



Swaroop Dhavale

Implementation of Direct Numerical Simulation Framework  
in the Finite Volume Methods-Based Open-Source Code  
Athena++ for Astrophysical Flows

School of Aerospace, Transport and Manufacturing  
Computational Fluid Dynamics

MSc  
Academic Year: 2024–2025

Supervisors: Dr László Könözy and Dr Tamás Józsa  
August 2025



School of Aerospace, Transport and Manufacturing  
Computational Fluid Dynamics

MSc

Academic Year: 2024–2025

Swaroop Dhavale

Implementation of Direct Numerical Simulation Framework  
in the Finite Volume Methods-Based Open-Source Code  
Athena++ for Astrophysical Flows

Supervisors: Dr László Könözy and Dr Tamás Józsa  
August 2025

This thesis is submitted in partial fulfilment of the  
requirements for the degree of MSc.

© Cranfield University 2025. All rights reserved. No part of  
this publication may be reproduced without the written per-  
mission of the copyright owner.

## **Academic Integrity Declaration**

I declare that:

- the thesis submitted has been written by me alone.
- the thesis submitted has not been previously submitted to this university or any other.
- that all content, including primary and/or secondary data, is true to the best of my knowledge.
- that all quotations and references have been duly acknowledged according to the requirements of academic research.

I understand that to knowingly submit work in violation of the above statement will be considered by examiners as academic misconduct.

# Abstract

This thesis investigates the performance of the open-source code Athena++ for the simulation of viscous incompressible flows across laminar and turbulent regimes, with particular emphasis on adaptive mesh refinement (AMR). Benchmark test cases, including steady-state channel flow, the lid-driven cavity flow, and the Taylor–Green vortex (TGV) in both two and three dimensions, were employed to assess the solver’s accuracy, stability, and computational efficiency. Spatial and temporal discretization schemes were systematically evaluated through grid convergence studies, supported by Richardson extrapolation and Grid Convergence Index (GCI) analysis, to verify the numerical implementation and establish the expected order of accuracy.

An AMR strategy based on a local grid Reynolds number ( $Re_{\text{grid}}$ ) was implemented and tested in both laminar and turbulent TGV flows. For two-dimensional cases, computational savings of up to 44% were achieved, while in three dimensions with moderate Reynolds numbers, the savings were up to 35%. The AMR solutions successfully reproduced global flow characteristics such as kinetic energy decay and dissipation rate, but also highlighted the need for higher-order reconstruction and time integration schemes, as well as the limitations of a single refinement criterion.

Finally, under-resolved direct numerical simulations (DNS) of the three-dimensional TGV at  $Re = 1600$  demonstrated that Athena++, equipped with high-order reconstruction (PPM) and time integration (RK4), can capture turbulence dynamics in close agreement with benchmark solutions when using grid resolutions consistent with Kolmogorov theory. However, AMR revealed an unphysical increase in the spectral energy density at dissipative scales, possibly due to numerically generated instabilities introduced as a result of mesh-size variations. The findings confirm that Athena++ provides accurate and robust solutions for viscous flows and highlight the potential of AMR to reduce computational cost in under-resolved DNS, provided that the mesh refinement and de-refinement criteria are guided by the physical scales of turbulence.

**Keywords:** Athena++, viscous incompressible flows, Grid Convergence Index (GCI), adaptive mesh refinement (AMR), computational efficiency, Taylor–Green vortex (TGV), direct numerical simulation (DNS)



# Acknowledgements

I wish to express my deepest gratitude to my supervisor, Dr. László Könözy, for his structured and thoughtful guidance throughout this research. His clear direction in defining objectives and formulating strategies to achieve them not only shaped the course of this work but also instilled in me the confidence to address challenges effectively and pursue solutions with determination.

My sincere thanks are also extended to my co-supervisor, Dr. Tamas Jozsa, for providing valuable foundational knowledge during earlier projects and module assignments, particularly with regard to open-source codes and development tools. His insights and perspectives significantly enhanced my technical proficiency and deepened my understanding, which proved to be crucial in carrying out this research successfully.

I am especially indebted to Dr. Nicolas Ledos from University Milano-Bicocca, Italy for his continuous technical support and guidance throughout the course of this work. His expertise with the open-source code Athena++, coupled with his scientific advice and consistent encouragement, was instrumental in overcoming technical challenges and in accomplishing the objectives of this project.

I would also like to acknowledge Dr. Shinsuke Takasao from University of Osaka, Japan for his insightful analysis and critical evaluation of this work. His constructive ideas, meaningful discussions, and valuable feedback provided important perspectives that enriched the outcomes of this research and strengthened its overall contribution.

Finally, I wish to thank my family and friends for their constant encouragement and support, which has been of great value during the course of this work.

# Table of Contents

|                                                                    |            |
|--------------------------------------------------------------------|------------|
| <b>Academic Integrity Declaration</b>                              | <b>i</b>   |
| <b>Abstract</b>                                                    | <b>ii</b>  |
| <b>Acknowledgements</b>                                            | <b>iii</b> |
| <b>Table of Contents</b>                                           | <b>iv</b>  |
| <b>List of Figures</b>                                             | <b>vii</b> |
| <b>List of Tables</b>                                              | <b>ix</b>  |
| <b>List of Abbreviations</b>                                       | <b>x</b>   |
| <b>1 Introduction</b>                                              | <b>1</b>   |
| 1.1 Background . . . . .                                           | 1          |
| 1.2 Motivation . . . . .                                           | 3          |
| 1.3 Objectives of the Study . . . . .                              | 3          |
| <b>2 Literature Review</b>                                         | <b>4</b>   |
| <b>3 Theoretical Background and Numerical Methodology</b>          | <b>8</b>   |
| 3.1 Hydrodynamic and Magnetohydrodynamic Flows . . . . .           | 8          |
| 3.1.1 Characteristics of Turbulence . . . . .                      | 9          |
| 3.1.1.1 Irregularity and Statistical Nature . . . . .              | 9          |
| 3.1.1.2 Coherent Structures and Energy Dissipation . . . . .       | 10         |
| 3.1.1.3 Enhanced Transport and Mixing . . . . .                    | 10         |
| 3.1.1.4 Complexity and Modeling Challenges . . . . .               | 11         |
| 3.1.2 Scales of Turbulence . . . . .                               | 11         |
| 3.1.2.1 Richardson’s Hypothesis . . . . .                          | 12         |
| 3.1.2.2 Kolmogorov’s Hypotheses . . . . .                          | 12         |
| 3.1.2.3 Energy Spectrum . . . . .                                  | 13         |
| 3.1.3 Numerical Approaches for Turbulence Simulation . . . . .     | 14         |
| 3.1.4 Computational Requirements for DNS . . . . .                 | 15         |
| 3.2 Athena++: An Overview . . . . .                                | 17         |
| 3.3 Governing Equations implemented in Athena++ . . . . .          | 17         |
| 3.3.1 Governing Equations for Viscous Hydrodynamic Flows . . . . . | 18         |

|          |                                                                                                   |           |
|----------|---------------------------------------------------------------------------------------------------|-----------|
| 3.3.2    | Compact Form of Governing Equations for Viscous Hydrodynamic Flows . . . . .                      | 19        |
| 3.4      | Numerical Procedure used in Athena++ . . . . .                                                    | 21        |
| 3.4.1    | Finite Volume Discretization . . . . .                                                            | 21        |
| 3.4.2    | Godunov Method and Approximate Riemann Solvers . . . . .                                          | 23        |
| 3.4.3    | Spatial Reconstruction – PLM and PPM . . . . .                                                    | 24        |
| 3.4.3.1  | Piecewise Linear Method (PLM) . . . . .                                                           | 24        |
| 3.4.3.2  | Piecewise Parabolic Method (PPM) . . . . .                                                        | 25        |
| 3.4.4    | Time Integration – Runge–Kutta Methods . . . . .                                                  | 26        |
| 3.4.5    | Adaptive Mesh Refinement . . . . .                                                                | 27        |
| 3.4.5.1  | Flux Correction at Refinement Boundaries . . . . .                                                | 28        |
| 3.4.5.2  | Time Stepping with AMR . . . . .                                                                  | 29        |
| <b>4</b> | <b>Results and Discussion</b>                                                                     | <b>30</b> |
| 4.1      | Two-Dimensional Channel Flow . . . . .                                                            | 32        |
| 4.1.1    | Flow Domain, Fluid Properties and Reynolds Number . . . . .                                       | 32        |
| 4.1.2    | Boundary Conditions and Reference Pressure . . . . .                                              | 33        |
| 4.1.3    | Implementation of Boundary Conditions in Athena++ . . . . .                                       | 33        |
| 4.1.4    | Grid Convergence and Error Analysis . . . . .                                                     | 33        |
| 4.1.5    | Numerical Solution from the Finest Grid . . . . .                                                 | 35        |
| 4.2      | Two-Dimensional Lid Driven Cavity . . . . .                                                       | 36        |
| 4.2.1    | Flow Domain, Fluid Properties and Reynolds Number . . . . .                                       | 36        |
| 4.2.2    | Boundary Conditions and Reference Pressure . . . . .                                              | 37        |
| 4.2.3    | Implementation of Boundary Conditions in Athena++ . . . . .                                       | 37        |
| 4.2.4    | Numerical Solution . . . . .                                                                      | 37        |
| 4.2.4.1  | Velocity Profiles at the Midlines of Cavity . . . . .                                             | 37        |
| 4.2.4.2  | Velocity Contours with Streamlines . . . . .                                                      | 38        |
| 4.2.4.3  | Vorticity Contours and Vorticity at Moving Wall . . . . .                                         | 38        |
| 4.3      | Two-Dimensional Taylor-Green Vortex . . . . .                                                     | 40        |
| 4.3.1    | Flow Domain, Fluid Properties and Reynolds Number . . . . .                                       | 40        |
| 4.3.2    | Boundary Conditions and Reference Pressure . . . . .                                              | 41        |
| 4.3.3    | Grid Convergence and Error Analysis . . . . .                                                     | 42        |
| 4.3.4    | Velocity Field in 2D Taylor-Green Vortex Flow . . . . .                                           | 42        |
| 4.3.5    | Adaptive Mesh Refinement for the 2D Taylor-Green Vortex . . . . .                                 | 43        |
| 4.3.5.1  | The AMR Criteria and Parameters . . . . .                                                         | 43        |
| 4.3.5.2  | Analysis of AMR Process . . . . .                                                                 | 44        |
| 4.3.5.3  | Analysis of AMR-based Solutions . . . . .                                                         | 46        |
| 4.3.5.4  | AMR with Higher Order Schemes . . . . .                                                           | 46        |
| 4.3.6    | Computational Time required for Simulations . . . . .                                             | 47        |
| 4.4      | Three-Dimensional Taylor-Green Vortex . . . . .                                                   | 49        |
| 4.4.1    | Three-Dimensional Taylor-Green Vortex: Laminar Flow . . . . .                                     | 50        |
| 4.4.1.1  | Simulation of 3D TGV Flow at $Re = 100$ : . . . . .                                               | 50        |
| 4.4.1.2  | Simulation of 3D TGV Flow at $Re = 400$ : . . . . .                                               | 57        |
| 4.4.2    | Three-Dimensional Taylor-Green Vortex: Turbulent Flow . . . . .                                   | 59        |
| 4.4.2.1  | Comparison of Under-resolved DNS Solutions for $Re = 1600$ using Uniform and AMR Meshes . . . . . | 59        |

|          |                                                                                                   |           |
|----------|---------------------------------------------------------------------------------------------------|-----------|
| 4.4.2.2  | Comparison of Under-resolved DNS Solutions for $Re = 3000$ using Uniform and AMR Meshes . . . . . | 64        |
| <b>5</b> | <b>Conclusions</b>                                                                                | <b>67</b> |
| <b>6</b> | <b>Future Work</b>                                                                                | <b>69</b> |
|          | <b>References</b>                                                                                 | <b>70</b> |
| <b>A</b> | <b>Ethical Approval Letter</b>                                                                    | <b>73</b> |

# List of Figures

|      |                                                                                                                                  |    |
|------|----------------------------------------------------------------------------------------------------------------------------------|----|
| 1.1  | Two-dimensional slice of current density and mass density from a super-sonic MHD turbulence DNS . . . . .                        | 2  |
| 3.1  | Turbulent flows in engineering applications and in natural phenomena . .                                                         | 9  |
| 3.2  | Instantaneous velocity obtained in two different experimental measure-ments under identical conditions . . . . .                 | 10 |
| 3.3  | Kinetic energy transport from large scale structures to smaller scales . . .                                                     | 10 |
| 3.4  | Typical energy spectrum of turbulent flows . . . . .                                                                             | 13 |
| 3.5  | Volume-averaged conserved variables $\mathbf{U}$ and time- and area-averaged com-ponents of the fluxes of $\mathbf{U}$ . . . . . | 22 |
| 3.6  | Schematic representation of cell-averaged states and discontinuities at cell interfaces . . . . .                                | 23 |
| 3.7  | Schematic representation of a possible wave structure evolved from an initial discontinuity after a finite time . . . . .        | 24 |
| 3.8  | Piecewise linear reconstruction of states at cell interfaces . . . . .                                                           | 25 |
| 3.9  | Adaptive mesh refinement methods . . . . .                                                                                       | 28 |
| 3.10 | Flux correction at coarse-fine interfaces in AMR . . . . .                                                                       | 29 |
| 4.1  | Comparison of numerical results with analytical solution . . . . .                                                               | 34 |
| 4.2  | Absolute error at $y = 0$ vs grid resolution . . . . .                                                                           | 35 |
| 4.3  | Contours of streamwise velocity colored by magnitude . . . . .                                                                   | 35 |
| 4.4  | Schematic of lid driven cavity with velocity boundary conditions . . . . .                                                       | 36 |
| 4.5  | Velocity profiles in lid-driven cavity at $Re = 400$ . . . . .                                                                   | 38 |
| 4.6  | Streamlines with velocity magnitude and vorticity contours at $Re = 400$ .                                                       | 39 |
| 4.7  | Vorticity at the lid of cavity at $Re = 400$ . . . . .                                                                           | 39 |
| 4.8  | Mean kinetic energy and $L_2$ error at different time instances . . . . .                                                        | 42 |
| 4.9  | Velocity magnitude contours at different time instances . . . . .                                                                | 43 |
| 4.10 | Root grid $64 \times 64$ and its MeshBlocks . . . . .                                                                            | 45 |
| 4.11 | MeshBlocks resulting from adaptive mesh refinement and de-refinement .                                                           | 45 |
| 4.12 | Comparison of mean kinetic energy evolution . . . . .                                                                            | 46 |
| 4.13 | Comparison of mean kinetic energy evolution . . . . .                                                                            | 47 |
| 4.14 | Grid convergence of 3D TGV simulations for $Re = 100$ . . . . .                                                                  | 51 |
| 4.15 | Absolute errors in dissipation rate of 3D TGV for $Re = 100$ . . . . .                                                           | 51 |
| 4.16 | Velocity magnitude on $z$ -plane at different time instances for $Re = 100$ . .                                                  | 52 |
| 4.17 | Iso-surfaces of vorticity at time $t = 0$ . . . . .                                                                              | 53 |
| 4.18 | Iso-surfaces of vorticity at different time instances for $Re = 100$ . . . . .                                                   | 53 |
| 4.19 | Comparison of solutions from uniform mesh and AMR for $Re = 100$ . . .                                                           | 54 |

|      |                                                                                                                          |    |
|------|--------------------------------------------------------------------------------------------------------------------------|----|
| 4.20 | Root mesh used for AMR . . . . .                                                                                         | 55 |
| 4.21 | Mesh resulting from adaptive refinement and de-refinement . . . . .                                                      | 56 |
| 4.22 | Mesh resulting from adaptive refinement and de-refinement . . . . .                                                      | 56 |
| 4.23 | Change in time step sizes with flow time . . . . .                                                                       | 57 |
| 4.24 | Mean kinetic energy and kinetic energy dissipation rate at $Re = 400$ . . .                                              | 58 |
| 4.25 | Velocity magnitude on $z$ -plane at $t = 15$ for $Re = 1600$ and the MeshBlocks                                          | 60 |
| 4.26 | Mean kinetic energy and kinetic energy dissipation rate at different time<br>instances for flow at $Re = 1600$ . . . . . | 61 |
| 4.27 | Iso-surfaces of vorticity at time $t = 0$ for $Re = 1600$ . . . . .                                                      | 61 |
| 4.28 | Iso-surfaces of vorticity at different time instances for $Re = 1600$ . . . . .                                          | 62 |
| 4.29 | Energy spectra at $t = 9$ and $t = 15$ for $Re = 1600$ . . . . .                                                         | 63 |
| 4.30 | Change in timestep size of integration with flow time . . . . .                                                          | 64 |
| 4.31 | Mean kinetic energy and kinetic energy dissipation rate at different time<br>instances for flow at $Re = 3000$ . . . . . | 65 |
| 4.32 | Energy spectra at $t = 9$ and $t = 15$ for $Re = 3000$ . . . . .                                                         | 66 |

# List of Tables

|      |                                                                           |    |
|------|---------------------------------------------------------------------------|----|
| 4.1  | Numerical settings used in Athena++ simulations . . . . .                 | 30 |
| 4.2  | Fluid properties considered for 2D channel flow . . . . .                 | 32 |
| 4.3  | Boundary conditions and reference pressure used in the simulations . . .  | 33 |
| 4.4  | Implementation of primitive variable boundary conditions in ghost cells . | 34 |
| 4.5  | Boundary conditions and reference pressure used for LDC simulations . .   | 37 |
| 4.6  | Implementation of primitive variable boundary conditions in ghost cells . | 37 |
| 4.7  | Fluid properties considered for 2D Taylor-Green vortex flow . . . . .     | 41 |
| 4.8  | Boundary conditions and reference pressure used for 2D TGV simulations    | 41 |
| 4.9  | AMR simulation configurations . . . . .                                   | 44 |
| 4.10 | Key AMR configuration parameters in Athena++ . . . . .                    | 44 |
| 4.11 | Higher order schemes used with AMR simulation . . . . .                   | 47 |
| 4.12 | Comparison of Computational Times . . . . .                               | 48 |
| 4.13 | AMR simulation configurations used for $Re = 100$ . . . . .               | 54 |
| 4.14 | Computational times required for 3D TGV $Re = 100$ simulations . . . . .  | 57 |
| 4.15 | Computational times required for 3D TGV $Re = 400$ simulations . . . . .  | 59 |
| 4.16 | Computational times required for 3D TGV $Re = 1600$ simulations . . . .   | 64 |
| 4.17 | Computational times required for 3D TGV $Re = 3000$ simulations . . . .   | 66 |

# List of Abbreviations

## Acronyms

|      |                                                   |
|------|---------------------------------------------------|
| AMR  | Adaptive Mesh Refinement                          |
| BC   | Boundary Condition                                |
| CFD  | Computational Fluid Dynamics                      |
| CFL  | Courant-Friedrichs-Lewy number                    |
| DNS  | Direct Numerical Simulation                       |
| FEM  | Finite Element Method                             |
| FVM  | Finite Volume Method                              |
| GCI  | Grid Convergence Index                            |
| HLL  | Harten–Lax–van Leer                               |
| HLLC | Harten–Lax–van Leer–Contact                       |
| HPC  | High Performance Computing                        |
| ILES | Implicit Large-Eddy Simulation                    |
| ISM  | Interstellar Medium                               |
| LES  | Large-Eddy Simulation                             |
| MHD  | Magnetohydrodynamics                              |
| PDE  | Partial Differential Equation                     |
| PLM  | Piecewise Linear Method                           |
| PPM  | Piecewise Parabolic Method                        |
| RANS | Reynolds-Averaged Numerical Simulation            |
| SATM | School of Aerospace, Technology and Manufacturing |
| SMR  | Static Mesh Refinement                            |
| TGV  | Taylor–Green Vortex                               |
| 2D   | Two Dimensional                                   |
| 3D   | Three Dimensional                                 |

## Greek Symbols

|               |                                            |
|---------------|--------------------------------------------|
| $\delta$      | Kronecker delta                            |
| $\Delta x$    | Base length of differential control volume |
| $\Delta t$    | Timestep size                              |
| $\varepsilon$ | Energy dissipation rate                    |
| $\gamma$      | Ratio of specific heats                    |
| $\eta$        | Kolmogorov length scale                    |
| $\kappa$      | Thermal conductivity coefficient           |



|                       |                                                |
|-----------------------|------------------------------------------------|
| $\lambda_T$           | Taylor micro-scale                             |
| $\mu$                 | Dynamic viscosity of fluid                     |
| $\nu$                 | Kinematic viscosity of fluid                   |
| $\Omega$              | Computational domain                           |
| $\rho$                | Density of fluid                               |
| $\sigma$              | Limited slope                                  |
| $\tau_{ij}$           | Viscous stress tensor components               |
| $\tau_\eta$           | Characteristic time scale of the smallest eddy |
| $\phi$                | Slope limiter function                         |
| $\boldsymbol{\omega}$ | Vorticity vector                               |
| $\omega$              | Vorticity magnitude                            |
| $\xi$                 | Polynomial coefficient                         |

### Latin Symbols

|              |                                                        |
|--------------|--------------------------------------------------------|
| $A$          | Differential area element                              |
| $\mathbf{B}$ | Magnetic flux density vector                           |
| $C_{CFL}$    | Courant number                                         |
| $c$          | Velocity of sound                                      |
| $D_h$        | Hydraulic diameter                                     |
| $\mathbf{F}$ | Vector of fluxes in the $x$ -direction                 |
| $\mathbf{G}$ | Vector of fluxes in the $y$ -direction                 |
| $\mathbf{H}$ | Vector of fluxes in the $z$ -direction                 |
| $k$          | Wavenumber; turbulent kinetic energy per unit mass     |
| $\ell$       | Integral length scale                                  |
| $L$          | Characteristic length                                  |
| $L_e$        | Hydrodynamic entrance length                           |
| $N$          | Number of grid points/cells                            |
| $N_t$        | Number of time steps                                   |
| $P$          | Thermodynamic pressure                                 |
| $p_{ref}$    | Reference pressure                                     |
| $\mathbf{q}$ | Heat flux vector                                       |
| $Re$         | Reynolds number                                        |
| $Re_T$       | Intrinsic Reynolds number                              |
| $S$          | Strain rate                                            |
| $T$          | Characteristic time scale of large eddy                |
| $\mathbf{U}$ | Vector of conservative variables                       |
| $U$          | Characteristic velocity scale of large eddy            |
| $\mathbf{V}$ | Velocity vector                                        |
| $\dot{K}$    | Rate at which kinetic energy leaves the largest eddies |
| $u_\eta$     | Characteristic velocity scale of the smallest eddy     |
| $x$          | $x$ -direction                                         |
| $y$          | $y$ -direction                                         |
| $z$          | $z$ -direction                                         |

# Chapter 1

## Introduction

The accurate and efficient numerical simulation of fluid flows remains a central challenge in computational fluid dynamics (CFD), particularly for problems spanning a wide range of spatial and temporal scales. Advances in numerical methods, computing power, and adaptive strategies have enabled the study of increasingly complex flows in both engineering and astrophysical contexts, where fluid dynamics governs processes such as star formation, disk accretion, and the evolution of interstellar and intergalactic media. At the frontier, direct numerical simulations (DNS) deliver unmatched physical detail but require immense computational resources, placing them beyond routine use and motivating the development of adaptive approaches, such as Athena++’s adaptive mesh refinement (AMR) capability, that aim to preserve high-fidelity accuracy while significantly reducing computational expense.

### 1.1 Background

The finite volume method (FVM) is widely used in CFD for its conservative formulation, robustness, and ability to handle complex geometries. Modern FVM-based tools are applied to a broad range of hydrodynamic problems, enabling detailed investigation of flow behaviour beyond the reach of experiments.

Mesh resolution is a key factor in simulation accuracy and cost. Flows with sharp gradients—such as shocks, discontinuities, boundary layers, shear layers, and vortices—require higher resolution in limited regions. Using a uniformly fine mesh to meet this requirement across the entire domain is computationally expensive. AMR addresses this by dynamically refining the mesh where needed and coarsening it elsewhere, allowing more efficient use of computational resources.

The effectiveness of AMR depends on factors such as refinement criteria, coarsening strategies, interpolation schemes, and numerical stability across different flow regimes. Evaluating these aspects is essential to determine the practical capability of a CFD tool.

As an illustration of the computational demands of high-fidelity simulations, Figure 1.1 shows results from the world’s largest supersonic magnetohydrodynamic (MHD) turbu-

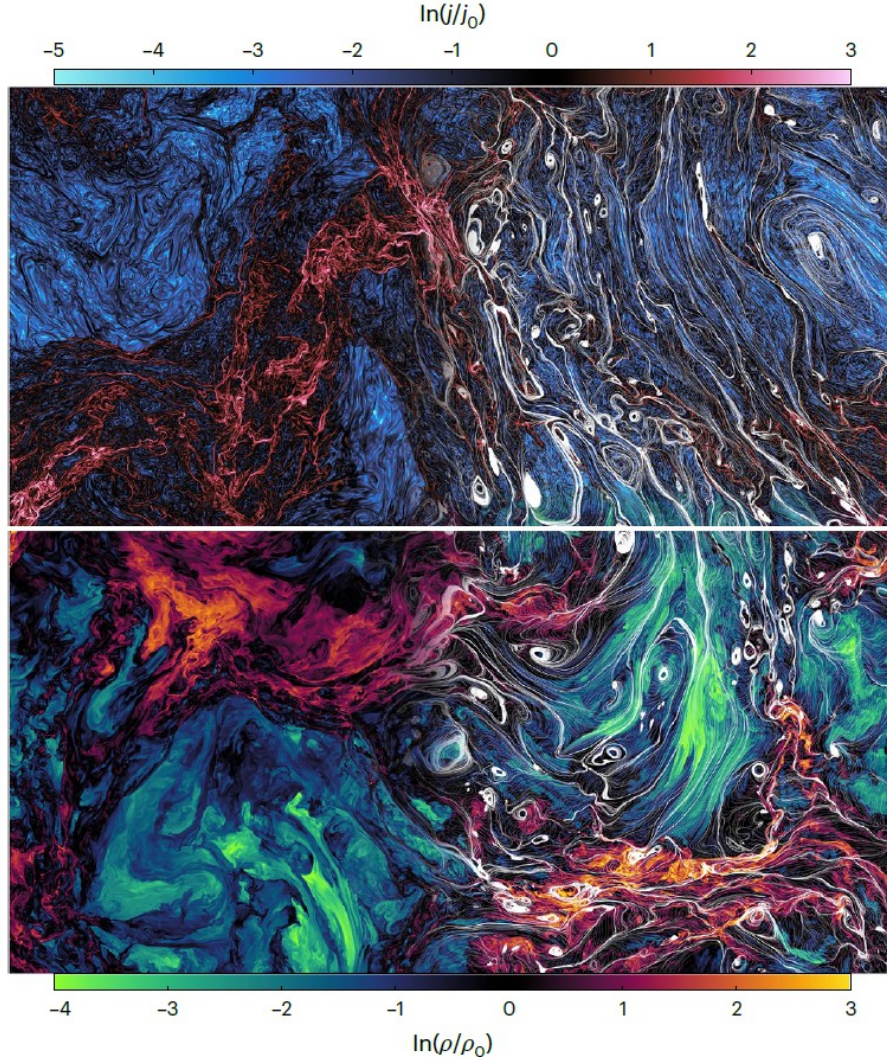


Figure 1.1: Two-dimensional slice of current density and mass density from a supersonic MHD turbulence DNS, showing fine-scale current sheets and shocked filaments (adapted from Beattie et al. [1])

lence simulation. The calculation resolved a triply periodic domain at  $10,080^3$  grid cells, capturing both supersonic and subsonic turbulent cascades with a self-generated, dynamo-sustained magnetic field. The image presents a two-dimensional slice of the logarithmic current density  $\ln\left(\frac{j}{j_0}\right)$  (top) and mass density  $\ln\left(\frac{\rho}{\rho_0}\right)$  (bottom), with magnetic field lines overlaid in white. Fractal current sheets (red in the top panel) coincide with dense shocked filaments (yellow in the bottom panel), while extended voids appear in blue and green. This unprecedented calculation required over 80 million CPU hours on nearly 140,000 compute cores of a high-performance supercomputer, integrated for about two large-eddy turnover times to obtain statistically converged results. Although such direct numerical simulations (DNS) provide unparalleled physical detail, their cost places them beyond routine use, highlighting the need for adaptive approaches, such as Athena++'s adaptive mesh refinement (AMR) capability, that can focus resolution where it matters most while

reducing computational expense.

