ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ
KATEDRA TEORETICKÉ ELEKTROTECHNIKY

AUTOREFERÁT

DISERTAČNÍ PRÁCE

Large-scale Numerical Simulations of Magnetohydrodynamics Phenomena in Astrophysics

PLZEŇ 2018

Mgr. Lukáš Korous

ZÁPADOČESKÁ UNIVERZITA V PLZNI
FAKULTA ELEKTROTECHNICKÁ
KATEDRA TEORETICKÉ ELEKTROTECHNIKY

Autoreferát disertační práce k získání akademického titulu „Doktor“

Mgr. Lukáš Korous

Large-scale Numerical Simulations of Magnetohydrodynamics Phenomena in Astrophysics

PLZEŇ 2018

Mgr. Lukáš Korous

Disertační práce byla vypracována v kombinovaném doktorském studiu na Katedře teoretické elektrotechniky (KTE)

Uchazeč: Mgr. Lukáš Korous
Fakulta Elektrotechnická
Katedra Teoretické Elektrotechniky
Univerzitní 26, Plzeň

Školitel: prof. Ing. Pavel Karban, Ph.D.
Fakulta Elektrotechnická
Katedra Teoretické Elektrotechniky
Univerzitní 26, Plzeň

Oponenti: ....
....
....
....

Datum státní doktorské zkoušky: 7.11.2016
Datum odevzdání práce: 19.9.2018
Autoreferát byl rozeslán dne:

Obhajoba disertační práce se koná dne před komisí v oboru
………………………………………………………………
na Fakultě Elektrotechnické Západočeské univerzity v Plzni, Univerzitní 26,
v místnosti ………………………………. v ………………………………

S disertační prací je možno seznámit se na pracovišti Oddělení výzkumu a vývoje v budově FEK v Plzni, Husova 11, v místnosti HJ 203 u Ing. Barbary Trojanowské.

PLZEŇ 2018 Mgr. Lukáš Korous

Annotation

Abstract (English)

The objective of this Doctoral Thesis was to develop, implement and test new algorithms for the large-scale solution of nonstationary compressible MHD equations based on higher-order discontinuous Galerkin (DG) methods. The basis for the new methods will be the discontinuous Galerkin methods and adaptive mesh refinement (AMR) algorithms. The new algorithms will be implemented and tested in the framework of the open source library deal.II, and they will be applied to selected existing benchmarks for magnetohydrodynamic phenomena and real problems of MHD in astrophysics.

Keywords

numerical simulation, finite element method, MHD equations, adaptivity, discontinuous Galerkin method, astrophysics, solar flares, AMR, distributed computing

Abstract [Česky]

Záměrem této práce je navrhnout, implementovat a otestovat nové algoritmy pro rozsáhlé simulace nestacionárních jevů spadajících do oblasti stlačitelné magnetohydrodynamiky.
Vytvořený software bude založen na použití nespojité Galerkinovy metody (discontinuous Galerkin, DG) s vyššími řády přesnosti. Zároveň bude použita metoda automatického zjemňování výpočetní triangulace (automatic mesh refinement, AMR).
Vytvořené algoritmy budou testovány ve frameworku deal.II a
budou aplikovány na existující benchmarky pro magnetohydrodynamické jevy a na skutečné problémy v astrofyzice.

Keywords [Česky]

numerická simulace, metoda konečných prvků, MHD rovnice, adaptivní algoritmy, nespojitá Galerkinova metoda, astrofyzika, sluneční erupce, AMR, distribuované výpočty

Contents

1 Introduction

The term magnetohydrodynamics (MHD) covers all physical phenomena that involve both electromagnetic (EM) field and a fluid that carries the EM field. Such phenomena are very interesting, yet very complex to study. The behavior of such a fluid is utilized in some industrial applications - liquid-metal cooling of nuclear reactors, magnetic fluid in dampers, sensors for precise measuring of angular velocities, etc. Such phenomena occur in nature as well - the most significant of which are the processes that take place inside and on the surface of stars - which is the topic of the next section.

1.1 Magnetohydrodynamics in Astrophysics

There are several phenomena in the universe that we can look at as magnetohydrodynamic in nature - planets consisting of metals, interplanetary space, but mainly - stars. If we talk about the nearest star - and the only one we are able to study well enough - the Sun - these phenomena include those that occur in the Sun's photosphere (the layer of Sun that is visible): Sun spots, but also phenomena that occur above the Sun (further from the center of the Sun): in Sun's chromosphere, or even corona (solar flares) - even phenomena that originate from the Sun, but then spread through our solar system - solar winds, space weather. All these phenomena have a large impact on the lives of all of us. For example, solar flares (that are often followed by ejection of mass out of the Sun - the so-called coronal mass ejections - CMEs) have impact on the Earth's magnetic field which in turn has impact on the electronic communication down on Earth (because the communication satelites used for transmissions may be damaged by the disturbances in the magnetic field). Also, people operating at high altitudes, both in airplanes and manned space missions are exposed to the energetic particles coming from the Sun (this term is sometimes called cosmic rays). For all the above reasons, it is of great importance to understand the phenomena of space weather, and other MHD phenomena that occur in space.

Magnetic flux tube model of plasma

We are interested in the process of evolution of the flux tube eruption. Such an eruption was captured during observations by Kotrč et al. (2012), and the captured image is shown in Figure 1.1 taken from Kotrč et al. (2012) (Figure 2 in the article).

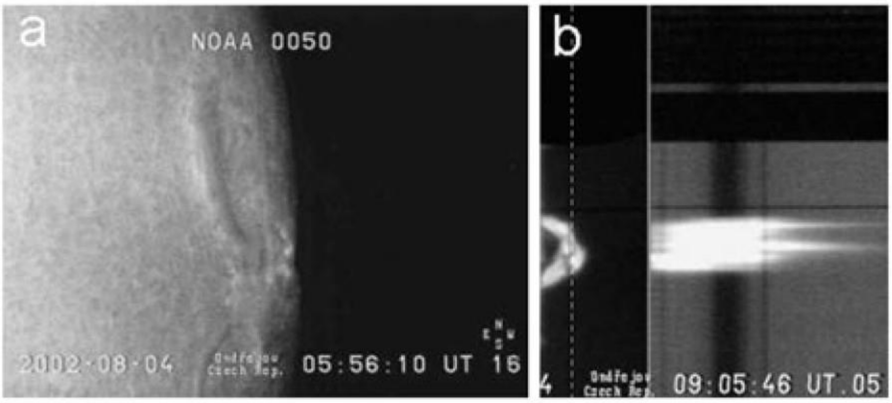


Figure 1.1: Observation of a limb event. $H\alpha$ slit-jaw is the middle (b) part, taken from Kotrč et al. (2012).

For the modeling purposes, we are interested in the so-called Hα slit-jaw which is the side-view of the magnetic flux tube during its evolution.

In order to model this phenomena physically, and geometrically, we utilize the magnetic field model by Titov and Demoulin (Titov and Demoulin (1999)) which describes a twisted flux tube as part of a torus with minor radius a (being the radius of the tube, not shown in Figure 1.2) and major radius R submerged below the photosphere of the Sun by $d < R$, oriented as in Figure 1.2 (taken from Titov and Demoulin (1999)) with total current I.

The magnetic configuration is kept in a global equilibrium by the action of the Lorentz force due to the overlying magnetic field. The sources of this ambient field are modeled by a sub-photospheric line current I0 and a pair of magnetic charges

$+q$, $-q$ located at distance L from the center of the torus, all located at the major axis of the torus.

