Helices in MHD Flow: Numerical Results from Penalized Pseudospectral Simulations

Malcolm Roberts[1], Kai Schneider, Wouter Bos, Jorge Morales, Matthieu Leroy

Aix-Marseille University

Séminaire Equations aux Dérivées Partielles, Strasbourg, 2014-03-04

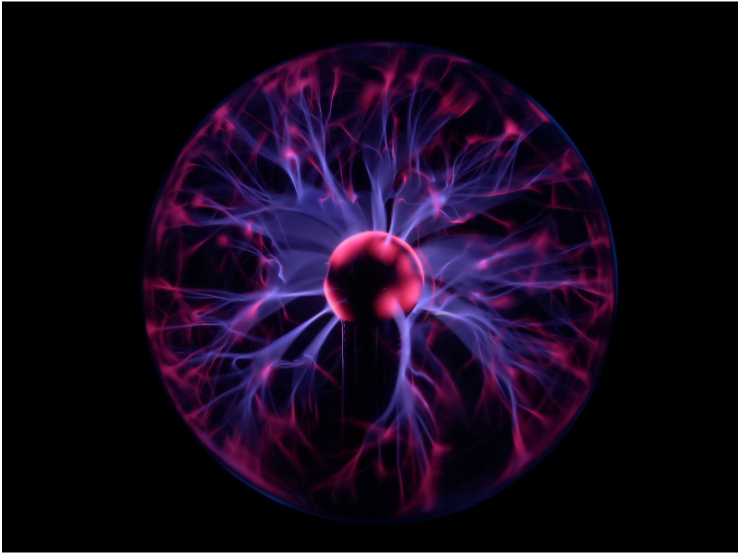
[1]malcolm.i.w.roberts@gmail.com, www.malcolmiwroberts.com

Helices in MHD Flow: Numerical Results from Penalized Pseudospectral Simulations

The self-organization of magnetohydrodynamic (MHD) flows is an important behaviour which can lead to a better understanding of the underlying dynamics, and is important for applications in engineering and the natural sciences. MHD flows can be extremely complex and difficult to simulate, which is further complicated by the fact that many flows of physical interest are bounded in what may be very complicated domains.

In this talk, I will present some recent results simulations using the pseudospectral method with boundaries implemented via penalization. The flow geometry is bounded in a periodic cylinder with no-slip conditions for the velocity and with the magnetic field forced by imposing a helical flow at the boundary. I will show how seemingly minor changes in the cross-sectional geometry and wrapping number for the helical forcing drastically changes the flow self-organization.

Simulation of Self-Organization in MHD Flow



Outline

- Presentation of model
- Numerical Method
  - Pseudospectral Method
  - Penalty method
  - Method for determining the penalty field
- Simulations with circular cross-section
- Simulations with elliptical cross-section
- Current Work
  - Implicitly Dealiased Convolutions
  - Neumann Boundary Conditions
- Conclusions

Governing Equations: MHD

Let $\boldsymbol{u}$ be the velocity of an electric field with magnetic field $\boldsymbol{B}$.
The velocity changes as

$$\frac{\partial \boldsymbol{u}}{\partial t} = \boldsymbol{u} \times \boldsymbol{\omega} + \boldsymbol{j} \times \boldsymbol{B} - \boldsymbol{\nabla} P + \nu \nabla^2 \boldsymbol{u}$$

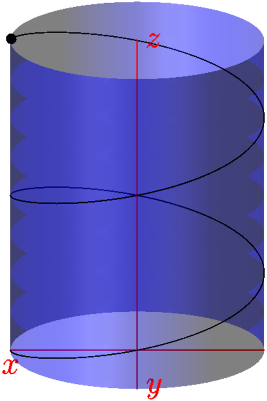
where $\boldsymbol{\omega} = \nabla \times \boldsymbol{u}$ is the vorticity, $\boldsymbol{j} = \nabla \times \boldsymbol{B}$ is the current density, P the pressure, and ν is the kinematic viscosity.
The magnetic field changes as

$$\frac{\partial \boldsymbol{B}}{\partial t} = \boldsymbol{\nabla} \times (\boldsymbol{u} \times \boldsymbol{B}) + \lambda \nabla^2 \boldsymbol{B}$$

where λ is the magnetic diffusivity.
We require the velocity and magnetic field be solenoidal:

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 \qquad \boldsymbol{\nabla} \cdot \boldsymbol{B}{=}0$$