## 1.2 Motivation

Athena++ employs a finite volume method based framework. While it has been applied to a range of hydrodynamic and MHD problems, a systematic evaluation of its AMR performance for viscous hydrodynamic flows is lacking. Previous studies have used static mesh refinement for DNS-like simulations, but to the best of current knowledge, Athena++ has not been employed for DNS of viscous hydrodynamic flows using AMR. Understanding how effectively Athena++ adapts the mesh across laminar and turbulent regimes is key to assessing its suitability for accurate and efficient simulations.

Under-resolved DNS configurations—where the mesh is intentionally too coarse to resolve all scales—offer a stringent test of AMR’s ability to recover essential flow features. If AMR can selectively refine critical regions, simulations may approach DNS fidelity without the full computational cost. Evaluating Athena++ in this context provides both scientific insight into its numerical performance and practical guidance for its application to large-scale simulations with limited resources.

## 1.3 Objectives of the Study

The aim of this study is to evaluate the performance of Athena++, with particular emphasis on its AMR capability, through the following objectives:

1. **Benchmark testing in laminar and turbulent regimes** – Assess Athena++ against established viscous flow benchmarks to evaluate accuracy, stability, and efficiency.
2. **Assessment of AMR performance** – Investigate how effectively AMR improves local resolution in key flow regions while maintaining computational efficiency.
3. **Under-resolved DNS evaluation** – Test AMR in a benchmark under-resolved DNS setup to determine its ability to capture essential flow features compared to uniform mesh simulations.
4. **Comparative analysis** – Compare Athena++ results with reference data to identify strengths, limitations, and potential improvements in the AMR strategy.

Through this structured investigation, the study aims to provide a comprehensive understanding of Athena++’s capabilities and limitations in applying AMR to viscous hydrodynamic flows.

# Chapter 2

## Literature Review

This chapter reviews literature in five main areas relevant to the present work. It begins with historical developments in turbulence research, outlining the transition from early analytical theories to the advent of DNS for resolving turbulent structures in detail. The focus then shifts to applications of DNS in astrophysical hydrodynamics, where it has been used to study multi-scale phenomena such as detonations and stellar flows. The third part examines the finite volume method as a widely adopted framework for simulations of astrophysical magnetohydrodynamic flows, along with the open-source tools that have enabled reproducible and accessible studies of astrophysical flows. The fourth section reviews foundational work on AMR and its progression to the efficient, scalable implementation in Athena++. The chapter concludes with a brief discussion of benchmark problems that serve as standard tests for validating numerical methods and assessing performance.

A seminal work on analytical turbulence theory by Steven A. Orszag [2] provides a comprehensive survey of the state of turbulence theory at the time. The formulation of turbulence problems in terms of moments is outlined, with particular emphasis on the closure problem, illustrated through the quasi-normal approximation. The concept of dynamical relaxation through nonlinear interactions is introduced, leading to the development of eddy relaxation times and the direct-interaction approximation (DIA). Numerical solutions of turbulence equations are compared with direct numerical simulations of the Navier–Stokes equations, highlighting the need to connect analytical theories with practical shear-flow applications. The study concludes by acknowledging the persistent challenges of turbulence research while proposing systematic improvements to theoretical predictions and closer integration of numerical simulations with models for practical problems such as pressure drops in pipe flow.

Orszag and Patterson [3] were among the pioneers of DNS of turbulence, making a significant contribution to the field. The Navier–Stokes equations were solved numerically for three-dimensional homogeneous isotropic turbulence using a pseudo-spectral method, providing insights into the energy spectrum of turbulent flows. Simulations at wind-tunnel-scale Reynolds numbers ( $Re_\lambda \approx 35$ ) demonstrated that, contrary to prevailing skepticism, DNS could be both feasible and economical at moderate  $Re$  values encountered in experiments. Strong agreement was observed with predictions of the

direct-interaction approximation, particularly for large-scale flow dynamics. Although the resolution was insufficient to capture inertial-range features such as the Kolmogorov spectrum, the results highlighted the importance of retaining a long spectral tail to model energy dissipation accurately. This study provided comprehensive turbulence data and established DNS as a benchmark for evaluating turbulence theories under controlled conditions.

The study of fully developed incompressible turbulent channel flow by Kim, Moin, and Moser [4] in 1987 marked a major step in the development of DNS. As the first successful simulation of wall-bounded turbulence, it demonstrated the feasibility of resolving detailed turbulent structures directly and strongly influenced subsequent research on channel flows. The simulation databases were made publicly available, providing a lasting resource for further analysis and comparison. The results became an important benchmark for assessing the accuracy of DNS and set a reference standard for future studies. This work was later extended by Moser, Kim, and Mansour [5] who presented simulations at higher Reynolds numbers, building on and broadening the scope of the 1987 study.

While these early DNS efforts established the feasibility and value of resolving turbulent flows in canonical fluid dynamics settings, this approach was also used for astrophysical hydrodynamics, where DNS has provided critical insights into complex multi-scale processes such as nuclear detonations and stellar flows. DNS has become a valuable tool for studying astrophysical detonations in degenerate matter, showing that nuclear burning often develops a cellular structure due to transverse instabilities at the detonation front [6]. This multidimensional character of detonations is central to modeling Type Ia supernovae, but simulations face major challenges because the nuclear flame width ( $10^{-5}$ – $10$  cm) is many orders of magnitude smaller than the stellar radius ( $10^7$ – $10^8$  cm), making full resolution computationally impractical. To overcome this, sub-grid models are used to approximate small-scale flame physics within full-star simulations, guided by DNS data that provide essential quantities such as detonation speed, burning timescales, element abundances, and energy release under white dwarf conditions. Incorporating these results into large-scale simulations enables more accurate modeling of Type Ia supernovae without resolving every fine-scale feature.

Brandenburg et al. [7] used DNS, mean-field simulations (MFS), and implicit large-eddy simulations (ILES) to show that strongly stratified hydromagnetic turbulence with a vertical imposed field can produce magnetic flux concentrations resembling sunspots. The simulations reveal large-scale instabilities forming flux tubes that extend several density scale heights with energies comparable to the turbulent kinetic energy. These structures are confined by downflows and inflows, supporting vertical flux concentration as a viable mechanism for sunspot formation.

Beattie et al. [1] carried out very high-resolution DNS of compressible, magnetized interstellar turbulence with computational grids up to  $10,080^3$  cells. The study clearly identified coexisting kinetic and magnetic energy cascades, revealing distinct turbulence regimes shaped jointly by compressibility and magnetization. These results provide valuable constraints on interstellar medium (ISM) turbulence and offer predictions that can be

tested with upcoming radio observations.

DNS has yielded important insights into localized astrophysical processes, but modeling large-scale magnetized flows requires numerical frameworks that combine accuracy, robustness, and computational efficiency. In this context, the Finite Volume Method (FVM) has become the dominant approach for magnetohydrodynamic simulations, particularly in astrophysics.

The popularity of FVM stems from inherent conservation properties, robust handling of shocks and discontinuities, and effectiveness in maintaining the divergence-free condition of magnetic fields, an essential requirement for physically accurate MHD solutions. FVM also offers flexibility for implementation on unstructured grids, curvilinear coordinates, and with adaptive mesh refinement (AMR), making it well-suited for complex geometries. Widely adopted by the community, many leading MHD codes—such as Athena++ [8, 9], PLUTO [10, 11], FLASH [12], RAMSES [13], ENZO [14], and MPI-AMRVAC [15]—are built on FVM frameworks. A benchmark study by Kritsuk et al. [16] comparing nine astrophysical MHD codes found that those based on FVM generally performed better than their finite difference counterparts. Further advancements, such as the fourth-order accurate FVM introduced by Felker and Stone in the Athena code using upwind constrained transport, demonstrate the method’s capacity for high accuracy and stability in multidimensional MHD problems, reinforcing its preference in the field [17]. Despite these advances, many astrophysical problems involve a wide range of interacting spatial and temporal scales, making AMR a crucial extension to the FVM framework.

Early theoretical progress on adaptive mesh refinement was driven by advances in a posteriori error estimation techniques. Babuska and Rheinboldt [18] provided rigorous error bounds and systematic strategies for mesh adaptation, thereby contributing to the theoretical foundations of adaptive finite element methods (FEM). Their work built upon earlier insights from Crouzeix and Raviart [19], who first introduced formulations of a posteriori error estimates and demonstrated their use in adaptive refinement for elliptic problems. By expanding the framework with greater mathematical generality and practical algorithms applicable to a wide class of problems, Babuska and Rheinboldt established a foundation that has shaped modern adaptive FEM and influenced decades of research in computational mechanics and numerical analysis.

Berger & Oliger [20] introduced the first structured AMR algorithm tailored for time-dependent hyperbolic partial differential equations (PDEs). Using Richardson-type estimates of truncation error, grids are refined or coarsened as needed to meet the specified accuracy. The method used a hierarchy of nested, logically rectangular grids that adaptively refined both space and time domains to track dynamic solution features such as shocks and wave fronts. This approach enabled significant computational savings while preserving high-resolution accuracy where needed, and it laid the foundation for future AMR developments. As a foundational algorithm, it influenced a broad spectrum of applications in fluid dynamics, astrophysics, and meteorology. Building upon this framework, Berger & Colella [21] developed a more robust and practical AMR method specifically for compressible flows with strong shocks. Their work introduced a conservative, block-

structured AMR scheme that ensured proper flux corrections across coarse-fine boundaries, maintaining global conservation laws. The method was especially well-suited for shock hydrodynamics and remains the most widely referred in CFD. Together, these two works established the theoretical and algorithmic basis for structured AMR, shaping its adoption in numerous high-resolution simulation codes across science and engineering.

The AMR implemented in Athena++ framework [8] represents an advancement specifically tailored for astrophysical fluid dynamics but designed for broad applicability across various physics domains. It utilizes a block-based AMR structure organized in a tree hierarchy, enabling efficient and scalable performance on massively parallel systems. A key innovation is its task-list-driven dynamic execution model, which overlaps communication and computation, facilitating multiphysics simulations with different physics modules in distinct regions. The framework incorporates robust MHD solvers derived from the original Athena code, enhanced with features like higher-order integration, curvilinear coordinates support, and super-time-stepping for diffusion processes. Performance benchmarks demonstrate a very good scalability, with over 80% parallel efficiency on half a million threads, highlighting its suitability for exascale computing.

To complement advances in adaptive mesh refinement, numerical methods must also be validated against well-defined benchmark problems that serve as standard references for accuracy and performance. The benchmark problems employed in this work encompass a set of canonical cases designed to validate computational accuracy, assess solver robustness, and evaluate mesh adaptation strategies. The steady, fully developed plane channel flow, exhibiting a parabolic velocity distribution, is taken from standard fluid mechanics references such as *Fundamentals of Fluid Mechanics* by Munson et al. [22] and *Fluid Mechanics* by Frank White [23], providing a baseline verification of laminar flow predictions. The two-dimensional lid-driven cavity problem is validated against the high-fidelity numerical data set of Ghia et al. [24], which remains a widely accepted standard for code-to-code comparison in incompressible CFD. The two-dimensional Taylor–Green vortex, originally formulated by Taylor [25] to examine the decay of vortices in a viscous fluid, is employed to investigate the accuracy of vorticity transport and dissipation modeling. Its three-dimensional counterpart, referencing the work of Brachet et al. [26], Rees et al. [27], and Shirokov and Elizarova [28], extends this analysis to transitional and decaying turbulence regimes. All cases are implemented within the Athena++ framework, with the Taylor–Green vortex problems in two- and three dimensions also serving as primary testbeds to assess solver performance and AMR capability to capture multiscale flow structures.



# Chapter 3

## Theoretical Background and Numerical Methodology

This chapter establishes the theoretical and methodological foundation for the present work, linking the underlying physics of the problem to the computational tools used in its investigation. It begins with an overview of hydrodynamic and magnetohydrodynamic flows, introducing the governing principles and focusing on turbulence—its nature, key characteristics, and relevant physical scales. A brief review follows of numerical strategies for simulating turbulent flows, including DNS, Large-Eddy Simulation (LES), and Reynolds-Averaged Navier–Stokes (RANS) methods. While applicable to both hydrodynamic and MHD turbulence, these approaches require additional modeling for MHD and are less commonly applied. In this study, the focus is on viscous hydrodynamic flows in laminar and turbulent regimes, with an under-resolved DNS configuration used to evaluate the performance of Athena++’s AMR capability.

Building on this background, the chapter then turns to Athena++, the simulation framework used throughout this study. The presentation covers the governing equations implemented within the code and the numerical procedures through which they are solved, with attention given to features relevant to the current application.

Finally, the chapter examines Athena++’s AMR capability, a key component in resolving multiscale flow structures efficiently. The theoretical basis of AMR is outlined, followed by the guiding design philosophy and the specific implementation strategy employed in this work.

### 3.1 Hydrodynamic and Magnetohydrodynamic Flows

Hydrodynamic flows, governed by the Navier–Stokes equations, encompass a wide range of phenomena from laminar pipe flows and boundary layers in engineering systems to large-scale atmospheric and oceanic currents in geophysical contexts. A striking engineering example is the trailing wake vortex behind a jetliner, which vividly illustrates the persistence and complexity of hydrodynamic structures in real-world flows (Figure 3.1a). When electrically conducting fluids are present, such as plasmas, liquid metals, or salt-water, the interaction between the flow and magnetic fields gives rise to magnetohydro-

dynamic behavior, described by the coupled Navier–Stokes and Maxwell’s equations. Natural manifestations include solar prominences, where the motion of hot plasma is shaped by the Sun’s magnetic field, producing spectacular large-scale MHD structures (Figure 3.1b). In both hydrodynamic and MHD systems, turbulence often emerges at sufficiently high Reynolds numbers, introducing a complex, multiscale character to the motion. The following subsection examines the fundamental characteristics and scales of turbulence relevant to the present study.

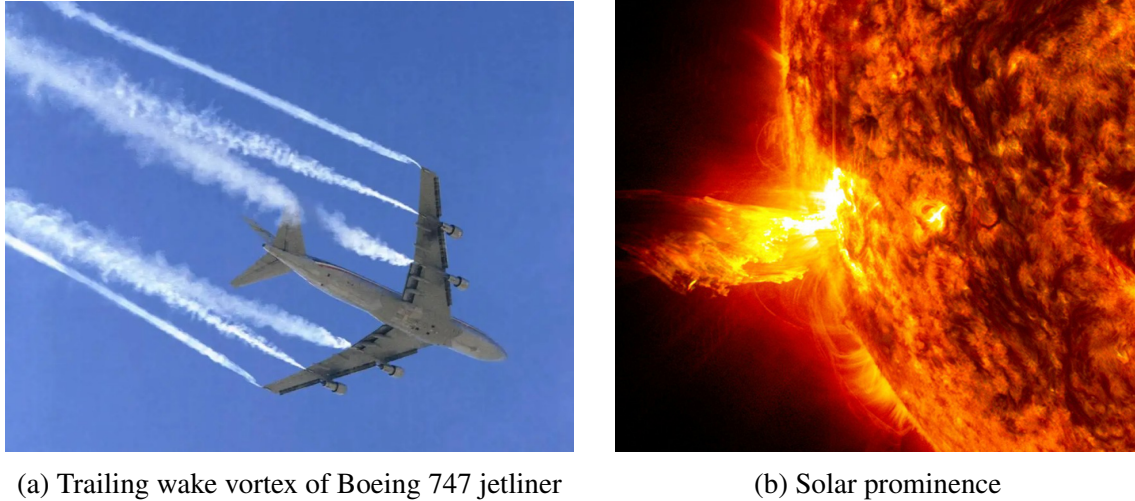


Figure 3.1: Turbulent flows in engineering applications and in natural phenomena (Source: NASA, <https://www.nasa.gov>)

### 3.1.1 Characteristics of Turbulence

Turbulence is not tied to a specific fluid but is a property of fluid motion itself. Its dynamics are largely independent of the molecular properties of the fluid and are instead governed by the principles of continuum mechanics, specifically fluid mechanics. The shared characteristics of turbulence form the basis for distinguishing it from laminar flow, although each turbulent flow also exhibits features unique to its initial and boundary conditions.

#### 3.1.1.1 Irregularity and Statistical Nature

Turbulent flows display irregular, unsteady motion in both space and time, a hallmark feature illustrated in Figure 3.2. Even under nominally identical experimental conditions, repeated velocity measurements produce different instantaneous histories due to small variations in the setup, yet time-averaged values remain consistent. This combination of instantaneous randomness with statistical reproducibility is a key reason why turbulence research relies heavily on statistical methods rather than deterministic solutions.

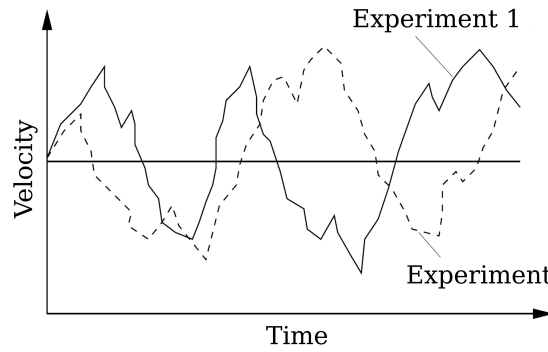


Figure 3.2: Instantaneous velocity obtained in two different experimental measurements under identical conditions (adapted from [29])

### 3.1.1.2 Coherent Structures and Energy Dissipation

Turbulence contains coherent rotational structures—regions of strong vorticity that form, interact, and break down unpredictably. Large vortices fragment into smaller ones, transferring energy to progressively smaller scales until viscous dissipation converts it into heat, as shown in Figure 3.3. This energy cascade is inherently dissipative, and large structures require a continuous energy supply from the mean flow to persist.

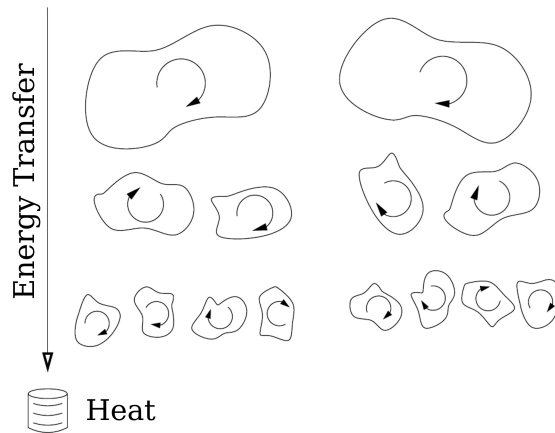


Figure 3.3: Kinetic energy transport from large scale structures to smaller scales (adapted from [29])

### 3.1.1.3 Enhanced Transport and Mixing

The multiscale agitation of turbulence greatly increases the effective diffusivity for the transport of mass, momentum, and energy. Velocity fluctuations promote rapid mixing and entrainment, drawing surrounding fluid into the turbulent region and enabling dispersion far faster than molecular diffusion alone.

### 3.1.1.4 Complexity and Modeling Challenges

Despite being governed by deterministic equations, turbulence remains one of the most challenging problems in classical physics. The interaction between coherent structures and background fluctuations produces nonlinear, non-equilibrium behavior that resists general analytical solutions. A common simplification is to decompose the velocity field into mean and fluctuating components, forming the basis of many theoretical and numerical models, though only as an approximation of reality.

The combination of these characteristics gives turbulence its inherently multiscale nature, where energy, momentum, and scalar quantities are distributed across a broad range of spatial and temporal scales. Understanding this scale separation is essential for both theoretical analysis and numerical modeling, and forms the basis of the following discussion on turbulence scales.

### 3.1.2 Scales of Turbulence

Turbulent flow comprises a spectrum of coherent vortical structures, with dynamics governed by the scales of the associated eddies. Assuming turbulence consists of eddies of varying sizes, their interaction can be described by Richardson's energy cascade hypothesis [30], illustrated schematically in Figure 3.3. While the cascade concept cannot be formally derived from first principles, it forms a framework supported by extensive experimental evidence.

Turbulent flows contain a hierarchy of eddies ranging from large, energetic structures to the smallest dissipative motions. Through mutual interactions, eddies deform and strain one another, transferring kinetic energy from the largest to progressively smaller scales. This process, first articulated by Richardson [30] and later quantified through Kolmogorov's statistical theory [31], proceeds without appreciable viscous influence in the inertial subrange until the motion reaches scales where viscosity dominates and the energy is dissipated as heat. The size of the smallest eddies is determined by both the energy input at the large scales and the fluid viscosity, and their relative size compared to the largest eddies is set by the Reynolds number.

The length scales of turbulence are commonly described in three categories:

1. **Integral length scale** ( $\ell$ ): The largest scale of turbulence, representing the size of the energy-containing eddies that are set by the geometry and boundary conditions of the flow. Examples include a fraction of the pipe diameter in internal flows, the boundary-layer thickness over an airfoil, or kilometer-scale eddies in the atmosphere.
2. **Taylor microscale** ( $\lambda_T$ ): An intermediate scale associated with the mean spatial extent of velocity gradients. It lies between the large energy-containing eddies and the smallest dissipative scales, representing a transition region where both inertia and viscosity play a role.

3. **Kolmogorov scale ( $\eta$ )**: The smallest scale of motion, at which viscous effects dominate and turbulent kinetic energy is dissipated. Its size depends on the kinematic viscosity  $\nu$  and the dissipation rate  $\varepsilon$ , and is often in the micrometer-to-millimeter range in engineering and geophysical flows.

These scales are of direct relevance to both experimental and numerical investigations. In computational fluid dynamics, they inform the grid resolution requirements, while in physical experiments they guide the selection of sensor size and measurement resolution.

### 3.1.2.1 Richardson's Hypothesis

In Richardson's view, the largest eddies in a turbulent flow have velocity and length scales comparable to those of the mean flow, giving Reynolds numbers  $Re \gg 1$  where viscous effects are negligible. These large, anisotropic eddies depend on boundary conditions and receive continuous energy input from the surrounding fluid. With no dissipation at these scales, energy is transferred to smaller eddies at a rate estimated dimensionally as  $u^3/\ell$ , where  $u$  and  $\ell$  are the characteristic velocity and length scale, and  $\tau = \ell/u$  is the eddy turnover time.

The cascade proceeds through successive eddy break-ups (Figure 3.3), transferring energy to progressively smaller scales under inertial forces until  $Re \sim 1$ . At this point, viscous effects dominate and kinetic energy is dissipated at the smallest scales at a rate  $\varepsilon$  set by the large-scale production, i.e.,  $\varepsilon$  is independent of molecular viscosity and remains of order  $u^3/\ell$ .

### 3.1.2.2 Kolmogorov's Hypotheses

Building on the concept of the energy cascade, Kolmogorov [31] proposed a framework for quantifying the scales and energy transfer in high- $Re$  turbulence. Kolmogorov hypothesised that anisotropy imposed on the large scales by boundary conditions is progressively lost as energy is transferred to smaller eddies, and that the smallest scales of motion—sufficiently far from boundaries and singularities—are locally isotropic and statistically universal. Their statistics depend primarily on the energy flux from the large scales and the viscous dissipation rate  $\varepsilon$ . Provided these small scales evolve on timescales short compared to the large-eddy turnover time, they can rapidly adjust to the incoming energy flux, leading to a dynamically balanced budget in which the energy transfer rate equals the dissipation rate.

At the smallest (Kolmogorov) scales, the dynamics are governed solely by  $\varepsilon$  and the kinematic viscosity  $\nu$ , yielding the dimensional estimates

$$\eta \sim \left( \frac{\nu^3}{\varepsilon} \right)^{1/4}, \quad u_\eta \sim (\nu \varepsilon)^{1/4}, \quad \tau_\eta \sim \left( \frac{\nu}{\varepsilon} \right)^{1/2} \quad (3.1)$$

Consistency with Richardson's hypothesis requires that the Reynolds number based on the Kolmogorov scales,  $Re_\eta$ , be  $\mathcal{O}(1)$ , since the viscous and inertial effects are balanced.

Using the large-scale scaling  $\varepsilon \sim u^3/\ell$ , the Kolmogorov and large-scale quantities are related by

$$\frac{\eta}{\ell} \sim \text{Re}^{-3/4}, \quad \frac{u\eta}{u} \sim \text{Re}^{-1/4}, \quad \frac{\tau_\eta}{\tau} \sim \text{Re}^{-1/2} \quad (3.2)$$

where  $\text{Re}$  is the flow Reynolds number. These ratios are much less than unity for practical turbulent flows, reflecting the separation between the smallest and largest scales, which increases with  $\text{Re}$ .

Between the largest and smallest scales lies an intermediate range where viscous effects are negligible and eddies act purely to transfer energy downscale. The statistics in this inertial subrange are governed solely by  $\varepsilon$ , and no characteristic length-, velocity-, or time-scale exists apart from those of the boundaries of the range. A useful measure in this regime is the Taylor microscale  $\lambda_T$ , representing the mean spatial extent of velocity gradients, which is related to the large and Kolmogorov scales by

$$\frac{\lambda_T}{\ell} \sim \text{Re}^{-1/2}, \quad \frac{\lambda_T}{\eta} \sim \text{Re}^{1/4} \quad (3.3)$$

In the inertial subrange, the energy flux remains constant across scales, as no energy is supplied by the mean flow or lost through dissipation. Consequently, both the characteristic velocity and turnover time decrease systematically with eddy size.

### 3.1.2.3 Energy Spectrum

The turbulent kinetic energy distribution across scales is conveniently represented in spectral space, where a length scale is expressed via its wavenumber  $k$  (Figure 3.4). The velocity field is decomposed into Fourier modes, and the one-dimensional spectrum  $E(k)$  measures the kinetic energy content of each scale through the product of the Fourier coefficients and their complex conjugates.

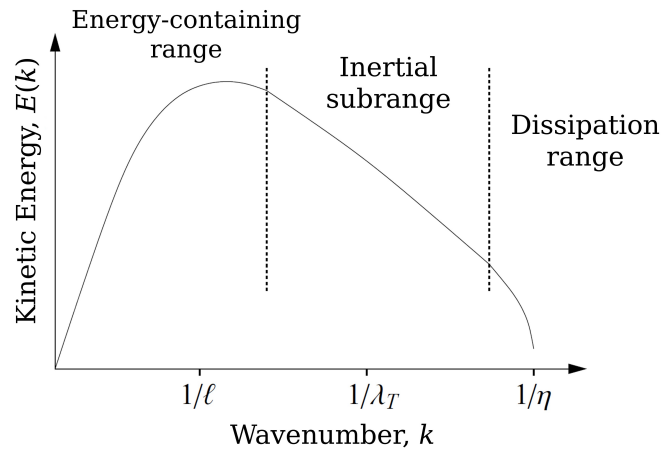


Figure 3.4: Typical energy spectrum of turbulent flows

The spectrum is typically divided into the energy-containing range (small  $k$ ), dominated by large anisotropic eddies that store most of the energy, and the universal equilibrium range (large  $k$ ), composed of small isotropic eddies. The latter is further split into the dissipation range, where viscosity dominates at the Kolmogorov scales and energy is dissipated, and the inertial subrange, where energy is transferred between scales by inertial forces without direct production or dissipation (Figure 3.4).