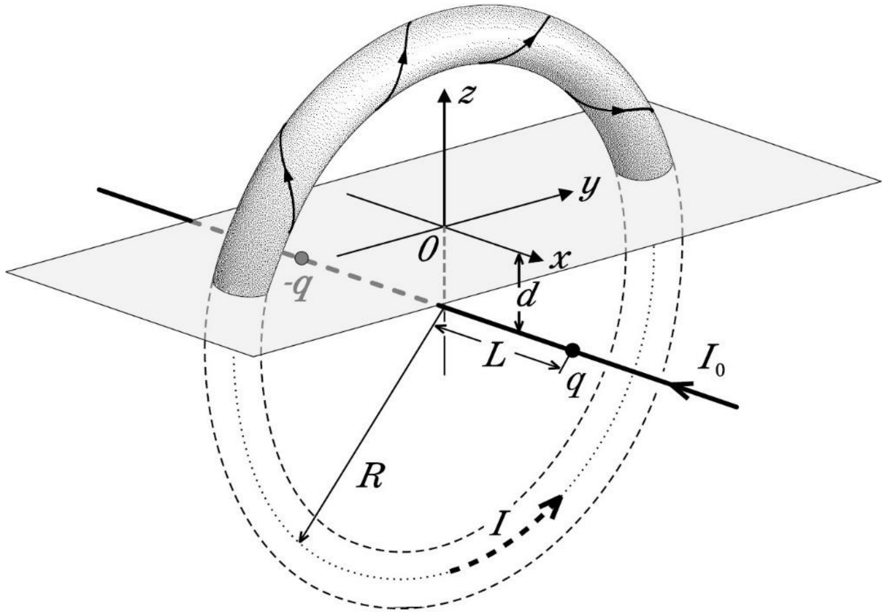


Figure 1.2: The magnetic field under study, taken from Titov and Demoulin (1999).

This geometrical/physical model needs to be properly modeled mathematically (including proper boundary conditions), then approached numerically, and finally computed using software that - to achieve reasonable accuracy of the solution - needs to be precisely implemented, must utilize approaches common in the high-performance computing (HPC) field, and must be heavily optimized.

In the article Kotrč et al. (2012) which is a reference paper for this work, the model was set up according to observations, and numerical approach of Finite Difference Method (FDM) was used. The results obtained there are presented in Figures 1.3 and 1.4, and to compare them with Figure 1.1, also in the form in Figure 1.5 - where the colors are mapped to black and white to be comparable with the observation.

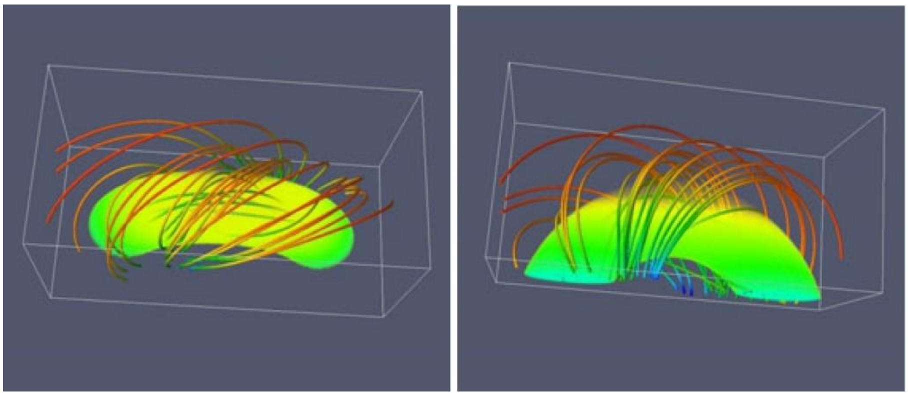


Figure 1.3: Results from Kotrč et al. (2012), initial state, density volume and magnetic field isolines - top & side view.

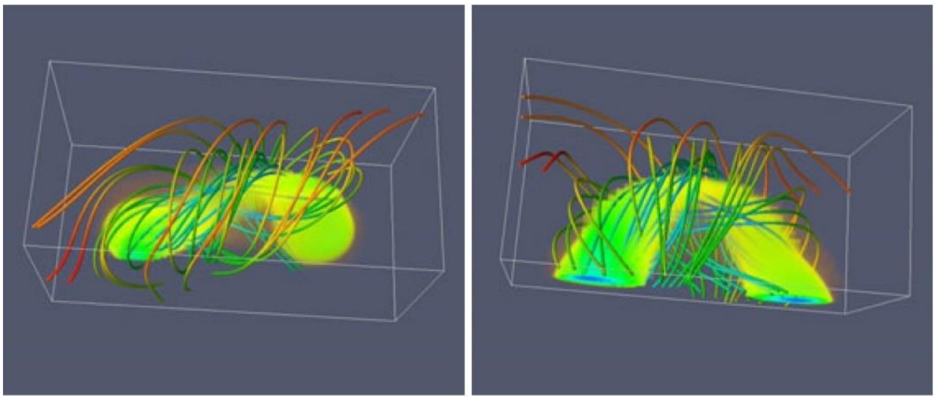


Figure 1.4: Results from Kotrč et al. (2012), state at $t = 14$, density volume and magnetic field isolines - top & side view.

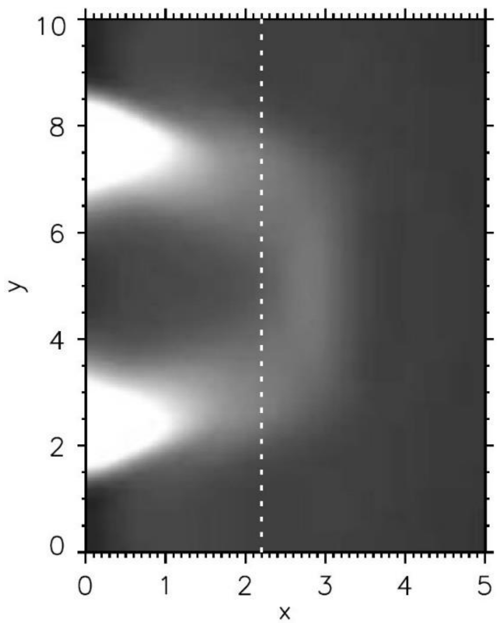


Figure 1.5: Computed approximation of the observed event (Figure 1.1), presented in Kotrč et al. (2012).

The aim of this work is to be able to compute results of such problems with much higher resolution, and more importantly to implement a generic solver which can handle any similar problem with arbitrary geometric and physical parameters.

Magnetic reconnection and other phenomena

An additional topic, interesting from the astrophysical point of view and related to magnetohydrodynamics, is the magnetic reconnection.

From the numerical perspective, to investigate such phenomena is to bring the complexity of multiple scales present in the physical world - magnetic reconnection occurs at substantially different scales than e.g. the solar flares. To handle this numerically with adequate resolution, and reasonable computational costs, the implementation must be able to limit the number of degrees of freedom used for the discretization to such that yield the largest accuracy increase - namely, in this work, this is achieved via a state-of-the-art Adaptive Mesh Refinement approach.

2 Aim of this work

Main goal that was set for this work is implementation of a software package, capable of numerically solving the magnetohydrodynamic equations with the following attributes:

- The code must implement mathematically correct and clean methods, prefer- ably with no parameter-dependent algorithms
- The code must allow for a broad spectrum of solution attributes occurring (shocks, oscillations, high energies, ...)
- The code must be able to capture very ne details of the solution through higher resolution, in reasonable computing time
- The code must be able to run a large-scale simulation, for which a single computer is not enough, and utilization of modern distributed computing approach is a must
- The code must be easy to use for the physicists, must be written using industry-standard modern object-oriented programming language

To achieve this, the work is divided into several logically successive steps. First, the mathematical model is clearly constructed, and its formulation translated into a form suitable for the chosen numerical method (the Discontinuous Galerkin method) of solving a system of PDEs - which was chosen to deliver the first and second of the above requirement (handling sharp fronts, discontinuities and the like with mathematical cleanliness) - is described afterwards.

