u}) \frac{\partial \mathbf{u}}{\partial x} = 0, \quad (2.15)$$

where \mathcal{A} is a Jacobian matrix of the flux \mathbf{F} with respect to the solution vector \mathbf{u} , which has the following explicit relation

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ \hat{\gamma}H - u^2 - a^2 & (3 - \gamma)u & -\hat{\gamma}v & -\hat{\gamma}w & \hat{\gamma} \\ -uv & v & u & 0 & 0 \\ -uw & w & 0 & u & 0 \\ u[(\gamma - 2)H - a^2] & H - \hat{\gamma}u^2 & -\hat{\gamma}uv & -\hat{\gamma}uw & \gamma u \end{bmatrix}, \quad (2.16)$$

where $\hat{\gamma} = \gamma - 1$, $H = (\rho e + p)/\rho$ (Specific enthalpy) and $a = \sqrt{\gamma p/\rho}$ (Local speed of sound). This matrix has eigenvalues

$$\lambda_1 = u - a, \quad \lambda_2 = \lambda_3 = \lambda_4 = u, \quad \lambda_5 = u + a, \quad (2.17)$$

and eigenvectors

$$\left. \begin{aligned} \mathbf{K}^{(1)} &= \begin{bmatrix} 1 \\ u-a \\ v \\ w \\ H-ua \end{bmatrix}, \mathbf{K}^{(2)} = \begin{bmatrix} 1 \\ u \\ v \\ w \\ \frac{1}{2}V^2 \end{bmatrix}, \mathbf{K}^{(3)} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ v \end{bmatrix}, \\ \mathbf{K}^{(4)} &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \\ w \end{bmatrix}, \mathbf{K}^{(5)} = \begin{bmatrix} 1 \\ u+a \\ v \\ w \\ H+ua \end{bmatrix} \end{aligned} \right\}, \quad (2.18)$$

where $V = (u^2 + v^2 + w^2)$.

Equation (2.15) can be numerically integrated as

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \frac{\Delta t}{\Delta x} \left(\mathbf{F}_{i-\frac{1}{2}} - \mathbf{F}_{i+\frac{1}{2}} \right), \quad (2.19)$$

where half-integer indices are introduced for convenience and to distinguish intercell values from cell values on chosen grid. Variety of Riemann solvers differ by the way they calculate those intercell values.

Philip Roe proposed to calculate intercell fluxes using certain averaging procedure based on right and left values of corresponding local fluxes [12] as follows

$$\widetilde{(\bullet)} = \frac{\sqrt{\rho_L} (\bullet)_L + \sqrt{\rho_R} (\bullet)_R}{\sqrt{\rho_L} + \sqrt{\rho_R}}, \quad (2.20)$$

where (\bullet) stands for u, v, w and H . After finding these averaged values, one also needs to find \widetilde{V} and \widetilde{a} as

$$\begin{aligned} \widetilde{V} &= \widetilde{u}^2 + \widetilde{v}^2 + \widetilde{w}^2, \\ \widetilde{a} &= \left[\hat{\gamma} \left(\widetilde{H} - \frac{1}{2} \widetilde{V}^2 \right) \right]^{\frac{1}{2}}. \end{aligned}$$

Next step is to substitute all values defined in (2.16)-(2.18) with averaged values. Finally, according to Roe, intercell flux can be found as

$$\mathbf{F}_{i+\frac{1}{2}} = \frac{1}{2} (\mathbf{F}_L + \mathbf{F}_R) - \frac{1}{2} \sum_{j=1}^5 \tilde{\alpha}_j \left| \tilde{\lambda}_j \right| \widetilde{\mathbf{K}}^j, \quad (2.21)$$

where α_j is so called wave strength — weight that takes into account contribution from each eigenvector. Roe's original approach is to calculate them as follows:

$$\begin{aligned} \tilde{\alpha}_3 &= \Delta u_3 - \tilde{v} \Delta u_1, \\ \tilde{\alpha}_4 &= \Delta u_4 - \tilde{w} \Delta u_1, \\ \tilde{\alpha}_2 &= \frac{\hat{\gamma}}{\tilde{a}^2} \left[\Delta u_1 \left(\tilde{H} - \tilde{u}^2 \right) + \tilde{u} \Delta u_2 - \overline{\Delta u_5} \right], \\ \tilde{\alpha}_1 &= \frac{1}{2\tilde{a}} [\Delta u_1 (\tilde{u} + \tilde{a}) - \Delta u_2 - \tilde{a} \tilde{\alpha}_2], \\ \tilde{\alpha}_5 &= \Delta u_1 - (\tilde{\alpha}_1 + \tilde{\alpha}_2), \end{aligned} \quad (2.22)$$

where $\Delta \mathbf{u} = \mathbf{u}_R - \mathbf{u}_L$ and $\overline{\Delta u_5} = \Delta u_5 - \tilde{\alpha}_3 \tilde{v} - \tilde{\alpha}_4 \tilde{w}$. \mathbf{u} — is a solution vector. This procedure can be naturally extended to the full 3D case, with simple directional rotation of coordinate system and considering each direction as flow direction in pseudo one dimensional problem.

Chapter 3

New Volume Penalization Methods

3.1 Brinkman-like Penalization for Low Fidelity Simulations

Due to the coarse grid used in Low Fidelity simulations slightly modified BPMs is developed, in order to match drag force that drives particles obtained numerically with experimental results. The idea is to introduce momentum penalization term which is quadratic in velocity

$$\frac{\partial \rho u_i}{\partial t} = RHS - \frac{\chi}{\eta} \rho (u_i - U_i^o)^2. \quad (3.1)$$

From one hand, it forces the solution to the proper value at the obstacle boundary even faster than in case of simple BP, on the other hand this expression makes modeling of the penalization easier and flow independent. Namely,

$$F_{drag} = \int_{\Omega} \frac{1}{\eta} \rho (u - U^o)^2 dV \sim \int_A \frac{1}{2} \rho C_d (u - U^o)^2 dA, \quad (3.2)$$

where Ω is the volume of the obstacle, and A is largest cross section area across the flow and χ is dropped since integration is performed only inside of the obstacle. For the sake of simplicity, let us consider uniform flow around triaxial ellipsoid with semi axes a, b and c . In that case $\Omega = \pi abc$ and $A = \pi ab$, under the assumption that flow goes along c semi axis. So, (3.2) simplifies into

$$\frac{\pi abc}{\eta} \sim \frac{1}{2} \pi ab C_d \implies \eta \sim \frac{c}{2C_d}. \quad (3.3)$$

3.2 Characteristic Based Volume Penalization

First VPM used in High Fidelity simulations is Brinkman Penalization. For testing purposes 1D Riemann problem for Euler equations are solved (see Fig. 3.1).

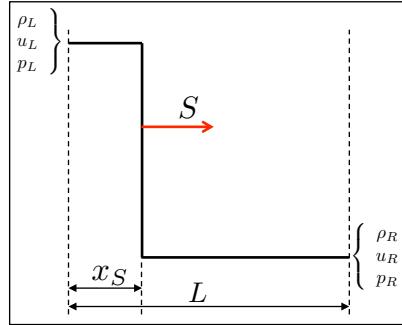


Figure 3.1: Riemann problem set up

To compare results, first, normal self-sustained shock problem is solved with reflection on the right domain boundary. Then right domain is extended and part of the domain is considered as solid obstacle and BP is used to model that solid obstacle/wall. Results revealed that using BP without any modification to supersonic flow leads to significant numerical errors such as phase lag and amplitude error, e.g. pressure seepage (see Fig. 3.2).

