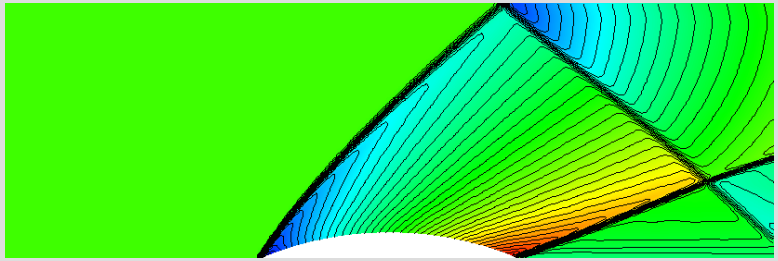
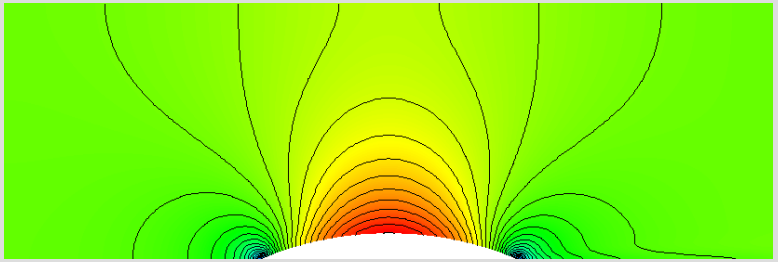
*L.Gasparini*
*Fondmetal Technologies, Italy*

*S. Mallikarjunan*
*Aerospace MS student, Univ. of Illinois*

# Further development of the compressible flow solver *centralFoam*





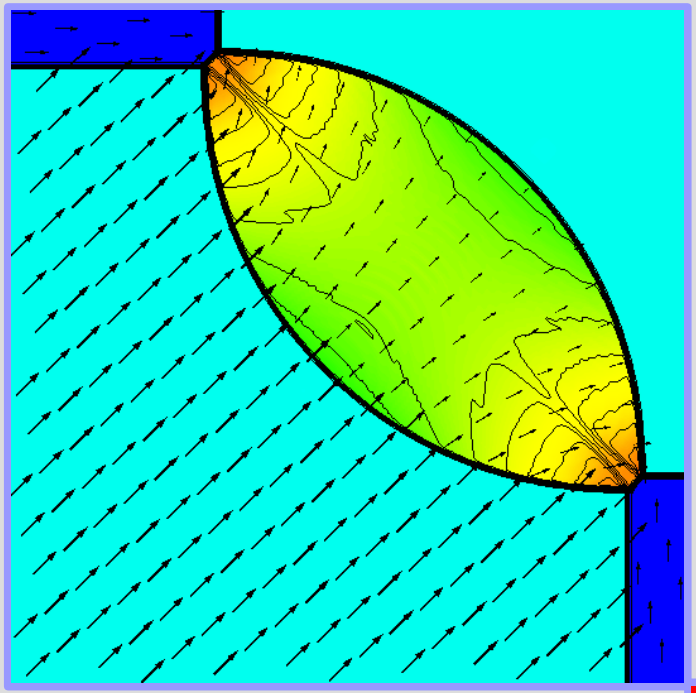
# Outline

- The solver centralFoam
- New developments
  - illustrated by a few example applications
- Conclusion

# The solver *centralFoam*

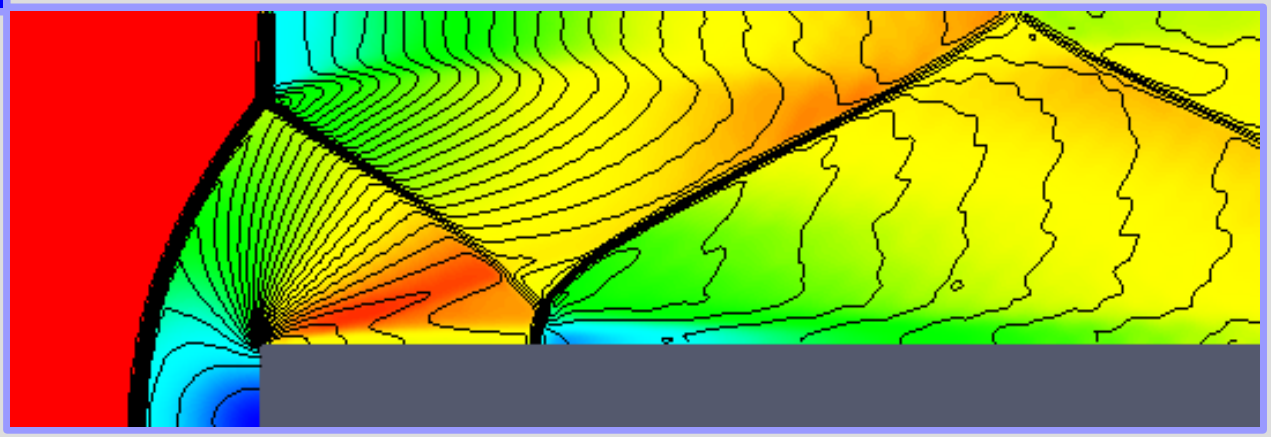
- Density-based solver for non-viscous, compressible flows, using conservative variables formulation.
- Godunov-like, Riemann-solver-free approach, using central-upwind schemes.
- Explicit (forward Euler) time integration.
- From it, *rhoCentralFoam* solver was developed, for **viscous** compressible flows, and included in OpenFOAM 1.5.
- Ref: *http://openfoamwiki.net/index.php/TestLucaG*.

# The solver *centralFoam*



Example computations:

- A 2D Riemann problem.
- Mach 3 forward step in a channel.



# New developments

- SSP, Runge-Kutta time integration.
- Time-preconditioning.
- Viscous flows (incompressible, laminar).
- Steady-state convergence acceleration:
  - Local time stepping;
  - Bulk viscosity damping;
  - Residual smoothing.
- Dual time stepping.

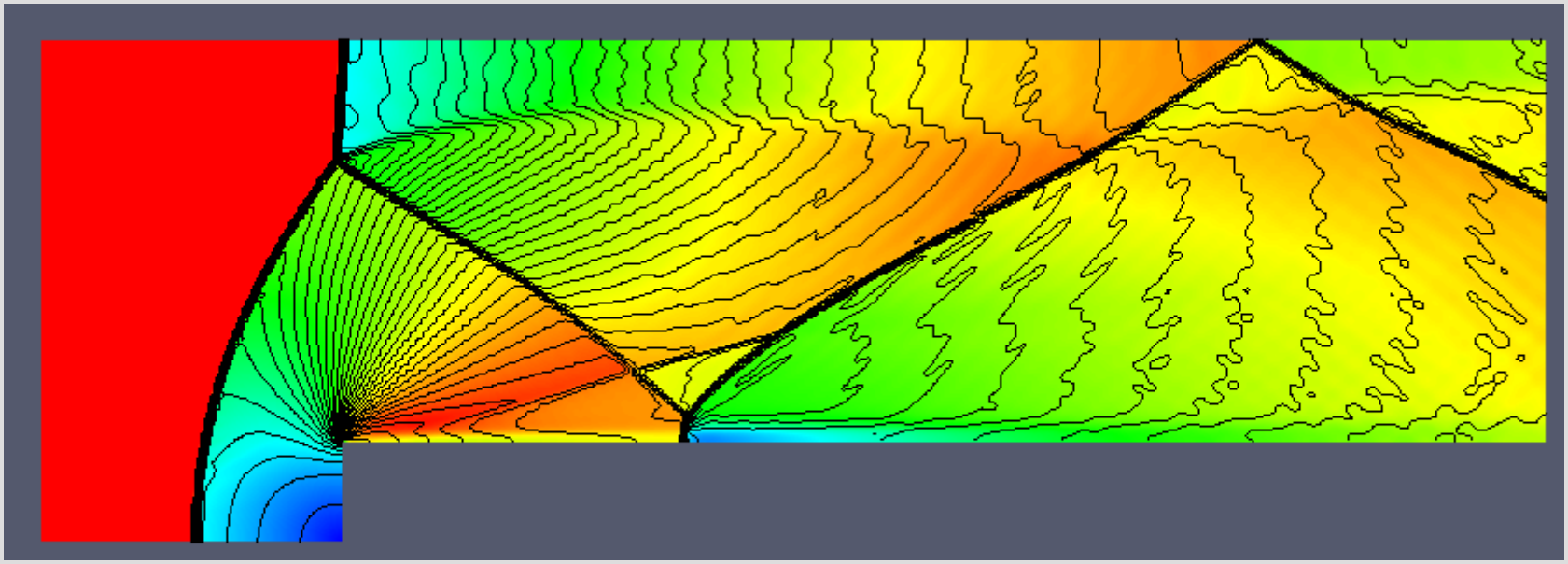
# SSP, R-K time stepping

- $2^{nd}$ and $3^{rd}$ order, multi-stage R-K scheme.
- According to theory, stability limits are:
  - CFL $\leq$ 0.5 for 1-stage, $1^{st}$ order (i.e. fwd Euler)
  - CFL $\leq$ 0.5 for 2-stages, $2^{nd}$ order
    - ⇨ *relative efficiency = 0.5*
  - CFL $\leq$ 2.0 for 5-stages, $2^{nd}$ order
    - ⇨ *relative efficiency = 0.8*
- In practice:
  - Hi-speed flow (shocks) ⇨ 2-stages, CFL ~ 0.4
    or ⇨ 5-stages, CFL ~ 1.5
  - Lo-speed (subsonic) flow ⇨ 5-stages, CFL = 2
- Ref: *Spiteri, R. J. and Ruuth, S., 2002.*

# SSP, R-K time stepping

- An example application is the 2D unsteady flow around a supersonic forward step in a channel.
- Results are shown for time integration using:
  - Fwd Euler, $CFL_{MAX} = 0.2$ (fails for $CFL_{MAX} = 0.4$)
  - R-K 2-stages, $2^{nd}$ order, $CFL_{MAX} = 0.4$
  - R-K 5-stages, $2^{nd}$ order, $CFL_{MAX} = 1.6$
- All solutions computed on a very fine, uniform, 160x384 + 40x96 cells grid.
- Ref: *Woodward, P. and Colella, P., 1984.*

