Convergence_WaveSystem_Centered_SQUARE_squares

November 13, 2018

1 Centered scheme for the Wave System

1.1 The Wave System on the square

We consider the following Wave system with periodic boundary conditions

$$\left\{\begin{array}{l} \partial_t p + c^2\nabla\cdot\vec{q} = 0 \\ \partial_t\vec{q} + \quad \vec{\nabla} p = 0 \end{array}\right. .$$

The wave system can be written in matrix form

$$\partial_t \begin{pmatrix} p \\ \vec{q} \end{pmatrix} + \begin{pmatrix} 0 & c^2\nabla\cdot \\ \vec{\nabla} & 0 \end{pmatrix} \begin{pmatrix} p \\ \vec{q} \end{pmatrix} = \begin{pmatrix} 0 \\ \vec{0} \end{pmatrix}$$

In d space dimensions the wave system is an hyperbolic system of $d+1$ equations

$$\partial_t U + \sum_{i=1}^{d} A_i \partial_{x_i} U = 0, \quad U = {}^t(p, \vec{q})$$

where the jacobian matrix is

$$A(\vec{n}) = \sum_{i=1}^{d} n_i A_i = \begin{pmatrix} 0 & c^{2t}\vec{n} \\ \vec{n} & 0 \end{pmatrix}, \quad \vec{n} \in \mathbb{R}^d.$$

has $d+1$ eigenvalues $-c, 0, \dots, 0, c$.

The wave system also takes the conservative form

$$\partial_t U_i + \nabla \cdot F(U) = 0,$$

where the flux matrix F is defined by

$$F(U)\vec{n} = A(\vec{n})U, \quad \vec{n} \in \mathbb{R}.$$

On the square domain $\Omega = [0,1] \times [0,1]$ we consider the initial data

$$\left\{\begin{array}{l} p_0(x,y) = constant \\ q_{x0}(x,y) = \sin(\pi x)\cos(\pi y) \\ q_{y0}(x,y) = -\sin(\pi y)\cos(\pi x) \end{array}\right. .$$

The initial data (p_0, q_x, q_y) is a stationary solution of the wave system.

1.2 The Centered scheme for the Wave System

The domain Ω is decomposed into cells C_i.

$|C_i|$ is the measure of the cell C_i.

f_{ij} is the interface between two cells C_i and C_j.

s_{ij} is the measure of the interface f_{ij}.

d_{ij} is the distance between the centers of mass of the two cells C_i and C_j.

The semi-discrete colocated finite volume equation is

$$\partial_t U + \frac{1}{|C_i|}\sum s_{ij} F_{ij} = 0,$$

where U_i is the approximation of U in the cell C_i,

F_{ij} is a numerical approximation of the outward normal interfacial flux from cell i to cell j usually in the upwind form

$$F_{ij} = \frac{F(U_i)+F(U_j)}{2}\vec{n} - D(\vec{n})\frac{U_i - U_j}{2}.$$

In the case of the centered scheme the upwind matrix is

$$D_{centered}(\vec{n}) = 0.$$

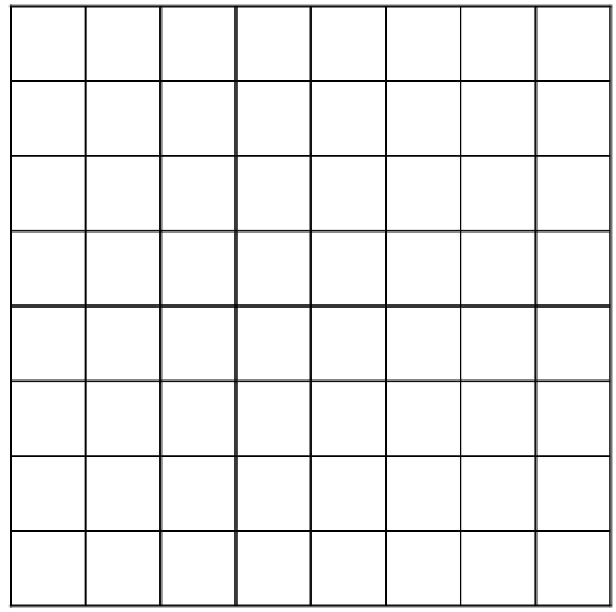
1.3 The script

```
#Condition initiale
pressure_field, velocity_field = initial_conditions_wave_system(my_mesh)

#Pas de temps
dt = cfl * dx_min / c0

#Matrice des systèmes linéaires
divMat=computeDivergenceMatrix(my_mesh,nbVoisinsMax,dt,test_bc)

# Construction du vecteur inconnu
Un=cdmath.Vector(nbCells*(dim+1))
for k in range(nbCells):
    Un[k*(dim+1)+0] =     pressure_field[k]
    Un[k*(dim+1)+1] =rho0*velocity_field[k,0]
    Un[k*(dim+1)+2] =rho0*velocity_field[k,1]


# Création du solveur linéaire
LS=cdmath.LinearSolver(divMat,Un,iterGMRESMax, precision, "GMRES","ILU")

# Time loop
while (it<ntmax and time <= tmax and not isStationary):
    LS.setSndMember(Un)
    Un=LS.solve();
    Un.writeVTK
```

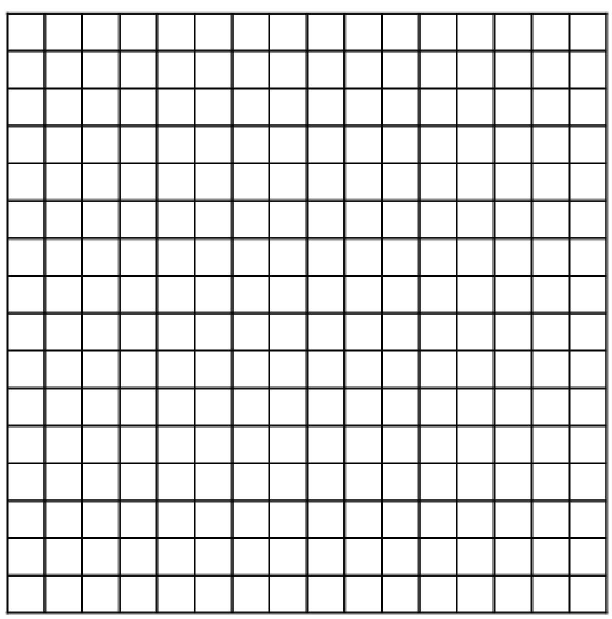
```
# Automatic postprocessing :  save 2D picture and plot diagonal data
#===========================
diag_data=VTK_routines.Extract_field_data_over_line_to_numpyArray(my_ResultField,[0,1,0],[1,0,0]
plt.legend()
plt.xlabel('Position on diagonal line')
plt.ylabel('Value on diagonal line')
if len(sys.argv) >1 :
    plt.title('Plot over diagonal line for finite volumes \n for Wave system on a 2D square '+my
    plt.plot(curv_abs, diag_data, label= str(nbCells)+ ' cells mesh')
    plt.savefig("FiniteVolumes2D_square_ResultField_"+str(nbCells)+ '_cells'+"_PlotOverDiagonalL
```

