Oral presentation | Numerical methods
Numerical methods-V
Wed. Jul 17, 2024 4:30 PM - 6:30 PM Room A

[9-A-04] A novel pressure-based solver for subsonic compressible flows

*Jerome Jansen[1], Stephane Glockner[1], Arnaud Erriguible[1] （1. Institut de mécanique et d'ingénierie de Bordeaux）

Keywords: Compressible, Pressure correction method, Low Mach, Manufactured solution

A novel pressure-based solver for subsonic compressible flows

J. Jansen[1,2], S. Glockner[1,2], D. Sharma[1,2], A. Erriguible[1,2]

Wednesday July 17, 2024

ICCFD12 Kobe, Session Numerical methods

[1] Univ. Bordeaux, CNRS, Bordeaux INP, I2M, UMR 5295, F-33400, Talence, France
[2] Arts et Metiers Institute of Technology, CNRS, Bordeaux INP, Hesam Universite, I2M, UMR 5295, F-33400 Talence, France

Table of contents

Introduction

Two main approaches for solving subsonic compressible flows

Pressure-based methods:

- Come from incompressible community (Chorin, Patankar, ...)
- **Solving** p from Momentum Cons. + Mass Cons. and then **compute** ρ **with EoS**
- Extended to all "speed flows" with low and high Mach approches

Density-based methods:

- Come from compressible (supersonic) community (Lax, Godunov, ...)
- **Solving** ρ from Mass Cons. and then **compute** p **with EoS**
- Extended to low Mach with approches to overcome the singular limit of $c \to \infty$

Pressure-based methods can be:

- **coupled with an iterative process** (SIMPLE/PISO, Wall *et al.*[1], Cang and Wang[2])
- **time-splitted or prediction/correction approach** (Chorin, Goda methods)

Motivation of the work: Starting from Goda incompressible **incremental pressure correction method**[3], propose an extended algorithm for compressible subsonic flows with full 2nd order time accuracy

Warning: Conservation equations are written in primitive form $\to$ we only consider smooth field variations (no shocks)

[1] C. Walls, C.D. Pierce, P. Moin, JCP, 2002

[2] C. Cang, P.Y. Wang, CF, 2023

[3] K. Goda, JCP, 30(1):76-95, 1979

Equations of compressible flow in primitive form

Modelling over time t and along space $\boldsymbol{x}$ a **monophasic** fluid of density ρ, pressure p, temperature T and velocity $\boldsymbol{v}$ flowing with respect of **the conservations of** :

- Mass

$$\frac{\partial \rho}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} \rho + \rho \boldsymbol{\nabla} \cdot \boldsymbol{v} = 0$$

- Momentum

$$\rho \left(\frac{\partial \boldsymbol{v}}{\partial t} + (\boldsymbol{v} \cdot \nabla) \boldsymbol{v} \right) = -\boldsymbol{\nabla} p + \boldsymbol{\nabla} \cdot (\mu \dot{\gamma}) - \frac{2}{3} \boldsymbol{\nabla} (\mu \boldsymbol{\nabla} \cdot \boldsymbol{v}) + \rho \boldsymbol{g}$$

- Energy, expressed in variable (c_p, T) or (c_v, T)

$$\rho c_p \left(\frac{\partial T}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} T \right) = T \beta_p \left(\frac{\partial p}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} p \right) + \boldsymbol{\nabla} \cdot (\lambda \boldsymbol{\nabla} T) + \Phi_d(\boldsymbol{v})$$

Closure of the model with an Equation of State (EoS) for all thermophysical properties :

$$p = EoS(\rho, T), \chi_T = EoS(p, T), c_p = EoS(p, T), \ \ldots$$

Notations are as follows :

$$\beta_p = -\frac{1}{\rho} \left. \frac{\partial \rho}{\partial T} \right|_p , \ \chi_T = \frac{1}{\rho} \left. \frac{\partial \rho}{\partial p} \right|_T , \ \dot{\gamma} = \boldsymbol{\nabla} \boldsymbol{v} + \boldsymbol{\nabla} \boldsymbol{v}^T , \ \Phi_d = -\frac{2\mu}{3} (\boldsymbol{\nabla} \cdot \boldsymbol{v})^2 + \frac{\mu}{2} \dot{\gamma} : \dot{\gamma}$$

Derivation of an equation for pressure in primitive variable

By applying the material derivative on $p = f(T, \rho)$:

$$\frac{\mathrm{D}p}{\mathrm{D}t} = \left.\frac{\partial p}{\partial \rho}\right|_T \frac{\mathrm{D}\rho}{\mathrm{D}t} + \left.\frac{\partial p}{\partial T}\right|_\rho \frac{\mathrm{D}T}{\mathrm{D}t} = \frac{1}{\rho \chi_T}\frac{\mathrm{D}\rho}{\mathrm{D}t} + \frac{\beta_p}{\chi_T}\frac{\mathrm{D}T}{\mathrm{D}t}$$

Using mass and energy conservation, we get a pressure equation as

$$\frac{\mathrm{D}p}{\mathrm{D}t} = -\rho c^2 \boldsymbol{\nabla} \cdot \boldsymbol{v} + \frac{\beta_p c^2}{c_p}\left(\boldsymbol{\nabla} \cdot (\lambda \boldsymbol{\nabla} T) + \Phi_d\right)$$

This equation describing the evolution of pressure will be used to derived an **implicit pressure equation** within our method.