In the energy-containing range, the spectrum peaks at scales of order the integral length  $\ell$ , determined by the largest velocity correlations. Its shape depends on the strain rate  $S$ , dissipation rate  $\varepsilon$ , and flow geometry. In the dissipation range,  $\varepsilon$  and the kinematic viscosity  $\nu$  set the scale, directly linking the spectrum to fluid properties. Between these limits lies the inertial subrange, whose statistics depend solely on  $\varepsilon$  and whose extent increases with Reynolds number. Using dimensional arguments, Kolmogorov [31] obtained the celebrated  $-5/3$  law,

$$E(k) = C \varepsilon^{2/3} k^{-5/3} \quad (3.4)$$

where  $C$  is a universal constant. Although derived for homogeneous, isotropic, and stationary turbulence, this scaling has been widely confirmed in both experiments and simulations.

### 3.1.3 Numerical Approaches for Turbulence Simulation

Although decades of experimental and theoretical work have advanced the understanding of turbulence, accurately predicting turbulent flows remains challenging due to the absence of a universally applicable theoretical framework and closure model. In CFD, three main approaches are commonly employed: Direct Numerical Simulation (DNS), Reynolds-Averaged Navier–Stokes (RANS), and Large-Eddy Simulation (LES), each with distinct strengths and limitations. This section briefly outlines these methods, with emphasis on DNS, which forms the primary numerical approach in this work. The subsequent discussion will focus on its computational requirements, while later chapters will highlight its application within Athena++ and the role of AMR in enhancing its efficiency.

#### Reynolds-Averaged Navier–Stokes (RANS)

RANS methods solve the time-averaged Navier–Stokes equations, with turbulence effects represented by closure models such as  $k$ – $\varepsilon$  or  $k$ – $\omega$ . While computationally efficient and widely used in engineering practice, RANS accuracy is strongly dependent on the turbulence model and can be limited in flows with strong unsteadiness, separation, or complex three-dimensionality.

#### Large-Eddy Simulation (LES)

LES resolves the large, energy-containing turbulent eddies directly, while modeling the effect of smaller scales through a subgrid-scale (SGS) model. It offers higher fidelity than RANS and captures unsteady large-scale structures, but remains computationally expensive at high Reynolds numbers, especially for wall-bounded flows.

### Direct Numerical Simulation (DNS)

DNS is a high-fidelity computational technique that resolves all relevant spatial and temporal scales of turbulence without empirical modeling. Unlike RANS or LES, DNS captures the complete spectrum of turbulent motion—including the smallest dissipative scales—providing an unfiltered representation of flow dynamics. This enables fundamentally accurate, first-principles investigations into turbulence physics, making DNS indispensable in basic research and model development.

The strength of DNS lies in its precise, non-empirical insights into complex flows. However, its computational cost scales steeply with Reynolds number, restricting its use to simple geometries, low-to-moderate Reynolds numbers, or limited domains. Despite this, DNS has been extensively applied to homogeneous isotropic turbulence, transitional boundary layers, turbulent channel and pipe flows, and selected astrophysical and geophysical problems. The method’s ability to produce benchmark-quality data makes it essential for validating turbulence models and advancing theoretical understanding.

The computational requirements of DNS can be estimated using turbulence scaling laws, providing a basis for understanding its limitations and motivating the under-resolved DNS approach adopted in this work. These estimates, along with the specific benchmark problem used for evaluation, are discussed in the following sections.

#### 3.1.4 Computational Requirements for DNS

A central challenge in DNS is the need for a computational domain large enough to contain the largest naturally occurring eddies, while employing a grid fine enough to fully resolve the smallest dissipative scales. Even on today’s most powerful computers, this restricts DNS to Reynolds numbers orders of magnitude lower than those encountered in full-scale aeronautical or geophysical flows.

For homogeneous isotropic turbulence, the scaling of DNS cost with Reynolds number can be estimated from turbulence theory. The largest eddies can be characterised by a single length scale  $\ell$  and a velocity scale  $u$ , with a turnover time  $\tau = \ell/u$ . The turbulent kinetic energy per unit mass is

$$k \approx \frac{1}{2}u^2 \quad (3.5)$$

representing the energy contained in turbulent fluctuations and hence a measure of turbulence intensity.

The rate at which kinetic energy leaves the largest eddies is

$$\dot{K} \approx \frac{u^2}{\tau} \quad (3.6)$$

For statistically steady turbulence, this transfer rate is equal to the dissipation rate  $\varepsilon$ :

$$\dot{K} \approx \varepsilon \quad (3.7)$$



From these relations, the large-eddy time and length scales follow as

$$\tau \sim \frac{u^2}{\varepsilon}, \quad \ell \sim \frac{u^3}{\varepsilon} \quad (3.8)$$

The turbulence (integral) Reynolds number is then

$$\text{Re}_T = \frac{u\ell}{\nu} = \frac{u^4}{\nu\varepsilon} \quad (3.9)$$

where  $\nu$  is the kinematic viscosity.

From Kolmogorov theory, the smallest (dissipative) scales depend only on  $\varepsilon$  and  $\nu$ , giving the Kolmogorov scales (cf. Equation 3.1), which are restated here for clarity in the DNS context.

$$\eta = \left(\frac{\nu^3}{\varepsilon}\right)^{1/4}, \quad u_\eta = (\nu\varepsilon)^{1/4}, \quad \tau_\eta = \left(\frac{\nu}{\varepsilon}\right)^{1/2} \quad (3.10)$$

The number of grid points required in each spatial direction to resolve all scales is of order

$$\frac{\ell}{\eta} = \frac{u^3}{\varepsilon} \cdot \frac{\varepsilon^{1/4}}{\nu^{3/4}} = \left(\frac{u^4}{\nu\varepsilon}\right)^{3/4} = \text{Re}_T^{3/4} \quad (3.11)$$

Hence, the total number of grid points in 3D is

$$N^3 \sim \left(\frac{\ell}{\eta}\right)^3 \Rightarrow N^3 \sim \text{Re}_T^{9/4} \quad (3.12)$$

The timestep  $\Delta t$  is constrained by the CFL condition, scaling with the smallest grid spacing:

$$\Delta t \sim \frac{\eta}{u} \Rightarrow \Delta t \sim \tau \text{Re}_T^{-3/4} \quad (3.13)$$

The number of timesteps to simulate one large-eddy turnover time is therefore

$$N_t \sim \frac{\tau}{\Delta t} \sim \text{Re}_T^{3/4} \quad (3.14)$$

The total computational cost scales with the number of grid points and the number of timesteps:

$$\text{Cost} \sim N^3 \times N_t \Rightarrow \text{Cost} \sim \text{Re}_T^{9/4} \times \text{Re}_T^{3/4} \sim \text{Re}_T^3 \quad (3.15)$$

This steep scaling demonstrates why DNS remains feasible only at relatively low  $\text{Re}_T$  and in simple geometries, motivating the use of approaches such as under-resolved DNS in combination with adaptive mesh refinement to target high-resolution computations in regions of interest. To investigate these strategies, this thesis employs the open-source code Athena++, a high-order, flexible framework originally developed for astrophysical flows but well suited for hydrodynamic and magnetohydrodynamic applications. The following section provides an overview of Athena++ and its numerical infrastructure, before introducing the governing equations of viscous hydrodynamics that form the basis of the present study.

## 3.2 Athena++: An Overview

Athena++ is a modern, high-order, grid-based framework for solving the equations of hydrodynamics and magnetohydrodynamics (MHD), widely used in astrophysical fluid simulations. As the successor to the original Athena code [9], it is implemented in C++ for improved modularity, computational performance, and scalability.

At its core, Athena++ employs the Godunov method—a conservative, shock-capturing approach for hyperbolic systems of partial differential equations such as the Euler and MHD equations. This method computes inter-cell fluxes by solving local Riemann problems, enabling accurate treatment of discontinuities including shocks and contact surfaces. The code supports multiple Riemann solvers (e.g., Roe, HLLC, HLLD), high-order reconstruction schemes, and unsplit time integration, allowing robust simulation of a broad range of compressible flows.

A key feature of Athena++ is its block-structured Adaptive Mesh Refinement (AMR), which dynamically refines or coarsens the mesh in regions of interest according to user-specified criteria (e.g., density gradients, vorticity, magnetic field strength) [8]. This ensures high resolution in dynamically active regions—such as shocks or vortices—while reducing computational cost in quiescent areas. The AMR design supports efficient parallelization via domain decomposition, making it suitable for multiscale problems spanning large dynamic ranges, from compact stellar accretion zones ( $\sim 10^{11}$  cm) to the circumgalactic medium ( $\sim 10^{24}$  cm).

In this work, Athena++ is used as the solver for under-resolved DNS studies for purely hydrodynamic viscous flows, exploiting its AMR capability to concentrate DNS-level resolution where it is most needed. This serves as a step toward bridging high-fidelity turbulence studies with large-scale astrophysical simulations.

## 3.3 Governing Equations implemented in Athena++

By default, Athena++ solves the equations of ideal MHD in conservative form [9]. Ideal MHD describes the dynamics of a perfectly conducting fluid interacting with magnetic and electric fields, combining the principles of fluid dynamics with Maxwell’s equations for electromagnetism. In this formulation, the following assumptions are made:

1. The electrical conductivity is assumed infinite (no resistive dissipation).
2. The magnetic field is “frozen” into the fluid and advected with it.
3. Viscosity, resistivity, and heat conduction are neglected.

Under these assumptions, the governing equations for mass, momentum, total energy, and magnetic induction in conservative form are:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 \quad (3.16)$$

$$\frac{\partial(\rho \mathbf{V})}{\partial t} + \nabla \cdot [\rho \mathbf{V} \otimes \mathbf{V} + (P^* \mathbf{I} - \mathbf{B} \otimes \mathbf{B})] = 0 \quad (3.17)$$

$$\frac{\partial E}{\partial t} + \nabla \cdot [(E + P^*) \mathbf{V} - \mathbf{B}(\mathbf{B} \cdot \mathbf{V})] = 0 \quad (3.18)$$

$$\frac{\partial \mathbf{B}}{\partial t} - \nabla \times (\mathbf{V} \times \mathbf{B}) = 0 \quad (3.19)$$

where

|                                      |                                                                        |
|--------------------------------------|------------------------------------------------------------------------|
| $\rho$                               | : mass density of fluid                                                |
| $\mathbf{V}$                         | : velocity vector field                                                |
| $\rho \mathbf{V}$                    | : mass flux density or momentum density                                |
| $\rho \mathbf{V} \otimes \mathbf{V}$ | : momentum density flux tensor                                         |
| $P^*$                                | : total pressure (sum of thermodynamic pressure and magnetic pressure) |
| $\mathbf{I}$                         | : identity tensor                                                      |
| $\mathbf{B}$                         | : magnetic field vector                                                |
| $\mathbf{B} \otimes \mathbf{B}$      | : magnetic stress tensor                                               |
| $E$                                  | : total energy density (internal ( $e$ ), kinetic and magnetic energy) |

In ideal MHD, the total pressure  $P^*$  is given by

$$P^* = P + \frac{1}{2} \mathbf{B}^2 \quad (3.20)$$

where,  $P$  is the thermodynamic pressure and  $\frac{1}{2} \mathbf{B}^2$  is the magnetic pressure.

The total energy  $E$  can be given by

$$E = \frac{1}{2} \rho \mathbf{V}^2 + \frac{P}{\gamma - 1} + \frac{1}{2} \mathbf{B}^2 \quad (3.21)$$

where,  $\frac{P}{\gamma - 1}$  is the internal (thermal) energy density,  $\gamma$  is the adiabatic index (ratio of specific heats), and  $\frac{1}{2} \mathbf{B}^2$  is the magnetic energy density.

Thus, in addition to Equations 3.16 to 3.19, the equation of state is also used to consider the internal energy density. Numerically, the magnetic energy density is the same as the magnetic pressure; however, those terms have different physical interpretations.

### 3.3.1 Governing Equations for Viscous Hydrodynamic Flows

In this study, the magnetic terms from the ideal MHD equations are omitted to focus on purely viscous hydrodynamic flows. The governing equations therefore reduce to the compressible Navier–Stokes equations for a Newtonian viscous, heat-conducting fluid.

These equations form the basis of the DNS framework to be tested in Athena++, where both inertial and viscous effects are resolved.

In conservation form, the equations for mass, momentum, and total energy are:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{V}) = 0 \quad (3.22)$$

$$\frac{\partial (\rho \mathbf{V})}{\partial t} + \nabla \cdot [\rho \mathbf{V} \otimes \mathbf{V} + P \mathbf{I} - \boldsymbol{\tau}] = 0 \quad (3.23)$$

$$\frac{\partial E}{\partial t} + \nabla \cdot [(E + P) \mathbf{V} - \boldsymbol{\tau} \cdot \mathbf{V} + \mathbf{q}] = 0 \quad (3.24)$$

where the viscous stress tensor for a Newtonian fluid is

$$\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu (\nabla \cdot \mathbf{V}) \delta_{ij}. \quad (3.25)$$

$E$  is the total energy density

$$E = \frac{1}{2} \rho |\mathbf{V}|^2 + \rho e \quad (3.26)$$

and the heat flux vector  $\mathbf{q}$  is given by Fourier's law

$$\mathbf{q} = -\kappa \nabla T \quad (3.27)$$

The system is closed with the equation of state for an ideal gas

$$P = (\gamma - 1) \rho e \quad (3.28)$$

where  $\gamma$  is the ratio of specific heats.

### 3.3.2 Compact Form of Governing Equations for Viscous Hydrodynamic Flows

In Cartesian coordinates, Equations 3.22–3.24 can be expressed compactly as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} + \frac{\partial \mathbf{H}}{\partial z} = 0 \quad (3.29)$$

where,  $\mathbf{U}$  is the vector of conserved variables, while  $\mathbf{F}$ ,  $\mathbf{G}$  and  $\mathbf{H}$  are the vectors of fluxes respectively in x, y, and z directions.

The vectors  $\mathbf{U}$ ,  $\mathbf{F}$ ,  $\mathbf{G}$  and  $\mathbf{H}$  can be written as,

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix} \quad (3.30)$$

The flux vectors consist of inviscid and viscous parts

$$\mathbf{F} = \mathbf{F}^{(i)} - \mathbf{F}^{(v)}, \quad \mathbf{G} = \mathbf{G}^{(i)} - \mathbf{G}^{(v)}, \quad \mathbf{H} = \mathbf{H}^{(i)} - \mathbf{H}^{(v)} \quad (3.31)$$

The **inviscid** (Euler) parts are

$$\mathbf{F}^{(i)} = \begin{bmatrix} \rho u \\ \rho u^2 + P \\ \rho uv \\ \rho uw \\ (E + P)u \end{bmatrix}, \quad \mathbf{G}^{(i)} = \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + P \\ \rho vw \\ (E + P)v \end{bmatrix}, \quad \mathbf{H}^{(i)} = \begin{bmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + P \\ (E + P)w \end{bmatrix} \quad (3.32)$$

The **viscous** parts are

$$\mathbf{F}^{(v)} = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{xy} \\ \tau_{xz} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} - q_x \end{bmatrix} \quad (3.33)$$

$$\mathbf{G}^{(v)} = \begin{bmatrix} 0 \\ \tau_{yx} \\ \tau_{yy} \\ \tau_{yz} \\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - q_y \end{bmatrix} \quad (3.34)$$

$$\mathbf{H}^{(v)} = \begin{bmatrix} 0 \\ \tau_{zx} \\ \tau_{zy} \\ \tau_{zz} \\ u\tau_{zx} + v\tau_{zy} + w\tau_{zz} - q_z \end{bmatrix} \quad (3.35)$$

The next section presents the numerical procedure used in Athena++ to solve the governing equations.

### 3.4 Numerical Procedure used in Athena++

Athena++ employs the finite-volume method as its core spatial discretization strategy, ensuring exact conservation of mass, momentum, and energy in the discrete sense. By integrating the governing equations over control volumes and updating cell-averaged conserved quantities via fluxes across cell interfaces, this approach is well suited to capturing shocks, discontinuities, and complex flow features. The method naturally accommodates Cartesian meshes and extends efficiently to AMR, making it particularly robust for high-resolution hydrodynamic and magnetohydrodynamic simulations.

#### 3.4.1 Finite Volume Discretization

In Athena++, equations 3.22 to 3.24 are discretized using the framework of the finite-volume method, which is well suited for handling conservation laws in CFD. The integration of equations of motion is performed on a 3D Cartesian grid. The computational flow domain with dimensions  $(L_x, L_y, L_z)$  along the spatial coordinates  $(x, y, z)$  is discretized into  $(n_x, n_y, n_z)$  finite-volume cells in each respective direction. Each cell, identified by indices  $(i, j, k)$ , is considered to have its center at a position denoted by  $(x_i, y_j, z_k)$ .

For clarity in describing the numerical methodology, it is assumed that the grid cells are uniform in size, with cell widths given by  $\delta x = L_x/n_x$ ,  $\delta y = L_y/n_y$ , and  $\delta z = L_z/n_z$ . Time is discretized independently into non-uniform  $n$  time steps between an initial time  $t_0$  and a final time  $t_f$ , with each time step denoted by  $\delta t$ , which may vary to satisfy stability or accuracy requirements.

Working with discretization based on the integral form of equations is particularly advantageous for simulating flows that involve shocks and discontinuities ([32], [33]). This approach contrasts with the differential form, which may not be robust under such conditions. Equation 3.29 is first integrated over the volume of a control cell. Applying the divergence theorem converts volume integrals of divergences of fluxes into surface integrals, and a subsequent integration over the time interval  $\delta t$  yields the discrete evolution equations (Equation 3.36).

$$\frac{\mathbf{U}_{i,j,k}^{n+1} - \mathbf{U}_{i,j,k}^n}{\delta t} = - \frac{\mathbf{F}_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} - \mathbf{F}_{i-\frac{1}{2},j,k}^{n+\frac{1}{2}}}{\delta x} - \frac{\mathbf{G}_{i,j+\frac{1}{2},k}^{n+\frac{1}{2}} - \mathbf{G}_{i,j-\frac{1}{2},k}^{n+\frac{1}{2}}}{\delta y} - \frac{\mathbf{H}_{i,j,k+\frac{1}{2}}^{n+\frac{1}{2}} - \mathbf{H}_{i,j,k-\frac{1}{2}}^{n+\frac{1}{2}}}{\delta z} \quad (3.36)$$

The conserved variables are volume-averaged quantities.

$$\mathbf{U}_{i,j,k}^n = \frac{1}{\delta x \delta y \delta z} \int_{z-\frac{1}{2}}^{z+\frac{1}{2}} \int_{y-\frac{1}{2}}^{y+\frac{1}{2}} \int_{x-\frac{1}{2}}^{x+\frac{1}{2}} \mathbf{U}(x, y, z, t^n) dx dy dz \quad (3.37)$$

The notations  $x_{i+1/2}$  and  $x_{i-1/2}$  denote the locations of the interfaces along  $x$ -direction between the cells centered respectively at  $x_{i+1}$  and  $x_i$ , and at  $x_{i-1}$  and  $x_i$ . Similarly, the

other notations denote interfaces between the cells located along the other two directions.

The fluxes  $\mathbf{F}$ ,  $\mathbf{G}$  and  $\mathbf{H}$  are evaluated on the faces of each computational grid cell, oriented normal to those respective faces. The superscript  $n + 1/2$  associated with the flux terms indicates a temporal average over the interval, rather than representing the flux at the specific time  $t = n + 1/2$ . These fluxes represent both time- and area-averaged quantities and are expressed as follows:

$$\mathbf{F}_{i+\frac{1}{2},j,k}^{n+\frac{1}{2}} = \frac{1}{\delta y \delta z \delta t} \int_{t^n}^{t^{n+1}} \int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \mathbf{F}(x_{i+\frac{1}{2}}, y, z, t) dy dz dt \quad (3.38)$$

$$\mathbf{G}_{i,j+\frac{1}{2},k}^{n+\frac{1}{2}} = \frac{1}{\delta x \delta z \delta t} \int_{t^n}^{t^{n+1}} \int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{G}(x, y_{j+\frac{1}{2}}, z, t) dx dz dt \quad (3.39)$$

$$\mathbf{H}_{i,j,k+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{\delta x \delta y \delta t} \int_{t^n}^{t^{n+1}} \int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \mathbf{H}(x, y, z_{k+\frac{1}{2}}, t) dx dy dt \quad (3.40)$$

Figure 3.5 illustrates a representative control volume (cell), where volumetric averages of conserved quantities are defined at cell centers. The face-centered time and area-averaged fluxes  $\mathbf{F}_{i+\frac{1}{2},j,k}$ ,  $\mathbf{G}_{i,j+\frac{1}{2},k}$  and  $\mathbf{H}_{i,j,k+\frac{1}{2}}$  at the right faces of the computational cell are also shown.

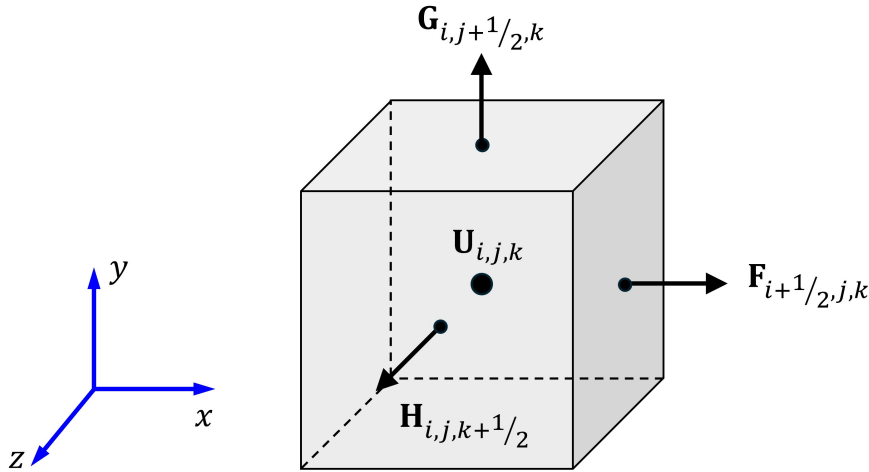


Figure 3.5: Volume-averaged conserved variables  $\mathbf{U}$  and time- and area-averaged components of the fluxes of  $\mathbf{U}$

Equations 3.36 through 3.40 are mathematically exact, with no approximations introduced up to this point, considering the limit as  $\delta t$  approaches zero. Within the finite-volume framework, the implementation of a numerical algorithm necessitates stable and precise approximations of the time- and area-averaged fluxes governed by these equations. In the

context of high-speed hydrodynamic and magnetohydrodynamic flows, higher-order numerical schemes tend to enhance accuracy predominantly in regions of smooth flow; however, they offer limited benefits in the presence of shocks or discontinuities and present increased complexity when coupled with AMR. In this study, for simple two-dimensional benchmark flows, a second-order accurate discretization in space and time was employed. Based on this assessment, higher-order schemes were subsequently utilized for the remaining simulations.

### 3.4.2 Godunov Method and Approximate Riemann Solvers

In the finite-volume formulation, the solution is stored as cell-averaged conserved quantities located at the centers of computational cells. These values are treated as uniform within each cell, producing a piecewise-constant representation of the solution across the domain. At the interfaces between cells, this representation creates discontinuities in the solution profile. In flows where such discontinuities naturally arise such as shock waves, contact discontinuities, or rarefaction waves it becomes essential to evolve the solution in a way that preserves the correct physical behavior. Figure 3.6 illustrates this situation, showing discontinuities at cell interfaces for some initial time level.

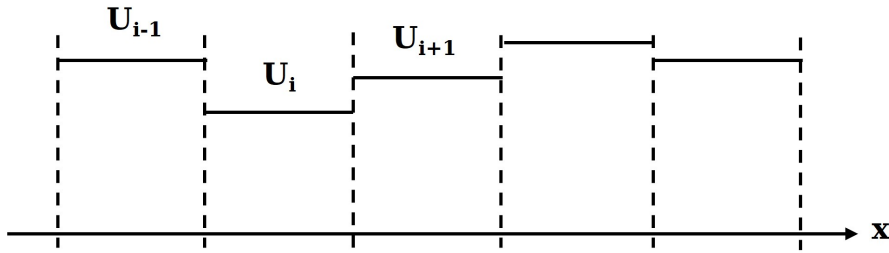


Figure 3.6: Schematic representation of cell-averaged states and discontinuities at cell interfaces

This challenge is addressed by the Godunov method [32], in which the evolution at each cell interface is obtained by solving a local Riemann problem: the initial-value problem defined by two uniform states separated by a discontinuity. The solution of this problem predicts how the discontinuity evolves into a set of waves shock waves, contact discontinuities, and/or rarefaction waves that transport mass, momentum, and energy between neighboring cells. Figure 3.7 provides a schematic of the wave pattern that may develop after a finite time.

In principle, one could use the exact Riemann solution at each interface. However, computing the exact solution at every timestep is computationally expensive and, in many cases, unnecessarily detailed. For updating the finite-volume solution, only the state at the cell interface is required to compute the numerical fluxes. This motivates the use of approximate Riemann solvers, which capture the essential wave structure with significantly reduced computational cost.



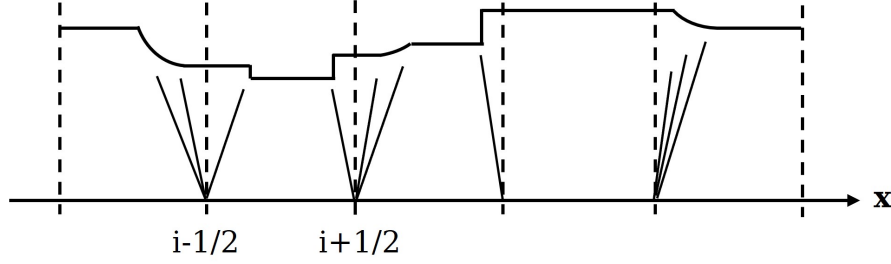


Figure 3.7: Schematic representation of a possible wave structure evolved from an initial discontinuity after a finite time

For hydrodynamic flows, Athena++ provides several approximate Riemann solvers, including Roe, Harten–Lax–van Leer (HLL), and Harten–Lax–van Leer–Contact (HLLC), as well as a low-dissipation variant of HLLC [9, 8]. In this study, the HLLC solver is employed. As an extension of the original HLL solver, the HLLC method restores the missing contact and shear waves, thereby improving accuracy in flows with strong discontinuities while maintaining robustness. Comprehensive derivations of the HLL and HLLC solvers are presented in Toro’s text [32], and these methods are widely adopted in CFD.

Even with an accurate Riemann solver, the piecewise-constant representation of the solution limits the method to first-order spatial accuracy. To achieve higher-order accuracy in smooth regions, Athena++ applies spatial reconstruction schemes—such as piecewise linear (PLM) or piecewise parabolic (PPM)—to reconstruct more accurate interface states before solving the Riemann problem. These reconstruction techniques are discussed in the following subsection.

### 3.4.3 Spatial Reconstruction – PLM and PPM

The first-order Godunov scheme assumes each cell contains a constant average value, producing piecewise-constant profiles across the grid. While robust, this representation is overly diffusive, especially in smooth regions, as discontinuities exist at every cell interface. To reduce dissipation and increase accuracy, spatial reconstruction is applied to recover a higher-order profile within each cell from the discrete averages.

In Athena++, this is accomplished by reconstructing left ( $\mathbf{U}^L$ ) and right ( $\mathbf{U}^R$ ) states at each interface for the Riemann solver.

#### 3.4.3.1 Piecewise Linear Method (PLM)

PLM assumes the solution varies linearly within each cell, with slopes computed from neighboring averages and limited to avoid spurious oscillations near discontinuities (Figure 3.8).

