Higher Order Mapped Spectral Elements for 2D MHD

Bernhard Hientzsch
Courant Institute of Mathematical Sciences
New York University
mailto:Bernhard.Hientzsch@na-net.ornl.gov
http://www.math.nyu.edu/~hientzsc

April 25, 2006

CEMM Meeting Spring 2006
Dallas, Texas

Working with H.R. Strauss at NYU.

Current work and plans, I

- *Mapped elements* (straightline, isoparametric). Semi-implicit MHD. Versions running: C version, MATLAB version. C modules wrapped in different ways: flat vector (callable from FORTRAN), original data structures (callable as PYTHON extensions), initialization and main program (to call FORTRAN). C program reads mesh from file and possibly restart data. Writes tables and regular check points. Scripts (now mostly PYTHON) to visualize and analyze. Can now write different semi-implicit-type time-advances in PYTHON and FORTRAN with modules. Starting to integrate with simplified version of M3D.

- *Rectangular mesh of elements*: Semi-implicit MHD. Some more runs. Some variation on the equations used (include gravitation). Implementing more boundary conditions and problems (periodic boundary conditions, Kelvin-Helmholtz problem).

Current work and plans, II

- C^1*-continuous elements*: At first, only Laplace and Helmholtz solves for C^1-solution with C^0 right hand sides and results transformed back to C^0 degrees of freedom. Right hand sides etc. from old code. Maybe later, C^1 elements exclusively and 4th order formulation Jardin.M3D-C1.

- *Higher order time-stepping*: Higher order splitting methods Karniadakis, Israeli, Orzag; JCP '91, requires only Helmholtz and Laplace (as before).

- *Fully implicit time-stepping*: Crank-Nicholson, Backward Euler. Newton method. Explicit formulae for Jacobian. Explicit assembly including static condensation. Fast application. Direct solution. For iterative solution, need inexact solver (like semi-implicit scheme for Newton system) Chacon. Close to proof-of-concept.

- *Larger PDE systems*: Integration with M3D. Stand-alone Hall MHD code under design.

Outline

- Equations: 2D incompressible resistive MHD.
- Formulation: Vorticity-Flux
- Problem: Tilting mode
- Fixed mesh, increasing degrees.
- Fixed degree, different meshes (mesh packing). Comparisons.

Incompressible resistive MHD: primitive variables

$$\nabla u := \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right)$$

$$\nabla^2 u := \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$$

$\mathbf{B}$: magnetic field.

$\mathbf{v}$: velocity field.

ρ: density, here equals one. μ: viscosity. η: resistivity.

$$\frac{\partial \mathbf{B}}{\partial t} = \mathbf{curl}(\mathbf{v} \times \mathbf{B}) + \eta \nabla^2 \mathbf{B} \tag{1}$$

$$\rho \frac{\partial \mathbf{v}}{\partial t} = -\rho \mathbf{v} \cdot \nabla \mathbf{v} + \mathbf{curl}\,\mathbf{B} \times \mathbf{B} + \rho\mu \nabla^2 \mathbf{v} \tag{2}$$

$$\nabla \cdot \mathbf{v} = 0 \tag{3}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{4}$$

Incompressible MHD: potential form in 2D (vorticity-flux)

$$[a, b] := \frac{\partial a}{\partial x}\frac{\partial b}{\partial y} - \frac{\partial b}{\partial x}\frac{\partial a}{\partial y}.$$

$\mathbf{v} = \mathbf{curl}\,\phi$ with velocity flux ϕ. $\mathbf{B} = \mathbf{curl}\,\psi$ with magnetic flux ψ.

Ω: vorticity. $C = \nabla^2\psi$: current density.

$$\frac{\partial \Omega}{\partial t} = [\nabla^2\psi, \psi] - [\Omega, \phi] + \mu\nabla^2\Omega \quad (5)$$

$$\frac{\partial \psi}{\partial t} = -[\psi, \phi] + \eta\nabla^2\psi \quad (6)$$

$$\nabla^2\phi = \Omega \quad (7)$$

Tilting mode problem - Setup

Introduce polar coordinates $x = r\cos\theta$, $y = r\sin\theta$ and use separable form in polar coordinates,

$$\psi(t=0) = \psi_{0,rad}(r)\cos\theta \qquad \Omega(t=0) = \epsilon\Omega_{0,pert}(r)$$

with the radial functions (k being the first positive zero of J_1, $k \approx 3.8317$)

$$\psi_{0,rad}(r) = \begin{cases} \frac{2J_1(kr)}{kJ_0(k)} & \text{for } r \leq 1 \\ \frac{r^2-1}{r} & \text{for } r \geq 1 \end{cases} \qquad \Omega_{0,pert}(r) = 4(r^2-1)\exp(-r^2)$$

The following boundary conditions were used in this problem:

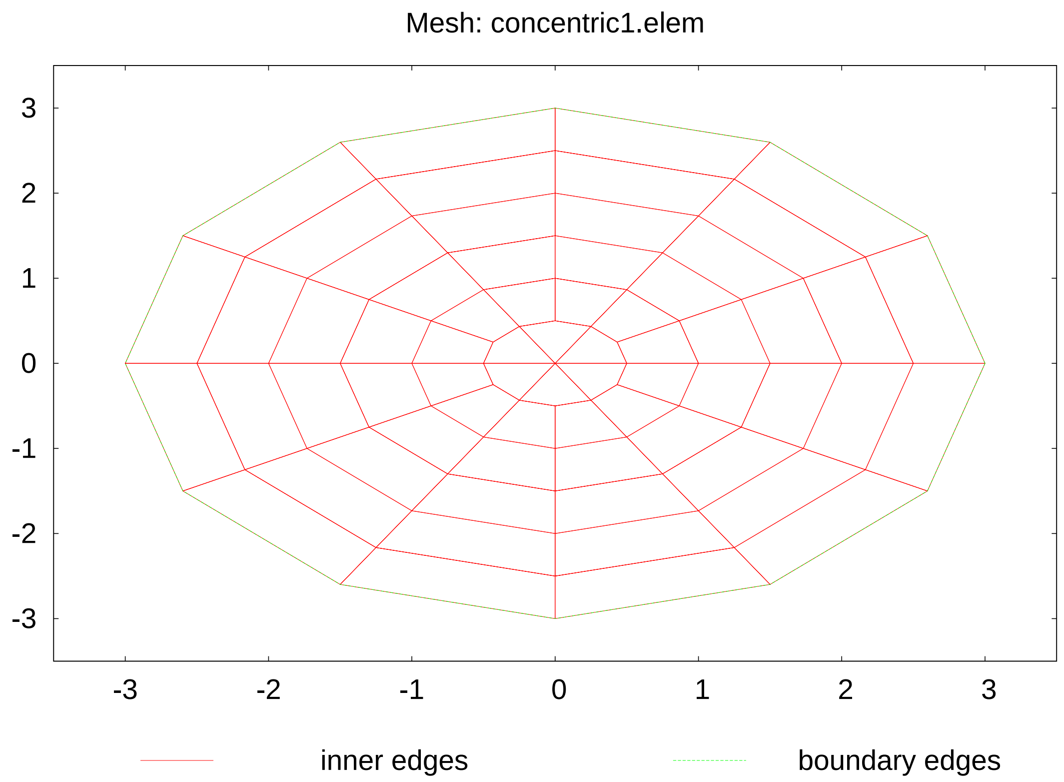
$$\phi = 0 \qquad C = 0 \qquad \frac{\partial\psi}{\partial t} = 0 \qquad \Omega = 0$$

Algorithm

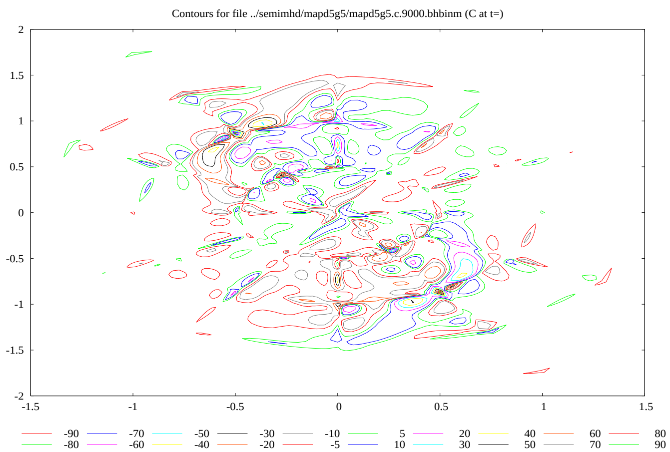
- Semi-implicit, leap-frog type time-stepping.
- Only Helmholtz, Laplace equations to be solved. Mass matrix, stiffness matrix, and Poisson bracket must be implemented.
- Direct solver. Cholesky, taking advantage of symmetry. Static condensation.
- Possible parallelization: statically condense Schur complement system on each processor to Super-Schur complement system on processor boundaries. (Or, assign one element to each processor.)
- For the move to iterative solvers for larger problems: Implement in PETSc or other library that allows parallel iterative methods. Need to implement preconditioners for Helmholtz and Laplace (DD).

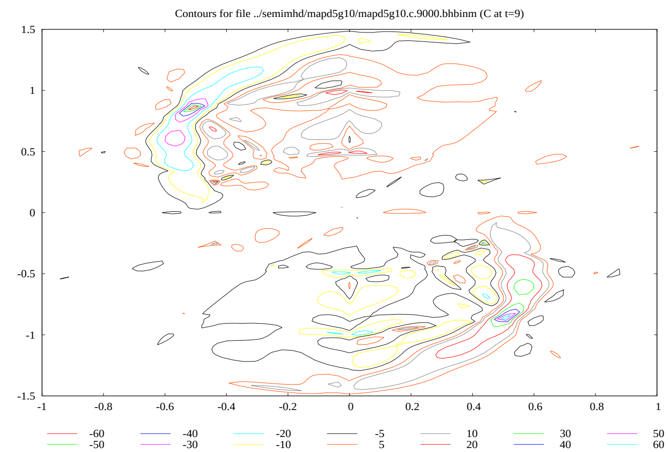
Standard example: Concentric Circles

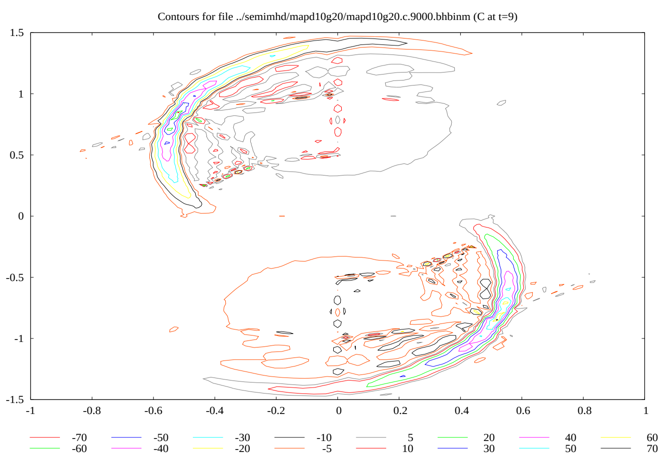
$$\Delta t = 0.001,\ \epsilon = 0.001,\ \mu = 0.005,\ \eta = 0.001.$$