# Forward step at M=3

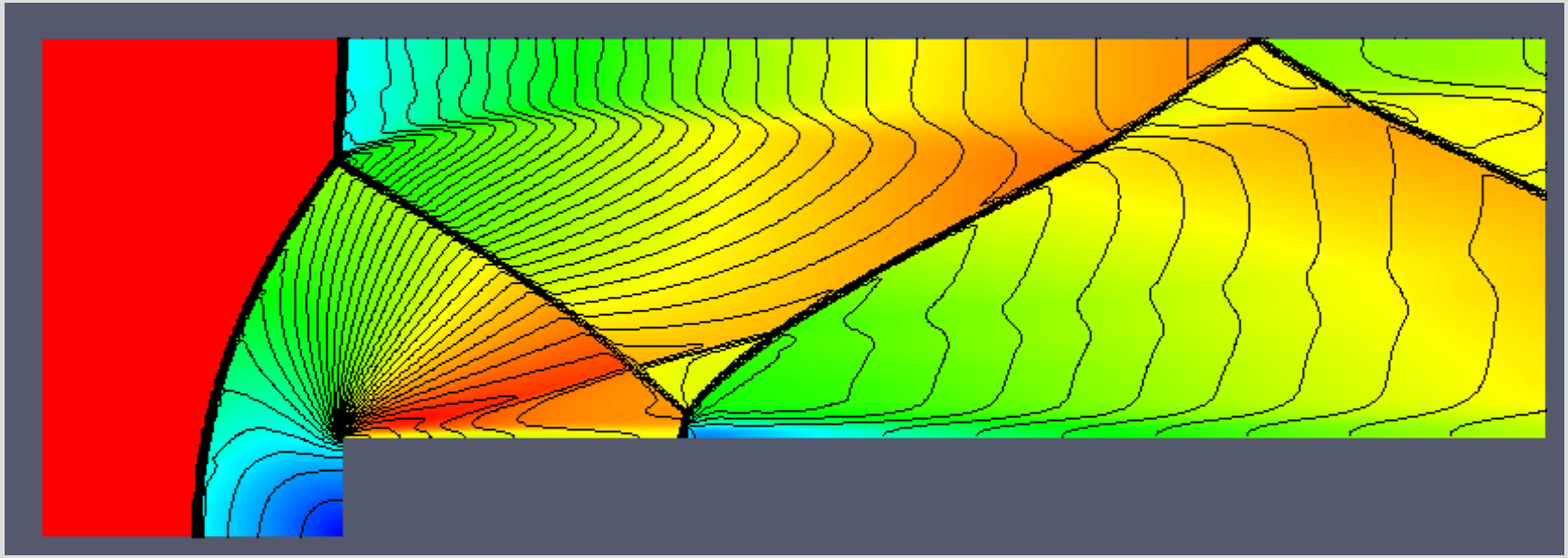


Fwd Euler, $CFL_{MAX} = 0.2$

Execution time = 8761s

*Velocity magnitude by colour, density by contours*

# Forward step at M=3

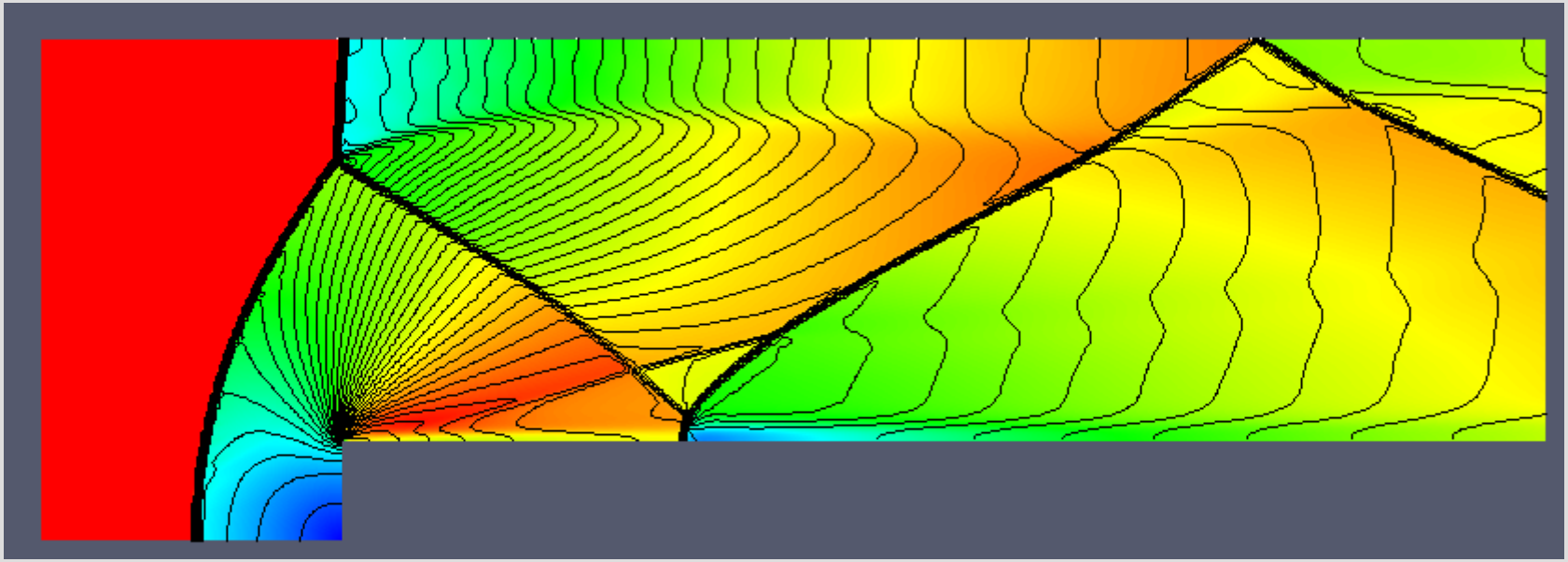


R-K 2-stages, $CFL_{MAX} = 0.4$

Execution time = 8495s

*Velocity magnitude by colour, density by contours*

# Forward step at M=3



R-K 5-stages, $CFL_{MAX} = 1.6$

Execution time = 5275s

*Velocity magnitude by colour, density by contours*

# Time-preconditioning

- *Problems:*
  - Convergence for low Mach number flows is poor due to system stiffness, as a result of disparate particle and acoustic velocities.
  - Numerical viscosity becomes too high.
- Time-preconditioning allows:
  - Computation of flows down to very low Mach..
  - ..as well as of truly constant density flows..
  - ..within the framework of a compressible flow solver.
- Ref: *Weiss, J. M. and Smith, W. A., 1995.*

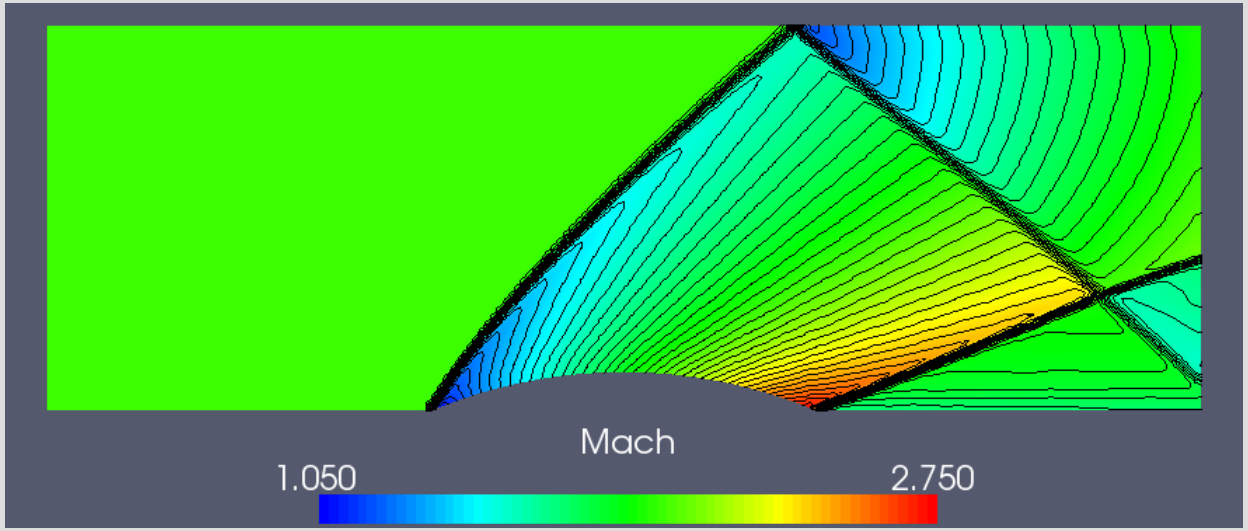
# Time-preconditioning

- Flow solver re-formulated for conservative or primitive variables, chosen at run-time:
  - Conservative variable formulation is simpler, and thus faster when applicable.
  - Primitive variable formulation is required for extremely low Mach or truly constant density flows.
- Time-preconditioning is optionally applied to both formulations and specializes for truly constant density flow.
- Preconditioner works for both inviscid and viscous flows.