1.4 Numerical results for centered scheme on cartesian meshes

1.4.1 Cartesian meshes

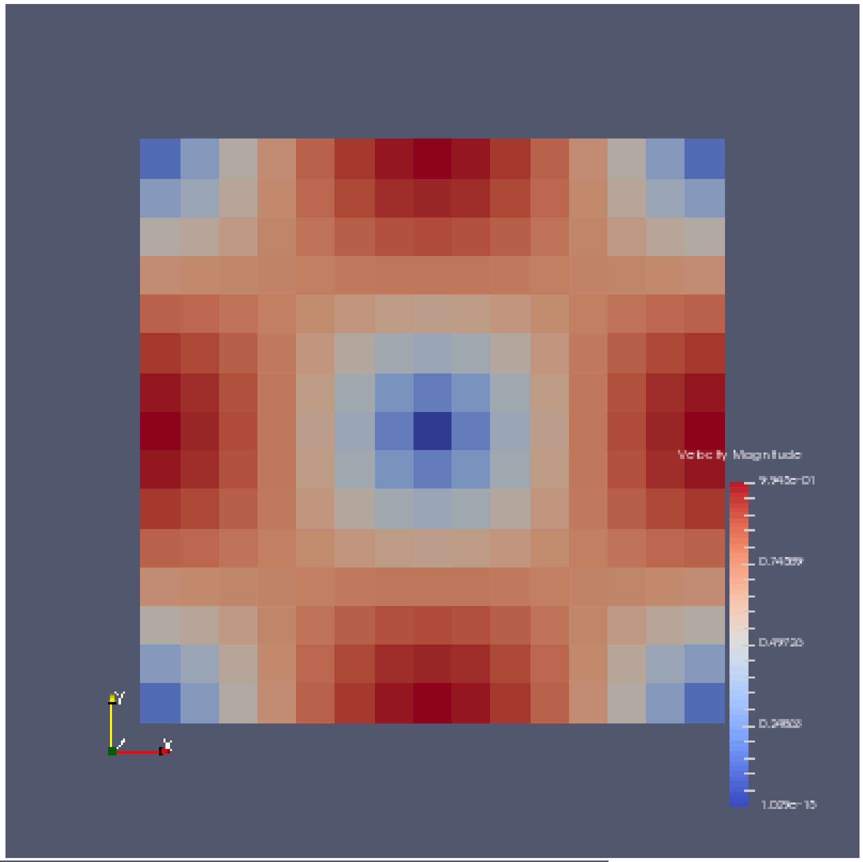


mesh 1 | mesh 2 | mesh 3 - | - - | -

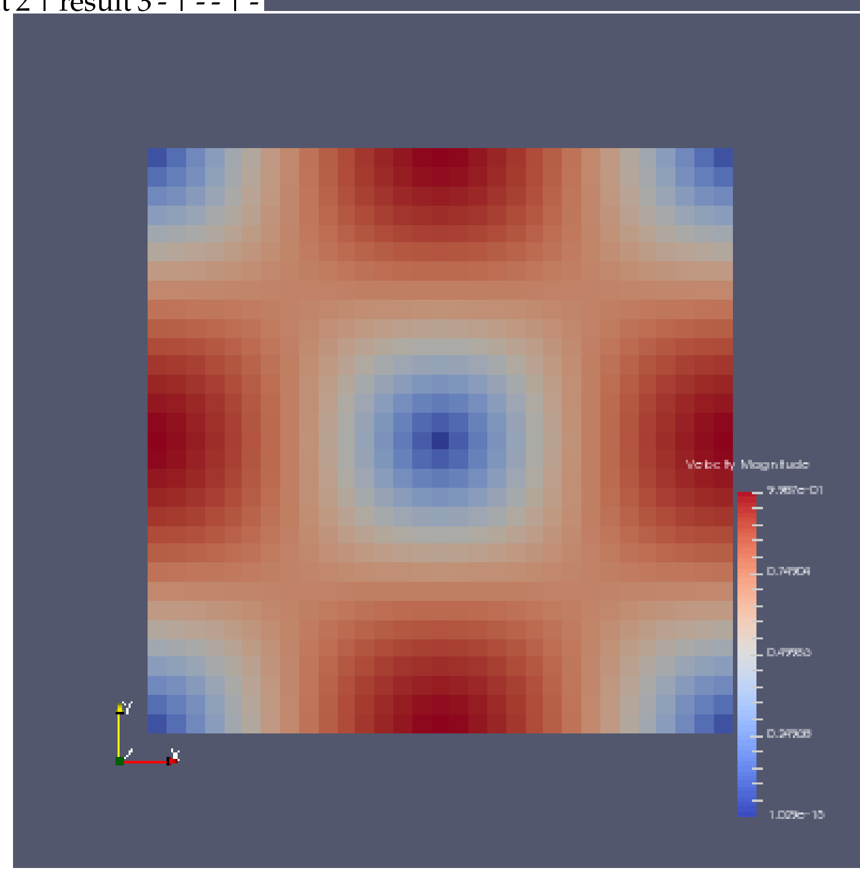


| |

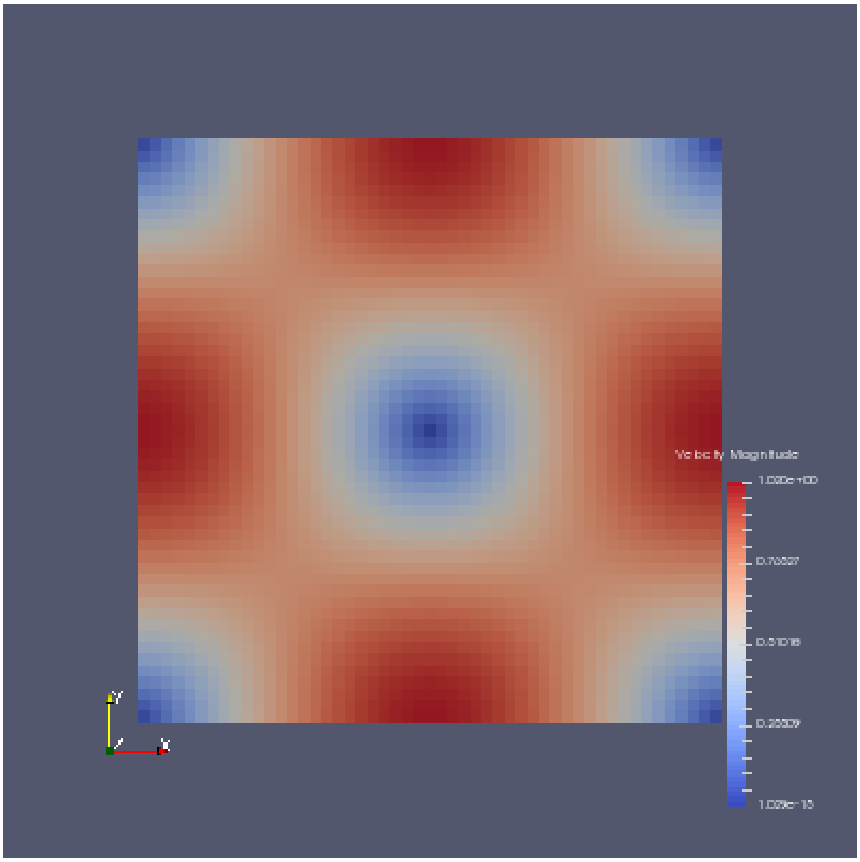
1.4.2 Velocity initial data (magnitude)



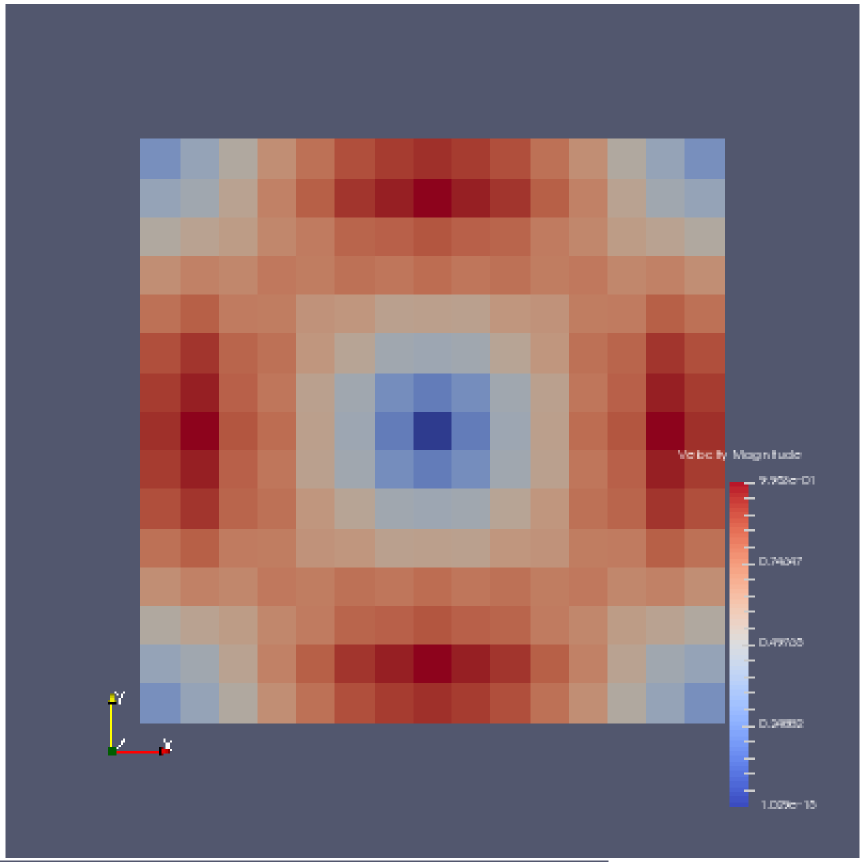
result 1 | result 2 | result 3 - | - - | -