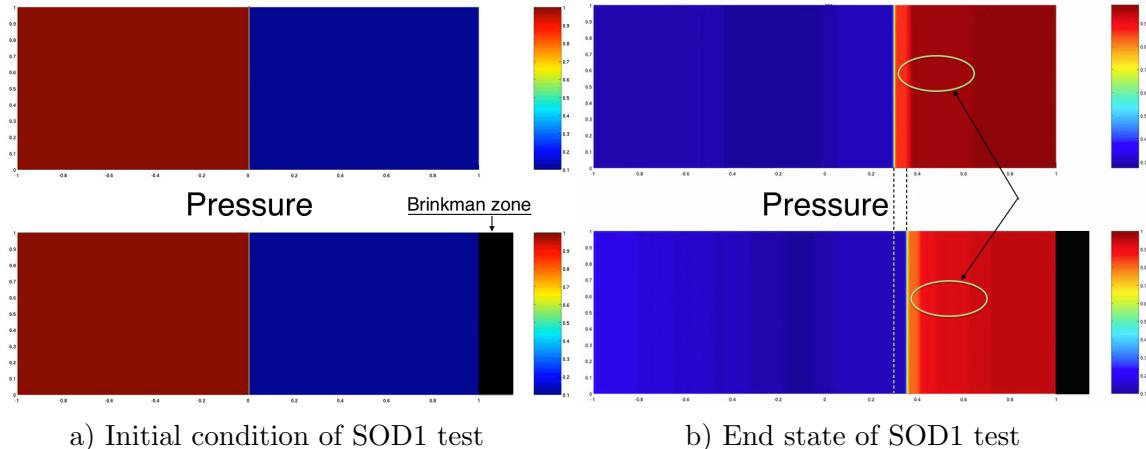


Figure 3.2: Normal reflection from the wall (top) and BP (bottom)

In order to overcome deficiencies of BP regarding flow regimes and boundary conditions, new VPM is developed. Let us consider general BC that one wants to mimic at the fluid-solid interface

$$\mathcal{L}(\bullet) = f_{boundary}, \quad (3.4)$$

where (\bullet) may stand for any of the integrated variables, and operator \mathcal{L} can describe either Dirichlet or Neumann boundary conditions, or Robin type BC. The idea is to replace evolution problem under the consideration as follows:

$$\frac{\partial \phi}{\partial t} = RHS \longrightarrow \frac{\partial \phi}{\partial t} = (1 - \chi)RHS - \frac{\chi}{\eta} [\mathcal{L}(\phi) - f_{boundary}]. \quad (3.5)$$

Obviously, BPM is a special case of CBVP, when $\mathcal{L} = \mathbb{1}$ and $f_{boundary} = U^o$ for velocity or $f_{boundary} = T^o$ for temperature. Let us consider the same Riemann problem, but using the following BC at the obstacle instead:

$$\left. \begin{array}{l} u_i^n|_{\partial\Omega} = U_i^{on} \\ \frac{\partial u_i^\tau}{\partial n} \Big|_{\partial\Omega} = 0 \\ \frac{\partial p}{\partial n} \Big|_{\partial\Omega} = 0 \end{array} \right\}, \quad (3.6)$$

where $u_i^n = u_j n_j n_i$ is a normal component of the velocity, $u_i^\tau = u_i - u_i^n$ is a tangential component and $\partial/\partial n \equiv n_j \partial/\partial x_j$ is a derivative in normal direction.

According to (3.5) evolution equations need to be changed as described below:

$$\left\{ \begin{array}{l} \frac{\partial u_i^n}{\partial t} = (1 - \chi)RHS - \frac{\chi}{\eta_b} (u_i^n - U_i^{on}) \\ \frac{\partial u_i^\tau}{\partial t} = (1 - \chi)RHS - \frac{\chi}{\eta_c} \frac{\partial u_i^\tau}{\partial n} \\ \frac{\partial p}{\partial t} = (1 - \chi)RHS - \frac{\chi}{\eta_c} \frac{\partial p}{\partial n}, \end{array} \right.$$

which can be translated to the equations in terms of integrated variables (with omitted

outside-of-the-obstacle part),

$$\left\{ \begin{array}{l} \frac{\partial \rho}{\partial t} = -\frac{1}{\eta_c} \frac{\partial \rho}{\partial n} \\ \frac{\partial \rho u_i^n}{\partial t} = -\frac{1}{\eta_b} \rho (u_i^n - U_i^{on}) + \nu_n \Delta u_i^n \\ \frac{\partial \rho u_i^\tau}{\partial t} = -\frac{1}{\eta_c} \frac{\partial \rho u_i^\tau}{\partial n} \\ \frac{\partial \rho e}{\partial t} = -\frac{1}{\eta_c} \frac{\partial \rho e}{\partial n} - \frac{1}{\eta_b} \rho u_j^n (u_j^n - U_j^{on}) + \frac{1}{\eta_c} u_j^n \frac{\partial \rho u_j^n}{\partial n} + u_j^n \nu_n \Delta u_j^n. \end{array} \right. \quad (3.7)$$

One can notice that in (3.7) there are extra terms in energy equation. They appear due to the fact that there is no so called “characteristic term” in normal momentum penalization and one need to add it and subtract in order to have characteristic term in energy equation as well. Also, extra diffusive term is added to the same normal momentum equation to enhance stability. These technique resulted improvements in comparison to regular BPM (see Fig. 3.3). One can find more results on using CBVP for variety of problems in Chapter 4, as well as benchmarks.

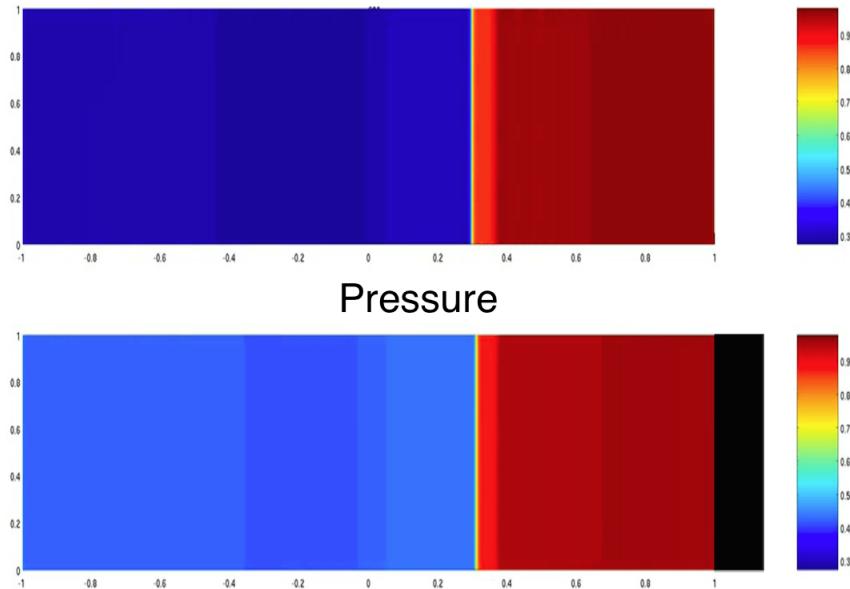


Figure 3.3: SOD1 test problem with CBVP

Chapter 4

High Fidelity Simulation Results

In this chapter one can find the results of benchmarking for CBVP applied to simple testing problem — Heat equation with different boundary conditions at the wall (Dirichlet, Neumann, Robin), as well as results of High Fidelity simulations with the use of CBVP for flows around different number and types of obstacles.