Remarks

- Pressure evolution equation can be see as another formulation of the energy conservation principle
- No restriction on Mach number, temperature gradient or fluid type are made
- In incompressible flow limit ($c \to \infty$ and $\beta_p = 0$), derived pressure equation reduced to $\boldsymbol{\nabla} \cdot \boldsymbol{v} = 0$
- Several authors have already developed and used this equation in **very similiar forms**:
  Kwatra *et al.*, JCP, 228(11), 2009 / Urbano *et al.*, JCP, 456:111034, 2022
  Toutant *et al.*, Physics Letters A, 381(44), 2017)

Derivation of the discretized pressure increment equation

After a prediction of the velocity field $\boldsymbol{v}^*$ computed using $\boldsymbol{\nabla} p^n$, the classical Goda pressure increment $\varphi = p^{n+1} - p^n$ equation reads

$$\boldsymbol{v}^{n+1} - \boldsymbol{v}^* = -\frac{\Delta t}{a\rho^\dagger}\boldsymbol{\nabla}\varphi\,,$$

with a time coefficient and $\rho^\dagger$ a time extrapolation.
We approximate $\boldsymbol{\nabla}\cdot\boldsymbol{v}^{n+1}$ ($\neq 0$ for compressible flows) using the following discretized pressure equation:

$$\frac{ap^{n+1} + bp^n + cp^{n-1}}{\Delta t} + \boldsymbol{v}^\dagger\cdot\boldsymbol{\nabla} p^\dagger = -(\rho c^2)^\dagger\boldsymbol{\nabla}\cdot\boldsymbol{v}^{n+1}\left(\frac{\beta_p c^2}{c_p}\right)^\dagger(\boldsymbol{\nabla}\cdot(\lambda\boldsymbol{\nabla} T) + \Phi_d)^\dagger$$

By rearranging the terms and by expressing φ, the divergence of the pressure increment equation for compressible flows reads

$$\frac{a}{(\rho c^2)^\dagger\Delta t}\varphi - \boldsymbol{\nabla}\cdot\left(\frac{\Delta t}{a\rho^\dagger}\boldsymbol{\nabla}\varphi\right) = -\boldsymbol{\nabla}\cdot\boldsymbol{v}^* + \left(\frac{\dot{S}_\varphi}{\rho c^2}\right)^\dagger$$

Remarks

- Discretized pressure increment equation is of elliptic nature (Helmholtz equation) with variable coefficients
- In incompressible flow limit ($c\to\infty$ and $\beta_p = 0$), it reduce to $\frac{\Delta t}{a\rho^\dagger}\Delta\varphi = \boldsymbol{\nabla}\cdot\boldsymbol{v}^*$
- Using the Chorin projection method, the same equation arise for the pressure variable (Kwatra *et al.*, JCP, 228(11), 2009 / Urbano *et al.*, JCP, 456:111034, 2022)

The algorithm of IPCMSF with 2nd order time discretization

1. Estimated density $\rho^\dagger$ computation:

$$\boxed{\rho^\dagger = 2\rho^n - \rho^{n-1}} \quad \text{or} \quad \frac{\partial \rho^\dagger}{\partial t} + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{v}) = 0$$

2. Predicted velocity $\boldsymbol{v}^*$ resolution[4] using $\boldsymbol{\nabla} p^n$ + BCs:

$$\rho^\dagger \left(\frac{a\boldsymbol{v}^* + b\boldsymbol{v}^n + c\boldsymbol{v}^{n-1}}{\Delta t} + \boldsymbol{\nabla} \cdot (\boldsymbol{v}^\dagger \otimes \boldsymbol{v}^*) - \boldsymbol{v}^* \boldsymbol{\nabla} \cdot \boldsymbol{v}^\dagger \right) = \boxed{-\boldsymbol{\nabla} p^n} + \boldsymbol{\nabla} \cdot (\mu^\dagger \dot{\boldsymbol{\gamma}}^*) - \boldsymbol{\nabla} \left(\frac{2\mu^\dagger}{3} \boldsymbol{\nabla} \cdot \boldsymbol{v}^* \right) + \rho^\dagger \boldsymbol{g}$$

3. Pressure increment φ **implicit** resolution + BCs:

$$\frac{a}{(\rho c^2)^\dagger \Delta t} \varphi - \boldsymbol{\nabla} \cdot \left(\frac{\rho^\dagger}{a \Delta t} \boldsymbol{\nabla} \varphi \right) = -\boldsymbol{\nabla} \cdot \boldsymbol{v}^* + \left(\frac{\dot{S}_\varphi}{\rho c^2} \right)^\dagger$$

4. Pressure update and velocity correction:

$$p^{n+1} = p^n + \varphi, \ \text{and} \ \boldsymbol{v}^{n+1} = \boldsymbol{v}^* - \frac{\rho^\dagger}{\Delta t} \boldsymbol{\nabla} \varphi$$

5. Temperature T^{n+1} resolution[2] + BCs:

$$\rho^\dagger c_p^\dagger \left(\frac{aT^{n+1} + bT^n + cT^{n-1}}{\Delta t} + (\boldsymbol{\nabla} \cdot (\boldsymbol{v} T) - T \boldsymbol{\nabla} \cdot \boldsymbol{v})^{n+1} \right) - \boldsymbol{\nabla} \cdot (\lambda^\dagger \boldsymbol{\nabla} T^{n+1})$$
$$- T^{n+1} \beta_p^\dagger \left(\frac{ap^{n+1} + bp^n + cp^{n-1}}{\Delta t} + \boldsymbol{v}^{n+1} \cdot \boldsymbol{\nabla} p^{n+1} \right) = \Phi_d^{n+1}$$

