ReZeus: new developments of oldtimer

Miljenko Čemeljić 席門傑

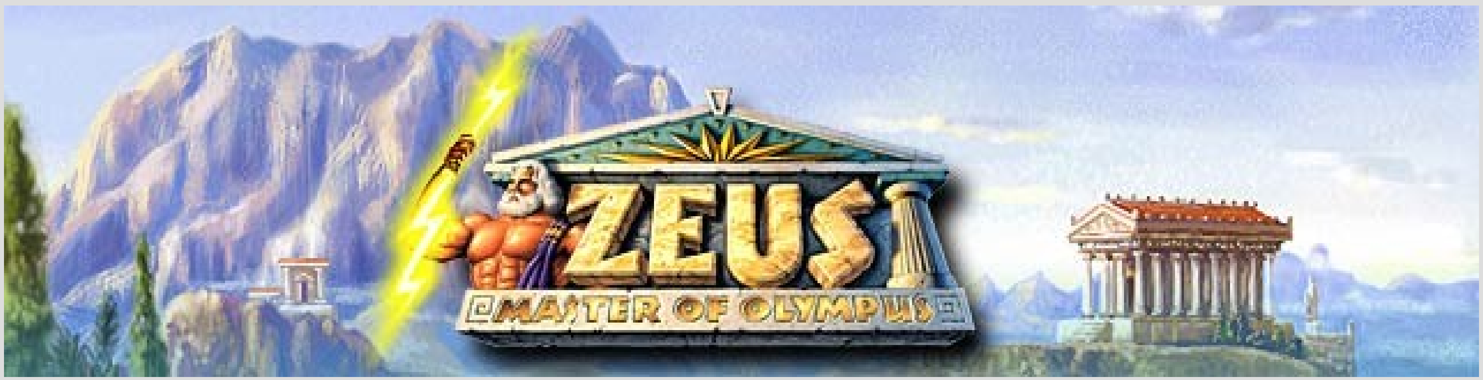
ASIAA/TIARA, Taiwan

The Third Cross-Strait Astrophysics Symposium,
Xiamen University, China, December 06, 2012

Outline

- History of Zeus code
- ReZeus
- Examples of simulations with ReZeus
- Summary

History of ZEUS code



-ZEUS family of codes (Stone & Norman 1992a, 1992b, 1992c; Clarke 1996) is one of oldest MHD codes still in use in astrophysical community. Hundreds of papers have been produced with it.

-(too) many versions, with various numerical methods implemented.

-recent refurbishing by one of original developers, gives new vitality to the original structure, by improving the boundary conditions setup, and adding new functionality

-there will be further improvements and developments, and we will see ZEUS around for next decade(s), although now it is already 20 years old.

History of ZEUS code-versions

-numerous versions of the code have been developed, including Michael Norman's ZEUSMP and the ZEUS module in ENZO; Jim Stone's ZEUS-2D and his version of ZEUS-3D.

In words of David A. Clarke in 2007:

"ZEUS-3D is a multi-physics computational fluid dynamics (CFD) code written in FORTRAN, designed primarily for, but not restricted to, astrophysical applications. It was born from the ZEUS-development project headed by Michael Norman at the NCSA in the late 1980s and early 1990s whose principal developers included myself (1986–1988; zeus04, 1990–1992; ZEUS-3D), Jim Stone (1988–1990; ZEUS-2D), and Robert Fiedler (1992–1994; ZEUSMP). Each of ZEUS-2D and ZEUS-3D (version 3.2; zeus32) were placed in the public domain in 1992, followed two years later by ZEUSMP, an MPI version of zeus32. These codes are still available from the Laboratory for Computational Astrophysics (LCA) site."

-there are many other versions, modified by subsequent users, all of them not accountable to each other.

History of ZEUS code-dzeus35

LCA version algorithms did not change much since the release of ZEUSMP.

-The only really usable version is ZEUSMP2, which is revamped ZEUSMP, after major debugging. In later versions of ZEUSMP2 there were corrected even the minor grid staggering errors which plagued all previous versions of Zeus3D code.

