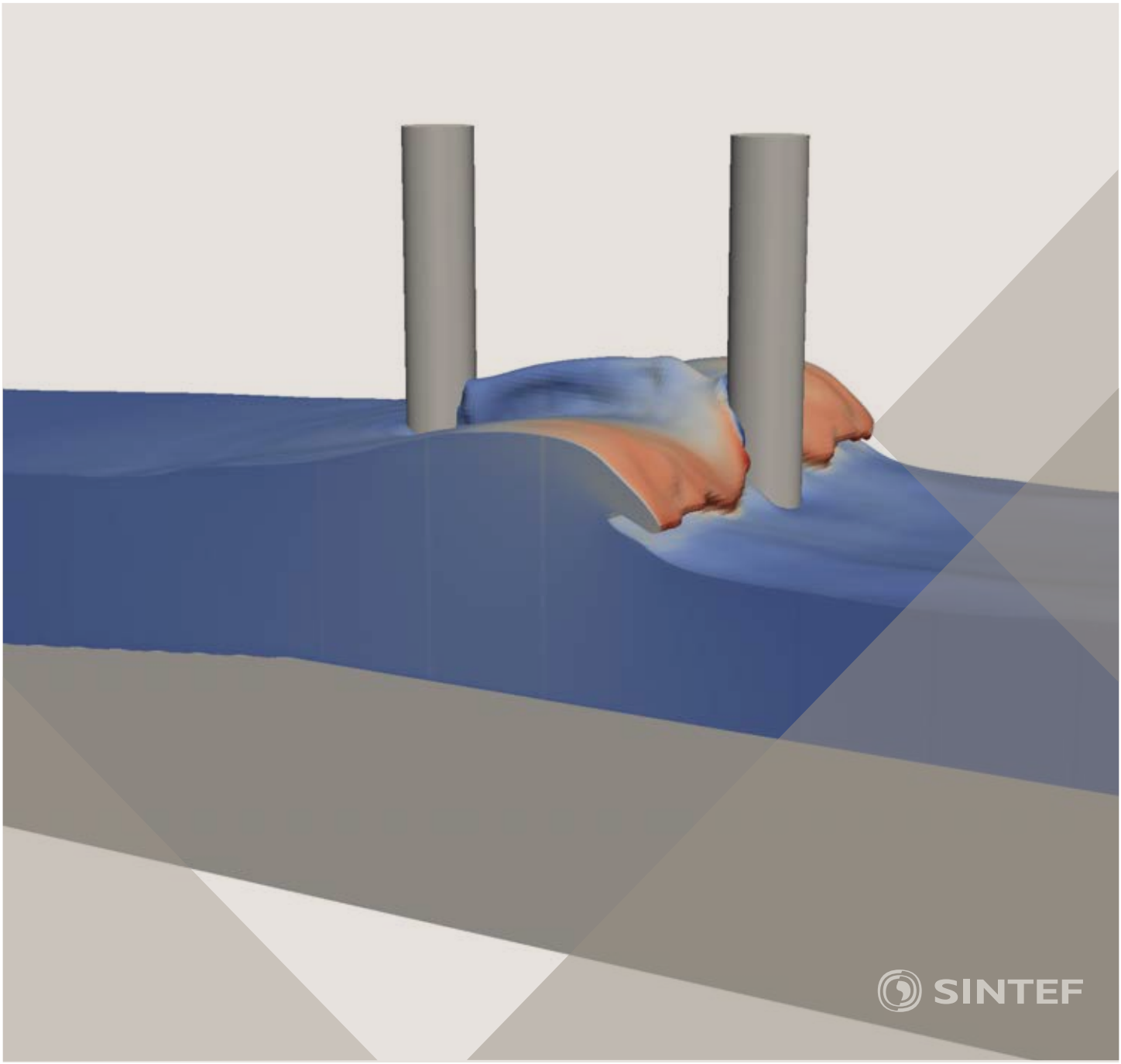
SINTEF
PROCEEDINGS
2

Proceedings of the 12th International Conference on Computational Fluid Dynamics in the Oil & Gas, Metallurgical and Process Industries

# Progress in Applied CFD – CFD2017



SINTEF Proceedings

Editors:
Jan Erik Olsen and Stein Tore Johansen

# Progress in Applied CFD – CFD2017

Proceedings of the 12th International Conference on Computational Fluid Dynamics in the Oil & Gas, Metallurgical and Process Industries

SINTEF Academic Press

SINTEF Proceedings no 2
Editors: Jan Erik Olsen and Stein Tore Johansen
**Progress in Applied CFD – CFD2017**
Selected papers from 10th International Conference on Computational Fluid Dynamics in the Oil & Gas, Metallurgical and Process Industries

Key words:
CFD, Flow, Modelling

Cover, illustration: Arun Kamath

ISSN 2387-4295 (online)
ISBN 978-82-536-1544-8 (pdf)



SINTEF Academic Press
Address: Forskningsveien 3 B
PO Box 124 Blindern
N-0314 OSLO
Tel: +47 73 59 30 00
Fax: +47 22 96 55 08

www.sintef.no/byggforsk
www.sintefbok.no

**SINTEF Proceedings**
SINTEF Proceedings is a serial publication for peer-reviewed conference proceedings on a variety of scientific topics.
The processes of peer-reviewing of papers published in SINTEF Proceedings are administered by the conference organizers and proceedings editors. Detailed procedures will vary according to custom and practice in each scientific community.