6. Thermophysical properties updates with EoS: $\rho^{n+1} = EoS(T^{n+1}, p^{n+1})$, $\beta_p^{n+1} = \ldots$

[4] Could be explicit or implicit

The algorithm of IPCMSF with 2nd order time discretization

Steps enumeration:

1. Estimated density $\rho^\dagger$ computation
2. Predicted velocity $\boldsymbol{v}^*$ resolution using p^n
3. Pressure increment φ **implicit** resolution
4. Pressure update and velocity correction
5. Temperature T^{n+1} resolution
6. Thermophysical properties updates with EoS

Boundary conditions:

Step 1. If mass conservation solved, Dirichlet or Neumann:
$\rho_\Gamma = f(\boldsymbol{x})$ or $\partial\rho/\partial n_\Gamma = g(\boldsymbol{x})$

Step 2. Dirichlet or Neumann:
$\boldsymbol{u}_\Gamma = \boldsymbol{f}_u(\boldsymbol{x})$ or $\partial u/\partial n_\Gamma = \boldsymbol{g}_u(\boldsymbol{x})$

Step 3. Homogeneous neumann:
$\partial\varphi/\partial n_\Gamma = 0$

Step 5. Dirichlet or Neumann:
$T_\Gamma = f_T(\boldsymbol{x})$ or $\partial T/\partial n_\Gamma = g_T(\boldsymbol{x})$

Remarks

- Implicit or explicit resolution of steps (2.) and (5.) are possible and case-dependent
- **Large** Δt availables thanks to step (3.)
- The computational cost is mainly at step (3.)
- **The Mass conservation equation can never be explicitly solved**

Numerical Methods available in Notus for IPCMSF

CFD code notus

- Open Source project started in 2015
- Based of Finite Volume Method on staggered grid with immersed boundary method
- **Massively parallel** written in modern Fortran (MPI/OpenMP)
- Numerical experimentations
- Multiphysics

notus
Computational Fluid Dynamics

- Time discretization: 2nd order Backward Differentiation Formula (BDF2)
- Diffusion and advection terms discretizations: implicit `o2_centered`
- Treatment of pressure advection term $\boldsymbol{v} \cdot \nabla p$: explicit `upwind_o2`
- **Space- and time-dependent** thermophysical properties:
  $\rightarrow$ 2nd order space extrapolation of face fields $\rho(\boldsymbol{x})$, $\mu(\boldsymbol{x})$ and $\lambda(\boldsymbol{x})$ at boundaries
  $\rightarrow$ 2nd order time extrapolation of cell centered fields $\rho^\dagger$, $\chi_T^\dagger$, $\beta_p^\dagger$, ...
- Solver momentum, pressure and temperature HYPRE_GMRES with left_jacobi and PFMG preconditioner (tol 10^{-12})
- If mass conservation solved: explicit RK-nssp3_o2 + WENO5

Verifications againt analytical solutions

Test-case	Geometry	Fluid	EoS	Physic
Isentropic injection	"0D"	Air	Perfect Gas	Fluid compression
Linear acoustic pulse	1D	Air	Perfect Gas	Acoustic propagation
Manufactured solution isotherm	2D	Air	Perfect Gas	None
Manufactured solution anisotherm $\mathrm{Ma} = 0.6$	2D	Air	Perfect Gas	None
Manufactured solution anisotherm $\mathrm{Ma} = 0.002$	2D	Air	Perfect Gas	None

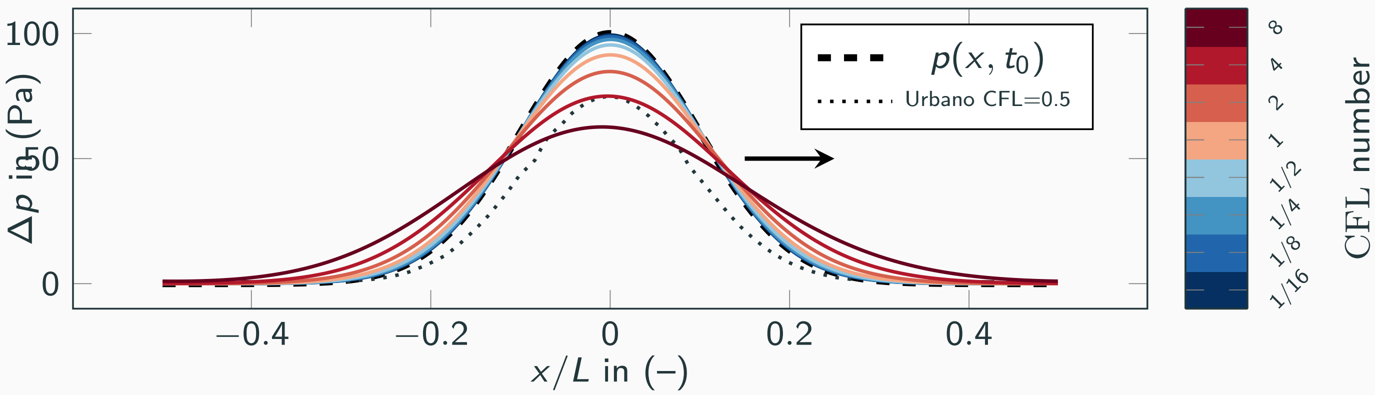
Objectives:

- Explore consistency of the proposed IPCMSF algorithm
- Explore time order accuracy of IPCMSF algorithm for $(\boldsymbol{u}, T, p)$
- Check bug-free implementation within Notus with increasing complexity of cases

Linear acoustic pulse propagation

From the resolution of the d'Alembert equation, analytical solutions are available:

$$p(x,t) = p_0 + \Delta p_0 \exp\left(-\frac{(x - c_0 t)^2}{2\Sigma^2}\right)$$



Δt in s	$\|\|\varepsilon_v\|\|_{L_2}$	order	$\|\|\varepsilon_v\|\|_{L_\infty}$	order	$\|\|\varepsilon_p\|\|_{L_2}$	order	$\|\|\varepsilon_p\|\|_{L_\infty}$	order	$\|\|\varepsilon_\rho\|\|_{L_2}$	order	$\|\|\varepsilon_\rho\|\|_{L_\infty}$	order
1.60×10^{-4}	4.568×10^{-2}	n/a	9.698×10^{-2}	n/a	1.842×10^{1}	n/a	3.911×10^{1}	n/a	1.528×10^{-4}	n/a	3.245×10^{-4}	n/a
8.00×10^{-5}	2.304×10^{-2}	0.987	5.057×10^{-2}	0.939	9.293	0.987	2.039×10^{1}	0.940	7.710×10^{-5}	0.987	1.692×10^{-4}	0.940
4.00×10^{-5}	7.871×10^{-3}	1.550	1.846×10^{-2}	1.454	3.176	1.549	7.451	1.452	2.635×10^{-5}	1.549	6.182×10^{-5}	1.452
2.00×10^{-5}	1.986×10^{-3}	1.987	4.873×10^{-3}	1.922	8.017×10^{-1}	1.986	1.968	1.920	6.651×10^{-6}	1.986	1.633×10^{-5}	1.920
1.00×10^{-5}	3.272×10^{-4}	2.601	7.510×10^{-4}	2.698	1.321×10^{-1}	2.601	3.036×10^{-1}	2.697	1.096×10^{-6}	2.601	2.519×10^{-6}	2.697

Temporal order of accuracy (BDF2, o2_centered),
First time step $\Delta t = 1.6 \times 10^{-4}$ s $\sim \mathrm{CFL}_{\mathrm{ac}} = 2.84 \times 10^{1}$.
Mesh size 512×8, $t_f = 2.88 \times 10^{-3}$ s.

Manufactured solutions for subsonic compressible flow

Proposed compressible MS[4] on square domain $\Omega = [0,1] \times [0,1]$ with:

- Dirichlet BCs for velocity, temperature (and density) fields
- Homogeneous neumann BC for pressure increment
- Perfect gas EoS of the fluid $\rho = p \big/ RT$:
- **Unpleasant** properties solution, *e.g.* $\partial p / \partial n_\Gamma \neq 0$ and $\boldsymbol{\nabla} \cdot \boldsymbol{u} \neq 0$

$$p = p_0 + p_1 \sin^2(\pi y)\cos^2(\pi x)\cos(2\pi f t)\,, \quad T = T_0 + T_1 \sin(\pi y)\cos(\pi x)\cos(2\pi f t)$$
$$u = u_0 \sin^2(\pi x)\sin(2\pi y)\cos(2\pi f t)\,, \quad v = u_0 \sin(2\pi x)\sin^2(\pi y)\cos(2\pi f t)$$