-David Clarke continued to develop ZEUS (Clarke, 1996; 2010), but there was no public version.

The first public version after many years was dzeus35; double precision ZEUS, v.3.5, in October 2010. It is a direct descendant of Zeus v.3.2.

-It still uses a static, staggered grid, where the scalars (ρ, e1, e2, eT) are zone-centred and components of the principal vectors (v, B) are face-centred. Secondary vectors such as the current density and the induced electric field are edge-centred. Thus, fluxes for zone-centred quantities are computed directly from the time-centred velocities at zone interfaces; no interpolations need be performed and no characteristic equations need be solved to obtain these velocities.

-In next versions Adaptive Mesh Refinement; AMR is envisioned.

-dzeus35 is OpenMP (not MPI!) version.

History of ZEUS code-dzeus35

Some technical points, to expose the difference between Zeus and other codes:

-Besides simplicity in coding, the principal advantage of the staggered mesh is that solenoidal condition on the magnetic field ($\nabla \cdot B = 0$) is maintained trivially everywhere on a 3-D grid, to within machine round-off error.

-Zone-centred schemes typically require a “flux cleansing” step or a diffusion step (e.g., FLASH) to eliminate or reduce numerically driven magnetic monopoles after each MHD cycle. This can introduce errors when used for some problems.

-ZEUS-3D is upwinded in the flow and Alfven waves and stabilised on compressional waves (sound waves in HD, fast and slow magnetosonic waves in MHD). Stabilisation is ensured by von Neumann-Richtmyer artificial viscosity.

The advantages of the ZEUS algorithm:

-speed (faster than Godunov codes)

-robustness in multi-dimensions

-ability to accommodate additional physics such as viscosity, radiation, self-gravity, etc., without adversely affecting its underlying (M)HD algorithm.

Equations which dzeus35 can solve

$$\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\vec{v}) = 0; \quad (1)$$

$$\frac{\partial\vec{s}}{\partial t} + \nabla\cdot\left(\vec{s}\vec{v} + (p_1+p_2+p_B)\mathsf{I} - \vec{B}\vec{B} - \mu\,\mathsf{S}\right) = -\rho\nabla\phi; \quad (2)$$

$$\frac{\partial e_1}{\partial t} + \nabla\cdot(e_1\vec{v}) = -p_1\nabla\cdot\vec{v} + \mu\mathsf{S}:\nabla\vec{v} - \mathcal{L}; \quad (3)$$

$$\frac{\partial e_2}{\partial t} + \nabla\cdot(e_2\vec{v} - \mathsf{D}\cdot\nabla e_2) = -p_2\nabla\cdot\vec{v}; \quad (4)$$

$$\frac{\partial e_T}{\partial t} + \nabla\cdot\left[(e_T+p_1+p_2-p_B)\,\vec{v} - \mu\mathsf{S}\cdot\vec{v} - \mathsf{D}\cdot\nabla e_2 + \vec{E}\times\vec{B}\right] = -\mathcal{L}; \quad (5)$$

$$\frac{\partial\vec{B}}{\partial t} + \nabla\times\vec{E} = 0, \quad (6)$$

ρ is the matter density;
$\vec{v}$ is the velocity;
$\vec{s}$ is the momentum density $= \rho\vec{v}$;
p_1 & p_2 are the partial pressures from the first and second fluids;
p_B is the magnetic pressure $= \frac{1}{2}B^2$;
I is the unit tensor;
$\vec{B}$ is the magnetic induction (in units where $\mu_0 = 1$);

