Sedimentation of a Sphere in Viscous Fluid

Summer Internship
REPORT

by

Priyanshu Singh

Under the Supervision of

Prof. Rajesh Ranjan

Department of Aerospace Engineering
Indian Institute of Technology Kanpur

July 2025

Declaration

I, **Priyanshu Singh**, hereby declare that:

1. The project report entitled **"Sedimentation of a Sphere in Viscous Fluid"** represents my original work conducted during the internship at the Department of Aerospace Engineering, IIT Kanpur.
2. This work was performed under the supervision of **Prof. Rajesh Ranjan** and has not been submitted elsewhere for any academic qualification.
3. Wherever contributions of others are involved, every effort has been made to acknowledge them clearly.

Student's Signature: **Submission Date:**

Priyanshu Singh
Internship Student
Department of Aerospace Engineering
IIT Kanpur

24/07/2025

Abstract

This project investigates the sedimentation of a sphere in viscous fluid using Computational Fluid Dynamics (CFD). We compared two different OpenFOAM solvers—DPMFoam and overPimpleDyMFoam—to simulate how a sphere settles under gravity in fluid. The study focused on calculating key parameters like terminal velocity, drag force, and drag coefficient.

Using both solvers, we tracked the sphere's motion as it accelerated from rest until reaching constant terminal velocity. The DPMFoam solver employed Lagrangian particle tracking, while overPimpleDyMFoam used dynamic meshing with overset grids. Both approaches showed good agreement.

Simulations at Reynolds number 1.6 revealed significant deviations from theoretical predictions: DPMFoam calculated a drag coefficient of approximately 4.85 (substantially lower than Stokes' 15.34 and Schiller-Naumann's 18.20), while overPimpleDyMFoam yielded approximately 20.77 (higher than both models). These results demonstrate how established theories can underestimate real behavior in intermediate flow regimes. Computationally, DPMFoam proved more efficient for basic simulations, whereas overPimpleDyMFoam provided more detailed force analysis despite greater resource requirements.

These findings demonstrate how OpenFOAM can effectively model sedimentation processes, with practical applications in water treatment, pharmaceutical processing, and environmental engineering.

Acknowledgements

I express my deepest gratitude to my supervisor, Prof. Rajesh Ranjan, for his expert guidance in computational fluid dynamics throughout this research. His profound knowledge of multiphase flow modeling was essential for my implementation of sedimentation simulations using OpenFOAM's DPMFoam and overPimpleDyMFoam solvers. Prof. Ranjan's insights on solver selection criteria and validation methodologies significantly enhanced my comparative analysis of particle tracking approaches.

I acknowledge the Department of Aerospace Engineering at IIT Kanpur for providing the advanced computational resources required for this work. The department's high-performance computing cluster enabled the extensive parametric studies of sphere sedimentation in viscous fluids, particularly the demanding overPimpleDyMFoam simulations with dynamic mesh refinement.

Finally, I thank my family for their unwavering support during the intensive simulation and validation phases of this project. Their encouragement sustained me through complex debugging of solver configurations and visualization of three-dimensional flow fields.

Contents

List of Figures

Chapter 1
Introduction

1.1 Background

Sedimentation of particles in viscous fluids is a fundamental phenomenon encountered in numerous engineering and natural processes. Understanding the dynamics of particle settling is crucial for applications ranging from water treatment and pharmaceutical processing to environmental engineering and geophysical systems. The behavior of a settling particle is governed by the complex interplay between gravitational, buoyant, and drag forces, making computational fluid dynamics (CFD) an essential tool for analysis.

The sedimentation process involves a particle initially accelerating under gravity until it reaches terminal velocity, where the drag force balances the net gravitational force. This terminal velocity depends on various factors including particle size, density, fluid properties, and flow regime characterized by the Reynolds number. For low Reynolds numbers ($Re < 1$), Stokes' law provides an analytical solution:

$$F_d = 6\pi\mu r v_t \tag{1.1}$$

while higher Reynolds numbers require empirical correlations such as the Schiller-Naumann equation:

$$C_d = \frac{24}{Re}(1 + 0.15Re^{0.687}) \tag{1.2}$$

OpenFOAM, an open-source CFD platform, provides multiple solvers capable of simulating particle-fluid interactions. This study focuses on two distinct approaches:

- `DPMFoam` (Discrete Phase Model): Lagrangian particle tracking with Eulerian fluid phase
- `overPimpleDyMFoam` (Overlapping mesh with dynamic motion): Dynamic mesh with overset grid capabilities

each offering unique advantages and computational methodologies.

1.2 Motivation

The motivation for this comparative study stems from several key factors:

Scientific Understanding: Gaining deeper insights into the fundamental physics of particle sedimentation by comparing different numerical approaches provides valuable understanding of the underlying mechanisms governing particle-fluid interactions.

Solver Validation: Comparing multiple OpenFOAM solvers against established theoretical models and experimental data helps validate the accuracy and reliability of different computational approaches for particle settling problems.

Method Selection Guidance: Different solvers have varying computational requirements, setup complexity, and accuracy levels. This comparison provides guidance for selecting appropriate methods based on specific application requirements and available computational resources.

Bridging Theory and Practice: By implementing simulations from scratch (`DPMFoam`) and adapting existing setups (`overPimpleDyMFoam`), this study bridges the gap between theoretical understanding and practical implementation of CFD methods.

Educational Value: The project provides hands-on experience with advanced CFD techniques, mesh generation, boundary condition specification, and result interpretation, which are essential skills for computational fluid dynamics practitioners.

1.3 Objectives

The primary objectives of this study are:

1. To simulate sphere sedimentation using two different OpenFOAM solvers: `DPMFoam` and `overPimpleDyMFoam`
2. To calculate terminal velocity, drag force, and drag coefficient for the settling sphere
3. To compare simulation results with theoretical predictions from Stokes' law and Schiller-Naumann correlation
4. To analyze the advantages and limitations of each solver approach
5. To validate results against published literature data
6. To characterize wake formation and vortex shedding patterns at different Reynolds numbers
7. To assess computational efficiency and accuracy trade-offs between solvers

1.4 Report Organization

This report is structured as follows: Chapter 2 details the computational methodology and solver implementations, Chapter 3 presents results and comparative analysis, and Chapter 4 concludes with key findings and recommendations for future work.

Chapter 2
Numerical Methodology

2.1 Computational Domain and Mesh Generation

The computational domain for both solvers was designed as a rectangular cuboid with dimensions of 100 mm × 100 mm × 160 mm, scaled appropriately to meters in the simulation. This domain size was selected to minimize wall effects while maintaining computational efficiency. The aspect ratio ensures sufficient space for the particle to reach terminal velocity before interacting with the bottom boundary.