## PREFACE

This book contains all manuscripts approved by the reviewers and the organizing committee of the 12th International Conference on Computational Fluid Dynamics in the Oil & Gas, Metallurgical and Process Industries. The conference was hosted by SINTEF in Trondheim in May/June 2017 and is also known as CFD2017 for short. The conference series was initiated by CSIRO and Phil Schwarz in 1997. So far the conference has been alternating between CSIRO in Melbourne and SINTEF in Trondheim. The conferences focuses on the application of CFD in the oil and gas industries, metal production, mineral processing, power generation, chemicals and other process industries. In addition pragmatic modelling concepts and bio-mechanical applications have become an important part of the conference. The papers in this book demonstrate the current progress in applied CFD.

The conference papers undergo a review process involving two experts. Only papers accepted by the reviewers are included in the proceedings. 108 contributions were presented at the conference together with six keynote presentations. A majority of these contributions are presented by their manuscript in this collection (a few were granted to present without an accompanying manuscript).

The organizing committee would like to thank everyone who has helped with review of manuscripts, all those who helped to promote the conference and all authors who have submitted scientific contributions. We are also grateful for the support from the conference sponsors: ANSYS, SFI Metal Production and NanoSim.

Stein Tore Johansen & Jan Erik Olsen



Organizing committee:

Conference chairman: Prof. Stein Tore Johansen
Conference coordinator: Dr. Jan Erik Olsen
Dr. Bernhard Müller
Dr.Sigrid Karstad Dahl
Dr.Shahriar Amini
Dr.Ernst Meese
Dr.Josip Zoric
Dr.Jannike Solsvik
Dr.Peter Witt

Scientific committee:

Stein Tore Johansen, SINTEF/NTNU
Bernhard Müller, NTNU
Phil Schwarz, CSIRO
Akio Tomiyama, Kobe University
Hans Kuipers, Eindhoven University of Technology
Jinghai Li, Chinese Academy of Science
Markus Braun, Ansys
Simon Lo, CD-adapco
Patrick Segers, Universiteit Gent
Jiyuan Tu, RMIT
Jos Derksen, University of Aberdeen
Dmitry Eskin, Schlumberger-Doll Research
Pär Jönsson, KTH
Stefan Pirker, Johannes Kepler University
Josip Zoric, SINTEF

## CONTENTS

# PARALLEL MULTIPHASE FLOW SOFTWARE FOR SOLVING THE NAVIER-STOKES EQUATIONS

**Fahad S. AL-RASHED**[1*], **Maher M. SHARIFF**[1†]

[1]Saudi Aramco, Dhahran, SAUDI ARABIA

[*] E-mail: fahad.rashed@aramco.com
[†] E-mail: maher.shariff@aramco.com

## ABSTRACT

A code based on finite element method was built and applied on the variable density incompressible Navier-Stokes equations for accurately simulating immiscible two phase flows. The algorithm simulates the interface between the two liquid phases with high accuracy; it utilizes both the level-set method with a third order strong stability property Runge-Kutta (SSPRK) time integrator and a second-order projection method for the momentum equation. The solver developed is based on deal.II, an open source framework code. Numerical assessments on the transport and momentum equations are presented to verify the code accuracy. Nonconforming manufactured solutions are shown to produce the expected convergence rate of the used numerical schemes. Simulation of classical Rayleigh-Taylor instability was carried out and shown to match those in the published work.

**Keywords:** CFD, projection methods, level set, LES .

## NOMENCLATURE

*Greek Symbols*

| | |
|---|---|
| $\rho$ | Density |
| $\mu$ | Dynamic viscosity |
| $\nu$ | Kinematic viscosity |
| $\phi$ | Level set |

*Latin Symbols*

| | |
|---|---|
| $\boldsymbol{u}$ | Velocity |
| $p$ | Pressure |
| $t$ | Time |
| $V$ | An appropriate space with proper boundary conditions |

All symbols are non-dimensional.