μ is the shear viscosity;
S is the viscid part of the stress tensor whose elements, S_{ij}, are given by:

$$S_{ij} = \partial_j v_i + \partial_i v_j - \tfrac{2}{3}\delta_{ij}\nabla\cdot\vec{v};$$

where ∂_i indicates partial differentiation with respect to the coordinates, x_i, $i = 1, 2, 3$, and where δ_{ij} is the usual "Kronecker delta";
ϕ is the gravitational potential, $\nabla^2\phi = 4\pi\rho$, in units where $G = 1$;
e_1 & e_2 are the internal energy densities of the first and second fluids;
$\mathcal{L}$ is the cooling function (of ρ and e_1) for nine coolants (HI, HII, CI, CII, CIII, OI, OII, OIII, SII) interpolated from tables given by Raga *et al.* (1997);
D is the (diagonal) diffusion tensor;
e_T is the total energy density $= e_1 + e_2 + \frac{1}{2}\rho v^2 + \frac{1}{2}B^2 + \phi$;
$\vec{E}$ is the induced electric field $= -\vec{v}\times\vec{B}$;

$$p_1 = (\gamma_1 - 1)e_1; \qquad p_2 = (\gamma_2 - 1)e_2,$$

with γ_1 and γ_2 being the ratios of specific heats for the two fluids.

-Radiation MHD using the flux-limited diffusion approximation (e.g., Turner & Stone, 2001) is only partially implemented. The Raga cooling functions are available, but are not yet fully debugged.
-one can use either the internal energy equation 3, or the total energy equation 5. In the former case internal energy density e_1, is strictly positive-definite, in the latter case total energy e_T is conserved.

*MC,added 13.06.2016: Use of ZEUS-3D, developed by D. Clarke at the ICA (http://www.ica.smu.ca) with support from NSERC, is acknowledged.

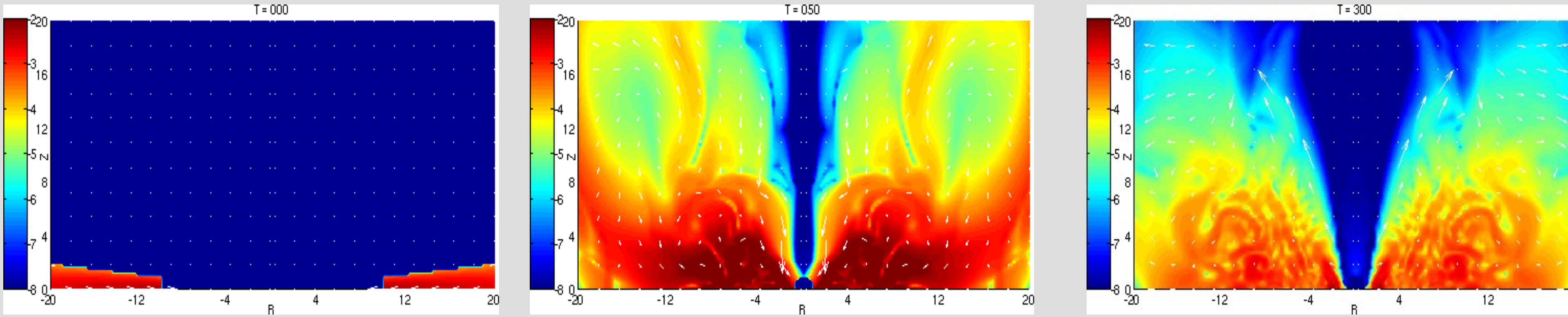
My version: dzeus36+resistivity=ReZeus

For my version I used version dzeus36, which is not yet public, but is only a modification, “work in progress” of dzeus35.