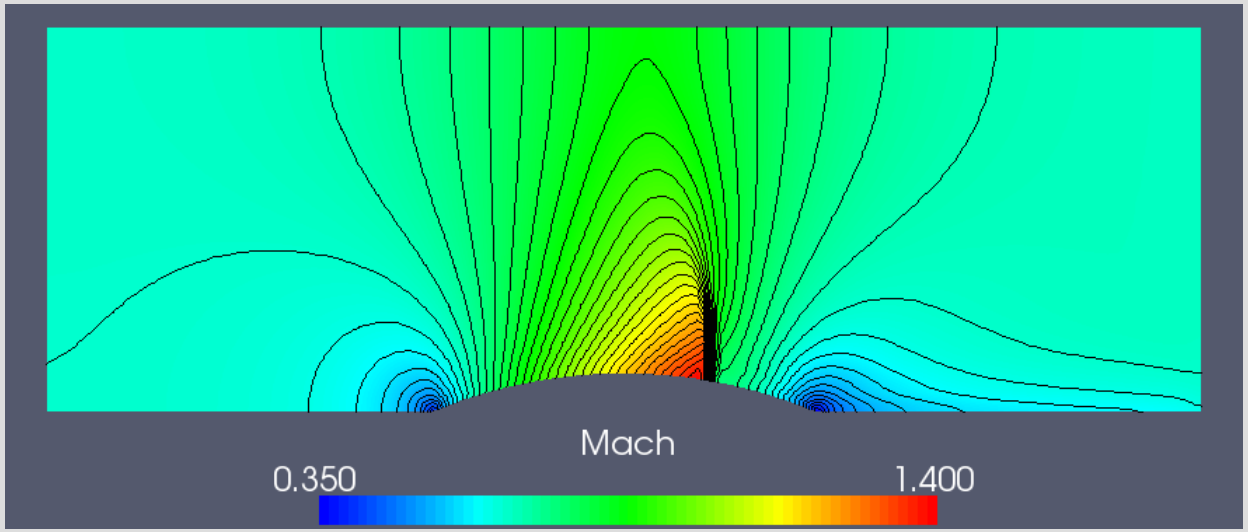
# Time-preconditioning

- An example application is the 2D inviscid flow on a 10% thick circular arc bump in a channel.
  - Changing inlet Mach number from 2, to 0.675, and to 0.01, supersonic, transonic and incompressible flows are obtained.
  - All cases shown solved using primitive variable formulation with preconditioning **ON** and same solver options except for R-K scheme and CFL.
  - Solutions are computed on a 192x128 cells uniform grid.

# Circular arc bump

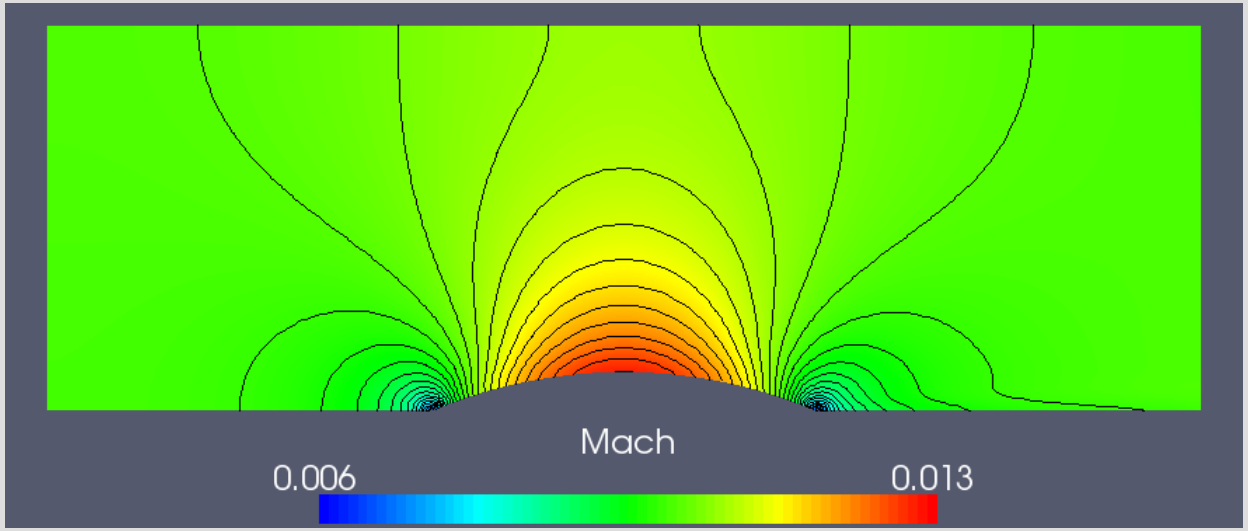



Mach = 2.0

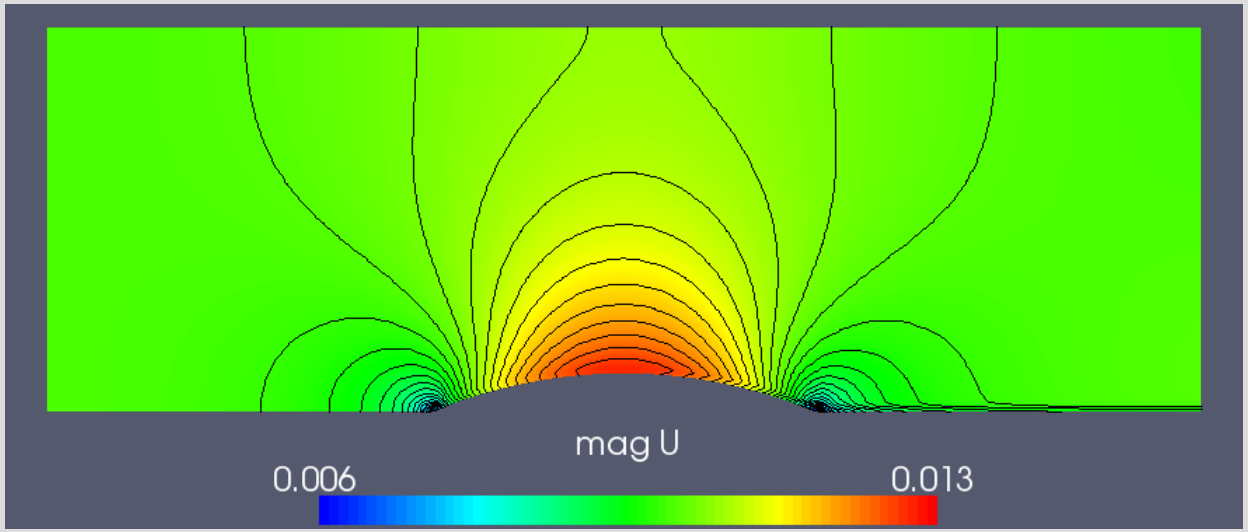



Mach = 0.675

# Circular arc bump




Mach = 0.01




*simpleFoam*
(incompressible)

# Viscous flow

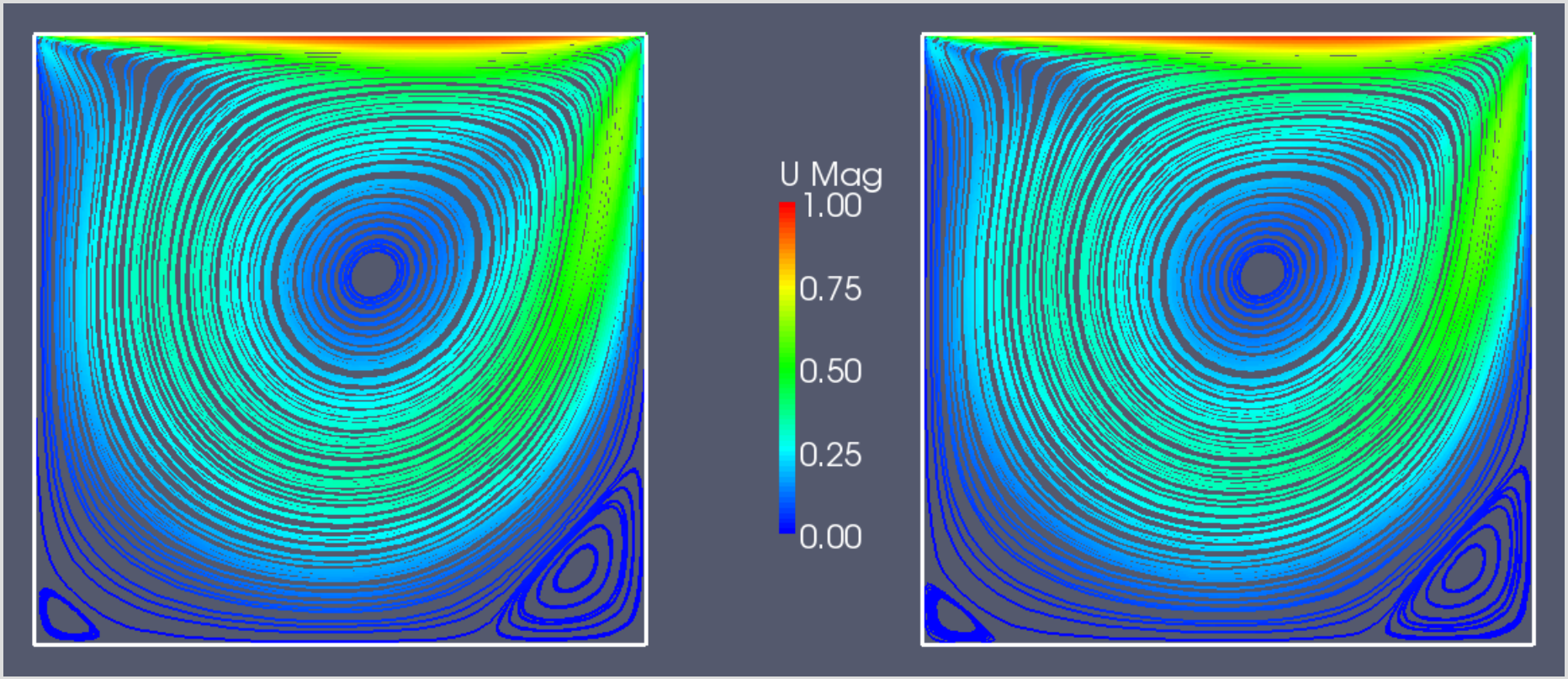
- Viscous terms have been added using a fully explicit formulation.
- This results in a viscous limitation on the time step, based on the von Neumann number, of the form:

  `maxNe` < O(1)
- At present, viscous terms have been added only for incompressible laminar flows; extension to compressible turbulent flow is, in principle, straightforward (thanks to OpenFoam technology).