## METHOD

The *variable density incompressible Navier-Stokes equations* are defined as follows:

$$\partial_t \rho + \mathrm{div}(\rho \boldsymbol{u}) = 0, \qquad \text{in } \Omega \times (0,T], \quad (1)$$

$$\rho[\partial_t \boldsymbol{u} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u}] - 2\mu \mathrm{div}(\nabla^s \boldsymbol{u}) + \nabla p = \rho \boldsymbol{f}, \qquad \text{in } \Omega \times (0,T], \quad (2)$$

$$\mathrm{div}(\boldsymbol{u}) = 0, \qquad \text{in } \Omega \times (0,T], \quad (3)$$

where $\Omega \subset \mathbb{R}^{2,3}$ and $\partial\Omega$ is the boundary, $\rho(\boldsymbol{x},t)$ is the density at $(\boldsymbol{x},t) \in \Omega \times [0,T]$, $\boldsymbol{u}(\boldsymbol{x},t)$ is the velocity vector field, $\mu$ is the dynamic viscosity, and $p(\boldsymbol{x},t)$ is the pressure. **Bold** variables are vector valued. Equation (1) is referred to as the transport equation, (2) is the momentum equation and (3) is the incompressibility constraint.

### Transport equation weak formulation

The weak formulation for the transport equation is: Find $\rho(\boldsymbol{x},t) \in V(\Omega)$ such that:

$$\int_\Omega v\left(\frac{\partial}{\partial t}\rho + \boldsymbol{u}\cdot\nabla\rho\right)\mathrm{dx} = 0, \qquad \forall v \in V(\Omega), \quad (4)$$

$$\rho(\boldsymbol{x},t) = \rho_{\partial\Omega}, \qquad \text{in } \partial\Omega_-, \quad (5)$$

$$\rho(\boldsymbol{x},0) = \rho_0, \qquad \rho_0 > 0, \quad (6)$$

where $V$ is an appropriate space for the transport equation with appropriate boundary conditions.
To accurately capture the density field $\rho$, we choose the 3rd order time integration method "Strong Stability Preserving Runge-Kutta" with three steps (abbreviated as SSPRK(3,3)) as described by (Gottlieb, 2005). The SSPRK(3,3) steps are:

$$y^{(1)} = y^k + \Delta t f(t^k, y^k), \quad (7)$$

$$y^{(2)} = \frac{1}{4}\left(3y^k + y^{(1)} + \Delta t f(t^{k+1}, y^{(1)})\right), \quad (8)$$

$$y^{k+1} = \frac{1}{3}\left(y^k + 2y^{(2)} + 2\Delta t f(t^{k+\frac{1}{2}}, y^{(2)})\right). \quad (9)$$

The strong stability preserving property is $\|y^{k+1}\| \leq \|y^k\|$. This makes it attractive in the transport equation case. The SSP property comes from the maximum principle preserving property of the Forward Euler method.

### The Level Set Model

The fluid mixture we are interested in modeling with the transport equations has two phases: oil, and water. Each has a different density value $\rho$. Since they do not mix, it is important that each phase must be distinct when modeled and the volume of each phase in $\Omega$ be conserved. Otherwise, the incompressibility condition $\mathrm{div}(\boldsymbol{u}) = 0$ will be violated. As a consequence, when solving the approximation of the transport equation, one needs to make sure the interface between two phases is tracked with enough accuracy. There are many methods to achieve such accuracy, which can be divided into two classes. In the first one, the interface is implicitly tracked by a function defined on the whole domain. Such methods include the level set method, and volume of fluid method. In

the second class, the interface is explicitly tracked with front-tracking methods. We will use the level set method between two phases.

The level set method was first introduced by (Osher and Sethian, 1988) to evolve the interface with speeds depending on the curvature of a given velocity field. The interface is tracked with a function $\Phi(\boldsymbol{x})$ to represent the $n-1$ dimensional interface $\Gamma \subset \Omega$ separating $\Omega$ into two phases $\Omega_1$ and $\Omega_2$. There are many ways to define $\Gamma$ but we are going to use the tanh function with the interface at $\Phi(\boldsymbol{x}) = 0.5$. The tanh function is defined as:

$$\Phi(\boldsymbol{x}) := \frac{1}{2}\left(1+\tanh\left(\frac{d(\boldsymbol{x})}{\gamma}\right)\right), \tag{10}$$

where $d$ is a distance from the interface function and $\gamma$ controls how steep the interface is. To describe the evolution of an interface that is transported along with a fluid, we can use $\Phi$ is used instead of $\rho$ in (4):

$$\int_\Omega v\left(\frac{\partial}{\partial t}\Phi + \boldsymbol{u}\cdot\nabla\Phi\right)\mathrm{dx} = 0, \qquad \forall v \in V(\Omega), \tag{11}$$

$$\Phi(\boldsymbol{x},t) = \Phi_{\partial\Omega}, \qquad \text{in } \partial\Omega_-, \tag{12}$$

$$\Phi(\boldsymbol{x},0) = \Phi_0, \qquad \Phi_0 > 0. \tag{13}$$

This essentially transports the $\Phi$ function instead of the density $\rho$. To reconstruct $\rho$ from $\Phi$, we use the function $H(\Phi)$:

$$H(\Phi(\boldsymbol{x})) = \begin{cases} \rho_1, & \Phi(\boldsymbol{x}) < 0.5, \\ \rho_2, & \Phi(\boldsymbol{x}) \geq 0.5, \end{cases} \tag{14}$$