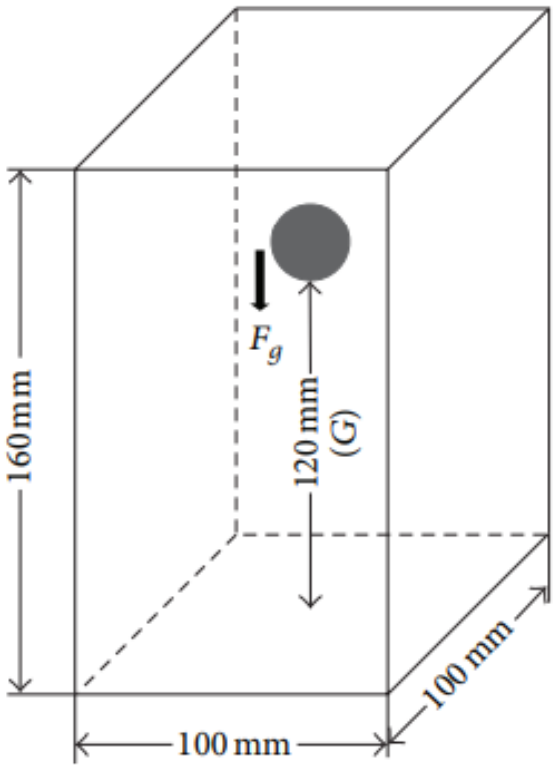

Figure 2.1: Computational domain geometry and dimensions (all units in meters). Adapted from: Alapati et al., *Simulation of Sedimentation of a Sphere in a Viscous Fluid Using the Lattice Boltzmann Method Combined with the Smoothed Profile Method*, Kyungsung University and Dong-A University.

The mesh was generated using a structured hexahedral grid with 70 × 70 × 100 cells, resulting in uniform cell grading throughout the domain. This mesh density provides a good balance between computational accuracy and efficiency, with adequate resolution to capture the flow field around the settling sphere.

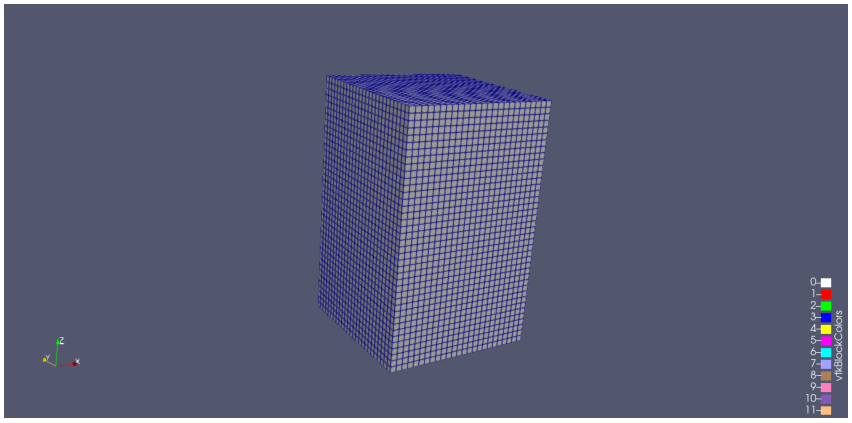

Figure 2.2: Complete view of the mesh structure for the simulation domain

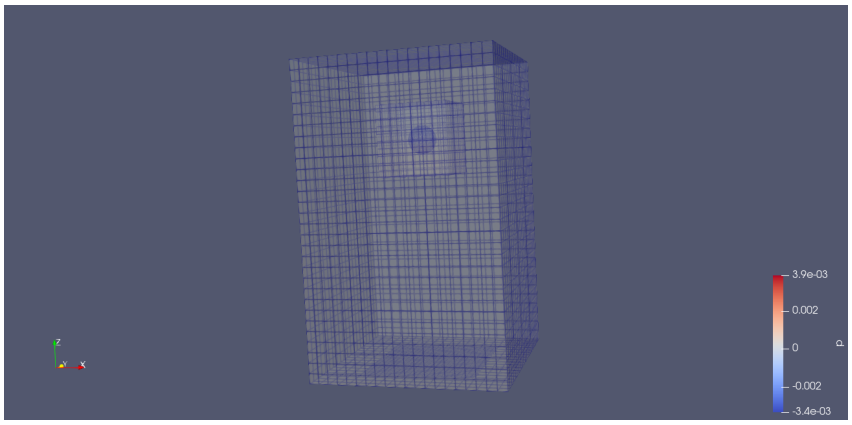

Figure 2.3: Close-up view of the refined mesh region around the sphere

The mesh quality was verified using OpenFOAM utilities, ensuring proper aspect ratios, skewness, and orthogonality parameters were within acceptable limits for stable numerical computation.

2.2 DPMFoam Solver Methodology

DPMFoam (Discrete Phase Model Foam) is a Lagrangian particle tracking solver that treats particles as discrete entities moving through a continuous fluid phase. The solver implements two-way coupling between the continuous and discrete phases, accounting for momentum exchange between the particle and surrounding fluid.

2.2.1 Governing Equations

The continuous phase (fluid) is governed by the incompressible Navier-Stokes equations:

Continuity equation:

$$\nabla \cdot \mathbf{U} = 0 \tag{2.1}$$

Momentum equation:

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot (\mathbf{U}\mathbf{U}) = -\frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{U} + \frac{S_p}{\rho} \tag{2.2}$$

where S_p represents the source term from particle-fluid interaction.