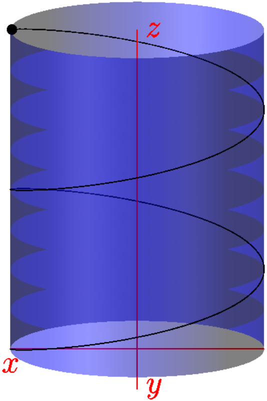
Boundary Conditions and Geometry



- The fluid is evolved in a periodic cylinder denoted Ω_{f}.
- The velocity is no-slip:
  - $\boldsymbol{u}|_{\partial\Omega_{\mathrm{f}}} = \mathbf{0}$
- The magnetic field is forced towards a helix:
  - $B_{\perp}|_{\partial\Omega_{\mathrm{f}}} = 0$
  - $B_z|_{\partial\Omega_{\mathrm{f}}} = B_0$
  - $B_\theta|_{\partial\Omega_{\mathrm{f}}} = B_c$

The wrapping number of the forcing (the inverse safety factor) is set to integer values.

Boundary Conditions and Geometry



We can also impose an elliptical cross-section, shown here with eccentricity $1/\sqrt{2}$.

Using a level-set approach (for example), very general geometries may be described.

As in the circular case, wrapping numbers are integral.

Initial Conditions and Physical Parameters

Physical parameters:

- $\nu = 4.5 \times 10^{-2}$
- $\lambda = 4.5 \times 10^{-2}$
- Prandtl number is unity.

Geometrical parameters:

- Major radius is set to 1.
- The length of the cylinder in the z-direction is 8.

Initial conditions:

- The magnetic field matches the boundary conditions.
- The velocity field is perturbed with a random field.
- The perturbation has kinetic energy of order 10^{-6}.

Numerical Method

The source terms are computed via the pseudospectral method.

Boundary conditions are imposed via the penalty method.

The system is advanced in time using an Adams-Bashforth method, with Laplacian terms treated implicitly.

Pseudospectral Method

Let $\hat{\boldsymbol{u}}_{\boldsymbol{k}}$ and $\hat{\boldsymbol{B}}_{\boldsymbol{k}}$ be the Fourier transform of $\boldsymbol{u}$ and $\boldsymbol{B}$.
The Fourier transform of the governing equations are

$$\frac{\partial \hat{\boldsymbol{u}}_{\boldsymbol{k}}}{\partial t} = \mathcal{F}\left(\boldsymbol{u} \times \boldsymbol{\omega}\right) + \mathcal{F}\left(\boldsymbol{j} \times \boldsymbol{B}\right) - i\boldsymbol{k}\hat{P}_{\boldsymbol{k}} - \nu k^2 \hat{\boldsymbol{u}}_{\boldsymbol{k}},$$

with the pressure determined via $\nabla \cdot \boldsymbol{u} = 0 \iff i\boldsymbol{k} \cdot \hat{\boldsymbol{u}}_{\boldsymbol{k}} = 0$,
and

$$\frac{\partial \hat{\boldsymbol{B}}_{\boldsymbol{k}}}{\partial t} = i\boldsymbol{k} \times \mathcal{F}\left(\boldsymbol{u} \times \boldsymbol{B}\right) - \lambda k^2 \hat{\boldsymbol{B}}_{\boldsymbol{k}}$$

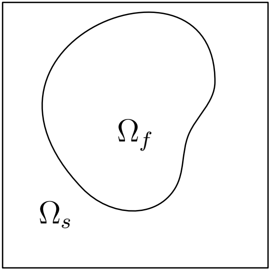
The nonlinear terms are computed by:

- 2/3-padding the input data
- transforming from Fourier space to physical space
- multiplying the fields
- transforming back into Fourier space.

Pseudospectral Method

The use of FFTs make the pseudospectral method efficient.

FFTs can only be used when the computational domain Ω is a periodic box.



We embed the fluid domain Ω_f inside Ω.

The solid domain is $\Omega_\mathsf{s} = \Omega/\Omega_\mathsf{f}$.

We *penalize* the motion of the fluid in the solid domain with penalization parameter η.

Penalty Method

Let χ_{Ω_s} be the characteristic function for Ω_s.
The penalized velocity evolution equations is

$$\frac{\partial \boldsymbol{u}}{\partial t} = \boldsymbol{u} \times \boldsymbol{\omega} + \boldsymbol{j} \times \boldsymbol{B} - \boldsymbol{\nabla} P + \nu \nabla^2 \boldsymbol{u} - \frac{\chi_{\Omega_s}}{\eta} \boldsymbol{u},$$

corresponding to homogeneous Dirichlet boundary conditions.
The penalized evolution equation for $\boldsymbol{B}$ is

$$\frac{\partial \boldsymbol{B}}{\partial t} = \boldsymbol{\nabla} \times (\boldsymbol{u} \times \boldsymbol{B}) + \lambda \nabla^2 \boldsymbol{B} - \frac{\chi_{\Omega_s}}{\eta} (\boldsymbol{B} - \boldsymbol{B}_s)$$

where $\boldsymbol{B}_s$ is the penalization field.
Source terms are projected onto the solenoidal manifold via a Helmholtz decomposition.