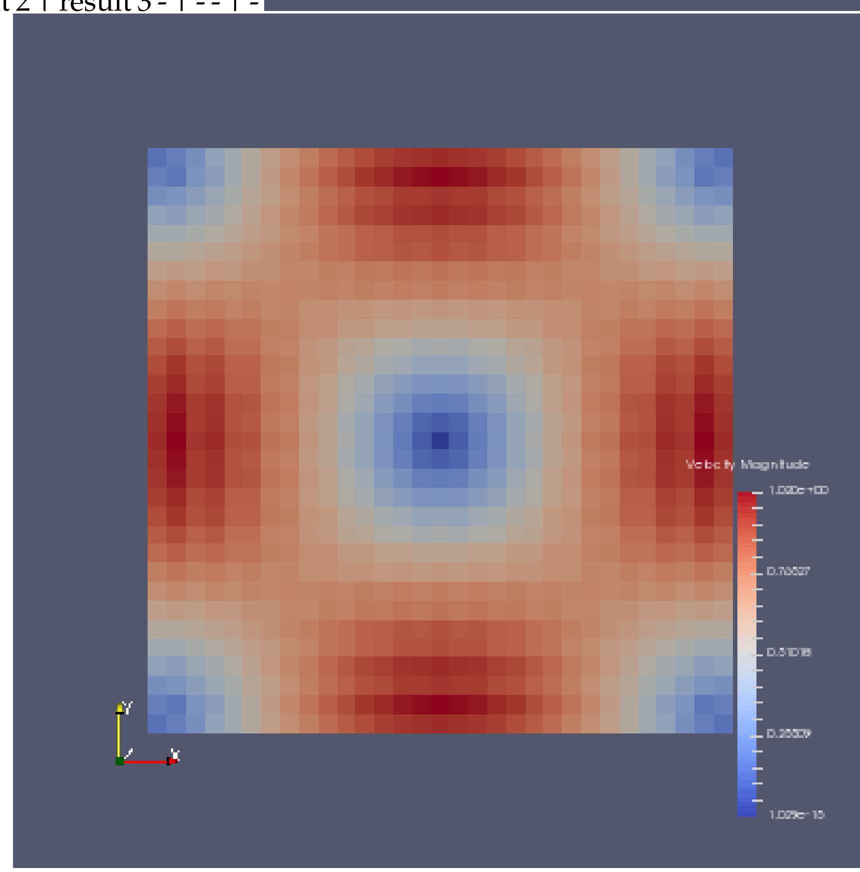
| |


Velocity Magnitude
1.020e+00
0.76527
0.51018
0.25509

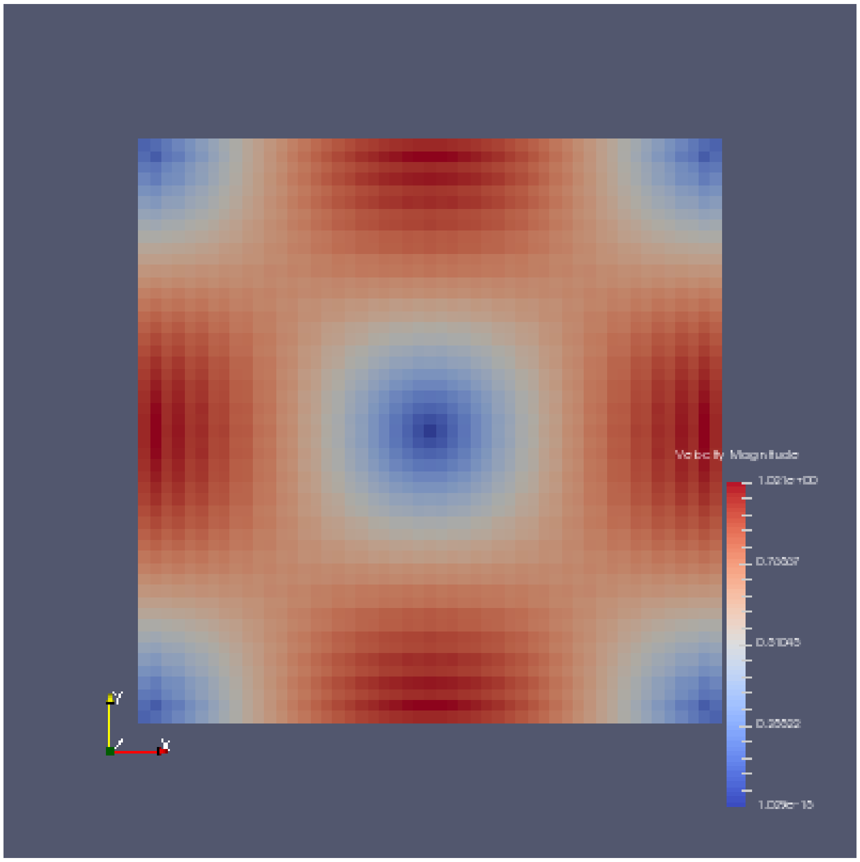
1.4.3 Stationary velocity (magnitude)

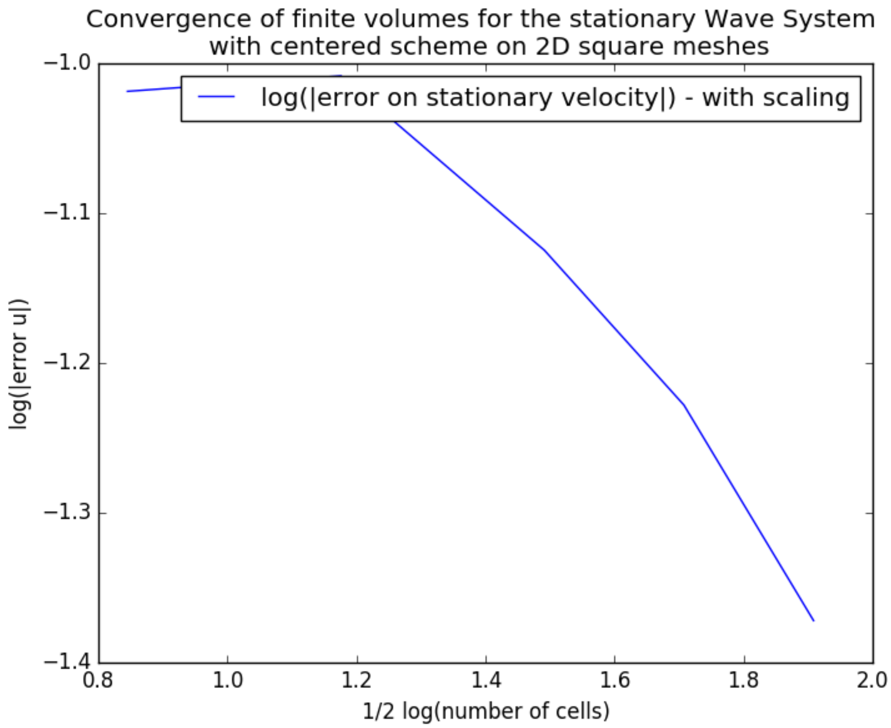


result 1 | result 2 | result 3 - | - - | -



| |

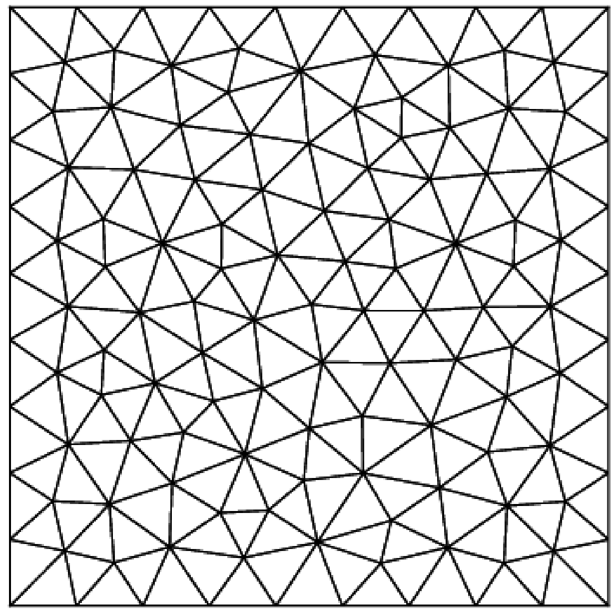

Velocity Magnitude
1.021e+00
0.51043
0.25522


Convergence of finite volumes for the stationary Wave System
with centered scheme on 2D square meshes
log(|error on stationary velocity|) - with scaling
log(|error u|)
1/2 log(number of cells)
−1.0
−1.1
−1.2
−1.3
−1.4
0.8
1.0
1.2
1.4
1.6
1.8
2.0