where $\rho_1$, $\rho_2$ are the densities of the fluids in $\Omega_1$ and $\Omega_2$ respectively ($\rho_1 < \rho_2$). $H(\Phi)$ will produce density fields that have discontinuous transitions between phases, which are undesirable when dealing with PDEs that expect smooth enough functions. There are many functions that create smoother transitions such as:

$$H(\Phi(\boldsymbol{x})) = \frac{\rho_2-\rho_1}{2} + \frac{\rho_2+\rho_1}{2}\tanh\left(\frac{\Phi(\boldsymbol{x})}{\gamma}\right), \tag{15}$$

where $\alpha$ controls how steep the transition between the two densities is. The advantage of this reconstruction is that it produces the closest density field close to (14) with some retained smoothness. Another candidate $H$ function is:

$$H(\Phi(\boldsymbol{x})) = (\rho_2-\rho_1)\Phi(\boldsymbol{x}) + \rho_1, \tag{16}$$

which is a linear scaling of the level set to the densities in $\Omega$. It is robust but translates the undesirable oscillations that extends beyond $\Phi(\boldsymbol{x}) > 1$ or $\Phi(\boldsymbol{x}) < 0$. This issue may be solved by clipping the reconstruction at a certain radius $\alpha$ around 0.5 ($0 \leq \alpha \leq 0.5$) :

$$H(\Phi(\boldsymbol{x})) = \begin{cases} \rho_1, \Phi(\boldsymbol{x}) \leq 0.5-\alpha, \\ \rho_2, \Phi(\boldsymbol{x}) \geq 0.5+\alpha, \\ \big(\Phi(\boldsymbol{x}) - (0.5-\alpha)\big)\dfrac{\rho_2-\rho_1}{2\alpha} + \rho_1, \text{otherwise.} \end{cases} \tag{17}$$

This reconstruction introduces relatively sharp changes in the density gradient and affects the stability of simulation runs. Finally, the last reconstruction we are going to introduce has the property of having slope zero at the $0.5 \pm \alpha$ points and being a transition polynomial of third degree ($0 \leq \alpha \leq 0.5$):

$$H(\Phi(\boldsymbol{x})) = \begin{cases} \rho_1, & \text{if } \Phi(\boldsymbol{x}) \leq 0.5-\alpha, \\ \rho_2, & \text{if } \Phi(\boldsymbol{x}) \geq 0.5+\alpha, \\ \dfrac{(4\alpha - 2\Phi(\boldsymbol{x}) + 1)(2\alpha + 2\Phi(\boldsymbol{x}) - 1)^2}{32\alpha^3} & \\ \qquad (\rho_2-\rho_1) + \rho_1, \text{ otherwise.} \end{cases} \tag{18}$$

Compared to the clipped reconstruction (17), the above has smooth gradient transitions and was found to have a stabilizing effect when used in the simulations below. This transition is comparable to the Heaviside function (14) in (Sussman and Fatemi, 1999) but has the advantage of being polynomial in nature.

## Entropy-Viscosity

The Entropy-Viscosity is (at least) a second-order stabilization term introduced by (Guermond *et al.*, 2011a) and (Guermond and Pasquetti, 2011). It has the advantage of having a less diffusive effect on the solution and thus allowing the construction of stabilized second order numerical schemes. Using the transport equation weak form:

$$\int_\Omega v_h\left(\frac{\partial \rho_h}{\partial t} + \boldsymbol{u}_h\cdot\nabla\rho_h - \mathrm{div}\,(\nu\nabla\rho)\right)\mathrm{dx} = 0, \quad \forall v_h \in V_h(\Omega), \tag{19}$$

and $\nu$ is calculated for each cell separately as follows. Define $E(\phi)$ as convex functions that satisfies the differential inequality:

$$\partial_t E(\phi) + \boldsymbol{u}\cdot\nabla E(\phi) < 0, \tag{20}$$

where $\phi$ is the level set function mentioned in section and $E(\phi)$ is the entropy function. For examples, one can use:

$$E(\phi) = \begin{cases} \frac{1}{p}(\phi-\frac{1}{2})^p \text{ where } p = 1,2,\ldots, \\ -\log(|\phi(1-\phi)| + 10^{-14}). \end{cases} \tag{21}$$

In the fully discretized setting, use $\phi^n, \phi^{n-1}$ and compute the following values for each quadrature points $q_k, q_f$ in cell $k$ and face $f$:

$$R^{n+1/2}(q_k) = \frac{\Pi_{\mathcal{T}_h} E(\phi^n) - \Pi_{\mathcal{T}_h} E\left(\phi^{n-1}\right)}{\Delta t} + \frac{1}{2}\left(\boldsymbol{u}^n\cdot\nabla\Pi_{\mathcal{T}_h} E(\phi^n) + \boldsymbol{u}^{n-1}\cdot\nabla\Pi_{\mathcal{T}_h} E\left(\phi^{n-1}\right)\right) \tag{22}$$

$$J^n(q_f) = u^n\cdot\boldsymbol{n}[\![\partial_n\Pi_{\mathcal{T}_h} E(\phi^n)]\!]|_f. \tag{23}$$