A generic limited slope  $\sigma_i$  can be written as

$$\sigma_i = \phi \left( \frac{U_i - U_{i-1}}{\Delta x}, \frac{U_{i+1} - U_i}{\Delta x} \right) \quad (3.41)$$

where  $\phi$  is a slope limiter function (e.g., minmod, van Leer) [32, 33]. The interface states are then

$$U_{i+1/2}^L = U_i + \frac{\sigma_i \Delta x}{2}, \quad U_{i-1/2}^R = U_i - \frac{\sigma_i \Delta x}{2} \quad (3.42)$$

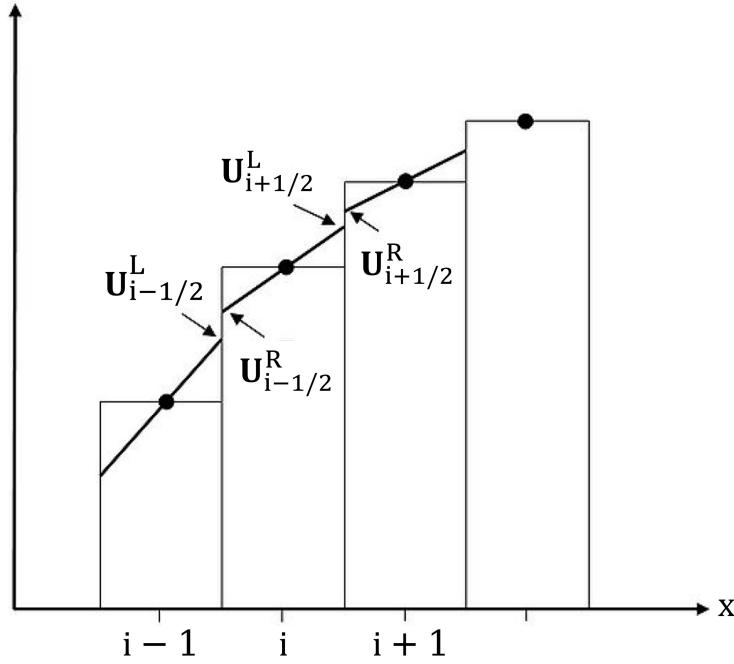


Figure 3.8: Piecewise linear reconstruction of states at cell interfaces

### 3.4.3.2 Piecewise Parabolic Method (PPM)

PPM reconstructs a parabolic profile in each cell, yielding third-order spatial accuracy in smooth regions. A general parabolic form is:

$$U(x) = U_i + a_i \xi + b_i \xi^2, \quad \xi \in \left[ -\frac{1}{2}, \frac{1}{2} \right] \quad (3.43)$$

where coefficients  $a_i$  and  $b_i$  are chosen to match neighboring cell averages and enforce monotonicity constraints. PPM significantly reduces numerical dissipation compared to PLM, at the cost of additional computational effort.

Both PLM and PPM aim to provide more accurate interface states to the Riemann solver (HLLC in this study), which then computes the time- and area-averaged fluxes used in the finite-volume update.

### 3.4.4 Time Integration – Runge–Kutta Methods

The spatial reconstruction techniques described above (PLM and PPM) provide higher-order interface states for the Riemann solver. To maintain overall accuracy in both space and time, the temporal evolution of the conserved variables  $\mathbf{U}$  must also be integrated using a time-stepping scheme of matching order. In Athena++, explicit Runge–Kutta (RK) methods are employed for this purpose.

The semi-discrete form of the finite-volume equations can be written as

$$\frac{d\mathbf{U}_{i,j,k}}{dt} = \mathcal{L}(\mathbf{U}_{i,j,k}) \quad (3.44)$$

where  $\mathcal{L}(\mathbf{U}_{i,j,k})$  is the spatial discretization operator, representing the discrete divergence of numerical fluxes

$$\mathcal{L}(\mathbf{U}_{i,j,k}) = -\frac{\mathbf{F}_{i+\frac{1}{2},j,k} - \mathbf{F}_{i-\frac{1}{2},j,k}}{\Delta x} - \frac{\mathbf{G}_{i,j+\frac{1}{2},k} - \mathbf{G}_{i,j-\frac{1}{2},k}}{\Delta y} - \frac{\mathbf{H}_{i,j,k+\frac{1}{2}} - \mathbf{H}_{i,j,k-\frac{1}{2}}}{\Delta z} \quad (3.45)$$

**Two-stage Runge–Kutta method (RK2):** The RK2 method is used to achieve second-order temporal accuracy. In this study, RK2 is employed in combination with the PLM reconstruction.

First intermediate stage:

$$\mathbf{U}^{(1)} = \mathbf{U}^n + \Delta t \mathcal{L}(\mathbf{U}^n) \quad (3.46)$$

Update the solution:

$$\mathbf{U}^{n+1} = \frac{1}{2}\mathbf{U}^n + \frac{1}{2} \left[ \mathbf{U}^{(1)} + \Delta t \mathcal{L}(\mathbf{U}^{(1)}) \right] \quad (3.47)$$

**Four-stage Runge–Kutta method (RK4):** The RK4 method is used to achieve fourth-order temporal accuracy. In this study, RK4 is employed in combination with the PPM reconstruction.

$$\mathbf{U}^{(1)} = \mathbf{U}^n + \frac{\Delta t}{2} \mathcal{L}(\mathbf{U}^n) \quad (3.48)$$

$$\mathbf{U}^{(2)} = \mathbf{U}^n + \frac{\Delta t}{2} \mathcal{L}(\mathbf{U}^{(1)}) \quad (3.49)$$

$$\mathbf{U}^{(3)} = \mathbf{U}^n + \Delta t \mathcal{L}(\mathbf{U}^{(2)}) \quad (3.50)$$

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \frac{\Delta t}{6} \left[ \mathcal{L}(\mathbf{U}^n) + 2\mathcal{L}(\mathbf{U}^{(1)}) + 2\mathcal{L}(\mathbf{U}^{(2)}) + \mathcal{L}(\mathbf{U}^{(3)}) \right] \quad (3.51)$$

The choice of time integrator reflects a trade-off between computational cost and accuracy, as higher-order schemes require additional stages per timestep but can substantially reduce temporal discretization errors.

### 3.4.5 Adaptive Mesh Refinement

In Athena++, the computational domain is represented as a uniformly divided rectangular region, referred to as the Mesh. This domain is further subdivided into a regular array of subvolumes called MeshBlocks. Each MeshBlock contains a specified number of mesh cells and store the discrete values of the dependent variables in the form of an  $N$ -dimensional array. While the number of cells per MeshBlock is arbitrary, it must be identical across all MeshBlocks, and the decomposition of the Mesh into MeshBlocks is likewise arbitrary.

For uniform-grid calculations, MeshBlocks provide a natural means of parallelization through domain decomposition. Typically, the total number of MeshBlocks is chosen to match the number of physical processors available, although this is not strictly required. For serial calculations on a uniform grid, a single MeshBlock containing all the mesh cells is sufficient. The root mesh is assigned a refinement level index of 0, and refinement (either static or dynamic) is applied at the level of MeshBlocks. If  $n$  is the maximum refinement level, then the grid resolution at the finest level is  $2^{(n-1)}$  times finer than the root level.

Adaptive mesh refinement (AMR) builds upon this structure by dynamically adjusting grid resolution to the evolving flow field. Instead of employing a uniformly fine mesh across the entire computational domain, AMR refines the mesh locally in regions of interest while maintaining coarser resolution elsewhere. This allows multiscale structures to be resolved efficiently while keeping the overall grid count manageable. Regions can also be derefined when high resolution is no longer necessary, ensuring optimal use of computational resources.

Broadly, three major approaches to AMR are employed:

1. Patch-based refinement: This method was proposed by Berger and Oliger [20] in which finer mesh patches are overlaid onto regions of interest (3.9a).
2. Block-based refinement: The domain is partitioned into blocks, and individual blocks are refined or coarsened based on refinement criteria (3.9b).
3. Cell-based refinement: The refinement is applied directly to individual cells, resulting in a highly flexible grid hierarchy. While efficient in terms of resolution, this method poses significant challenges for parallelization and load balancing (3.9c).

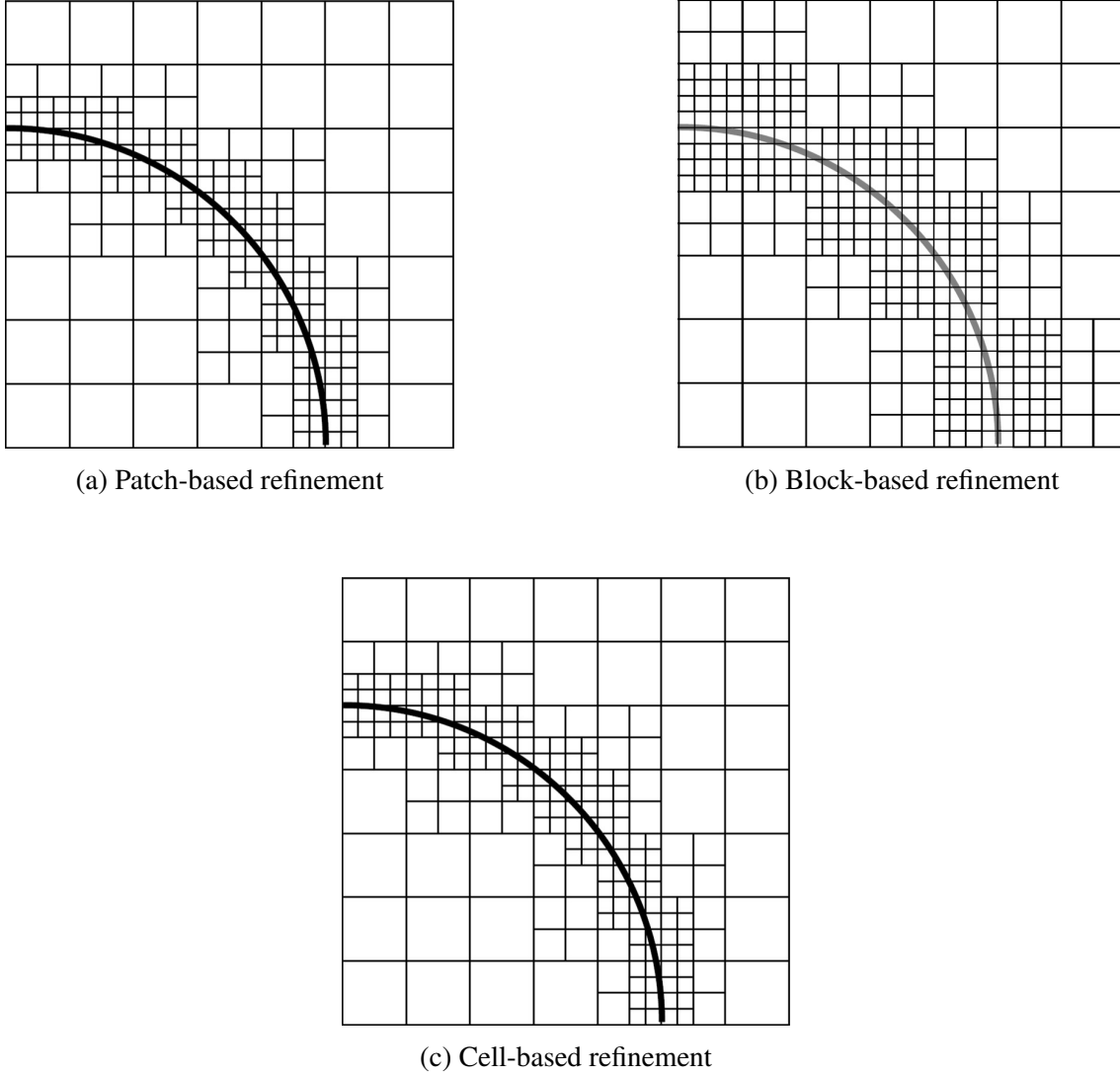


Figure 3.9: Adaptive mesh refinement methods (adapted from [34])

Athena++ adopts a block-based AMR framework, which balances resolution efficiency with parallel scalability. Unlike patch-based methods, which are more difficult to load balance on distributed-memory systems, the block-based approach integrates effectively with domain decomposition strategies and ensures robust performance on modern high-performance computing architectures.

#### 3.4.5.1 Flux Correction at Refinement Boundaries

In simulations with static mesh refinement (SMR) or AMR, conservation across interfaces of MeshBlocks at different refinement levels must be preserved. This requires the area-integrated flux through a coarse-cell face to equal the sum of the fluxes through the corresponding fine-cell faces. To enforce this condition, Athena++ implements a flux correction procedure [21], in which the flux on a coarse face (blue) is replaced by the more accurate fluxes from the fine faces (red) (Figure 3.10). This correction ensures global

conservation of mass, momentum, and energy across refinement boundaries in three dimensions.

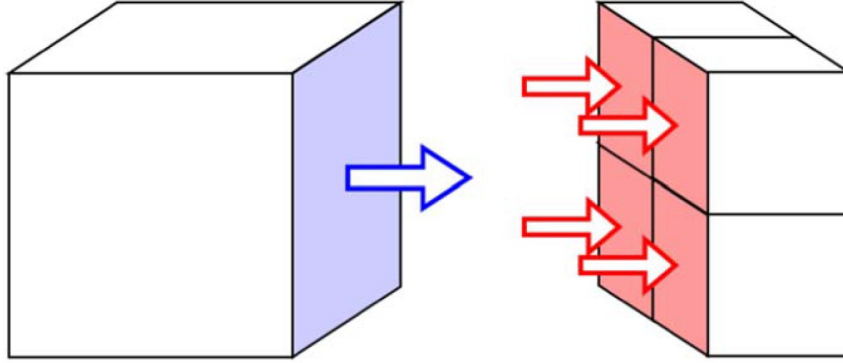


Figure 3.10: Flux correction at coarse-fine interfaces in AMR (adapted from [8])

### 3.4.5.2 Time Stepping with AMR

In finite-volume schemes for hyperbolic conservation laws, the timestep is constrained by the Courant–Friedrichs–Lewy (CFL) condition, which depends on the maximum signal speed in the flow. For hydrodynamics, this speed is  $|\vec{v}| + c$ , where  $\vec{v}$  is the fluid velocity and  $c$  is the sound speed. In magnetohydrodynamics, it is  $|\vec{v}| + c_f$ , where  $c_f$  is the fast magnetosonic speed. In three dimensions, the condition can be written as

$$\Delta t \leq C_{\text{CFL}} \min \left( \frac{\Delta x}{|\vec{v}_x| + c}, \frac{\Delta y}{|\vec{v}_y| + c}, \frac{\Delta z}{|\vec{v}_z| + c} \right) \quad (3.52)$$

where  $\Delta x$ ,  $\Delta y$ , and  $\Delta z$  are the grid spacings in each direction, and  $C_{\text{CFL}} < 1$  is the Courant number ensuring stability.

On an AMR mesh, the stable timestep therefore scales with the spatial resolution at each refinement level. A common approach is adaptive time stepping, where level  $n$  advances with a timestep  $2^n$  times smaller than that of the root level ( $n = 0$ ). This approach maintains stability but requires interpolation in both space and time across fine–coarse interfaces to enforce flux conservation.

In Athena++, a simplified approach is adopted. All refinement levels are advanced with the same fixed timestep. This avoids the complexities of synchronizing multiple timesteps and eliminates the need for temporal interpolation at refinement boundaries, while remaining efficient for parallel computations. The rationale for this choice is described in detail by Stone et al. [8].

# Chapter 4

## Results and Discussion

This section presents a sequence of test problems simulated using Athena++ to validate the implementation of viscous, incompressible hydrodynamic flows. The progression begins with simple, steady-state laminar cases such as plane channel flow and the two-dimensional lid-driven cavity, which serve as baseline checks for solver accuracy and for verifying the correctness of custom ‘wall’ boundary conditions (as Athena++ does not natively support them).

Once these foundational steady-state laminar flow cases confirm the reliability of the implementation, the focus shifts to more dynamic regimes. The two- and three-dimensional Taylor–Green vortex (TGV) problems are simulated over a range of Reynolds numbers, spanning laminar decay, transitional vortex breakdown, and turbulence evolution. For the three-dimensional cases, under-resolved DNS is performed in regimes exhibiting transition to turbulence, providing an additional evaluation of solver performance under reduced resolution. Such multidimensional cases are particularly relevant for future hydrodynamic and MHD applications, where accurate representation of turbulent dynamics is essential. As part of this work, an AMR capability is implemented in Athena++ with refinement criteria tailored to the flow physics, and the TGV simulations serve to assess its effectiveness in capturing the required flow features while improving computational efficiency relative to uniform meshes.

Unless stated otherwise, the simulations use the same numerical configuration, summarized in Table 4.1.

Table 4.1: Numerical settings used in Athena++ simulations

| Parameter              | Value / Description                               |
|------------------------|---------------------------------------------------|
| Courant number         | 0.4                                               |
| Riemann solver         | HLLC (for hydrodynamic fluxes)                    |
| Time integration       | Second-order Runge–Kutta method                   |
| Spatial reconstruction | Piecewise linear method (PLM)                     |
| Ghost zones (cells)    | 2 (consistent with second-order spatial accuracy) |

The Courant number of 0.4 satisfies the CFL stability condition. Hydrodynamic fluxes are computed using the HLLC Riemann solver, with time integration performed via a second-order Runge–Kutta scheme. Spatial reconstruction uses the PLM, and two ghost cells are maintained to ensure the nominal second-order accuracy of the method. For the PPM reconstruction, which achieves third-order spatial accuracy, four ghost cells are employed to provide the required stencil width.



## 4.1 Two-Dimensional Channel Flow

As an initial validation case, a two-dimensional incompressible laminar channel flow is considered. This test case involves steady, pressure-driven flow between two parallel plates and admits an analytical solution—a parabolic velocity profile—which makes it well-suited for assessing numerical accuracy and verifying the implementation of boundary conditions.

### 4.1.1 Flow Domain, Fluid Properties and Reynolds Number

The computational domain has a height  $H = 1$  and a length  $L = 4H = 4$ , with the flow of fluid having properties considered as summarized in Table 4.2. The fluid density  $\rho = 1.0$ , kinematic viscosity  $\nu = 1.0$ , and an adiabatic index  $\gamma = 1.001$ , chosen to effectively enforce isothermal conditions.

Table 4.2: Fluid properties considered for 2D channel flow

| Property of Fluid             | Value                   |
|-------------------------------|-------------------------|
| Fluid density ( $\rho$ )      | 1.0                     |
| Kinematic viscosity ( $\nu$ ) | 1.0                     |
| Adiabatic index ( $\gamma$ )  | 1.001 (near-isothermal) |

The flow is characterized by the Reynolds number  $Re = 10$ . Considering height of the channel  $H$  as characteristic length, the Reynolds number can be expressed as

$$Re = \frac{U_0 H}{\nu} \quad (4.1)$$

The hydrodynamic entrance length, also referred to as the development length, is the distance required for a flow entering uniformly to become fully developed within a channel or pipe. For laminar flows, the entrance length is approximately related to the Reynolds number [22, 23], and the relation is given as

$$L_e \approx 0.06 Re_{D_h} D_h \quad (4.2)$$

where  $D_h$  is the hydraulic diameter of the geometry, and  $Re_{D_h}$  is the Reynolds number based on the average velocity and  $D_h$ . The hydraulic diameter is given by

$$D_h = \frac{4A}{P} \quad (4.3)$$

where  $A$  is the cross-sectional area of flow and  $P$  is the wetted perimeter of the section.

For the Reynolds number considered in this study, the entrance length is estimated to be approximately  $L_e \approx 1.2$ . Therefore, the chosen domain length is sufficient to ensure that the flow becomes fully developed before reaching the outlet.

#### 4.1.2 Boundary Conditions and Reference Pressure

No-slip wall boundary conditions are applied at the top and bottom boundaries using a user-defined implementation (Table 4.3). A uniform inflow condition is imposed at the inlet, while an outflow boundary condition is applied at the outlet.

Table 4.3: Boundary conditions and reference pressure used in the simulations

| Boundary            | Type         | Condition / Description                                                |
|---------------------|--------------|------------------------------------------------------------------------|
| Inlet (left)        | Inflow       | $u = 10, v = 0$ ; based on $Re = 10$                                   |
| Top and Bottom wall | No-slip wall | $u = 0, v = 0$                                                         |
| Outlet (right)      | Outflow      | $\frac{\partial u}{\partial x} = 0, \frac{\partial v}{\partial x} = 0$ |
| Reference pressure  | –            | $p_{ref} = 3 \times 10^6$ (to enforce low Mach number)                 |

A high value of the reference pressure is specified to maintain a very low Mach number (M), thereby ensuring that compressibility effects are negligible and that the flow remains effectively incompressible. The reference pressure can be calculated using the relation,

$$p_{ref} = \frac{U_0^2}{M^2} \cdot \frac{\rho}{\gamma} \quad (4.4)$$

where,  $U_0$  is the average velocity at the inlet. The  $U_0$  has been calculated using the Reynolds number considered for the study.

#### 4.1.3 Implementation of Boundary Conditions in Athena++

Athena++ natively provides an outflow boundary condition. In this study, additional user-defined routines were implemented to populate ghost cells at the inlet and solid walls with physically consistent primitive variables. At the inlet, density is fixed, pressure is copied from the first interior cell, and velocity is prescribed as  $u = U_0, v = 0$ . At solid walls, density is fixed, pressure is copied from the first interior cell, and velocities are extrapolated to enforce the no-slip condition  $u = v = 0$ . The complete ghost-cell filling strategy is summarized in Table 4.4.

#### 4.1.4 Grid Convergence and Error Analysis

A grid refinement study was carried out using Richardson extrapolation and the Grid Convergence Index (GCI) method of Roache [35] to quantify discretization error and verify

Table 4.4: Implementation of primitive variable boundary conditions in ghost cells

| Boundary | Density  | Pressure                       | Velocity                                                             |
|----------|----------|--------------------------------|----------------------------------------------------------------------|
| Inlet    | Constant | Same as in first interior cell | Constant ( $u = U_0, v = 0$ )                                        |
| Walls    | Constant | Same as in first interior cell | Extrapolated based on averaging so that $u = v = 0$ at wall location |

convergence. Simulations were performed on three uniformly refined grids— $128 \times 32$ ,  $256 \times 64$ , and  $512 \times 128$ —with a constant refinement ratio. The resulting consistency ratio of 1.003, being effectively unity, confirms asymptotic convergence and demonstrates grid-independent behavior, thereby validating the robustness of the discretization scheme.

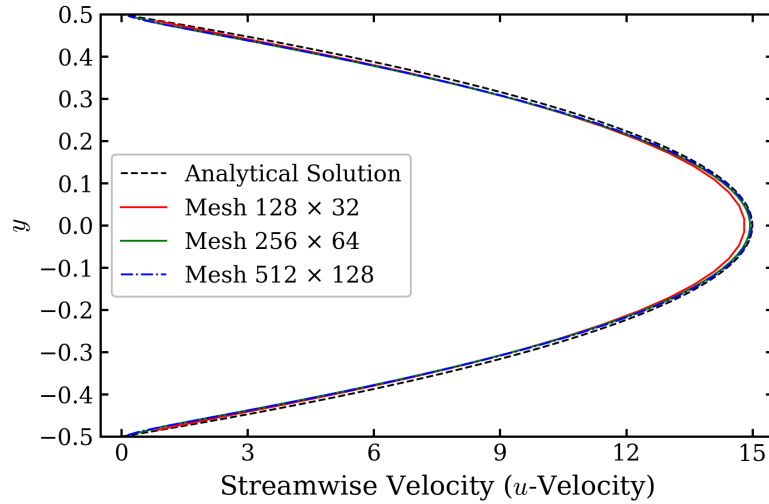


Figure 4.1: Comparison of numerical results with analytical solution

Figure 4.1 presents the streamwise velocity profiles at the outlet for three grid resolutions, compared against the analytical solution [22]. The profiles exhibit the characteristic parabolic distribution of fully developed channel flow, confirming the correct implementation of wall boundary conditions. Progressive grid refinement improves agreement with the analytical solution, consistent with the findings of the GCI analysis, although a finite deviation persists even at the finest resolution, attributable to discretization error.

As part of the error analysis, the pressure drop in the fully developed region of the channel flow was examined. The pressure gradient becomes constant in this region, and the pressure drop across a section of unit length was computed as approximately 121, which is in close agreement with the analytical value of 120. For the finest grid resolution of  $512 \times 128$ , the relative error in the pressure drop remains below 1.0%, indicating accurate

prediction of the pressure field.

Additionally, the absolute error in axial velocity at the channel center was evaluated (Figure 4.2). The results show a systematic decrease in error with grid refinement, consistent with the expected order of accuracy of the scheme. This confirms that the numerical method yields increasingly accurate velocity predictions as the resolution is enhanced.

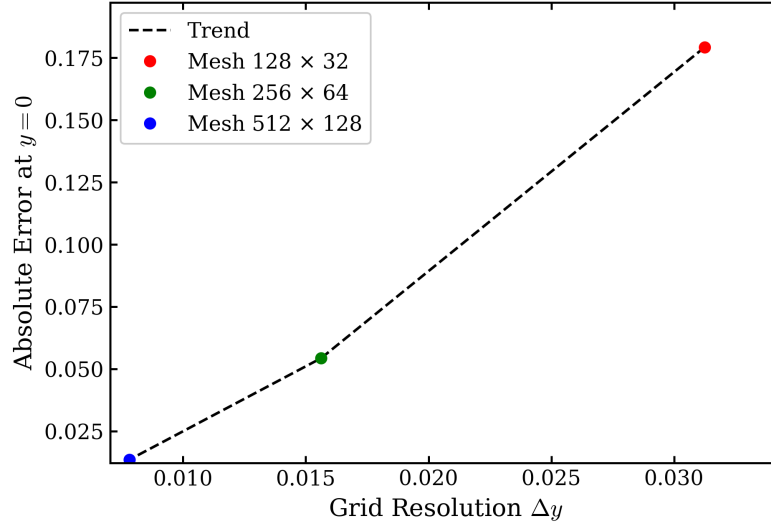


Figure 4.2: Absolute error at  $y = 0$  vs grid resolution

#### 4.1.5 Numerical Solution from the Finest Grid

Figure 4.3 illustrates the contours of the magnitude of streamwise velocity in the channel flow. The streamwise velocity exhibits a characteristic parabolic distribution in the cross-stream direction (Figure 4.1), with maximum values at the channel centerline and nearly zero velocity at the walls due to the no-slip boundary condition. Along the streamwise direction, the profile becomes fully developed, indicating a steady and fully developed laminar flow. The contours are closely packed near the channel walls and gradually spread out toward the centerline, indicating steep velocity gradients near the boundaries and a more uniform flow in the core region.

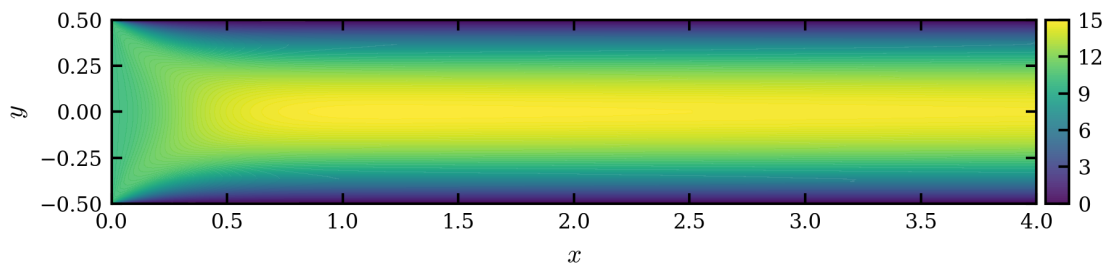


Figure 4.3: Contours of streamwise velocity colored by magnitude

## 4.2 Two-Dimensional Lid Driven Cavity

The two-dimensional lid-driven cavity is a classical benchmark problem in CFD, widely used to assess the accuracy and stability of numerical solvers. It involves the incompressible flow of a Newtonian fluid inside a square cavity, driven solely by the uniform motion of the top lid while the remaining walls are stationary. Despite its geometric simplicity, the problem exhibits complex recirculating flow patterns and well-documented reference solutions, making it a stringent test for validating the performance of Athena++.