Penalty Method

Advantages:

- Proof of convergence, $\mathcal{O}\left(\sqrt{\eta}\right)$.
- Deals with complex geometries.
- Easy to implement.

Disadvantages:

- Only first-order accurate in space.
- Stiff in time: $dt \approx \eta$.
- Theory mostly developed for Dirichlet boundary conditions.

Current Directions:

- Improving convergence and reducing stiffness.
- Generalizing boundary conditions.

Determining the Penalty Field

The penalty field $\boldsymbol{B}_s$ should

- match the boundary conditions at $\partial\Omega_{\mathsf{f}}$,
- be solenoidal,
- and be as regular as possible.

Determining the Penalty Field

For circular geometries, we can make use of the fact that, in cylindrical coordinates,

$$\hat{\boldsymbol{r}} \cdot \boldsymbol{B}_s = 0,$$

so

$$\boldsymbol{B}_s = B_c f(r)\hat{\boldsymbol{\theta}} + B_0\hat{\boldsymbol{z}}, \tag{1}$$

with $f(r)$ a smooth function that is equal to 1 at the boundary and goes to zero within the periodic box.

The formulation given in equation (1) is necessarily solenoidal.

Similarly, any $\boldsymbol{B}_s$ corresponding to solid-body motion is guaranteed to be both smooth and solenoidal.

Determining the Penalty Field

We can also find such fields in general.
Suppose that we are given boundary conditions $\boldsymbol{v}_{\mathrm{bc}}$ on $\partial\Omega_{\mathsf{f}}$ for the field $\boldsymbol{v}$.
Suppose also that

$$\int_{\partial\Omega_{\mathsf{f}}} \boldsymbol{v}_{\mathrm{bc}} \cdot \hat{\boldsymbol{n}}\, ds = 0,$$

so that the boundary conditions are consistent with a solenoidal field $\boldsymbol{v}$.
We find the penalization field $\boldsymbol{v}_{\mathsf{s}}$ in the computational domain Ω by solving

$$\kappa \nabla^2 \boldsymbol{v}_{\mathsf{s}} - \frac{\chi_{\partial\Omega_{\mathsf{f}}}}{\eta_\tau} \left(\boldsymbol{v}_{\mathsf{s}} - \boldsymbol{v}_{\mathrm{bc}}\right) = 0. \tag{2}$$

Determining the Penalty Field

We solve equation (2) using by pseudo-time-stepping and the pseudospectral method.

The field is made solenoidal by performing a Helmholtz decomposition on $\boldsymbol{v}_\mathsf{s}$ after each pseudo-time-step.

Pseudo-time-stepping is stopped when

$$\|\boldsymbol{v}_\mathsf{s} - \boldsymbol{v}_{\mathrm{bc}}\|_{\infty,\partial\Omega_\mathsf{f}} < 0.2 \times \sqrt{\eta},$$

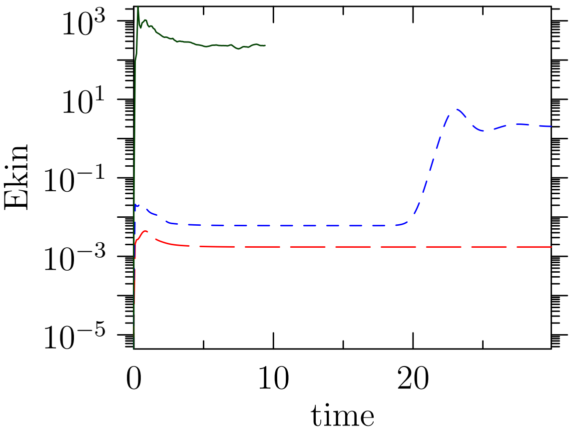
which implies that the error in the boundary conditions is less than the expected error from the penalty method.

Simulations: Circular Cross-Section

Simulations were performed on `ada.idris.fr` and `turing.idris.fr`.

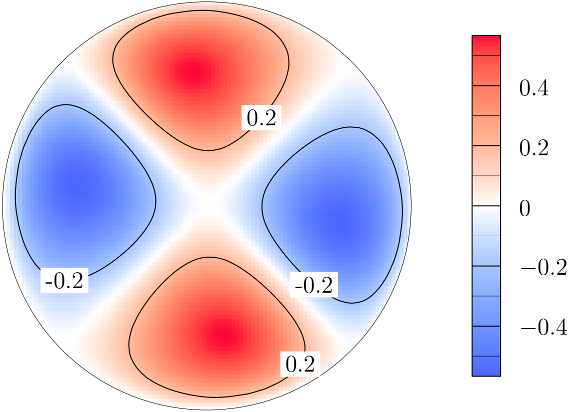
- For low forcing amplitudes, the axial velocity was negligible.
- Simulations with $\|(B_0, B_c)\| \gtrapprox 15$ showed exponential growth of the axial kinetic energy.
- The axial kinetic energy eventually reached a stable plateau.
- Increasing wrapping number decreased the axial kinetic energy growth rate.
- The velocity field self-organized into helical pairs.