4.1 Benchmark of using Characteristic Based Volume Penalization

Applicability of CBVP on different BC is tested on 1D heat equation (4.1).

$$\frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2}. \quad (4.1)$$

On the right domain boundary the wall is placed, so one can observe reflection of the incident wave using CBVP and compare it to exact solution. The main error controlling parameter in CBVP is penalization parameter η , so it is varied to achieve convergence with exact solution.

4.1.1 Dirichlet Boundary Condition

This boundary condition (4.2), also called isothermal boundary condition, specifies temperature (integrated variable for heat equation) as a constant value at the obstacle/wall interface. In this particular benchmark the value is equal to 0.

$$T|_{wall} = 0 \quad \Rightarrow \quad \frac{\partial T}{\partial t} = k \frac{\partial^2 T}{\partial x^2} - \frac{\chi}{\eta_b} T. \quad (4.2)$$

Varying penalization parameter η_b led to solutions with different accuracy (see Fig. 4.1).

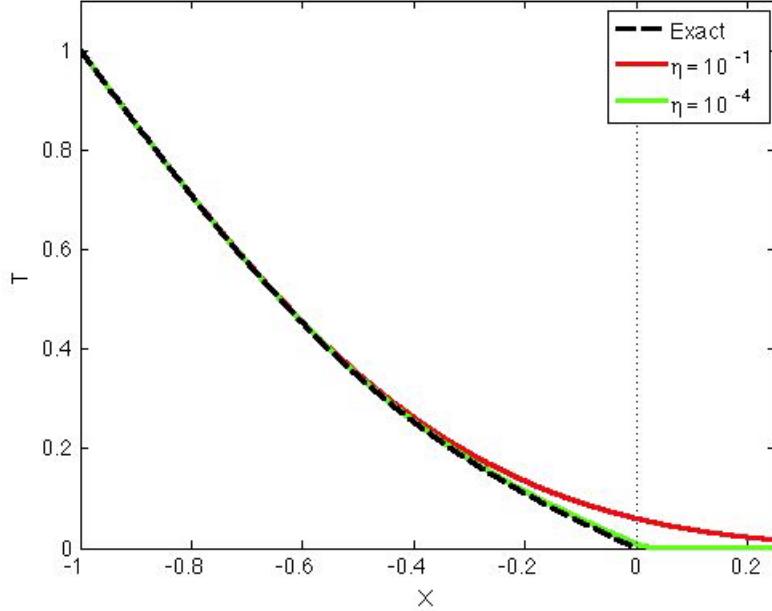


Figure 4.1: Temperature profile at the end of simulation. Dirichlet boundary condition

One can see that $\eta_b = 10^{-4}$ converges to the exact solution well. In fact, rate of convergence for this simulation is $\sqrt{\eta_b}$ [2], since it is Brinkman type penalization [8].

4.1.2 Neumann Boundary Condition

This boundary condition (4.3) specifies constant rate of heat amount going in to or out of the system. In case of zero rate it is called adiabatic boundary condition. In this benchmark latter is considered.

$$\left. \frac{\partial T}{\partial x} \right|_{wall} = 0 \quad \implies \quad \frac{\partial T}{\partial t} = (1 - \chi)k \frac{\partial^2 T}{\partial x^2} - \frac{\chi}{\eta_c} \frac{\partial T}{\partial x}. \quad (4.3)$$

Analogously to Dirichlet BC, varying penalization parameter η_c led to different accuracy of the solution (see Fig. 4.2). Rate of convergence for this type of simulations is η_c [2]. One can notice that penalization parameter for Neumann type BC is different than for

Dirichlet type BC, which is connected to different time scales of the processes. In case of BP solution at the boundary decays inside of the obstacle, on the other hand for characteristic BC solution is translated inside of the obstacle.

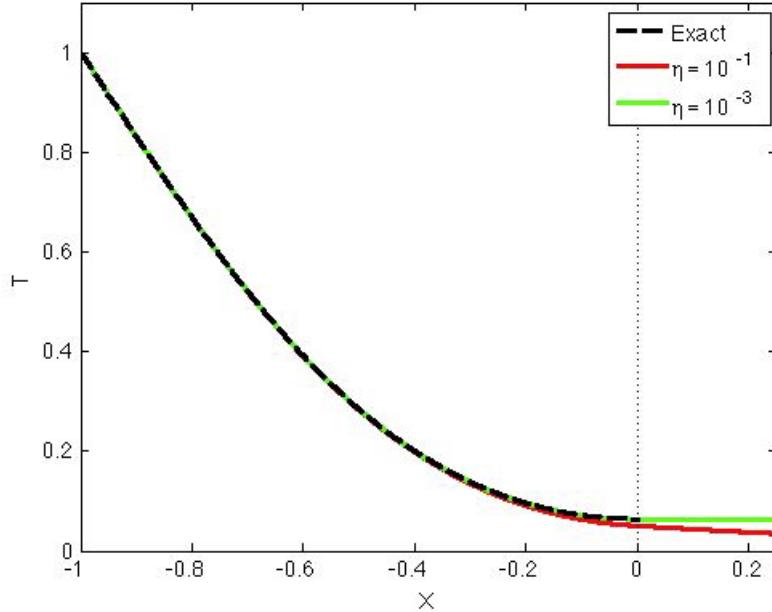


Figure 4.2: Temperature profile at the end of simulation. Neumann boundary condition

4.1.3 Robin Boundary Condition

Last benchmark problem is to mimic Robin boundary condition at the obstacle interface, when both the solution value and its derivative are specified at the boundary. In this particular problem (4.4) is used as a BC.

$$T + 2 \frac{\partial T}{\partial x} = 5 \quad \Rightarrow \quad \frac{\partial T}{\partial t} = (1 - \chi)k \frac{\partial^2 T}{\partial x^2} - \frac{\chi}{\eta_r} \left(T + 2 \frac{\partial T}{\partial x} - 5 \right). \quad (4.4)$$

In this type of simulations rate of convergence is predominated due to characteristic based terms as η_r [2] (see Fig. 4.3).

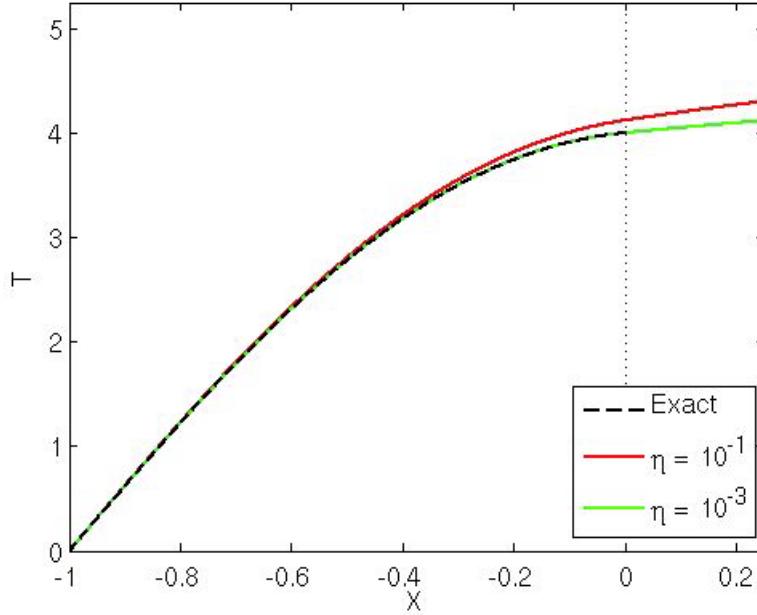


Figure 4.3: Temperature profile at the end of simulation. Robin boundary condition

4.2 Simulation Results

After the method is validated and convergence rates are studied, variety of 2D simulations of supersonic inviscid flow are performed. Namely, flow around single and multiple stationary and moving cylinders, wedges with subcritical and supercritical angles. It is worth to mention that at this point no quantitative analysis of applicability of CBVP is done for these simulations. All results are treated based on the assumption that convergence rates defined in heat equation benchmarks hold for these simulations according to the used BCs (either $\sim \sqrt{\eta}$ or $\sim \eta$).