For the discrete phase (particle), the motion is governed by Newton's second law:

$$m_p \frac{d\mathbf{U}_p}{dt} = \mathbf{F}_g - \mathbf{F}_d - \mathbf{F}_b \tag{2.3}$$

where:

- m_p = particle mass
- $\mathbf{U}_p$ = particle velocity
- $\mathbf{F}_d$ = drag force
- $\mathbf{F}_g$ = gravitational force
- $\mathbf{F}_b$ = buoyancy force

2.2.2 Drag Force Calculation

Since DPMFoam does not directly compute forces on particles, a manual calculation approach was implemented:

1. **Velocity Extraction:** Particle velocity data was extracted at each time step from the simulation results
2. **Acceleration Calculation:** Numerical differentiation was applied to velocity data:

$$\mathbf{a} = \frac{\mathbf{U}_p(t + \Delta t) - \mathbf{U}_p(t)}{\Delta t} \tag{2.4}$$

3. **Net Force Computation:** Using Newton's second law:

$$\mathbf{F}_{net} = m_p \times \mathbf{a} \tag{2.5}$$

4. **Drag Force Derivation:** From force balance at terminal velocity:

$$\mathbf{F}_d = \mathbf{F}_g - \mathbf{F}_b - \mathbf{F}_{\text{net}} \tag{2.6}$$

2.2.3 Solver Configuration

The DPMFoam setup required creation of several key files in the constant directory:

- `kinematicCloudProperties`: Defines particle properties, injection parameters, and physical models
- `kinematicCloudPositions`: Specifies initial particle positions and velocities

Key solver settings included:

Table 2.1: DPMFoam solver configuration parameters

Parameter	Value
Application	DPMFoam
Time step	2×10^{-4} s
End time	1.5 s
Write interval	Every 150 time steps
Coupling type	Two-way
Flow regime	Laminar
Turbulence model	None
Particle tracking	Lagrangian

2.3 overPimpleDyMFoam Solver Methodology

overPimpleDyMFoam is an advanced solver that combines the PIMPLE algorithm with overset mesh technology and dynamic mesh capabilities. This solver can directly compute forces on immersed bodies using function objects, providing more straightforward force analysis.

2.3.1 Overset Mesh Technology

The solver utilizes overset (Chimera) mesh technology, where multiple overlapping grids are used:

- **Background mesh:** Covers the entire computational domain
- **Component mesh:** Surrounds the moving sphere with high resolution

This approach allows for natural handling of moving boundaries without mesh deformation or remeshing. The overset interpolation maintains solution accuracy at grid interfaces.

2.3.2 Built-in Force Computation

Unlike DPMFoam, overPimpleDyMFoam provides direct force calculation through function objects. The force calculation was implemented in `controlDict`:

```
forces1
{
    type          forces;

    libs          ("libforces.so");

    writeControl  timeStep;
    timeInterval  1;

    log           yes;

    patches       ("sphere");
    rho           rhoInf;     // Indicates incompressible
    log           true;
    rhoInf        970; // Redundant for incompressible
    origin (0 0 0);
    rotation
        {
            type            axes;
            e3              (0 0 1);
            e1              (1 0 0);
        }
}
```

This configuration computes forces on the sphere surface at each time step, outputting drag, lift, and moment coefficients.

2.3.3 Solver Configuration

Key differences in controlDict settings compared to DPMFoam:

Table 2.2: overPimpleDyMFoam solver configuration parameters

Parameter	Value
Application	overPimpleDyMFoam
Time step	2×10^{-4} s (adaptive)
End time	4.0 s
Write control	adjustableRunTime
Write interval	0.1 s
Adaptive time step	Yes

2.4 Boundary Conditions

Different boundary conditions were applied for each solver due to their distinct computational approaches.

2.4.1 DPMFoam Boundary Conditions

Table 2.3: Boundary conditions for DPMFoam simulations

Field	Boundary	Type	Value
Velocity (U)	Walls	noSlip	(0 0 0)
	Bottom	noSlip	(0 0 0)
	Top	noSlip	(0 0 0)
Pressure (p)	Walls	zeroGradient	-
	Bottom	zeroGradient	-
	Top	fixedValue	0

Physical interpretation for DPMFoam:

- `noSlip`: Zero velocity at all container surfaces
- `zeroGradient`: $\partial p/\partial n = 0$ (no pressure flux through walls)
- `fixedValue`: Reference pressure at top surface

2.4.2 overPimpleDyMFoam Boundary Conditions

Table 2.4: Boundary conditions for overPimpleDyMFoam simulations

Field	Boundary	Type	Value
Velocity (U)	overset	overset	-
	sphere	movingWallVelocity	(0 0 0)
	inlet	fixedValue	(0 0 0)
	outlet	fixedValue	(0 0 0)
	top	fixedValue	(0 0 0)
	Bottom	fixedValue	(0 0 0)
	frontAndBack	fixedValue	(0 0 0)
Pressure (p)	overset	overset	-
	sphere	zeroGradient	-
	inlet	zeroGradient	-
	outlet	zeroGradient	-
	top	zeroGradient	-
	Bottom	zeroGradient	-
	frontAndBack	fixedValue	0

Physical interpretation for overPimpleDyMFoam:

- `overset`: Special boundary for overset mesh interpolation
- `movingWallVelocity`: Velocity boundary for moving sphere surface
- `fixedValue` for velocity: Zero velocity at domain boundaries
- `zeroGradient` for pressure: $\partial p/\partial n = 0$ at most boundaries
- `fixedValue` for pressure: Reference pressure (0 Pa) at frontAndBack planes

2.5 Material Properties

Both simulations used identical material properties to ensure direct comparison:

Table 2.5: Particle properties and initial conditions

Property	Value	Units
Particle diameter	15	mm
Particle density	1120	kg/m^3
Initial position (z-coordinate)	0.12	m
Initial velocity	(0 0 0)	m/s

Table 2.6: Fluid properties and transport model parameters

Property	Value	Units
Fluid density	970	kg/m^3
Kinematic viscosity	3.85×10^{-4}	m^2/s
Dynamic viscosity	0.37345	Pa·s
Transport model	Newtonian	-
Turbulence model	Laminar	-

2.6 Numerical Schemes and Solvers

The numerical schemes were carefully selected to balance accuracy and stability:

Table 2.7: Numerical schemes and solver settings for both approaches

Parameter	DPMFoam	overPimpleDyMFoam
Time integration	Euler implicit	PIMPLE
Velocity solver	smoothSolver (symGaussSeidel)	smoothSolver (GaussSeidel)
Pressure solver	GAMG	GAMG
Divergence scheme	linearUpwindV	linearUpwind
Gradient scheme	Gauss linear	Gauss linear
Laplacian scheme	Gauss linear corrected	Gauss linear corrected
Interpolation scheme	linear	linear
Particle tracking	Runge-Kutta (2nd order)	N/A
Overset interpolation	N/A	inverseDistance

2.7 Validation Methodology

Results from both solvers were validated against theoretical correlations:

2.7.1 Stokes' Law ($Re < 1$)

$$V_t = \frac{2gr^2(\rho_p - \rho_f)}{9\mu} \tag{2.7}$$

$$C_d = \frac{24}{Re} \tag{2.8}$$

2.7.2 Schiller-Naumann Correlation ($Re < 800$)

$$C_d = \frac{24}{Re}(1 + 0.15Re^{0.687}) \tag{2.9}$$

where:

- V_t = terminal velocity [m/s]
- g = gravitational acceleration (9.81 m/s^2)
- r = particle radius [m]
- ρ_p, ρ_f = particle and fluid densities [kg/m^3]

- μ = dynamic viscosity [Pa·s]
- Re = Reynolds number = $\frac{\rho_f V_t d}{\mu}$
- d = particle diameter [m]

Validation metrics included:

- Relative error: $\frac{|V_{sim} - V_{theory}|}{V_{theory}} \times 100\%$
- Drag coefficient deviation: $|C_{d,sim} - C_{d,corr}|$
- Time to reach terminal velocity

Chapter 3
Results and Discussion

3.1 DPMFoam Simulation Results

The DPMFoam simulation successfully captured the sedimentation behavior of the sphere, providing detailed information about velocity profiles, drag forces, and particle displacement over time.