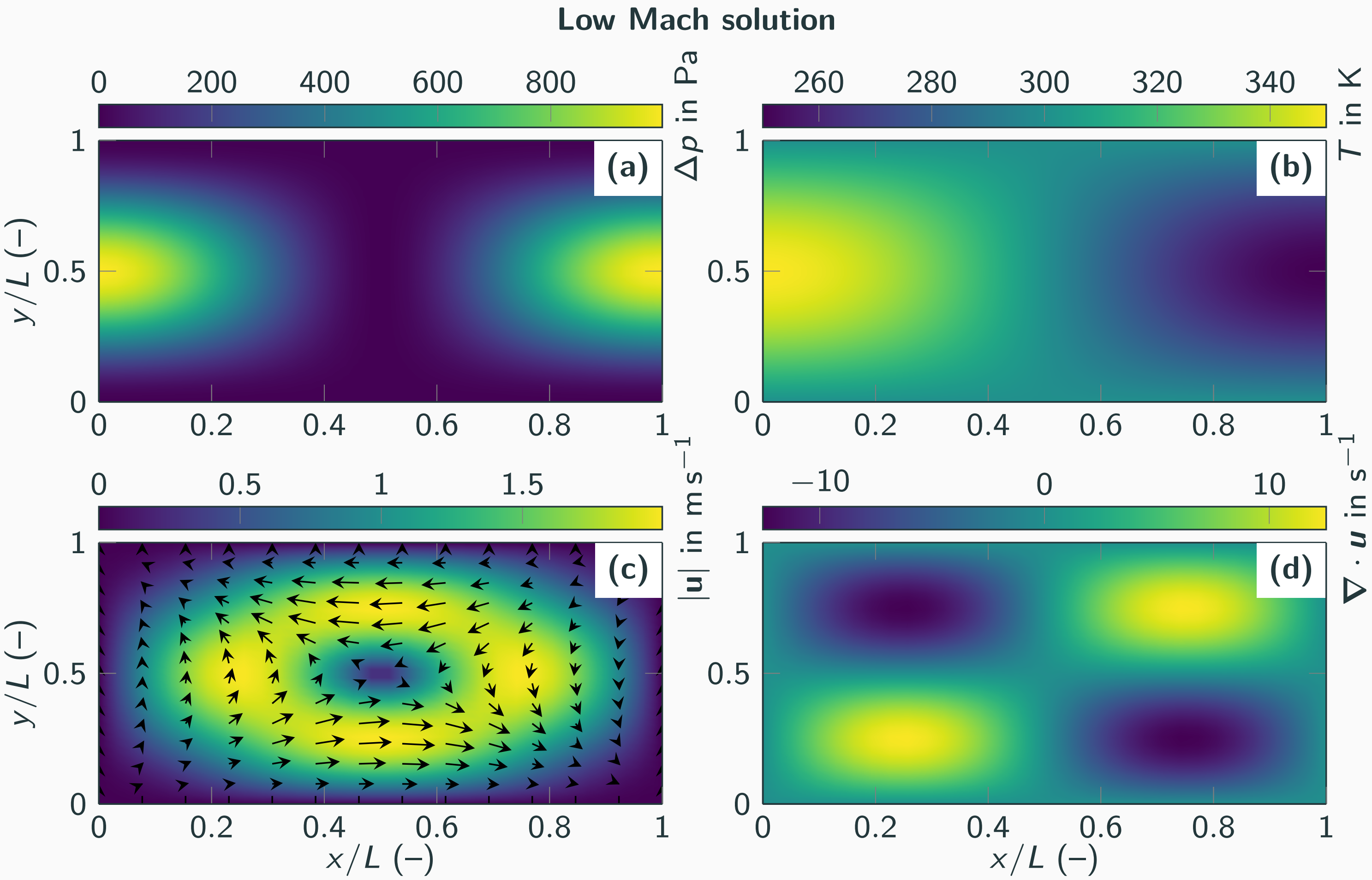
Two differents MS

"High Mach" case:	Low Mach case:
$u_0 = 200$ m/s	$u_0 = 2$ m/s
$p_1 = 2 \times 10^3$ Pa	$p_1 = 2 \times 10^3$ Pa
$T_1 = 40$ K	$T_1 = 40$ K
$\mathrm{Re}_0 = 1.26 \times 10^7$	$\mathrm{Re}_0 = 1.26 \times 10^5$
$\mathrm{Ma}_0 = 5.8 \times 10^{-1}$	$\mathrm{Ma}_0 = 5.8 \times 10^{-3}$

Objectives: Explore time order accuracy of IPCMSF on a **"full" subsonic compressible test-case** (but unphysical)

[4] extended from the incompressible one proposed by Guermond *et al.* 2006

Manufactured solutions for subsonic compressible flow

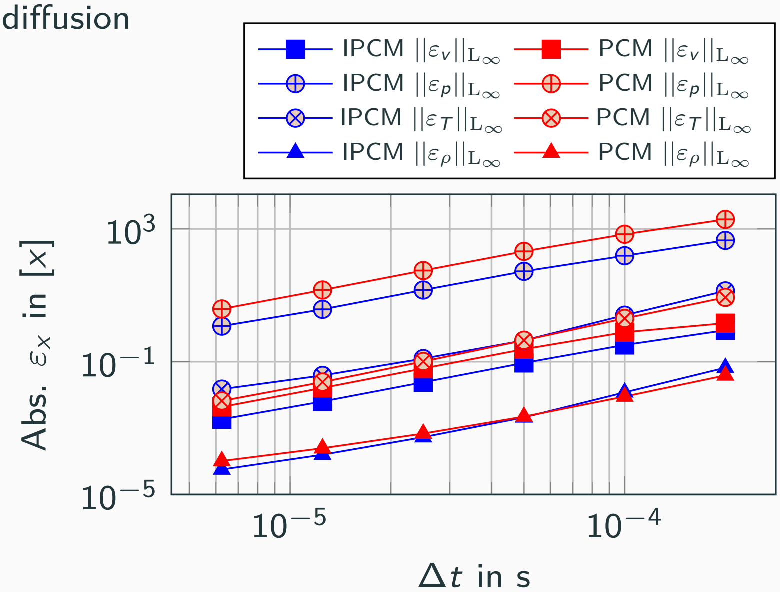


Low Mach solution temporal convergence

Comparison of temporal order accuracy of the anisothermal low Mach manufactured solution for **IPCMSF and PCMSF**[4]
MS parameters: $u_0 = 2$ m/s, $p_1 = 2 \times 10^3$ Pa, $T_1 = 40$ K, $\mathrm{Re}_0 = 1.26 \times 10^5$ - $\mathrm{Ma}_0 = 5.8 \times 10^{-3}$
First time step $\Delta t = 2 \times 10^{-4}$ s $\simeq \mathrm{CFL_{ac}} = 1.78 \times 10^1$.
Mesh size 256^2 and $t_f = 2 \times 10^{-3}$ s,
implicit o2_centered for advection and diffusion

IPCMSF orders

Δt s	$o(\|\|\varepsilon_v\|\|_\infty)$	$o(\|\|\varepsilon_p\|\|_\infty)$	$o(\|\|\varepsilon_T\|\|_\infty)$	$o(\|\|\varepsilon_\rho\|\|_\infty)$
2.0×10^{-4}	n/a	n/a	n/a	n/a
1.0×10^{-4}	1.472	1.537	2.432	2.465
5.0×10^{-5}	1.798	1.559	2.527	2.503
2.5×10^{-5}	1.934	1.873	1.816	1.980
1.3×10^{-5}	1.921	1.930	1.683	1.754
6.3×10^{-6}	1.784	1.678	1.369	1.481