Then get the maximum $R_k^{n+1/2} = \max_{q_k\in k}\left|R^{n+1/2}(q_k)\right|$ and $J_k^n = \max_{f\in k}\max_{q_f\in f}\left|J^n(q_f)\right|$. Note that we are using the Crank-Nicolson scheme to calculate $R$ giving us a second order accurate value for $R$. The viscosity $\nu_k$ will then be:

$$\nu_k = \min\left(C_m h|\mathbf{u}|_{L^\infty}, C_e h^2 \frac{R^{n+1/2} + J_k^n}{\|E(\phi^n) - \overline{E(\phi^n)}\|_{L^\infty(\Omega)}}\right), \tag{24}$$

where $\overline{E(\phi)} = \frac{1}{|\Omega|}\int_\Omega E(\phi)$ and $\|E(\phi^n) - \overline{E(\phi^n)}\|_{L^\infty(\Omega)}$ is a normalization factor. The amount of artificial viscosity is proportional to the entropy production but bounded from above by the linear artificial viscosity. If the solution is smooth and entropy production is very small, little or no artificial viscosity is added. Some disadvantages remain such as coefficients $C_e, C_m$ to tune and the ambiguity of $h$.

### Compression for the Level Set

For the level set method to work, the curved shape of the level set function over the boundary must be maintained to prevent adding non-physical effects to the model. The stabilization viscosity diffuses the level set interface as the simulation marches in time. Consequently, with the presence of the diffusion term, we add the compression (or anti-diffusion) term $\text{div}\left(C_K\frac{\nu}{h}(1-\phi_h)\phi_h\frac{\nabla\phi_h}{|\nabla\phi_h|}\right)$ to the transport equation (4):

$$\int_{\Omega\times[0,T]}\frac{\partial}{\partial t}\phi_h+\boldsymbol{u}\cdot\nabla\phi_h - \text{div}\left(\nu\nabla\phi_h-C_K\frac{\nu}{h}\phi_h(1-\phi_h)\frac{\nabla\phi_h}{|\nabla\phi_h|}\right)dxdt=0, \quad (25)$$

where the level set $\phi\in[0,1]$ and defined at $\phi=0.5$. This compression term eliminates the need for a separate reinitialization step. In practice, it has been observed that the compression term in (25) induces "fingering" effect in simulations. It is the result of perturbations in the initial level set that the compression term gradually propagates resulting in the level set extending like fingers. To mitigate that, a smoothed out $\phi_h^*$ is used in the normal of the compression front $\frac{\nabla\phi_h}{|\nabla\phi_h|}$ where $\phi_h^*$ is the solution to $\phi_h^*-h^2\Delta\phi_h^*=\phi, \nabla\phi_h^*\cdot n=0$ on $\partial\Omega$. We will denote $S$ as the operator that maps $\phi$ to the corresponding $\phi^*$ (i.e. $S\phi_h=\phi_h^*$).
Let us detail the algorithm for solving (25):

1. Initialize the level set by normalizing the initial density scalar field.
$$\phi_h^0=\frac{\rho_h^0-\rho_{\min}}{\rho_{\max}-\rho_{\min}},$$
2. For each of the SSPRK(3,3) steps below, we need to solve the following:
$$L^n(\boldsymbol{u}_h,\phi_h,\phi_h^*)=-\boldsymbol{u}_h\cdot\nabla\phi_h - \text{div}\left(\nu\nabla\phi_h-C_k\frac{\nu}{h}\phi_h(1-\phi_h)\frac{\nabla\phi_h^*}{|\nabla\phi_h^*|}\right), \quad (26)$$
when solved for each of the three steps below, the values are
$$\phi_h^{(1)}=\phi_h^n+\Delta tL^n(\boldsymbol{u}_h^n,\phi_h^n,S\phi_h^n), \quad (27)$$
$$\phi_h^{(2)}=\frac{1}{4}\left(3\phi_h^n+\phi_h^{(1)} + \Delta tL^{n+1}(2\boldsymbol{u}_h^n-\boldsymbol{u}_h^{n-1},\phi_h^{(1)},S\phi_h^{(1)})\right), \quad (28)$$
$$\phi_h^{n+1}=\frac{1}{3}\left(\phi_h^n+2\phi_h^{(2)} + 2\Delta tL^{n+\frac{1}{2}}\left(\frac{1}{2}\left[3\boldsymbol{u}_h^n-\boldsymbol{u}_h^{n-1}\right],\phi_h^{(2)},S\phi_h^{(2)}\right)\right). \quad (29)$$
3. Lastly, we "denormalize" the level set with a reconstruction function such as:
$$\rho_h^{n+1}=H(\phi_h)(\rho_{\max}-\rho_{\min})+\rho_{\min}. \quad (30)$$

It is worth mentioning that when the entropy-viscosity vanishes in well resolved regions of the solution, the compression stops working and the sharpness of the level set interface is lost. This may be remedied by using some "antivanish" viscosity $\nu_{\text{antivanish}}=\nu+\nu_\varepsilon$ where $\nu_\varepsilon$ is a small positive amount of viscosity that maintains the balance between diffusion and compression and, thus, maintains the slope of the level set.

