An introduction to Volume-Of-Fluid simulation of interfaces

Stéphane Zaleski

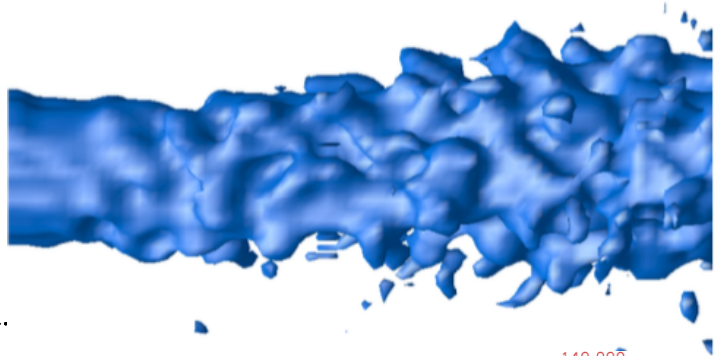
Institut Jean Le Rond ∂'alembert,
CNRS & Université Pierre et Marie Curie -- UPMC – Paris 6

web site http://www.ida.upmc.fr/~zaleski

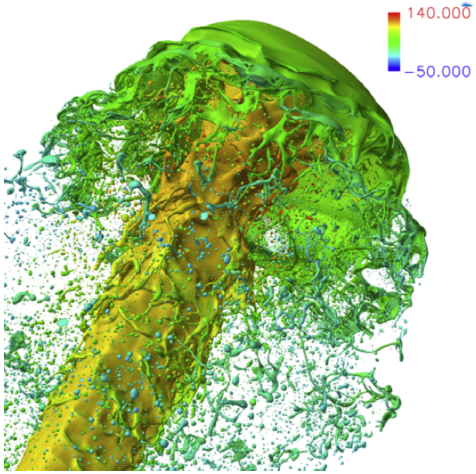
Simulation of atomizing flows progresses at an amazing rate

In 2004 about 5 million grid points

Bianchi Scardovelli Zaleski SAE



+ Berlemont, Hermann, Desjardins, Le Chenadec & Pitsch, Ashgriz, Sirignano ...

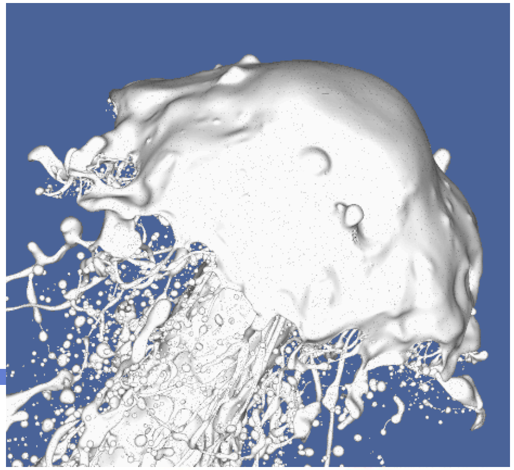


In 2010 about 6 billion grid points. 2 million CPU hours on 5760 cores.

Shinjo & Umemura. IJMF 2010

In 2016 adaptive simulation « equivalent » to 64 billion grid points 10 thousand CPU hours on 1356 cores.

S. Popinet http://basilisk.fr/src/examples/atomisation.c



How to model & simulate such flows ?

- 1 Compute an evolving surface : computational geometry

- Solve

  2 the Navier-Stokes equations with

  3 surface tension,

and

  4 variable viscosity and density

1. Compute surface evolution

it is a kind of computational geometry

Why is it difficult to follow evolving surfaces ?

- geometrical complexity (curved surfaces, how they cut – change topology)

- numerical stability issues of the most obvious methods.

- accuracy issues (high accuracy is needed)

Compute surface evolution:

Two formulations:

1) express surface velocity:

$$V_S = \mathbf{u} \cdot \mathbf{n}.$$

2) Use the characteristic function $\chi = 1$ in phase 1 and $\chi = 0$ in phase 2.

$$\partial_t \chi + \mathbf{u} \cdot \nabla \chi = 0.$$

Three main methods for surface evolution

- Front-Tracking with marker particles.

- Level-Set with smooth marker function F.

- Volume of Fluid (VOF) with discontinuous characteristic function χ .

Methods based on microscopic physics

- Phase field (Cahn Hilliard – Van der Waals equations)

- Lattice-Gas Cellular Automata

- Lattice Boltzmann

Hybrid methods

any combination of two of the above six. Popular mixes:

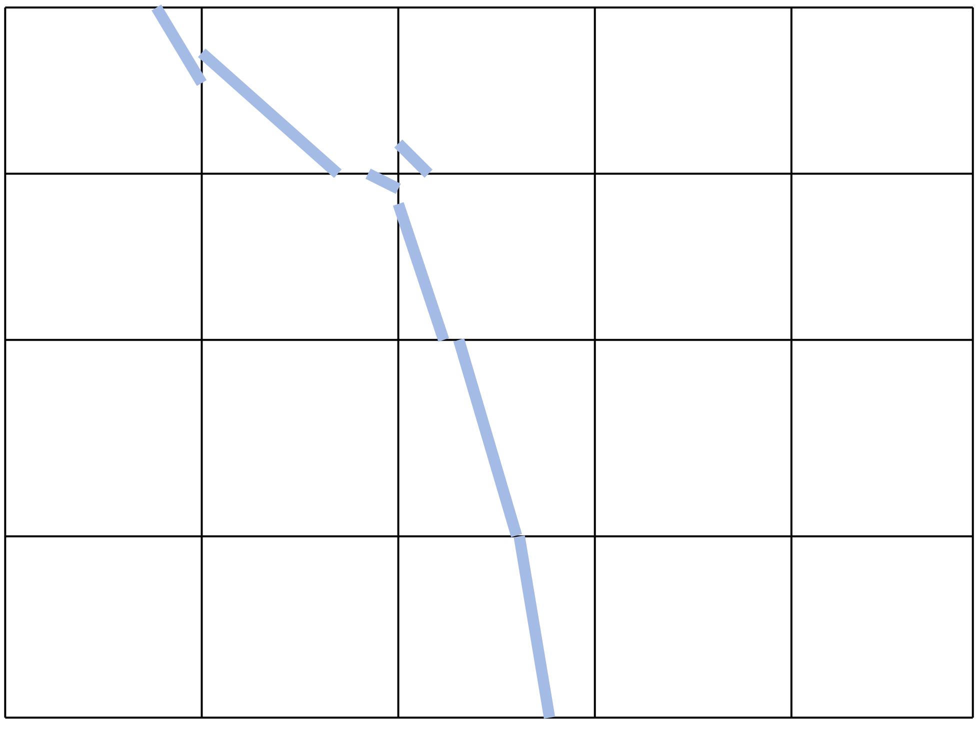
- Combined Level-Set Volume Of Fluid (CLSVOF, Berlemont, Ménard)

- Particle – Level-Set

- Lattice Boltzmann – Front Tracking

Just figuring out which method to use is already a challenge !

The Volume-of-Fluid method



C_{ij} = Volume of « fluid » in cell ij. Far from perfect

A bit of history

Second-order (linear) reconstruction methods:
de Bar, Kraken code, 1974

Momentum conserving
Rudman 1996

Height function methods yield better surface tension & curvature:
Sussman, 2003 , Popinet 2009

Coupled methods : with Level Sets or Front Tracking
Sussman, 2003, Aulisa, Manservisi, Scardovelli 2003

Exactly (machine accuracy) mass conserving in 3D
Weymouth & Yue 2010

3D curvature tests
Desjardins 2015, Popinet 2015.

Two kinds of VOF methods:

- « off the shelf » methods for hyperbolic PDE / gas dynamics. (OpenFoam, JADIM etc. .) unstructured grids

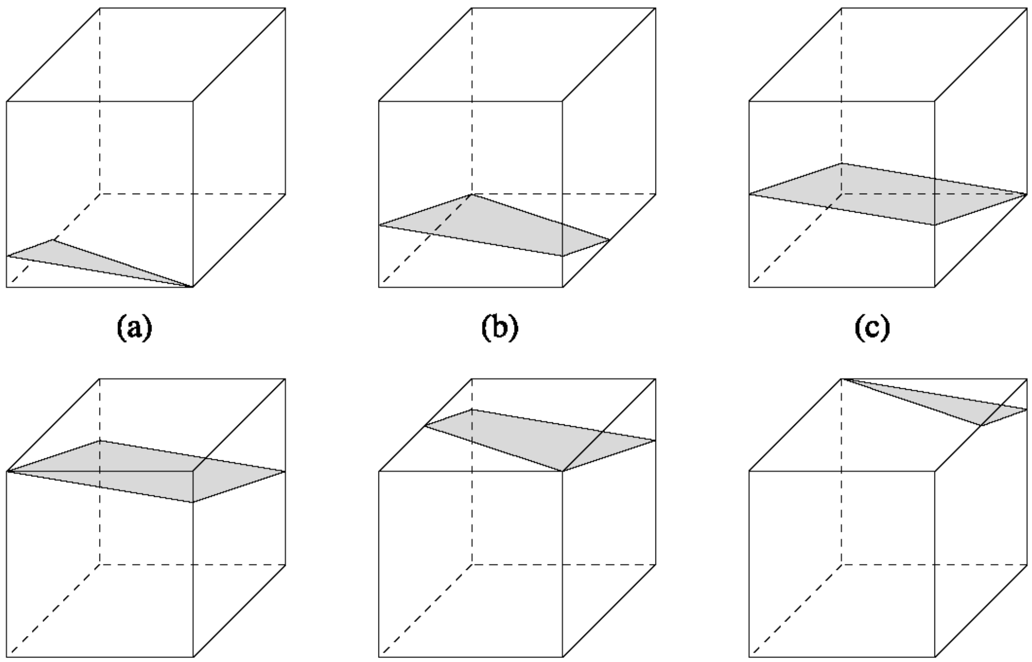
- methods involving geometric operations (Surfer, Gerris, Basilisk, ParisSimulator) regular grids

Reconstruction in 3D

- compute interface normal **n** (finite difference, more precise methods such as height functions)

- place interface segment once interface normal is known (12 critical cases in 3D for all $n_i > 0$)

Reconstruction



(a) (b) (c)

(d) (e) (f)

There are 12 critical cases in 3D. Here are 6 of them.

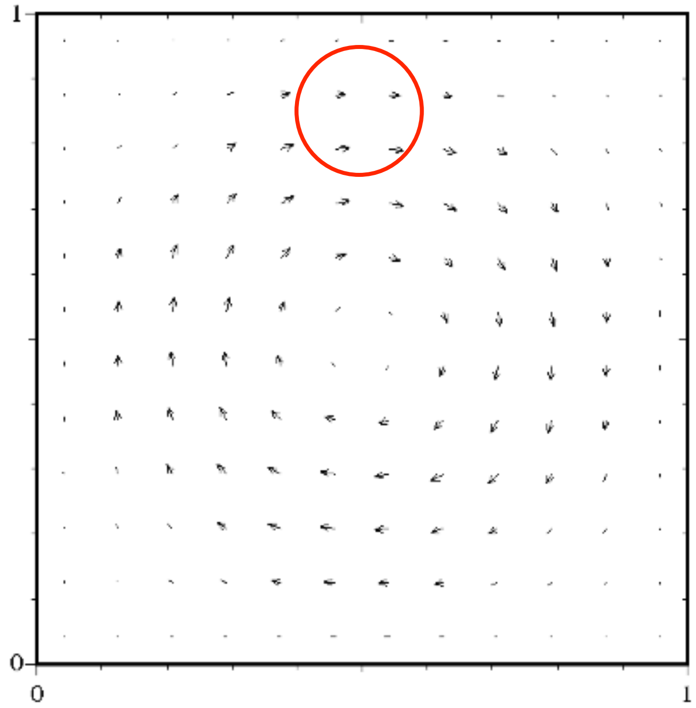
Surface evolution tests

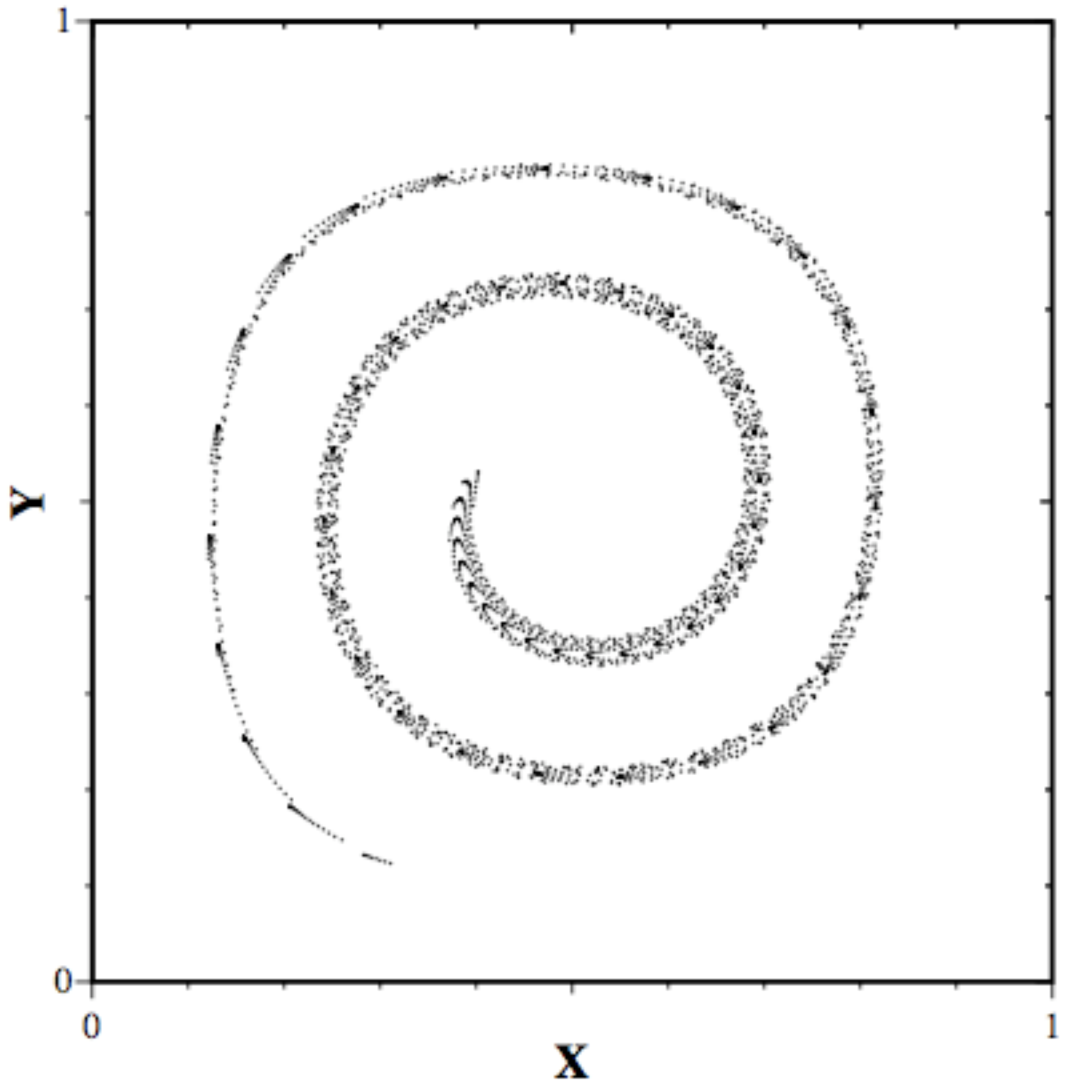
Kothe and Rider's spiralling, stretching and reversing flow.
stream function:

$$\Psi = \sin^2(\pi x)\sin^2(\pi y)\cos\left(\frac{\pi t}{T}\right)$$

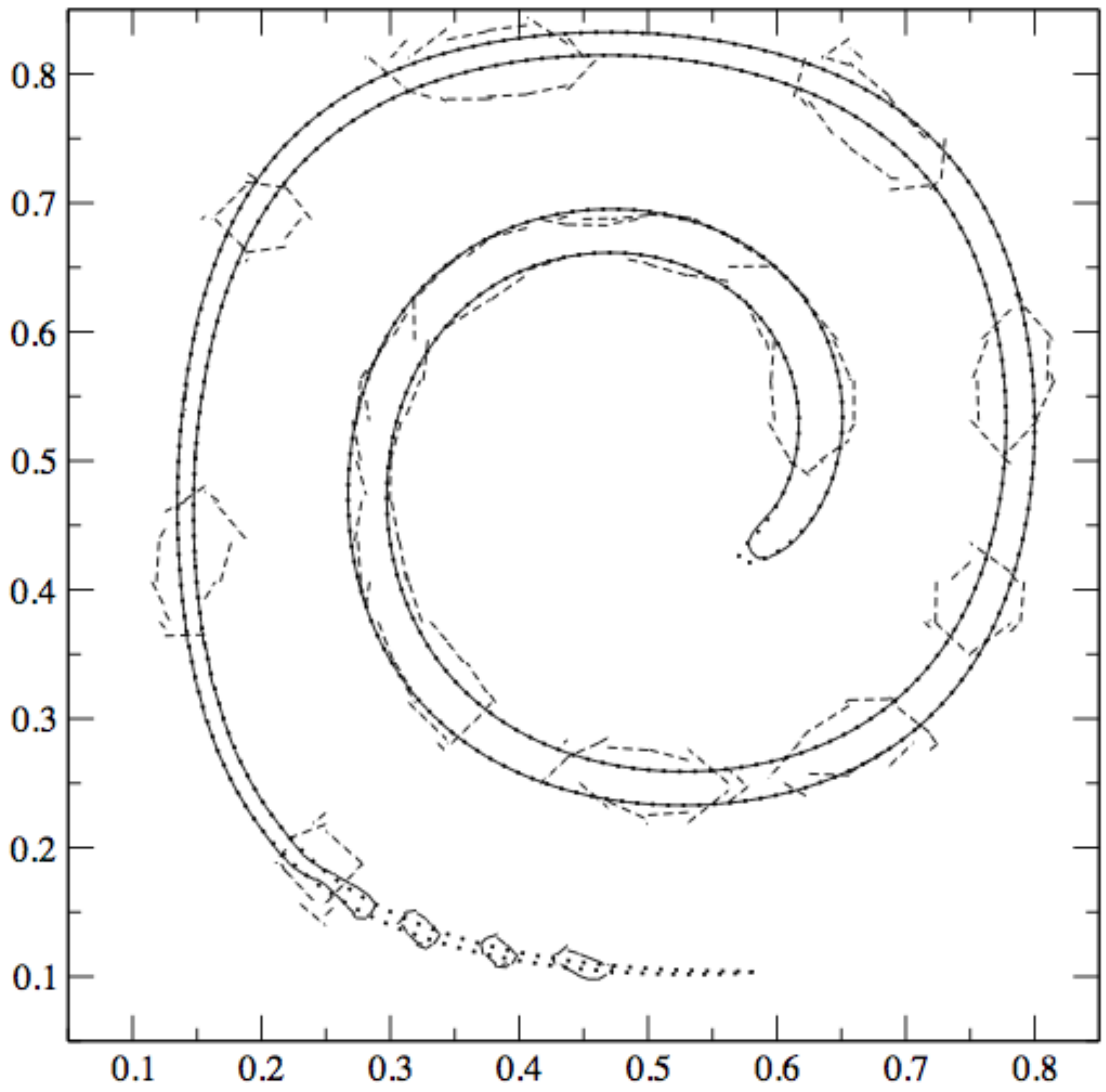
Kothe and Rider vortex in cell test:

$$\Psi = \sin^2(\pi x)\sin^2(\pi y)\cos\left(\frac{\pi t}{T}\right)$$





Reference solution obtained by particle tracking for large T.



VOF and reference solution .

2. Navier-Stokes equations with interfaces

$$\partial_t(\rho\mathbf{u}) + \nabla\cdot(\rho\mathbf{u}\mathbf{u}) = -\nabla p + \nabla\cdot(2\mu\mathbf{D}) + \sigma\kappa\delta_S\mathbf{n} + \rho\mathbf{g},$$

where the strain-rate tensor is:

$$D_{ij} = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} + \frac{\partial u_i}{\partial x_j}\right),$$

and both fluids are considered incompressible

$$\nabla\cdot\mathbf{u} = 0\,.$$

Compressible fluids: possible but difficult and less relevant.

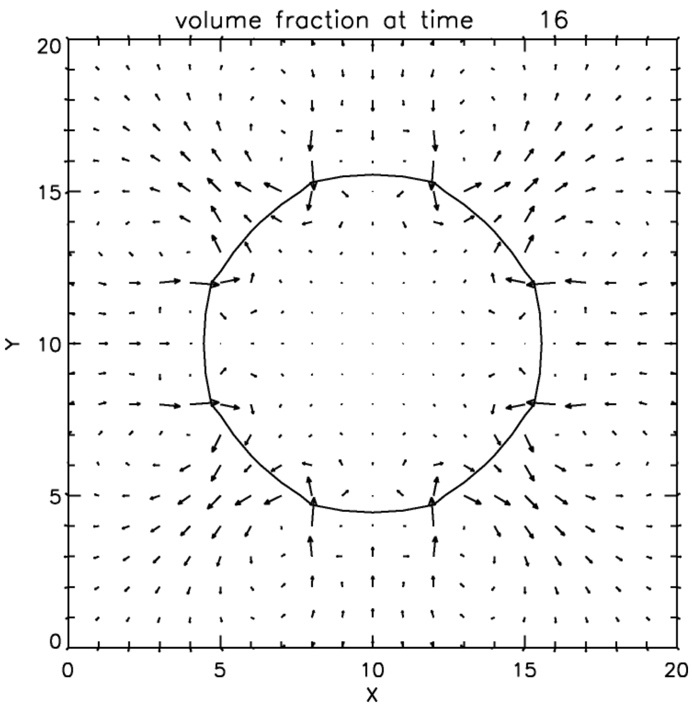
3. Surface tension

Treatment of surface tension by Continuous Surface Force («CSF» method, Brackbill, Kothe and Zemach JCP 1993)

$$\sigma \kappa \mathbf{n} \delta_S \approx \sigma \kappa^h \nabla^h C$$

Many methods for κ .

Challenge: large density ratio and large surface tension



Spurious currents
around a static bubble.

Origin of spurious currents: the discrete static equibrium equation (Laplace's law)

$$-\nabla^h p + \sigma\kappa \ \nabla^h C = 0$$

is not verified in some discretization schemes !

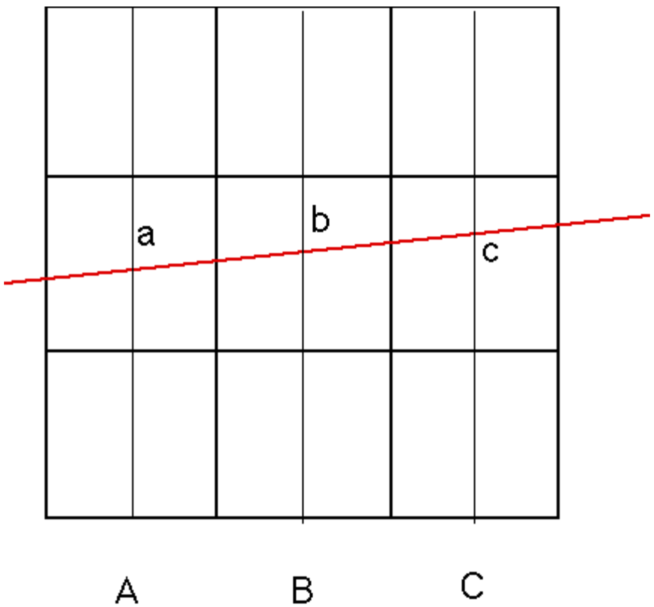
Solution

1) Use « balanced force » algorithm (Francois et al. J. Comput. Phys. 2006) (More generally use ghost-fluid methods that fix discretization of discontinuous functions)
2) Improve computation of curvature κ (HF methods, levels sets) .
3) Wait long enough for physical droplet oscillations to be damped. (see Popinet JCP 2009).

How to perform accurate curvature computations:
use height function methods

height at i : h_i = Aa. At second order:

$$h_i = C_{i,j-1} + C_{i,j} + C_{i,j+1}$$



Then h_i is a second-order approximation of the exact interface position.

Three estimates of height h_i then yield curvature κ from a finite difference approximation of the second derivative of h.

The issue of spurious currents is not solved: the community now focuses on spurious currents in a moving bubble (in a uniform velocity field). (see work of T. Abadie and D. Legendre at IMFT on microfluidics)

Then typically spurious velocity fluctuations with a dimensionless magnitude of the order of 10^{-3} are observed.

4. Variable viscosity

- Two possibilities in mixed cells: arithmetic mean or harmonic mean

- The harmonic mean is exact for a linear, parallel shear flow.

4. Variable density

- use momentum-conserving methods (Rudman, Raessi and Bussman, Le Chenadec, Berlemont, Ménard etc..) : Advect the momentum near the interface using the same scheme used for the VOF color function.

- use extrapolation methods (Sussman et al., Xiao, Dianat & Mc Guirk) : extrapolate the liquid velocity field in gas nodes.

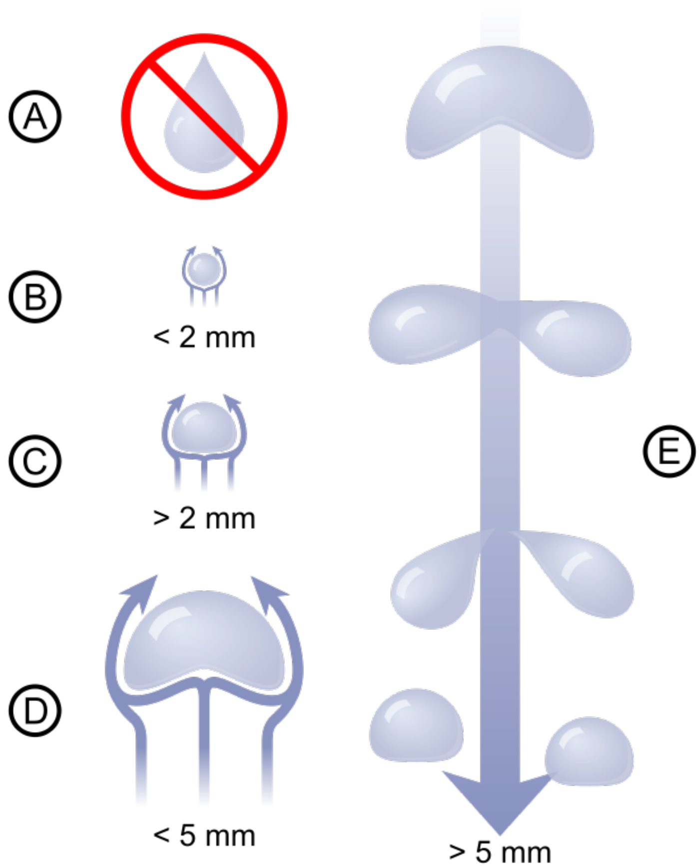
- combine above methods with flux limiters.

- filtering

need other ideas: for instance, doing a falling rain drop of 1,5 mm is already very difficult and requires 200 grid points / diameter.

An example of a difficult high Re flow: raindrops

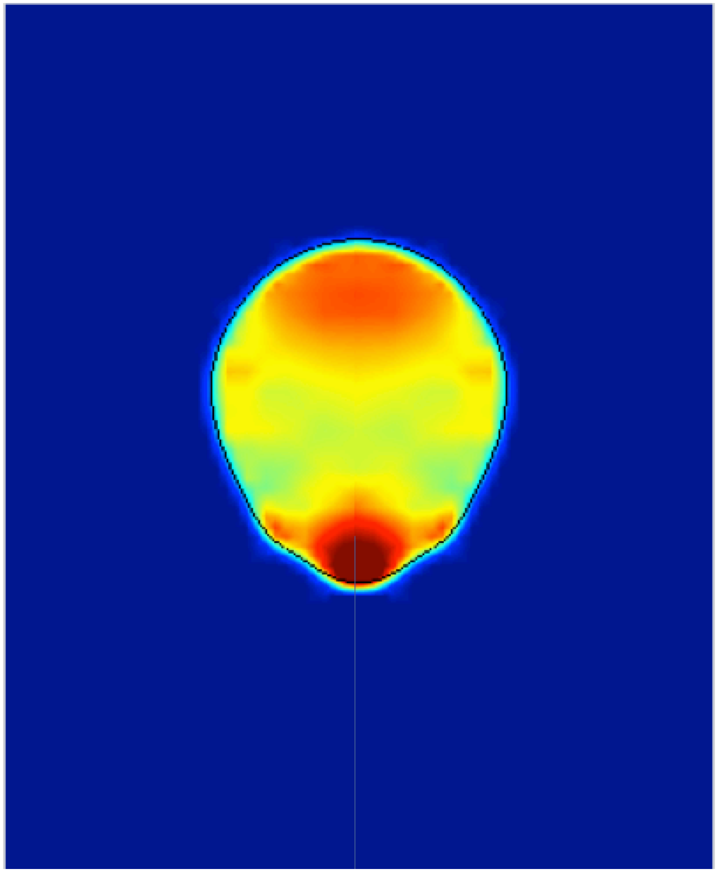
The problem has large air/water density ratio + surface tension.



Use Gerris for a 2mm drop

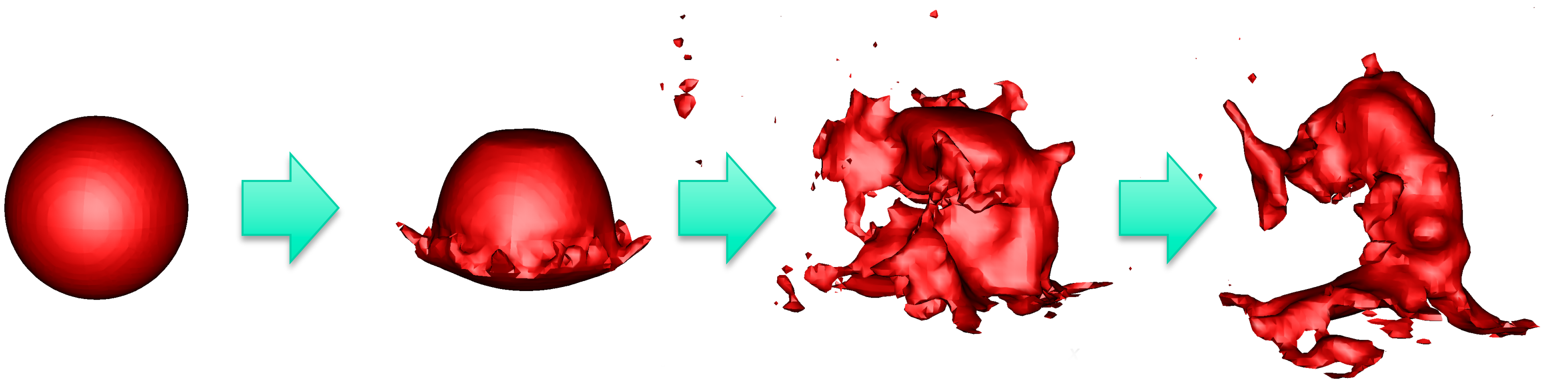
With unsufficient grid resolution the droplet shape and pressure distributions are completely wrong. The deformation increases in time, creating a droplet that is elongated in the vertical axis.

Results obtained with VOF similar results with CLSVOF.



pressure map

It is worse in 3D (After Feng Xiao, Danat & Mc Guirk, Leicester)



Solution: improve the mesh refinement.