# Viscous flow

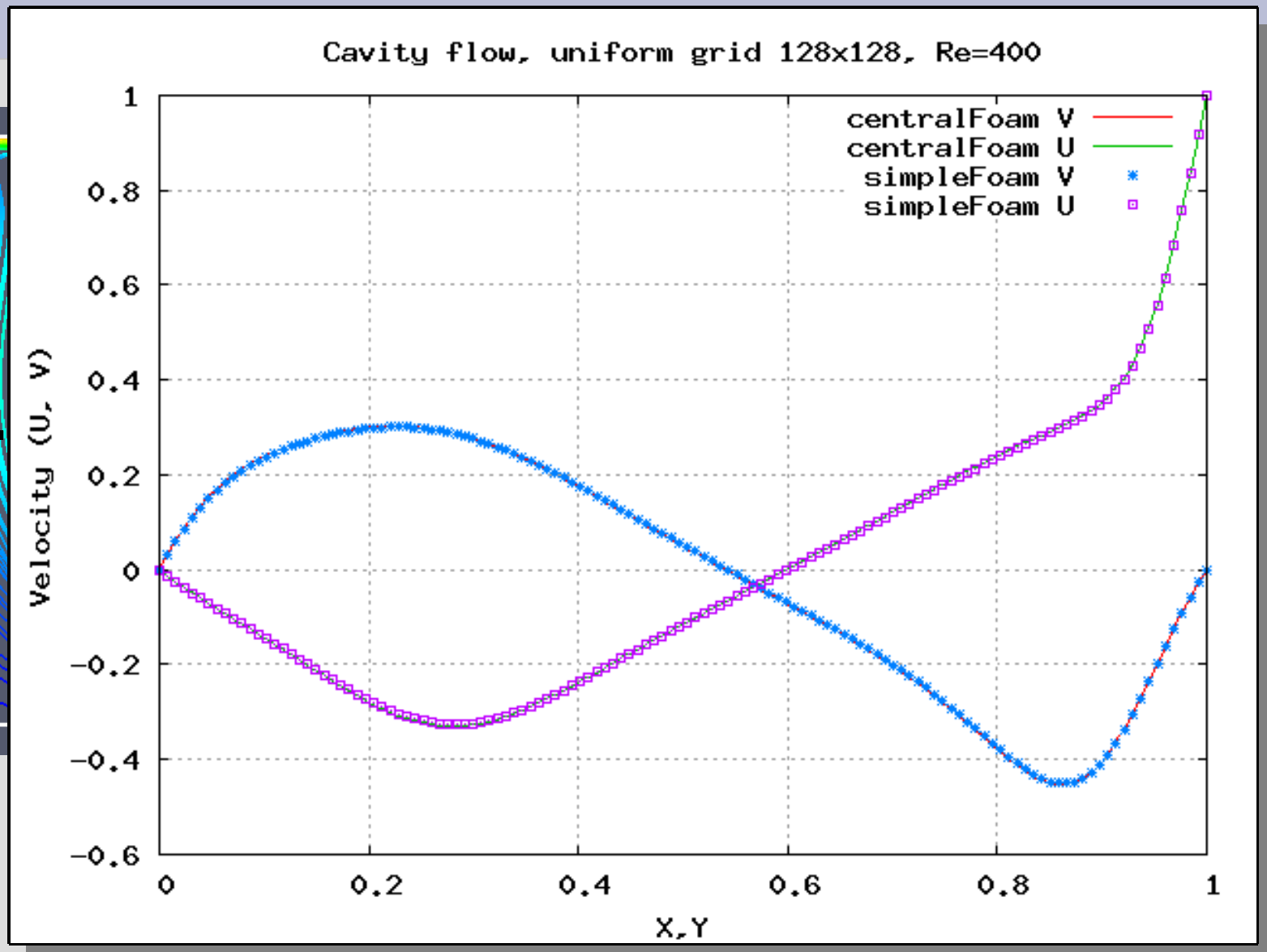
- A first example application is the very basic 2D laminar viscous flow in a square cavity.
  - The solution for the case of Re=400 is computed on a 128x128 cells uniform grid.
- A second classical test case is the 2D laminar flow around a circular cylinder in a channel.
  - At Re=20 the solution is steady.
  - The solution is computed on 16K cells unstructured grid.
  - Ref: *Schäfer, M. and Turek, S.*

# Lid-driven cavity

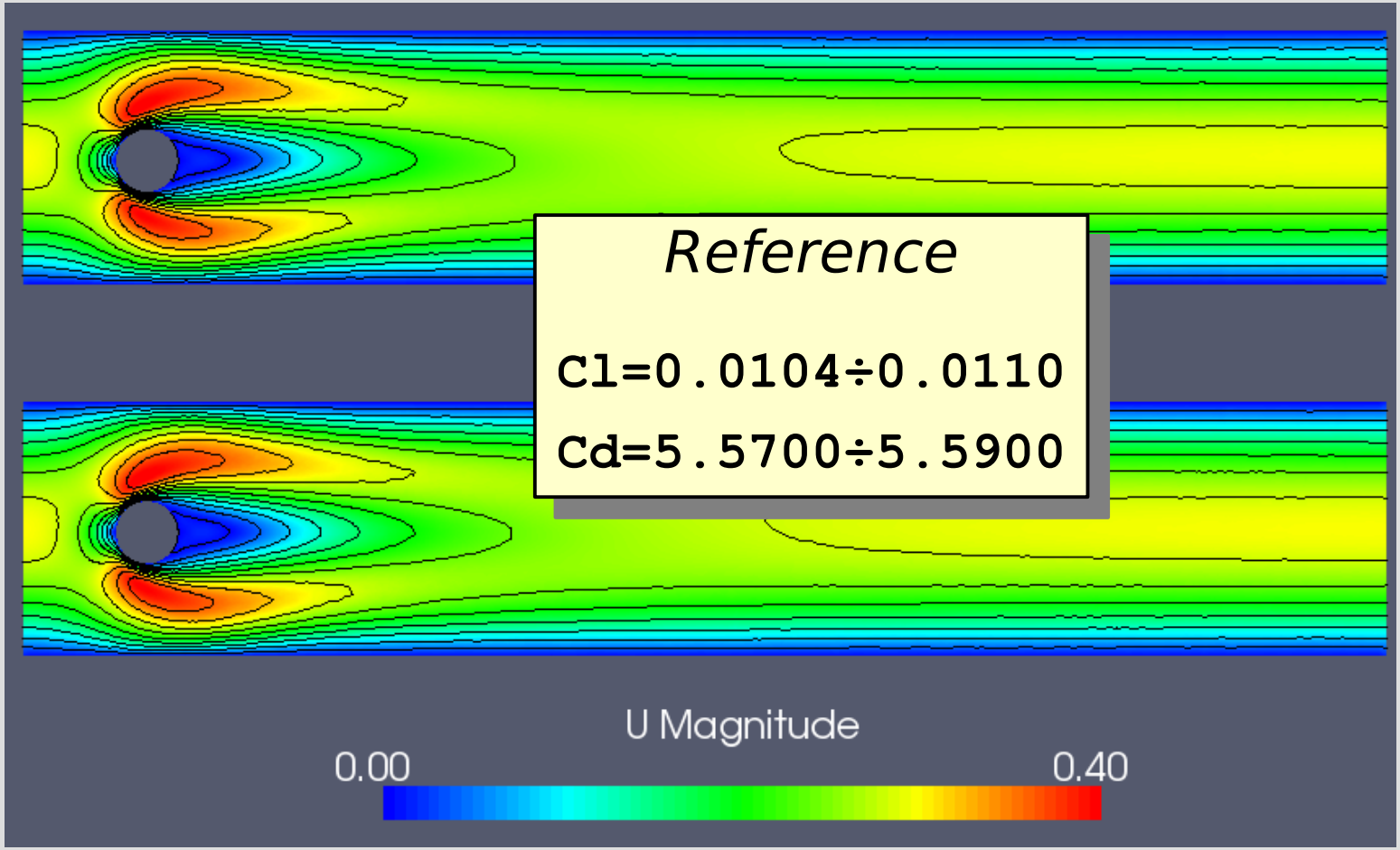



*centralFoam* *simpleFoam*

# Lid-driven cavity



# 2D circular cylinder Re=20




*centralFoam*

**Cl=0.0147**

**Cd=5.5974**

*icoFoam*

**Cl=0.0185**

**Cd=5.6366**

# Convergence acceleration

- Old, well-known techniques are applied to accelerate convergence to **steady-state**.
- Local time stepping:
  - To advance solution at the fastest possible rate in each cell.
- Bulk viscosity damping:
  - To help damping pressure waves.
  - Ref: *Mazaheri, K. and Roe, P. L., 2003.*
- Residual smoothing:
  - To increase the stability limit (CFL) of the R-K time stepping schemes and to provide additional error damping mechanism.
  - Ref: *Enander, R. and Sjögreen, B., 1996.*