The Discontinuous Galerkin method (DG) is then described in detail, in the entire process from the integral equations down to algorithms performing basic operations. Pitfalls of the numerical solution of the MHD equations (divergence constraint, slope limiting), and chosen way of solving them is given next, while keeping the above requirements in mind.

Lastly, the biggest challenge is the satisfaction of the requirement of both the ne resolution of the solution, and the capability of running large-scale simulations on modern distributed computing architecture - while still providing all the other attributes of the implemented code. The approach to this, which involves mainly the Adaptive Mesh Refinement technique (AMR) together with Domain Decomposition technique, is presented after. And finally, the verifications, and benchmarks and actual usage on an astrophysical problem are presented.

3 State of the Art

Only recently, the scientific computation community, due to the advances in computer and supercomputer capabilities, has started with non-trivial numerical simulations of such complex physical phenomena that the MHD model describes. Since both for industrial applications, and obviously for astrophysical application of the MHD model, it is quite expensive (or downright impossible) to perform any experiments, the bene t of being able to simulate the phenomena on a computer is very large.
There exist several available numerical simulation codes, such as (Stone et al., 2008a), (Norman et al., 1992), (Kestener et al., 1992), (Skála, J. et al., 2015). These codes have been successfully applied to a range of problems in astrophysics.

There are many numerical methods implemented in these codes, such as the finite difference method ((Skála, J. et al., 2015)), finite volume method ((Kestener et al., 1992)), and the (continuous) finite element method ((Skala and Barta, 2012)).

There have been some attempts to employ also the discontinuous Galerkin method ((Rossmanith, 2013), (Mocz et al., 2014)), but so far, no open-source generic software employing this method is available. Moreover, as the astrophysical interest lies in multi-scale problem, a software that can handle such problems would be much more beneficial. An approach that can achieve this capability of solving multi-scale problems is the Adaptive Mesh Refinement technique (AMR). What we understand under this term is not only a mesh refinement that is local (i.e. non-uniform), but a mesh refinement that does not originate in the problem description, and neither is invoked programmatically with user input. The term 'adaptivity' means that through a predefined refinement indicator, which is a function operating on the set of elements of the triangulation, elements to be refined are chosen automatically. This refinement indicator is calculated from the solution, and in effect makes the triangulation 'adapt' to the solution - hence, AMR.

The reason for the development of a new code is two-fold. First, there is a unique collaboration between the Astronomical Institute of the Czech Academy of Sciences and the University of West Bohemia, where astrophysicists work together with electrical engineers (from theoretical and numerical modeling backgrounds), and the developed code will be usable for both simulating of astrophysical MHD phenomena, and industrial MHD applications.

Second, the newly developed code is based on locally-adaptive Discontinuous Galerkin method, which yields several advantages over the existing codes (which use e.g. finite difference, or finite volumes methods) developed at institutions of such high quality as Princeton - (Stone et al., 2008a), (Norman et al., 1992). The advantages are especially of performance, and automation nature - method of higher order together with AMR yields results qualitatively and quantitatively comparable to low order uniform mesh methods, but with computational cost that can easily be an order of magnitude smaller. Automation is mentioned here related to the AMR, which, without user interaction, can optimize the computational triangulation for a particular time instance in the evolution of the modeled phenomena.

Another bene t (namely over (Norman et al., 1992)) of the newly created soft- ware are the use of modern object-oriented programming techniques and experience gained on creating finite element software ((Solin et al., 2014), (Ma et al., 2012), (Korous and Solin, 2012)). The implementation related to this work is written in the C++ language, with the use of existing software packages that are proven, and used by a wide community of researchers all over the world - deal.II ((Bangerth et al., 2015)), Trilinos (Heroux et al. (2005)), P4EST (Burstedde et al. (2011)), Intel Parallel Studio (Intel Corporation (2017)), UMFPACK ((Davis, 2006)), Paraview(Kitware Inc. (2017)), and others.

The state of the art of numerical simulation of magnetohydrodynamics can be summarized as a state when the mathematical theory of the equations is quite solid, but the methods to solve the equations numerically in the most optimal and fast way are still being improved. The numerical solution is not merely about theoretical convergence rates and attributes of the method, but also the actual implementation plays an important role - i.e. programming, hardware, and software, and execution, both before, during, and very importantly after the actual method invocation (of the so-called postprocessing of results). In all aspects of implementation, there is space for new approaches, new ideas, new milestones, that can expand the capabilities of today's numerical solution of magnetohydrodynamics phenomena.

4 Content summary

A short summary / outline of the content of the work is given in the following several paragraphs.

4.1 Mathematical model

The equations of magnetohydrodynamics are derived and studied in Section 2.1 of the work, in the form

$$\begin{aligned}
\frac{\partial \rho}{\partial t} &= -\nabla \cdot (\boldsymbol{\pi}), \\
\frac{\partial \boldsymbol{\pi}}{\partial t} &= -\nabla \cdot (\boldsymbol{\pi} \otimes \boldsymbol{u}) + \nabla \cdot \left(\frac{1}{\mu_0} \boldsymbol{B}\boldsymbol{B} - \frac{B^2}{2} \mathrm{I} \right) + \rho \boldsymbol{g} - \nabla p, \\
\frac{\partial U}{\partial t} &= -\nabla \cdot \left[\boldsymbol{\pi} \left(\frac{u^2}{2} + e \right) + p\boldsymbol{u} - \frac{1}{\mu_0} \boldsymbol{B} \times \boldsymbol{E} \right] + \rho \boldsymbol{g} \cdot \boldsymbol{u} \\
\frac{\partial \boldsymbol{B}}{\partial t} &= -\nabla \times (\boldsymbol{u} \otimes \boldsymbol{B} - \boldsymbol{B} \otimes \boldsymbol{u}) + \frac{1}{\mu_0 \sigma} \left(\nabla^2 \boldsymbol{B} \right).
\end{aligned}$$

1 - MHD equations

Here, the quantities have their standard meanings, and these are given in the notation overview in Table 0.1 in the work.

For subsequent considerations, the above equations are transformed into the form

$$\frac{\partial \mathbf{\Psi}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{\Psi}) = \mathbf{S},$$

2 - Conservative form

called the *conservation form of MHD equations*, where

$$\mathbf{\Psi} = \begin{pmatrix} \rho \\ \pi_1 \\ \pi_2 \\ \pi_3 \\ U \\ B_1 \\ B_2 \\ B_3 \end{pmatrix},$$

$$\mathbf{F}_i = \begin{pmatrix} \pi_i \\ \frac{\pi_1 \pi_i}{\rho} - B_1 B_i + \frac{1}{2}\delta_{1i}\left(p + U_m\right) \\ \frac{\pi_2 \pi_i}{\rho} - B_1 B_i + \frac{1}{2}\delta_{2i}\left(p + U_m\right) \\ \frac{\pi_3 \pi_i}{\rho} - B_1 B_i + \frac{1}{2}\delta_{3i}\left(p + U_m\right) \\ \frac{\pi_i}{\rho}\left(\frac{\gamma}{\gamma - 1} p + U_k\right) + 2\eta\varepsilon_{ijk} J_j B_k + \frac{2}{\rho}\varepsilon_{ijk}\left(\pi_k B_i - \pi_i B_k\right) B_j \\ \frac{\pi_i B_1 - \pi_1 B_i}{\rho} + \eta\varepsilon_{1ij} J_j \\ \frac{\pi_i B_2 - \pi_2 B_i}{\rho} + \eta\varepsilon_{2ij} J_j \\ \frac{\pi_i B_3 - \pi_3 B_i}{\rho} + \eta\varepsilon_{3ij} J_j \end{pmatrix},$$

$$\mathbf{S} = \begin{pmatrix} 0 \\ \rho g_1 \\ \rho g_2 \\ \rho g_3 \\ \boldsymbol{\pi} \cdot \boldsymbol{g} \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

3 - Conserved variables, conservative fluxes, right-hand side vector

are the conserved variables, conservative fluxes, and the right-hand side vector respectively.