4.2.1 Flow around the Wedge

In this set of simulations supersonic flow around both subcritical and supercritical wedges are considered. Incident shock speed is $M = 1.5$ (see Fig. 4.4). In case of subcritical wedge, i.e. angle at the apex is less than certain critical angle θ_{crit} , secondary oblique shock is performed.

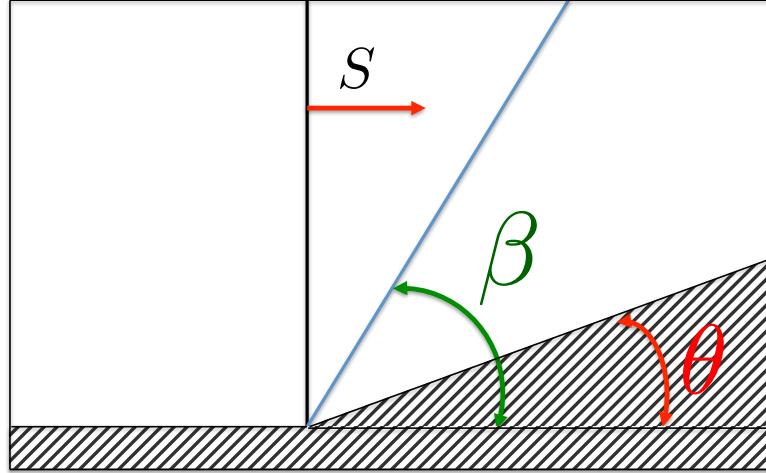


Figure 4.4: Schematic of the flow around the wedge

Between the wedge angle θ and oblique shock deflection angle β there is a relation called “ $\theta - \beta - M$ ” equation (4.5) (see Fig. 4.5).

$$\tan \theta = 2 \cot \beta \frac{M \sin^2 \beta - 1}{M^2 (\gamma + \cos 2\beta) + 2}. \quad (4.5)$$

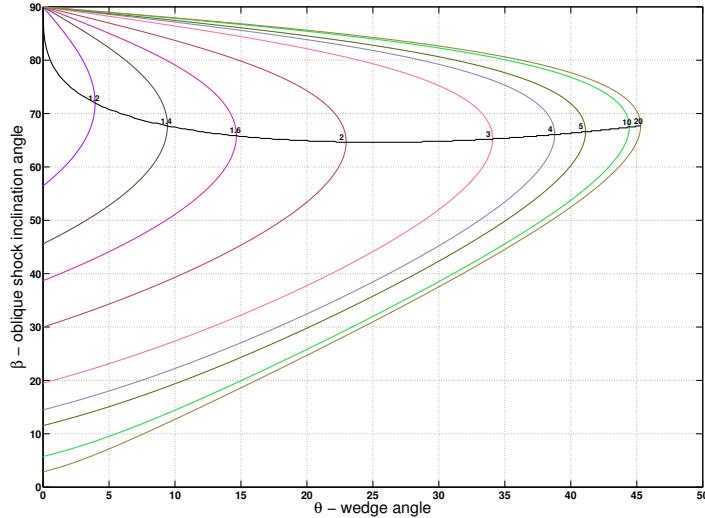


Figure 4.5: $\theta - \beta - M$ relation plot

For this particular setup, one can find that $\theta_{crit} \approx 15^\circ$, and subcritical wedge angle is

$\theta = 10^\circ$. For the wedge angle $\theta > \theta_{crit}$ detached bow shock is formed. Supercritical angle is $\theta = 25^\circ$.

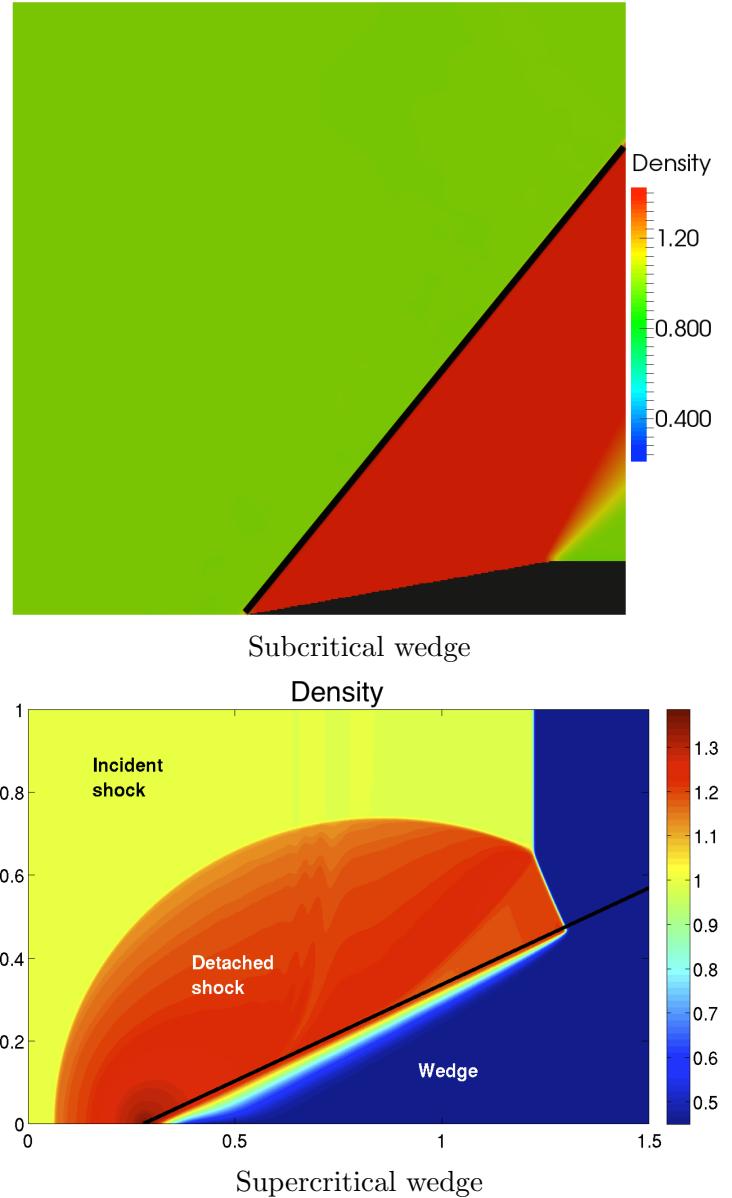


Figure 4.6: Flow around the wedge results

4.2.2 Flow around Single and Multiple Stationary Cylinders

In this set of simulations, same $M = 1.5$ shock impinges either single or multiple solid cylinders (see Fig. 4.7). Unlike the flow around the wedge, in case of cylinder

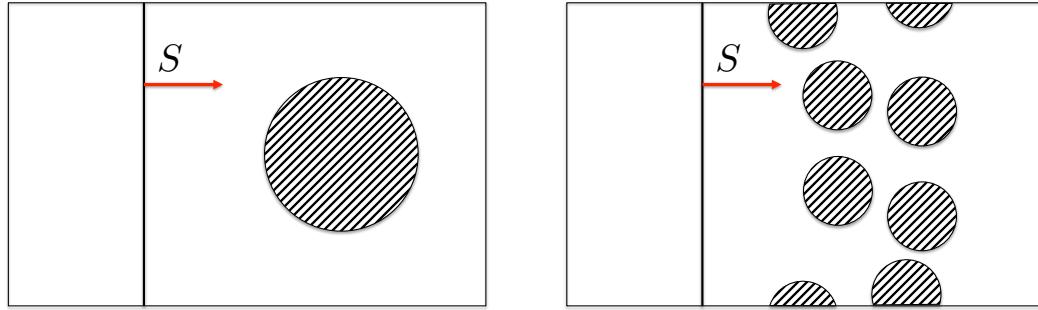


Figure 4.7: Single and multiple cylinders case schematics

always detached shock will be formed. There is no analytical solution to this problem.