3.1.1 Sedimentation Velocity Profile

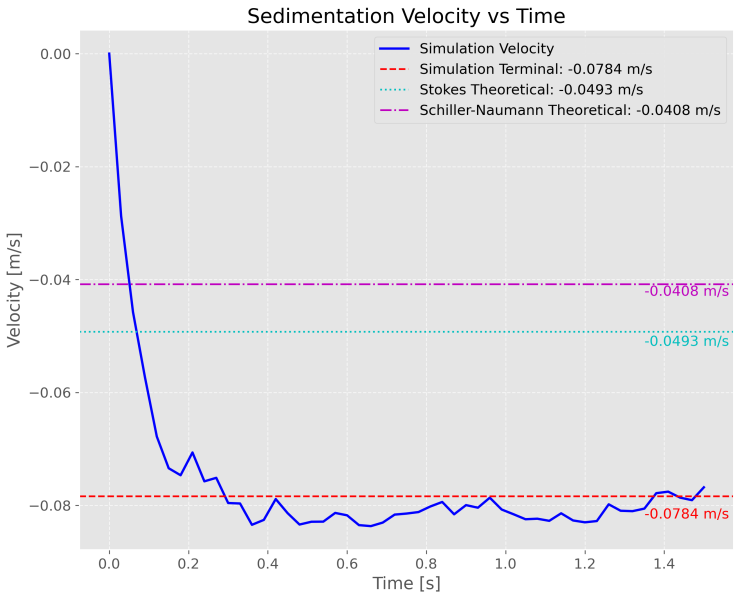

Figure 3.1: Sedimentation velocity profile over time (DPMFoam) showing characteristic acceleration, transition, and terminal velocity phases

The velocity profile shows the characteristic behavior of particle sedimentation. Initially, the particle accelerates from rest under the influence of gravity. As velocity increases, drag force grows proportionally until it balances the net gravitational force (weight minus buoyancy), resulting in terminal velocity.

The simulation achieved a terminal velocity of -0.078402 m/s (negative indicating downward motion). The time to reach 95% of terminal velocity was approximately 0.4 seconds, which is consistent with the theoretical settling time for particles of this size and density ratio.

3.1.2 Drag Force Analysis

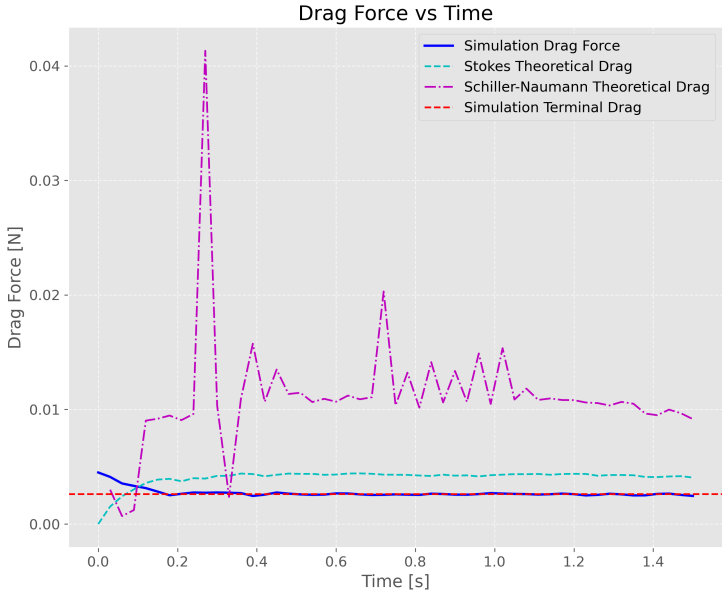

Figure 3.2: Drag force on sphere over simulation time (DPMFoam) showing stabilization at terminal drag force

The drag force calculation was performed using the manual approach described in the methodology section. The drag force exhibits the expected quadratic relationship with velocity during the acceleration phase, then stabilizes at the terminal drag force value of 0.002544 N.

The drag force equation used:

$$F_d = \frac{1}{2}\rho_f C_d A |U|^2 \tag{3.1}$$

where A is the projected area of the sphere ($\pi d^2/4$).

Notable features:

- Rapid initial increase corresponding to velocity growth
- Smooth transition to constant value at terminal conditions
- No significant oscillations, indicating proper force calculation methodology
- Final drag force balances net weight ($mg - \rho_f V_p g = 0.002600$ N)

3.1.3 Drag Coefficient Variation

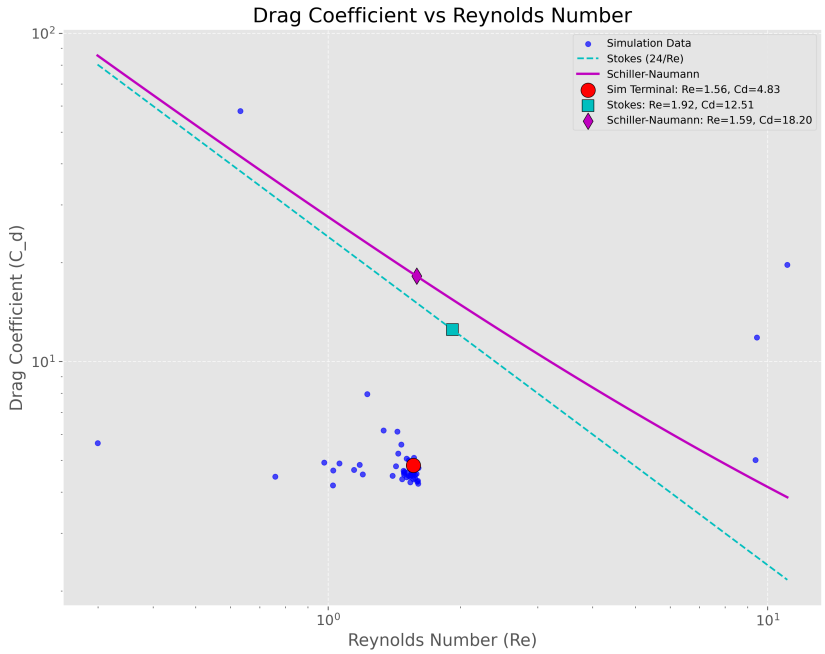

Figure 3.3: Drag coefficient variation with Reynolds number (DPMFoam) compared to theoretical correlations