Weak formulation

Due to the nature of numerical methods the work aims at applying, the strong formulation of the equations is not suitable, and therefore a weak formulation, yielding a solution in a Sobolev functional space, is needed.

Details of this approach are given in Section 2.2 of the work.

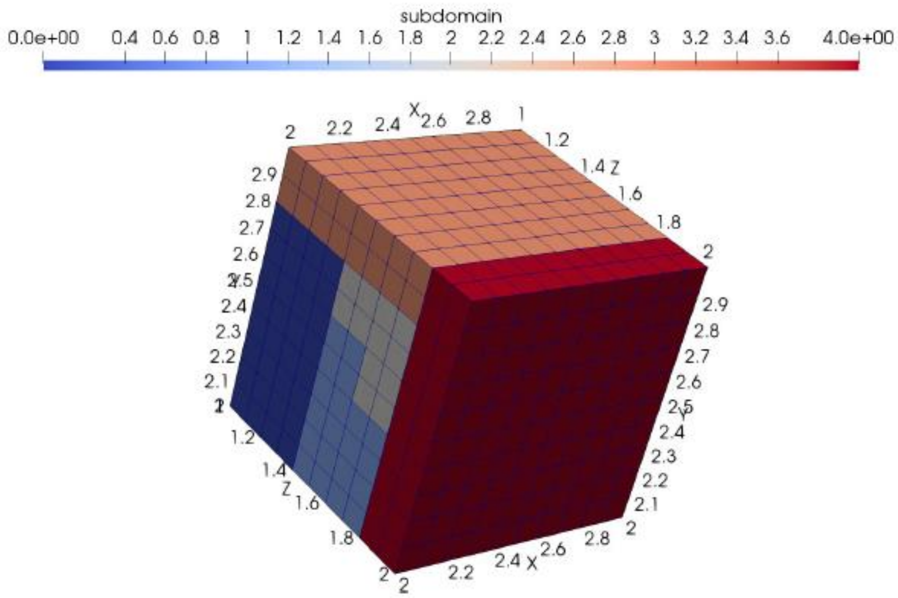
The subsequent Section 2.3 contains description about boundary conditions.

4.2 Numerical model

This, together with the section about Adaptive Mesh Refinement (AMR) is the core of the work, transforming above analytical equations into an algorithm, whose results could be transformed into the solution of the original equations.

Triangulation, distributed triangulation

Details on this topic are given in Section 3.1, and complexities arising from distributing the triangulation to multiple processors on multiple machines are discussed – which is a must to achieve the scalability required for the solution of problems the work aims at solving. The following figure shows an example of a distributed triangulation.



4 – A cubical domain with color-coded processors owning the particular element

Discontinuous Galerkin method

Details on this topic are given in Section 3.2 – with the important aspects – broken Sobolev space, numerical flux, and handling of boundary conditions, all being discussed.

Comparison of several options for the numerical flux – which has a large impact on the simulation results quality – is given in Section 3.2.3, and an example is present in the figure below.