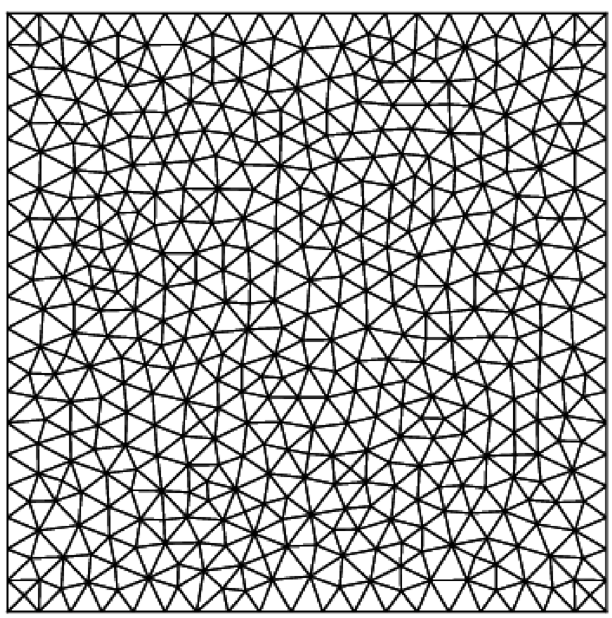
1.4.4 Convergence on stationary velocity