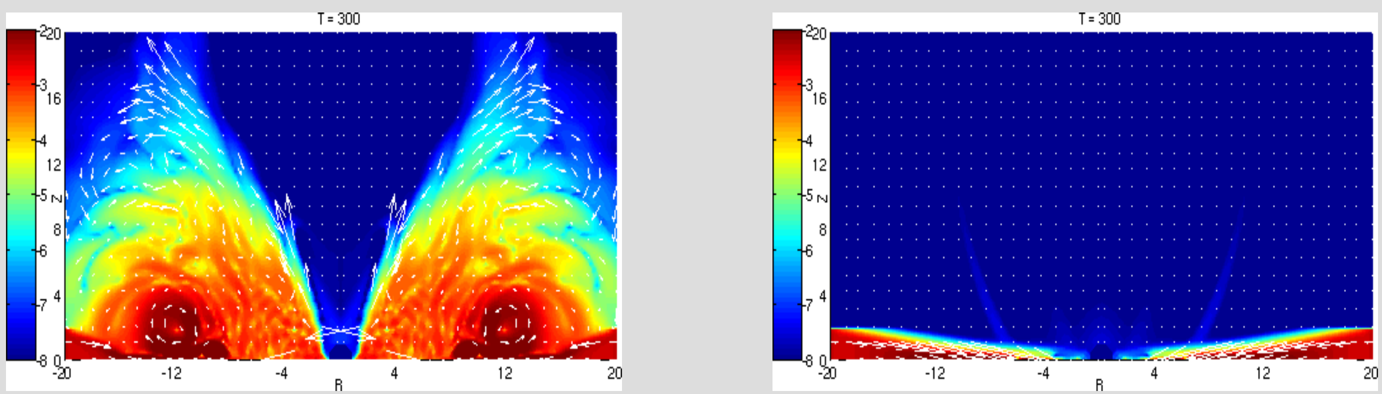
- I added electrical resistivity to this new version, and will present some preliminary results obtained by my version, which I call ReZeus.
- Resistivity η is added the same way as in Zeus347, subtracting ηj from the appropriate emfs. It is currently included only in the induction equation, and not in the energy equation(s), so that Ohmic heating is neglected.
- Resistive timestep has been added.

- Taiwan Rhesus monkey, the only primate in the island except humans.

ReZeus in a HD mode: star-disk problem



- There was a persisting problem in Zeus347, my previous version, based on Zeus-3D v.3.4.2, with the disk heating. The disk would puff too much (in figures above are mass fluxes), and it was not because of MoC (the usual suspect), as it was puffing-up even in the pure HD mode. I did not found the exact reason, but I had pinned it down to heating introduced by the code, probably because of use of internal, and not total energy.

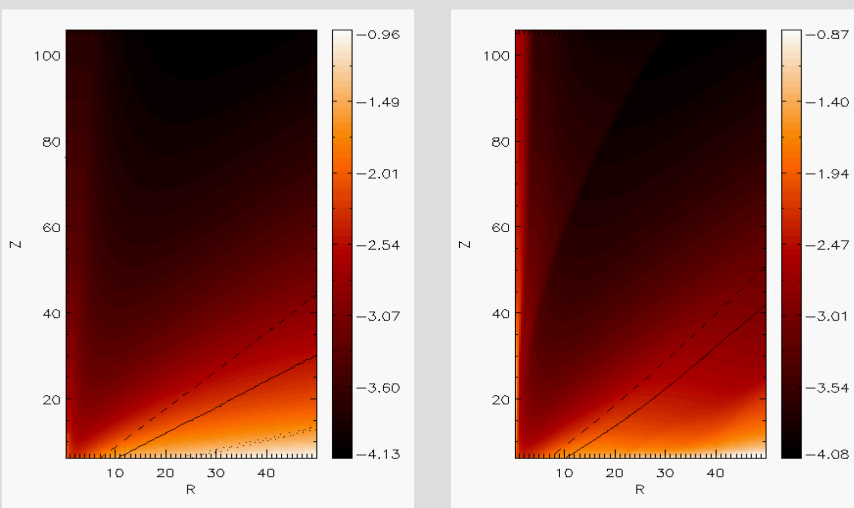


- In ReZeus, or more exactly, dzeus36, solutions with internal energy (left panel) look similar to Zeus347, but with total energy equation solved, the disk looks as it should (right panel), no puffing-up.