### 4.2.1 Flow Domain, Fluid Properties and Reynolds Number

Figure 4.4 shows a square-shaped cavity situated in the  $x - y$  plane, with each side of length  $L$ . The origin of the coordinate system,  $(x, y) = (0, 0)$ , is positioned at the geometric center of the cavity. The upper boundary (lid) of the cavity moves with a uniform velocity  $U$  in the  $x$  direction, while the remaining walls remain stationary.

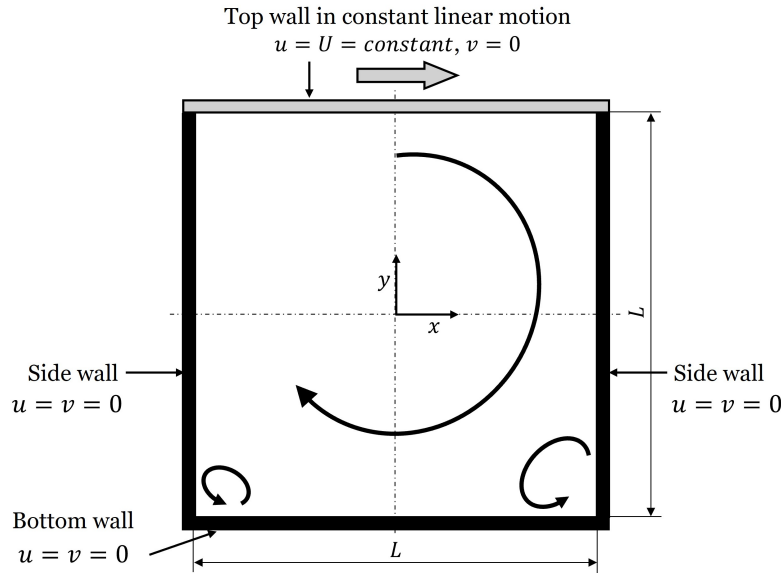


Figure 4.4: Schematic of lid driven cavity with velocity boundary conditions

The fluid properties - density ( $\rho$ ), kinematic viscosity ( $\nu$ ), and temperature ( $T$ )) - are assumed to be constant throughout the domain. The flow generated by the motion of the lid is considered steady, incompressible, and unaffected by gravitational forces. The fluid properties used for the lid-driven cavity (LDC) simulation is identical to those employed for the channel flow simulations, as listed in Table 4.2.

To enable comparison with the widely used benchmark results of Ghia et al. [24], a Reynolds number of  $Re = 400$  is considered for the LDC flow simulation. In this context, the Reynolds number is defined based on the cavity height and the velocity of the moving lid. Accordingly, the top wall is assigned a uniform velocity of  $u = 400$ , which corresponds to the specified Reynolds number when combined with the chosen fluid properties.

### 4.2.2 Boundary Conditions and Reference Pressure

The boundary conditions applied and the reference pressure used are summarized in Table 4.6.

Table 4.5: Boundary conditions and reference pressure used for LDC simulations

| Boundary              | Type            | Description                      |
|-----------------------|-----------------|----------------------------------|
| Side and bottom walls | Stationary wall | $u = 0, v = 0$                   |
| Top wall              | Moving wall     | $u = 400, v = 0$                 |
| Reference pressure    | –               | $p_{\text{ref}} = 3 \times 10^6$ |

### 4.2.3 Implementation of Boundary Conditions in Athena++

The specification of primitive variables—density, pressure, and velocity—in the ghost cells is carefully formulated to ensure physical consistency at the boundaries and to maintain the accuracy of the simulation. The treatment of velocity condition is done depending on whether the wall boundary is stationary or moving. A summary of the implementation strategy adopted within the Athena++ framework is provided in Table 4.6.

Table 4.6: Implementation of primitive variable boundary conditions in ghost cells

| Boundary        | Density  | Pressure               | Velocity                                                 |
|-----------------|----------|------------------------|----------------------------------------------------------|
| Stationary wall | Constant | Mirrored from interior | Extrapolated to ensure wall velocity is zero ( $u = 0$ ) |
| Moving wall     | Constant | Mirrored from interior | Extrapolated to enforce lid velocity ( $u = 400$ )       |

### 4.2.4 Numerical Solution

The lid-driven cavity simulation is carried out using a uniform grid of  $128 \times 128$  cells. To evaluate the precision of the numerical results, the velocity profiles are extracted along the centerlines of the cavity and compared with the reference solution of Ghia et al. [24].

#### 4.2.4.1 Velocity Profiles at the Midlines of Cavity

The horizontal velocity ( $u$ ) along the vertical centerline at  $x = 0$ , and the vertical velocity ( $v$ ) along the horizontal centerline at  $y = 0.5$  are shown. At a Reynolds number of  $\text{Re} = 400$ , the overall trend of the centerline velocity profiles is in excellent agreement with the benchmark solution. The horizontal velocity profile accurately captures the maximum

velocity near the moving lid and the shear layer development across the cavity height. Similarly, the vertical velocity profile effectively reproduces the flow reversal and the location of the velocity extrema. This close correspondence with the benchmark data serves as a strong validation of the implemented boundary conditions and demonstrates the robustness and accuracy of the Athena++ solver for simulating incompressible flows.

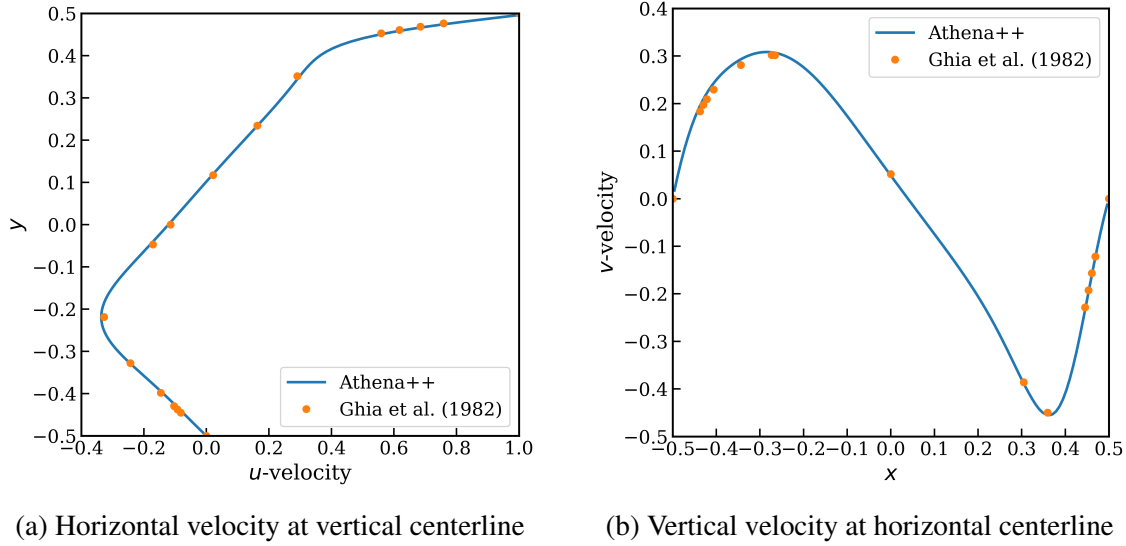


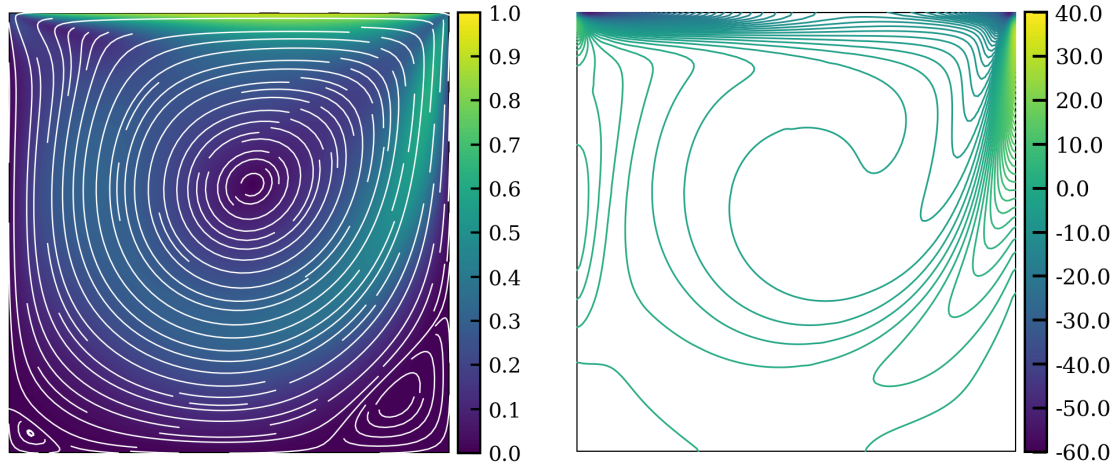
Figure 4.5: Velocity profiles in lid-driven cavity at  $Re = 400$

#### 4.2.4.2 Velocity Contours with Streamlines

Figure 4.6a presents the velocity magnitude contours and streamlines obtained from the simulation results. The contours indicate that the highest velocities occur near the moving lid, while lower velocities dominate the recirculating interior. The flow field exhibits a well-defined primary vortex in the center and secondary vortices in the lower corners. The structure and location of these vortices show strong qualitative agreement with the reference solution of Ghia et al. [24], confirming the accuracy of the numerical solution in capturing the key features of lid-driven cavity flow at  $Re = 400$ .

#### 4.2.4.3 Vorticity Contours and Vorticity at Moving Wall

Figure 4.12 presents the vorticity contours obtained from the numerical solution of the cavity flow. Regions of high and low vorticity are clearly resolved. Strong rotational effects are observed near the moving lid, where shear is most intense. Areas with steep vorticity gradients, indicated by densely packed contour lines, appear prominently near the cavity walls and corners. Overall, the vorticity distribution agrees well with established benchmark solution [24], demonstrating the accuracy and reliability of the numerical results.



(a) Streamlines and contours of velocity magnitude

(b) Vorticity contours colored by magnitude

Figure 4.6: Streamlines with velocity magnitude and vorticity contours at  $Re = 400$

Overall, the vorticity values calculated from the Athena++ solution match the benchmark very well along the length of the moving lid, with most points differ slightly. Between  $x \approx 0.3-0.5$ , Athena++ tends to over-predict the vorticity by a few units and again near  $x \approx 0.9-1.0$ , to under-predict the vorticity by a few units. Larger discrepancies may appear close to the sidewalls, where the numerical curve shows sharp end spikes due to corner singularities, sensitivity to grid resolution and boundary-condition treatment when computing vorticity from velocity gradients. Taken together, the comparison indicates grid-converged agreement in the core region, with possible errors confined to the very near-wall zones.

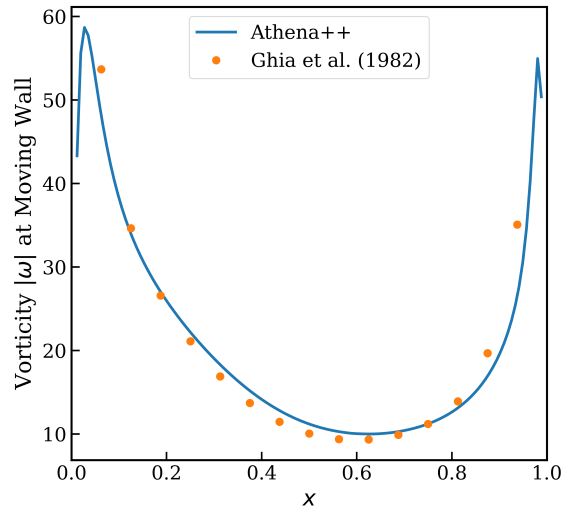


Figure 4.7: Vorticity at the lid of cavity at  $Re = 400$



### 4.3 Two-Dimensional Taylor-Green Vortex

To evaluate the AMR capabilities of Athena++, the two-dimensional (2D) Taylor-Green vortex (TGV) problem has been used as a preliminary test in this work. This classical problem provides an idealized setup for examining the dissipation of kinetic energy as a result of viscous effects, which makes it suitable for validating both the hydrodynamic flow solver and the performance of AMR under smooth and decaying flow conditions.

#### 4.3.1 Flow Domain, Fluid Properties and Reynolds Number

A 2D TGV flow field is considered in a periodic square domain, where  $(x, y) \in [0, 2\pi] \times [0, 2\pi]$ , with the initial velocity conditions given by,

$$u(x, y, 0) = U_0 \sin(x) \cos(y) \quad (4.5)$$

$$v(x, y, 0) = -U_0 \cos(x) \sin(y) \quad (4.6)$$

where  $U_0$  is the initial velocity amplitude.

The initial pressure field for incompressible flow (derived from the Poisson equation for pressure) is given by

$$p(x, y, 0) = p_{ref} - \frac{1}{4} (\cos(2x) + \cos(2y)) \quad (4.7)$$

where  $p_{ref}$  is the reference or background pressure.

These initial conditions create a smooth, incompressible flow field characterized by symmetric vortex structures that decay over time due to viscous dissipation. The analytical solution of the 2D TGV at time  $t$  is given by [25].

$$u(x, y, t) = U_0 \sin(x) \cos(y) e^{-2\nu t} \quad (4.8)$$

$$v(x, y, t) = -U_0 \cos(x) \sin(y) e^{-2\nu t} \quad (4.9)$$

Here,  $\nu$  is the kinematic viscosity of fluid.

The pressure field is given by,

$$p(x, y, t) = p_{ref} - \frac{1}{4} (\cos(2x) + \cos(2y)) e^{-4\nu t}. \quad (4.10)$$

The evolution of these vortices provides a stringent test for the resolution and accuracy of numerical schemes, particularly under AMR, where the flow features must be captured without excessive refinement elsewhere.

The Reynolds number (Re) for TGV is typically defined as

$$\text{Re} = \frac{U_0 L}{\nu} \quad (4.11)$$

For this study, a Reynolds number of  $\text{Re} = 3000$  is considered. The initial velocity amplitude is set to  $U_0 = 10.0$ . Taking the characteristic length as the side length of the flow domain,  $L = 2\pi$ , the corresponding kinematic viscosity is computed as  $\nu = 0.02094$ . The fluid properties employed in the simulations are summarized in Table 4.7.

Table 4.7: Fluid properties considered for 2D Taylor-Green vortex flow

| Property of Fluid             | Value   |
|-------------------------------|---------|
| Density ( $\rho$ )            | 1.0     |
| Kinematic viscosity ( $\nu$ ) | 0.02094 |
| Adiabatic index ( $\gamma$ )  | 1.6667  |

### 4.3.2 Boundary Conditions and Reference Pressure

Periodic boundary conditions (PBCs) are employed in the Taylor-Green vortex problem to simulate an unbounded, spatially repeating domain, ensuring that fluid structures leaving one boundary seamlessly re-enter from the opposite side. In Athena++, PBCs are implemented by copying values across opposing domain faces, with ghost cell zones automatically updated to maintain consistency across boundaries. This approach preserves the symmetry and conservation properties of the flow, and eliminates artificial boundary effects making it particularly well-suited for studying the problems involving vortex dynamics and the onset or development of turbulence.

The boundary conditions and reference pressure used are given in Table 4.8. The reference pressure as given by Equation 4.4, has been chosen such that the Mach number of the flow is estimated well below 0.001.

Table 4.8: Boundary conditions and reference pressure used for 2D TGV simulations

| Boundary            | Type     | Description/Value                                                                                                                                                                 |
|---------------------|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| All sides of domain | Periodic | $\phi(x_{\min}, y, t) = \phi(x_{\max}, y, t)$<br>$\phi(x, y_{\min}, t) = \phi(x, y_{\max}, t)$<br>where $\phi$ denotes density, pressure, velocity components and other variables |
| Reference pressure  | —        | $p_{\text{ref}} = 1 \times 10^6$                                                                                                                                                  |

### 4.3.3 Grid Convergence and Error Analysis

A grid refinement study was performed to assess the convergence and accuracy of the numerical solutions. Simulations were conducted on three successively refined grids— $128 \times 128$ ,  $256 \times 256$ , and  $512 \times 512$ —with a uniform refinement ratio of 2.0. The consistency ratio, computed from the grid convergence indices based on the mean kinetic energy  $E(t)$  in the domain at  $t = 20$ , was found to be 1.18, indicating that the solution lies within the asymptotic range of convergence.

The mean kinetic energy at time  $t$  is

$$E(t) = \frac{1}{|A|} \int_{\Omega} \frac{1}{2} (u(x,y,t)^2 + v(x,y,t)^2) dA \quad (4.12)$$

where  $\Omega$  is the 2D computational domain and  $|\Omega|$  its area.

Figure 4.8 presents the evolution of kinetic energy for all three grid resolutions, compared against the analytical solution. The results show that the  $512 \times 512$  grid yields a solution that closely matches the analytical profile. Additionally, the evolution of the  $L_2$  error over time confirms that the error is lowest for the finest mesh. For all grids, the error remains nearly constant throughout the simulation, indicating stable numerical behavior.

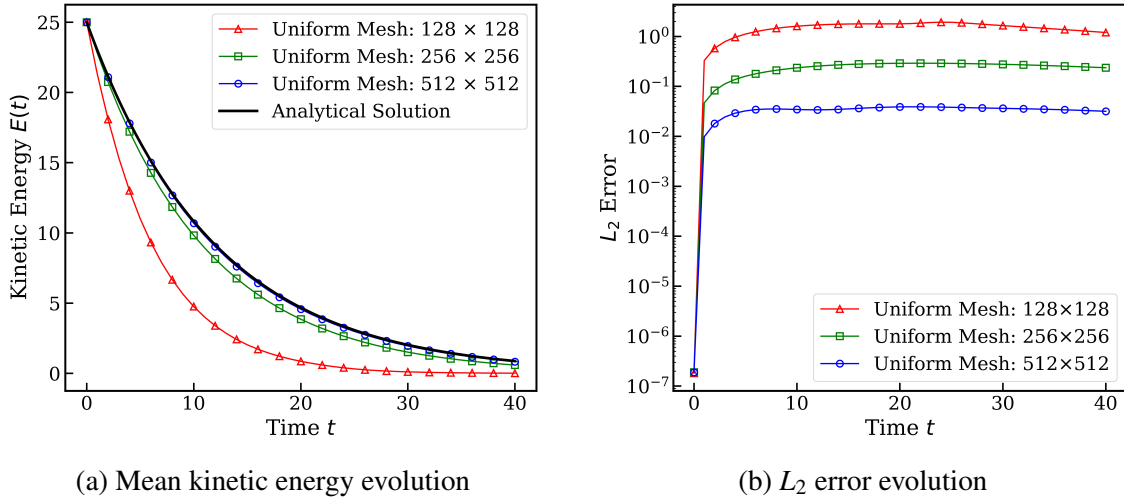


Figure 4.8: Mean kinetic energy and  $L_2$  error at different time instances

### 4.3.4 Velocity Field in 2D Taylor-Green Vortex Flow

The velocity field of the 2D TGV initially exhibits symmetric vortical structures with distinct vortex centers (Figure 4.9). Over time, viscous dissipation causes a continuous decay in the kinetic energy of the flow, gradually weakening the vortices. This leads to a noticeable reduction in the overall magnitude of the velocity, as observed in the velocity contours at time  $t = 20$ .

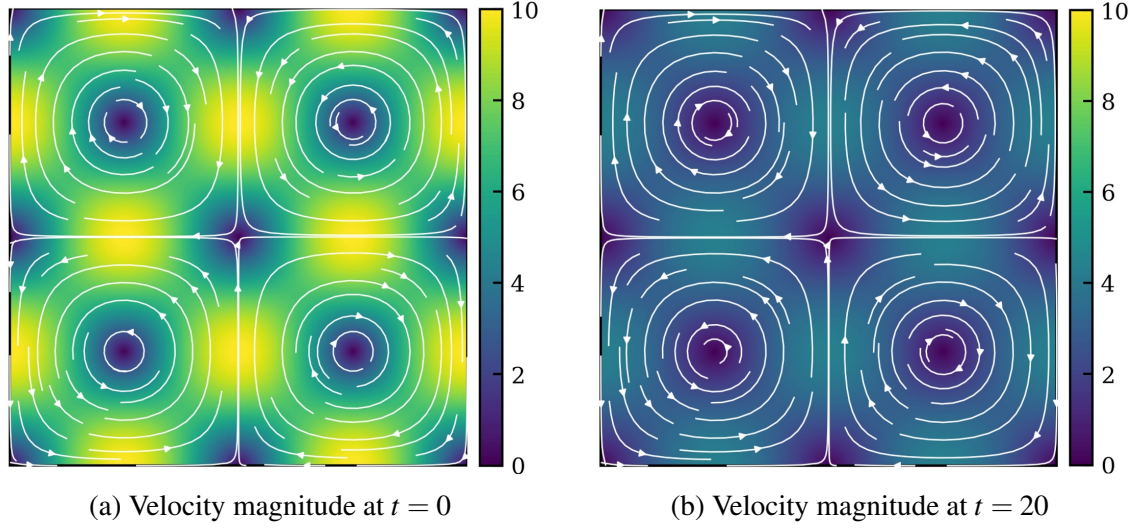


Figure 4.9: Velocity magnitude contours at different time instances

### 4.3.5 Adaptive Mesh Refinement for the 2D Taylor-Green Vortex

The uniform grid simulation performed at a resolution of  $512 \times 512$  served as a high-fidelity reference for the Taylor-Green vortex problem in  $\text{Re} = 3000$ . However, it also highlighted the significant computational cost associated with resolving the vortex structures uniformly across the entire domain. To mitigate this, AMR was used to concentrate computational resources in dynamically evolving regions of interest, thus aiming to improve efficiency without compromising accuracy.

#### 4.3.5.1 The AMR Criteria and Parameters

The AMR capability available in Athena++ was utilized, with refinement guided by the local grid-based Reynolds number, defined as

$$\text{Re}_{\text{grid}} = \frac{\Delta x \cdot u_{\text{local}}}{\nu} \quad (4.13)$$

where  $\Delta x$  denotes the local grid spacing,  $u_{\text{local}}$  is the local velocity magnitude and  $\nu$  is the kinematic viscosity. This criterion ensured that regions of higher flow velocity, particularly around vortex cores and along the shear interfaces between counter-rotating structures, were allocated finer resolution, whereas relatively quiescent regions were maintained at a coarser resolution.

Two sets of AMR simulations were performed using different root mesh resolutions, as summarized in Table 4.13. The refinement threshold was specified to reach the targeted minimum  $\text{Re}_{\text{grid}}$  of 5.0. The de-refinement criterion was set to  $0.5 \times \text{Re}_{\text{grid}}$ . Thus, a balance between spatial resolution and computational efficiency was aimed.

Table 4.9: AMR simulation configurations

| Root Grid Resolution | Maximum Levels           | Refinement | Effective Resolution | Maximum |
|----------------------|--------------------------|------------|----------------------|---------|
| $64 \times 64$       | 4 (including root level) |            | $512 \times 512$     |         |
| $128 \times 128$     | 3 (including root level) |            | $512 \times 512$     |         |

To improve stability in the refinement process, the de-refinement parameter in Athena++ was configured such that a MeshBlock must remain refined for at least 5 consecutive time steps before being coarsened. This mechanism suppresses the immediate destruction of newly refined MeshBlocks and promotes a gradual change in the grid hierarchy, thereby preventing abrupt variations in the solution’s evolution.

The main AMR parameters used in the simulations are summarized in Table 4.10.

Table 4.10: Key AMR configuration parameters in Athena++

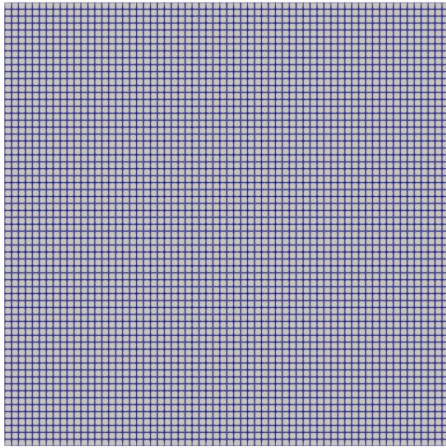
| Parameter                | Value / Description                                                             |
|--------------------------|---------------------------------------------------------------------------------|
| Refinement criterion     | $\text{Re}_{\text{grid}} > \text{Target Re}_{\text{grid}}$ (target value = 5.0) |
| De-refinement criterion  | $\text{Re}_{\text{grid}} < \frac{1}{2} \times \text{Target Re}_{\text{grid}}$   |
| Maximum refinement level | 4 (with root grid $64 \times 64$ )<br>3 (with root grid $128 \times 128$ )      |
| De-refinement delay      | 5 time steps (de-refinement count)                                              |

#### 4.3.5.2 Analysis of AMR Process

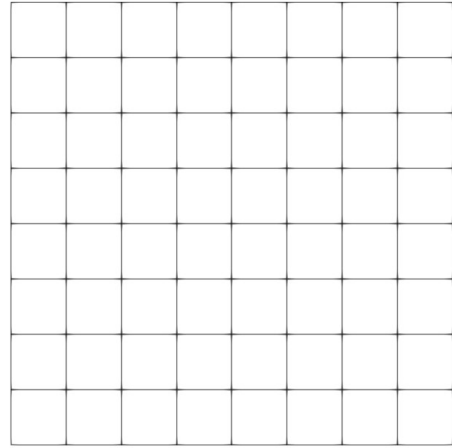
The computational domain, discretized using a uniform Cartesian root grid of resolution  $64 \times 64$ , is illustrated in Figure 4.10a. This root grid is composed of  $8 \times 8$  uniform MeshBlocks, each consisting of  $8 \times 8$  cells, as shown in Figure 4.10b. The AMR algorithm is activated right from the initial condition and uses the Reynolds number,  $\text{Re}_{\text{grid}}$ , as the refinement criterion. The refinement and de-refinement behavior over time is depicted in Figure 4.11, which shows the MeshBlock configurations at time instances  $t = 0, 10, 20$ , and 30 for the  $64 \times 64$  root mesh case. Regions where  $\text{Re}_{\text{grid}}$  exceeds the predefined threshold of 5.0 are refined to provide greater resolution, particularly in areas with high velocity magnitudes.

The refinement of MeshBlocks (hence of the grid) up to 3 levels can be seen in Figures 4.11a and 4.11b. As the Taylor–Green vortex evolves, viscous dissipation leads to a decay in kinetic energy, resulting in reduced velocity magnitudes and hence lower  $\text{Re}_{\text{grid}}$  values. This prompts the AMR algorithm to gradually coarsen the mesh in those regions.

The test case using the root grid  $128 \times 128$  follows the same  $\text{Re}_{\text{grid}}$ -based refinement logic and allows for up to two additional refinement levels.