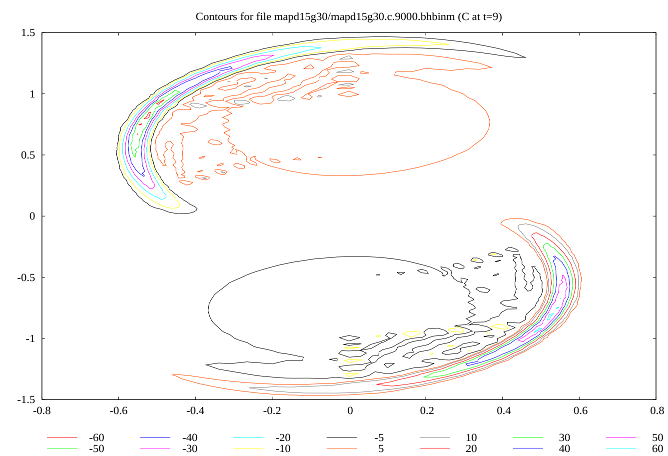


Standard example: increasing degrees, $t = 9.0$

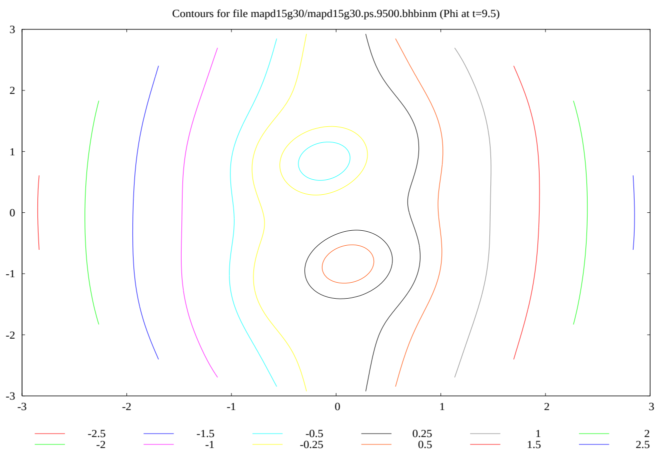


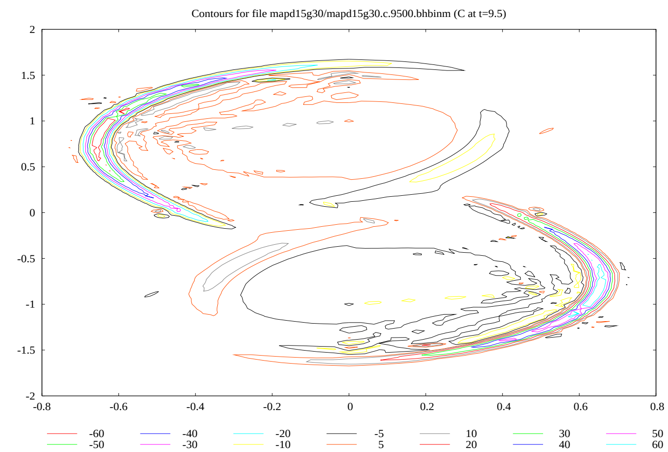