$4{,}096^2$

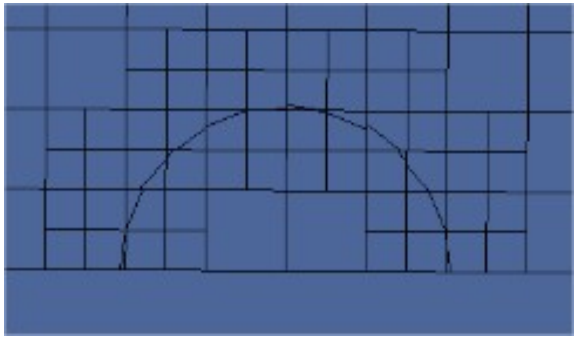
$D/\Delta x = 8$

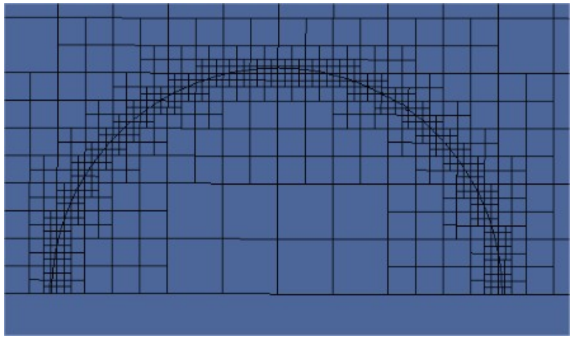
$32{,}768^2$

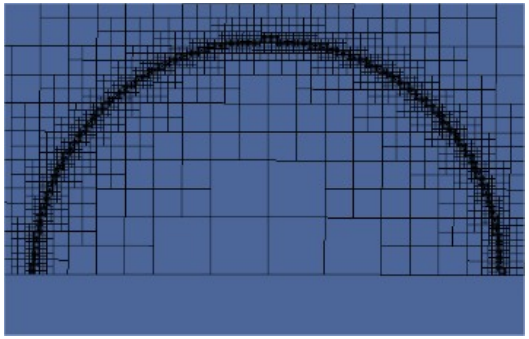
$D/\Delta x = 64$

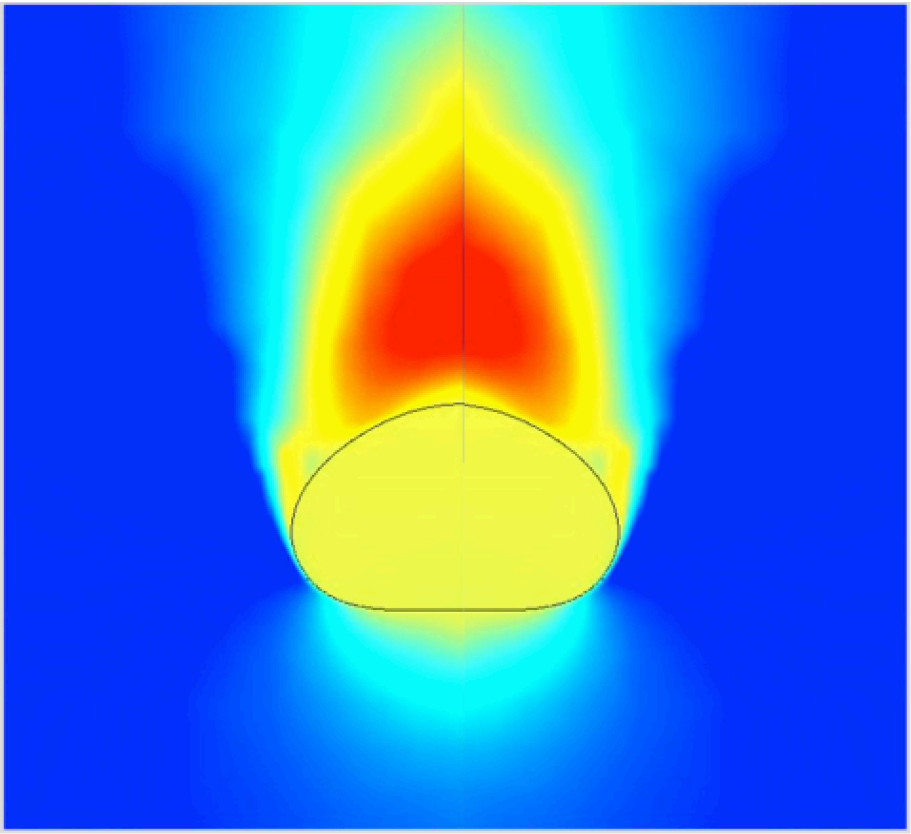
$131{,}072^2$

$D/\Delta x = 256$!!!

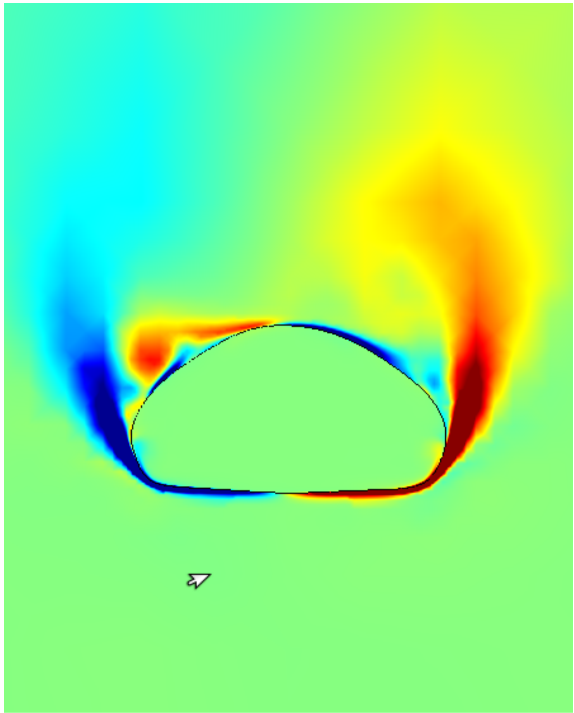








velocity field (axi)



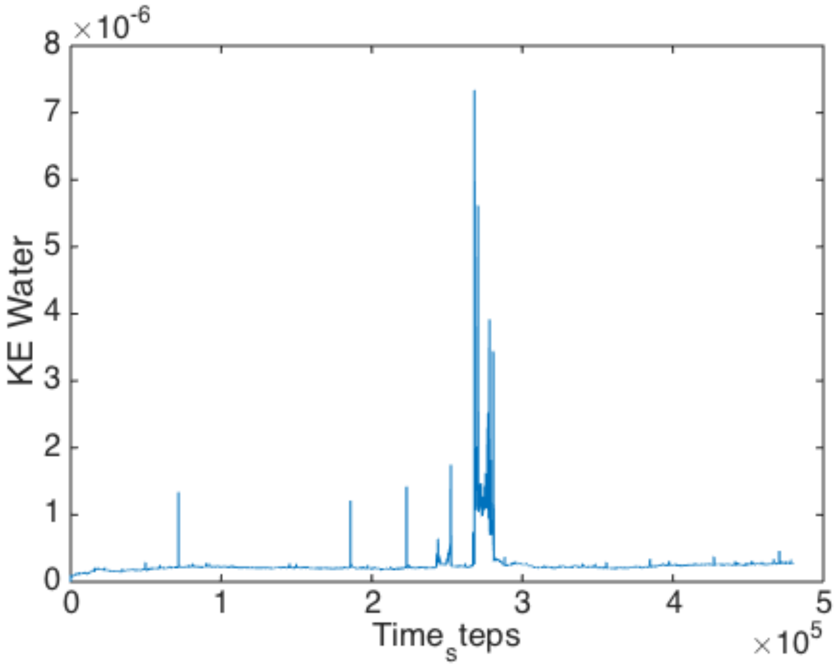
vorticity field (2D)

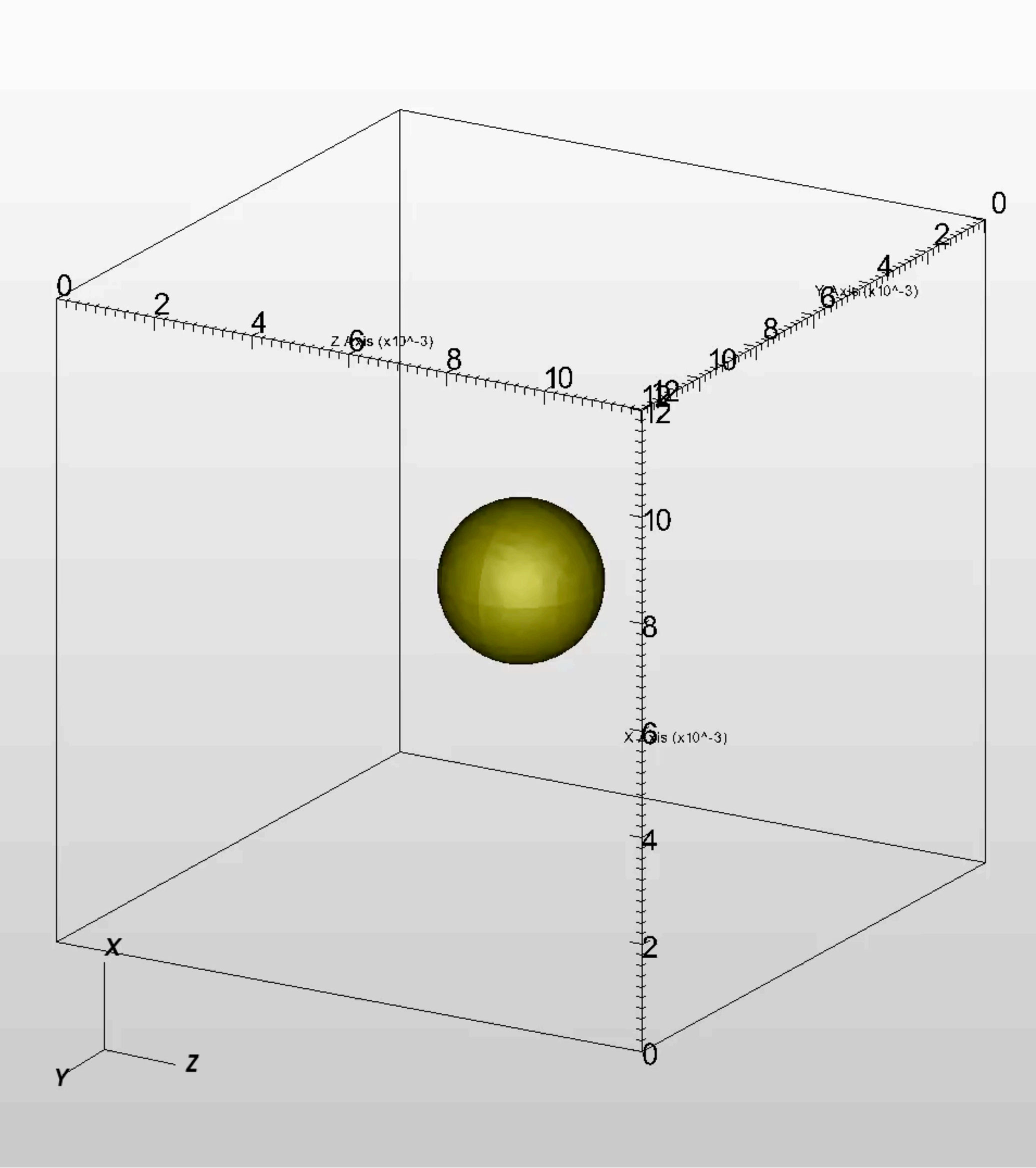
McGuirk, Xiao & Dianat's solution: use extrapolation.

our solution: use momentum-conserving VOF.

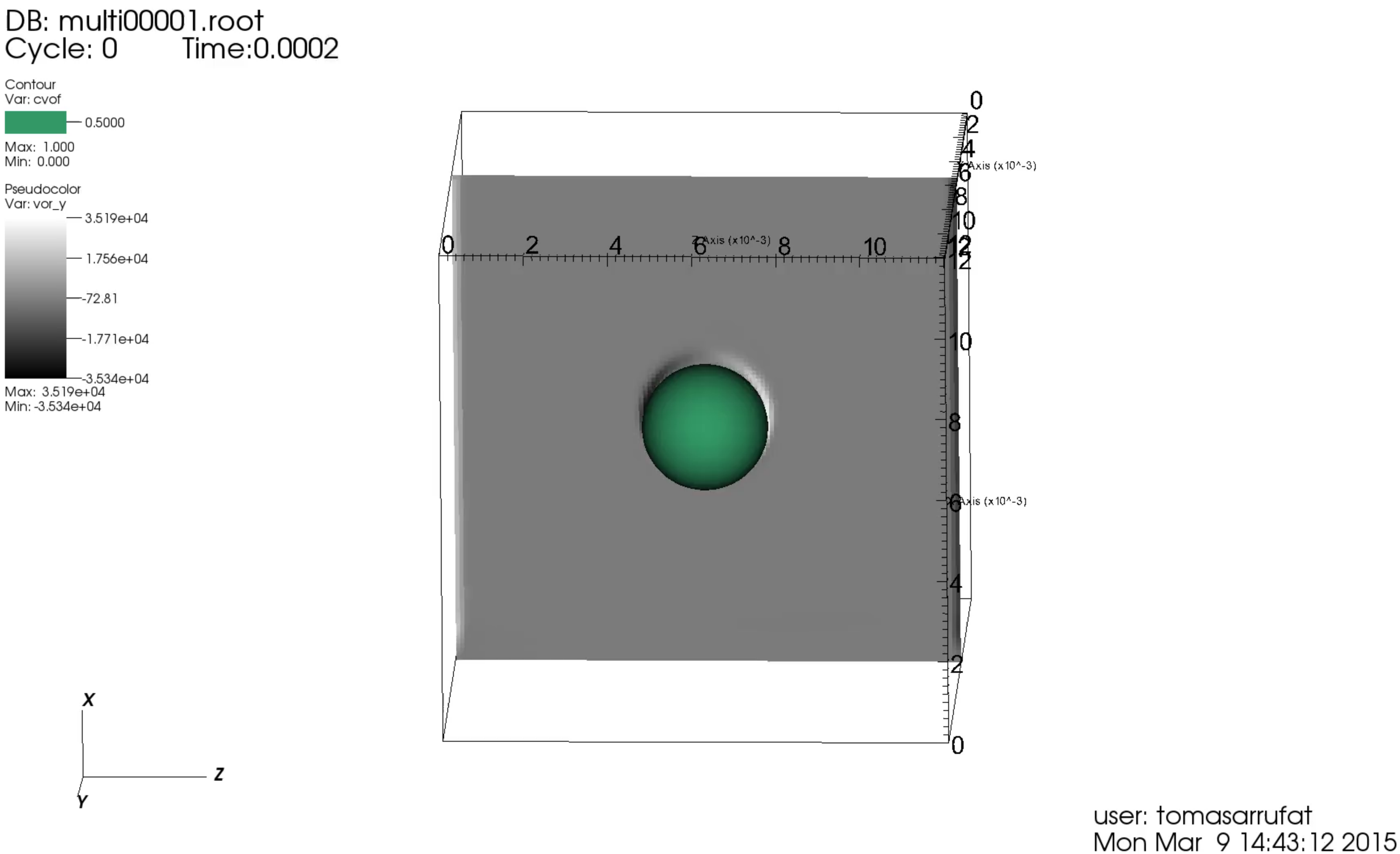
Refinement level 1.D/Δx=15.
Spurious currents observed.