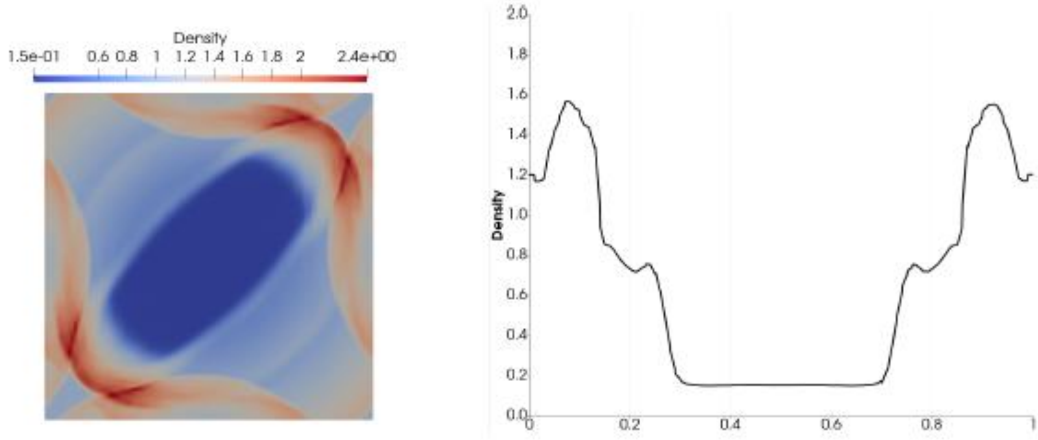


Figure 3.11: Density at time $t = 0.3$, Lax-Friedrichs flux with $\alpha = 0.5$

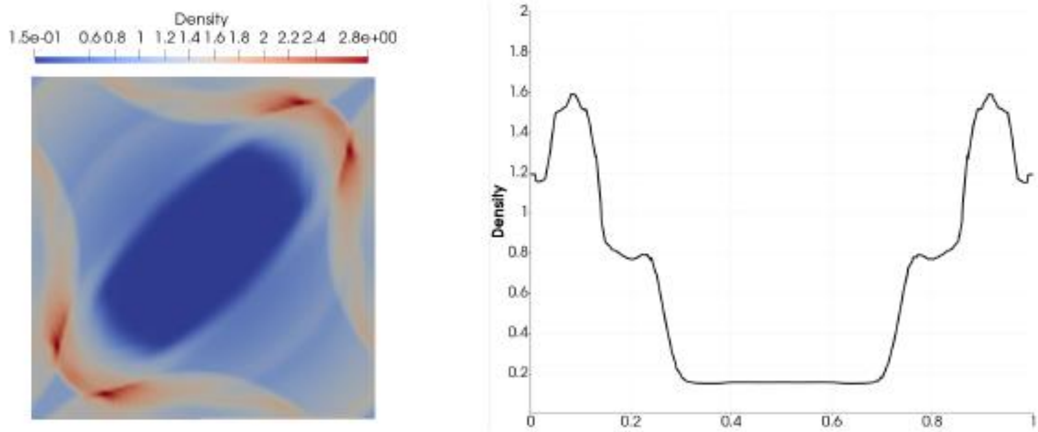


Figure 3.12: Density at time $t = 0.3$, Lax-Friedrichs flux with $\alpha = 1.2$

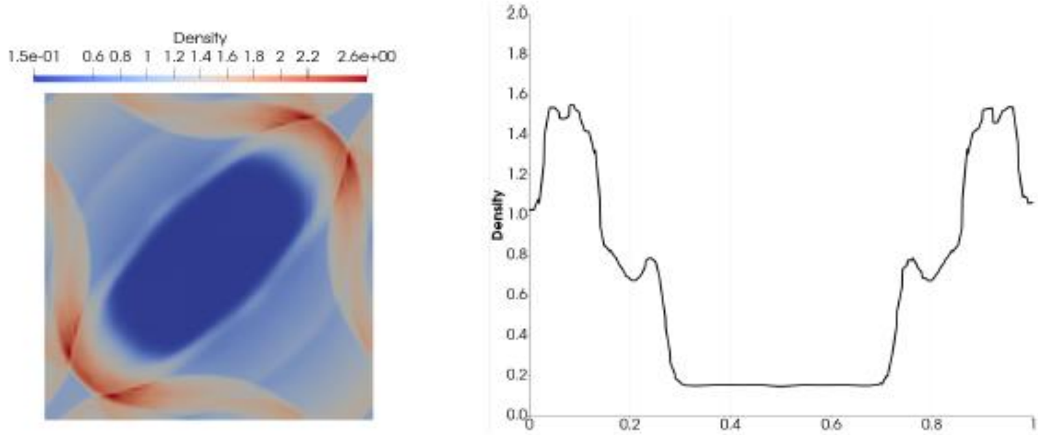


Figure 3.13: Density at time $t = 0.3$, HLLD flux

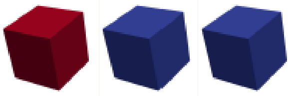
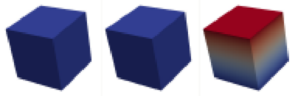
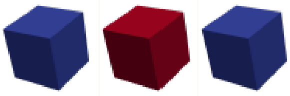
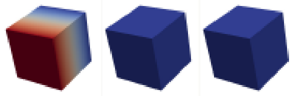
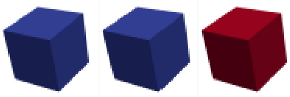
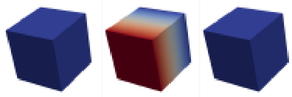
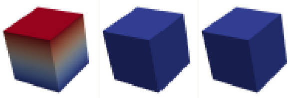
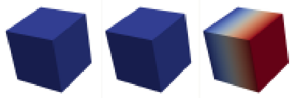
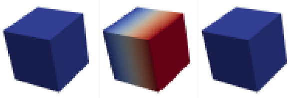
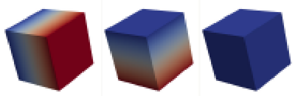
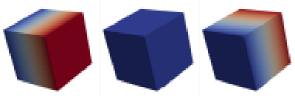
5 - Comparison of numerical fluxes

Divergence-free functional space

Details on this topic are given in Section 3.3, this is an important feature of the implemented solution – and it is the way of ensuring that the Gauss's law,

$$\nabla \cdot B = 0,$$

holds for the magnetic field component of the obtained solution. To ensure that, we are looking for the magnetic field component only in a space of functions that satisfy Gauss's law pointwise. The illustration of such a (vector) functional space is given in the figure below.

$\begin{pmatrix} B_x(x,y,z) \\ B_y(x,y,z) \\ B_z(x,y,z) \end{pmatrix}$	Visualization	$\begin{pmatrix} B_x(x,y,z) \\ B_y(x,y,z) \\ B_z(x,y,z) \end{pmatrix}$	Visualization
$\begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$		$\begin{pmatrix} 0 \\ 0 \\ y \end{pmatrix}$	
$\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$		$\begin{pmatrix} z \\ 0 \\ 0 \end{pmatrix}$	
$\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$		$\begin{pmatrix} 0 \\ z \\ 0 \end{pmatrix}$	
$\begin{pmatrix} y \\ 0 \\ 0 \end{pmatrix}$		$\begin{pmatrix} 0 \\ 0 \\ x \end{pmatrix}$	
$\begin{pmatrix} 0 \\ x \\ 0 \end{pmatrix}$		$\begin{pmatrix} x \\ -y \\ 0 \end{pmatrix}$	
$\begin{pmatrix} x \\ 0 \\ -z \end{pmatrix}$			

6 - Divergence-free basis functions

Time stepping considerations

A solution is being sought in the spatial domain at every time step of the time interval being inspected. The time stepping conditions are of the form

$$\tau_{max} = \min\left\{\frac{\Delta x_{min}}{c_{max}}, \frac{\Delta x_{min}^2}{2\eta_{max}}\right\},$$

where Δx_{min} is the smallest spatial step in the domain (the diameter of the smallest element), and the result is the maximum admissible time-step length. Expressions for other quantities can be found in Section 3.4.2 in the work.

Algebraic formulation

As the overall goal of the work is to obtain distribution of qualities of interest (density, momentum, energy, magnetic field), in the entire time-space domain, a necessary step is the calculation of fully discretized problem using solution of large linear algebraic problems.

The actual algebraic problem is described in Section 3.5, and is of the form

$$Ay = b,$$

where, however, the expressions for A, and even more for b, are quite nontrivial, and involve all the previous considerations to be applied.

Numerical integration

Next in the series of steps towards solving an MHD problem with the discontinuous Galerkin method is actual computation of the elements of A, b from the previous step (algebraic formulation).

To do that, one needs to evaluate integrals of the weak formulation over triangulation elements supporting the basis functions of the standard, as well as the divergence-free finite element spaces. This section describes the process of doing so.

Assembling the algebraic problem

Now, when all is prepared for evaluating contributions to the linear algebraic problem that needs to be solved to obtain spatial distribution of quantities of interest on one time level, it is important to establish an algorithm how to do that effectively. Since the counts of all that needs to be evaluated are very high, care must be taken to perform the calculation as effectively as possible.

A description of the assembling algorithm is in the following figure.

Algorithm 1: Assembling of the algebraic problem Equation (3.53)

\# 1 - Loop over elements

foreach $K^i \in T_h$ **do**

  Data: Quadrature points $\{\boldsymbol{x}_1^i, ..., \boldsymbol{x}_{\boldsymbol{n}}^i\}$

  Data: Jacobian of the mapping J_{K^i} mapping the reference element (unit cube) to the actual element

  Data: Quadrature weights $\{w_1, ..., w_{\boldsymbol{n}}\}$

  \# Loop over quadrature points

  foreach $\boldsymbol{j} \in \{1, ..., \boldsymbol{n}\}$ **do**

    Set: $(JxW)_{\boldsymbol{j}} = J_{K^i} \times w_{\boldsymbol{j}}$

    \# Loop over test functions

    foreach $v \in v_h\left(K^i\right)$ **do**

      Data: l - index of v in the global system, i.e. row in Equation (3.50) - Equation (3.52)

      \# Loop over basis functions

      foreach $u \in v_h\left(K^i\right)$ **do**

        Data: m - index of u in the global system, i.e. column in Equation (3.50)

        $a_{lm} \;+= \; (JxW)_{\boldsymbol{j}} \; a_{lmi\boldsymbol{j}}$

      $b_l \;+= \; (JxW)_{\boldsymbol{j}} \; b_{li\boldsymbol{j}}$

\# 2 - Loop over faces

foreach $\Gamma_{ij} \in T_h$ **do**

  Data: Quadrature points $\left\{\boldsymbol{x}_1^{ij}, ..., \boldsymbol{x}_{\boldsymbol{n}_f}^{ij}\right\}$

  Data: Jacobian of the mapping $J_{K^i} = J_{K^-}$ mapping the reference face (unit square) to the actual face, where K^i, K^j are elements adjacent to Γ_{ij} if this is an internal face, or $K^i = K^j$ if this is a boundary face.