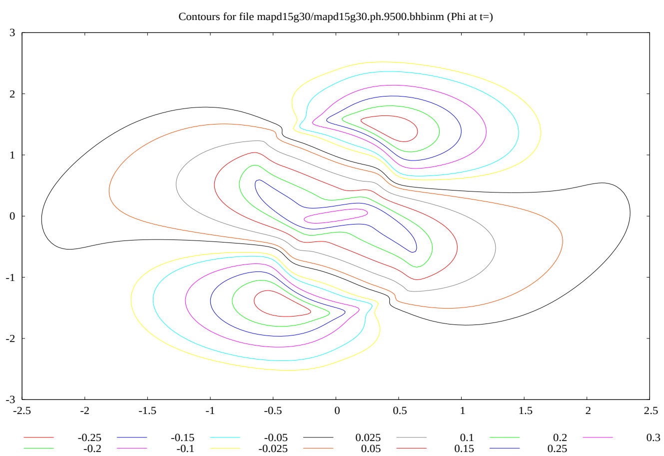


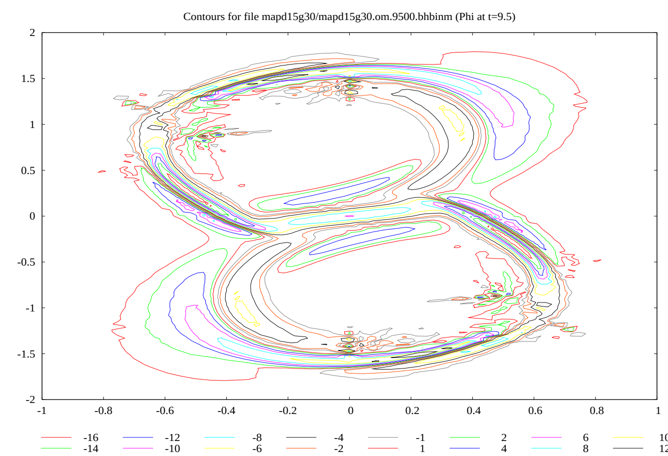


Element degree 15, integration degree 30, $t = 9.5$

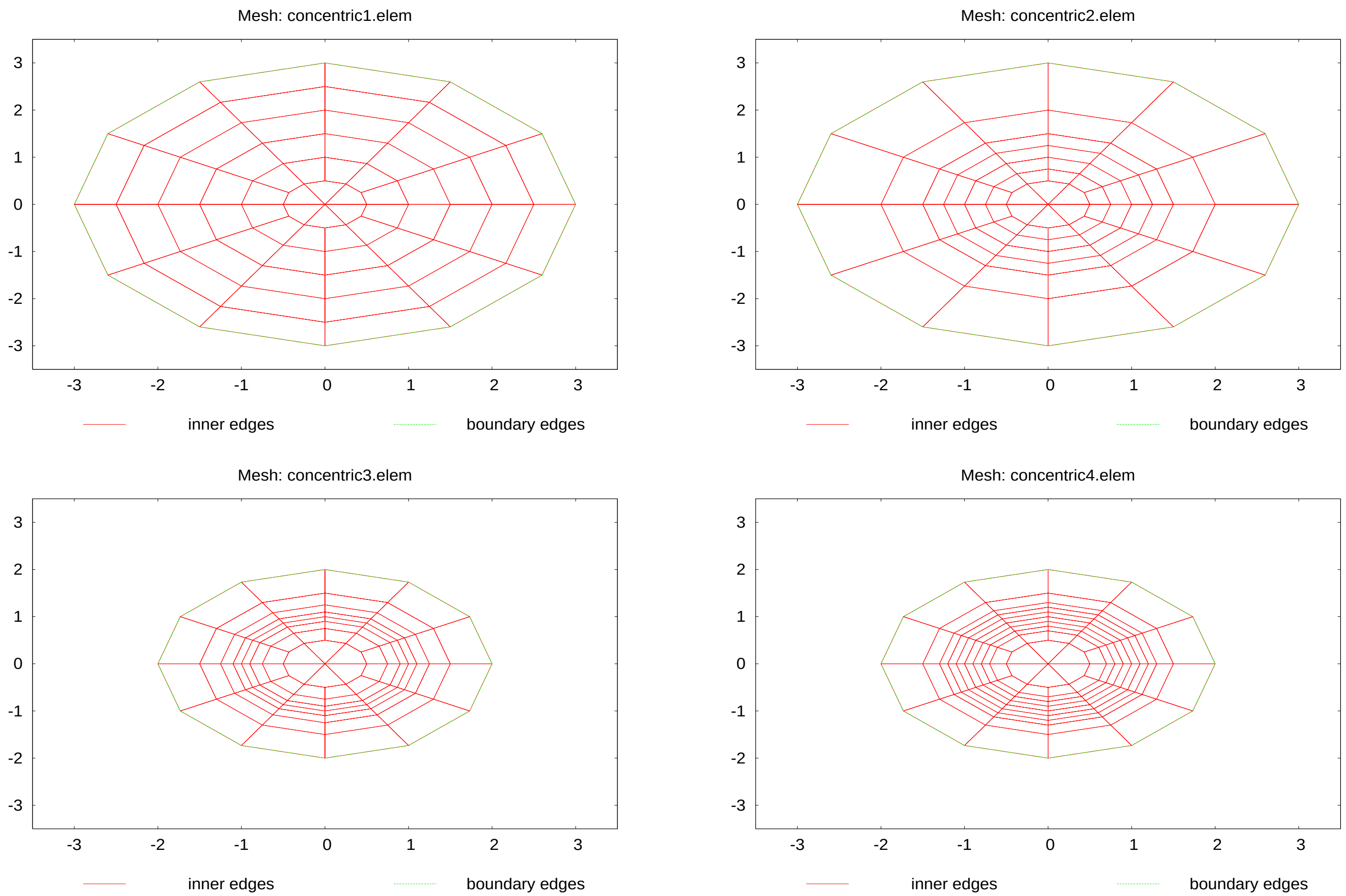