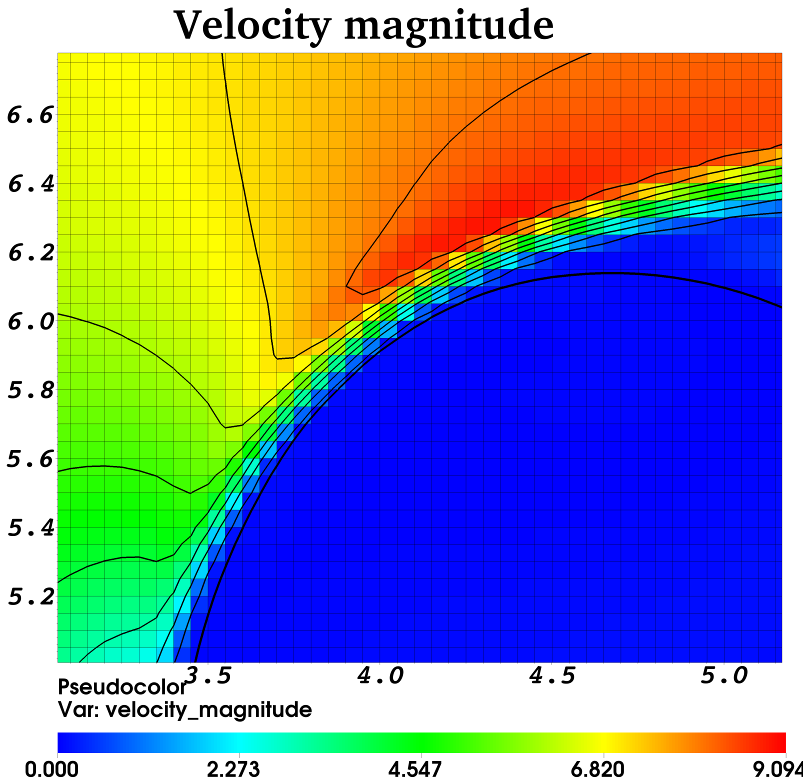
Kinetic energy of the droplet

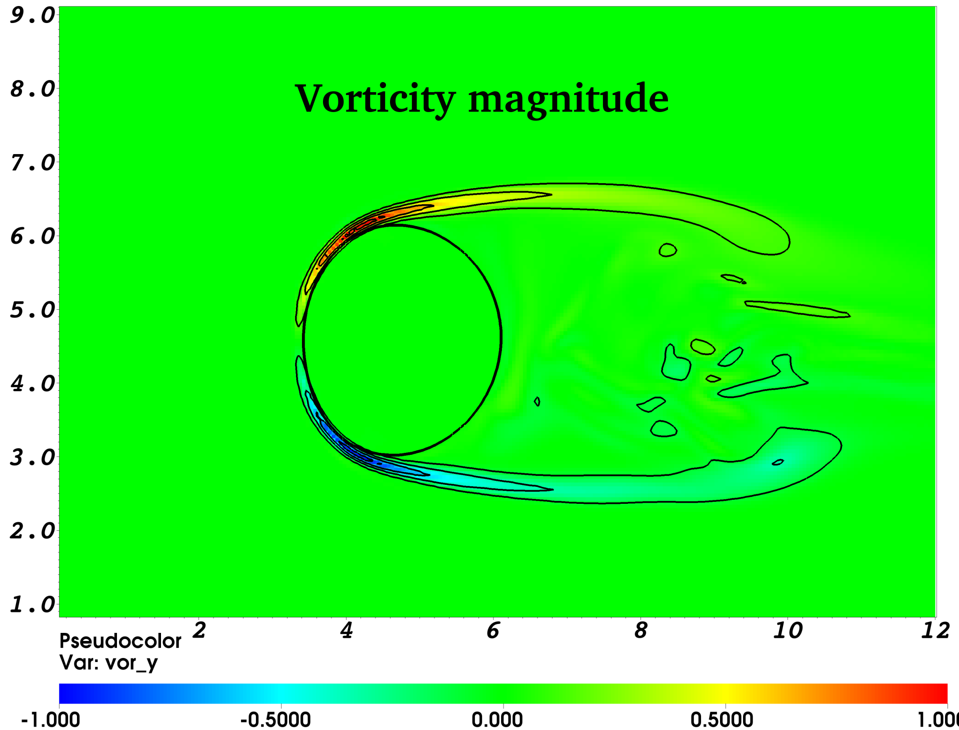




diameter d=8 mm





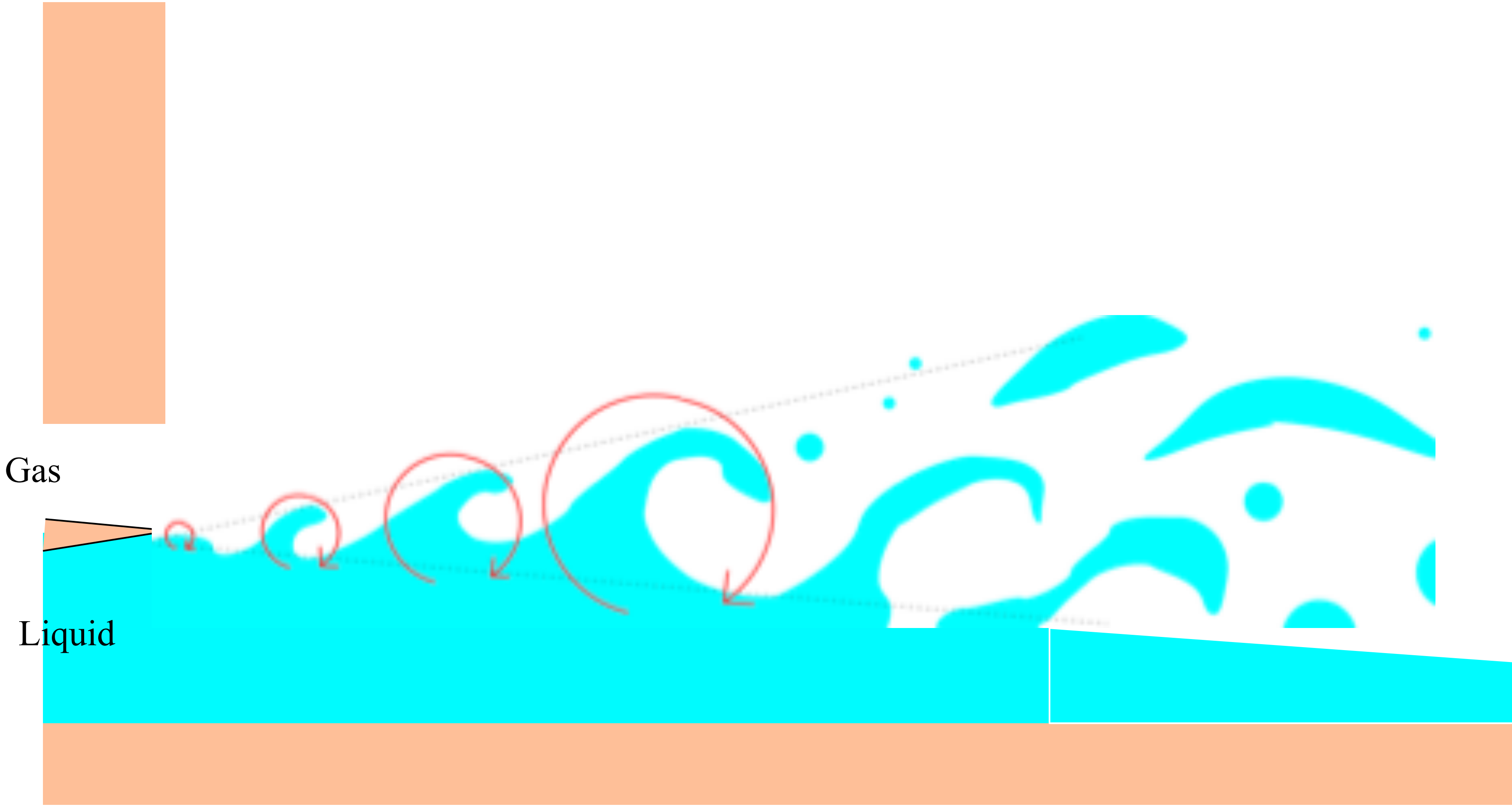


- Even for D/Δx = 60 the boundary layer is only covered by 5 cells.
- The boundary layer is very small relative to the droplet diameter.
- Such results suggest that the accurate solution of air flow with water droplets can be extremely challenging.

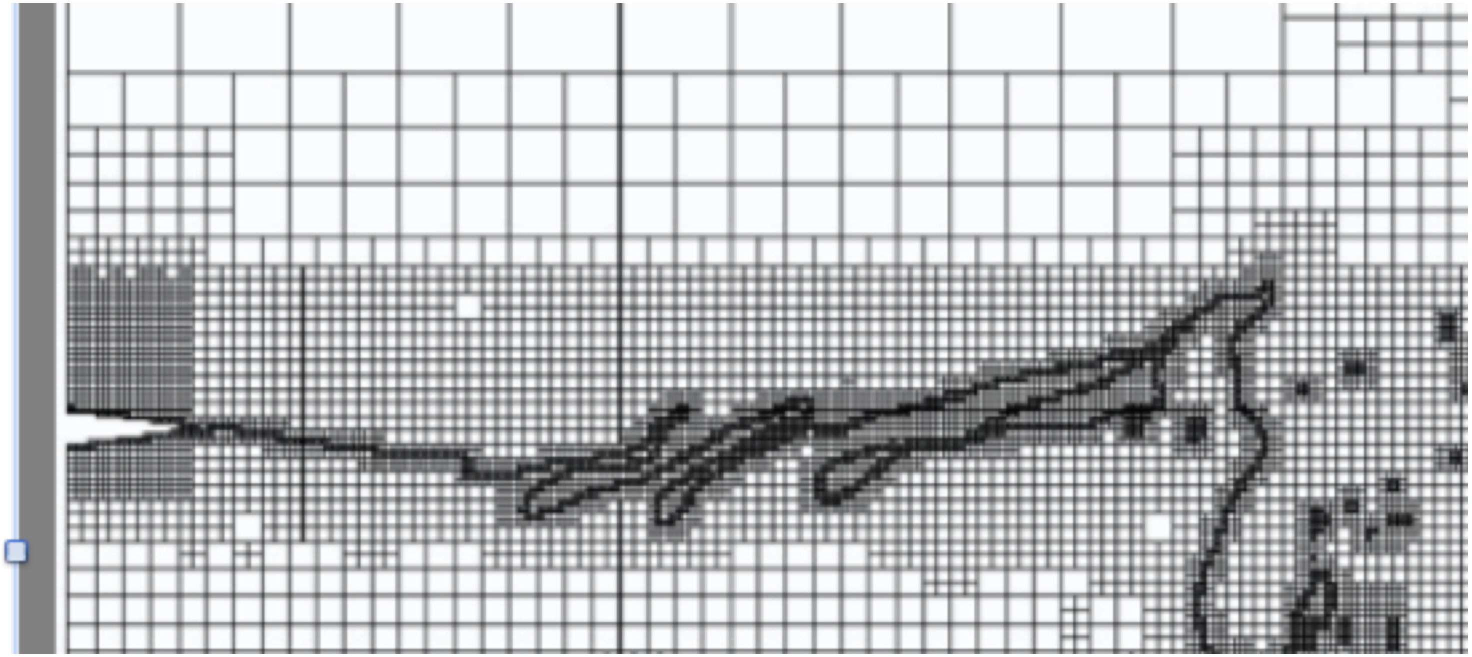
2D simulations, Gerris

2D simulations of the planar « Grenoble » setup.

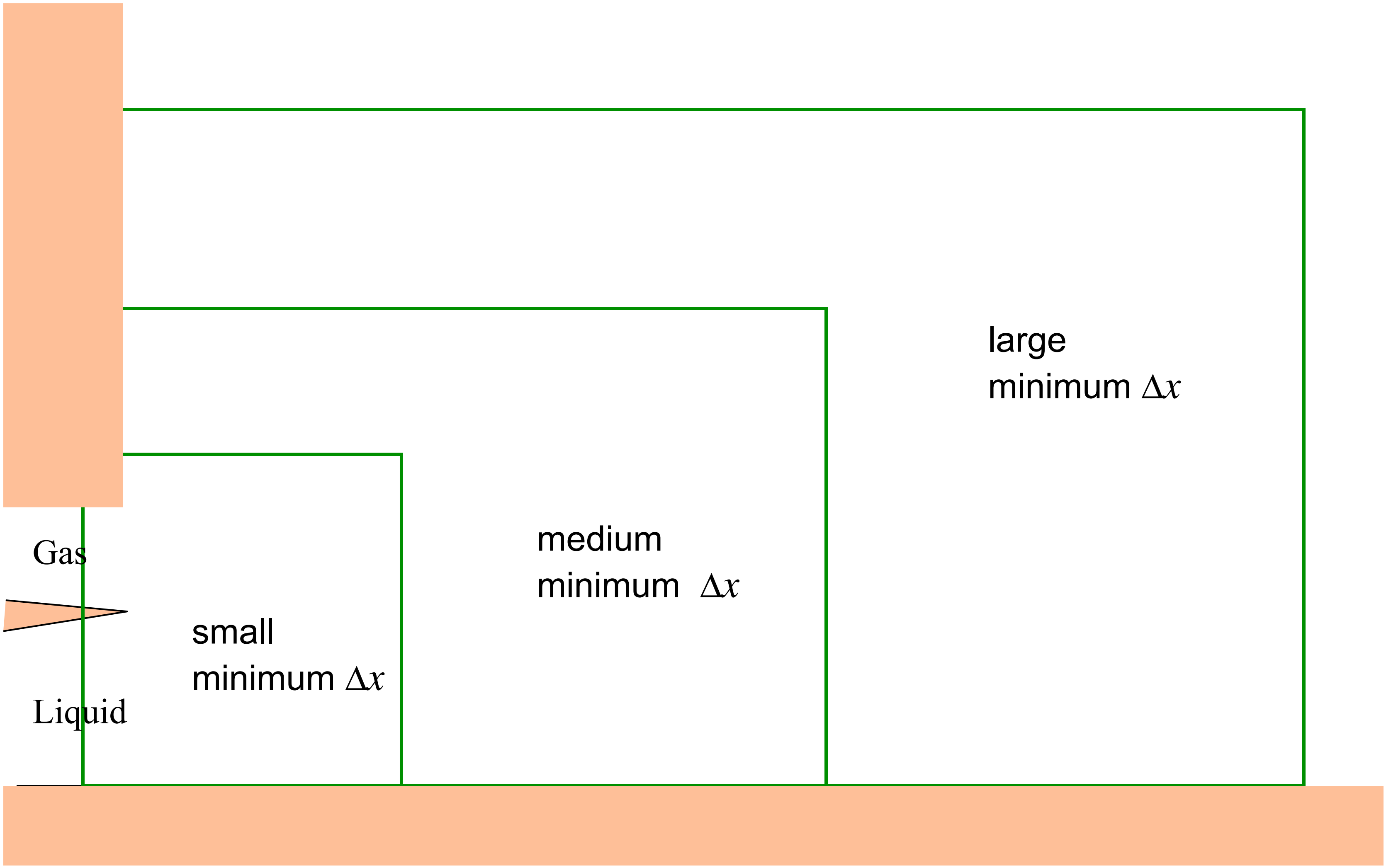
The Grenoble quasi 2D experiment set up



Use Gerris flow solver (S. Popinet) with adaptive oct-tree and quad-tree grids

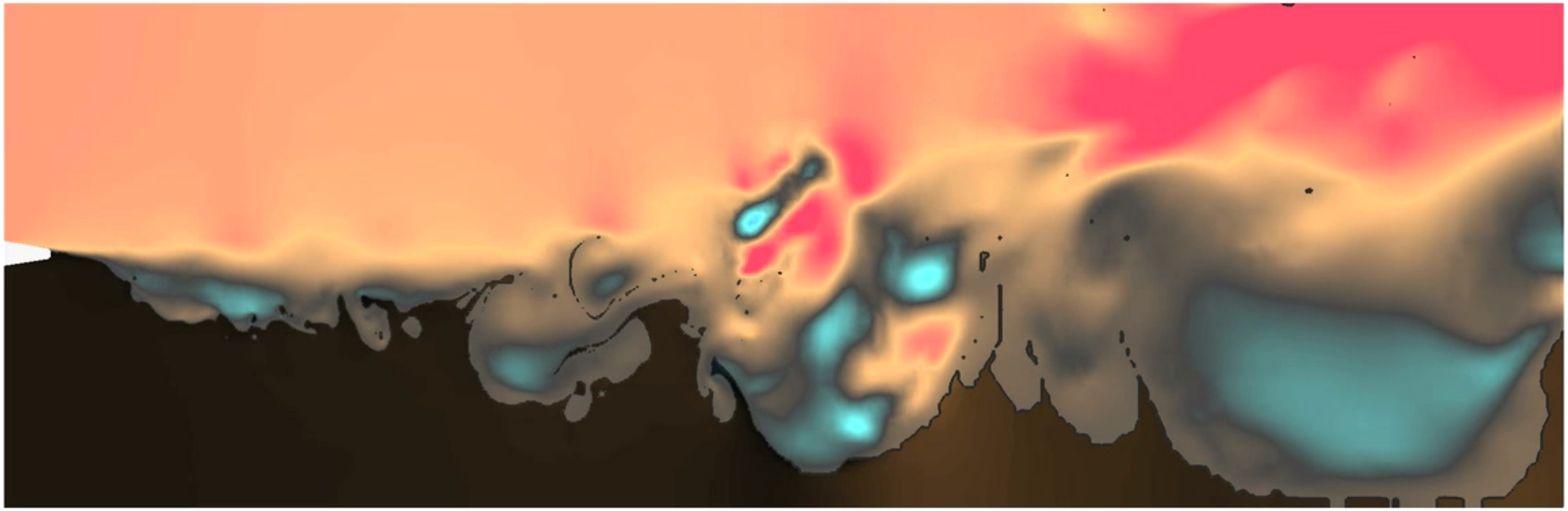


Elementary multiscale treatment: Navier-Stokes with variable minimum grid size according to a subdivision of the computational domain.