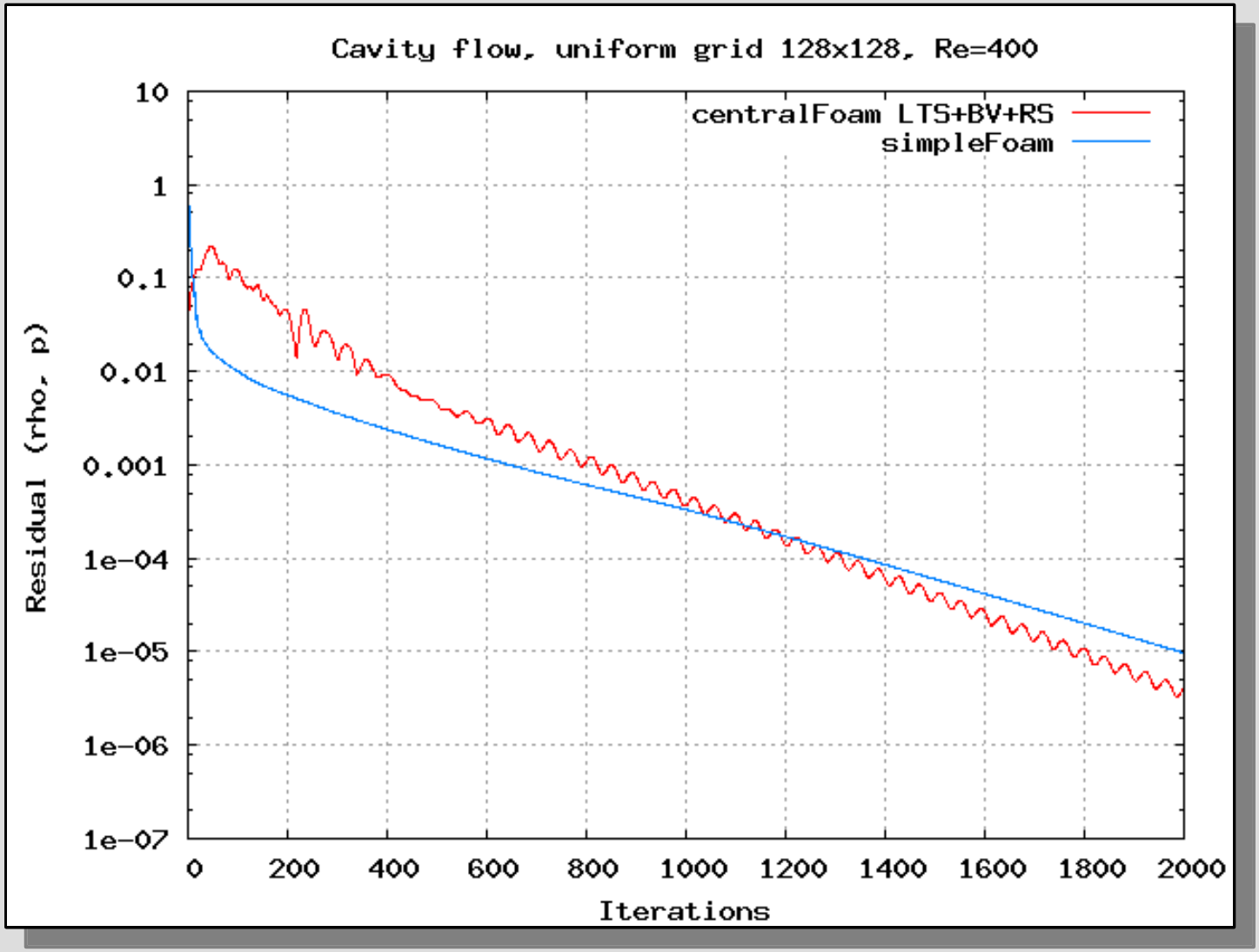
# Convergence acceleration

- Bulk viscosity damping:
  - Is mostly effective for inviscid internal flows (e.g. channel bump).
  - Computational overhead is minimal (one additional term to the momentum equation).
- Residual smoothing:
  - `no`, `implicit` or `implicitExplicit` smoothing option chosen at run time.
  - Allows 2 to 4 times larger time step.
  - Implicit equations approximately solved using few (3-10) iterations of chosen `smoothSolver`.
  - Very effective, with moderate computational overhead (15-30%).

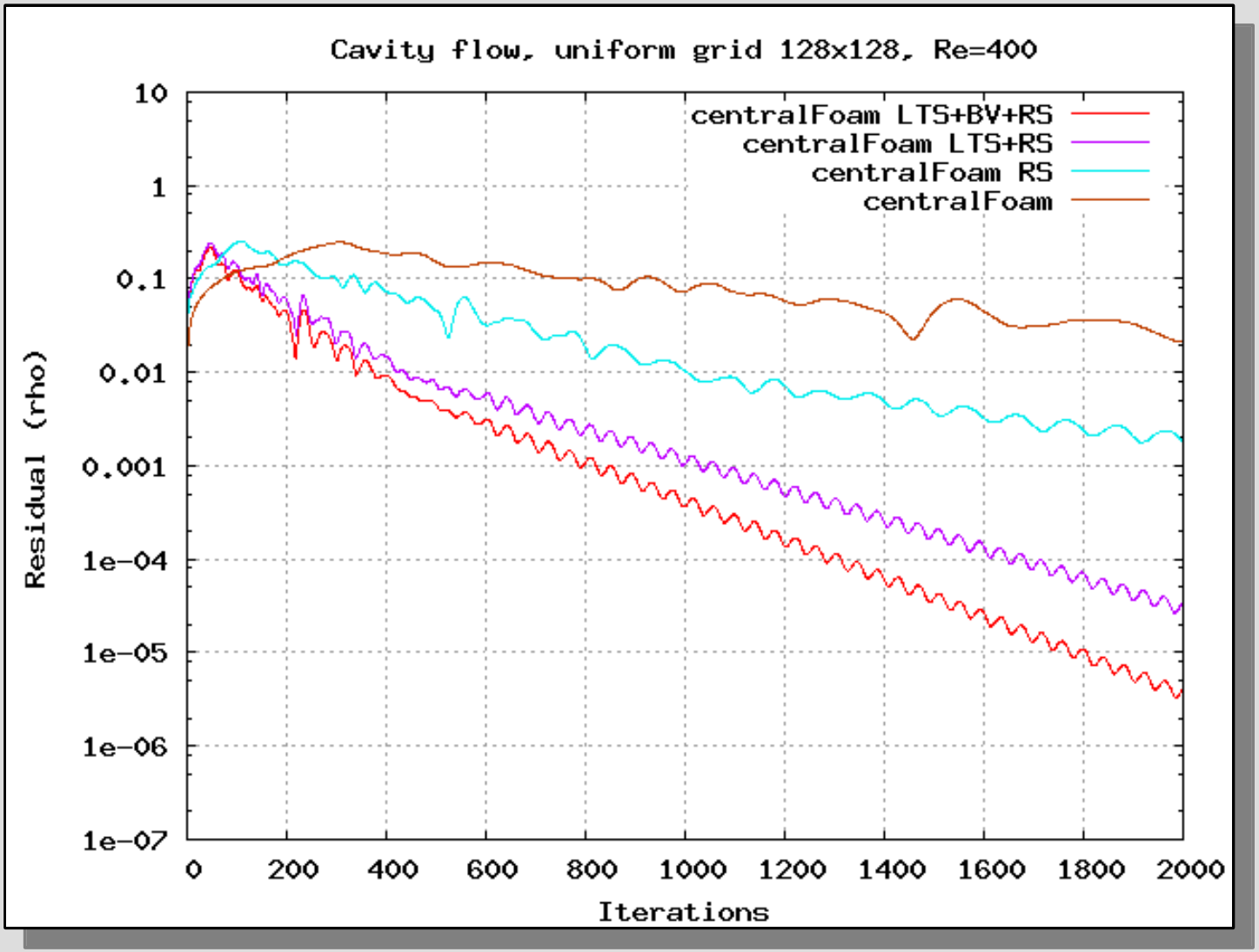
# Convergence acceleration

- As a first example, the effectiveness of the different acceleration techniques is shown for the case of the cavity flow at Re=400.
- A second example is visually showing the effectiveness of the combination of LTS, RS and BV, compared to pure R-K stepping, to reach steady-state for the transonic case of the arc bump at M=0.675.