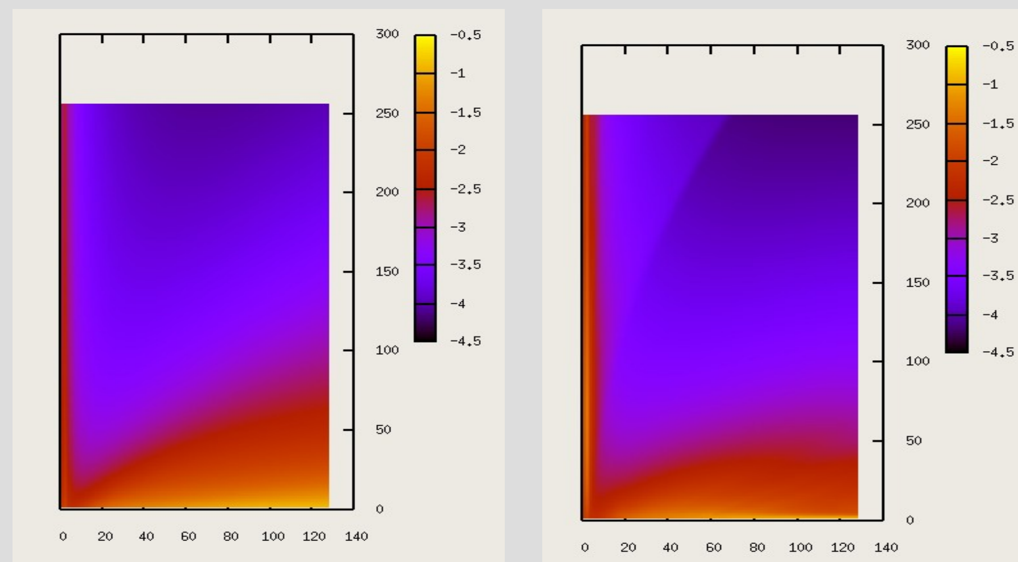
Test of ReZeus in MHD mode: radially self-similar outflow

- In Cemeljic et al. (2008) we reported an extension of radially self-similar exact numerical simulations of N. Vlahakis (2000) in the resistive-MHD simulations, using NIRVANA code. Matching of solutions was very good, but not so in Zeus-3D v.3.4.2. I had to switch to NIRVANA code, which early version 2.0 was similar to Zeus-3D, just written in C. Later I switched to PLUTO code, which is different (it uses Godunov scheme).

P L U T O



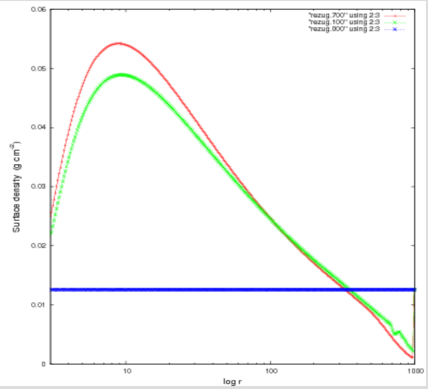
R e Z e u s

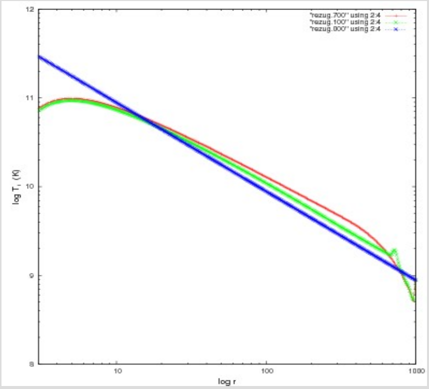


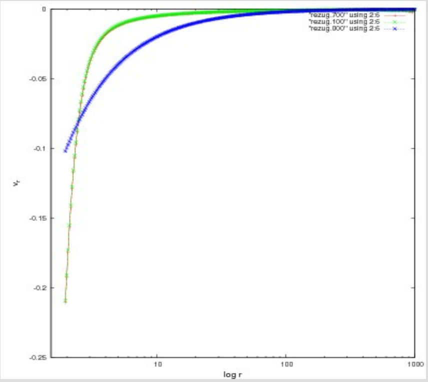
- ReZeus, (and dzeus36 for ideal-MHD case), with total energy option, solves the problem equal to Nirvana and PLUTO codes. Left panel shows matter density in initial conditions, and right panel is a result of ideal-MHD simulation after about 10 000 numerical time-steps, when the solution is relaxed and reaches stationary state. It differs in detail from initial state, because of modifications needed for numerical setup, but the critical surfaces in the flow match well with the initial ones.