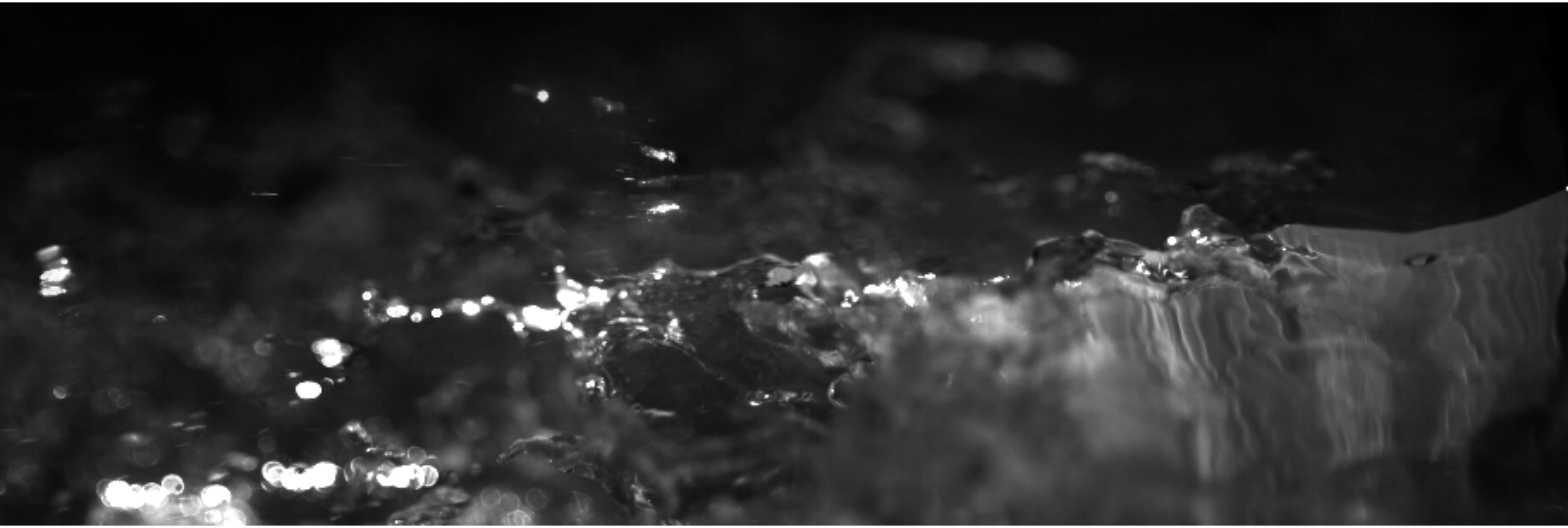


Simulation with a separator plate at density ratio ($1/r = 100$)

m	r	Re_g	Re_l	We_g	We_l	M
0.017	0,01	2640	290	19	8	2,4



Movie by Daniel Fuster and Jérôme Hoepffner using the Gerris Flow solver



Compare to experiments in Grenoble (Cartellier, Matas) . Flow from right to left.
Video with help of Jérôme Hoepffner and Jon Soundar.

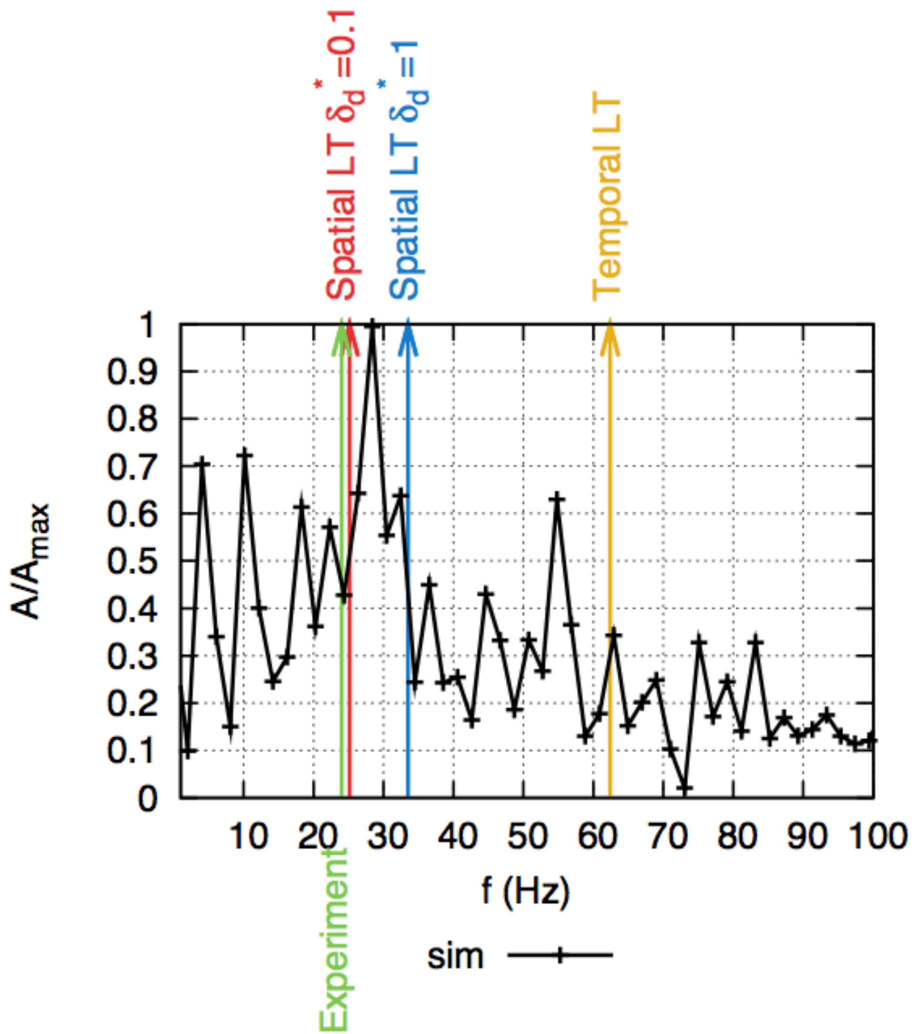
Now the ultimate test ! Compare :

- Experiments
- Numerics
- Linear theory

We need linear theory for spatially developping flows

(our group + many groups)

Comparison experiment-simulation



3D simulations

Perform simulation with the ParisSimulator code developped with Gretar Tryggvason, Ruben Scardovelli, Daniel Fuster and Stanley Yue Ling.

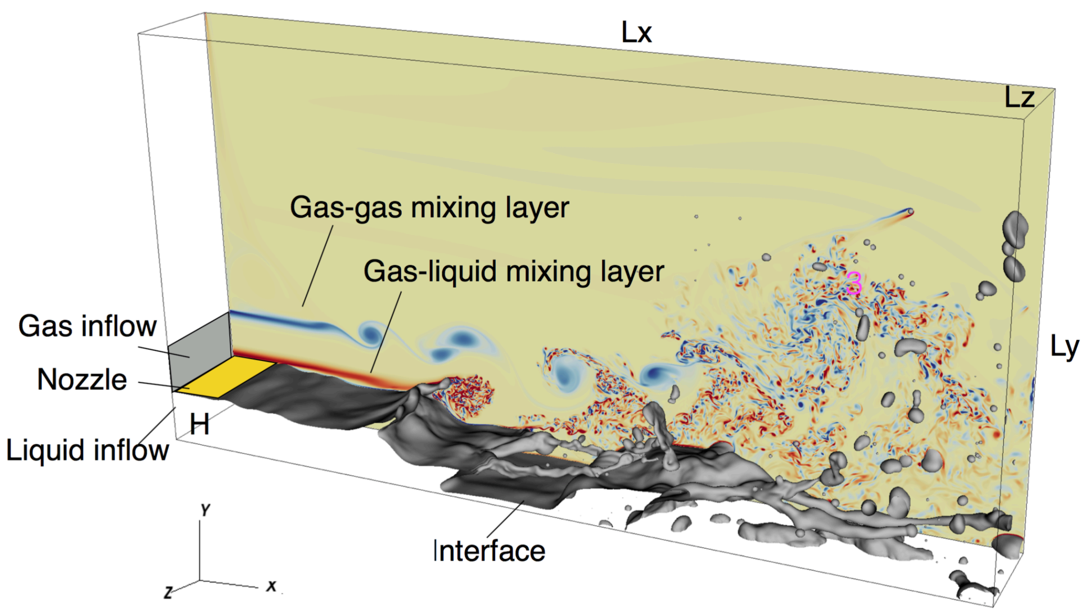
- Multimethod code with VOF and Front Tracking (In what follows we use only VOF)
- Momentum conserving.
- Regular grid, excellent parallelisation tested up to 32,000 cores.
- Written in Fortran 2003
- Free code

Real air-water parameters in ambient conditions are still too hard for a 3D detailed simulation.

Thus design a « synthetic » case. Parameters are chosen so that there is significant droplet production while avoiding exceedingly large Re and We to allow converged simulation.

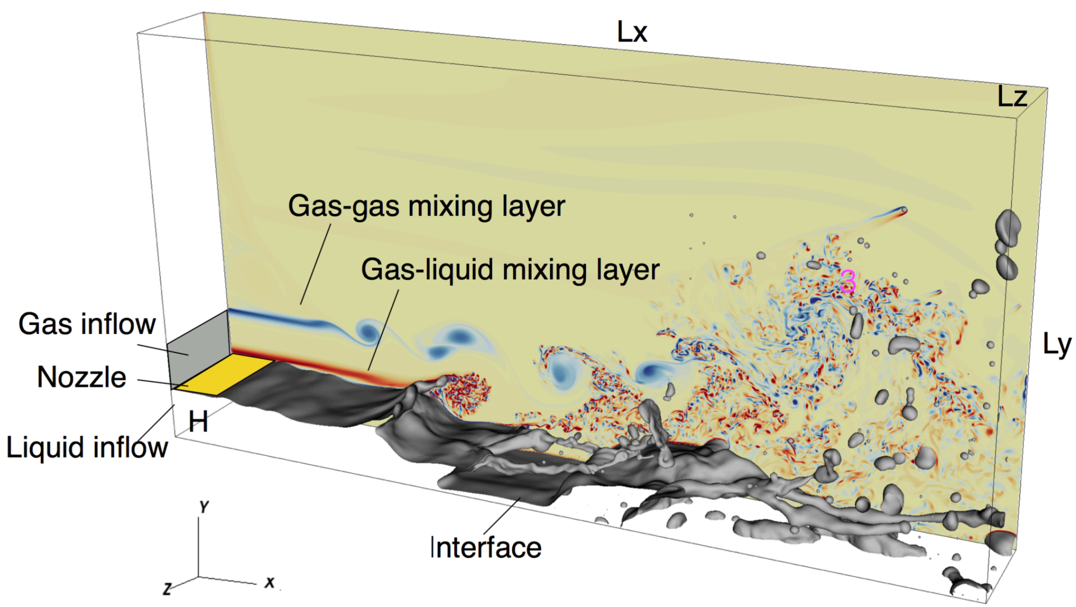
“A20” synthetic case: dimensional values

	Density kg/m³	Viscosity Pa-s	Surface Tension N/m	Jet Height H mm	Boundary Layer mm	Injection Velocity m/s
Gas	50	$5\ 10^{-5}$	0.05	0.8	0.1	10
Liquid	1000	10^{-3}		0.8	0.1	0.5



“A20” synthetic case: dimensionless values

M	$Re_{g,\delta}$	$Re_{g,H}$	$We_{g,\delta}$	r	m	v
$\frac{\rho_g U_g^2}{\rho_l U_l^2}$	$\frac{\rho_g U_g \delta}{\mu_g}$	$\frac{\rho_g U_g H}{\mu_g}$	$\frac{\rho_g U_g^2 \delta}{\sigma}$	$\frac{\rho_l}{\rho_g}$	$\frac{\mu_l}{\mu_g}$	$\frac{U_l}{U_g}$
20	1000	8000	10	20	20	20



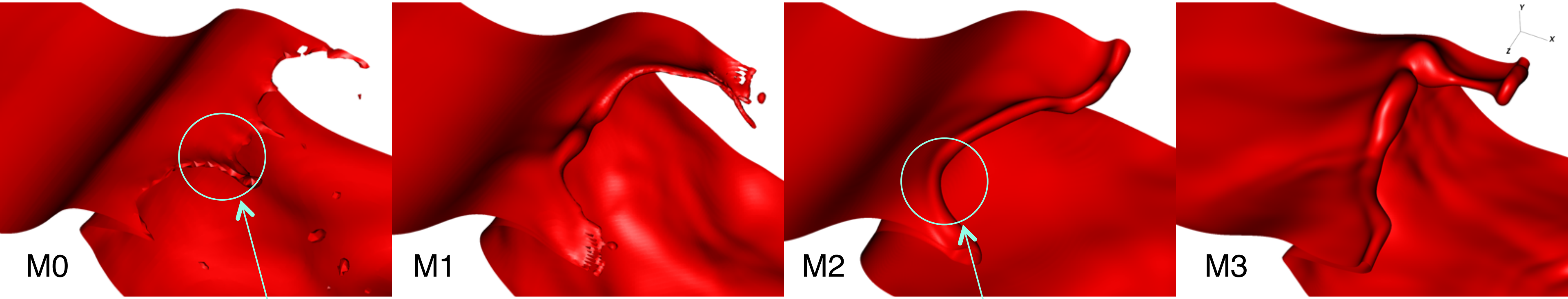
Grids

Domain: L_x=16 H, L_y=8 H,
L_z (various values, here 2 H) end-time: U_g t/H=400

Grids	h(μm)	H/h	# of cells	# of time steps	Total CPU time (hr)
M0	25	32	8.4 Million	$4.9\ 10^4$	$2.5\ 10^3$
M1	12.5	64	67 Million	10^5	$4.3\ 10^4$
M2	6.25	128	537 Million	$2.2\ 10^5$	$5\ 10^5$
M3	3.125	256	4 Billion	$4.5\ 10^5$	$20\ 10^6$

CPU time estimate based on performance on TGCC-CURIE machine

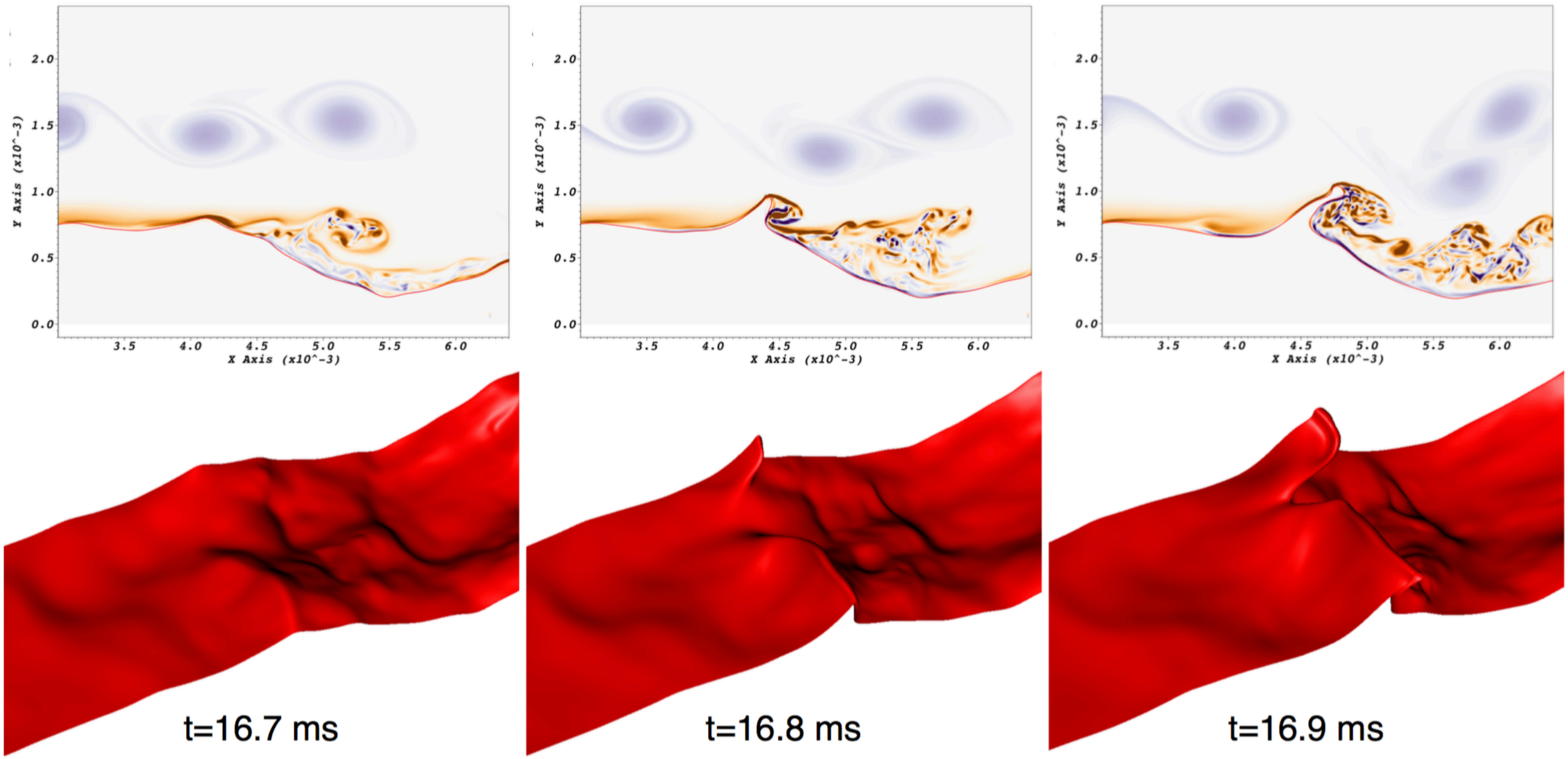
Effect of mesh resolution.