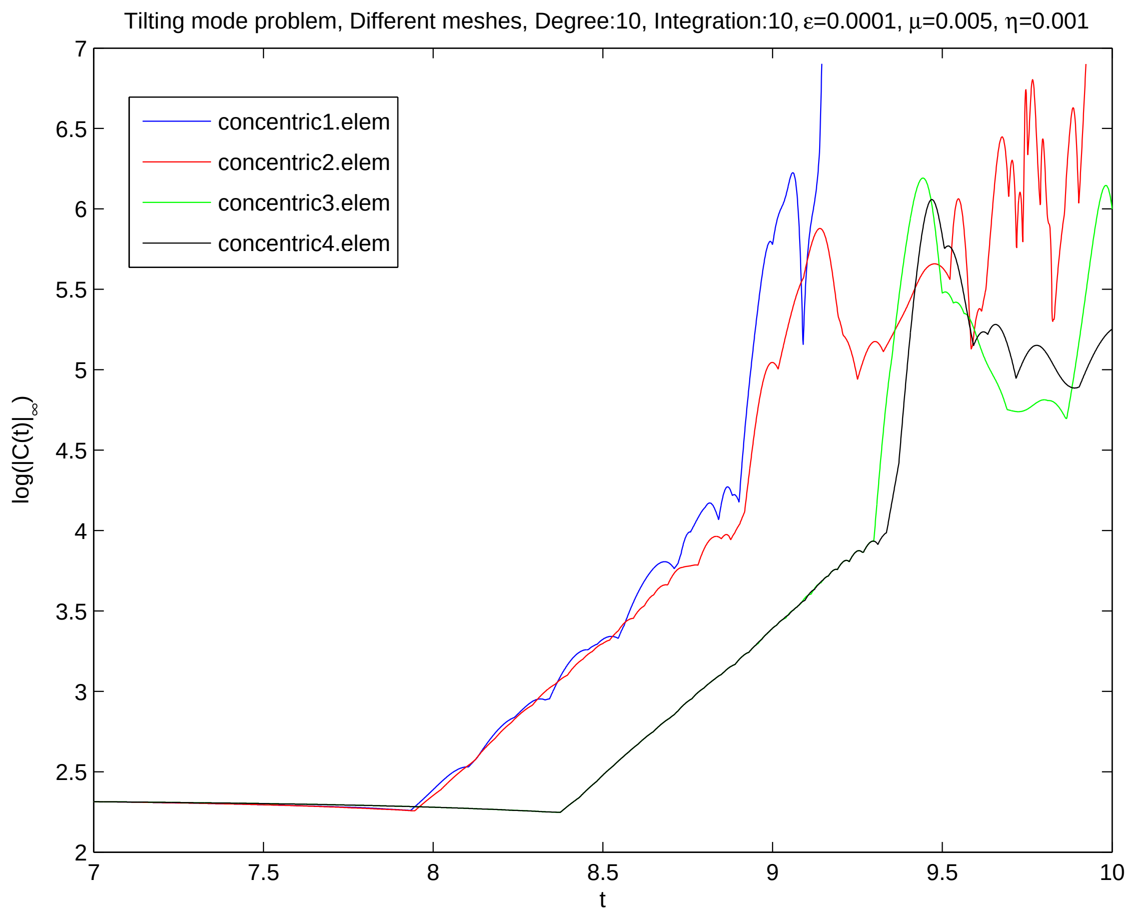




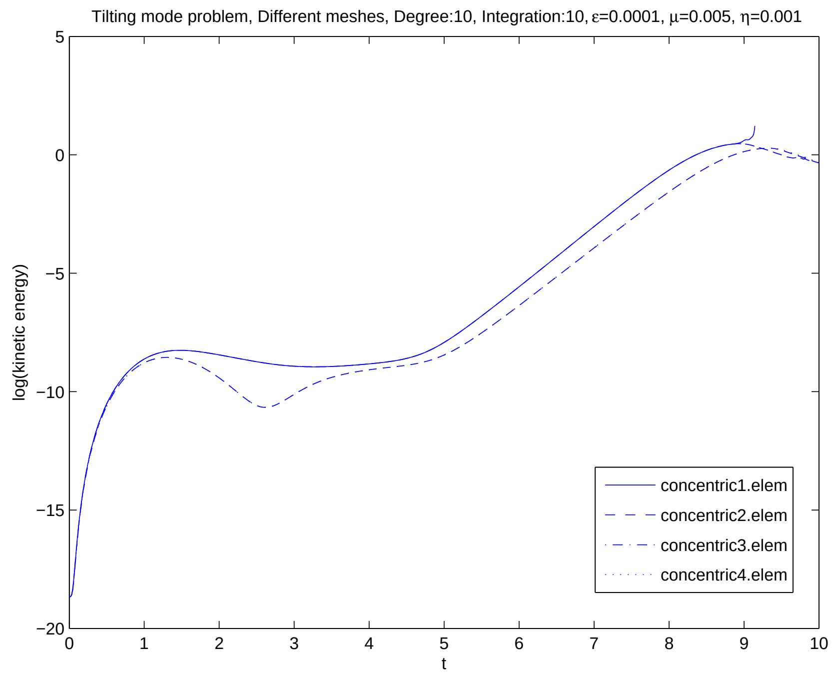
Concentric circle meshes – grid packing

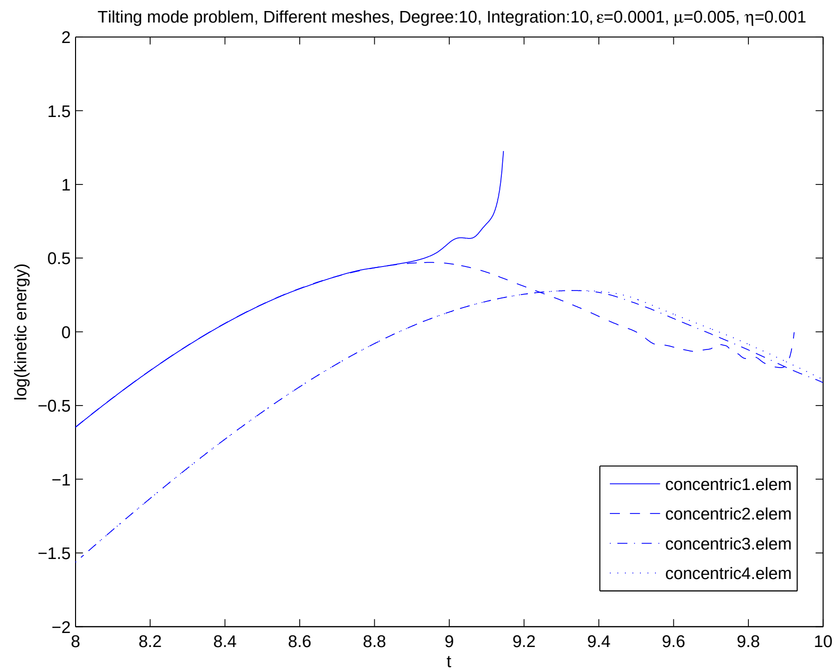


Comparing growth of maximum current.