1.5 Numerical results for centered scheme on triangular meshes

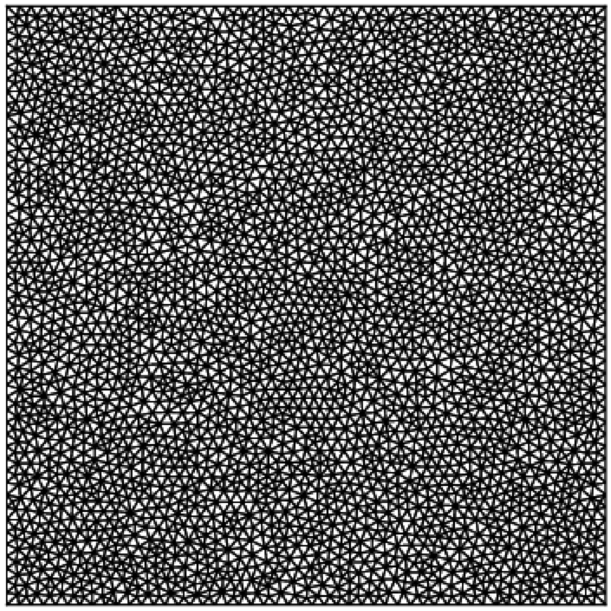
1.5.1 Triangular meshes



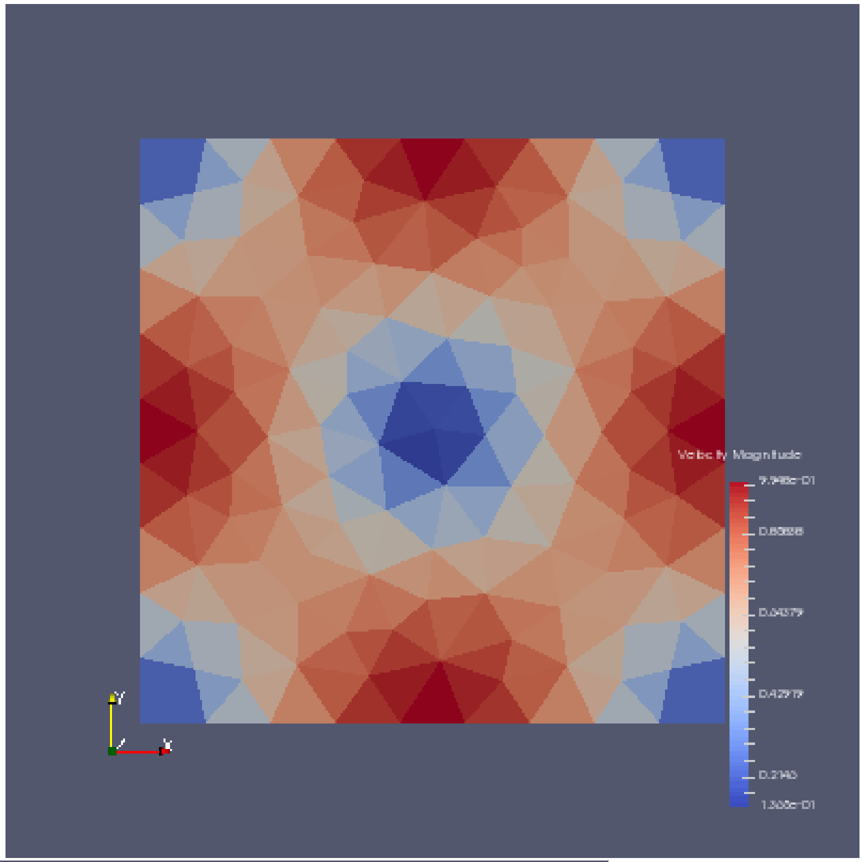
mesh 1 | mesh 2 | mesh 3 - | - - | -



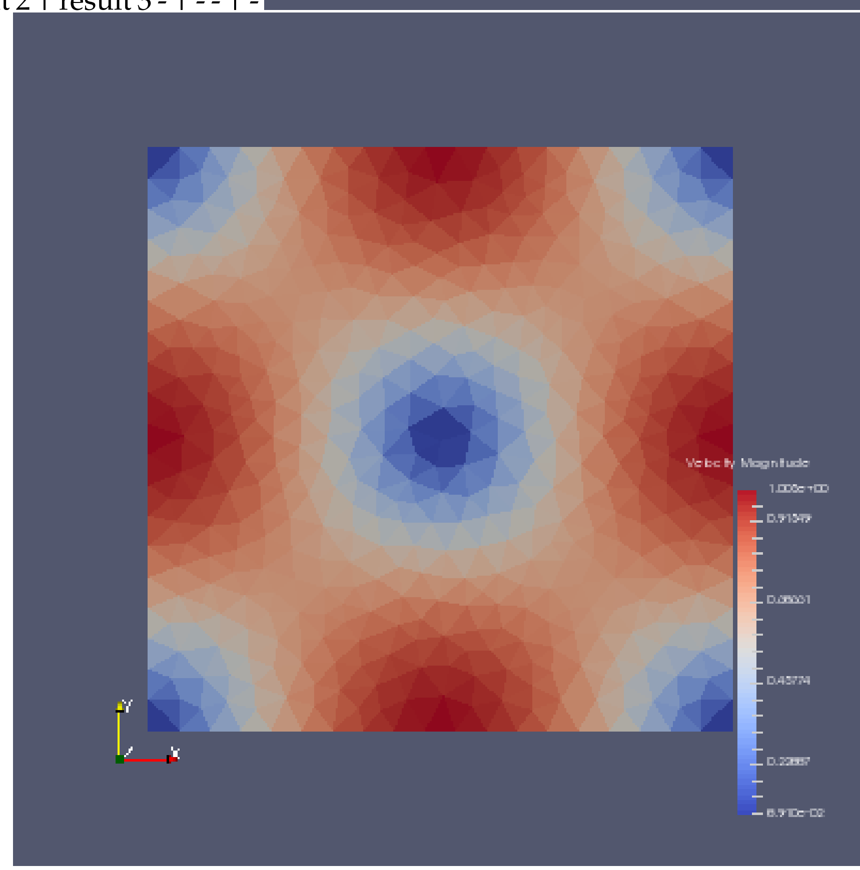
| |



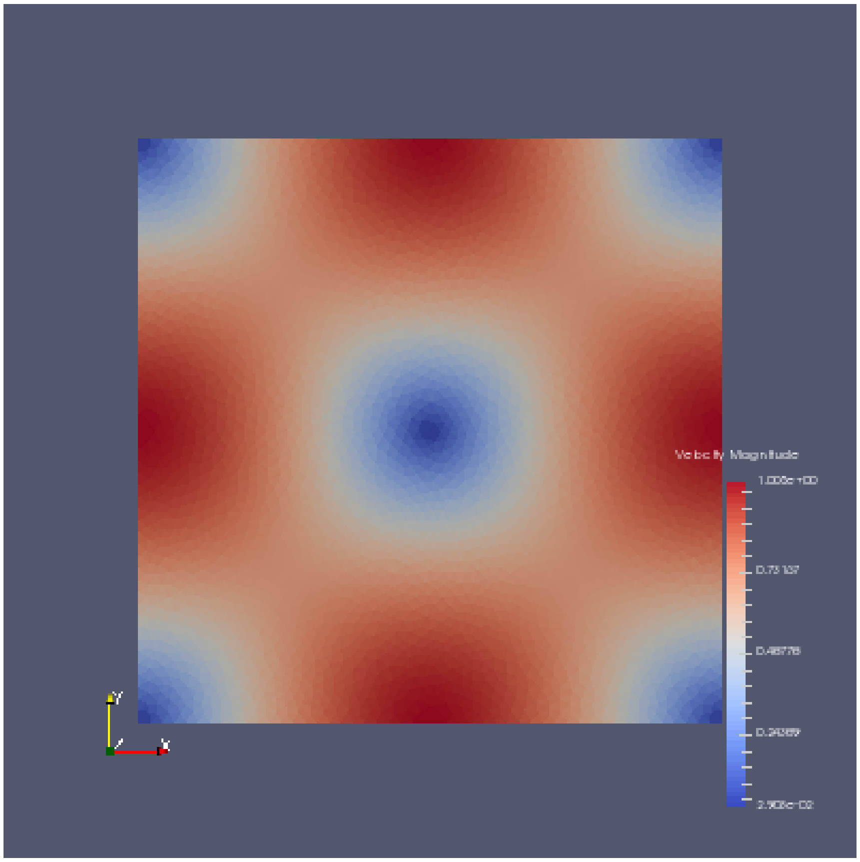
1.5.2 Velocity initial data (magnitude)



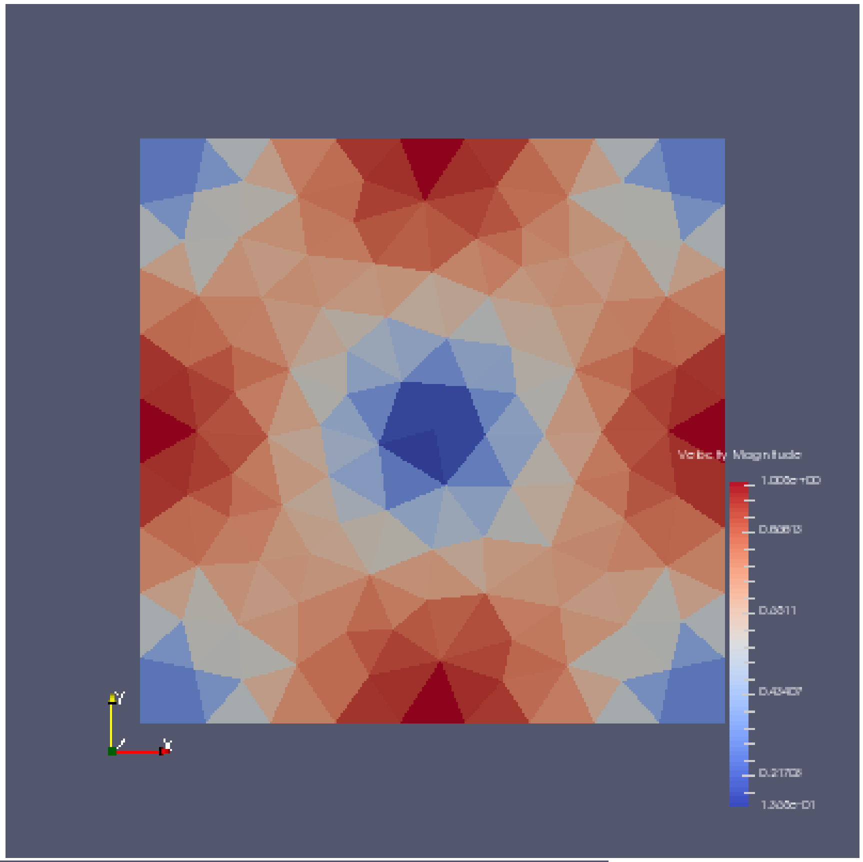
result 1 | result 2 | result 3 - | - - | -