Principal result:

- Both methods are **2nd order accurate in time**,
- Error magnitudes with IPCMSF are lower, mainly for the pressure field

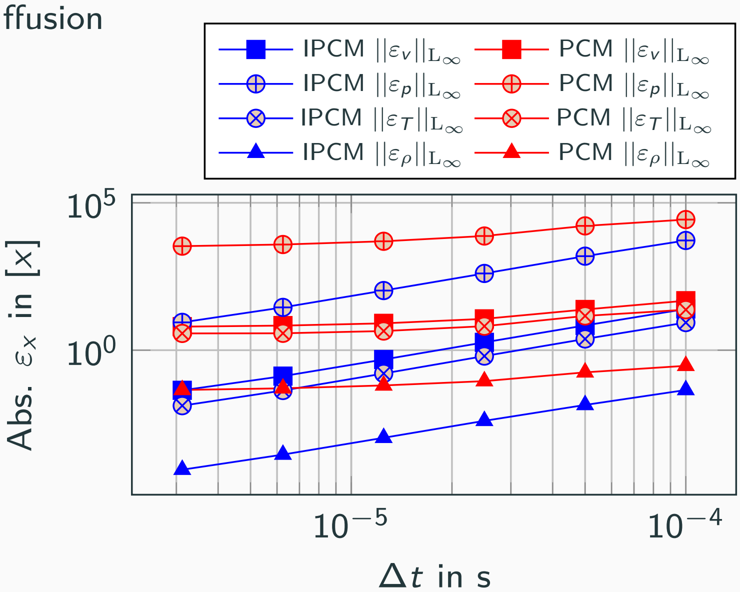
[4] Urbano *et al.*, JCP, 456:111034, 2022

High Mach solution temporal convergence

Comparison of temporal order accuracy of the anisothermal high Mach manufactured solution for **IPCMSF and PCMSF**
Parameters: $u_0 = 200$ m/s, $p_1 = 2 \times 10^3$ Pa, $T_1 = 40$ K, $\mathrm{Re}_0 = 1.26 \times 10^7$ - $\mathrm{Ma}_0 = 5.8 \times 10^{-1}$
First time step $\Delta t = 1 \times 10^{-4}$ s $\simeq \mathrm{CFL}_{\mathrm{ac}} = 8.88$.
Mesh size 256^2 and $t_f = 2 \times 10^{-3}$ s,
implicit o2_centered for advection and diffusion

IPCMSF orders

Δt s	o($\lVert\varepsilon_v\rVert_\infty$)	o($\lVert\varepsilon_p\rVert_\infty$)	o($\lVert\varepsilon_T\rVert_\infty$)	o($\lVert\varepsilon_\rho\rVert_\infty$)
1.0×10^{-4}	n/a	n/a	n/a	n/a
5.0×10^{-5}	1.851	1.770	1.847	1.684
2.5×10^{-5}	1.907	1.947	1.931	1.790
1.3×10^{-5}	1.934	1.929	1.964	1.911
6.3×10^{-6}	1.873	1.897	1.915	1.894
3.1×10^{-6}	1.598	1.689	1.711	1.703



Principal result:

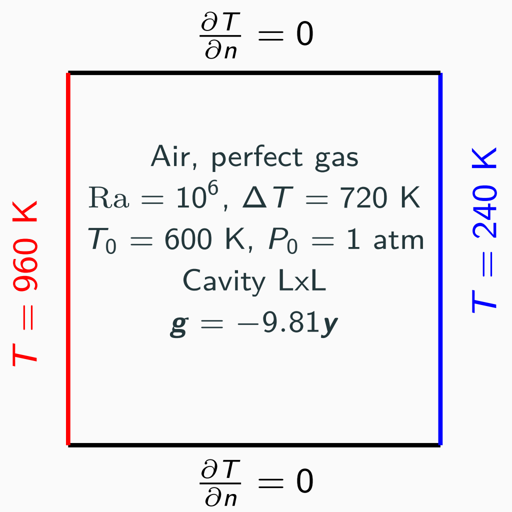
- IPCMSF is **2nd order accurate in time** while PCMSF is 1st order,
- Error magnitudes with IPCMSF are lower