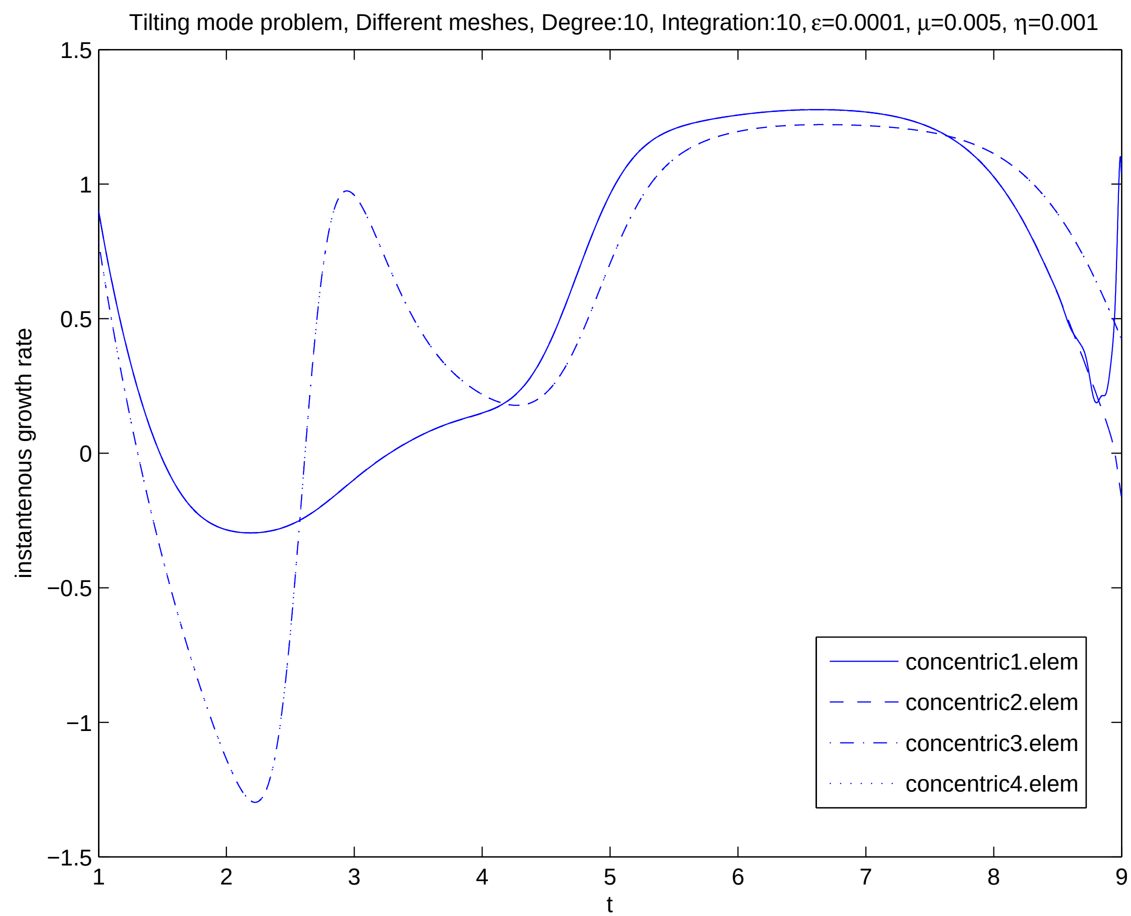


Comparing growth of kinetic energy.

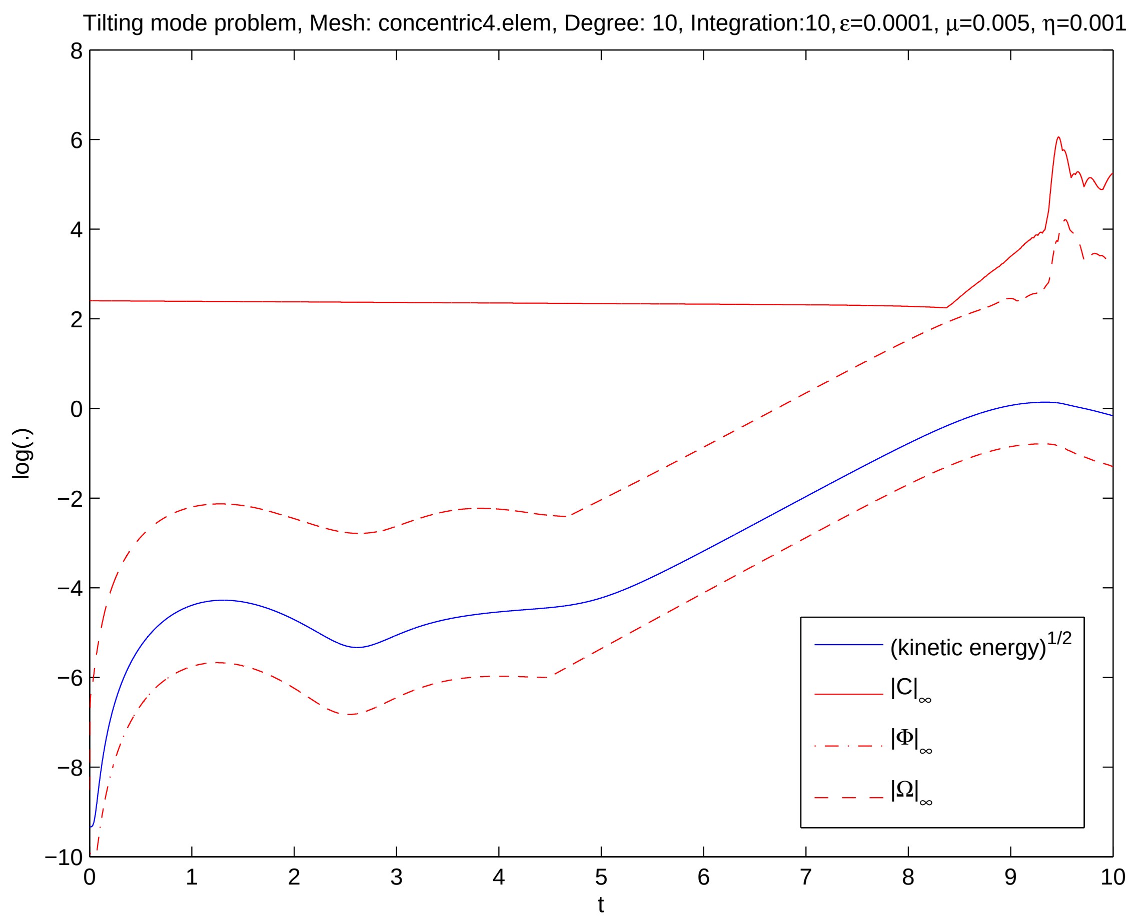




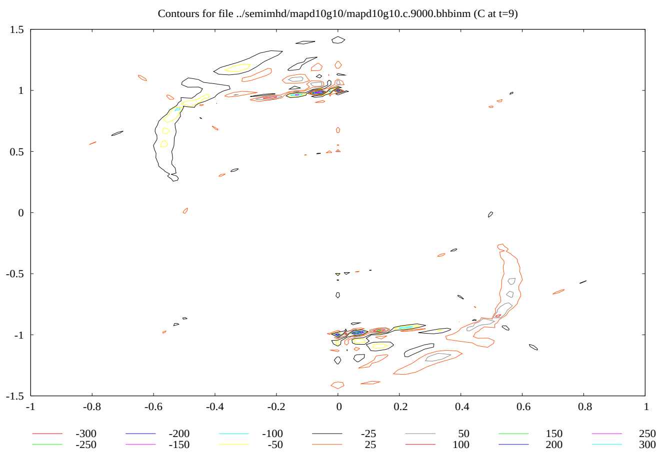
Comparing instantenous growth rates.

