**Blue Waters Petascale Semester Curriculum v1.0**

**Unit 11: Domain Science: Astrophysical Fluid Dynamics**

**Lesson 1: Introduction to Domain Science**

**Instructor Guide**

*Developed by* *Roman Voronov and Marc Gagné for the Shodor Education Foundation, Inc.*



*Except where otherwise noted, this work by The Shodor Education Foundation, Inc. is licensed under CC BY-SA 4.0. To view a copy of this license, visit*[*https://creativecommons.org/licenses/by-sa/4.0*](https://creativecommons.org/licenses/by-sa/4.0)

*Browse and search the full curriculum at*[*http://shodor.org/petascale/materials/semester-curriculum*](http://shodor.org/petascale/materials/semester-curriculum)

*We welcome your improvements! You can submit your proposed changes to this material and the rest of the curriculum in our GitHub repository at*[*https://github.com/shodor-education/petascale-semester-curriculum*](https://github.com/shodor-education/petascale-semester-curriculum)

*We want to hear from you! Please let us know your experiences using this material by sending email to* [*petascale@shodor.org*](mailto:petascale@shodor.org)

Basic definitions as shown in the slides:

**Diffraction Limit of Light Microscopy**: In 1873, the German physicist Ernst Abbe realized that the resolution of optical imaging instruments, including telescopes and microscopes, is fundamentally limited by the diffraction of light. His finding indicated that ultimately the resolution of an imaging instrument is not constrained by the quality of the instrument, but by the wavelength of light used and the aperture of its optics. Specifically, that a microscope cannot resolve objects that are of comparable size to, or smaller than, the wavelength of light that is being used to image it.

“An introduction to super-resolution microscopy of living cells”

<https://www.youtube.com/watch?v=E-goSpv7gj8>

**Density Functional Theory**: Electronic scale simulation that performs computational quantum mechanical modelling (i.e., solves the Schrödinger equation) in physics, chemistry and materials science to investigate the electronic structure (or nuclear structure) (principally the ground state) of many-body systems, in particular atoms, molecules, and the condensed phases.

“Basics of DFT in 10 minutes”

<https://www.youtube.com/watch?v=UEOrDnwtq6s>

**Molecular Dynamics (MD)**: Atomistic scale simulation method that uses Newton's equations of motion to computationally simulate the time evolution of a set of interacting atoms. Such techniques are dependent on a description of how the molecules will interact – a force field (which is typically assumed to be represented via an explicit algebraic equation, such as the Lennard Jones potential) – and are popular in materials chemistry, biochemistry and biophysics.

“Molecular Dynamics in 5 Minutes”

<https://www.youtube.com/watch?v=veBZYlD6AF4&t=137s>

**Metropolis Monte Carlo (MC)**: Atomistic scale simulation technique to sample the infinite number of available configurations of a material by introducing random motions to the atoms and then either accepting or rejecting them according to the Boltzmann distribution.

**Coarse Grained MD/MC**: Mesoscopic simulation technique for modeling the behaviour of complex systems using their simplified (e.g.., a pseudo “atom” representing a group of atoms) representation.

**Lattice Boltzmann Method/Lattice Gas Automata**: Lattice Gas Automata are a type of cellular automaton used to simulate fluid flows, which were the precursor to the Lattice Boltzmann methods. Lattice Boltzmann methods is a very popular CFD method that treats the fluid as “particle densities”, whose motion is constrained to a lattice. Fluid properties are accounted for via the “collisions” of these particle densities. It is a very scalable method that can handle large systems with complex boundary conditions very easily.

“Of Foxes, Attackers, ... and the Lattice Boltzmann Method”

<https://www.youtube.com/watch?v=8qorVDJS1QA>

**Continuum Mechanics**: Simulation methods that solve conservation equations (e.g.., force, mass, energy balance) in order to model transport phenomena. These methods assume that matter is continuous (i.e., there are no atoms or electrons). Whether or not this is a good assumption is typically quantified using the Knudsen number. Essentially, the average travel distance that particles travel prior to experiencing collisions must be much smaller than the characteristic size of the geometry confining them, in order for the assumption to apply.

**Top-Down Machine Learning**: Pattern search methods that work by establishing associations in large sets of data. For example, neural networks are patterned after how a human brain learns. These methods do not necessarily have any knowledge of the physics occurring in the system in order to make predictions about it.

## Code and Documentation

The Astrophysical Fluid Dynamics unit utilizes PLUTO version 4.3. PLUTO is a freely-distributed software for the numerical solution of mixed hyperbolic/parabolic systems of partial differential equations (conservation laws) targeting high Mach number flows in astrophysical fluid dynamics. The code is designed with a modular and flexible structure whereby different numerical algorithms can be separately combined to solve systems of conservation laws using the finite volume or finite difference approach based on Godunov-type schemes.

Equations are discretized and solved on a structured mesh that can be either static or adaptive. The AMR interface relies on the Chombo library for parallel calculations over block-structured, adaptively refined grids.

The code is written in the C programming language while the AMR interface also requires C++ and Fortran.

PLUTO is a highly portable software and can run from a single workstation up to several thousands processors using the Message Passing Interface (MPI) to achieve highly scalable parallel performance.

The software is developed at the Dipartimento di Fisica, Torino University in a joint collaboration with INAF, Osservatorio Astronomico di Torino and the SCAI Department of CINECA.

### Documentation links:

* PLUTO 4.3 User’s Guide: <http://plutocode.ph.unito.it/userguide.pdf>
* Method Paper, static grid version: Mignone et al. 2007: <https://arxiv.org/pdf/astro-ph/0701854.pdf>
* Method Paper, adaptive grid version: Mignone et al. 2012: <https://arxiv.org/pdf/1110.0740.pdf>
* VisIt: <https://wci.llnl.gov/simulation/computer-codes/visit/manuals>
* Chombo - Software for Adaptive Solutions of Partial Differential Equations: <https://commons.lbl.gov/display/chombo/Chombo+-+Software+for+Adaptive+Solutions+of+Partial+Differential+Equations>

### Code links:

1. Module 11.1:
   1. PLUTO 4.3: <http://plutocode.ph.unito.it/download.html>
   2. Rayleigh-Taylor Instability test code: <http://plutocode.ph.unito.it/Doxygen/Test_Problems/_m_h_d_2_rayleigh___taylor_2init_8c.html>
2. Module 11.2:
   1. VisIt 3.1: <https://wci.llnl.gov/simulation/computer-codes/visit/executables>
   2. HD Jet test code: <http://plutocode.ph.unito.it/Doxygen/Test_Problems/_h_d_2_jet_2init_8c.html>
3. Module 11.3:
   1. Sedov-Taylor blast wave test code: <http://plutocode.ph.unito.it/Doxygen/Test_Problems/_h_d_2_sedov_2init_8c.html>
4. Module 11.4:
   1. MHD Blast Wave test code: <http://plutocode.ph.unito.it/Doxygen/Test_Problems/_m_h_d_2_blast_2init_8c.html>
5. Module 11.5:
   1. Chombo library: download, install, remake PLUTO: <https://commons.lbl.gov/display/chombo/Chombo+Download+Page>
   2. 3D shock cloud test code: <http://plutocode.ph.unito.it/Doxygen/Test_Problems/_m_h_d_2_shock___cloud_2init_8c.html>