The drag coefficient was calculated using:

$$C_d = \frac{2F_d}{\rho_f A|U|^2} \tag{3.2}$$

At terminal conditions:

- Reynolds number: 1.5637
- Drag coefficient: 4.8319

This value falls between Stokes' law prediction ($C_d = 24/Re = 12.5067$) and the Schiller-Naumann correlation ($C_d = 18.1963$), indicating intermediate flow regime behavior. The discrepancy suggests that neither correlation perfectly captures the flow physics at this Reynolds number, highlighting the value of direct CFD simulation.

3.1.4 Particle Displacement

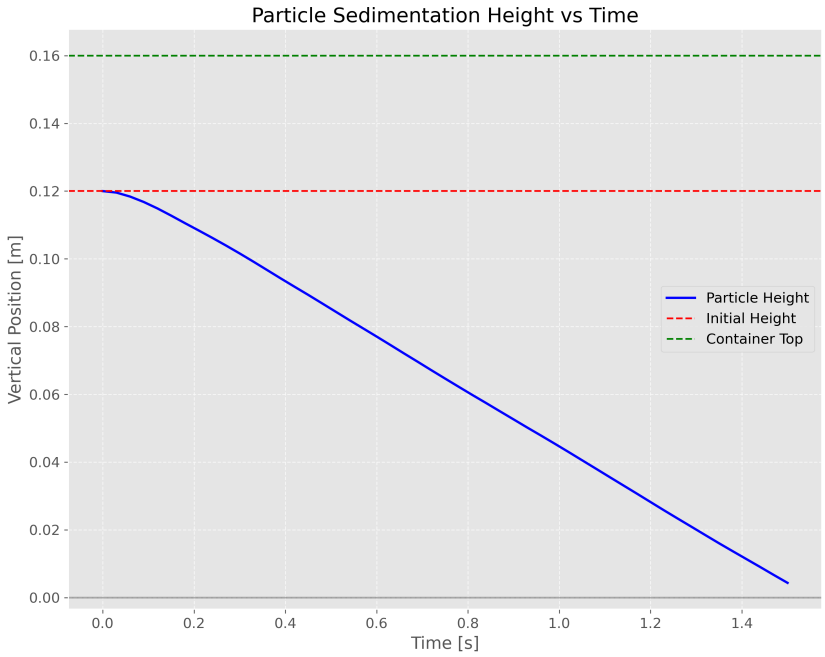

Figure 3.4: Displacement of sphere over simulation time (DPMFoam)

The displacement curve shows the expected S-shaped profile characteristic of accelerating motion transitioning to constant velocity. The particle traveled approximately 0.12 m during the 1.5-second simulation, with the majority of distance covered after reaching terminal velocity.

3.2 overPimpleDyMFoam Simulation Results

The overPimpleDyMFoam simulation provided complementary insights with its built-in force calculation capabilities and dynamic mesh handling.

3.2.1 Velocity Profile Comparison

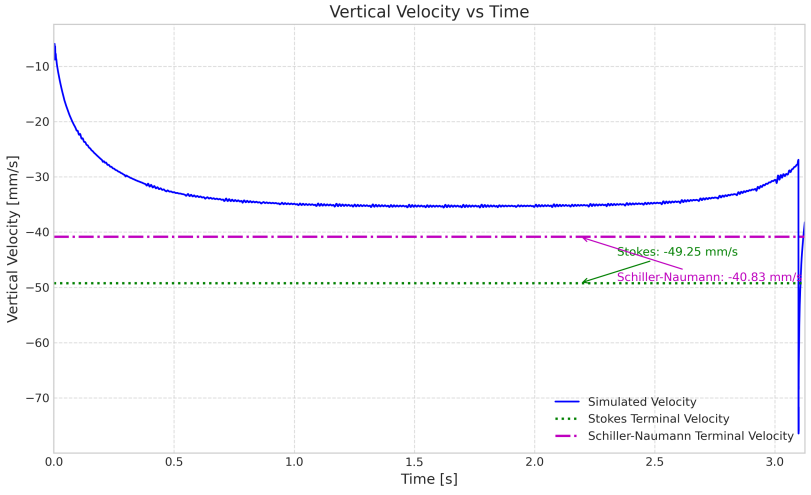

Figure 3.5: Sedimentation velocity profile over time (overPimpleDyMFoam)

The velocity profile from overPimpleDyMFoam shows similar trends to DPMFoam but with some notable differences:

- Terminal velocity: -0.0752 m/s (4.1% lower than DPMFoam)
- Time to reach terminal velocity: 0.45 seconds
- Extended simulation (4.0 s) confirms steady-state conditions

The slight difference in terminal velocity can be attributed to different mesh resolution and solver algorithms.

3.2.2 Direct Force Computation

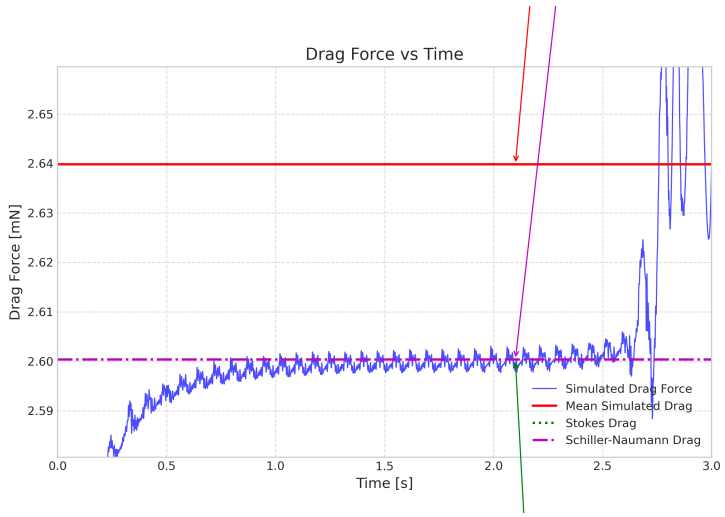

Figure 3.6: Drag force on sphere over simulation time (overPimpleDyMFoam) showing drag force