### Projection method for the momentum equation

Initialize the algorithm with $\rho^0=\rho_0,\boldsymbol{u}^0=\boldsymbol{u}_0,p^0=p_0,\varphi^0=q^0=0$ then proceed as follows:

1. Setup intermediate variables:
$$\begin{aligned}\rho^* &= \rho^{n+1}+\frac{1}{6}\text{BDF}_2(\rho^{n+1}),\\ &\quad \text{where BDF}_2(\phi^{n+1})=3\phi^{n+1}-4\phi^n+\phi^{n-1},\\ p^* &= p^n+\frac{1}{3}\left(4\delta\psi^n-\delta\psi^{n-1}\right),\\ \boldsymbol{u}^* &= 2\boldsymbol{u}^n-\boldsymbol{u}^{n-1}.\end{aligned}$$
2. Prediction:
$$\frac{3\rho^*\boldsymbol{u}^{n+1}-4\rho^{n+1}\boldsymbol{u}^n+\rho^{n+1}\boldsymbol{u}^{n-1}}{2\Delta t}-\rho^{n+1}\boldsymbol{u}^*\cdot\nabla\boldsymbol{u}^{n+1} + \frac{1}{2}\text{div}\left(\rho^{k+1}\boldsymbol{u}^*\right)\boldsymbol{u}^{n+1}-\mu\Delta\boldsymbol{u}^{n+1}+\nabla p^\star=\rho^{n+1}\boldsymbol{f}^{n+1}, \quad \boldsymbol{u}^{n+1}\big|_{\partial\Omega}=0,$$
3. Projection:
$$\begin{aligned}\Delta\delta\psi^{n+1} &= \frac{3\rho_{\min}}{2\Delta t}\text{div}\,(\boldsymbol{u}^{k+1}), \quad \partial_n\delta\psi^{n+1}=0,\\ \delta q^{n+1} &= -\text{div}\,(\boldsymbol{u}^{n+1}),\end{aligned}$$
4. Pressure correction: $p^{n+1}=\psi^{n+1}-\mu q^{n+1}$.

where BDF stands for Backwards Difference Formula.
This variable density projection method is shown to have a error of $O(\Delta t^2)$ in the $L^2$ norm. The stability proof can be found in (Guermond and Salgado, 2011, §5.4)

### Large Eddy Simulation

The Large Eddy Simulation (LES) is based on the – at least – 2nd order entropy-viscosity method (Guermond *et al.*, 2011b). The concept behind LES is separating the flow into large – or resolved – and small – or subgrid – scales. For a good overview of LES, see (John, 2004).
When dealing with the Navier-Stokes equations, LES is added as a cell-wise viscosity $\nu_K\geq 0$ to the term $-2\nu\text{div}\,(\nabla^s\boldsymbol{u})$. The result is a viscosity $\nu+\nu_K$. The classical Smagorinsky model ((Smagorinsky, 1963)) uses:

$$\nu_K:=C_s\delta_K^2\|\nabla^s\boldsymbol{u}\|,$$

where $C_s$ is the Smagorinsky constant and $\delta_k$ is the width of the filter (which is proportional to $h_K$). (Guermond *et al.*, 2011c) proposed the following Entropy-Viscosity approach:

$$\nu_K:=\min\left(C_mh_K|\boldsymbol{u}|,C_eh_K^2\frac{|D_h(\boldsymbol{x},t)|}{\|\boldsymbol{u}_h^2\|_{L^\infty(\Omega)}}\right),$$

where

$$D_h(\boldsymbol{x},t):=\partial_t\left(\frac{1}{2}\boldsymbol{u}_h^2\right)+\text{div}\left(\left(\frac{1}{2}\boldsymbol{u}_h^2+p_h\right)\boldsymbol{u}_h\right)-Re^{-1}\Delta\left(\frac{1}{2}\boldsymbol{u}_h^2\right) + Re^{-1}\left(\nabla\boldsymbol{u}_h\right)^2-\mathbf{f}\cdot\boldsymbol{u}_h \quad (31)$$

where $h_K$ is the mesh size locally, $\|\boldsymbol{u}_h^2\|_{L^\infty(\Omega)}$ is a normalizing term, and $C_m,C_e$ are appropriate constants. The first term $C_mh_K|\boldsymbol{u}|$ is the first order artificial viscosity. When the mesh is fine enough to simulate all the scales, $h_K^2|D_h(\boldsymbol{x},t)|$ is much smaller than the first-order artificial viscosity. This makes $\nu_K$ a consistent viscosity that vanishes when scales of all levels are resolved.