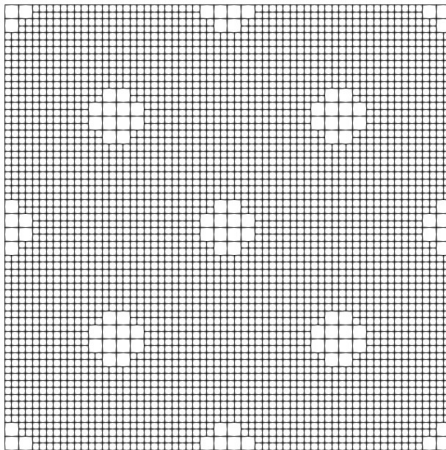


(a) Root grid  $64 \times 64$

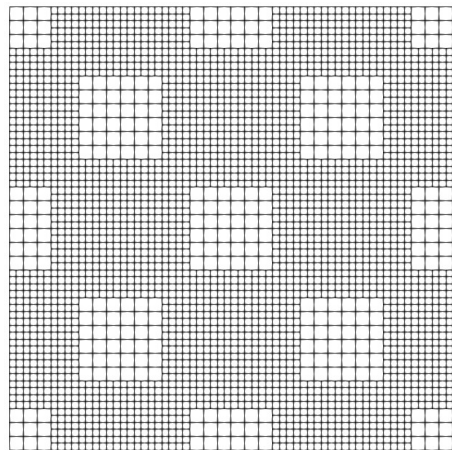


(b) MeshBlocks of root grid  $64 \times 64$

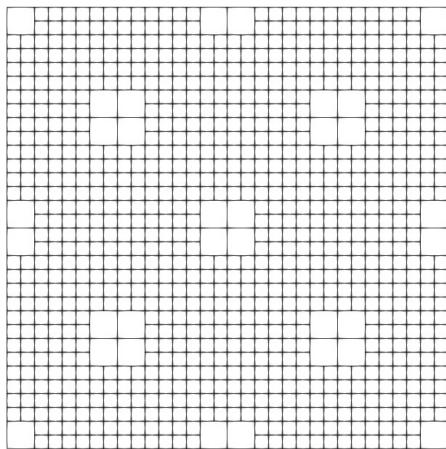
Figure 4.10: Root grid  $64 \times 64$  and its MeshBlocks



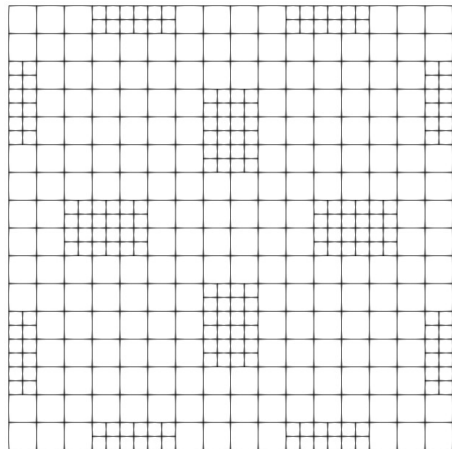
(a) Adaptive mesh refinement,  $t = 0$



(b) Adaptive mesh refinement,  $t = 10$



(c) Adaptive mesh refinement,  $t = 20$



(d) Adaptive mesh refinement,  $t = 30$

Figure 4.11: MeshBlocks resulting from adaptive mesh refinement and de-refinement

#### 4.3.5.3 Analysis of AMR-based Solutions

To assess the performance and accuracy of the AMR approach, the numerical solutions obtained using the discussed refinement strategy were compared against those from the uniform-grid simulations. The comparison focused primarily on the evolution of mean kinetic energy over time and the total computational time required for each simulation.

The figure shows the time evolution of kinetic energy  $E(t)$  comparing the results from two AMR cases with the analytical solution and the numerical solution obtained using  $512 \times 512$  uniform mesh. Both AMR cases show good agreement with the analytical trend, demonstrating the effectiveness of the refinement strategy. Among the two, the AMR configuration with the  $128 \times 128$  root mesh exhibits slightly better trend compared to the  $64 \times 64$  case. However, a gradual deviation in the mean kinetic energy from the analytical solution becomes noticeable in both AMR cases beyond  $t \approx 15$ , likely due to accumulated numerical dissipation and resolution limits in the refined regions.

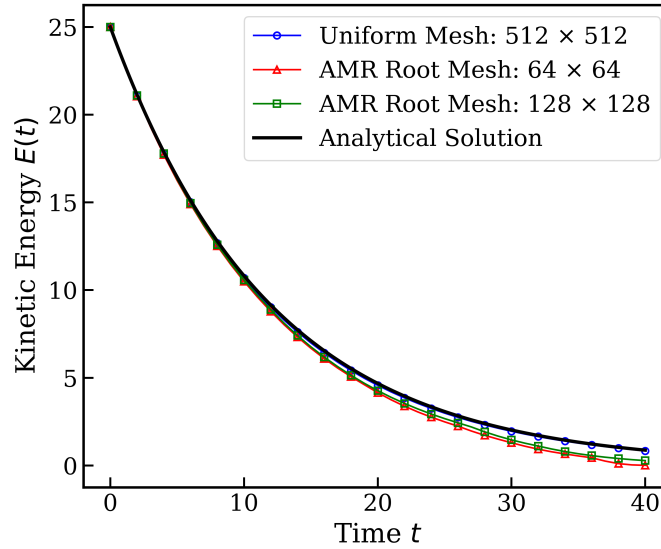


Figure 4.12: Comparison of mean kinetic energy evolution

#### 4.3.5.4 AMR with Higher Order Schemes

A simulation of the 2D TGV flow at a Reynolds number of 3000 was performed using the third-order accurate Piecewise Parabolic Method (PPM) for spatial reconstruction in combination with a fourth-order time integration scheme (Table 4.11). A root mesh of  $64 \times 64$  with four refinement levels was employed, and the grid Reynolds number was set to  $Re_{\text{grid}} = 10$  to ensure sufficient resolution of the smallest dynamically relevant scales. The results show good agreement with the analytical solution, as illustrated in Figure 4.13, although the solution exhibits a slightly reduced decay rate compared to the analytical solution after  $t \approx 30$ .



The results obtained for 2D TGV flow underscore the necessity of employing higher-order spatial reconstruction scheme together with higher-order time integrator to obtain improved accuracy. This requirement becomes even more critical for the simulation of complex three-dimensional flows, and therefore forms the basis for the subsequent study of the three-dimensional Taylor–Green vortex using higher-order schemes.

Table 4.11: Higher order schemes used with AMR simulation

| Parameter              | Value / Description              |
|------------------------|----------------------------------|
| Spatial reconstruction | Piecewise parabolic method (PPM) |
| Time integration       | Fourth-order Runge–Kutta method  |

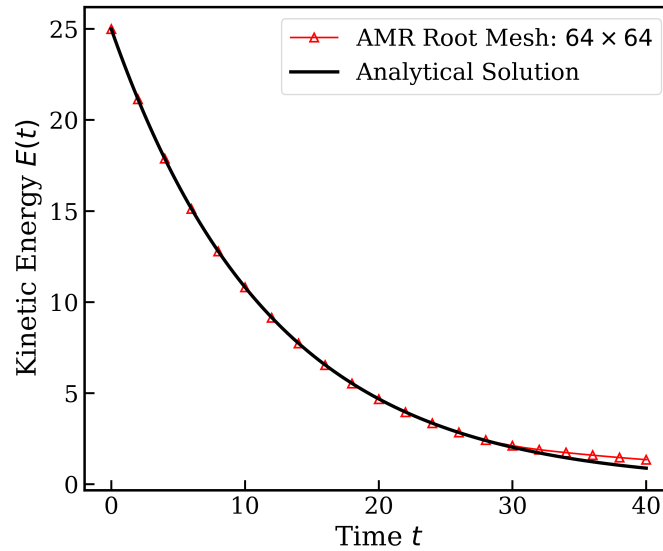


Figure 4.13: Comparison of mean kinetic energy evolution

#### 4.3.6 Computational Time required for Simulations

Table 4.12. All runs were performed using 16 processors on the HPC system, with the uniform mesh case of  $512 \times 512$  cells serving as the reference. The AMR cases demonstrate clear efficiency gains compared to the uniform mesh. For example, the case with a root mesh of  $64 \times 64$  reduces the runtime by about 18%, while still maintaining adequate accuracy in dynamically active regions.

Among the AMR setups, the root mesh of  $128 \times 128$  shows the largest improvement, achieving approximately 44% reduction in runtime relative to the uniform mesh case. Furthermore, when higher-order schemes are employed on the  $64 \times 64$  root mesh, the computational savings increase to around 32%, indicating that AMR combined with higher-order accuracy provides a favorable balance between efficiency and fidelity. These findings highlight the effectiveness of AMR in reducing computational cost while preserving



resolution, and they further motivate the use of AMR together with higher-order reconstruction and time integration schemes in the subsequent three-dimensional TGV study.

Table 4.12: Comparison of Computational Times

| <b>Mesh</b>                                                                               | <b>Computational Time</b> | <b>Reduction in Runtime (%)</b> |
|-------------------------------------------------------------------------------------------|---------------------------|---------------------------------|
| Uniform Mesh, $512 \times 512$                                                            | 44 hrs 30 min             | —                               |
| AMR, Root Mesh, $64 \times 64$                                                            | 36 hrs 20 min             | 18%                             |
| AMR, Root Mesh, $128 \times 128$                                                          | 24 hrs 47 min             | 44%                             |
| AMR, Root Mesh, $64 \times 64$<br>(higher-order schemes, $\text{Re}_{\text{grid}} = 10$ ) | 30 hrs 8 min              | 32%                             |

## 4.4 Three-Dimensional Taylor-Green Vortex

The three-dimensional (3D) Taylor–Green vortex (TGV) is a standard benchmark for studying the transition from laminar to turbulent flow and the subsequent decay of turbulence. Its well-defined initial conditions and rich dynamical evolution make it an ideal case for validating numerical solvers under both fully resolved and under-resolved DNS settings. Based on insights from the two-dimensional TGV, the present study employs PPM for spatial reconstruction and RK4 for time integration to better capture the onset and evolution of turbulence, while also assessing the role of AMR in resolving small-scale structures without uniform high-resolution grids.

### Flow Domain, Fluid Properties and Reynolds Number:

The simulation is carried out in a periodic cubic domain,  $(x, y, z) \in [-\pi, \pi]^3$ , where the flow evolves from a smooth, large-scale initial condition to increasingly complex and chaotic structures. The initial conditions are defined as follows.

$$u(x, y, z, 0) = U_0 \sin(x) \cos(y) \cos(z) \quad (4.14)$$

$$v(x, y, z, 0) = -U_0 \cos(x) \sin(y) \cos(z) \quad (4.15)$$

$$w(x, y, z, 0) = 0 \quad (4.16)$$

where  $U_0$  is the initial velocity amplitude.

The initial pressure field for incompressible flow is given by

$$p(x, y, z, 0) = p_{\text{ref}} + \frac{1}{16} U_0^2 (\cos(2x) + \cos(2y)) (\cos(2z) + 2) \quad (4.17)$$

where  $p_{\text{ref}}$  is the reference or background pressure.

The 3D TGV flows encompassing both laminar and turbulent regimes have been investigated. Simulations of the laminar flow conditions offer valuable insights into the robustness of the numerical solver, as well as the grid resolution requirements and the accuracy of the employed discretization schemes.

An initial velocity amplitude  $U_0 = 1.0$  is used for all the 3D TGV simulations. The Reynolds number is defined as per Equation 4.11, with the characteristic length ( $L$ ) taken as, 1.0. The choice of  $L = 1.0$  allows comparing the results with benchmark solutions.

All fluid properties, except for the kinematic viscosity, are kept consistent with the values listed in Table 4.7. For the present study, with  $U_0 = 1$  and  $L = 1$ , Equation 4.11 gives the kinematic viscosity as  $\nu = 1/\text{Re}$ .

### Boundary Conditions and Reference Pressure:

As in the 2D TGV case, periodic boundary conditions are applied on all faces of the cubic domain, as summarized in Table 4.8. The reference pressure is determined using Equation 4.4, assuming a Mach number of 0.29 and the specified fluid properties. The resulting reference pressure is  $p_{ref} = 7.134$ , and it has been considered for all the 3D TGV cases.

## 4.4.1 Three-Dimensional Taylor-Green Vortex: Laminar Flow

To evaluate the solver performance in the laminar regime, the 3D TGV problem is simulated at Reynolds numbers of 100 and 400. At these relatively low Reynolds numbers, the flow remains smooth and free from turbulence, making the cases well-suited for assessing numerical accuracy.

### 4.4.1.1 Simulation of 3D TGV Flow at $Re = 100$ :

A grid convergence study is conducted to examine the influence of resolution on key flow quantities. The results of this study form the basis for the subsequent analysis and validation.

#### Grid Convergence and Error Analysis:

The simulations were conducted on a series of three uniformly refined grids— $32^3$ ,  $64^3$ , and  $128^3$ —corresponding to a constant refinement ratio of 2.0 between successive levels of resolution. The consistency ratio, computed from the grid convergence indices at  $t = 5$  based on the mean kinetic energy (Equation 4.18) was determined to be 1.003, suggesting that the solution lies within the asymptotic range of convergence.

The mean kinetic energy is defined as

$$E(t) = \frac{1}{|V|} \int_{\Omega} \frac{1}{2} (u(x,y,z,t)^2 + v(x,y,z,t)^2 + w(x,y,z,t)^2) dV \quad (4.18)$$

where  $\mathbf{u} = (u, v, w)$  denotes the velocity field and  $|V|$  is the volume of the computational domain.

The corresponding mean kinetic energy dissipation rate is obtained from the temporal decay of  $E(t)$  as

$$\varepsilon(t) = -\frac{dE(t)}{dt} \quad (4.19)$$

For an incompressible viscous flow,  $\varepsilon(t)$  can be expressed in terms of the magnitude of the vorticity as

$$\varepsilon(t) = 2\nu \frac{1}{|V|} \int_{\Omega} |\boldsymbol{\omega}(x,y,z,t)|^2 dV \quad (4.20)$$

where  $\nu$  is the kinematic viscosity and  $\boldsymbol{\omega} = \nabla \times \mathbf{u}$  is the vorticity vector.

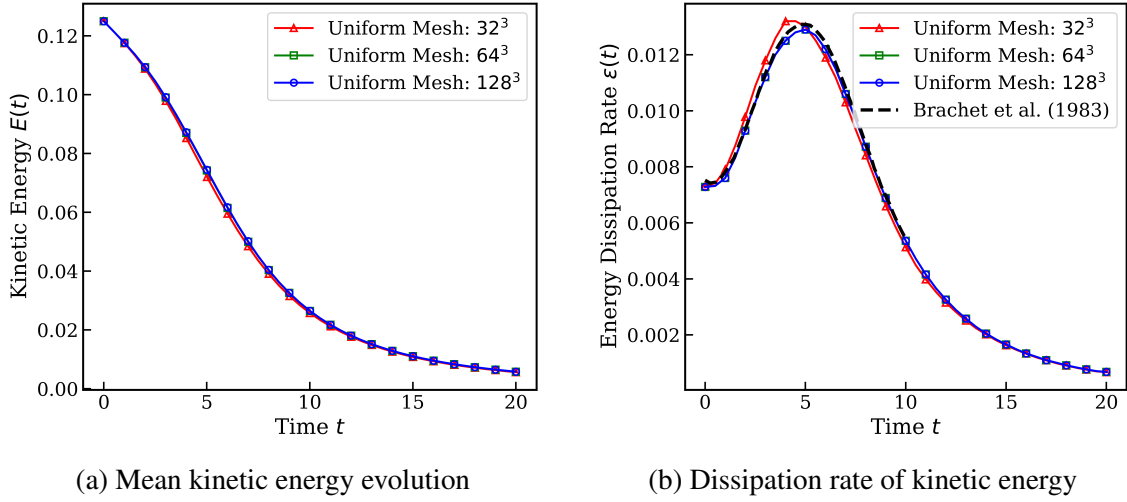


Figure 4.14: Grid convergence of 3D TGV simulations for  $Re = 100$

Figure 4.14 illustrates the time evolution of kinetic energy for all three resolutions. The energy dissipation rate is compared against the benchmark solution from Brachet et al. [26]. The results indicate that the  $64^3$  and  $128^3$  grid provide solutions in close agreement with the benchmark data. Furthermore, the temporal evolution of the absolute error in mean kinetic energy demonstrates that the error is lowest for the finest grid. However, the solution obtained using  $64^3$  resolution also provides sufficiently accurate solution.

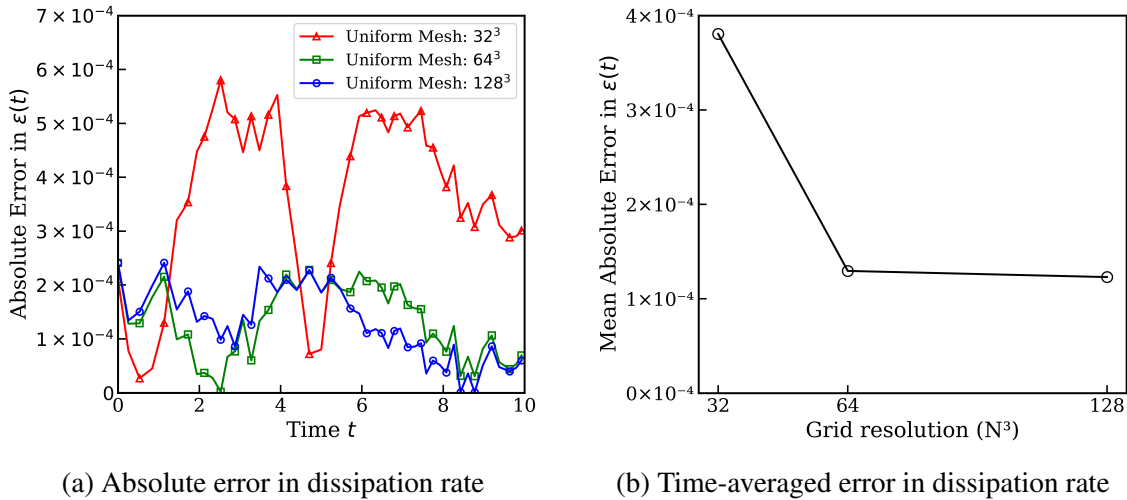


Figure 4.15: Absolute errors in dissipation rate of 3D TGV for  $Re = 100$

Figures 4.15a and 4.15b present the absolute errors in the dissipation rate,  $\epsilon(t)$ , for uniform meshes of resolutions  $32^3$ ,  $64^3$ , and  $128^3$ . Figure 4.15a shows the temporal varia-

tion of the error for each mesh, while Figure 4.15b summarizes the corresponding time-averaged absolute errors as a function of grid resolution. The results indicate that increasing the resolution from  $32^3$  to  $64^3$  yields a substantial reduction in error, whereas the improvement from  $64^3$  to  $128^3$  is comparatively small. Notably, the averaged error for the  $64^3$  mesh is nearly identical to that of the  $128^3$  mesh, suggesting that  $64^3$  provides an adequate resolution for capturing the dissipation rate in this case.

### Velocity Field at Different Time Instances:

Figure 4.16 shows contours of velocity magnitude with in-plane velocity vectors on the  $z$ -plane through  $(x, y, z) = (0, 0, 0)$  at times  $t = 0$  and  $t = 5$ . The contours illustrate the laminar decay of the vortex structure, with no chaotic velocity patterns observed. At  $t = 0$ , the contours are dominated by bright yellow regions corresponding to higher velocity magnitudes, whereas by  $t = 5$  the flow field has weakened and become smoother as the vortex decays. This visual observation of vortex decay is consistent with the temporal evolution of kinetic energy and energy dissipation rate (Figure 4.14).

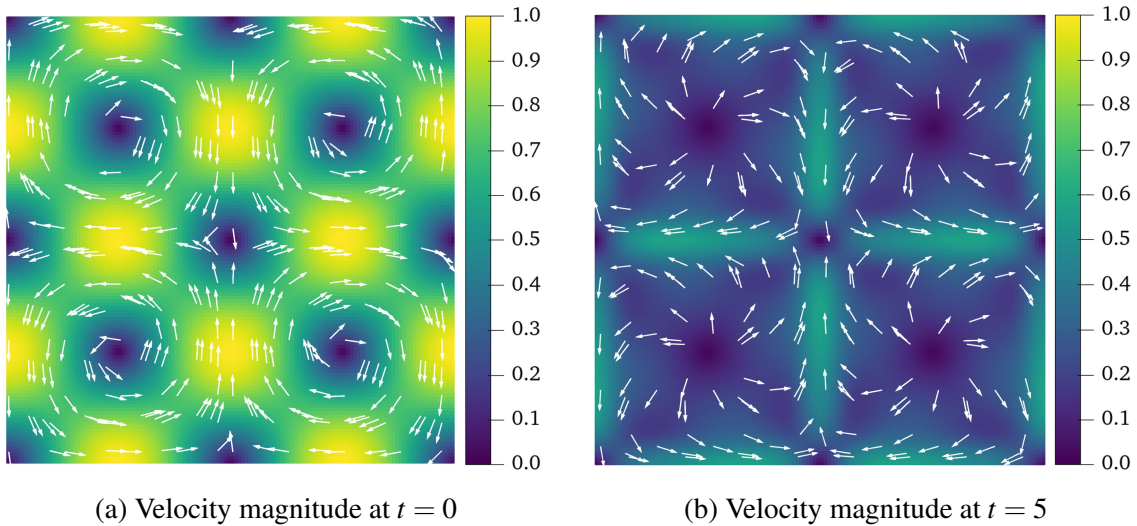
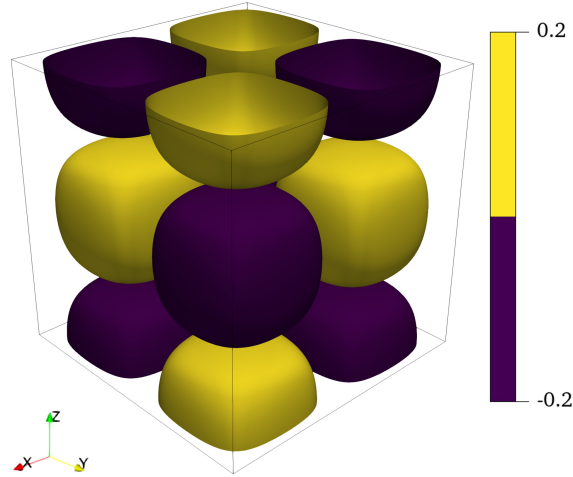


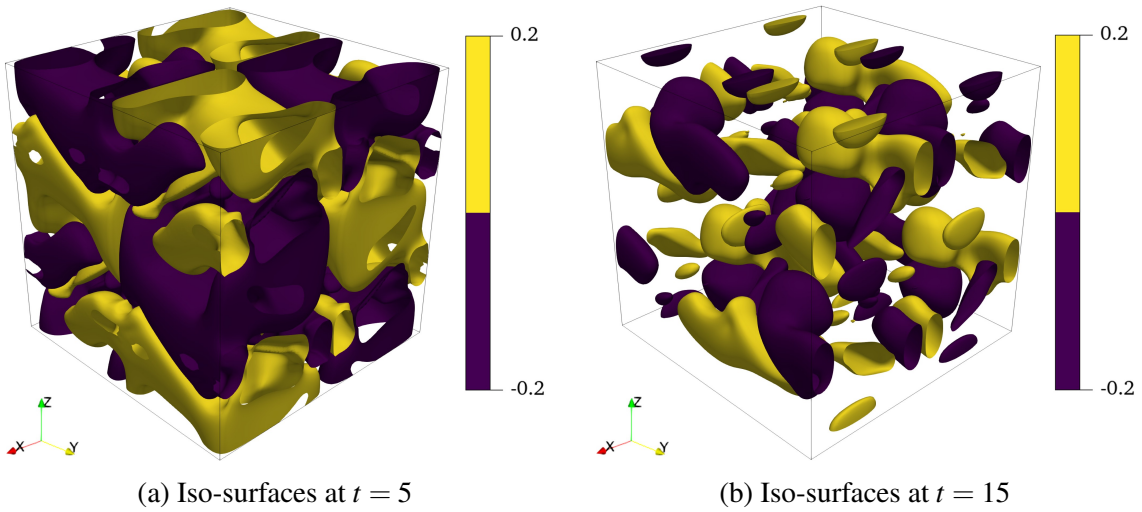
Figure 4.16: Velocity magnitude on  $z$ -plane at different time instances for  $Re = 100$

### Vorticity Iso-surfaces at Different Time Instances:

At  $t = 0$ , the iso-surfaces of the  $z$ -component of the vorticity  $\omega_z$  shown in Figure 4.17, display the regular periodic structure prescribed by the Taylor–Green vortex initial condition, with yellow indicating  $\omega_z = 0.2$  and dark purple indicating  $\omega_z = -0.2$ . This well-organized pattern provides a useful baseline for evaluating the subsequent flow evolution.

Figure 4.17: Iso-surfaces of vorticity at time  $t = 0$ 

The evolution at  $t = 5$  and  $t = 15$  (Figure 4.18), highlights the progressive breakdown of the initial large-scale vortices into smaller structures through nonlinear interactions. At later times, these fine-scale vortical features persist but gradually weaken as viscous dissipation reduces the overall flow intensity, while remnants of the initial anisotropy remain visible. These observations confirm that the solver successfully captures the transition from organized vortex structures to small-scale dissipation, consistent with the expected dynamics of the Taylor–Green vortex.

Figure 4.18: Iso-surfaces of vorticity at different time instances for  $\text{Re} = 100$ 

#### Solutions obtained from Uniform and AMR Meshes:

In this section, the decay of the 3D TGV flow is studied at Reynolds numbers,  $\text{Re} = 100$  and using both uniform mesh resolution and the AMR approach. The numerical results are compared with benchmark solutions from Brachet et al. [26]. The AMR strategy is

guided by the refinement criterion based on the local grid Reynolds number, as defined in Equation 4.13. The comparison focused primarily on the evolution of mean kinetic energy over time, energy dissipation rate and the total computational time required for each simulation.

Two sets of AMR simulations were performed, as summarized in Table 4.13. The refinement threshold was chosen to achieve a target minimum grid-based Reynolds number,  $Re_{\text{grid}}$ , of 4.9. This value corresponds to the maximum resolution attainable on a uniform grid of  $128^3$  cells. The de-refinement criterion was set to  $0.5 \times Re_{\text{grid}}$ .

Table 4.13: AMR simulation configurations used for  $Re = 100$

| Root Resolution | Grid | Refinement Levels (including root) | Refinement Threshold     | De-refinement Threshold       |
|-----------------|------|------------------------------------|--------------------------|-------------------------------|
| $32^3$          |      | 3                                  | $Re_{\text{grid}} = 4.9$ | $0.5 \times Re_{\text{grid}}$ |
| $64^3$          |      | 2                                  | $Re_{\text{grid}} = 4.9$ | $0.5 \times Re_{\text{grid}}$ |

To enhance stability in the refinement process, the de-refinement parameter in Athena++ was configured such that a MeshBlock must remain at a refined level for at least five consecutive time steps before being coarsened. This approach prevents the immediate removal of newly refined MeshBlocks and promotes gradual adjustments to the grid hierarchy. As a result, abrupt changes in the numerical solution are minimized, and a balance between spatial resolution and computational efficiency is maintained.