Simulations: Circular Cross-Section



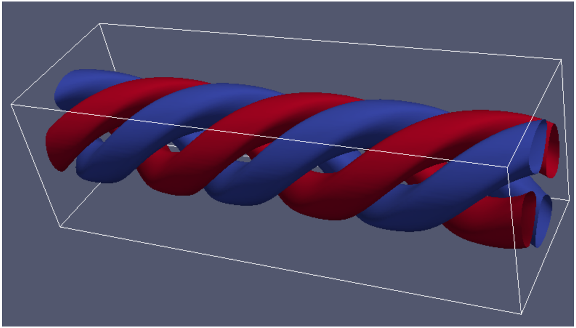
Kinetic energy as a function of time for different forcing parameters.

Simulations: Circular Cross-Section



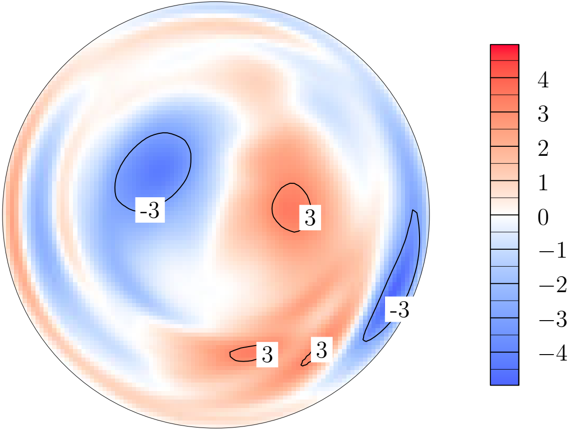
Axial velocity for $B_c = 7.06$, $B_0 = 4.5$, wrapping number 2.

Simulations: Circular Cross-Section



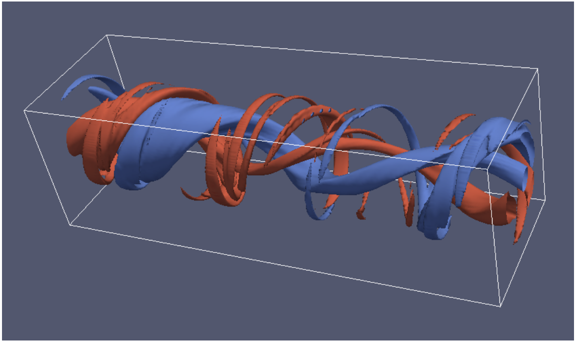
Axial velocity for $B_c = 7.06$, $B_0 = 4.5$, wrapping number 2.

Simulations: Circular Cross-Section



Axial velocity for $B_c = 70.6$, $B_0 = 4.5$, wrapping number 20.

Simulations: Circular Cross-Section



Axial velocity for $B_c = 70.6$, $B_0 = 4.5$, wrapping number 20.

Simulations: Circular Cross-Section

Simulations with circular cross sections exhibited:

- Growth of axial kinetic energy for large forcing amplitude.
- Growth was positively correlated with forcing wrapping number.
- The flow self-organized into a variety of helical modes.
- Large enough energy growth produced a transition to turbulence.
- Turbulent flows were composed of a high-mode boundary layer with a low-order helical mode away from the boundary.

Circular geometries produce helical modes.

By removing symmetries, what happens to the helical modes?

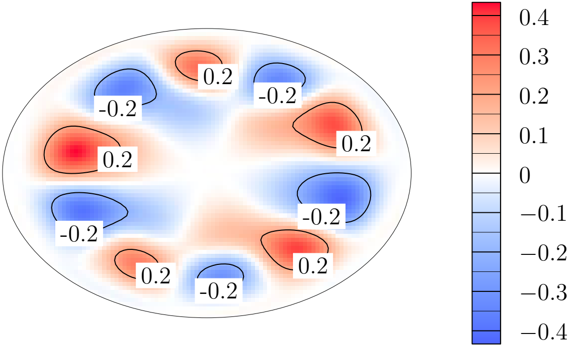
Simulations: Elliptical Cross-Section

Increasing eccentricity suppressed growth of axial kinetic energy.

The first instance of self-organization accrued at $\|(B_0, B_c)\| = 60$ for our simulations.