## NUMERICAL RESULTS

We test the schemes discussed in the previous sections numerically and present them here.

### Validation

Here, we will present the validation of the projection method with density $\rho(\boldsymbol{x},t) = 1$. This is a constant density test performed on a variable density equation for validation purposes. Using $\Omega = (0,1)^d$ domain with a uniform mesh and cell-wise $[\mathbb{Q}_2]^d$ continuous finite elements, we introduce the following simple linear polynomial manufactured solution for the momentum equation:

$$\boldsymbol{u}(\boldsymbol{x},t) = (1+t)\begin{pmatrix} x+y \\ x-y \end{pmatrix}, \quad p(\boldsymbol{x},t) = (1+t)xy, \tag{32}$$

$$\boldsymbol{u}(\boldsymbol{x},t) = (1+t)\begin{pmatrix} 1+z \\ 1+x \\ 1+y \end{pmatrix}, \quad p(\boldsymbol{x},t) = (1+t)xyz. \tag{33}$$

for $d = 2,3$ respectively. We solve the equation (2) with $\mu = 1$ running until final time $T = 1$. The projection step is disabled, which means that the exact pressure is interpolated every time step. The boundary condition $\boldsymbol{u}|_{\partial\Omega} = \boldsymbol{u}(\boldsymbol{x},t)|_{\partial\Omega}$ is enforced. The time step is changed to roughly achieve a Courant-Friedrichs-Lewy condition (CFL) of 0.25. As expected, table 1 shows that the error is machine epsilon which means that the algorithm reproduces the conforming manufactured solutions exactly.

| | cells | $\boldsymbol{u}_{\text{dofs}}$ | $\Delta t$ | $\|e_{\boldsymbol{u}}\|_{L2}$ | $\|e_{\boldsymbol{u}}\|_{H1}$ | $\text{CFL}_{\text{max}}$ |
|---|---|---|---|---|---|---|
| | 16 | 162 | 8E-03 | 3E-13 | 1E-12 | 0.2621 |
| 2D | 64 | 578 | 4E-03 | 1E-15 | 2E-14 | 0.2606 |
| | 256 | 2178 | 2E-03 | 1E-14 | 8E-14 | 0.2607 |
| | 8 | 375 | 3E-02 | 9E-16 | 1E-14 | 0.2601 |
| 3D | 64 | 2187 | 1E-02 | 5E-15 | 3E-14 | 0.2614 |
| | 512 | 14739 | 7E-03 | 7E-15 | 7E-14 | 0.2613 |

**Table 1:** Error values for running conforming manufactured solutions in a unit cube. We get the expected value of machine epsilon.

### Projection Scheme

Using $\Omega = (0,1)^d$ domain with a uniform mesh and cell-wise $[\mathbb{Q}_2]^d/\mathbb{Q}_1$ Taylor-Hood continuous finite elements, we introduce the following simple linear polynomial manufactured solution for the momentum equation:

$$\boldsymbol{u}(\boldsymbol{x},t) = (1+t)\begin{pmatrix} x+y \\ x-y \end{pmatrix}, \quad p(\boldsymbol{x},t) = (1+t)xy, \tag{34}$$

$$\boldsymbol{u}(\boldsymbol{x},t) = (1+t)\begin{pmatrix} 1+z \\ 1+x \\ 1+y \end{pmatrix}, \quad p(\boldsymbol{x},t) = (1+t)xyz. \tag{35}$$

with $d = 2,3$ respectively. We solve the equation (2) with $\mu = 1$ running until final time $T = 1$. The projection step is disabled, which means that the exact pressure is interpolated from the exact solution to the discrete space every time step. We enforce the following boundary condition $\boldsymbol{u}|_{\partial\Omega} = \boldsymbol{u}(\boldsymbol{x},t)|_{\partial\Omega}$. The source term is modified to reflect the exact solutions. As expected, table 2 shows that the error is machine epsilon ($\sim 0$), which means that the algorithm reproduces the conforming manufactured solutions exactly.

| | cells | $\boldsymbol{u}_{\text{dofs}}$ | $\Delta t$ | $\|e_{\boldsymbol{u}}\|_{L2}$ | $\|e_{\boldsymbol{u}}\|_{H1}$ |
|---|---|---|---|---|---|
| | 16 | 162 | 1E-02 | 8E-16 | 1E-14 |
| 2D | 64 | 578 | 5E-03 | 6E-15 | 4E-14 |
| | 256 | 2178 | 3E-03 | 2E-14 | 1E-13 |
| | 8 | 375 | 2E-02 | 1E-15 | 1E-14 |
| 3D | 64 | 2187 | 1E-02 | 3E-15 | 3E-14 |
| | 512 | 14739 | 5E-03 | 9E-15 | 8E-14 |

**Table 2:** Error values for running conforming manufactured solutions in a unit cube. We get a machine epsilon as expected.