  Data: Quadrature weights $\left\{w_1, ..., w_{\boldsymbol{n}_f}\right\}$

  \# Loop over quadrature points

  foreach $\boldsymbol{j}_f \in \{1, ..., \boldsymbol{n}_f\}$ **do**

    Set: $(JxW)_{\boldsymbol{j}_f} = J_{K^i} \times w_{\boldsymbol{j}_f}$ \# Here it does not matter if we choose J_{K^i} or J_{K^j}

    \# Loop over test functions

    foreach $v \in v_h\left(K^i\right)$ **do**

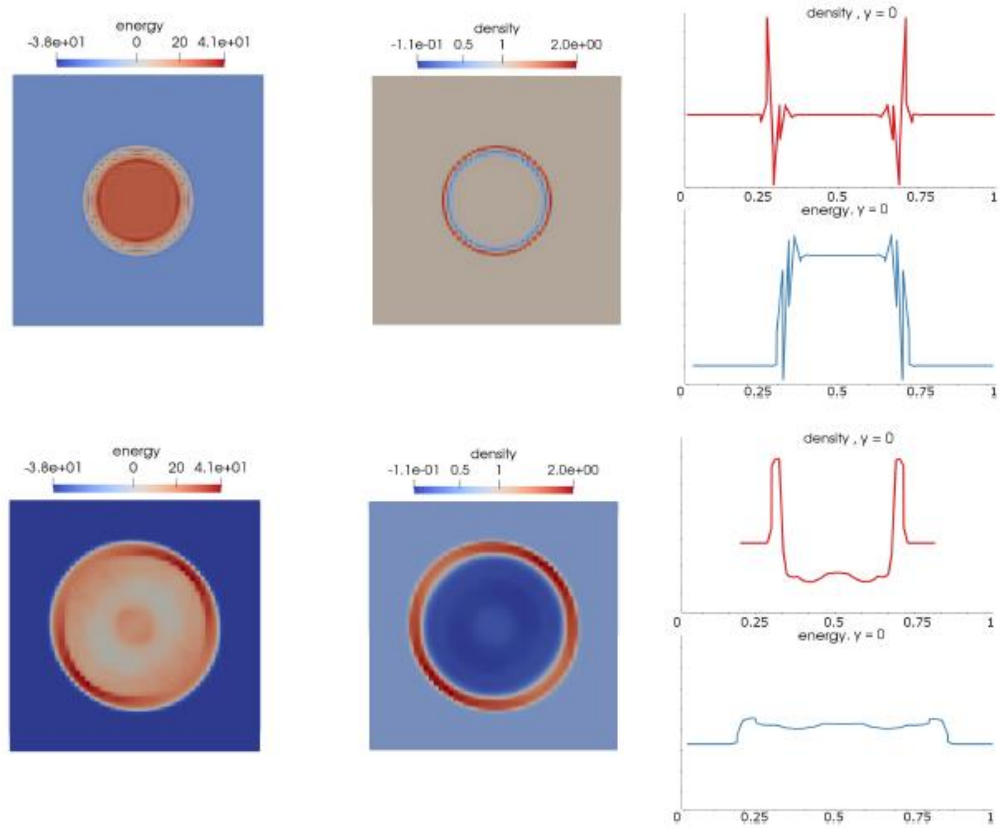
      Data: l - index of v in the global system, i.e. row in Equation (3.50) - Equation (3.52)

      $b_l \;+= \; (JxW)_{\boldsymbol{j}_f} \; b'_{lij\boldsymbol{j}_f}$

7 - Assembling procedure

Slope limiting

After a solution is obtained, it is usually not fully satisfactory, as it may exhibit (under some conditions) nonphysical oscillations caused by the lengthy numerical procedure of obtaining the solution. This is demonstrated in the figure below (note the spurious oscillations in the unlimited solution – namely in the line plots on the right), and the description of handling of this problem is given in Section 3.8 of the work.

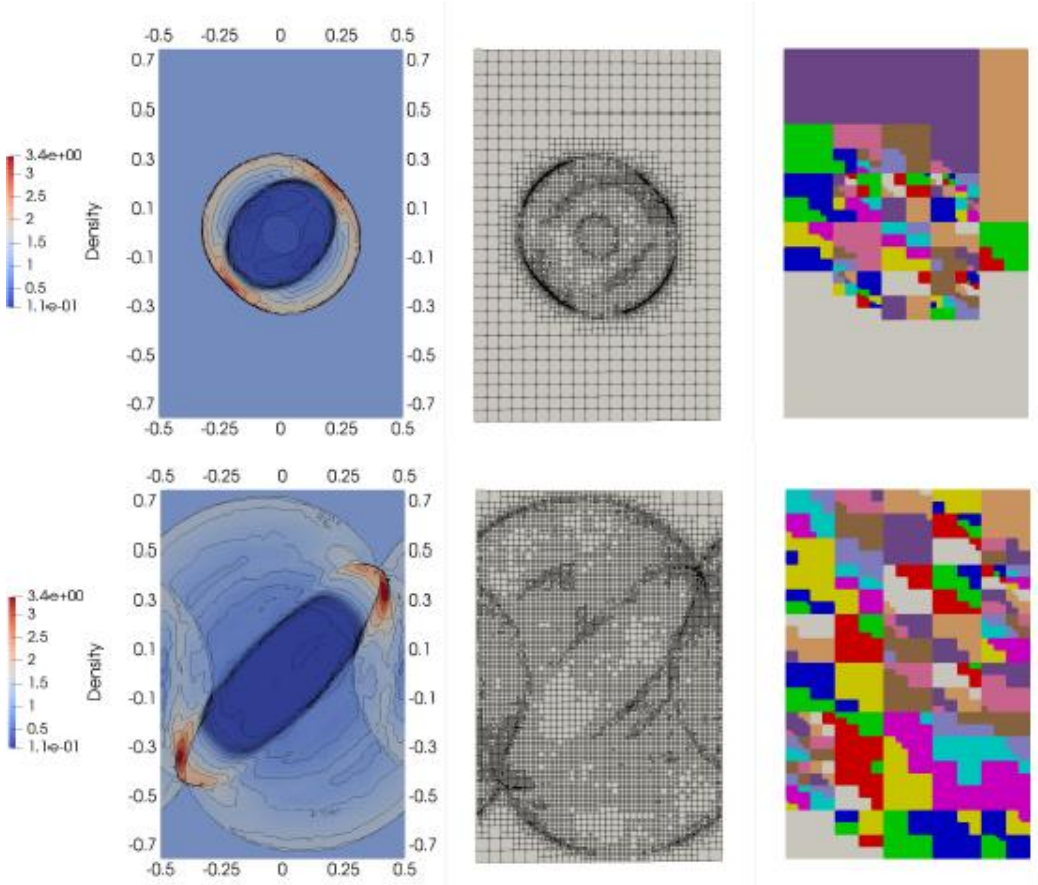


8 - Unlimited solution (top), limited solution (bottom)

4.3 Automatic mesh refinement

As the problems that the implemented software aims at solving can be very large, a mesh adaptive technique needs to be employed to balance the computation cost (which strongly depends on the number of mesh elements) and the quality of the obtained solution. This technique limits the number of elements to such parts of the computational domain, where this would lead to the largest drop of solution error. A thorough description of this technique, with some aspects of the variant used in the work is discussed in Section 4 of the work.

Illustration of the solution (left), the adapted triangulation (middle), and distribution of the elements to processors (right) is in the figure below.



9 - *Illustration of the solution (left), the adapted triangulation (middle), and distribution of the elements to processors (right) for two snapshots of evolving solution (two time intervals)*

The above considerations, including the Adaptive Mesh Refinement (AMR) technique, as applied to the astrophysical problem under study yield results illustrated in the next figure. Many variations of benchmarks, and problems studied are in Section 5 in the work.