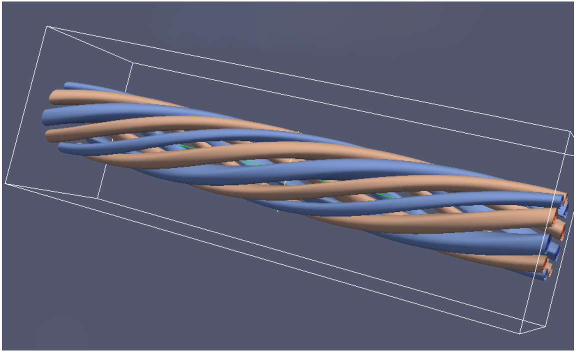
The mode azimuthal mode-number was much larger than in the circular case.

Simulations: Elliptical Cross-Section



Axial velocity for $B_c = 49.7$, $B_0 = 33.6$, wrapping number 1.

Simulations: Elliptical Cross-Section



Axial velocity for $B_c = 49.7$, $B_0 = 33.6$, wrapping number 1.

Simulations: Elliptical Cross-Section

The elliptical geometry

- Suppressed axial kinetic energy growth.
- Also exhibited self-organization into helical modes.
- The resulting helical structures had a larger azimuthal modenumber.
- Axial velocity tended to be concentrated farther away from the z-axis than in the circular case.

Summary: MHD Simulations

For sufficiently strong forcing, the simulations showed:

- The self-organization of the velocity into helical structures
- Low-mode helical structures persisted even when the flow was turbulent.
- Increasing the eccentricity stabilized the flow.

Current Work

- Implicitly Dealiased Convolutions
  - Uses less memory.
  - Faster.

- More general boundary conditions.
  - Homogeneous Neumann.
  - Non-penetration.

Implicitly Dealiased Convolutions

Evaluation of the nonlinear source term is the most expensive part of the pseudospectral method.

It is made more expensive by having to extend the input data from length N to $\frac{3}{2}N$, in each dimension.

For 3D simulations, FFTs are of size

$$\left(\frac{3}{2}N\right)^3 = \frac{27}{8}N^3 = 3.375\,N^3.$$

Implicitly Dealiased Convolutions

One can instead implicitly zero-pad the FFT.

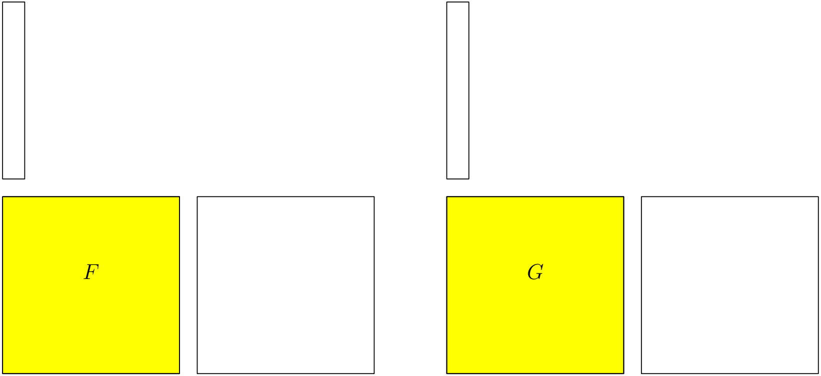
The output is partly in-place, and partly in a discontiguous work array of size $N/2$.

Implicitly zero-padded FFTs allow one to re-use work memory.

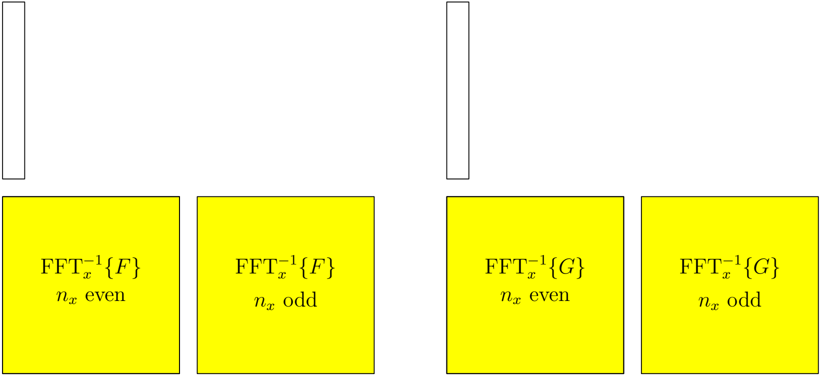
Joint work with John C. Bowman, University of Alberta.