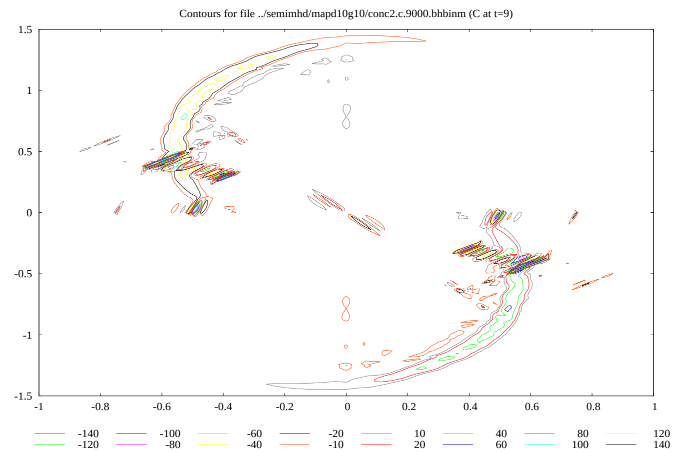


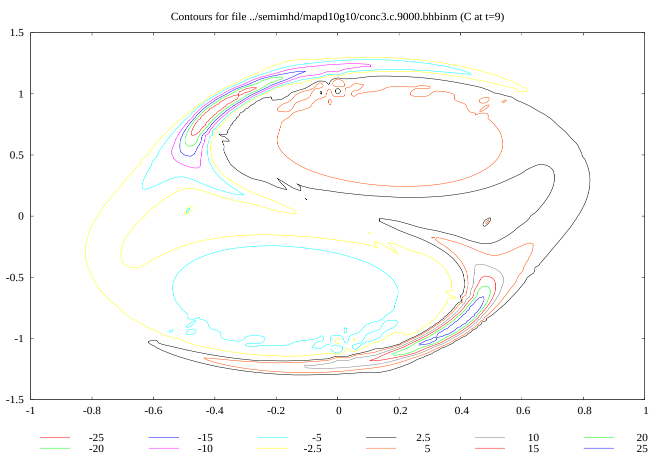
Growth of maximum of variables over time. (Mesh 4).

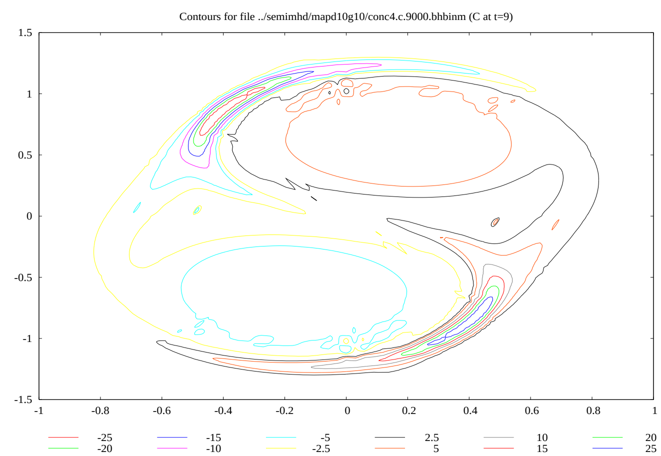


Current at $t = 9.0$ for the different meshes.