In case of single cylinder, one can see reflected bow shape shock (see Fig. 4.8).

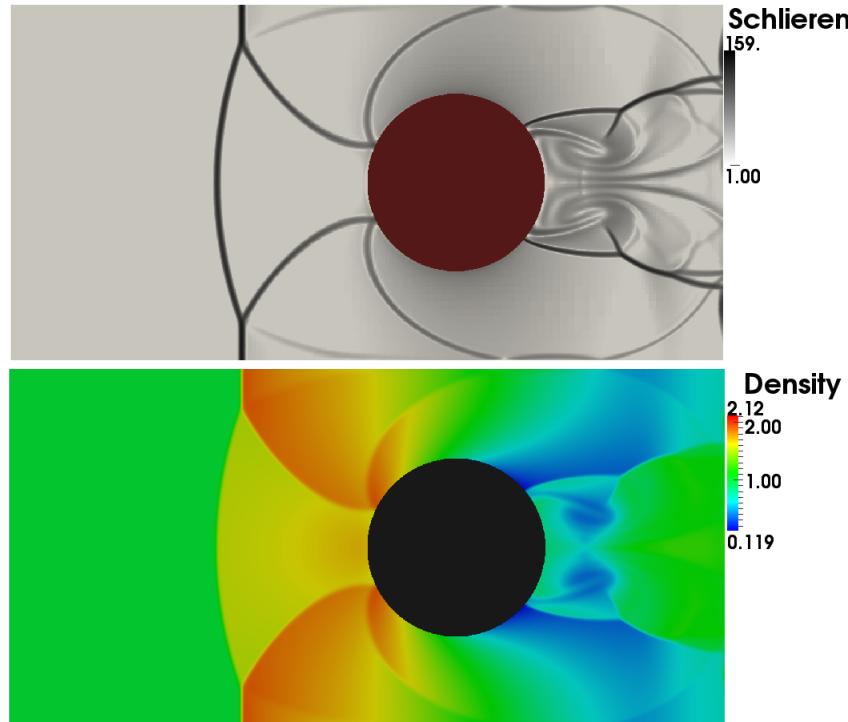


Figure 4.8: Supersonic flow around single cylinder

In case of multiple cylinders one can also notice shock-shock interactions (see Fig. 4.9).

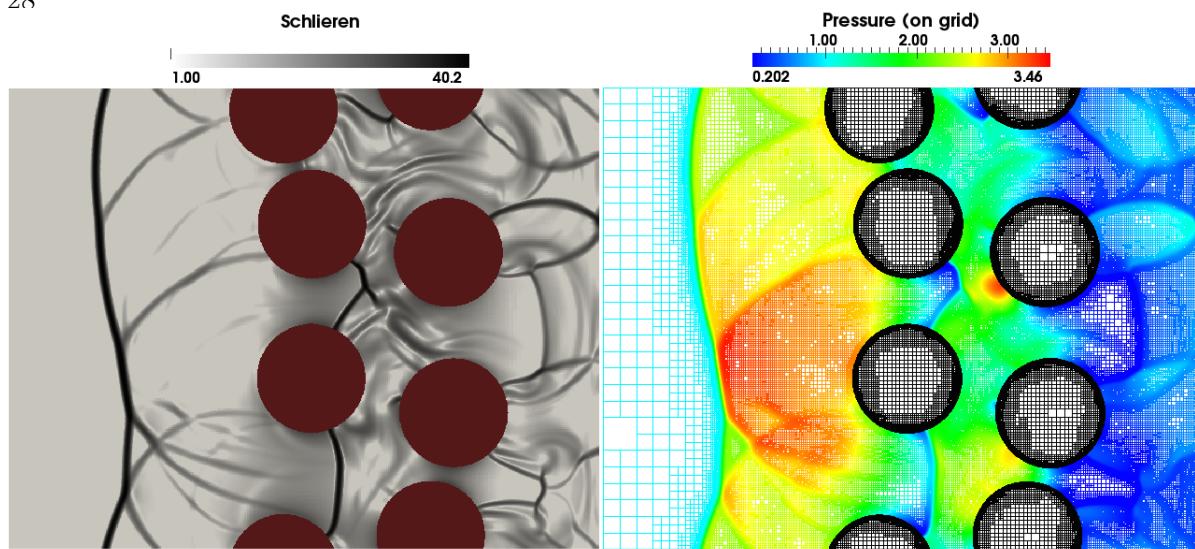


Figure 4.9: Supersonic flow around multiple cylinders

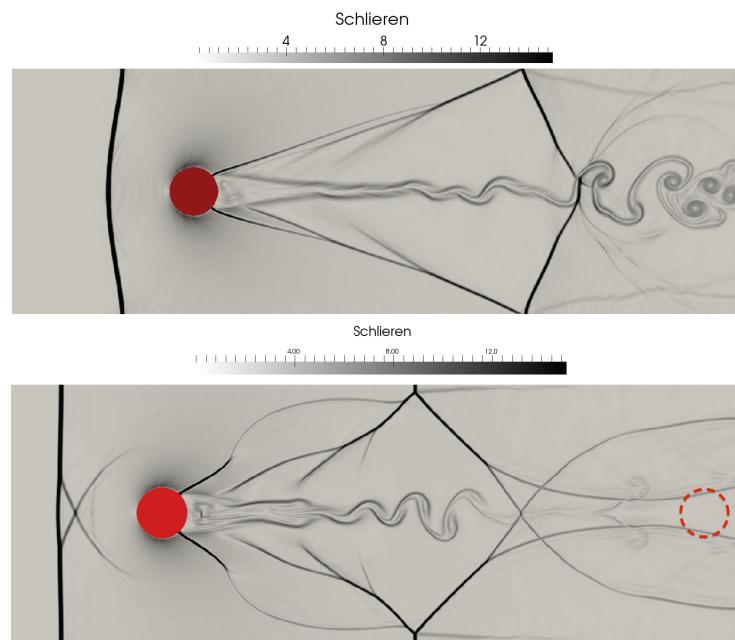


Figure 4.10: Supersonic flow around stationary and moving cylinders

4.2.3 Flow around Single Stationary and Moving Cylinder

This case is to demonstrate the ability of the CBVP to handle physically equivalent phenomena considered from two different frame of references, moving and non

moving, within the acceptable level accuracy. For this purpose, two identical numerical frameworks are developed. One with supersonic uniform background flow and stationary cylinder, and another with zero background flow and a single cylinder moving supersonically embedded to that flow. Flow pressure and density are kept identical in both set ups (see Fig. 4.10). One can notice that results are not completely identical. The reason is at this point methodology is not completely consistent when switching to moving frame of reference, and this issue will be addressed in future work (see Chapter 6).