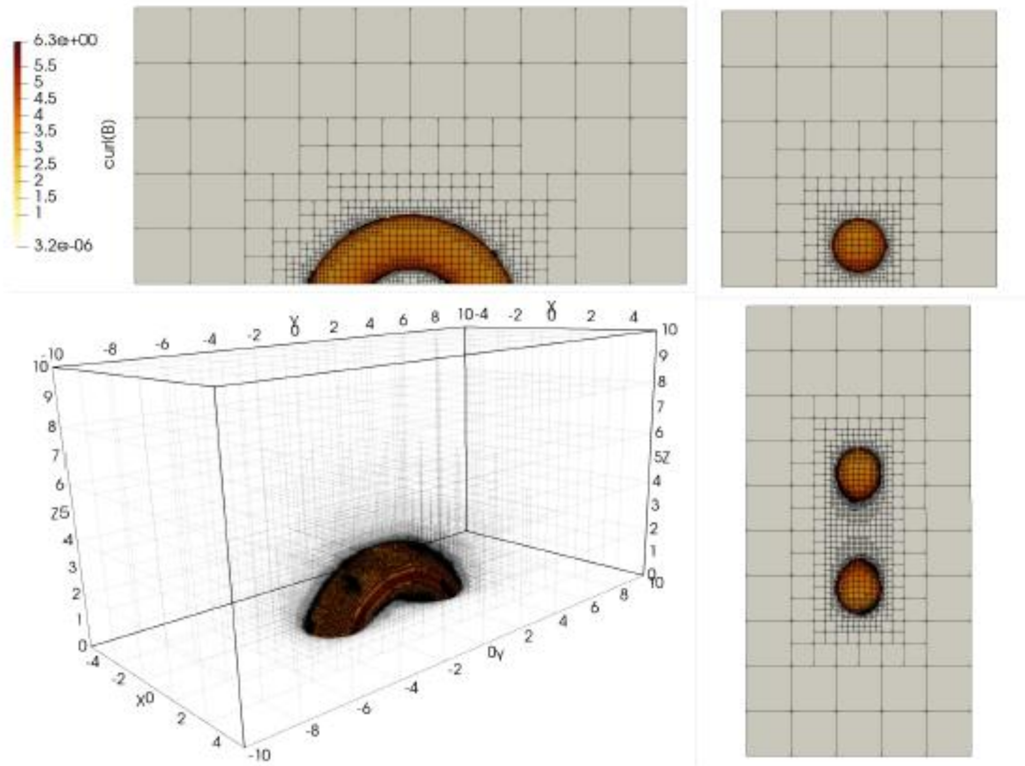


Figure 5.49: AMR step 15, 142147 cells

10 - AMR employed on the Titov-Demoulin problem

5 Conclusion, outlook

As for the mathematical, numerical, and technical (software) parts for such tool to be delivered, all problems that were to be solved, such as

- Shock-capturing for high-order DG scheme to prevent non-physical oscillations (through Vertex-based limiter, see Algorithm 2),
- Adaptive algorithm for the discretization of the space derivatives (through AMR, entire Chapter 4),
- A specific shapeset of basis and test functions based on Taylor expansions (through Section 3.3),
- Adaptive algorithm for the discretization of the time derivative (through CFL condition, see Equation (3.42)),

have been solved, and moreover performance level meets the needs of the use cases. All this has been shown on benchmark problems (see Section 5.1), as well as real- world Titov-Demoulin-based simulation. Of course, much can be improved upon, for example:

- Second-order scheme for the discretization of the time derivative,
- Adaptive algorithm for the discretization of the time derivative,
- Caching of values that are necessary in multiple spaces of the algorithm (utilizing the RAM),
- Further use of vectorization for evaluation of integral quantities,

but the original goal of preparing an easy-to-use, easy-to-extend, and well programmed, and tested software package, has been achieved:

- The code can utilize large-scale clusters through implementation being based on Message Passing Interface (MPI),
- The code is publicly available, and well documented - by following the schematics of the used numerical methods, as well as using clear naming conventions in the object model of the program,
- The code is quite ready for addition of new tests, new benchmarks, new examples, as well as easy parametrization of the existing ones.

5.1 Outlook

Further work will focus on real-world astrophysical problems, where the Titov-Demoulin-based simulations, although all mathematical and numerical apparatus is in place, still need work to satisfy the goal of being reliable and all-purpose tool for astrophysicists. Further work will focus on incorporating additional relevant physical phenomena - mainly study of the magnetic field reconnection - (Bárta et al., 2011), and other phenomena occurring both in solar physics and in industrial applications of plasma ow.

To sum up next steps with the already finished toolset, these are the logical next steps:

- Replicate fully the results of Kotrčč et al. (2012), including all parameters, and boundary condition specifics,
- Run much more detailed simulation over a larger domain for a longer time, to be able to inspect the destructive behavior of the astrophysical event on small scales,
- Continuously improve the performance of the code, x issues as they are discovered, and extend the implemented set of numerical schemes,
- Add additional relevant physics phenomena - resistivity, magnetic field reconnection, possibly relativistic effects,
- Get in touch with other possible users of the implemented software to enrich

the set of possible use cases.

Bibliography

Literature cited in this document and the work

Balsara, D. S. and Spicer, D. S. (1999). A Staggered Mesh Algorithm Using High Order Godunov Fluxes to Ensure Solenoidal Magnetic Fields in Magnetohydrodynamic Simulations. Journal of Computational Physics, 149:270 292.

Bangerth, W., Davydov, D., Heister, T., Heltai, L., Kanschat, G., Kronbichler, M., Maier, M., Turcksin, B., and Wells, D. (2015). The deal. II library, version 8.4. preprint.

Barth, T. and Jespersen, D. (1989). The design and application of upwind schemes on unstructured meshes. In 27th Aerospace Sciences Meeting, number AIAA-89- 0366. American Institute of Aeronautics and Astronautics.

Batten, P., Clarke, N., Lambert, C., and Causon, D. (1997). On the choice of wave speeds for the hllc riemann solver. SIAM Journal on Scientific Computing, 18(6):1553 1570.

Bárta, M., Büchner, J., Karlický, M., and Skála, J. (2011). Spontaneous current-layer fragmentation and cascading reconnection in solar flares. i. model and analysis. The Astrophysical Journal, 737(1):24.

Burstedde, C., Wilcox, L. C., and Ghattas, O. (2011). p4est: Scalable algorithms for parallel adaptive mesh refinement on forests of octrees. SIAM Journal on Scientific Computing, 33(3):1103 1133.

Courant, R., Friedrichs, K., and Lewy, H. (1967). On the partial difference equations of mathematical physics. IBM J. Res. Dev., 11(2):215 234.

Davis, T. (2006). Direct Methods for Sparse Linear Systems. Society for Industrial and Applied Mathematics.

Denner, F., Evrard, F., Serfaty, R., and van Wachem, B. G. (2017). Artificial viscosity model to mitigate numerical artefacts at fluid interfaces with surface tension. Computers & Fluids, 143:59 72.

Derigs, D., Winters, A., Gassner, G., Walch, S., and Bohm, M. (2017). Ideal GLM-MHD: About the entropy consistent nine-wave magnetic field divergence diminishing ideal magnetohydrodynamics equations. 364.

Dolej²í, V. and Feistauer, M. (2015). Discontinuous Galerkin Method: Analysis and Applications to Compressible Flow. Springer Series in Computational Mathematics. Springer International Publishing.