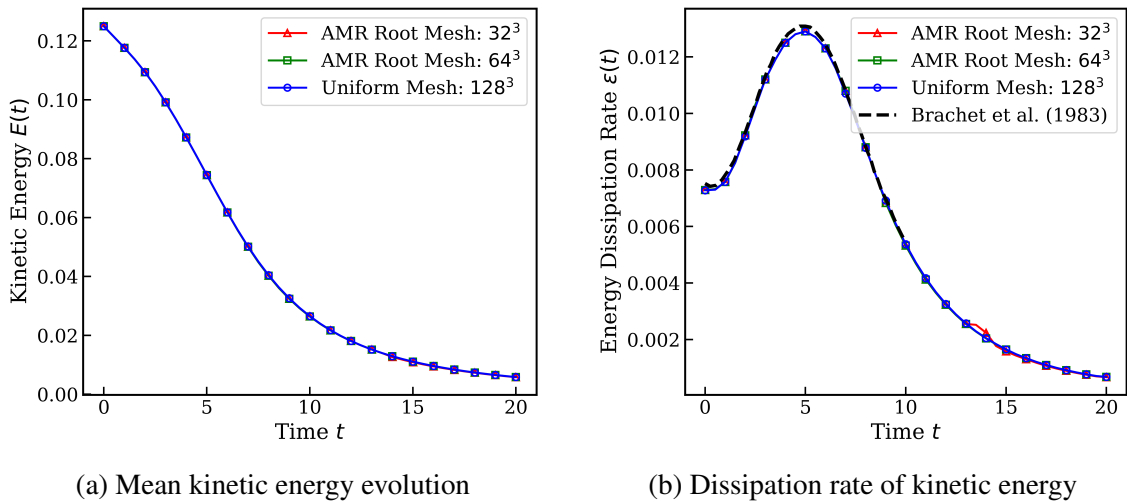


Figure 4.19: Comparison of solutions from uniform mesh and AMR for  $Re = 100$

Figure 4.19 shows the time evolution of the kinetic energy,  $E(t)$ , for the two AMR configurations, compared against the solution computed on uniform  $128^3$  mesh. Both AMR cases follow the uniform grid solution closely, indicating that the refinement strategy is

effective. The configuration with a  $64^3$  root grid exhibits a close agreement with fine grid in the energy–dissipation rate, whereas the  $32^3$  case begins to deviate from the expected dissipation rate for  $t > 13$ , likely due to accumulated numerical dissipation and the reduced effective grid resolution.

### Analysis of AMR Process:

This section analyzes the AMR configuration adopted for the simulations and examines its impact on both spatial resolution and computational cost.

**Domain setup and refinement criterion:** The computational domain, discretized using a uniform Cartesian root mesh of resolution  $32^3$ , is illustrated in Figure 4.20a. This root grid is composed of  $4^3$  uniform MeshBlocks, as shown in Figure 4.20b, with each Mesh-Block consisting of  $8^3$  uniform finite volume cells. The AMR algorithm is activated from the initial condition and uses the local grid-size-based Reynolds number,  $Re_{\text{grid}}$ , as the refinement criterion. Regions where  $Re_{\text{grid}}$  exceeds the predefined threshold of 4.9 are refined to provide greater resolution, particularly in areas with high velocity magnitudes.

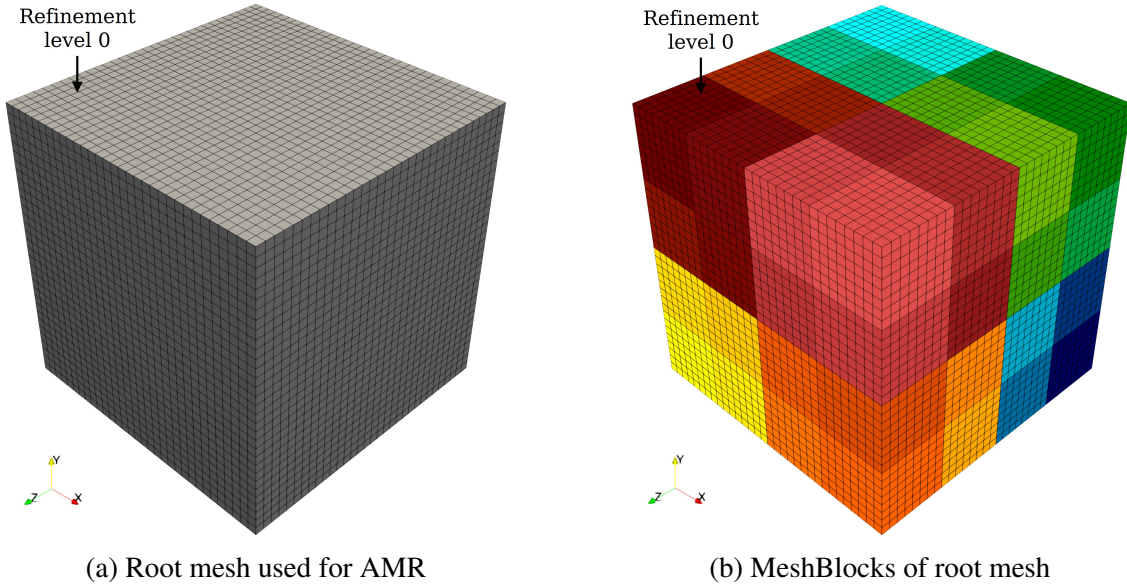


Figure 4.20: Root mesh used for AMR

**Refinement and coarsening behaviour:** The refinement and de-refinement behaviour over time is depicted in Figures 4.21 and 4.22, which shows the mesh configurations at  $t = 0$  and  $t = 10$ . Regions with refinement levels 1 and 2 can be seen respectively in Figures 4.21a and 4.22a. As the Taylor–Green vortex evolves, viscous dissipation leads to a decay in kinetic energy, reducing velocity magnitudes and lowering  $Re_{\text{grid}}$ . This prompts the AMR algorithm to gradually coarsen the mesh in less active regions, improving computational efficiency while maintaining resolution where needed.



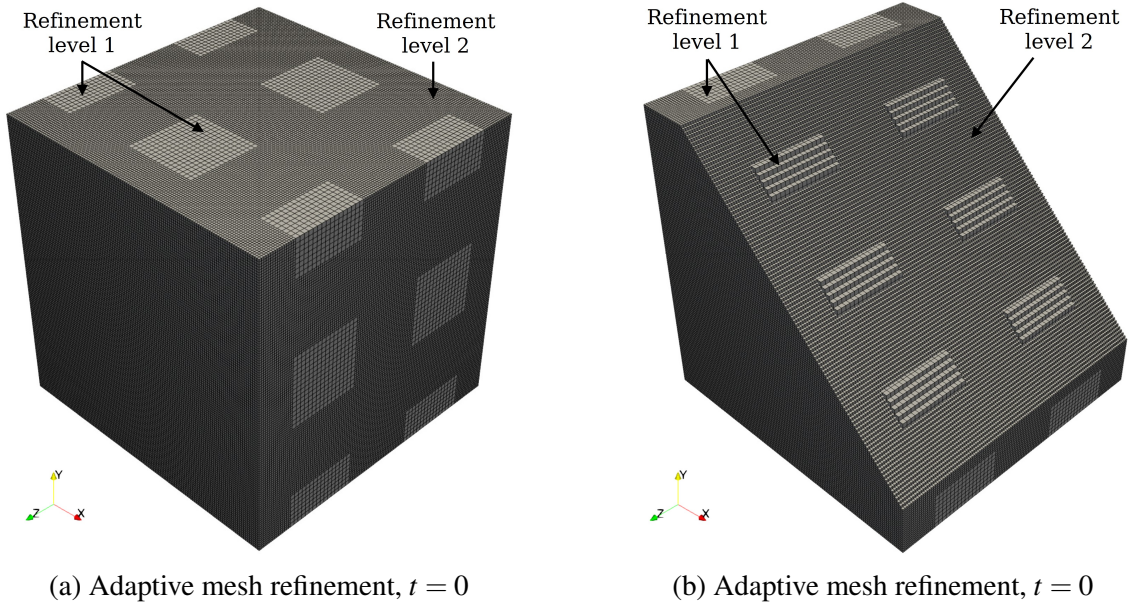


Figure 4.21: Mesh resulting from adaptive refinement and de-refinement

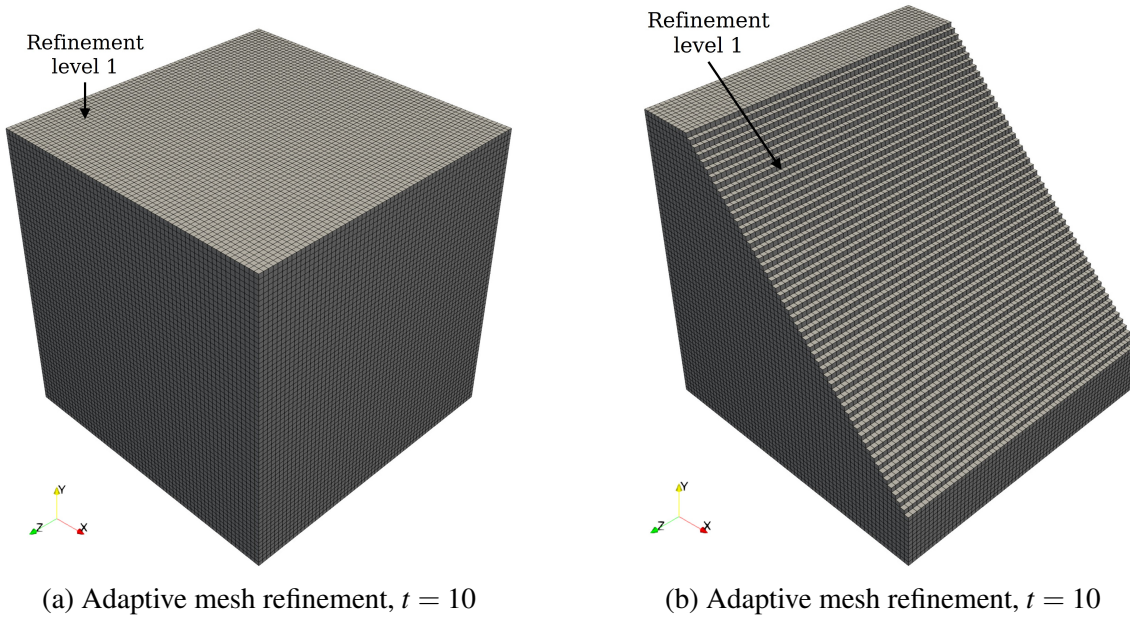


Figure 4.22: Mesh resulting from adaptive refinement and de-refinement

**Time Step Size versus Flow Time:** The variation of the time step size,  $\Delta t$ , with flow time,  $t$ , for the uniform and AMR cases is shown in Figure 4.23. For the uniform mesh  $128^3$ ,  $\Delta t$  increases only slightly, as the CFL condition is governed by the fixed fine grid spacing. In contrast, the AMR cases exhibit distinct jumps in  $\Delta t$  when de-refinement occurs. For the  $64^3$  root mesh (two refinement levels including root), the increase is moderate, reflecting one level of coarsening of the grid. The  $32^3$  root mesh (three refinement levels including root) shows the largest increase in  $\Delta t$  due to two levels of de-refinement, which

significantly reduces computational cost but also coarsens the temporal resolution in the later stages of the simulation.

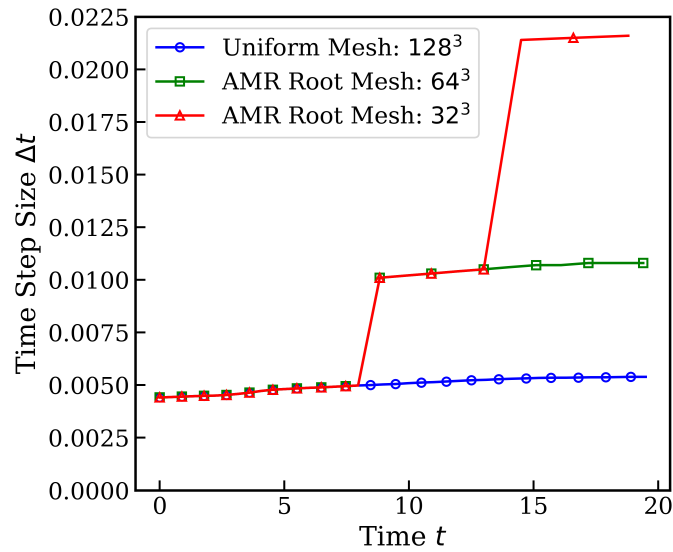


Figure 4.23: Change in time step sizes with flow time

#### Computational Time required for Simulations:

Table 4.14 compares the computational times for the 3D TGV simulations at  $Re = 100$  using uniform  $128^3$  mesh and two AMR configurations. Both AMR cases significantly reduce the total runtime, achieving savings of approximately 34–35% relative to the uniform-mesh case. These results demonstrate that AMR provides substantial computational efficiency without a notable loss in accuracy. The simulations were conducted in parallel on 16 processing cores of a mobile workstation.

Table 4.14: Computational times required for 3D TGV  $Re = 100$  simulations

| Mesh                   | Computational Time | Reduction in Runtime (%) |
|------------------------|--------------------|--------------------------|
| Uniform Mesh, $128^3$  | 55 min 45 sec      | —                        |
| AMR, Root Mesh, $32^3$ | 36 min 39 sec      | 34%                      |
| AMR, Root Mesh, $64^3$ | 36 min 5 sec       | 35%                      |

#### 4.4.1.2 Simulation of 3D TGV Flow at $Re = 400$ :

Following the assessment at  $Re = 100$ , where the flow remained smooth and well-resolved by both uniform and AMR meshes, an additional study was performed at a higher Reynolds

number of  $Re = 400$  to further evaluate solver performance in the laminar regime.

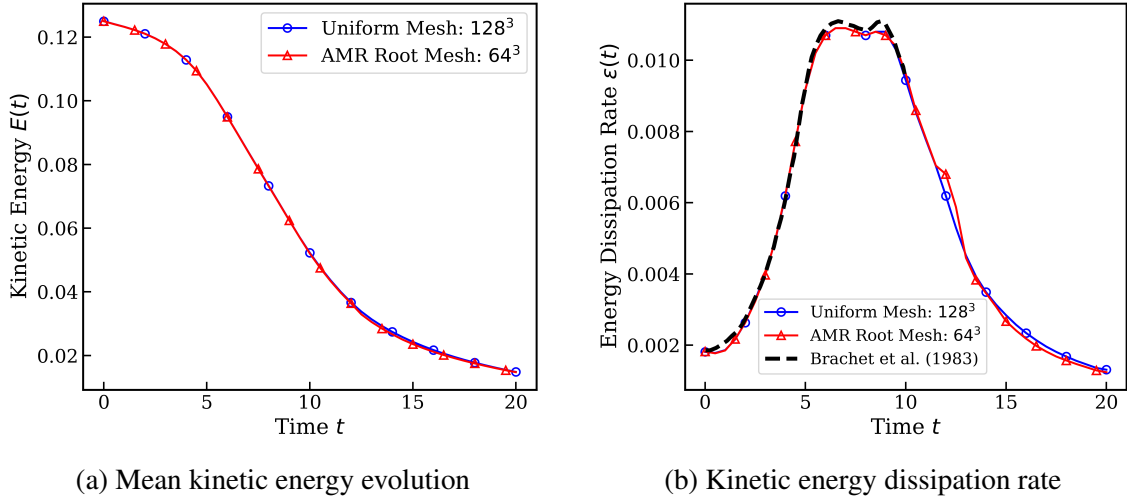


Figure 4.24: Mean kinetic energy and kinetic energy dissipation rate at  $Re = 400$

#### Solutions obtained from Uniform and AMR Meshes:

At  $Re = 400$ , the 3D TGV flow remains in the laminar regime, and the large-scale structures decay smoothly without transitioning to turbulence. The kinetic energy evolution, shown in Figure 4.24a, indicates close agreement between the  $128^3$  uniform mesh solution and the AMR approach with a  $64^3$  root mesh. Both approaches reproduce the overall decay trend observed in the benchmark pseudo-spectral solution of Brachet et al. [26], with only minimal deviations at intermediate times.

The dissipation rate, presented in Figure 4.24b, shows that both the uniform and AMR solutions capture the peak dissipation accurately, occurring at approximately  $t \approx 7$ . While the AMR solution exhibits slight underprediction during the decay phase after  $t \approx 14$ , the overall temporal evolution remains consistent with the uniform mesh and benchmark data. These results suggest that, for laminar conditions at  $Re = 400$ , the AMR approach can provide solutions of comparable accuracy to a fully refined uniform grid.

#### Computational Time required for Simulations:

The computational times required for the three-dimensional TGV simulations at  $Re = 400$  using a uniform  $128^3$  mesh and one AMR configuration are summarized in Table 4.15. Both simulations were performed in parallel on 16 processing cores of a mobile workstation. With the AMR refinement threshold set to  $Re_{\text{grid}} = 19.63$ , corresponding to the value achievable with the uniform mesh, no reduction in computational time was observed. The  $64^3$  root mesh with one additional refinement level required approximately 16% more time than the uniform case, primarily due to AMR overheads such as repeated refinement checks and MeshBlock creation and destruction. Thus, for this case, the AMR approach did not yield computational efficiency, in contrast to the results for  $Re = 100$ .

Table 4.15: Computational times required for 3D TGV  $Re = 400$  simulations

| Mesh                   | Computational Time | Reduction in Runtime (%) |
|------------------------|--------------------|--------------------------|
| Uniform Mesh, $128^3$  | 1 hr 3 min         | —                        |
| AMR, Root Mesh, $64^3$ | 1 hr 12 min        | -16%                     |

#### 4.4.2 Three-Dimensional Taylor-Green Vortex: Turbulent Flow

In the turbulent regime, the 3D TGV flow evolves rapidly from organized, large-scale vortical patterns to chaotic, small-scale turbulence, characterized by an energy cascade and enhanced dissipation. In contrast, at lower Reynolds numbers (laminar regime), the flow remains smooth and coherent, with vortices decaying gradually under viscous effects, without transitioning to turbulence. The turbulent regime exhibits significantly faster energy decay and more complex flow dynamics.

In this section, the decay of the 3D TGV flow is studied at two Reynolds numbers,  $Re = 1600$  and  $Re = 3000$ , using both uniform mesh resolution and the AMR approach. The numerical results are compared with benchmark solutions from Brachet et al. [26] as well as from Rees et al. [27]. The AMR strategy is guided by the refinement criterion based on the local grid Reynolds number, as defined in Equation 4.13.

##### 4.4.2.1 Comparison of Under-resolved DNS Solutions for $Re = 1600$ using Uniform and AMR Meshes

At  $Re = 1600$ , the 3D TGV undergoes a laminar-to-turbulent transition during the decay phase, characterized by the emergence of small-scale turbulent structures.

##### Grid Requirement for DNS study:

To fully resolve the scales present in turbulent flow using the DNS approach, the minimum required number of grid cells can be estimated using the relation given in Equation 3.12. Based on this criterion, the number of grid cells necessary to adequately capture the turbulent scales at  $Re = 1600$  is estimated to be  $253^3$ .

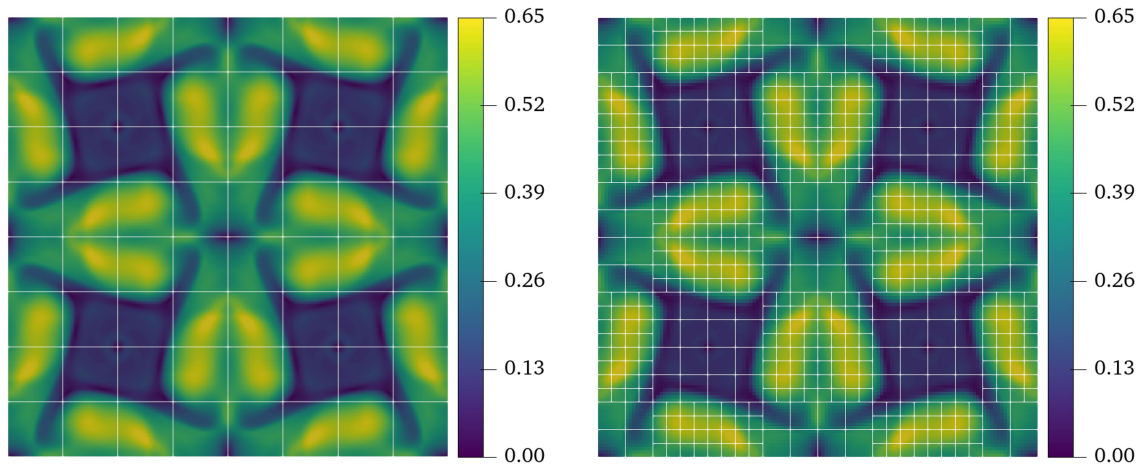
$$N^3 \approx Re^{\frac{9}{4}} \approx (1600)^{\frac{9}{4}} \quad (4.21)$$

Consequently, a uniform mesh  $256^3$  is used for the simulation. The corresponding Reynolds number based on the local grid size,  $Re_{\text{grid}}$ , calculated using the initial velocity amplitude, the size of the grid cell and the properties of the fluid, is found to be 39.27. Although a fully resolved DNS ideally requires  $Re_{\text{grid}} \approx 1.0$ , the chosen grid resolution is expected

to sufficiently capture the essential flow features and provide an adequately resolved solution for the case under consideration.

### Velocity Contours:

Figure 4.25 shows velocity magnitude contours on the  $z$ -plane through  $(0,0,0)$  at  $t = 15$  for flow at  $Re = 1600$ . The contours illustrate the distortion of the vortex structure and the onset of turbulence. In the AMR case, higher-velocity regions are refined by subdividing MeshBlocks, indicated by the overlaid white lines. The velocity field is consistent with the uniform-mesh result, with no qualitative differences observed. However, small-scale features cannot be inferred from a single plane section, and AMR may not fully capture the finest flow structures. This supports the validity of AMR for representing large-scale features prior to quantitative comparison of kinetic energy and dissipation rates.



(a) Velocity contours with uniform mesh MeshBlocks

(b) Velocity contours with AMR MeshBlocks

Figure 4.25: Velocity magnitude on  $z$ -plane at  $t = 15$  for  $Re = 1600$  and the MeshBlocks

### Kinetic Energy Decay and Energy Dissipation Rate:

Figure 4.26a shows the temporal evolution of the domain-averaged kinetic energy,  $E(t)$ , together with reference data from Rees et al. [27] which was obtained using a pseudo-spectral DNS method. Figure 4.26b presents the corresponding temporal evolution of the energy dissipation rate,  $\varepsilon(t)$ , along with benchmark results from Brachet et al. [26] and Rees et al. [27]. For the uniform  $256^3$  mesh, both the kinetic energy decay and dissipation rate evolution closely follow the benchmark trends, demonstrating good agreement with established DNS data. The flow energy decays rapidly in the early stages, accompanied by a correspondingly high dissipation rate.

Results from the AMR simulation, starting from a  $64^3$  root mesh and refining adaptively up to an equivalent  $256^3$  resolution in regions where  $Re_{\text{grid}} > 39.0$ , are also shown. The



AMR solution follows the uniform mesh solution up to  $t \approx 15$ , after which derefinement leads to a slight divergence. Nevertheless, the overall behaviour remains consistent with the uniform-mesh results. The dissipation rate peaks at  $t \approx 9$  are matching closely with the peak values reported by Brachet et al. [26] and Rees et al. [27].

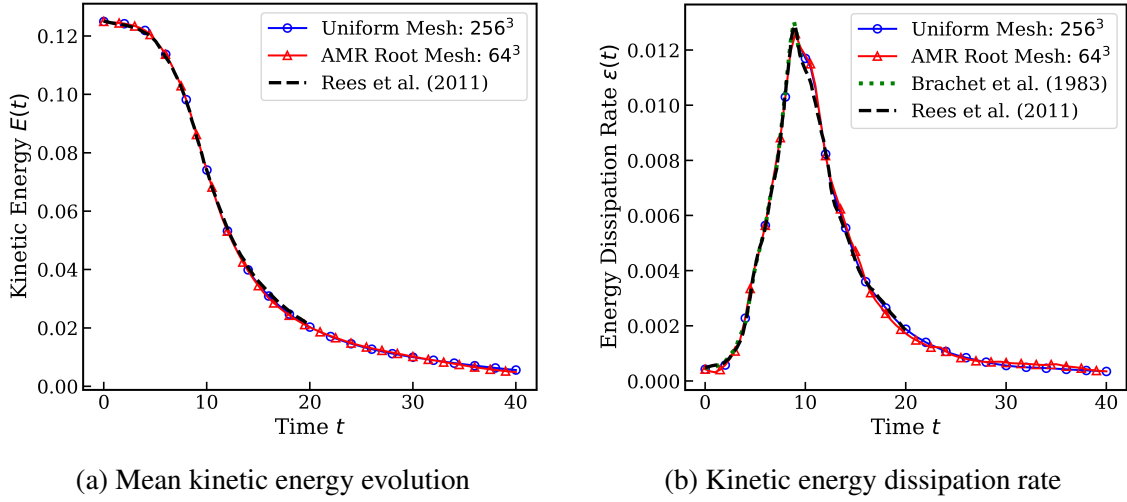


Figure 4.26: Mean kinetic energy and kinetic energy dissipation rate at different time instances for flow at  $Re = 1600$

#### Vorticity Iso-surfaces at Different Time Instances:

Figure 4.27 shows iso-surfaces of the  $z$ -component of the vorticity at  $t = 0$ . The yellow surfaces correspond to  $\omega_z = +0.7$  and dark purple to  $\omega_z = -0.7$ .

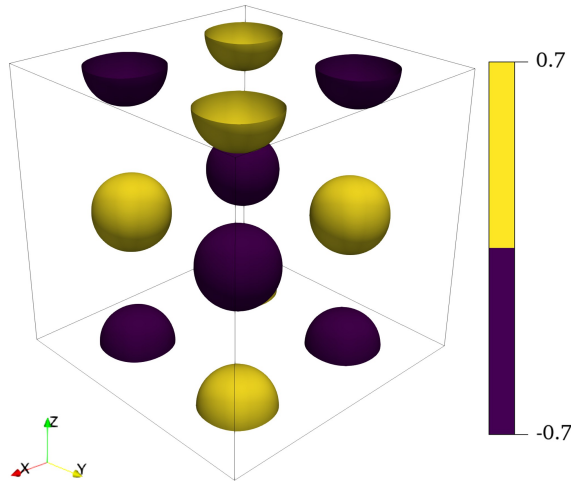


Figure 4.27: Iso-surfaces of vorticity at time  $t = 0$  for  $Re = 1600$

At the initial stage, the TGV exhibits a highly regular velocity distribution, reflecting the prescribed symmetric initial condition. The subsequent evolution at  $t = 10$  and  $t = 20$  is

presented in Figure 4.28. By  $t = 10$ , the initial structures have broken down into smaller vortical features, although remnants of the initial anisotropy of the Taylor–Green vortex remain visible. At  $t = 20$ , the flow develops into a more chaotic state with nearly isotropic vortex structures, characteristic of fully developed turbulence. From this stage, the flow may be interpreted as turbulent. At later times, the isotropic turbulence gradually decays due to viscous dissipation. These features of the flow evolution correspond closely to the classical 3D TGV dynamics reported in literature [26, 28].