Chapter 5

Low Fidelity Simulation Results

In this chapter Low Fidelity simulation results are presented. These type of simulations utilize coarse grid and volume penalization is applied through smearing surface defined drag force into artificial volume defined penalization force. Unlike High Fidelity simulations in these cases fully 3D configuration is considered. Due to the inability to resolve viscous boundary layer Euler equations are solved. Dynamic penalization parameters are taken from 3D DEM solver and coupled with CFD/Roe solver. At the end of time step CFD solver calculates forces related to both pressure and artificial forcing terms and passes back to DEM solver, completing two way coupling (see Fig. 5.1).

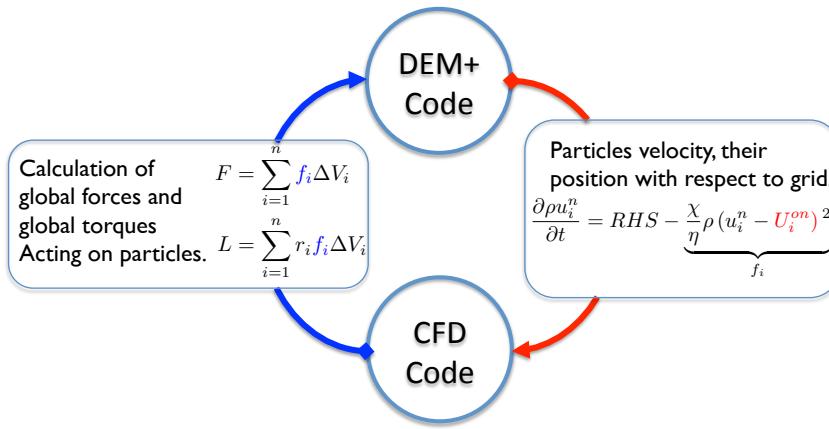


Figure 5.1: CFD/DEM coupling schematics

Set up for all simulations is presented in Fig. 5.2. Normal shock with $Ma = 5$ approaches either one or multiple ellipsoidal particles in upward direction. Gravitational

effects are neglected. Particles either fixed in space or can be driven by the flow, and interact with each other in case of several particles.

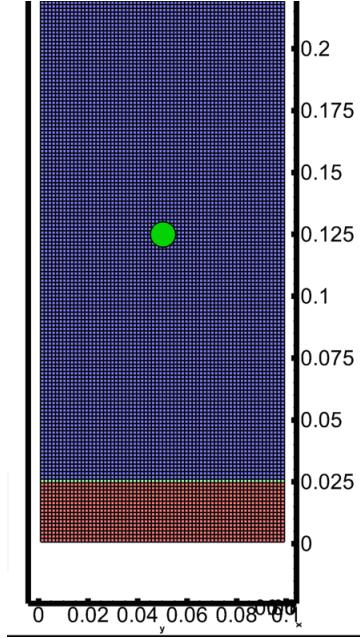


Figure 5.2: Low Fidelity simulations set up

Simulations were carried for wide variety of particles of ellipsoidal shape and size by changing values of semi axes for both free and stationary particles.

5.1 Flow around Single Stationary Sphere

For testing purposes VPM was first used for the supersonic flow around fixed spherical particle (see Fig. 5.3). On the left plot one can find pressure force and penalization force. Latter reaches a plateau after shock passes the obstacle since it is stationary and force is proportional to the square of relative velocity according to (3.2). Pressure force on the other hand becomes negative since energy equation is not penalized in a consistent way. This issue was resolved in later simulations. Also, one can notice obstacle on the left plot, which has polyhedral shape due to the coarse mesh.

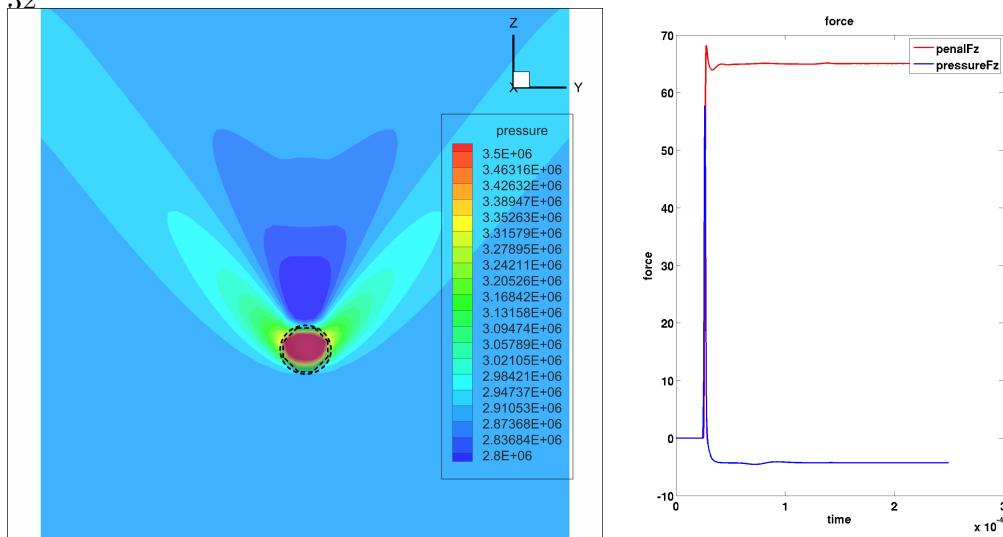


Figure 5.3: Flow around single stationary sphere

5.2 Flow around Moving Sphere

Next step in the project is to test the methodology on free spherical particle (see Fig. 5.4).

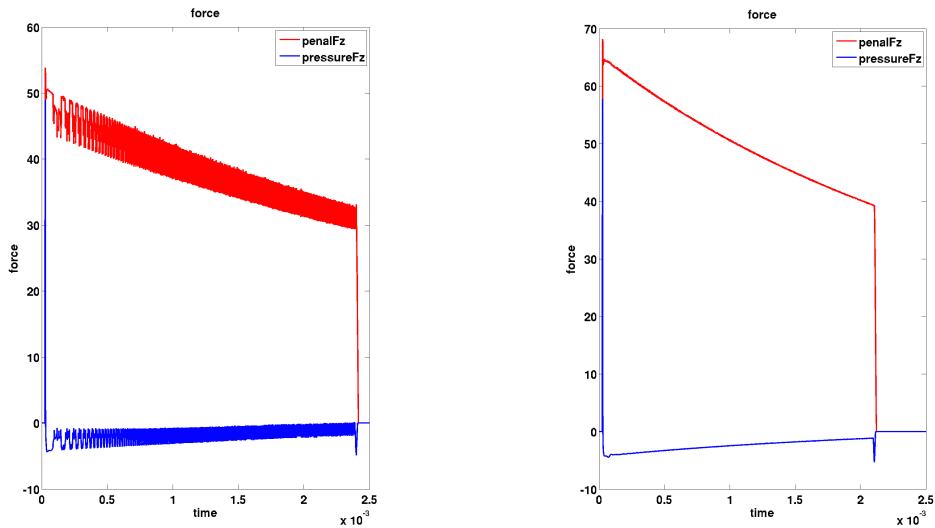


Figure 5.4: Flow around single free sphere

On the left plot forces demonstrate noisy behavior. The reason is in coarse grid, so when particle moves the grid cell that particle occupies at any given moment of time

might suddenly change when time advances. To solve this issue partial volumes taken by particles are calculated (see Fig. 5.5) and results can be found on the right plot. Negative pressure force is still present.