Implicitly Dealiased Convolutions

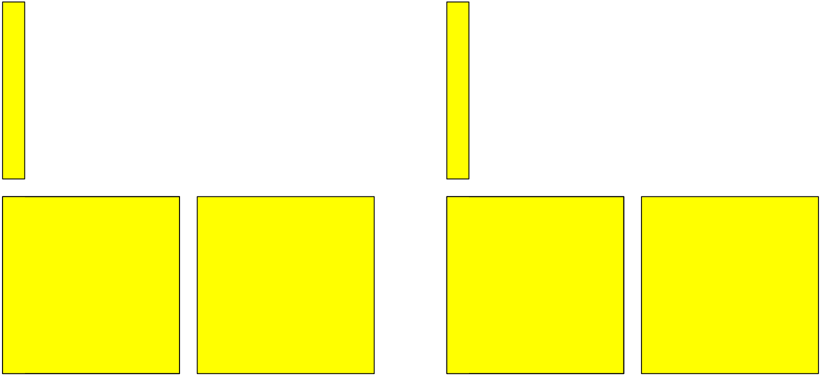
Implicitly Dealiased Convolutions



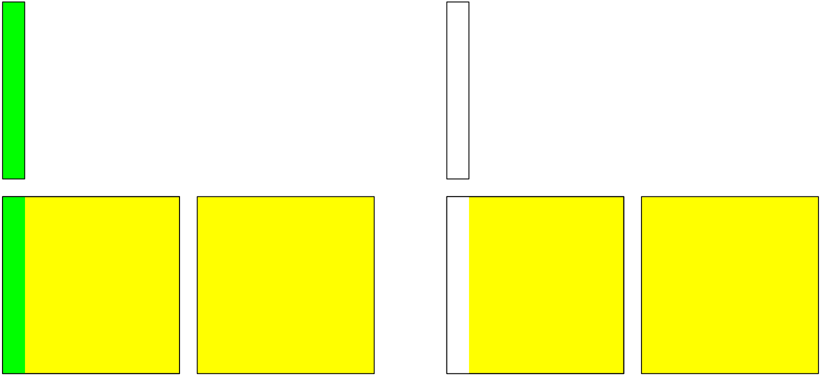
Implicitly Dealiased Convolutions



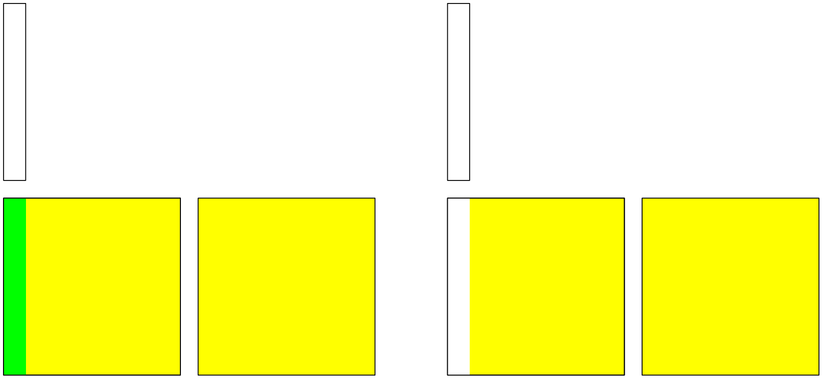
Implicitly Dealiased Convolutions



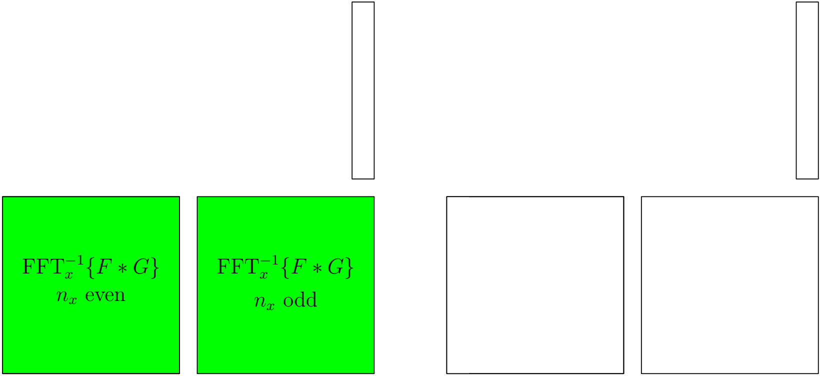
Implicitly Dealiased Convolutions



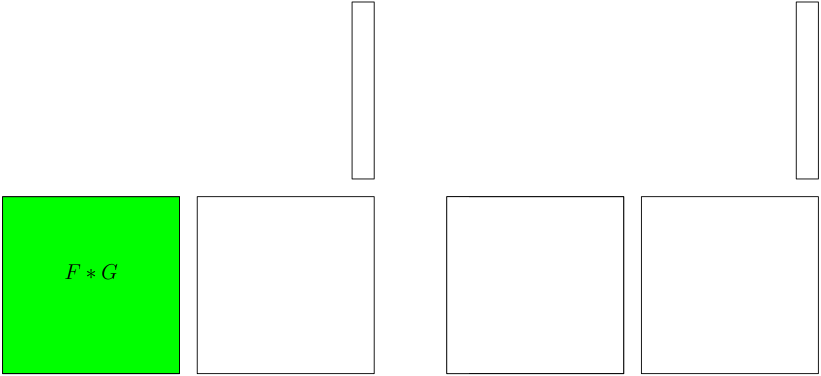
Implicitly Dealiased Convolutions



Implicitly Dealiased Convolutions



Implicitly Dealiased Convolutions



Implicitly Dealiased Convolutions

Benefits:

- Decreased memory use: $\frac{3}{2}N^3$ per input instead of $\frac{27}{8}N^3$.
- Increased speed: approximately twice as fast.
- Computational complexity reduced by skipping transforms of zeroed input.
- Fully multi-threaded using `OpenMP`.
- Available at `fftwpp.sf.net`, LGPL.

Current Work:

- Use function-pointers to allow arbitrary binary operations.
- Implementation of MPI routines.
- Modification of data format to work with real-complex FFTs.

General Boundary Conditions via Penalization

We specify the fluid domain via a level-set function:

$$\phi : \mathbb{R}^n \rightarrow \mathbb{R}$$

with

$$\Omega_\mathsf{f} = \{\boldsymbol{x} : \phi(\boldsymbol{x}) < 0\}$$

and

$$\Omega_\mathsf{s} = \{\boldsymbol{x} : \phi(\boldsymbol{x}) > 0\}.$$

The level-set function ϕ is a distance function from the boundary.

General Boundary Conditions via Penalization

The penalty method imposed boundary conditions modifying a PDE