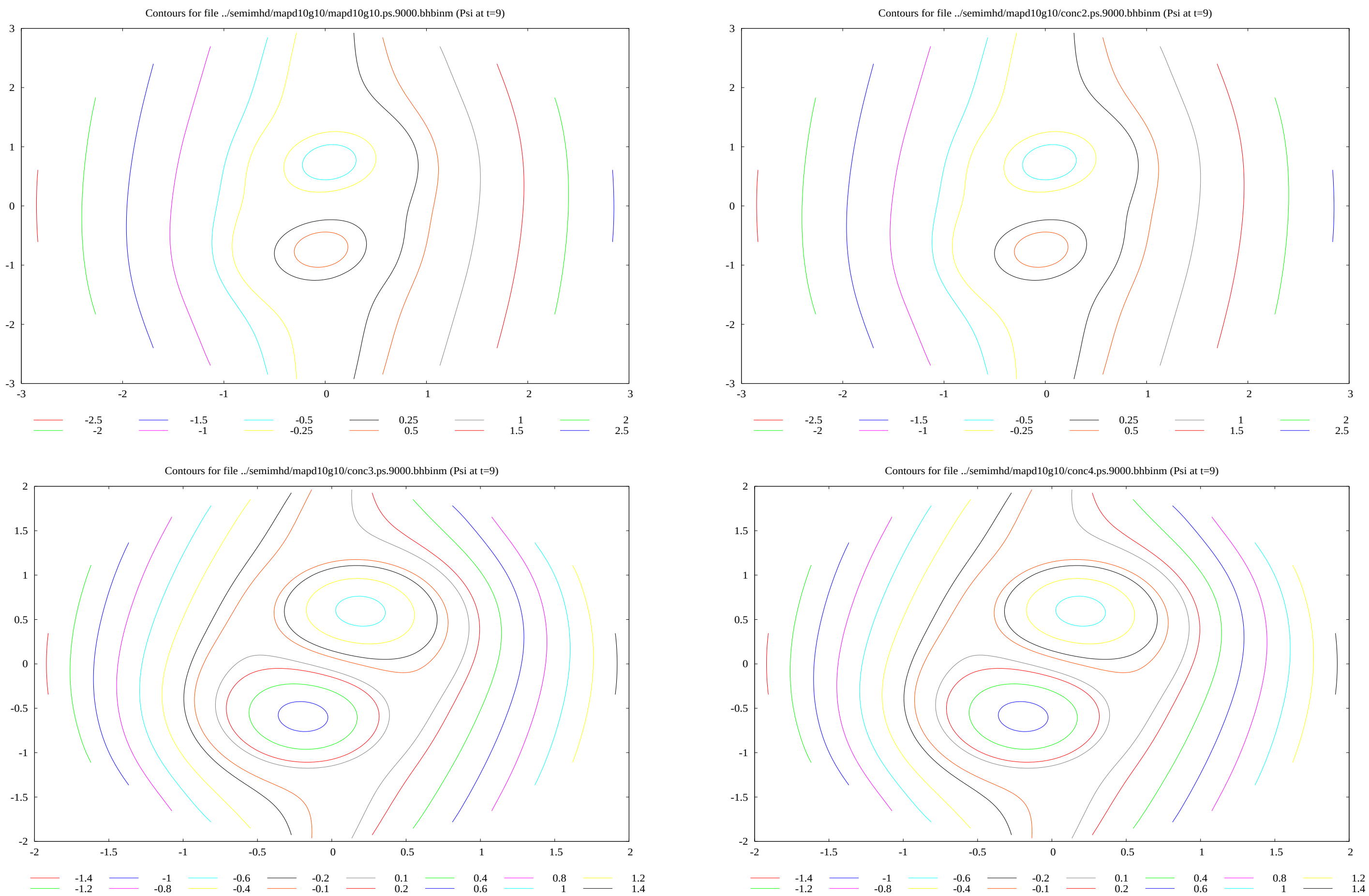




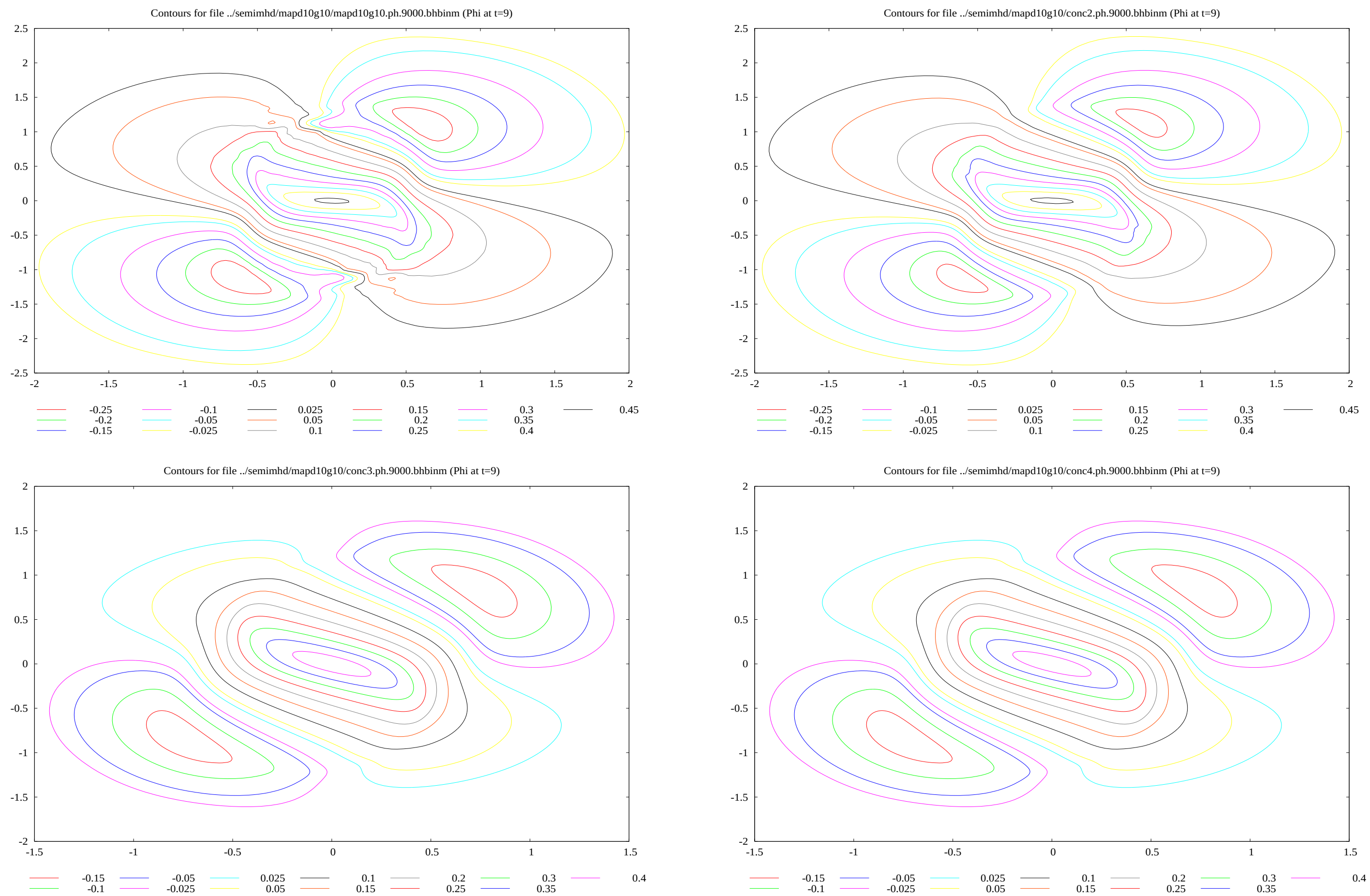




Ψ at $t = 9.0$ for the different meshes.



Φ at $t = 9.0$ for the different meshes.



Ω at $t = 9.0$ for the different meshes.