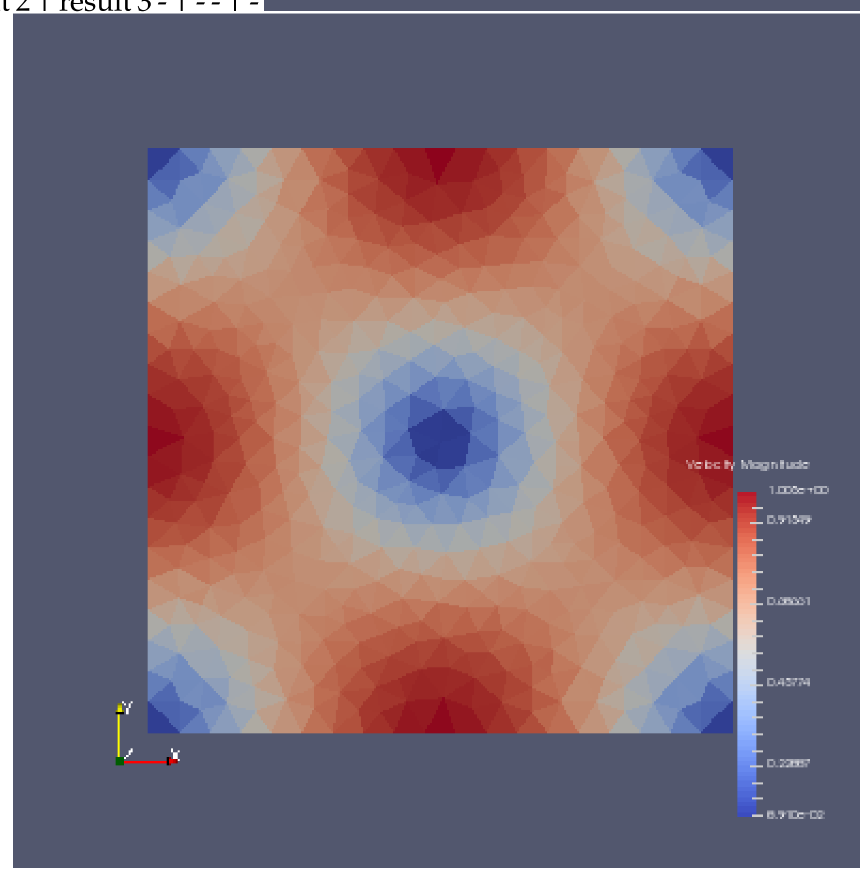
| |


Velocity Magnitude

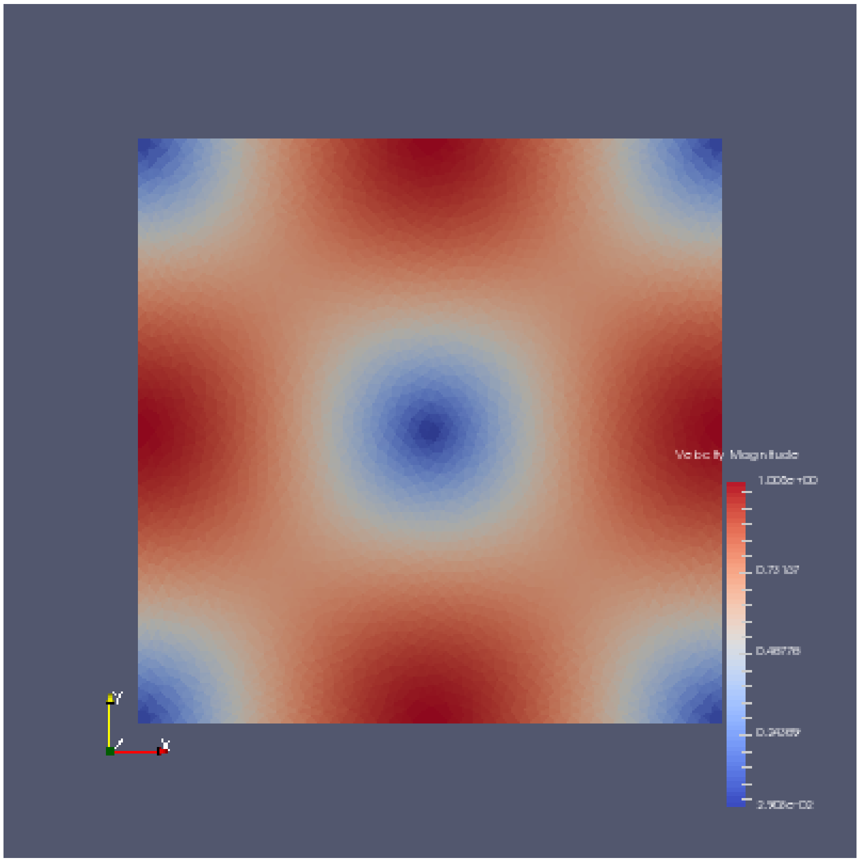
1.5.3 Stationary velocity (magnitude)

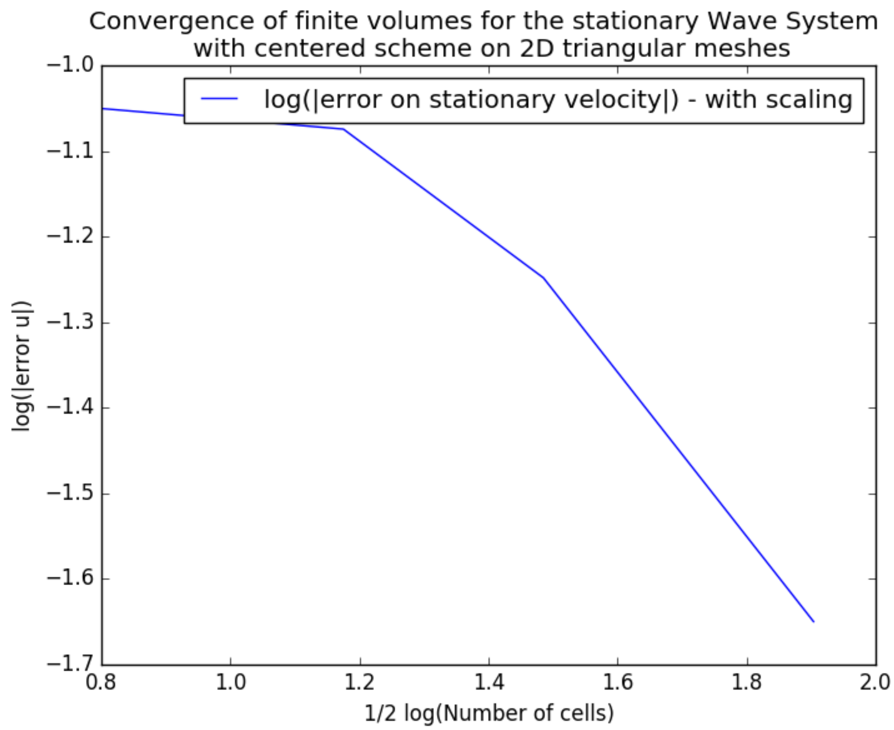


result 1 | result 2 | result 3 - | - - | -



| |

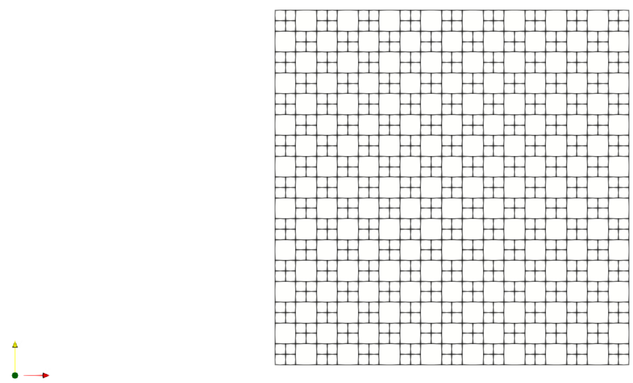

Velocity Magnitude
1.000e+00
0.73137
0.48778
0.24389
2.903e-02


Convergence of finite volumes for the stationary Wave System
with centered scheme on 2D triangular meshes
log(|error on stationary velocity|) - with scaling
log(|error u|)
1/2 log(Number of cells)
−1.0
−1.1
−1.2
−1.3
−1.4
−1.5
−1.6
−1.7
0.8
1.0
1.2
1.4
1.6
1.8
2.0

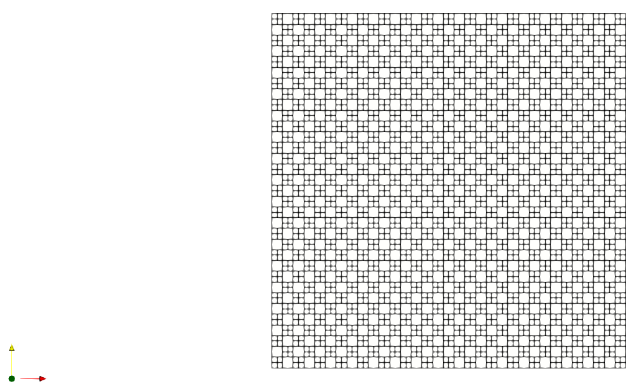
1.5.4 Convergence on stationary velocity