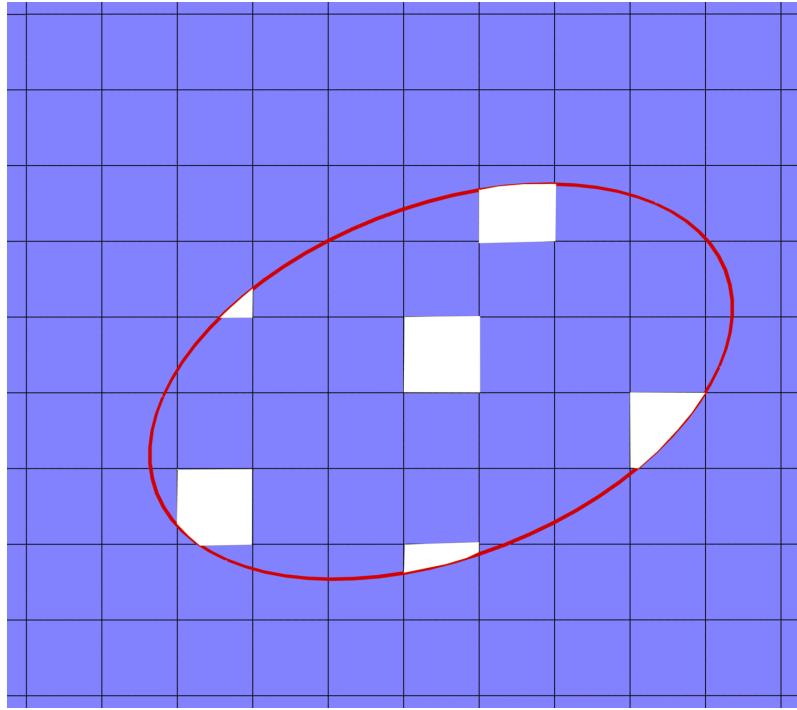


Figure 5.5: Grid cells occupied partially by the particle calculated more accurately

5.3 Flow around Several Free Ellipsoids

The method is tested on two and more particles (see Fig. 5.6). During the simulation smaller particle at lower position accelerates more due to the smaller mass and penalization force depends on the relative velocity only. After hitting bigger particle on top it slows down, but later accelerates again. This process continues until both particles stick together. Negative pressure is still present.

In case of substantial number of particles situation changes, instead of going through the particles and forming bow shape shocks as in case of few particles, shock reflects from the wall of particles as if they cannot be penetrated (see Fig. 5.7).

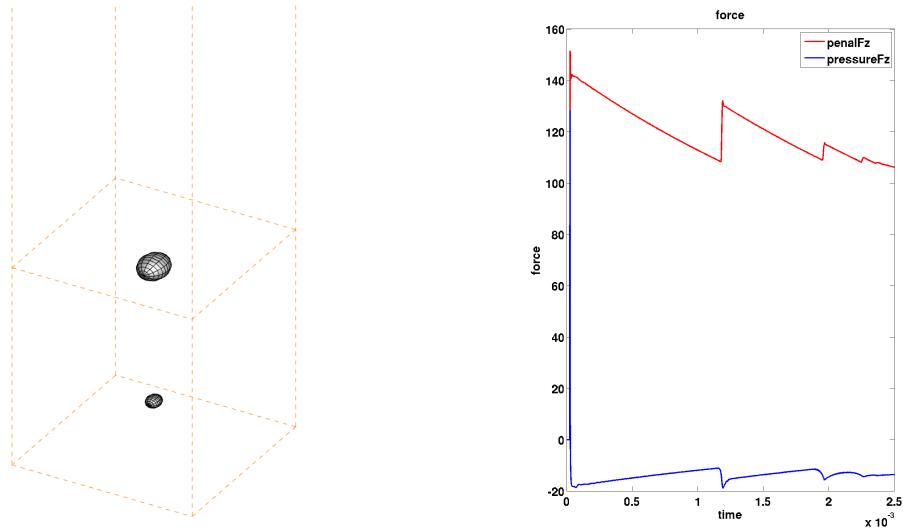


Figure 5.6: Flow around two ellipsoidal particles

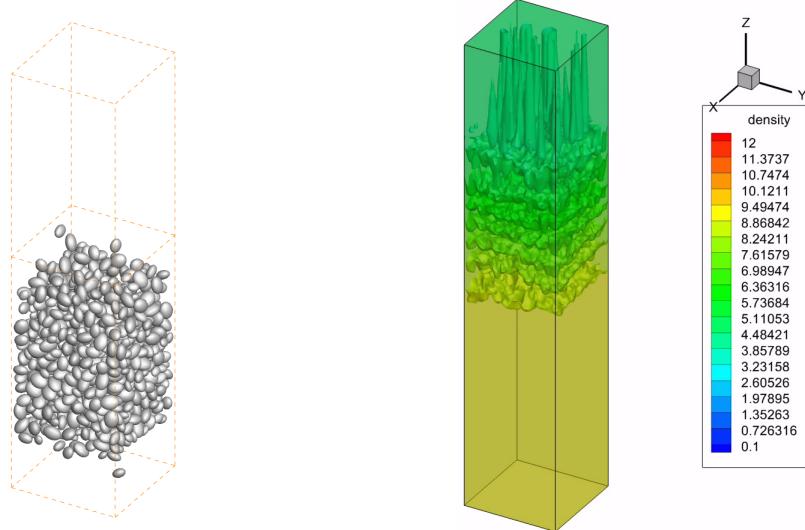


Figure 5.7: Flow around 1k ellipsoidal particles

5.4 Flow around “Disk” and “Needle” like obstacles

Ellipsoidal, and potentially polyellipsoidal, particles have a flexibility of imitating different shapes. For example, by squeezing one of the semi axis and having other two approximately equal to each other one can get disk shape particle. By squeezing two semi axis and keeping third one relatively large one can imitate needle like object (see Fig. 5.8).

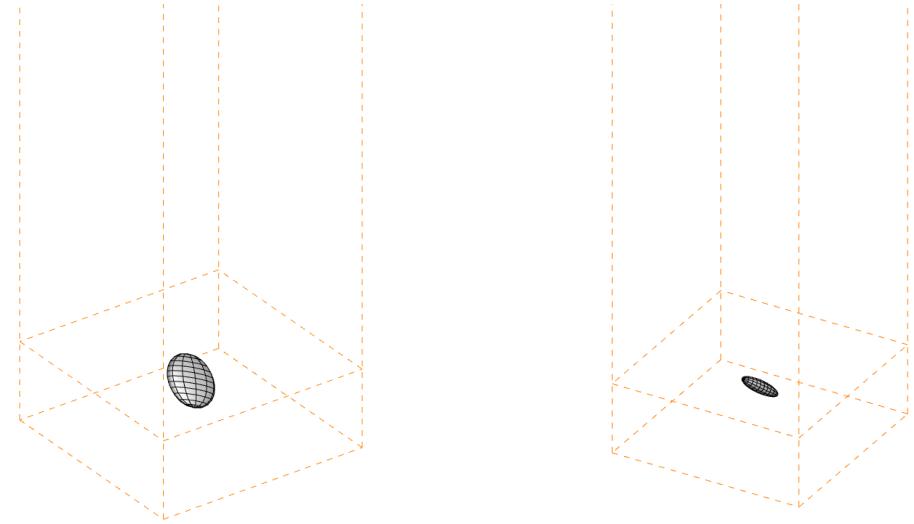


Figure 5.8: Disk shape (left) and needle shape (right) particles

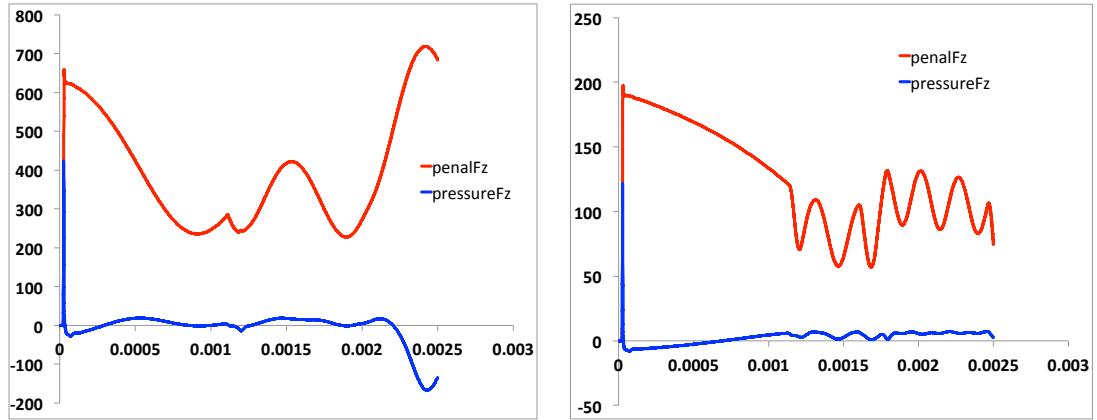


Figure 5.9: Dynamics of the flow around disk (left) and needle (right)