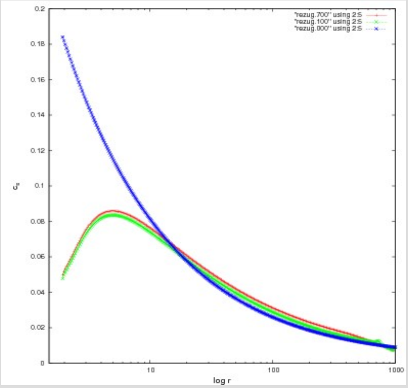
ReZeus as a solver in 1D magnetized advective accretion disk around BH (Shubhrangshu Ghosh)

- two temperature magnetized accretion flow model
- solved vertically averaged fluid eqs., heavily modifying the code. Work in progress.
- Here we show solutions for M_BH=10^9 M_sun without the magnetic field and for a single temperature, when the system reaches a quasi-stationary state, at 7x10^6 (red line). Initial setup is shown in blue line, and an intermediate step at t=5x10^5 in green line. r is expressed in units of Schwarzschild radius.
- 1) variation of the surface density with radial distance. 2) variation of the ion temperature with radial distance, 3) variation of radial velocity in units of c along r, 4) variation of sound speed in units of c along r

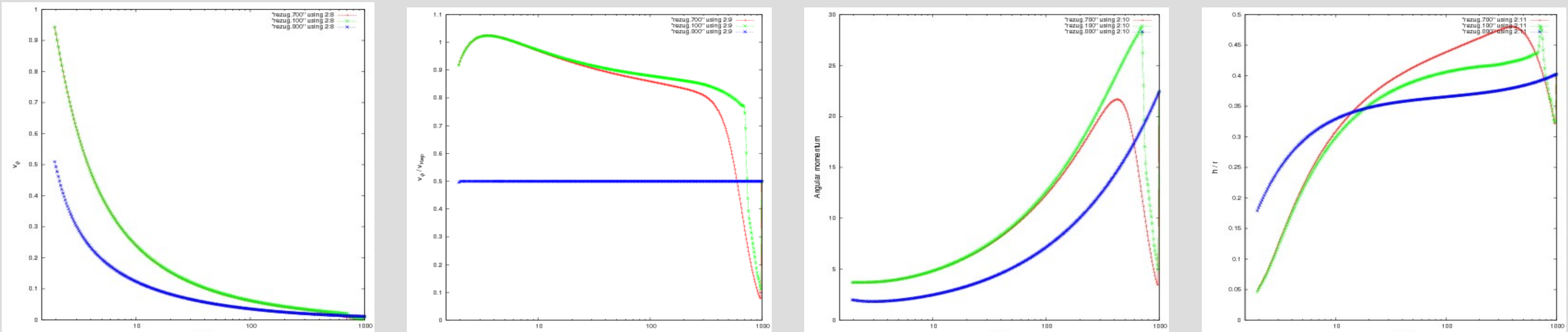








ReZeus as a solver in 1D magnetized advective accretion disk around BH (Shubhrangshu Ghosh)



- 5)Variation of azimuthal velocity in units of c along r, 6) Variation of the ratio of azimuthal velocity to Keplerian velocity along r, 7) Variation of the angular momentum in units of GM_{BH}/c along r, 8) variation of the ratio of accretion flow thickness to radius along r. Other parameters are as:
- \dot M = 10^{-3} {\dot M_{Eddington}} and \alpha = 0.01
- \dot M_{Eddington} = 1.39 \times M_{BH}/M_{Sun}
- Our initial conditions are ion temperature T_i= 0.2 * T_virial, Sigma (vertically integrated density) = 0.012589254 (taken from Nakamura et al. 1997, PASJ, 49, 503), angular momentum L = 0.5 * L_Keplerian
- matching with Nakamura is good, work on solutions with magnetic field is under way.

Summary

- ZEUS code is alive and kickin’
- Recent official version-dzeus35-which is to evolve further-already includes some interesting numerical methods.
- Test in HD mode for comparison with results from Zeus-3D v.3.4.2 shows improvements (star-disk setup).
- Test in MHD mode for comparison with results from other codes (NIRVANA, PLUTO) also shows improvements.
- We show how modified ReZeus in 1D can be used as a good solver for MHD equations.