1.6 Numerical results for centered scheme on checkerboard meshes

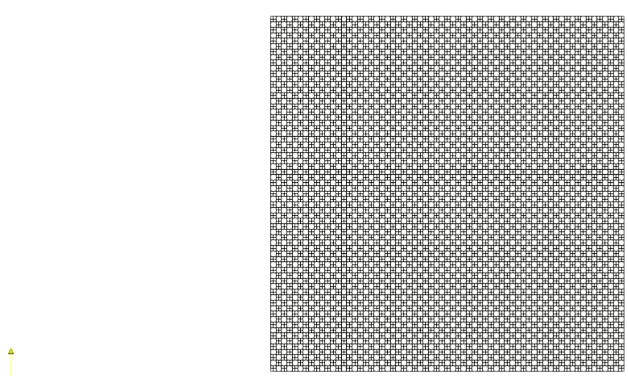
1.6.1 Checkerboard meshes



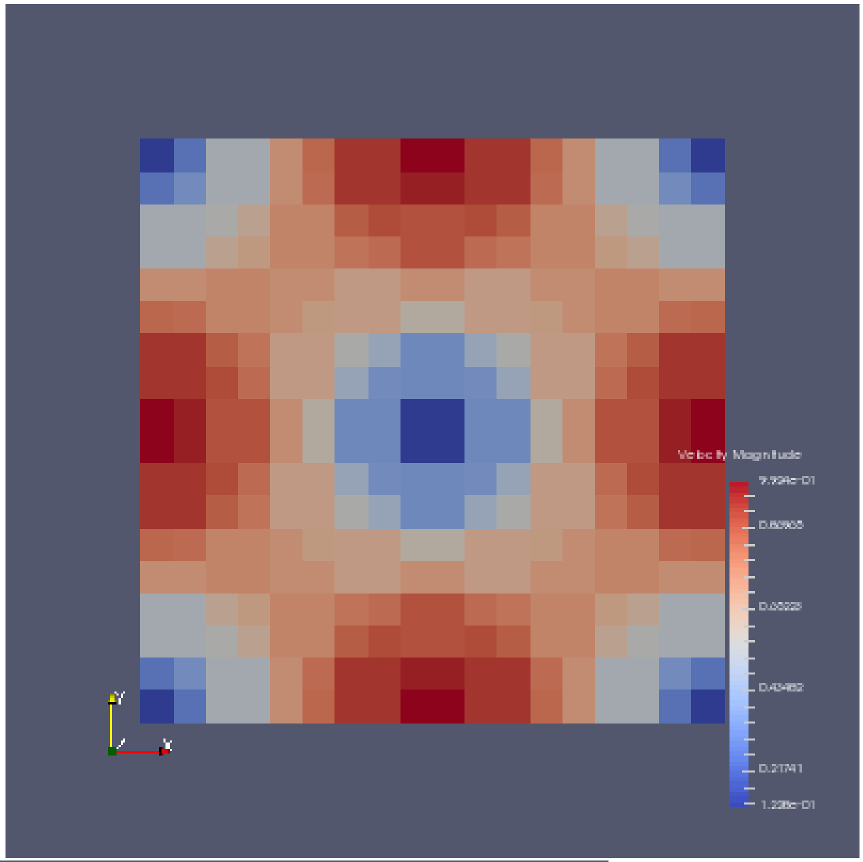
mesh 1 | mesh 2 | mesh 3 - | - - | -



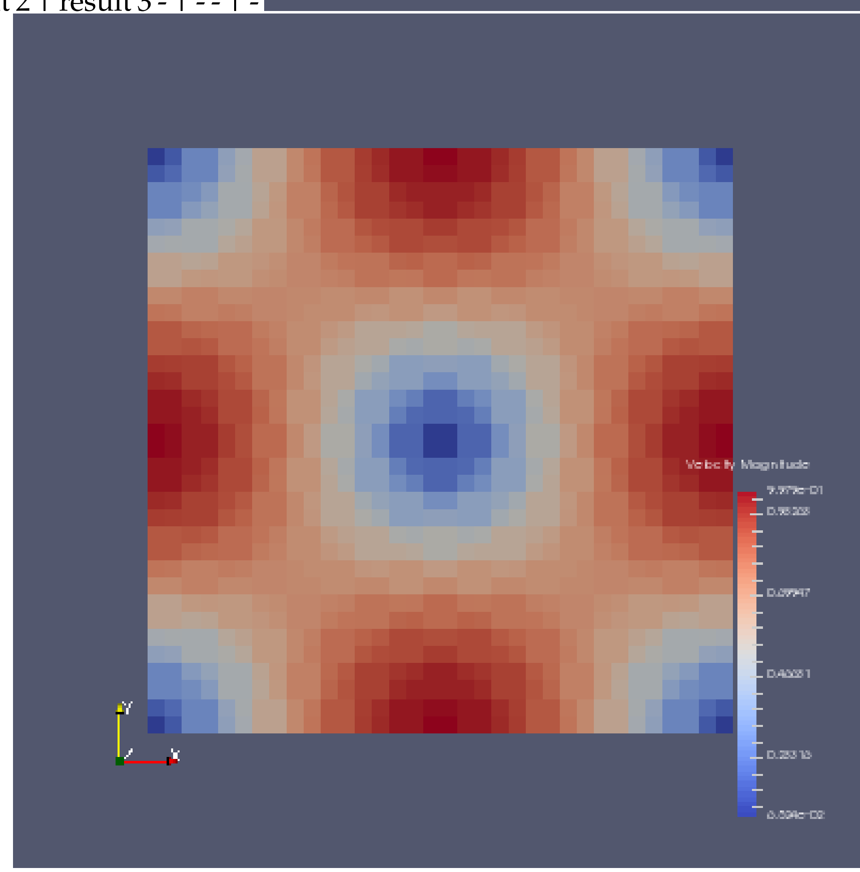
| |



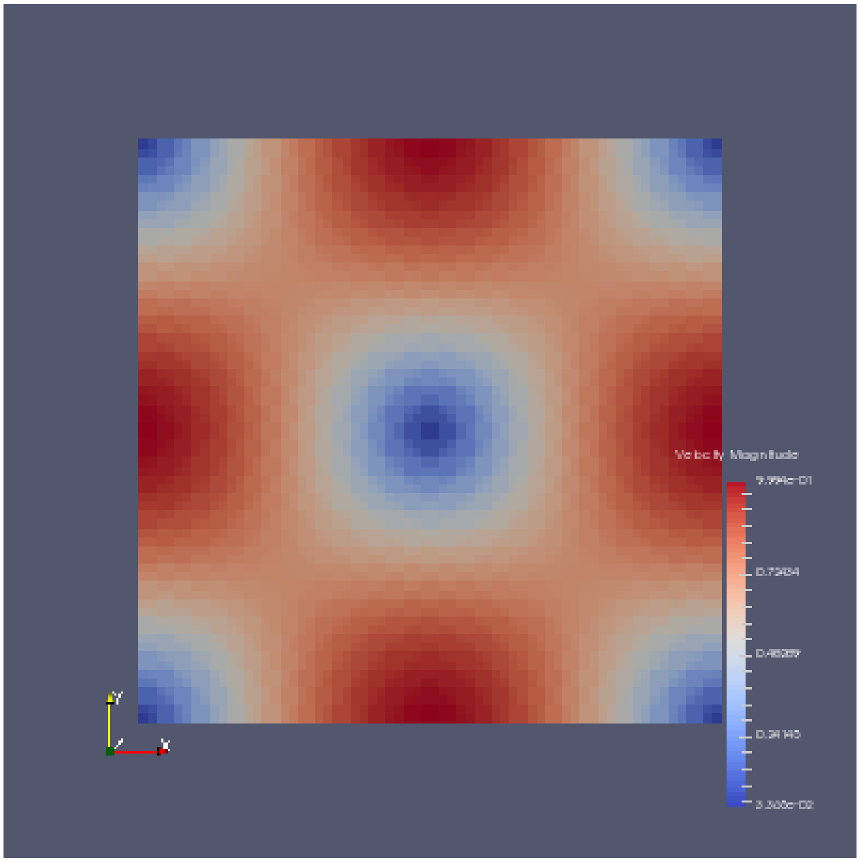
1.6.2 Velocity initial data (magnitude)



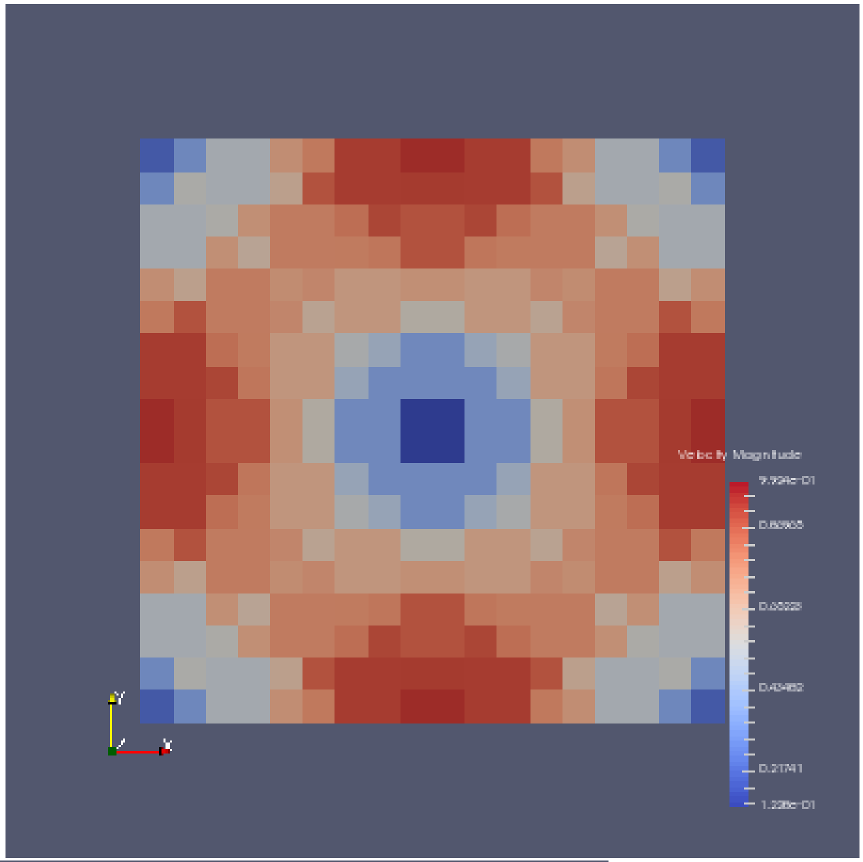
result 1 | result 2 | result 3 - | - - | -