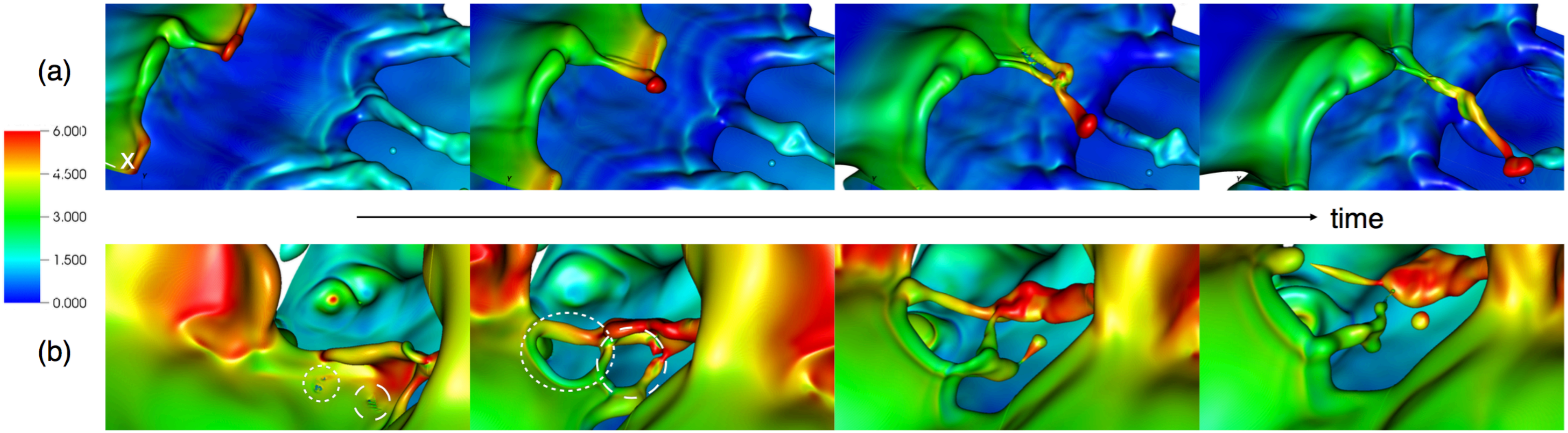


At low resolution, the tip of the sheet is « torn » in an irregular way.

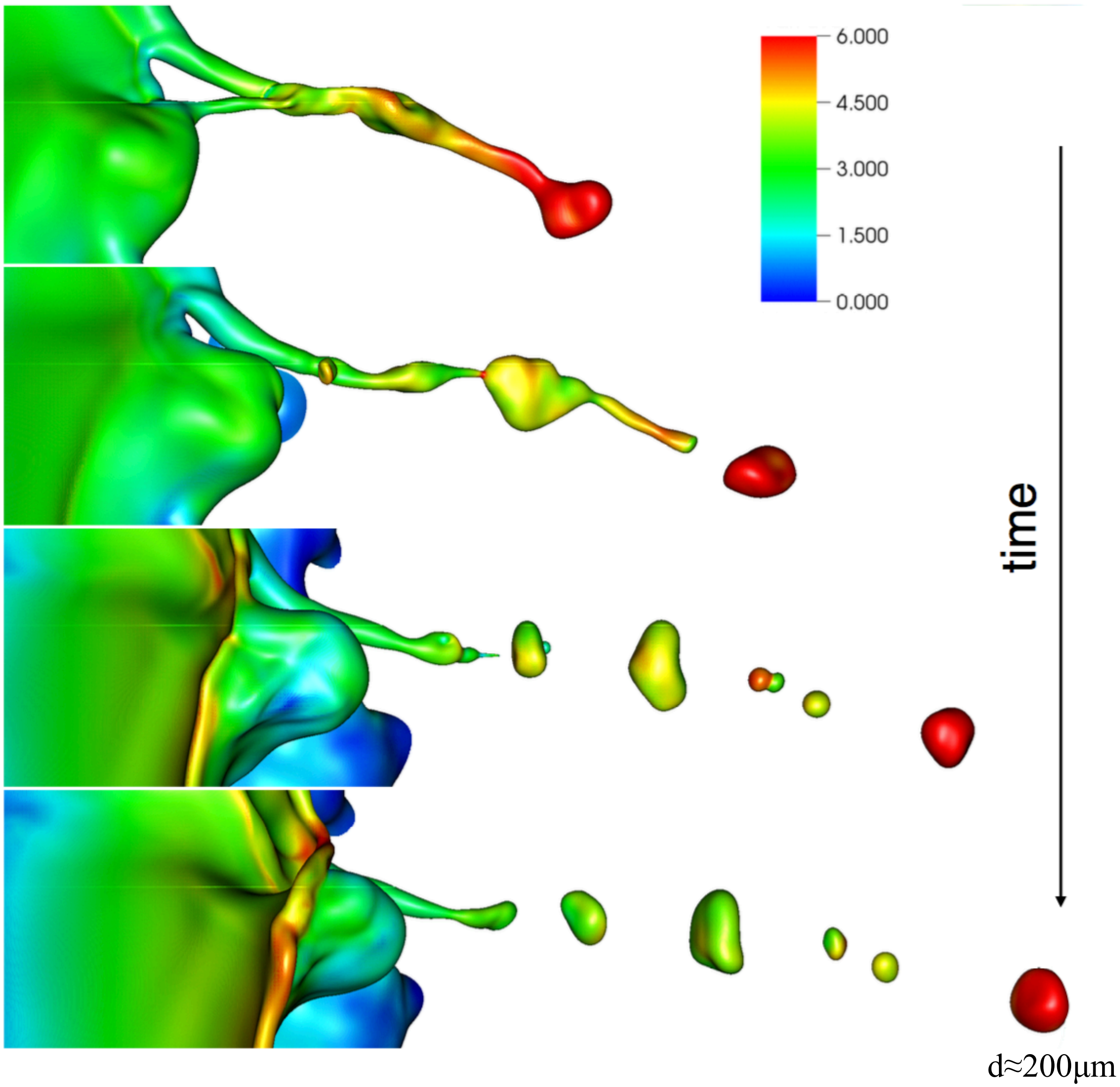
At high resolution, the tip of the sheet ends in a nice Taylor-Culick rim.

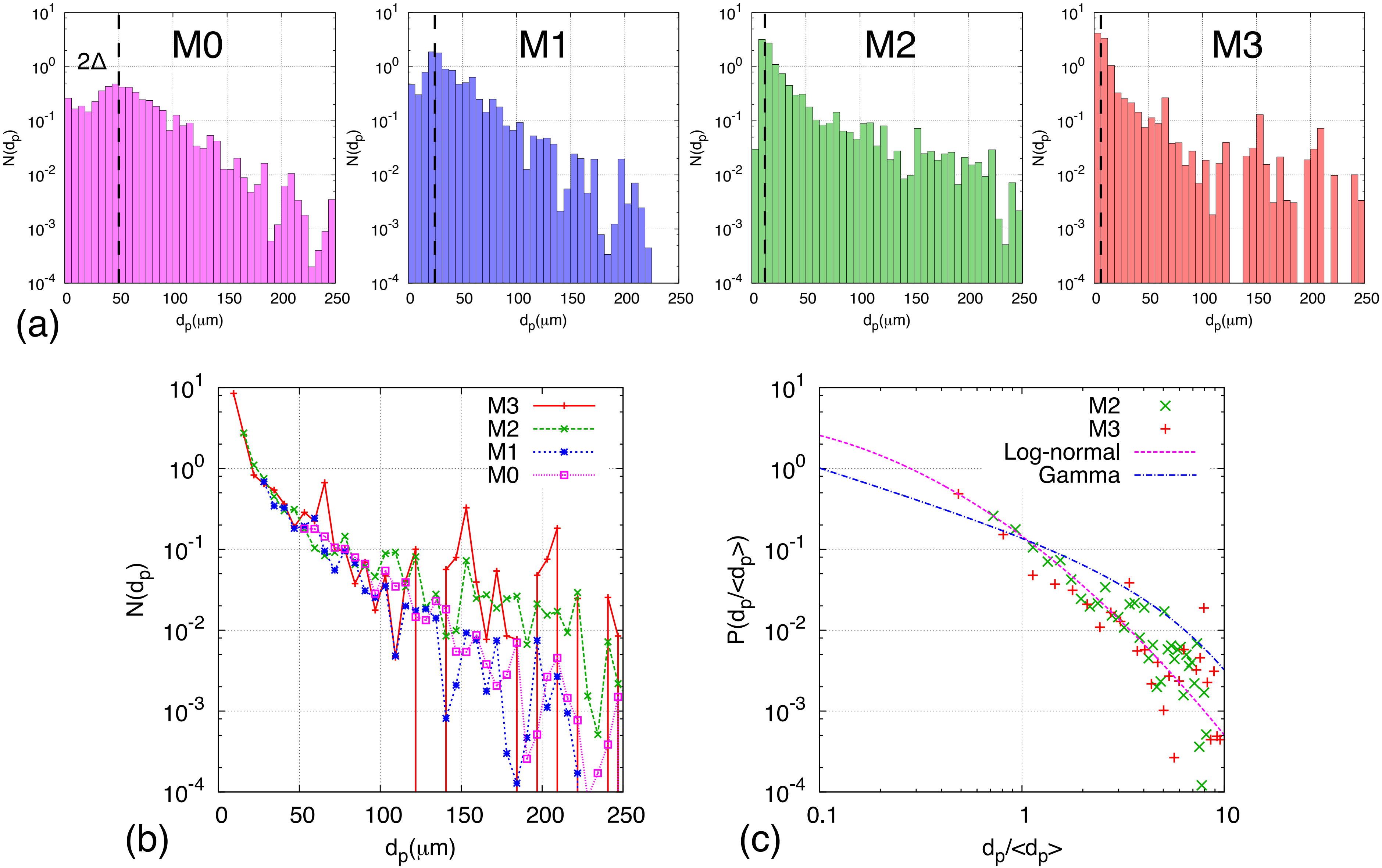
Sheet and rim formation dynamics





Ligaments formation due to (a) fingering from the tip of liquid sheet and (b) hole formation in the liquid sheet. The color on the interface indicates the streamwise velocity.


6.000
4.500
3.000
1.500
0.000
time
d≈200μm



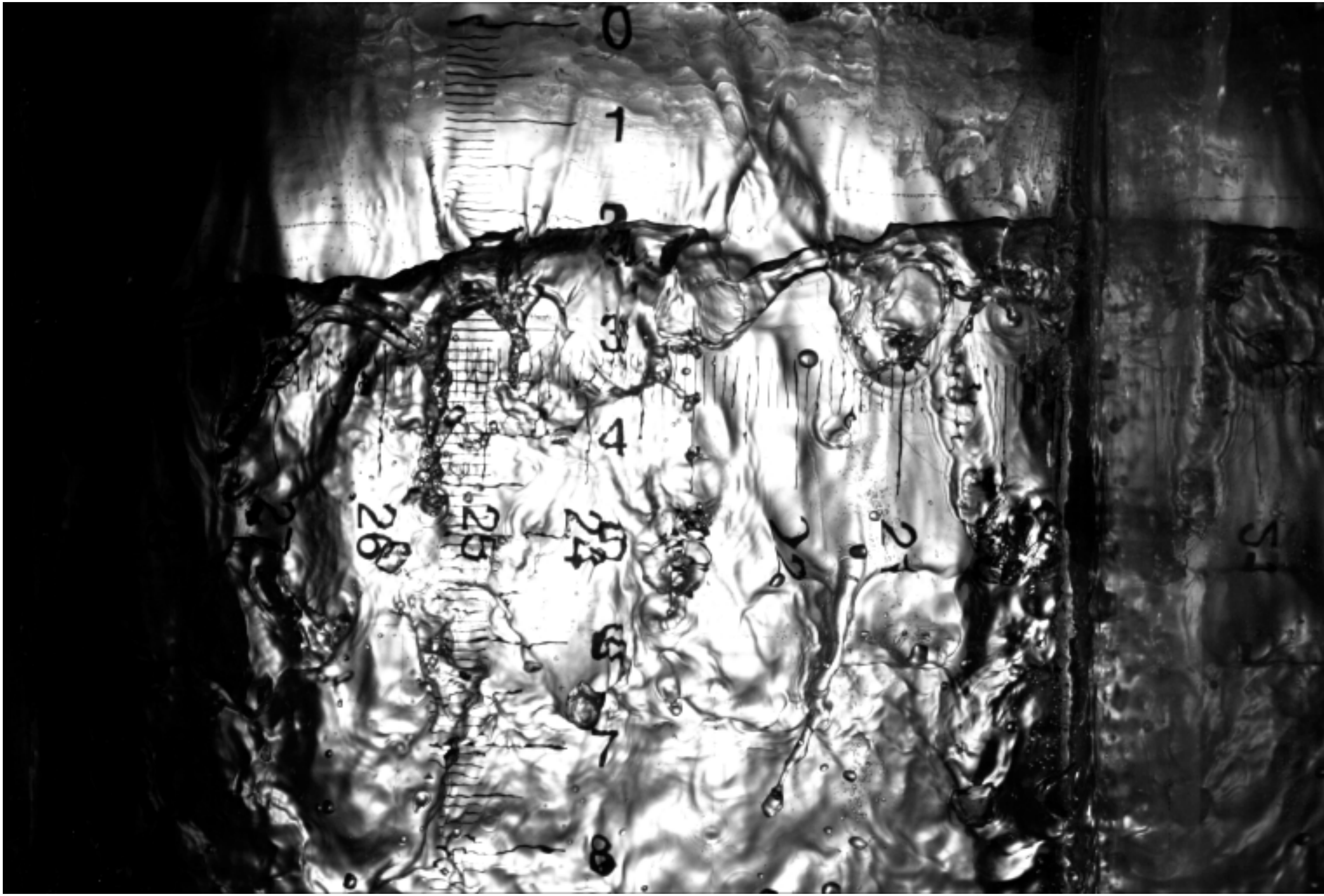
M0
2Δ
M1
M2
M3
N(dp)
dp(μm)
(a)
M3
M2
M1
M0
N(dp)
dp(μm)
(b)
M2
M3
Log-normal
Gamma
P(dp/<dp>)
dp/<dp>
(c)