G., R. H. (1997). Thomas, j. w.: Numerical partial differential equations. Finite difference methods. ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik, 77(5):386 386.

Gaburov, E. and Nitadori, K. (2011). Astrophysical weighted particle magnetohydrodynamics. Monthly Notices of the Royal Astronomical Society, 414(1):129 154.

Hariri, S., Park, J. B., Yu, F. K., Parashar, M., and Fox, G. C. (1993). A message passing interface for parallel and distributed computing. In [1993] Proceedings The 2nd International Symposium on High Performance Distributed Computing, pages 84 91.

Heroux, M. A., Bartlett, R. A., Howle, V. E., Hoekstra, R. J., Hu, J. J., Kolda, T. G., Lehoucq, R. B., Long, K. R., Pawlowski, R. P., Phipps, E. T., Salinger, A. G., Thornquist, H. K., Tuminaro, R. S., Willenbring, J. M., Williams, A., and Stanley, K. S. (2005). An overview of the Trilinos project. ACM Trans. Math. Softw., 31(3):397 423.

Intel Corporation (2017). Intel parallel studio.

J., R. W. and B., L. R. (2002). The use of classical Lax-Friedrichs riemann solvers with discontinuous Galerkin methods. International Journal for Numerical Methods in Fluids, 40(334):479 486.

Kestener, P. et al. (1992). Ramses-gpu.

Kitware Inc. (2017). Paraview 5.5.

Korous, L. (2012). Adaptive hp discontinuous Galerkin method for nonstationary compressible Euler equations.

Korous, L. and Solin, P. (2012). An adaptive meshes for non-stationary compressible Euler equations. Computing, 95(1):425 444.

Kotrč, P., Bárta, M., Schwartz, P., Kupryakov, Y., Kashapova, L., and Karlický, M. (2012). Modeling of ha eruptive events observed at the solar limb. 284.

Kuzmin, D. (2010). A vertex-based hierarchical slope limiter for p-adaptive discontinuous Galerkin methods. Journal of Computational and Applied Mathematics, 233(12):3077 3085. Finite Element Methods in Engineering and Science (FEMTEC 2009).

Londrillo, P. and Zanna, L. D. (2000). High-order upwind schemes for multidimensional magnetohydrodynamics. The Astrophysical Journal, 530(1):508.

Ma, Z., Korous, L., and Santiago, E. (2012). Solving a suite of {NIST} benchmark problems for adaptive {FEM} with the Hermes library. Journal of Computational and Applied Mathematics, 236(18):4846 4861. {FEMTEC} 2011: 3rd International Conference on Computational Methods in Engineering and Science, May 9 13, 2011.

Miyoshi, T. and Kusano, K. (2005). A multi-state HLL approximate Riemann solver for ideal magnetohydrodynamics. Journal of Computational Physics, 208:315 344.

Mocz, P., Vogelsberger, M., Sijacki, D., Pakmor, R., and Hernquist, L. (2014). A discontinuous Galerkin method for solving the fluid and magnetohydrodynamic equations in astrophysical simulations. mnras, 437:397 414.

Norman, M., Wilson, J., et al. (1992). Zeus.

Orszag, S. A. and Tang, C.-M. (1979). Small-scale structure of two-dimensional magnetohydrodynamic turbulence. Journal of Fluid Mechanics, 90:129 143.

Rossmanith, J. A. (2013). High-Order Discontinuous Galerkin Finite Element Methods with Globally Divergence-Free Constrained Transport for Ideal MHD. ArXiv e-prints.

Skala, J. and Barta, M. (2012). LSFEM implementation of MHD numerical solver. ArXiv e-prints.

Skála, J., Baru a, F., Büchner, J., and Rampp, M. (2015). The 3d MHD code goemhd3 for astrophysical plasmas with large Reynolds numbers. A&A, 580:A48.

Solin, P., Korous, L., and Kus, P. (2014). Hermes2d, a C++ library for rapid development of adaptive -fem and -dg solvers. Journal of Computational and Applied Mathematics, 270:152 165. Fourth International Conference on Finite Element Methods in Engineering and Sciences (FEMTEC 2013).

Stone, J., Gardiner, T., Teuben, P., et al. (2008a). Athena.

Titov, V. and Demoulin, P. (1999). Basic topology of twisted magnetic configurations in solar flares. 351:707 720.

Xisto, C. M., Páscoa, J. C., and Oliveira, P. J. (2014). A pressure-based high-resolution numerical method for resistive MHD. Journal of Computational Physics, 275:323 345.

Zachary, A., Malagoli, A., and Colella, P. (1994). A higher-order Godunov method for multidimensional ideal magnetohydrodynamics. SIAM Journal on Scientific Computing, 15(2):263 284.

Own publications

Journals with Impact Factor

[1] Vadym Aizinger, Dmitri Kuzmin, Lukas Korous, Scale separation in fast hierarchical solvers for discontinuous Galerkin methods, September 2015, Applied Mathematics and Computation 266:838-849, DOI: 10.1016/j.amc.2015.05.047

[2] Lukas Korous, Pavel Solin, An adaptive hp-DG method with dynamically- changing meshes for non-stationary compressible Euler equations, May 2012, COMPUTING, 95,1,425-444, DOI: 10.1007/s00607-012-0257-1

[3] Pavel Solin, Lukas Korous, Pavel Kus, Hermes2D, a C++ library for rapid development of adaptive hp-FEM and hp-DG solvers, November 2014, Journal of Computational and Applied Mathematics 270:152-165, DOI: 10.1016/j.cam.2014.02.007

[4] Pavel Solin, Ondrej Certik, Lukas Korous, Three anisotropic benchmark problems for adaptive finite element methods, March 2013, Applied Mathematics and Computation 219(13), DOI: 10.1016/j.amc.2010.12.080

[5] Zhonghua Ma, Lukas Korous, Erick Santiago, Solving a suite of NIST bench- mark problems for adaptive FEM with the Hermes library, December 2012, Journal of Computational and Applied Mathematics 236(18):4846-4861, DOI: 10.1016/j.cam.2012.02.004

[6] Pavel Solin, Lukas Korous, Space-time adaptive hp-FEM for problems with traveling sharp fronts, May 2012, Computing 95(1), DOI: 10.1007/s00607- 012-0243-7

[7] Pavel Solin, Lukas Korous, Adaptive higher-order finite element methods for transient PDE problems based on embedded higher-order implicit Runge-Kutta methods, February 2012, Journal of Computational Physics 231(4):1635- 1649, DOI: 10.1016/j.jcp.2011.10.023

Conference papers

[1] Lukas Korous, Pavel Karban, Distributed Implicit Discontinuous Galerkin MHD Solver, COMPUMAG 2017, Daejeon, Korea

[2] Ivo Dolezel, Lukas Korous, Pavel Karban, Frantisek Mach, Higher-Order Eggshell Method for Computation of Forces Acting on Ferromagnetic Bodies, January 2014, Conference: 9th IET International Conference on Computation in Electromagnetics (CEM 2014), DOI: 10.1049/cp.2014.0188

[3] Korous Lukas, Adaptive hp-DG Method for Nonstationary Compressible Euler Equations, Elektrotechnika a informatika 2012. Part 1., Elektrotechnika,53-56

Software

[1] Lukas Korous, Pavel Solin, Milan Hanus, Jakub Cerveny, The library for physical fields modelling using hp-adaptive finite element method Hermes2D

[2] Lukas Korous, Pavel Solin, Pavel Kus, David Andrs, Library for physical fields modelling in three-dimensional space using hp-adaptive FEM Hermes3D