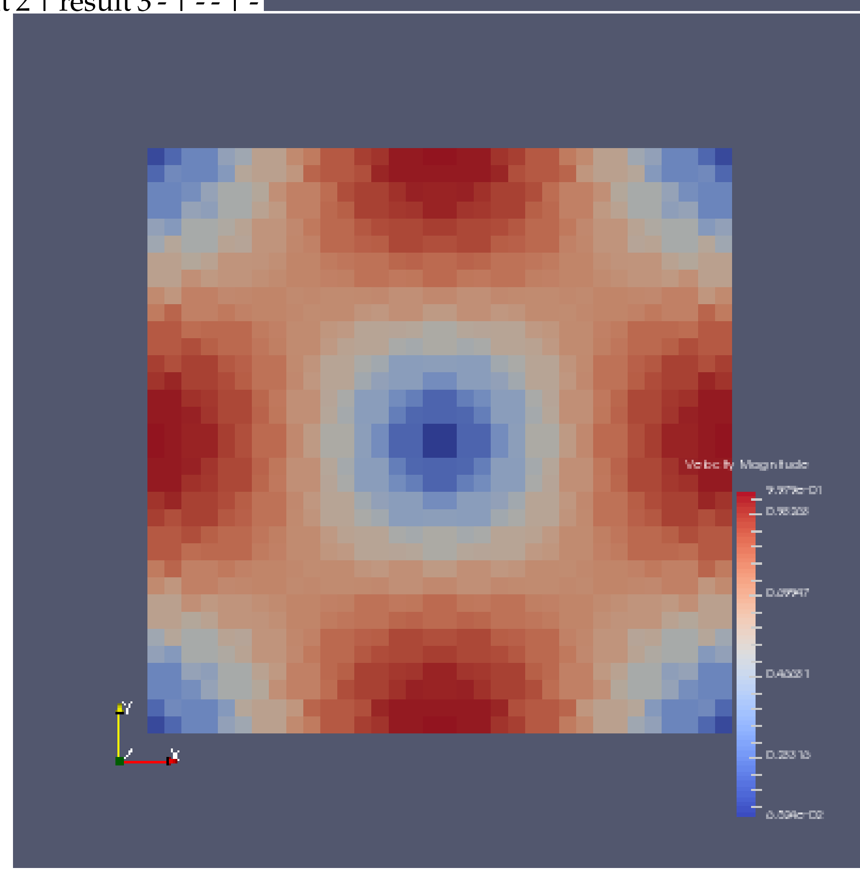
| |

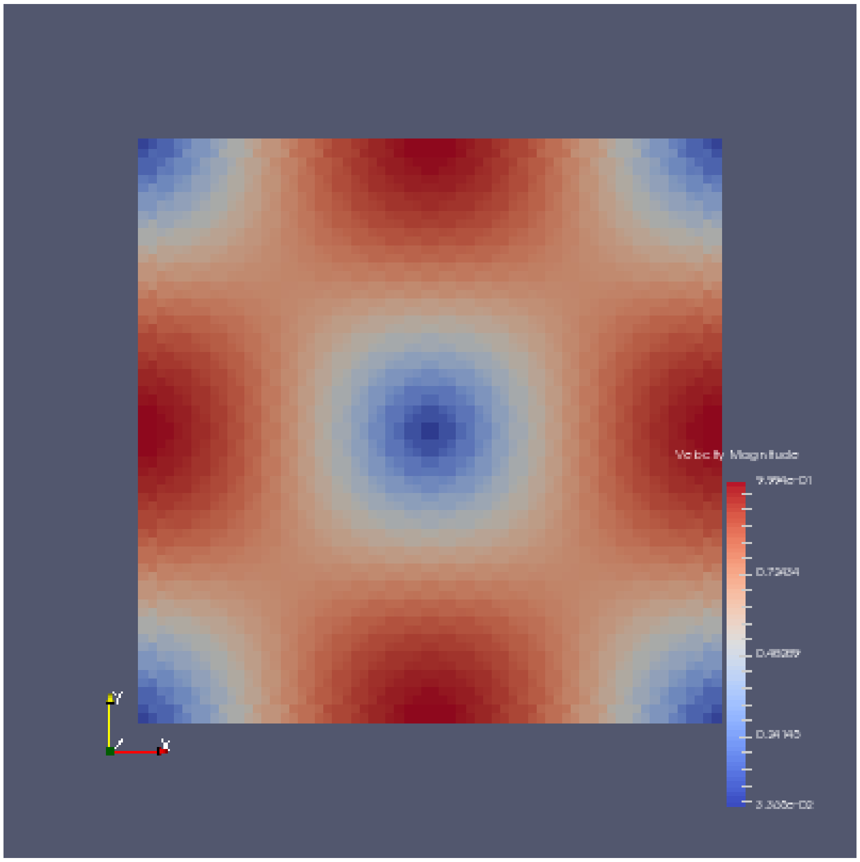

Velocity Magnitude
0.72434
0.48289
0.24145

1.6.3 Stationary velocity (magnitude)

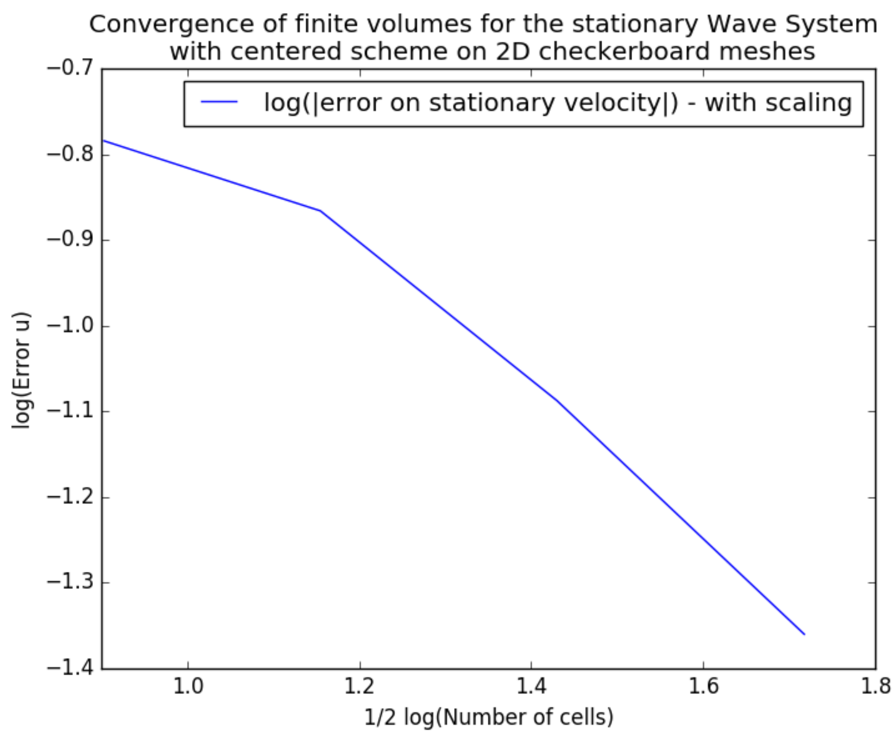


result 1 | result 2 | result 3 - | - - | -





1.6.4 Convergence on stationary velocity



Convergence of finite volumes for the stationary Wave System
with centered scheme on 2D checkerboard meshes
log(|error on stationary velocity|) - with scaling
log(Error u)
1/2 log(Number of cells)
−0.7
−0.8
−0.9
−1.0
−1.1
−1.2
−1.3
−1.4
1.0
1.2
1.4
1.6
1.8