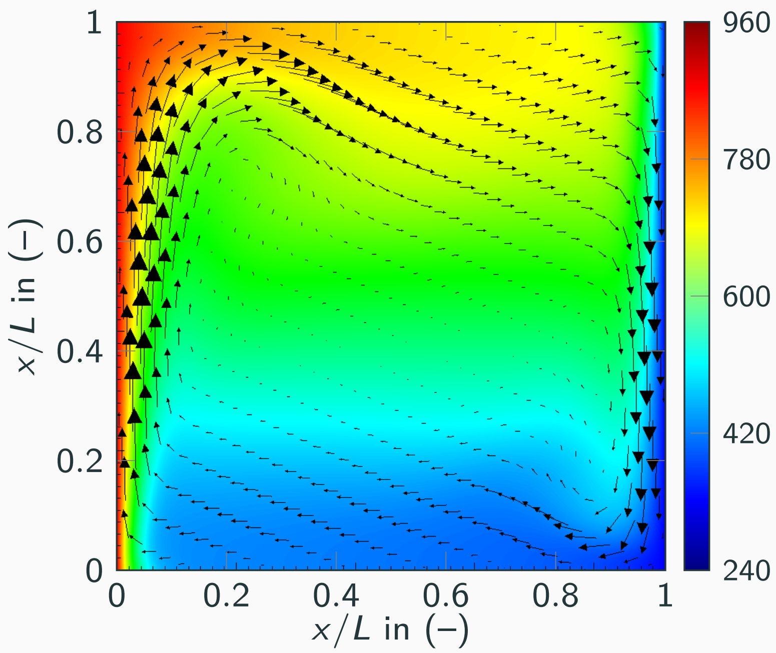
Validations on physical test-cases

Test-case	Geometry	Fluid	EoS	Physic
Miura test-case	1D	Supercritic CO2	NIST refprop	Piston effect
Incomp. convection $\mathrm{Ra} = 10^6$	2D	Air	Perfect Gas	Natural Convection at $\Delta T = 1$K
Comp. convection $\mathrm{Ra} = 10^6$ $\mu = cst$	2D	Air	Perfect Gas	Natural Convection at $\Delta T = 720$K
Comp. convection $\mathrm{Ra} = 10^6$ $\mu(T)$	2D	Air	Perfect Gas	Natural Convection at $\Delta T = 720$K
Comp. convection with IBM $\mathrm{Ra} = 5x10^6$	2D	Air	Perfect Gas	Natural Convection at $\Delta T = 120$K
Heated channel flow $\mathrm{Re} = 65$	2D	Air	Perfect Gas	Heat transfer in a pipe

Steady compressible natural convection $\mathrm{Ra} = 10^6$ with constant viscosity

Sketch and initial state :





Reference values[3]:
Nu=8.860, $\overline{p} = -1.456 \times 10^4$ Pa

Spatial convergence (Euler, o2_centered), $\mathrm{CFL_{ac}}$ =400, t_f=20 s

Mesh	Mass loss	order	Nu_L	order	Nu_R	order	Mean p	order	Mean $\boldsymbol{v}$	order	Mean T	order
64x64	-2.648×10^{-2}	n/a	9.049	n/a	9.079	n/a	-1.794×10^4	n/a	4.971×10^{-2}	n/a	5.645×10^2	n/a
128x128	-6.818×10^{-3}	n/a	8.910	n/a	8.918	n/a	-1.545×10^4	n/a	4.860×10^{-2}	n/a	5.682×10^2	n/a
256x256	-2.061×10^{-3}	2.048	8.871	1.817	8.872	1.840	-1.481×10^4	1.961	4.831×10^{-2}	1.925	5.693×10^2	1.791
512x512	-1.061×10^{-3}	2.250	8.860	1.838	8.860	1.880	-1.467×10^4	2.140	4.824×10^{-2}	2.064	5.696×10^2	1.967
1024x1024	-8.805×10^{-4}	2.466	8.857	1.825	8.857	1.848	-1.464×10^4	2.257	4.823×10^{-2}	2.110	5.696×10^2	1.994
Extrapolation	-8.405×10^{-4}	n/a	8.856	n/a	8.856	n/a	-1.463×10^4	n/a	4.822×10^{-2}	n/a	5.697×10^2	n/a

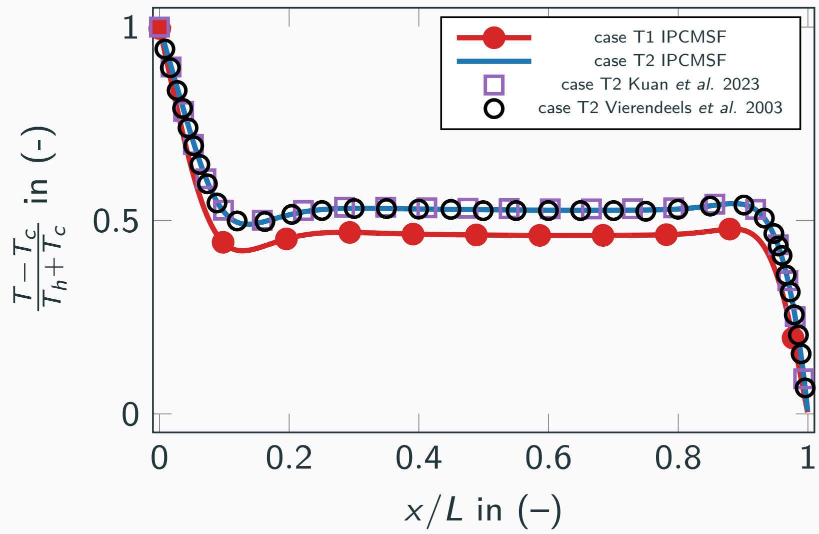
[3] P. Le Quéré *et al.*, ESAIM, 39(3):609-616, 2005

Steady compressible natural convection $\mathrm{Ra} = 10^6$ with variable viscosity