$$\frac{\partial \boldsymbol{v}}{\partial t} = \boldsymbol{S}$$

and adding a penalty term:

$$\frac{\partial \boldsymbol{v}}{\partial t} = \boldsymbol{S} - \frac{\chi_{\Omega_\mathrm{f}}}{\eta}\left(\boldsymbol{v} - \boldsymbol{v}_\mathrm{s}\right)$$

where $\boldsymbol{v}_\mathrm{s}$ is the penalty field.
This can be modified to penalize only the normal component to zero by evolving

$$\frac{\partial \boldsymbol{v}}{\partial t} = \boldsymbol{S} - \hat{\boldsymbol{n}}\frac{\chi_{\Omega_\mathrm{f}}}{\eta}\left(\hat{\boldsymbol{n}} \cdot \boldsymbol{v}\right).$$

General Boundary Conditions via Penalization

The boundary $\partial\Omega_{\mathrm{f}}$ is defined by

$$\partial\Omega_{\mathrm{f}} = \{\boldsymbol{x} : \phi(\boldsymbol{x}) = 0\}.$$

The normal to the surface $\partial\Omega_{\mathrm{f}}$ is easily computed:

$$\hat{\boldsymbol{n}} = \frac{\boldsymbol{\nabla}\phi}{\|\boldsymbol{\nabla}\phi\|}.$$

Thus, desirable properties of ϕ include:

- Monotonic with respect to signed distance from the boundary.
- Continuously differentiable.
- Gradient should be well-behaved.
- ϕ should give us a useful distance from the boundary.

General Boundary Conditions via Penalization

Consider the level-set function

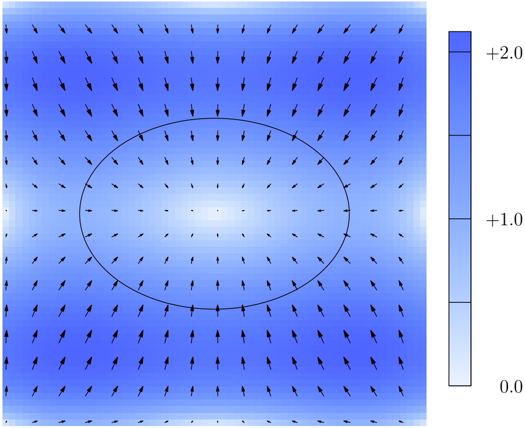
$$\phi(x,y) = x^2 + 2y^2 - 2^2,$$

which defines an elliptical region.

The derivative is easy to define in $\mathbb{R}^2$.

In a periodic box, things are not so straight-forward.

General Boundary Conditions via Penalization



Direction and magnitude of $\nabla\phi$.