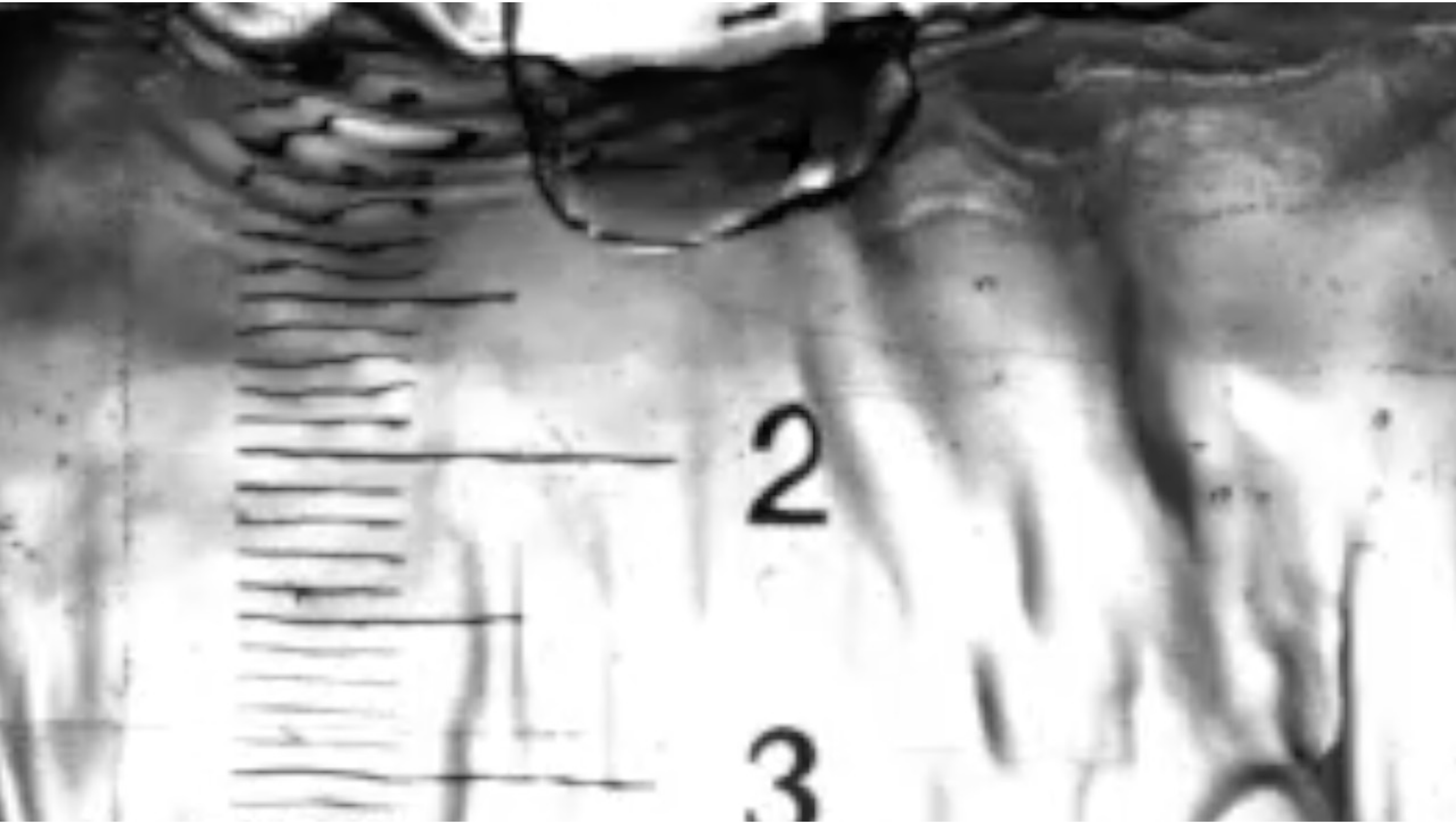
1) Is it possible to do a real Direct Numerical Simulation of atomisation, resolving all the scales ?

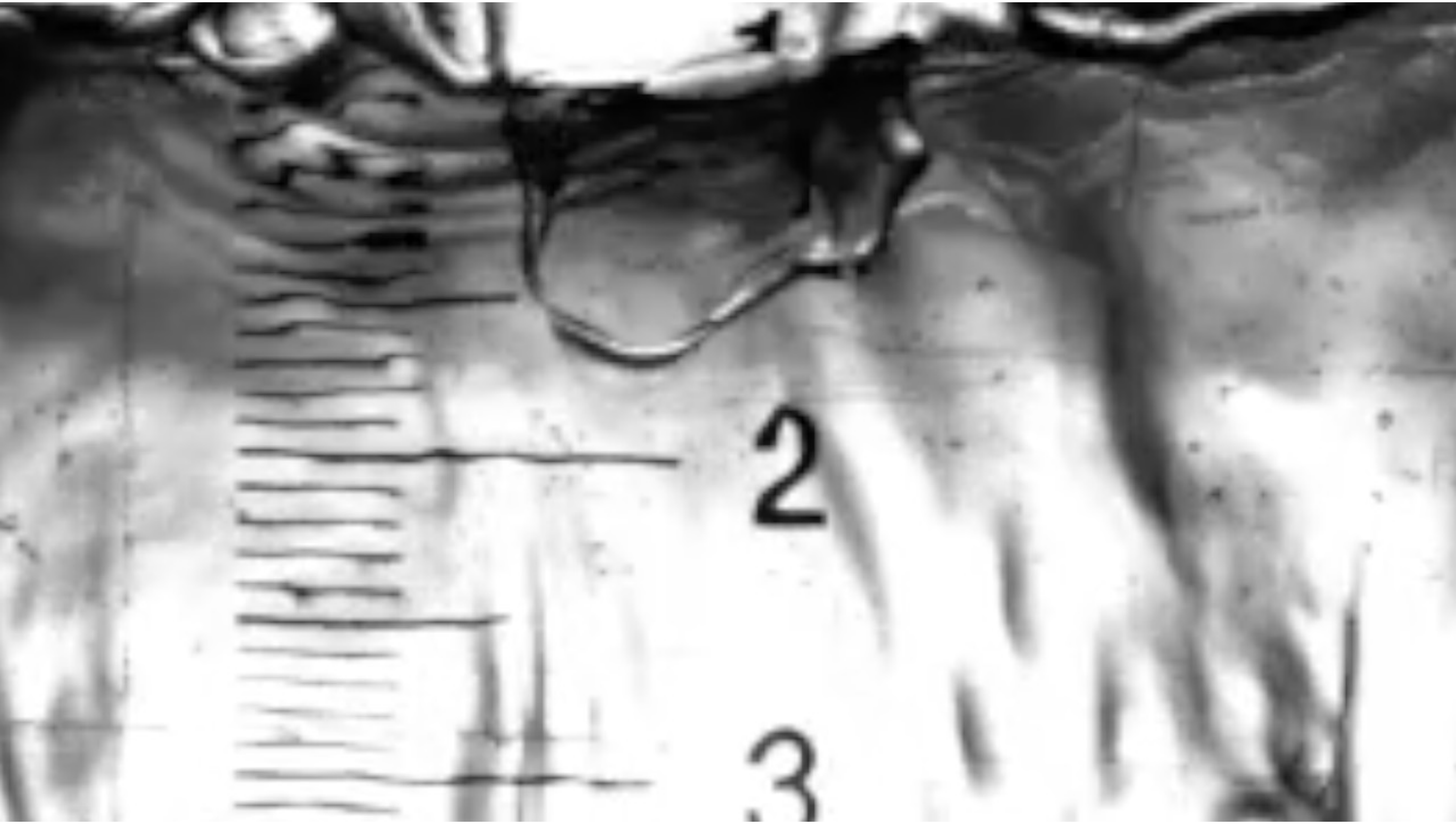
- Not yet. However some features (sheets, rims, part of the hole dynamics) are resolved, others (boundary layers, super-thin sheets) are not.

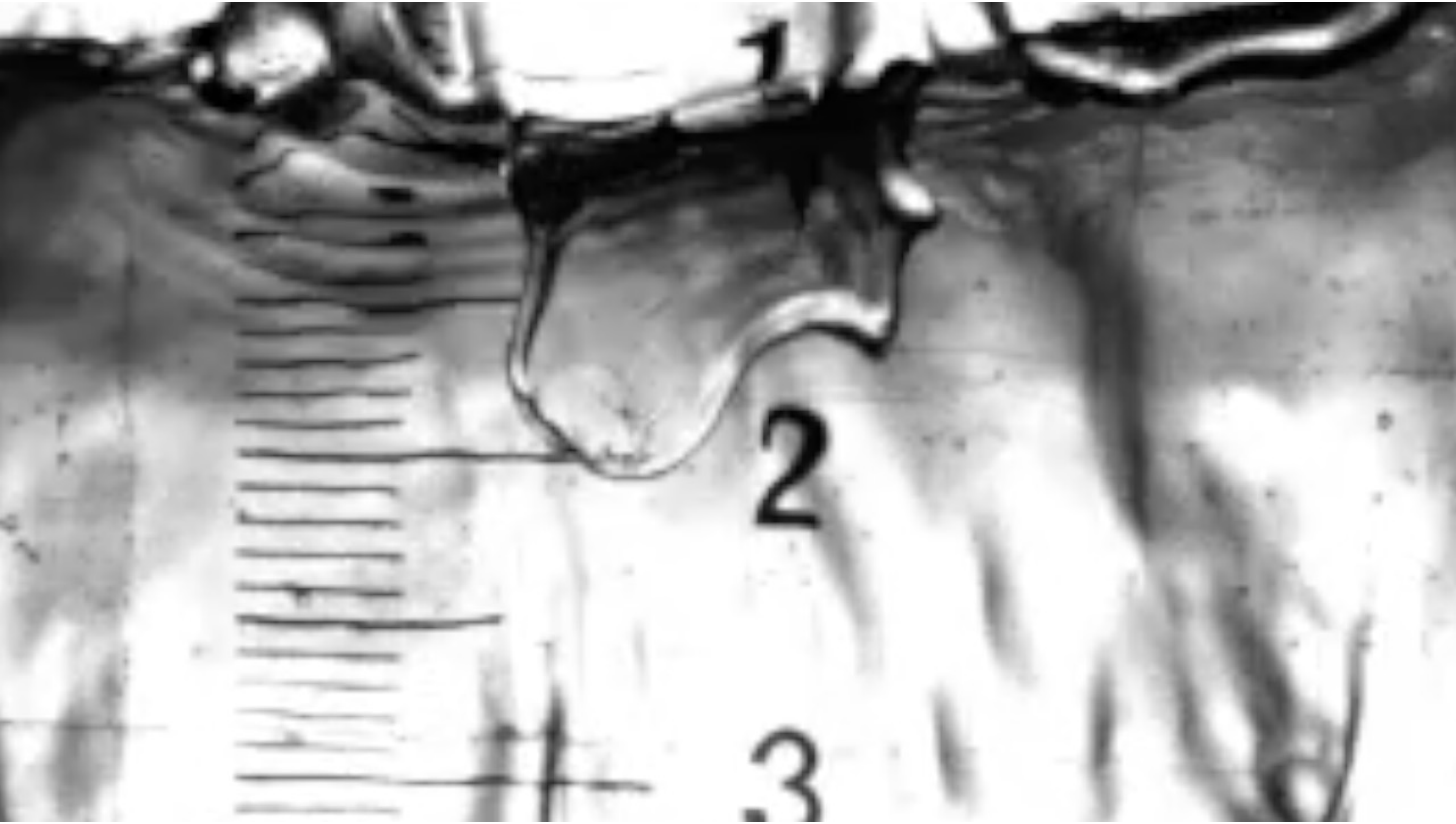
2) What can we learn from these very detailed simulations ?

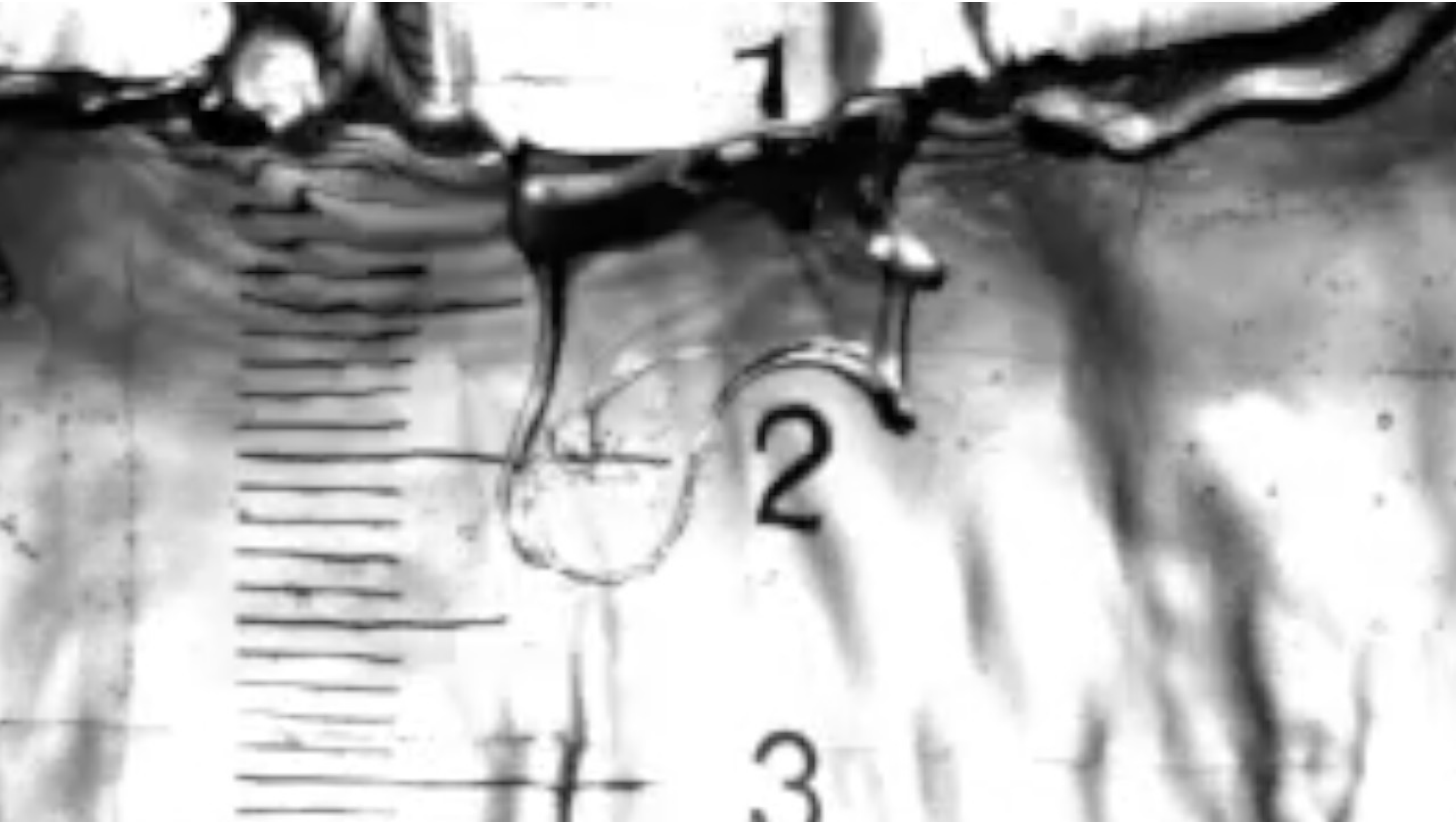
- How to look at the experiment again (mechanisms much more complex than expected).