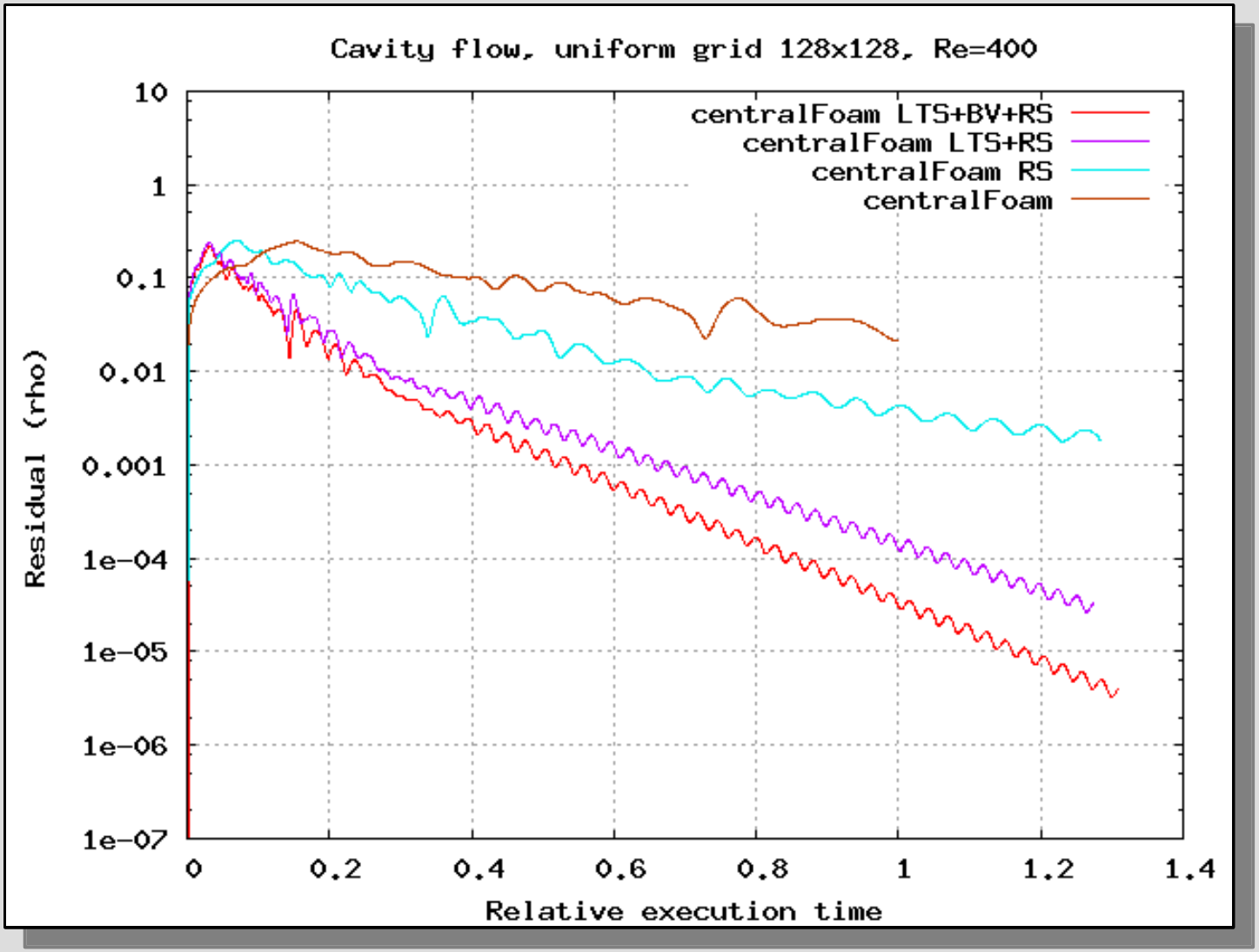
# Convergence acceleration



# Convergence acceleration



# Convergence acceleration



# Convergence acceleration

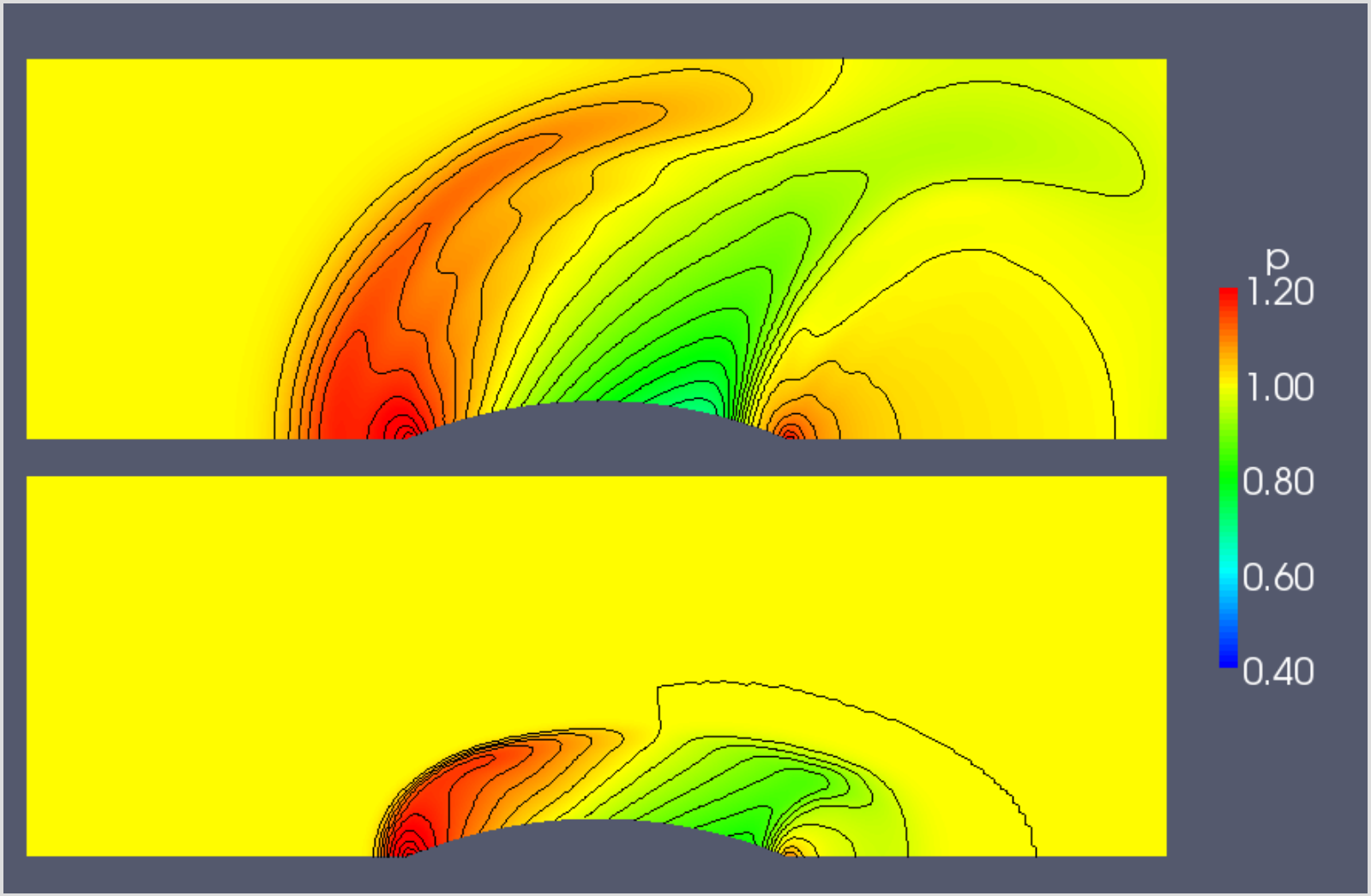
R-K, 2-stages

CFL = 0.4

LTS+BV+RS

R-K, 2-stages

CFL = 0.4




video

# Dual-time stepping

- Problem:
  - Time-preconditioning and primitive variables formulation destroy temporal accuracy.
- Dual-time stepping involves:
  - An outer loop through physical time..
  - ..which wraps an inner loop in pseudo-time.
  - Inner loop is run until convergence to a “steady-state” in pseudo-time.
  - Preconditioning is applied to pseudo-time time derivative only.
  - All steady-state convergence acceleration techniques can be applied.

# Dual-time stepping

- The time-dependent term is typically discretized using backward difference scheme (2nd order, three points scheme).
- The pseudo-time derivative is driven towards zero by the R-K multi-stage algorithm.
- The time dependent term is treated implicitly within each R-K stage, thus the pseudo-time step is determined only by stability condition of the multi-stage scheme.
- Physical time steps is limited only by accuracy, not by CFL.

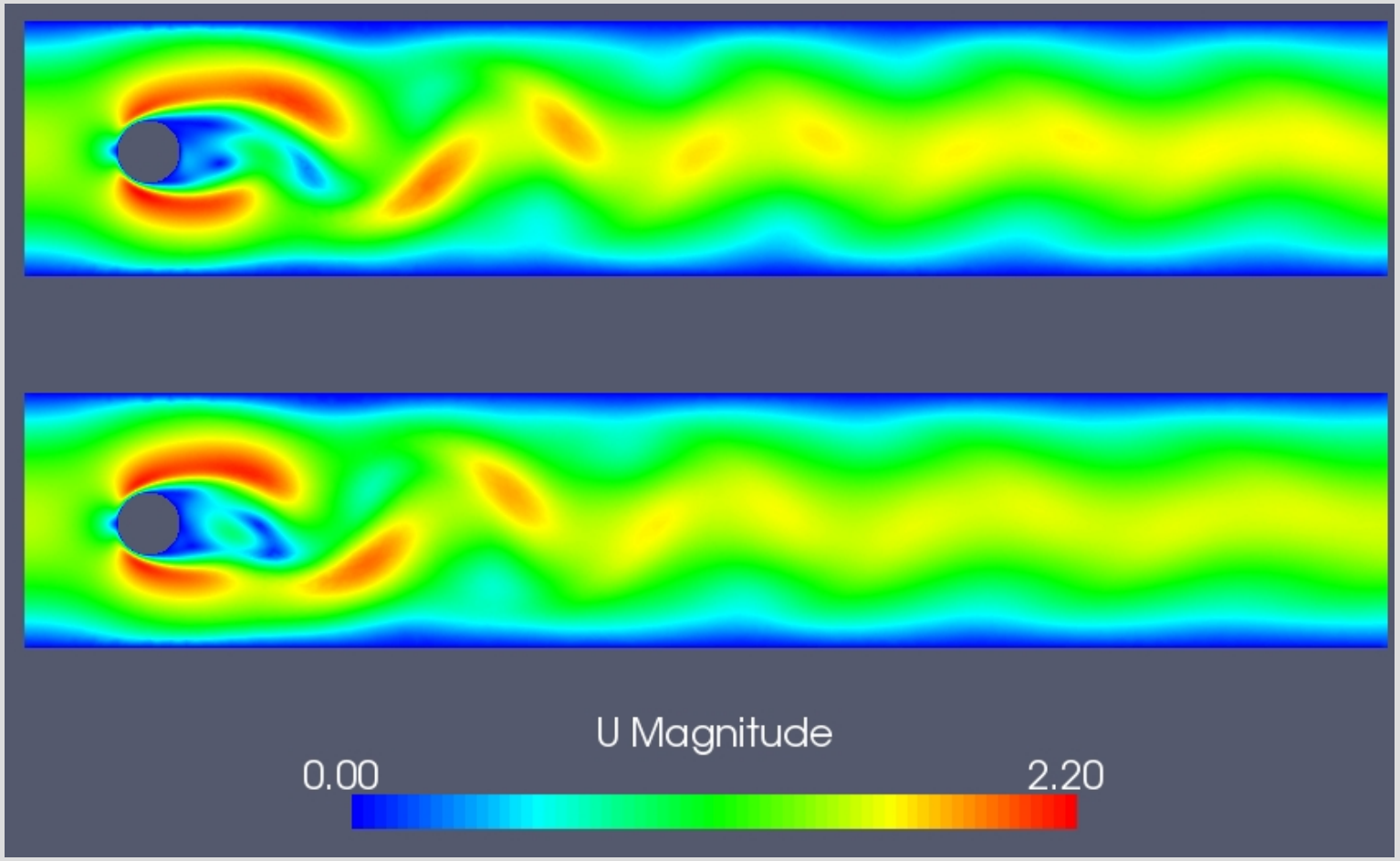
# Dual-time stepping

- An example application is again the 2D laminar flow around a circular cylinder in a channel.
  - At Re=100 the solution is unsteady and soon becomes periodic (von Karman vortex street).
  - The solution is computed on the same 16K cells unstructured grid used for the Re=20 case.
  - Ref: *Schäfer, M. and Turek, S.*

# 2D circular cylinder Re=100

- *icoFoam* uses:
  - `deltaT` = 0.0025, giving $CFL_{MAX}$ ~ 3.5;
  - `nCorrectors` = 2;
  - `nNonOrthogonalCorrectors` = 2.
- *centralFoam* uses:
  - `deltaT` = 0.0100 (NB: physical time step);
  - $10^{-1}$ residual reduction at each time step..
  - ..which requires ~ 40÷50 pseudo-time steps..
  - ..using RS, LTS and BV.