From results, one can see that penalization force oscillates at some point. The reason is particles bounce from vertical sides of the domain loosing energy (see Fig. 5.9). Negative pressure is still present.

5.5 Flow around Stationary Ellipsoid, Negative Pressure Fix

One can notice that all previous simulations have negative pressure issue. In order to fix it more accurate penalization is chosen (see Fig. 5.10). It reproduces density penalization according to (2.4) and energy penalization according to (3.7). Latter is

necessary step, since it makes equations consistent with each other, but former is meant to take into account the fact that despite small penalization parameter η that acts like permeability, fluid penetrates to the particle and in case of big particle or large amount of particles, error might become significant.

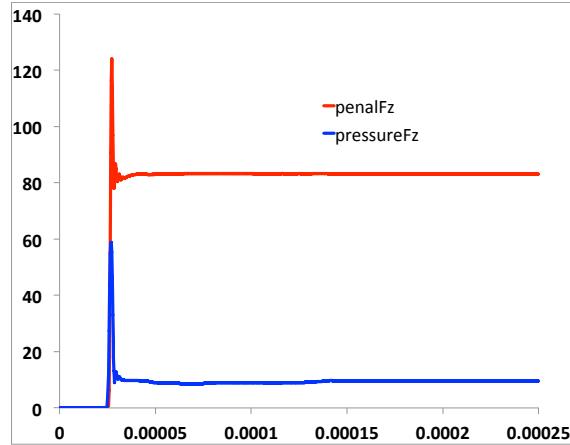


Figure 5.10: Flow around single ellipsoid with porosity fix and energy fix enabled

Chapter 6

Future Work

6.1 High Fidelity Simulations

1. **Invariance in inertial frame of reference.** For High Fidelity simulations it is planned to improve CBVP in order to handle proper switch to moving frame of reference to maintain invariance of dynamic variables such as pressure and density in inertial frame of reference. This is a necessary condition that method will work for moving obstacles properly.
2. **Complex shapes.** This step is planned to be accomplished in parallel with previous step. At this point only simple shape particles are considered — cylinders, spheres, ellipsoids, wedges. The plan is to be able to handle particles of arbitrary shape. Since we use the concept of distance function in our simulations, we might end up using Level Set methods [13] to find and track masking functions of more complex shapes in the future.
3. **Method validation for supersonic flows.** So far method is tested and benchmarked against simple structured problems such as Heat equation and 2D acoustic wave propagation, it is not tested on supersonic flows yet. It is intended to finish validation on both viscous and inviscid supersonic flows after the step 1.
4. **3D Cases.** After the method is validated and its applicability is justified, it is planned to perform 3D simulations of flows around complex geometries.

5. **Developing models.** Last step and ultimate goal is to develop models based on these High Fidelity simulations to use in Low Fidelity simulations.

6.2 Low Fidelity Simulations

1. **Parallelize the code.** At this point CFD/DEM coupled code is serial, which is getting problematic since size of the problem is increasing. First thing to improve in Low Fidelity simulations part of the project is properly parallelize and test the code. DEM code is already parallelized, so main problem is to parallelize CFD code and synchronize it with DEM part.
2. **More realistic cases.** Particles considered as of now are ellipsoidal. To imitate real particles that correspond to different ejecta from the explosion we are going to take into account parameters like roughness, to mimic sharp edged particles at statistical level.
3. **Match with experiments.** Low Fidelity simulations capable of reproducing physical experiments on blast wave propagation at qualitative level at the blast initiation stage. The plan is to improve methodology so numerical results match with experiments with desired level of confidence at accuracy using models obtained by performing High Fidelity simulations.
4. **Couple with other numerical frameworks.** This project is a part of big research on multi physics phenomena on blast wave propagation. There are several research groups working on different parts of the problem with different timescales. As a consequence there several numerical platforms each groups is working on. The ultimate goal of our Low Fidelity simulations is to couple CDF/DEM code with other codes to get final code that can handle all physical levels of the phenomena from the beginning to the very end.

Bibliography

- [1] R. Regueiro and et.al., “Onr muri project on soil blast modeling and simulation,” in Dynamic Behavior of Materials (B. Sang, D. Casem, and J. Kimberley, eds.), vol. 1 of Society for Experimental Mechanics Series, (Bethel, CT, USA), pp. 341–353, Society for Experimental Mechanics, Inc., Springer, 2014.
- [2] E. Brown-Dymkoski, N. Kasimov, and O. V. Vasilyev, “A characteristic based volume penalization method for general evolution problems applied to compressible viscous flows,” J. Comp. Phys., vol. 262, pp. 344–357, April 2014.
- [3] J. Regele and O. Vasilyev, “An adaptive wavelet-collocation method for shock computations,” Intl. Journal of CFD, vol. 23, no. 7, pp. 503–518, 2009.
- [4] K. Schneider and O. V. Vasilyev, “Wavelet methods in computational fluid dynamics,” Ann. Rev. Fluid Mech., vol. 42, pp. 473–503, 2010.
- [5] E. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction. Springer, 3 ed., 2009.
- [6] C. Peskin, “Numerical analysis of blood flow in the heart,” J. Comp. Phys., vol. 25, no. 3, pp. 220–252, 1977.
- [7] E. Arquis and J. Caltagirone, “Sur les conditions hydrodynamiques au voisinage d’une interface milieu fluide – milieu poreux: application à la convection naturelle,” C.R. Acad. Sci. Paris II, vol. 299, pp. 1–4, 1984.
- [8] Q. Liu and O. Vasilyev, “A brinkman penalization method for compressible flows in complex geometries,” J. Comp. Phys., vol. 227, no. 2, pp. 946–966, 2007.
- [9] D. L. Donoho, “Interpolating wavelet transforms,” tech. rep., Department of Statistics, Stanford University, October 1992.
- [10] W. Sweldens and P. Shroder, Wavelets in the Geosciences, vol. 90 of Lecture Notes in Earth Sciences, ch. 2, pp. 72–107. Springer, 2000.
- [11] J. Liandrat and P. H. Tchamitchian, “Resolution of the 1d regularized burgers equation using a spatial wavelet approximation,” Tech. Rep. 90-83, NASA, 1990.
- [12] P. Roe, “Approximate riemann solvers, parameter vectors, and difference schemes,” J. Comp. Phys., vol. 43, no. 2, pp. 357–372, 1981.

- [13] J. A. Sethian, Level Set Methods and Fast Marching Methods. Cambridge Monograph on Applied and Computational Mathematics, Cambridge University Press, 2 ed., 1999.