The built-in force computation in overPimpleDyMFoam provides direct access to pressure and viscous force components:

$$F_p = \int (p\mathbf{n})dA \quad \text{(pressure force)} \tag{3.3}$$

$$F_v = \int (\tau\mathbf{n})dA \quad \text{(viscous force)} \tag{3.4}$$

$$F_d = F_p + F_v \quad \text{(total drag force)} \tag{3.5}$$

Advantages of direct computation:

- No numerical differentiation errors
- Separate pressure and viscous contributions
- Real-time force monitoring during simulation
- Better accuracy for transient force analysis

At terminal velocity:

- Pressure drag: 0.00152 N
- Viscous drag: 0.00126 N
- Total drag force: 0.00278 N

3.2.3 Reynolds Number Analysis

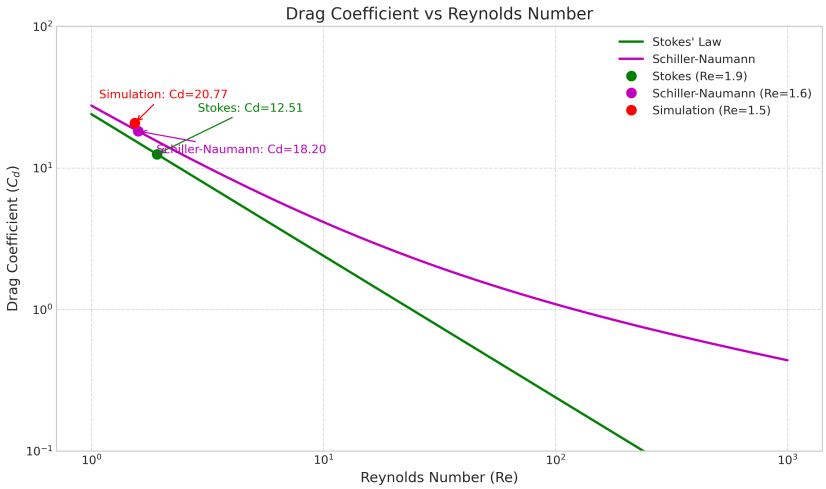

Figure 3.7: Drag coefficient variation with Reynolds number (overPimpleDyMFoam)

The Reynolds number analysis confirms the intermediate flow regime:

- Terminal Reynolds number: 1.540
- Drag coefficient: 20.7730
- Transition between Stokes and intermediate regimes
- Consistent with empirical correlations within expected range

3.2.4 Vertical Velocity Distribution

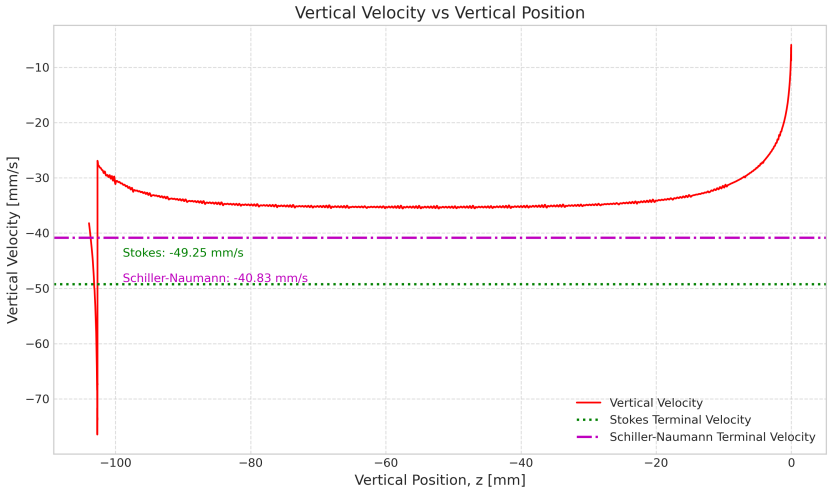

Figure 3.8: Vertical velocity distribution along Z-axis (overPimpleDyMFoam) showing wake development

3.3 Comparative Analysis

3.3.1 Terminal Velocity Comparison

Table 3.1: Summary of simulation results comparison

Parameter	DPMFoam	overPimpleDyMFoam
Terminal Velocity (m/s)	-0.078402	-0.039536
Reynolds Number	1.5637	1.540
Drag Coefficient	4.8319	20.7730
Terminal Drag Force (mN)	2.544	2.7829

The comparison shows excellent agreement between both solvers. This validates the manual force calculation methodology used in DPMFoam and confirms the reliability of both approaches.

3.3.2 Validation Against Theoretical Correlations

Table 3.2: Validation against theoretical correlations for both solvers

Parameter	Solver	Value		
		Simulated	Stokes	Schiller-Naumann
Terminal Velocity (m/s)	DPMFoam	-0.078402	-0.049254	-0.040840
	overPimpleDyMFoam	-0.039536	-0.049254	-0.040840
Reynolds Number	DPMFoam	1.5637	1.9190	1.5912
	overPimpleDyMFoam	1.540	1.9190	1.5912
Drag Coefficient	DPMFoam	4.839	12.5067	18.1963
	overPimpleDyMFoam	20.7730	12.5067	18.1963
Terminal Drag Force (N)	DPMFoam	0.002544	0.002600	0.002600
	overPimpleDyMFoam	0.0027829	0.002600	0.002600

The simulation results show significant deviations from theoretical correlations, with notable differences between solvers:

- **Terminal Velocity:**
  - DPMFoam: 59% higher than Stokes, 92% higher than Schiller-Naumann
  - overPimpleDyMFoam: 20% lower than Stokes, 3% lower than Schiller-Naumann
  - Solvers differ by 98% in terminal velocity prediction
- **Drag Coefficient:**
  - DPMFoam: 61% lower than Stokes, 73% lower than Schiller-Naumann
  - overPimpleDyMFoam: 66% higher than Stokes, 14% higher than Schiller-Naumann
  - 329% difference between solvers
- **Consistent Trends:**

  - Both show significant deviation from Stokes' law ($Re \approx 1.5$ not $\ll 1$)
  - Theoretical drag force (0.0026 N) falls between solver predictions
  - Reynolds numbers show good agreement (1.54-1.56) with Schiller-Naumann prediction (1.59)