# 2D circular cylinder Re=100

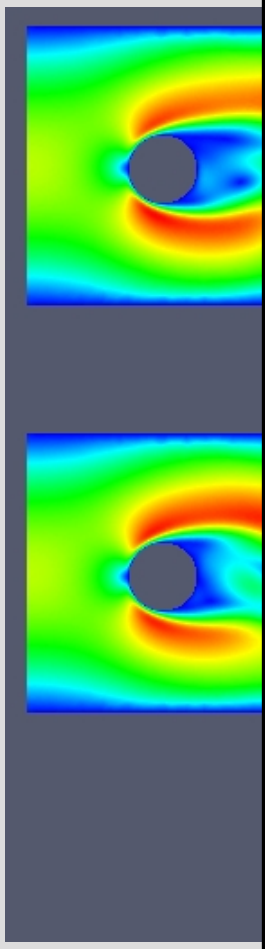


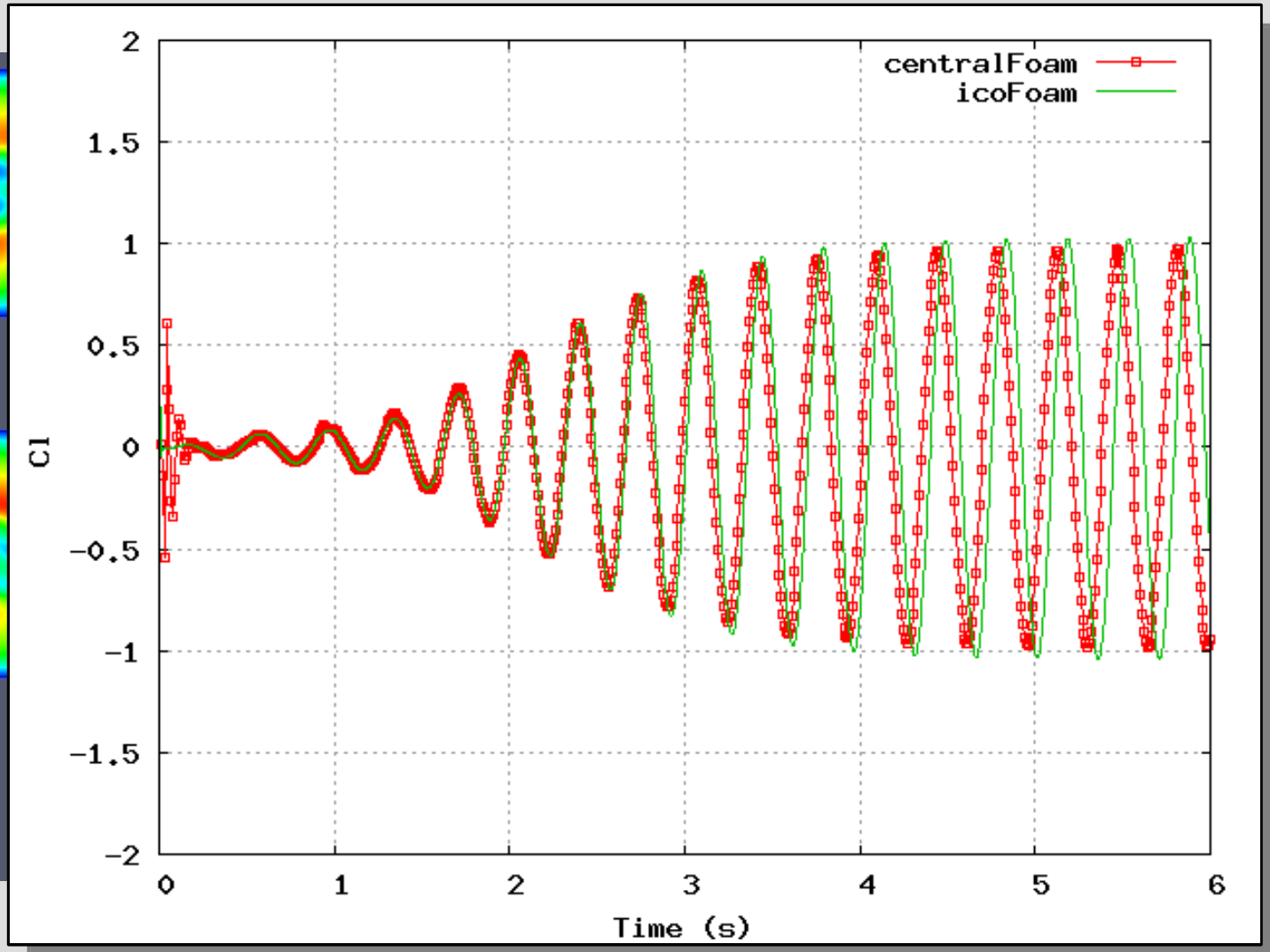

*centralFoam*

*icoFoam*

video

# 2D circular cylinder
# Re=100

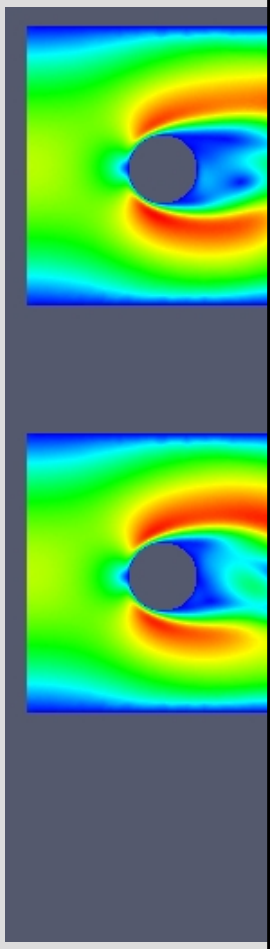


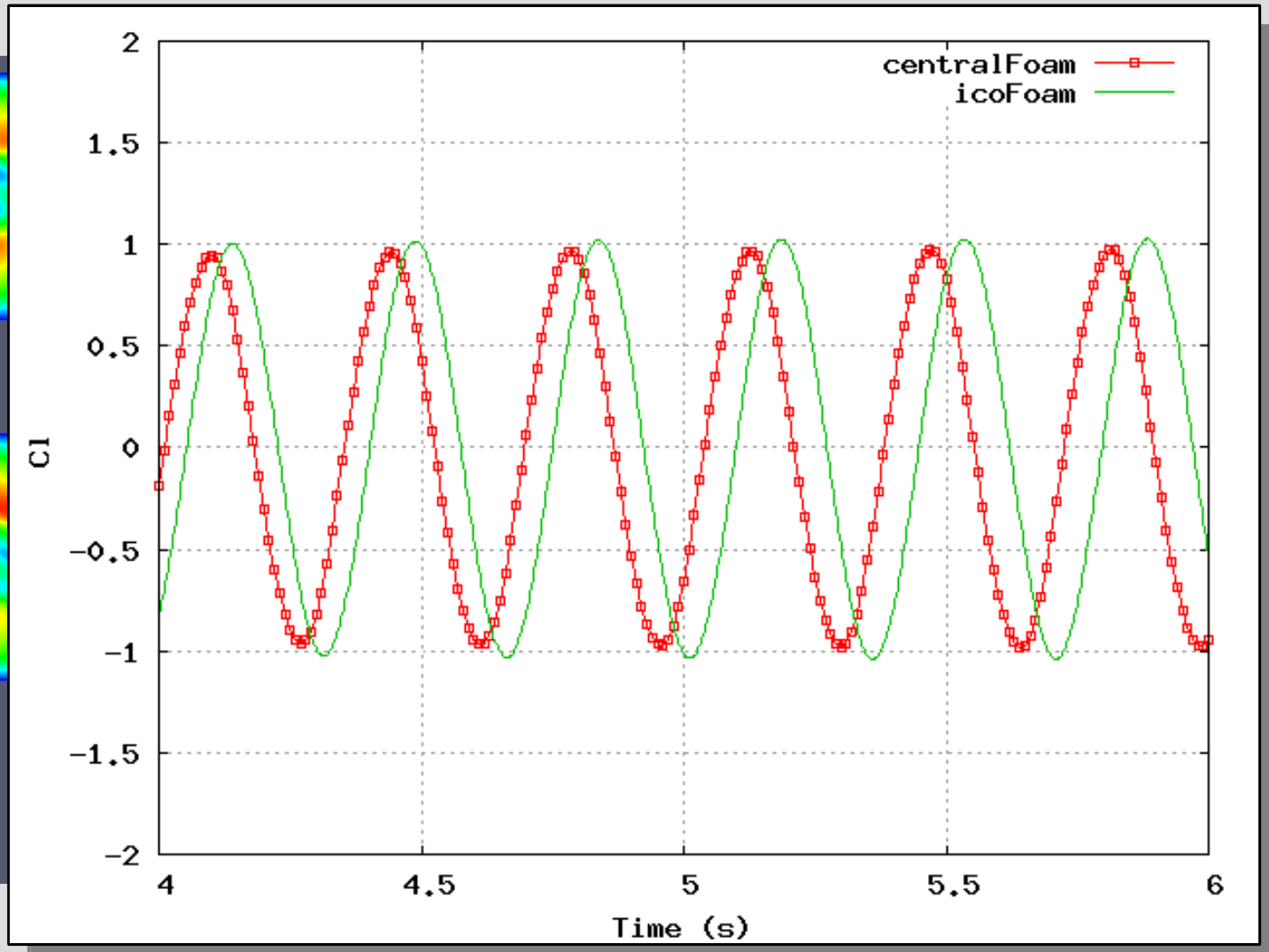



*ralFoam*

*oFoam*

# 2D circular cylinder
# Re=100

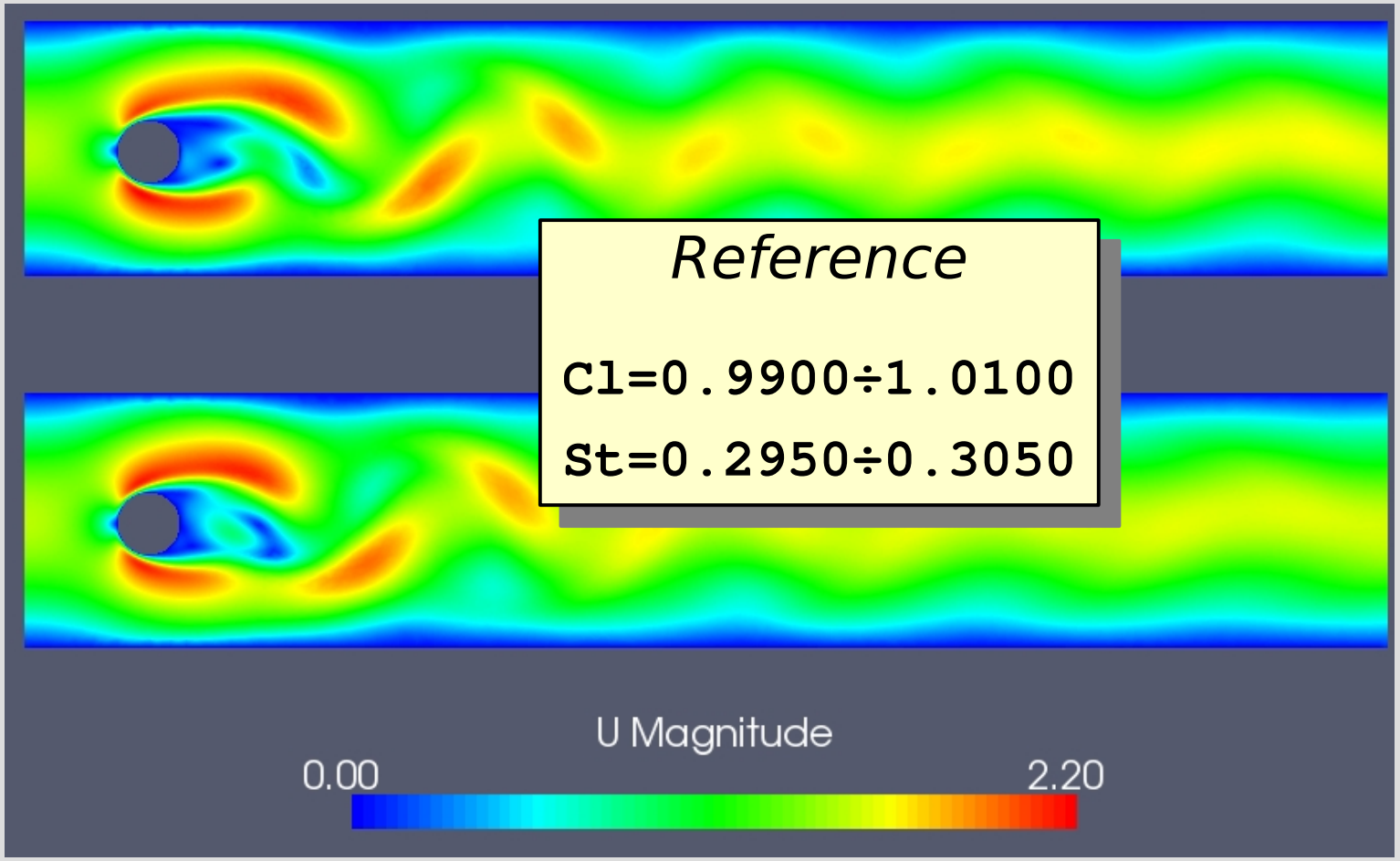





*ralFoam*

*oFoam*

# 2D circular cylinder Re=100




*centralFoam*

$Cl_{Max}$ = 0.975

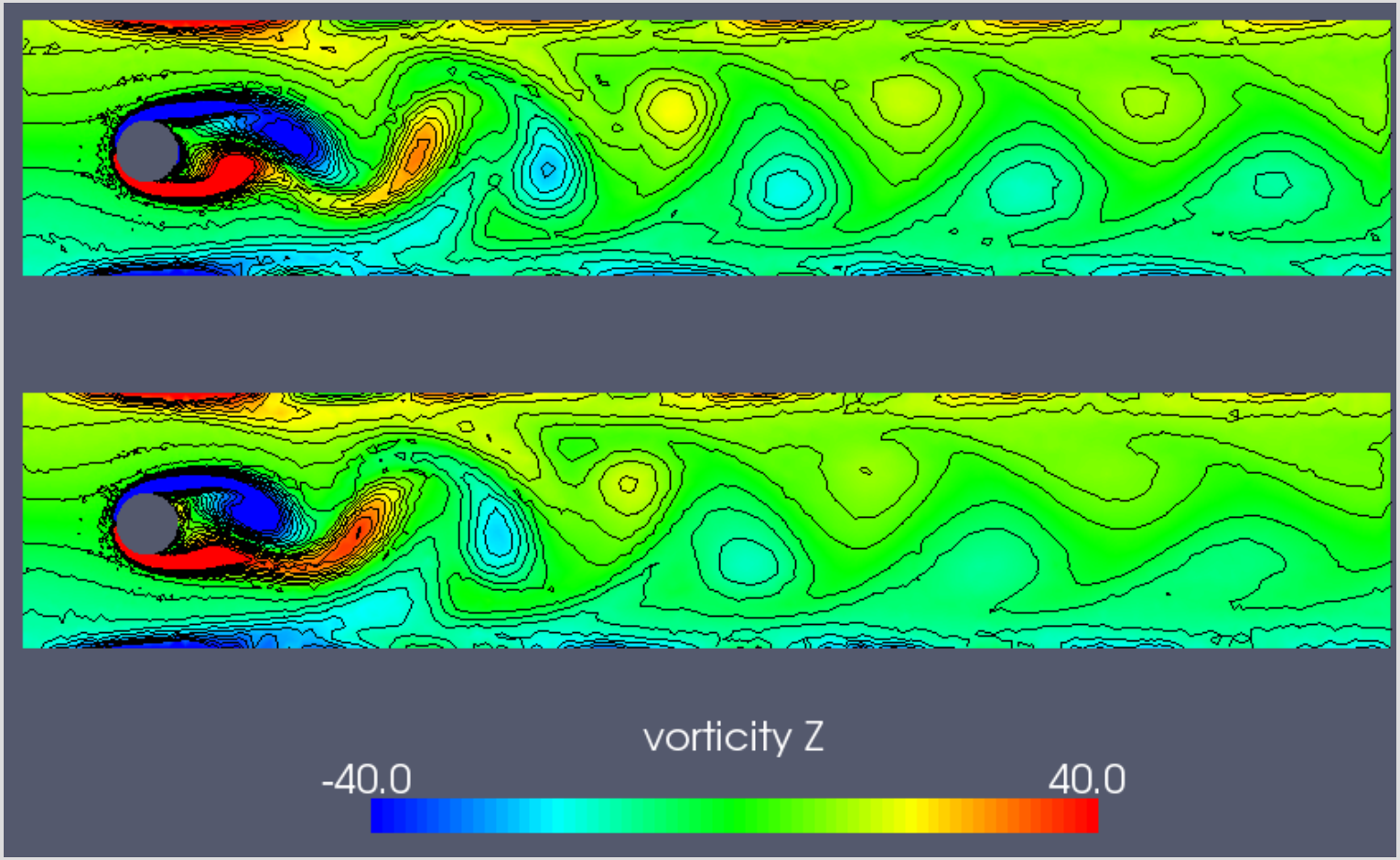
St = 0.294

*icoFoam*

$Cl_{Max}$ = 1.027

St = 0.288

# 2D circular cylinder Re=100




*centralFoam*

*icoFoam*

# 2D circular cylinder Re=100

- Execution time for *centralFoam* is much larger (~ 13 times), because the high Strouhal limits the physical `deltaT`.
- Running on a 4-times finer grids (60k), and keeping:
  - *icoFoam* CFL unchanged;
  - *centralFoam* physical `deltaT` unchanged;

  reduces the disadvantage to ~ 3.5 times.

# Conclusion

- The field of application of the density-based compressible flow solver c*entralFoam* has been extended to:
  - low-Mach number and incompressible flows.
  - laminar viscous flows.
- Convergence to steady state has been accelerated applying standard techniques.
- A dual-time scheme has been implemented, for the computation of unsteady flows when time-preconditioning is activated.

# Conclusion

The current single solver could be split into:

***cCentralFoam***

- conservative var. formulation for high-Mach.
- time accurate multistage R-K time stepping.
- steady state convergence acc. with LTS, RS, BV.

***pCentralFoam***

- precond. primitive var. formulation for low-Mach.
- multistage R-K smoothing.
- steady state convergence acc. with LTS, RS, BV.
- dual-time unsteady computation.

***icoCentralFoam***

- same as *pCentralFoam*...
- ...but specialized for constant density flow.

*L.Gasparini*
*Fondmetal Technologies, Italy*

*S. Mallikarjunan*
*Aerospace MS student, Univ. of Illinois*

# Further development of the compressible flow solver *centralFoam*

***...is still ongoing***

***Thank you!***