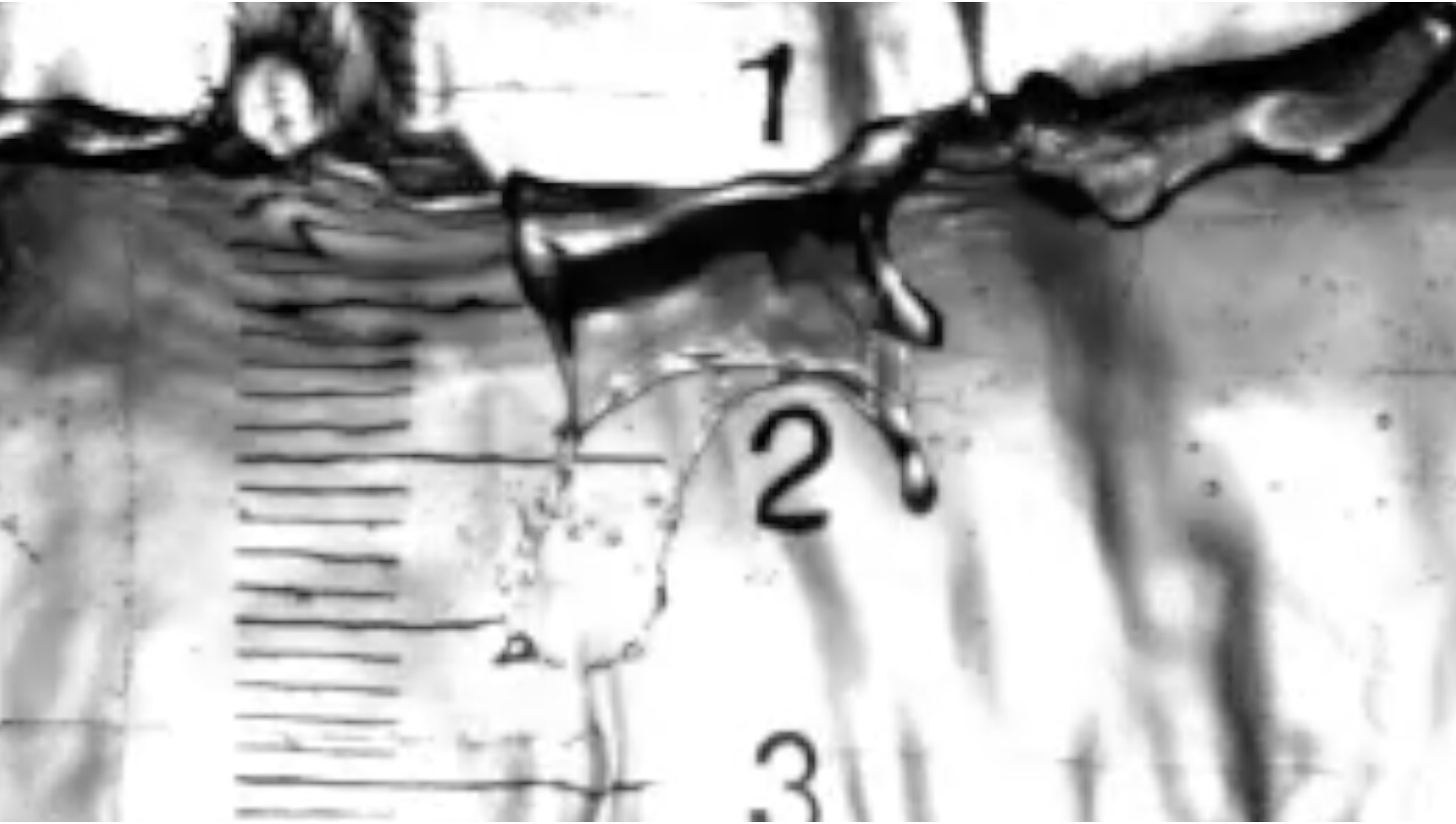

0
1
3
4
7
8


2
3


1
2
3

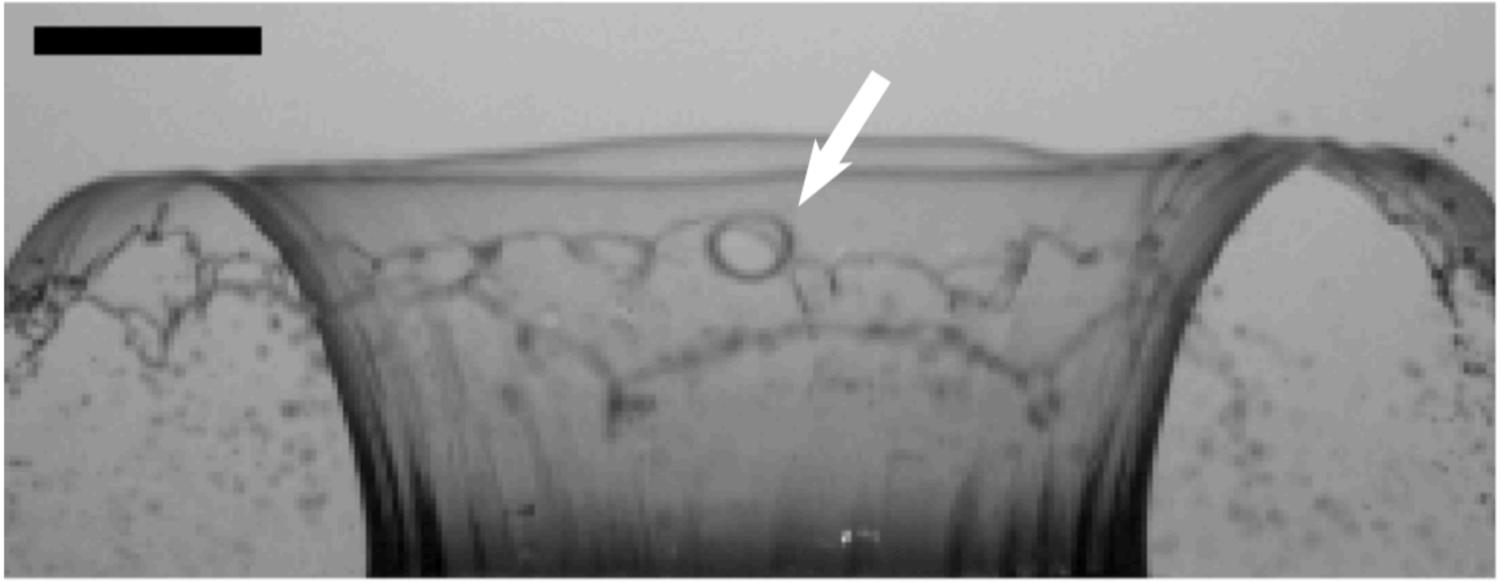

1
2
3


1
2
3


1
2
3

Marston et al JFM 2016

(*a*)



(*b*)

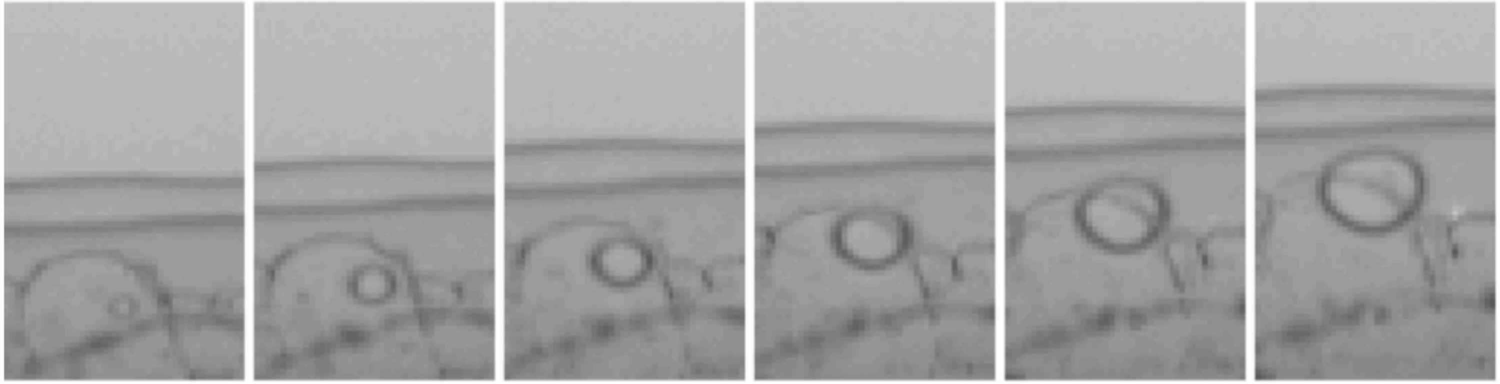


FIGURE 17. Example of hole formation in the crown used to estimate the wall thickness. $D_0 = 10$ mm, $V_0 = 10$ m s^{-1}, $P = 101$ kPa. The image in (*a*) (scale bar is 5 mm) is taken at $t = 1.9$ ms after impact whilst the sequence (*b*) shows the opening with time intervals of 67 μs. The opening speed in this realization is $V_{open} = 3$ m s^{-1} giving $\delta \approx 16$ μm.

Parellisation and Octree: new « basilisk » code by Stéphane Popinet.
Example: CORIA – Berlemont jet,
Resolution 4096^3 on 1536 cores ~ 10^4 hours (less than 5 hours wall clock time !!!)

Investigate the advantage of octree meshes:

Compression ratio is

$$C = \frac{\text{number of actual octree cells}}{\text{maximum number of cells}}$$

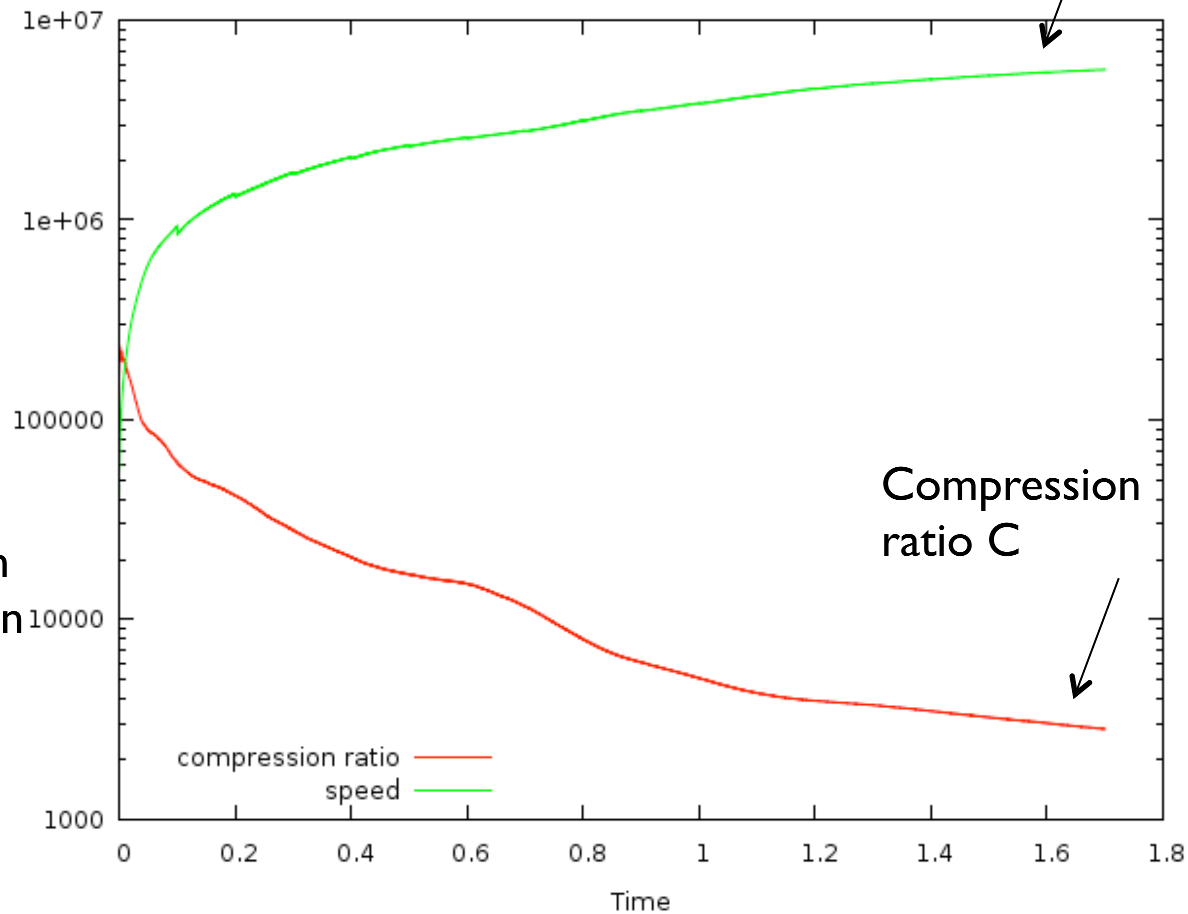
The effective size of 4096^3 is obtained at the maximum number of cells for a given smallest cell size.

The cell rate is the number of actual cells per second computed

$$R = \frac{\text{number of actual octree cells computed}}{\text{elapsed time}}$$

Cell rate R

Speed:
Cells/second
(green) going up in time:
cores work more efficiently



Compression ratio C

Fully resolved grid to octree compression ratio (red), going down in time: adaptation has less effect as more droplets/ sheets / ligaments form and require cells.

raw speed Z = RxC/np ~ 10^7

Cavitating
bubble

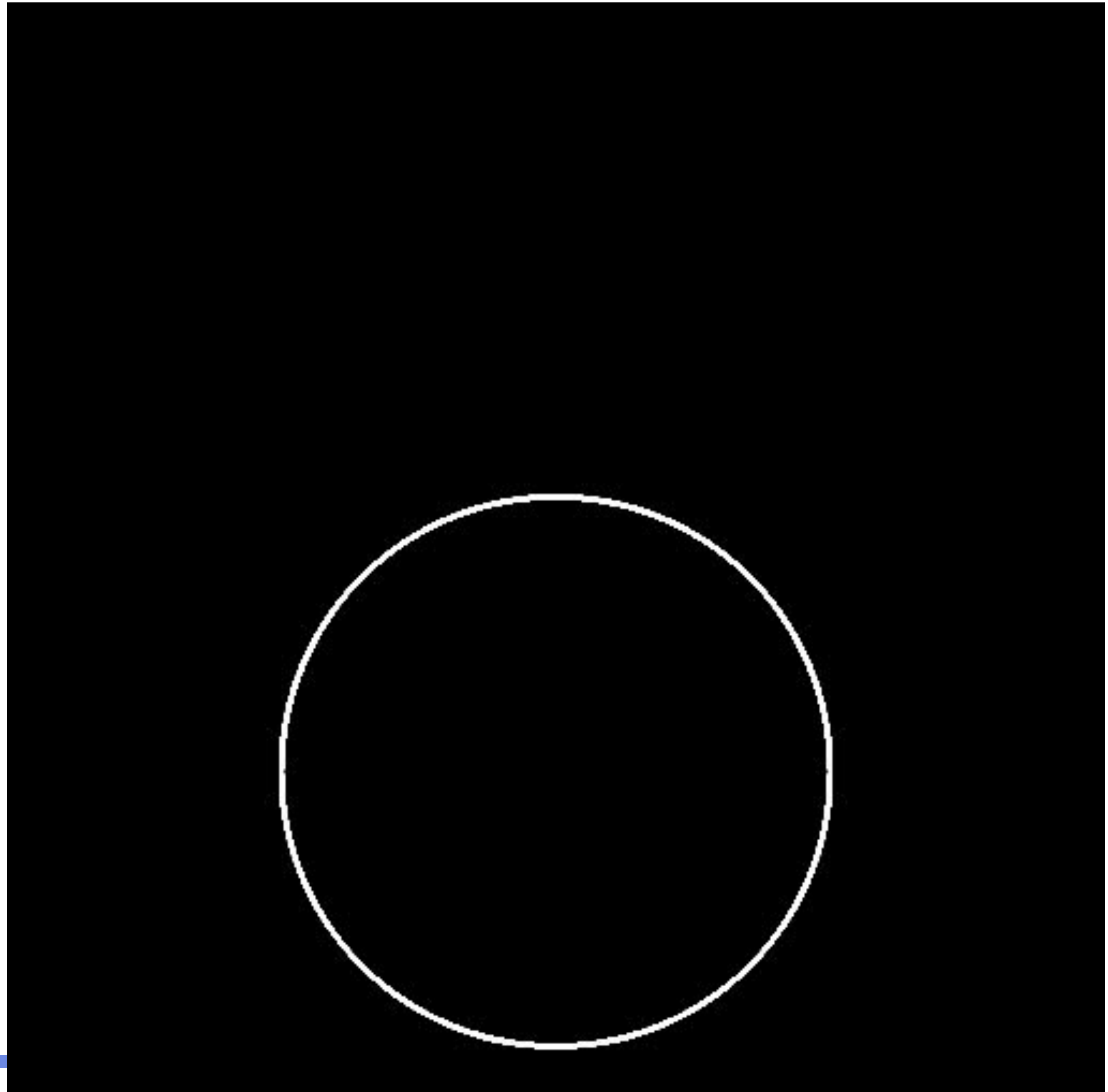
density ratio
1000

basilisk code

all-mach-number
scheme
VOF

Daniel Fuster
Stephane Popinet

2015



Thank you

Collaborators past and present (on this and related topics, somewhat arbitrary. In red: present at ICTAM)

Jie Li, Denis Gueyffier, Phil Yecko, Thomas Boeck, Jose-Maria Fullana, Ruben Scardovelli, Christophe Josserand, Stéphane Popinet, Pascal Ray, Luis Lemoyne, Shahriar Afkhami, Gaurav Tomar, Daniel Fuster, Anne Bagué, Jérôme Hoepffner, Gilou Agbaglah, Joseph John Soundar Jerôme, Arup K. Das, Bertrand Lagrée, Zhen Jian, Gounsetti Pare, Tomas Arrufat, Peng Cheng, Gretar Tryggvason, An-Bang Wang

Current Students and postdocs

Leon Malan, Yue Stanley Ling, Wojciech Aniszewski, Alexandre Guion, Pei-Hsun Tsai.

The end