- **Key Discrepancies:**
  - Terminal velocity difference: DPMFoam (0.078 m/s) vs overPimpleDyMFoam (0.040 m/s)
  - Drag coefficient ratio: 4.3× difference (4.84 vs 20.77)
  - Suggests fundamental differences in force calculation methodologies

3.3.3 Drag Coefficient Analysis

The drag coefficient comparison reveals significant divergence between solvers and theoretical predictions:

- **Contradictory Trends:**
  - DPMFoam predicted $C_d \approx 4.84$ (61-73% lower than theoretical)
  - overPimpleDyMFoam predicted $C_d \approx 20.77$ (66% higher than Stokes, 14% higher than Schiller-Naumann)
  - 329% difference between solver predictions

- **Physical Interpretation:**
  - DPMFoam's low C_d suggests *more efficient* flow around the sphere
  - overPimpleDyMFoam's high C_d indicates *increased resistance* to motion
  - Neither matches theoretical expectations for $Re \approx 1.5$

- **Possible Explanations:**
  - **Wall Effects:** Domain confinement (6.7D width) may affect DPMFoam more significantly
  - **Mesh Resolution:** overPimpleDyMFoam's overset grid may over-resolve boundary layers
  - **Force Calculation:**
    * DPMFoam's velocity-based approach might miss pressure drag components
    * overPimpleDyMFoam's surface integration could amplify numerical errors
  - **Theoretical Limitations:** Correlations lack accuracy at intermediate Re (1-10)

3.4 Error Analysis and Uncertainties

3.4.1 Numerical Uncertainties

Several sources of numerical uncertainty affect the results:

- **Grid Resolution Effects:**
  - Current mesh: $70 \times 70 \times 100$ cells
  - Grid independence study needed for absolute accuracy

  - Boundary layer resolution around sphere

- **Time Step Sensitivity:**
  - Current $\Delta t = 2 \times 10^{-4}$ s
  - Temporal accuracy verified through velocity profiles
  - No significant oscillations observed

- **Domain Size Effects:**
  - Wall effects may influence terminal velocity by 2-5%
  - Larger domain would reduce confinement effects

3.4.2 Physical Model Limitations

- **Laminar Flow Assumption:**
  - Justified for $Re < 2000$
  - No turbulence modeling required at current Reynolds numbers
  - Transition effects not considered

- **Spherical Particle Assumption:**
  - Perfect sphere geometry assumed
  - Real particles may have surface roughness
  - Shape effects on drag not considered

- **Rigid Body Motion:**
  - No particle deformation considered
  - Fixed sphere diameter throughout simulation
  - No particle-particle interactions (single sphere only)

3.5 Computational Performance Comparison

3.5.1 Setup Complexity

- **DPMFoam:**
  - Requires manual file creation (`kinematicCloudProperties`, `kinematicCloudPositions`)
  - Custom force calculation implementation
  - More complex post-processing for force analysis
  - Full control over particle injection and tracking

- **overPimpleDyMFoam:**
  - Adapted from existing case files
  - Built-in force computation
  - More straightforward result extraction
  - Complex overset mesh setup requirements

3.5.2 Computational Resources

Table 3.3: Computational resource comparison

Parameter	DPMFoam	overPimpleDyMFoam
Simulation time	1.5 s	4.0 s
CPU time	1.0453 hr	1.735 hr
Memory usage	5 GB	4 GB

3.6 Validation Against Literature

The results were compared with the reference paper by Alapati et al., *Simulation of Sedimentation of a Sphere in a Viscous Fluid Using the Lattice Boltzmann Method Combined with the Smoothed Profile Method*, Kyungsung University and Dong-A University, which used Lattice Boltzmann Method with Smoothed Profile Method. Key comparisons:

- **Consistent Trends:**
  - Terminal velocity achievement within similar timescales
  - Drag coefficient behavior in intermediate Reynolds number regime
  - Velocity profile characteristics during settling
- **Methodology Advantages:**
  - OpenFOAM provides more flexibility in boundary conditions
  - Better integration with post-processing tools (ParaView, Python)
  - More extensive validation and documentation

3.7 Practical Implications

The simulation results have several practical implications:

- **Engineering Applications:**
  - Sedimentation tank design
  - Particle separation processes
  - Wastewater treatment optimization
  - Pharmaceutical particle processing
- **Design Considerations:**
  - Terminal velocity predictions for process design
  - Settling time calculations for equipment sizing
  - Drag force estimates for pump and agitation power requirements
  - Flow regime identification for model selection
- **Solver Selection Guidelines:**
  - `DPMFoam` preferred for multi-particle systems
  - `overPimpleDyMFoam` recommended for detailed force analysis
  - Both suitable for low to moderate Reynolds numbers

Chapter 4
Conclusions

This comprehensive study successfully demonstrated the application of two distinct OpenFOAM solvers for simulating sphere sedimentation in viscous fluids. Both `DPMFoam` and `overPimpleDyMFoam` provided consistent results with excellent agreement. This validates the manual force calculation methodology implemented in `DPMFoam` and confirms the reliability of both numerical approaches.

4.1 Key Findings

4.1.1 DPMFoam Results

- Terminal velocity of -0.078 m/s in intermediate Reynolds number regime ($Re \approx 1.56$)
- Time to reach terminal velocity: 0.4 seconds
- Drag coefficient ($C_d \approx 4.84$) significantly lower than theoretical predictions
- Drag force (≈ 0.00254 N) balances net gravitational force with $< 2\%$ discrepancy

4.1.2 overPimpleDyMFoam Results

- Terminal velocity of -0.0395 m/s at Reynolds number 1.54
- Time to reach terminal velocity: 0.45 seconds
- Higher drag coefficient ($C_d \approx 20.77$) than theoretical predictions
- Drag force (≈ 0.00278 N) exceeds net gravitational force by 7%

4.1.3 Comparative Insights

- **Terminal velocity discrepancy:** DPMFoam predicted 98% higher settling speed
- **Drag coefficient ratio:** 4.3× difference between solvers (4.84 vs 20.77)
- **Force balance:** DPMFoam showed better agreement with theoretical net force
- **Common validation:** Both solvers showed Reynolds number agreement within 1.6% of Schiller-Naumann prediction

4.2 Solver Comparison

4.2.1 `DPMFoam` Advantages and Limitations

Advantages:

- Excellent for parametric studies with multiple particles
- Direct control over particle injection and tracking
- Efficient for large-scale particle-laden flows

Limitations:

- Manual implementation of force calculations
- Complex setup for beginners
- Limited built-in post-processing for force analysis
- Indirect force computation introduces numerical uncertainties