General Boundary Conditions via Penalization

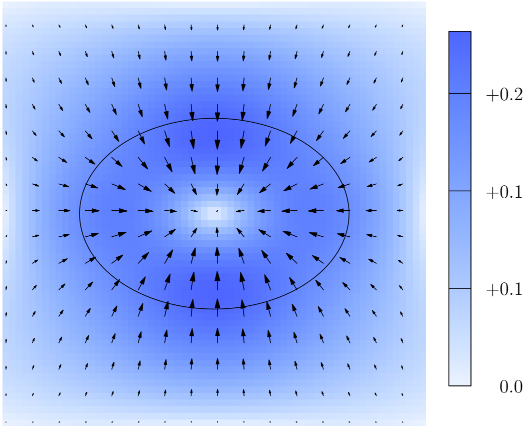
Let $A = |\min \phi|$, and consider

$$f = \frac{\phi}{\phi + 2A}$$

Then,

- $\phi = 0 \iff f = 0$
- $f : \mathbb{R}^n \to [-1, 1]$
- $\|\nabla f\|$ is largest near the boundary.
- $f = 1$ corresponds to $|\boldsymbol{x}| \to \infty$.

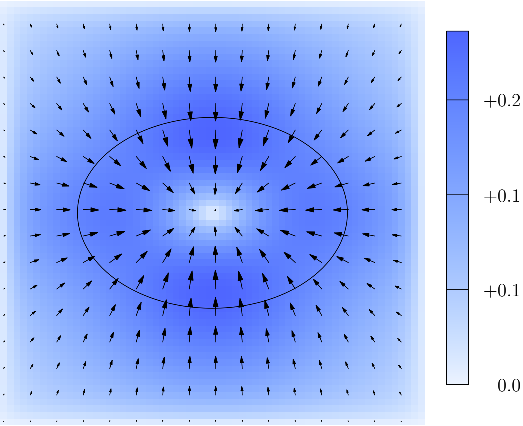
General Boundary Conditions via Penalization



Direction and magnitude of ∇f.

General Boundary Conditions via Penalization

We can also stretch the spatial coordinates in the wall domain:



Direction and magnitude of ∇f, stretched.

Neumann Boundary Conditions via Penalization

By considering $\boldsymbol{v}$ component-by-component, we can also impose Neumann boundary conditions:
Consider the evolution equation

$$\frac{\partial \theta}{\partial t} = S_\theta$$

Let $\boldsymbol{v} = \boldsymbol{\nabla}\theta$. The evolution equation for $\boldsymbol{v}$ is

$$\frac{\partial \boldsymbol{v}}{\partial t} = \boldsymbol{\nabla} S_\theta$$

We can penalize the normal component of $\boldsymbol{v}$:

$$\frac{\partial \boldsymbol{v}}{\partial t} = \boldsymbol{\nabla} S_\theta - \hat{\boldsymbol{n}} \frac{\chi_{\Omega_\mathrm{f}}}{\eta} \left(\hat{\boldsymbol{n}} \cdot \boldsymbol{V's} \right)$$

Neumann Boundary Conditions via Penalization

Imposing the normal component of $\boldsymbol{v}$ corresponds to homogeneous Neumann boundary conditions on θ:

$$\frac{\partial \theta}{\partial t} = S_\theta - \boldsymbol{\nabla}^{-1} \hat{\boldsymbol{n}} \frac{\chi_{\Omega_f}}{\eta} \left(\hat{\boldsymbol{n}} \cdot \boldsymbol{v} \right)$$

For pseudospectral simulations, we can invert the gradient operator with relative ease:

$$\frac{\partial \mathcal{F}\left[\theta\right]\left(\boldsymbol{k}\right)}{\partial t} = \mathcal{F}\left[S_\theta\right]\left(\boldsymbol{k}\right) - \frac{i\boldsymbol{k}}{\left(ik\right)^2} \cdot \hat{\boldsymbol{n}} \mathcal{F}\left[\frac{\chi_{\Omega_f}}{\eta}\left(\hat{\boldsymbol{n}} \cdot \boldsymbol{v}\right)\right].$$

Non-penetration of $\boldsymbol{u}$ and $\boldsymbol{\omega}$ (or $\boldsymbol{B}$ and $\boldsymbol{j}$) can be implemented via a combination of homogeneous Dirichlet and Neumann boundary conditions.

Conclusions

The goal of this work is the simulation of complex MHD flows. The fluid was confined to a periodic cylinder and the magnetic field helically forced at the boundary.

- The velocity self-organized into helices for sufficiently strong forcing amplitude.
- These helical modes survived even in turbulent regimes.
- Changing the cross-section of the cylinder dramatically changed the flow structure.

Current work:

- Implicitly dealiased convolutions.
- Non-penetration boundary conditions.

Merci pour votre attention!