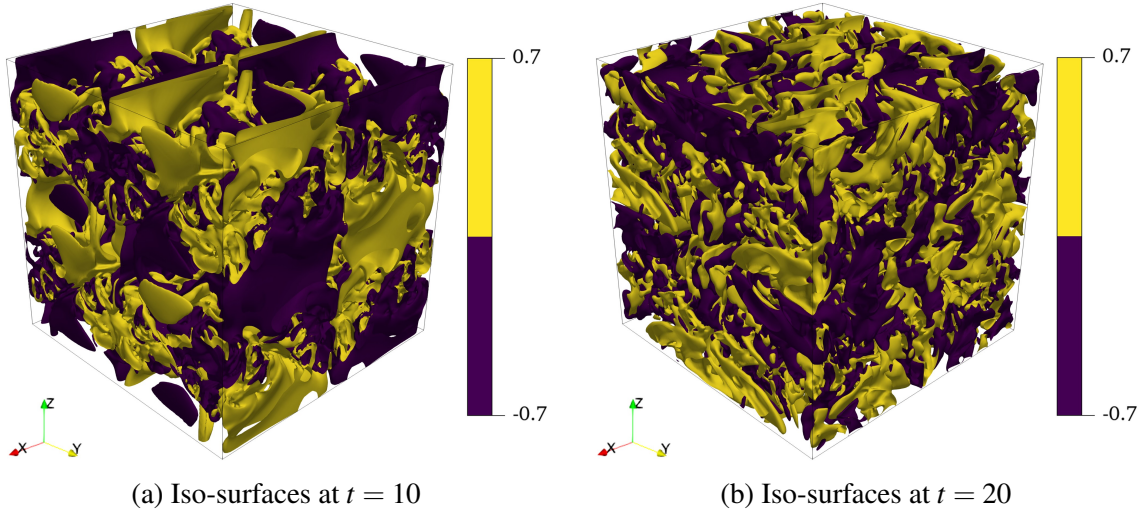


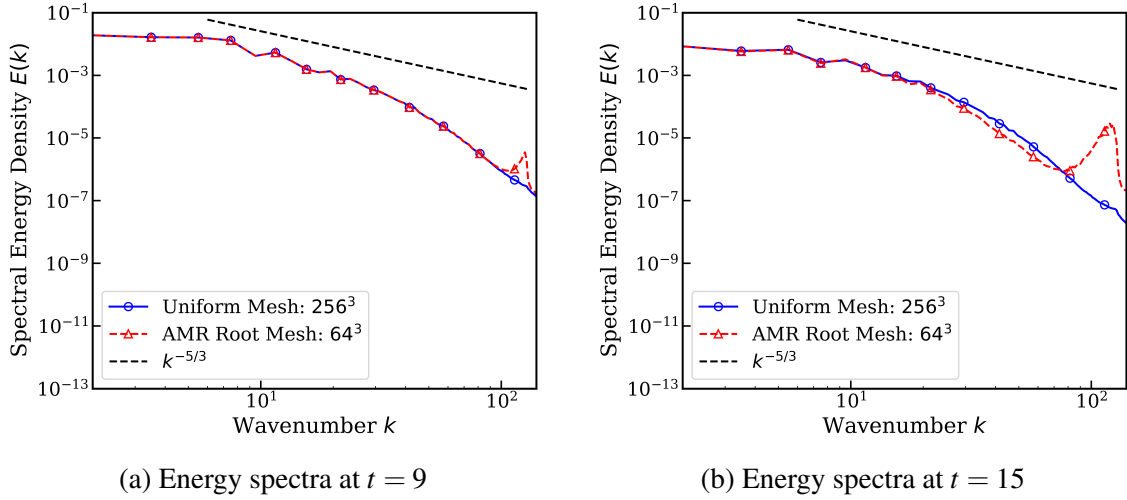
Figure 4.28: Iso-surfaces of vorticity at different time instances for  $\text{Re} = 1600$

### Energy Spectrum:

The maximum value of the dissipation rate determines the zone of the laminar–turbulent transition and the formation of the classical Kolmogorov  $k^{-5/3}$  scaling. The spectral density profiles  $E(k)$  of kinetic energy at time  $t = 9$  and  $t = 15$  are shown in Figure 4.29. For reference, the Kolmogorov law in spectral form,  $E(k) \sim k^{-5/3}$ , is also included.

The results obtained from both the uniform mesh and AMR approaches are similar at  $t = 9$ . The scaling law is approximately followed for spatial scales with wavenumbers between  $k = 8$  and  $k = 30$ . The inertial subrange is relatively narrow at this stage because the turbulence is still developing. However, as presented in [26] and [28], the inertial range behaviour of the flow is expected to extend from  $k \approx 8$  to  $k \approx 40$ . At  $t = 15$ , the uniform-mesh spectrum follows the expected trend, exhibiting a slightly wider inertial subrange consistent with the Kolmogorov scaling of  $k^{-5/3}$  (Figure 4.29b).

In contrast, the AMR solution, while closely matching the uniform-mesh spectrum across much of the inertial range, shows a non-physical rise in spectral energy density within the viscous dissipation range.

Figure 4.29: Energy spectra at  $t = 9$  and  $t = 15$  for  $Re = 1600$ 

The discrepancies at high wavenumbers ( $k > 60$ ) can be attributed to several factors. Repeated fine-coarse grid transitions, particularly under the de-refinement criterion, can introduce non-uniform truncation errors that are different at fine-coarse mesh boundaries than within uniform regions. These errors can generate numerical noise and instabilities, making the solution more sensitive to perturbations. In addition, the third-order spatial reconstruction scheme may be insufficient to fully capture the smallest scales once the mesh is adaptively coarsened. Together, these effects appear in the energy spectra as artificial contributions at smaller scales, representing numerical noise rather than physical flow energy, and highlight the need for careful error control and improved refinement strategies in AMR-based turbulence simulations.

#### Time Step Size versus Flow Time:

The variation of the timestep size  $\Delta t$  with simulation time  $t$  for the uniform mesh and AMR cases is shown in Figure 4.30. For the uniform mesh with a  $256^3$  resolution,  $\Delta t$  increases only gradually, as it remains constrained by the fixed fine grid spacing through the CFL condition.

The AMR case exhibits a distinct jump in  $\Delta t$  when mesh de-refinement occurs. For the  $64^3$  root mesh, refinement proceeds up to the third level. Once the de-refinement criterion is satisfied, grid coarsening begins. The resulting reduction in spatial resolution at later stages of the flow permits a larger stable timestep, thereby lowering the overall computational cost of the simulation.



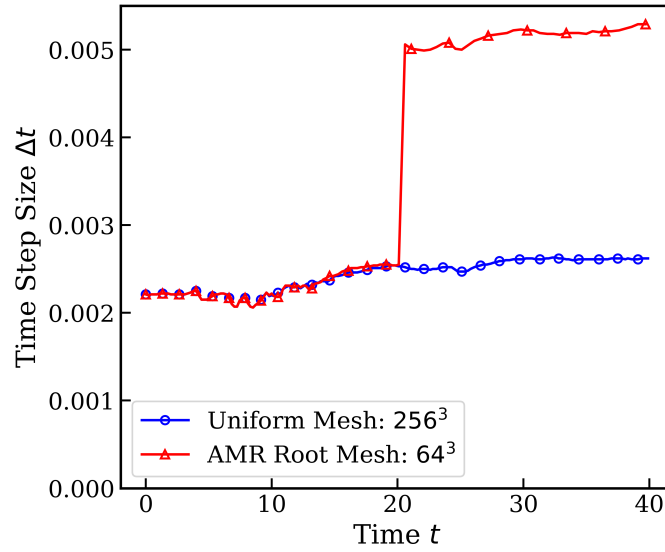


Figure 4.30: Change in timestep size of integration with flow time

#### Computational Time required for Simulations:

The computational times required for simulations of the 3D TGV flow set for  $Re = 1600$  are summarized in Table 4.16. The simulations were carried out using 16 processors on an HPC system. The AMR configuration with a root mesh  $64^3$  achieved a 37% reduction in runtime compared to the uniform mesh  $256^3$ , demonstrating the potential of AMR to deliver significant computational savings while maintaining overall agreement with the uniform mesh solution.

Table 4.16: Computational times required for 3D TGV  $Re = 1600$  simulations

| Mesh                  | Computational Time | Reduction in Runtime (%) |
|-----------------------|--------------------|--------------------------|
| Uniform Mesh $256^3$  | 46 hrs 32 min      | —                        |
| AMR, Root Mesh $64^3$ | 29 hrs 22 min      | 37%                      |

#### 4.4.2.2 Comparison of Under-resolved DNS Solutions for $Re = 3000$ using Uniform and AMR Meshes

At Reynolds number  $Re = 3000$ , the Taylor–Green vortex flow begins with laminar characteristics but undergoes transition to turbulence as instabilities amplify and vortex stretching intensifies. This regime is characterized by vortex sheet roll-up, reconnection events, and the emergence of high-vorticity regions, particularly around the time of peak energy dissipation. These features make the case especially suitable for evaluating the capability of adaptive methods to resolve both transitional and turbulent dynamics.

### Grid Resolution Requirement and Adopted Strategy:

To fully resolve the turbulent scales using DNS, the minimum grid resolution estimated from Kolmogorov theory (Equation 3.12) is approximately  $N^3 \approx 405^3$ . Achieving such a resolution is computationally prohibitive in the present context. Instead, a uniform grid of  $256^3$  cells is employed as a practical compromise. While insufficient to capture the smallest dissipative scales, it provides a suitable baseline for AMR evaluation. This under-resolved DNS setup is therefore used to assess the ability of AMR to capture the essential flow features while also examining whether it can deliver computational savings compared to uniform grids.

### Kinetic Energy Decay and Energy Dissipation Rate:

The time evolution of kinetic energy shows that the AMR solution closely follows the uniform mesh result with good agreement during the early stages. A slight deviation appears beyond  $t \approx 13.5$ , likely due to adaptive de-refinement in previously refined regions, which results in coarser local resolution and a corresponding loss of accuracy.

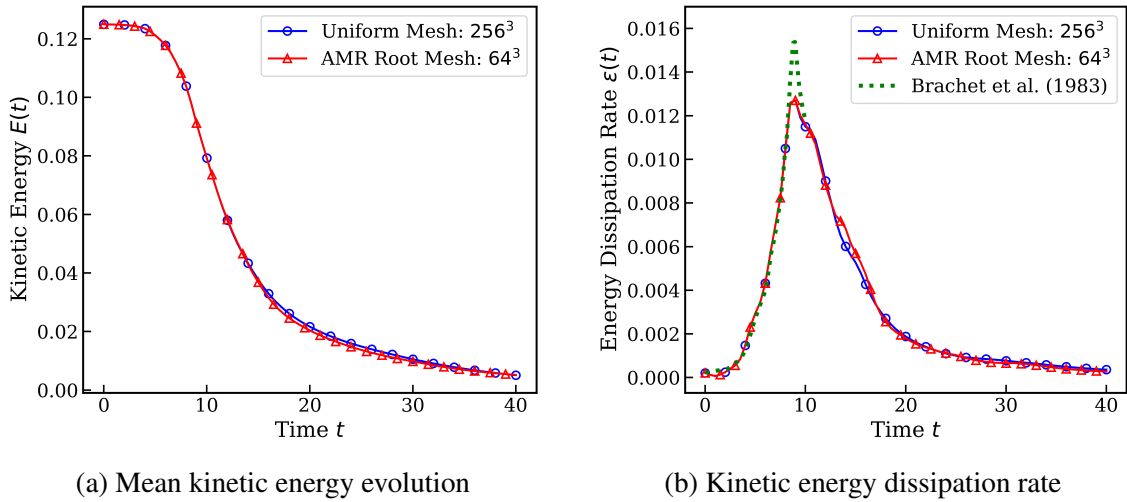


Figure 4.31: Mean kinetic energy and kinetic energy dissipation rate at different time instances for flow at  $Re = 3000$

The energy dissipation rate from the AMR mesh agrees well with the uniform mesh solution, with deviations appearing after  $t \approx 13.5$ . Both simulations predict the peak dissipation at  $t \approx 9$ , consistent with the reference solution [26], although the peak value is underpredicted due to insufficient resolution to capture the smallest dissipative scales.

### Energy Spectrum:

The kinetic energy spectral density  $E(k)$  at  $t = 9$  and  $t = 15$  is shown in Figure 4.32, with the Kolmogorov scaling law for the inertial subrange,  $E(k) \sim k^{-5/3}$ , included for reference. At both times, the spectra from the uniform mesh ( $256^3$  grid) and the AMR

solution (base grid  $64^3$  with three refinement levels) approximately follow the expected scaling over the range  $k \in [5, 50]$ . The inertial subrange is more pronounced than in lower Reynolds number cases (e.g.,  $\text{Re} = 1600$ ), reflecting the enhanced development of turbulence. Nevertheless, the full extent of the inertial range reported in Brachet et al. [26] ( $k \approx 10$  to  $k \approx 80$  for  $\text{Re} = 3000$ ) is not captured, primarily due to the under-resolved nature of the present simulations. In addition, the AMR spectra exhibit an unphysical rise in energy at high wavenumbers within the dissipation range, likely caused by interpolation errors across coarse-fine grid interfaces and instabilities introduced by changes in mesh resolution under AMR.

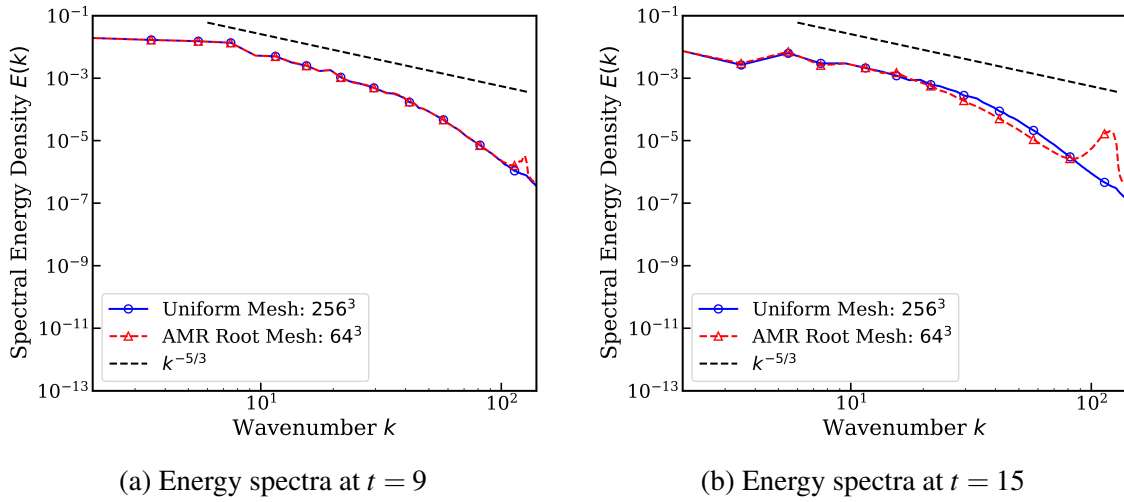


Figure 4.32: Energy spectra at  $t = 9$  and  $t = 15$  for  $\text{Re} = 3000$

#### Computational Time required for Simulations:

The computational times for the three-dimensional TGV simulations at  $\text{Re} = 3000$  are summarized in Table 4.17.

Table 4.17: Computational times required for 3D TGV  $\text{Re} = 3000$  simulations

| Mesh                  | Computational Time | Reduction in Runtime (%) |
|-----------------------|--------------------|--------------------------|
| Uniform Mesh $256^3$  | 48 hrs 20 min      | —                        |
| AMR, Root Mesh $64^3$ | 42 hrs 18 min      | 12.5%                    |

The AMR configuration with a  $64^3$  root grid and three refinement levels reduced the runtime by approximately 12.5% compared to the uniform  $256^3$  mesh, using 16 processors of the HPC system. This reduction, though modest compared to lower Reynolds number cases, demonstrates the potential of AMR to improve efficiency by selectively refining regions of interest. Further optimization of refinement strategies may allow such savings to be achieved without compromising accuracy at smaller turbulent scales.

# Chapter 5

## Conclusions

The present work has demonstrated that Athena++ can accurately and stably capture the physics of viscous incompressible flows across a range of laminar and turbulent regimes. The spatial reconstruction scheme PLM combined with the RK2 time integrator performed well for two-dimensional problems, producing velocity profiles in good agreement with benchmark solutions for both channel flow and the lid-driven cavity. The HLLC Riemann solver proved robust, and a Courant number of 0.4 ensured numerical stability. Grid convergence studies for the plane channel flow and the two-dimensional Taylor–Green vortex showed grid convergence index (GCI) consistency ratios between 1.0 and 1.2, confirming that the implementation achieves the expected order of accuracy. For three-dimensional TGV at  $Re = 100$ , the use of PPM with RK4 further validated the solver’s ability to deliver high-order accuracy, with GCI analysis yielding a consistency ratio close to unity.

Adaptive mesh refinement (AMR) was implemented in Athena++ with a refinement criterion based solely on the local grid Reynolds number,  $Re_{\text{grid}}$ . This strategy preserved fine resolution where  $Re_{\text{grid}}$  fell below the threshold while maintaining coarser meshes elsewhere. For the three-dimensional TGV at  $Re = 100$ , simulations were performed using PPM with RK4, and AMR achieved a computational time reduction of approximately 35% relative to a uniform mesh. The results for  $Re = 100$  and  $Re = 400$  showed good agreement with benchmark data, with only slight deviations at  $Re = 400$ . However, at  $Re = 400$ , no computational savings were obtained; in fact, AMR required about 16% more time compared to the uniform mesh, owing to the overhead from dynamic refinement and de-refinement.

For the under-resolved DNS of the  $Re = 1600$  three-dimensional TGV, PPM with RK4 on uniform meshes, based on the Kolmogorov resolution requirement  $N^3 \sim Re^{9/4}$ , yielded kinetic energy and dissipation rates in close agreement with benchmark solutions. When AMR was applied at  $Re = 1600$ , a computational time reduction of about 35% was achieved relative to the uniform mesh, while maintaining good agreement in the bulk evolution of flow quantities. However, the AMR spectra exhibited an unphysical rise in energy at high wavenumbers within the dissipation range, indicating limitations in accurately resolving the smallest scales. At the higher Reynolds number  $Re = 3000$ , both AMR and uniform meshes predicted the timing of the peak dissipation rate consistently

with benchmarks, although the peak value was underpredicted due to the highly under-resolved mesh. The AMR solution also followed the uniform mesh in overall energy and dissipation trends, with modest runtime savings of about 12.5%, and the corresponding spectra approximately followed the expected inertial scaling but showed an unphysical rise at the smallest scales.

These observations emphasize that particular care is required when applying AMR to turbulence simulations. Non-uniform truncation errors at fine–coarse mesh boundaries, compared to those within uniform regions, can generate numerical noise and instabilities, making the solution more sensitive to perturbations. In addition, errors in interpolation or flux correction across fine–coarse interfaces may contribute to spurious energy in the dissipation range of the spectra. Finally, standard reconstruction schemes on adaptively coarsened meshes may be insufficient to capture the smallest turbulent structures. Together, these factors highlight the need for carefully designed refinement criteria and error-control strategies if AMR is to be relied upon for accurately resolving turbulence across all scales. Nevertheless, with such improvements, AMR remains a promising approach for reducing computational cost while retaining accuracy in large-scale turbulence simulations.

# Chapter 6

## Future Work

Future improvements could focus on enhancing AMR refinement and time-stepping strategies, as well as broadening solver performance assessments. The current  $Re_{\text{grid}}$ -based refinement criterion could be augmented with additional indicators, such as velocity and vorticity gradients, to better capture turbulence and exert finer control over de-refinement. A hysteresis-based refinement/de-refinement strategy, using separate thresholds for refinement and coarsening, may also help minimize oscillatory mesh changes in marginal regions.

To reduce inaccuracies associated with time-step adjustments during de-refinement, controlled strategies could be adopted, such as delaying time-step changes or constraining them based on variations in key solution variables. Adaptive CFL numbers may also be explored to balance stability and accuracy depending on the refinement level. Remedies to avoid numerical instabilities include more accurate flux correction at fine-coarse interfaces, as well as buffer zones to smooth transitions between refinement levels. Hybrid approaches, such as coupling AMR with filtering or subgrid-scale models, may also help suppress spurious high-wavenumber energy.

Further studies could evaluate accuracy and computational cost using alternative spatial reconstruction schemes, as well as assess the effect of varying MeshBlock sizes to better understand AMR's trade-offs between resolution, cost, and stability. Solver performance could also be tested with newly proposed low-dissipation HLLC Riemann solvers to investigate potential reductions in numerical dissipation for both laminar and turbulent cases. Such improvements would enhance both the robustness and efficiency of AMR for turbulence simulations. Together, these steps would make AMR a more reliable and efficient tool for turbulence research.

# References

1. James R. Beattie, Christoph Federrath, Ralf S. Klessen, Salvatore Cielo, and Amitava Bhattacharjee. The spectrum of magnetized turbulence in the interstellar medium. *Nature Astronomy*, 05 2025.
2. Steven A. Orszag. Analytical theories of turbulence. *Journal of Fluid Mechanics*, 41:363–386, 04 1970.
3. Steven A. Orszag and G. S. Patterson. Numerical simulation of three-dimensional homogeneous isotropic turbulence. *Physical Review Letters*, 28:76–79, 01 1972.
4. John Kim, Parviz Moin, and Robert Moser. Turbulence statistics in fully developed channel flow at low reynolds number. *Journal of Fluid Mechanics*, 177:133–166, 04 1987.
5. Robert D. Moser, John Kim, and Nagi N. Mansour. Direct numerical simulation of turbulent channel flow up to  $Re=590$ . *Physics of Fluids*, 11:943–945, 04 1999.
6. Suzanne T Parete-Koon, Christopher R Smith, Thomas L Papatheodore, and E Bronson. A review of direct numerical simulations of astrophysical detonations and their implications. *Frontiers of Physics*, 8:189–198, 04 2013.
7. A Brandenburg, O Gressel, S Jabbari, N. Kleeorin, and I. Rogachevskii. Mean-field and direct numerical simulations of magnetic flux concentrations from vertical field. *Astronomy and Astrophysics*, 562:A53–A53, 12 2013.
8. James M Stone, Kengo Tomida, Christopher J White, and Kyle Gerard Felker. The athena++ adaptive mesh refinement framework: Design and magnetohydrodynamic solvers. *The Astrophysical Journal Supplement Series*, 249, 07 2020.
9. James R Stone, Thomas A Gardiner, Peter Teuben, John A Hawley, and Jacob B Simon. Athena: A new code for astrophysical mhd. *Astrophysical Journal Supplement Series*, 178:137–177, 09 2008.
10. A. Mignone, G. Bodo, S. Massaglia, T. Matsakos, O. Tesileanu, C. Zanni, and A. Ferrari. Pluto: A numerical code for computational astrophysics. *The Astrophysical Journal Supplement Series*, 170:228–242, 05 2007.
11. A. Mignone, C. Zanni, P. Tzeferacos, B. van Straalen, P. Colella, and G. Bodo. The pluto code for adaptive mesh computations in astrophysical fluid dynamics. *The Astrophysical Journal Supplement Series*, 198:7, 12 2011.

12. B. Fryxell, K. Olson, P. Ricker, F. X. Timmes, M. Zingale, D. Q. Lamb, P. MacNeice, R. Rosner, J. W. Truran, and H. Tufo. Flash: An adaptive mesh hydrodynamics code for modeling astrophysical thermonuclear flashes. *The Astrophysical Journal Supplement Series*, 131:273–334, 11 2000.
13. Romain Teyssier. Cosmological hydrodynamics with adaptive mesh refinement - a new high resolution code called ramses. *Astronomy and Astrophysics*, 385:337–364, 04 2002.
14. G L Bryan and M L Norman. Simulating x-ray clusters with adaptive mesh refinement. *arXiv (Cornell University)*, 01 1997.
15. O. Porth, C. Xia, T. Hendrix, S. P. Moschou, and R. Keppens. Mpi-amrvac for solar and astrophysics. *The Astrophysical Journal Supplement Series*, 214:4, 08 2014.
16. Alexei G Kritsuk, Åke Nordlund, David J Collins, Paolo Padoan, Michael L Norman, Tom Abel, Robi Banerjee, Lisa J Kewley, Mario Flock, Dongwook Lee, Pak Shing Li, Wolf-Christian Müller, Romain Teyssier, Sergey D Ustyugov, Christian Vogel, and Hao Xu. Comparing numerical methods for isothermal magnetized supersonic turbulence. *The Astrophysical Journal*, 737:13–13, 08 2011.
17. Kyle Gerard Felker and James M. Stone. A fourth-order accurate finite volume method for ideal mhd via upwind constrained transport. *Journal of Computational Physics*, 375:1365–1400, 08 2018.
18. I Babuska and Rheinboldt W C. Error estimates for adaptive finite element computations. *SIAM Journal on Numerical Analysis*, 15:736–754, 1978.
19. M Crouzeix and P.-A Raviart. Conforming and nonconforming finite element methods for solving the stationary stokes equations i. *Revue française d’automatique, informatique, recherche opérationnelle. Analyse numérique*, 7:33–75, 1973.
20. M J Berger and J Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53:484–512, 03 1984.
21. M.J. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics*, 82:64–84, 05 1989.
22. B. R. Munson, D. F. Young, and T. H. Okiishi. *Fundamentals of Fluid Mechanics*. John Wiley Sons, Inc., 4 edition, 2002.
23. F. M. White. *Fluid Mechanics*. McGrawHill, 4 edition, 1999.
24. U Ghia, K.N Ghia, and C.T Shin. High-re solutions for incompressible flow using the navier-stokes equations and a multigrid method. *Journal of Computational Physics*, 48:387–411, 12 1982.
25. G. I. Taylor. Lxxv. on the decay of vortices in a viscous fluid. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 46:671–674, 10 1923.



26. Marc E. Brachet, Daniel I. Meiron, Steven A. Orszag, B. G. Nickel, Rudolf H. Morf, and Uriel Frisch. Small-scale structure of the taylor–green vortex. *Journal of Fluid Mechanics*, 130:411, 05 1983.
27. Wim M. van Rees, Anthony Leonard, D.I. Pullin, and Petros Koumoutsakos. A comparison of vortex and pseudo-spectral methods for the simulation of periodic vortical flows at high reynolds numbers. *Journal of Computational Physics*, 230:2794–2805, 12 2011.
28. Ivan A. Shirokov and Tatiana G. Elizarova. Simulation of laminar–turbulent transition in compressible taylor–green flow basing on quasi-gas dynamic equations. *Journal of Turbulence*, 15:707–730, 07 2014.
29. Peter A Davidson. *Turbulence : an indroduction for scientists and engineers*. Oxford Univ. Press, 2009.
30. Lewis F Richardson. *Weather prediction by numerical process*. Cambridge Univ. Press, 2007.
31. A. N. Kolmogorov. The local structure of turbulence in incompressible viscous fluid for very large reynolds number. *Doklady Akademii Nauka SSSR*, 30:299–303, 1941.
32. Eleuterio F Toro and Springerlink (Online Service. *Riemann Solvers and Numerical Methods for Fluid Dynamics : A Practical Introduction*. Springer Berlin Heidelberg, 2009.
33. Randall LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge: Cambridge University Press; (Cambridge Texts in Applied Mathematics), 2002.
34. D Dunning, W Marts, R.W Robey, and P Bridges. Adaptive mesh refinement in the fast lane. *Journal of Computational Physics*, 406:109193–109193, 12 2019.
35. P. J. Roache. Perspective: A method for uniform reporting of grid refinement studies. *Journal of Fluids Engineering*, 116:405–413, 09 1994.

# **Appendix A**

## **Ethical Approval Letter**

The letter of ethical approval for this study, issued by the University, is included on the following page.



5 June 2025

Dear Mr Dhavale ,

Reference: CURES/25550/2025

Project ID: 29078

Title: Implementation of Direct Numerical Simulation Framework of the Finite Volume Methods based Open-Source Code Athena++ for Astrophysical Flows

We are pleased to inform you that you have successfully declared that your research project is a **Literature Review – based solely on openly available literature which is in the public domain, and you are undertaking desk-based research not involving any other form of data or information.**

You have also confirmed that your project **does not meet** any of the literature review specified exceptions, listed both within the relevant section of the CURES form and below:

1) Your supervisor has requested that you apply for approval through CURES because:

- The journals or data are in a sensitive area (please discuss this with your supervisor)
- The project will be embargoed **i.e. will not be publicly available via the Cranfield library immediately or longer term**

2) Approval is/will be specifically required by another external body e.g. journal publishers

Therefore, **you do not require ethical approval** and your CURES application will be automatically closed.

**Please keep a copy of this letter safe, if this exception is in relation to your thesis project, you will need to include a copy with your final thesis submission.**

If you have any queries, please contact CURES Support.

We wish you every success with your project.

Regards,

CURES Team