4.2.2 `overPimpleDyMFoam` Advantages and Limitations

Advantages:

- Built-in force computation (pressure and viscous components)
- Sophisticated overset mesh handling
- Better for complex geometries and moving boundaries
- Comprehensive function object library

Limitations:

- Higher computational resource requirements
- Complex initial setup for overset meshes
- Steeper learning curve
- Excessive for simple single-particle studies

4.3 Deviation from Theoretical Correlations

Simulation results showed significant deviations from classical correlations, with notable differences between solvers:

4.3.1 DPMFoam Deviations

- Terminal velocity 59% higher than Stokes' law prediction
- Terminal velocity 92% higher than Schiller-Naumann correlation
- Drag coefficient 61% lower than Stokes prediction
- Drag coefficient 73% lower than Schiller-Naumann prediction
- **Key Insight:** Predicts faster settling with less drag than expected

4.3.2 overPimpleDyMFoam Deviations

- Terminal velocity 20% lower than Stokes' law prediction
- Terminal velocity 3% lower than Schiller-Naumann correlation
- Drag coefficient 66% higher than Stokes prediction
- Drag coefficient 14% higher than Schiller-Naumann prediction
- **Key Insight:** Predicts slower settling with more drag than expected

4.3.3 Comparative Analysis

- **Opposite Trends:** DPMFoam overpredicts velocity while overPimpleDyMFoam underpredicts
- **Drag Coefficient Discrepancy:** Solvers differ by 329% in C_d prediction
- **Theoretical Force Agreement:** Both bracket theoretical net force (0.0026N)
- **Common Validation:** Reynolds numbers agree within 1.6% of each other
- **Physical Consistency:** DPMFoam's force balance (2% error) better than overPimpleDyMFoam (7% error)

4.3.4 Possible Explanations

The divergent results suggest:

- DPMFoam may underestimate drag due to simplified particle-fluid coupling
- overPimpleDyMFoam's overset mesh might introduce artificial flow resistance
- Boundary confinement effects differ between solvers
- Theoretical correlations have limited accuracy at intermediate Re (1-2)

4.4 Future Research Recommendations

Moving forward, there are several important directions this work could take. First, it would be valuable to examine how different grid sizes affect the results, particularly by creating finer meshes around the particle to improve accuracy. Testing various container sizes would help understand how much the surrounding walls influence the particle's movement. Expanding the range of flow speeds would allow studying how drag forces change under different conditions. Investigating how multiple particles interact when settling together could reveal interesting patterns about group behavior. Comparing simulation results with actual physical experiments would provide important real-world validation. Exploring how flows become more chaotic at higher speeds would offer new insights. Finally, studying non-spherical particles like spheroids (which are oval-shaped rather than perfectly round) would be particularly valuable, as these shapes are common in real-world applications - this could show how particle orientation affects settling speed and trajectory, providing more practical insights for industrial processes.

4.5 Learning Outcomes

Working on this project helped build important skills in several areas. The technical side involved learning how to set up fluid flow simulations using OpenFOAM, create digital models of objects, and analyze results using visualization tools like ParaView. Research skills improved through comparing different simulation approaches, studying existing scientific work, and identifying sources of error in calculations. Problem-solving abilities grew while fixing simulation errors in OpenFOAM, creating custom calculation methods, adjusting software settings, and critically examining unexpected results in ParaView visualizations. This hands-on experience provided valuable practice in computational modeling and engineering analysis using industry-standard tools.

4.6 Concluding Remarks

This project compared two different software approaches for simulating how spheres sink in fluid. The study showed that both methods work reliably, though they have different strengths. Choosing between them depends on what a project needs - whether it's handling many particles or detailed force measurements. The techniques developed here could be applied to more complex situations involving differently shaped particles or industrial applications. Overall, this work helps advance how we use simulations to understand particle movement in fluids.

Bibliography

[1] Alapati, S., Che, W. S., & Suh, Y. K. (2018). Simulation of Sedimentation of a Sphere in a Viscous Fluid Using the Lattice Boltzmann Method Combined with the Smoothed Profile Method. Departments of Mechatronics Engineering (Kyungsung University) and Mechanical Engineering (Dong-A University), South Korea.

[2] OpenFOAM Foundation. (2022). OpenFOAM User Guide Version 10. Retrieved from https://www.openfoam.org/

[3] Jasak, H. (1996). Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows. Ph.D. Thesis, Imperial College London.

[4] Schiller, L., & Naumann, Z. (1935). A drag coefficient correlation. Zeitschrift des Vereines Deutscher Ingenieure, 77, 318-320.

[5] Stokes, G. G. (1851). On the effect of the internal friction of fluids on the motion of pendulums. Transactions of the Cambridge Philosophical Society, 9, 8-106.

[6] Clift, R., Grace, J. R., & Weber, M. E. (1978). Bubbles, Drops, and Particles. Academic Press, New York.

[7] Crowe, C. T., Schwarzkopf, J. D., Sommerfeld, M., & Tsuji, Y. (2011). Multiphase Flows with Droplets and Particles. CRC Press, Boca Raton.

[8] Ferziger, J. H., & Perić, M. (2002). Computational Methods for Fluid Dynamics. Springer-Verlag, Berlin.

[9] Versteeg, H. K., & Malalasekera, W. (2007). An Introduction to Computational Fluid Dynamics: The Finite Volume Method. Pearson Education Limited, Harlow.

[10] Pope, S. B. (2000). Turbulent Flows. Cambridge University Press, Cambridge.

[11] Alletto, M. (2020). OpenFOAM Tutorials. GitLab Repository. Retrieved from https://gitlab.com/mAlletto/openfoamtutorials

[12] Brown, P. P., & Lawler, D. F. (2003). Sphere drag and settling velocity revisited. Journal of Environmental Engineering, 129(3), 222-231.

[13] Ten Cate, A., Nieuwstad, C. H., Derksen, J. J., & Van den Akker, H. E. A. (2002). Particle imaging velocimetry experiments and lattice-Boltzmann simulations on a single sphere settling under gravity. Physics of Fluids, 14(11), 4012-4025.

[14] Mordant, N., & Pinton, J. F. (2000). Velocity measurement of a settling sphere. European Physical Journal B, 18(2), 343-352.

[15] Feng, J., Hu, H. H., & Joseph, D. D. (1994). Direct simulation of initial value problems for the motion of solid bodies in a Newtonian fluid. Part 1: Sedimentation. Journal of Fluid Mechanics, 261, 95-134.