Temperature-dependent dynamic viscosity following Sutherland law:

$$\mu(T) = \mu^* \left(\frac{T}{T^*}\right)^{3/2} \frac{T^* + S}{T + S}$$

Reference values[5]:
Nu=8.687, $\overline{p}$ =−7651.35Pa



Spatial convergence (Euler, o2_centered), $\mathrm{CFL_{ac}}$ =400, t_f=20 s

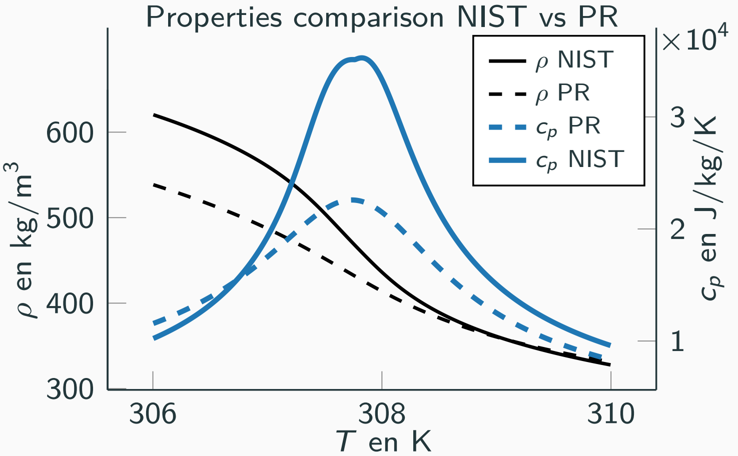
Mesh	Mass loss	order	Nu_L	order	Nu_R	order	Mean p	order	Mean $\boldsymbol{v}$	order	Mean T	order
64x64	-8.728×10^{-2}	n/a	8.628	n/a	8.656	n/a	-1.756×10^4	n/a	5.846×10^{-2}	n/a	6.001×10^2	n/a
128x128	-3.366×10^{-2}	n/a	8.641	n/a	8.653	n/a	-1.141×10^4	n/a	5.662×10^{-2}	n/a	6.055×10^2	n/a
256x256	-7.929×10^{-3}	1.059	8.682	−1.537	8.685	nan	-8.576×10^3	1.116	5.582×10^{-2}	1.200	6.075×10^2	1.422
512x512	-2.249×10^{-3}	2.179	8.684	4.214	8.685	7.218	-7.912×10^3	2.093	5.562×10^{-2}	2.020	6.081×10^2	1.701
1024x1024	-7.304×10^{-4}	1.904	8.685	2.506	8.685	3.018	-7.733×10^3	1.890	5.557×10^{-2}	1.858	6.083×10^2	1.827
Extrapolation	-1.766×10^{-4}	n/a	8.685	n/a	8.685	n/a	-7.666×10^3	n/a	5.555×10^{-2}	n/a	6.083×10^2	n/a

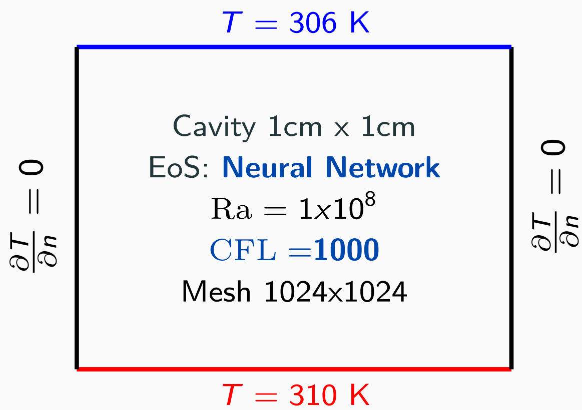
This test-case highlights the **importance of correct face field** material properties computations at boundaries (conductivity here)

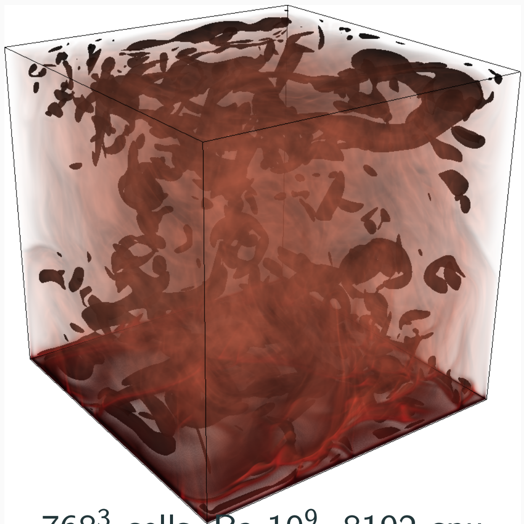
[5] P. Le Quéré *et al.*, ESAIM, 39(3):609-616, 2005

Application

Supercritical CO_2 convection crossing the Widom Line







768^3 cells, Ra 10^9, 8192 cpu
criterion $Q = 4e3$, Temperature volume rendering

Strong scalability test (TGCC rome)
on $1024^3 \sim 1 \times 10^9$ mesh size

cpu	node	time/ite in (s)	Eff.
8192	64	6.199	1.000
16384	128	3.962	0.782
32768	256	3.811	0.407

IPCMSF notus 55×10^3 cells/cpu

Conclusions and futurs works

Conclusions and Futurs works

Conclusions

- Verifications & Validation of the IPCMSF algorithm
- 2nd order accurate in time for all **primitive variables**
- Compatibility of temperature / pressure / density (Kwatra *et al.*, JCP, 228(11), 2009)

Futurs works on IPCMSF

- Extension to open flows configuration with specific φ outflow BC[6]
- Extension to 2nd order Immersed Boundary Method[7]
- Developping a full explicit RK method based on IPCMSF
- Extension to two-phase flow with mass and heat transferts

[6] A. Poux *et al.*, JCP, 230(10):40114027, 2012

[7] de Palma *et al.*, CF, 35(7):693-702, 2006