Now, we validate the scheme by running a convergence rate test. We use the same 2D setup as before with the following nonconforming manufactured solutions:

$$\boldsymbol{u}(\boldsymbol{x},t) = \begin{pmatrix} \cos(x)+\cos(y+t) \\ \sin(x)+\sin(y+t) \end{pmatrix},$$
$$p(\boldsymbol{x},t) = \cos(x+y+t). \tag{36}$$

We see in table 3 that we get the $O(\Delta t^2)$ in the $L^2$ norm as expected. The $H^1$ norms are a bit higher than the expected $O(\Delta t^{\frac{3}{2}})$.

| $\boldsymbol{u}_{\text{dofs}}$ | $\Delta t$ | $\|e_{\boldsymbol{u}}\|_{L2}$ | rate | $\|e_{\boldsymbol{u}}\|_{H1}$ | rate |
|---|---|---|---|---|---|
| 4802 | 2E-02 | 1.54E-04 | - | 1.04E-03 | - |
| 18818 | 1E-02 | 4.28E-05 | 1.85 | 3.11E-04 | 1.75 |
| 74498 | 5E-03 | 1.14E-05 | 1.9 | 9.01E-05 | 1.79 |
| 296450 | 2.5E-03 | 2.98E-06 | 1.94 | 2.57E-05 | 1.81 |
| $\boldsymbol{u}_{\text{dofs}}$ | $\Delta t$ | $\|e_p\|_{L2}$ | rate | $\|e_p\|_{H1}$ | rate |
| 4802 | 2E-02 | 1.37E-03 | - | 2.22E-02 | - |
| 18818 | 1E-02 | 4.10E-04 | 1.74 | 8.63E-03 | 1.36 |
| 74498 | 5E-03 | 1.18E-04 | 1.8 | 3.27E-03 | 1.4 |
| 296450 | 2.5E-03 | 3.31E-05 | 1.83 | 1.22E-03 | 1.42 |

**Table 3:** Convergence rate for the constant density projection method. The $\text{CFL}_{\text{max}}$ is at 0.64.

### Realistic Models

In this section, we will study the applications of variable density projection scheme on a more realistic model; the Rayleigh-Taylor instability test. We compare our results with the work of (Guermond *et al.*, 2011a). Specifically in the early times before turbulent behavior.

### Rayleigh-Taylor Instability

We now apply the method to a more realistic problem. We use the Rayleigh-Taylor instability test that (Tryggvason, 1988) used. Two fluids are initially at rest in the 2D domain $(-d/2, d/2) \times (-2d, 2d)$ and the heavier fluid is on top. The transition of the phase-field variable $\rho$ is as follows:

$$\rho(x,y,t=0) = \frac{\rho_{\max}+\rho_{\min}}{2} + \frac{\rho_{\max}-\rho_{\min}}{2}\tanh\left(\frac{y+\mu(x)}{\alpha d}\right), \tag{37}$$

where $\alpha \approx 0.04$ and the initial interface is slightly perturbed as follows:

$$\mu(x) = 0.1\cos(2\pi x/d). \tag{38}$$

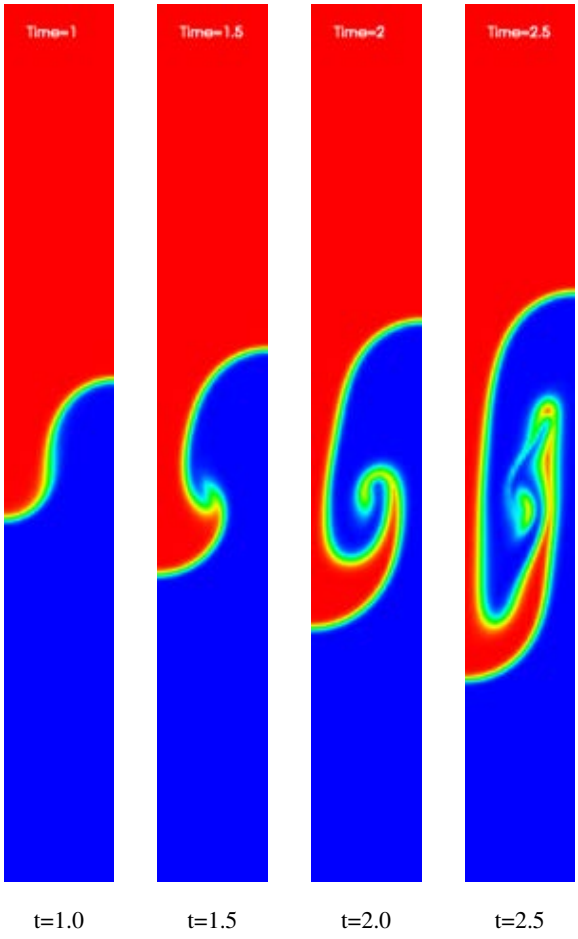



**Figure 1:** The Rayleigh-Taylor instability with density ratio of 3.

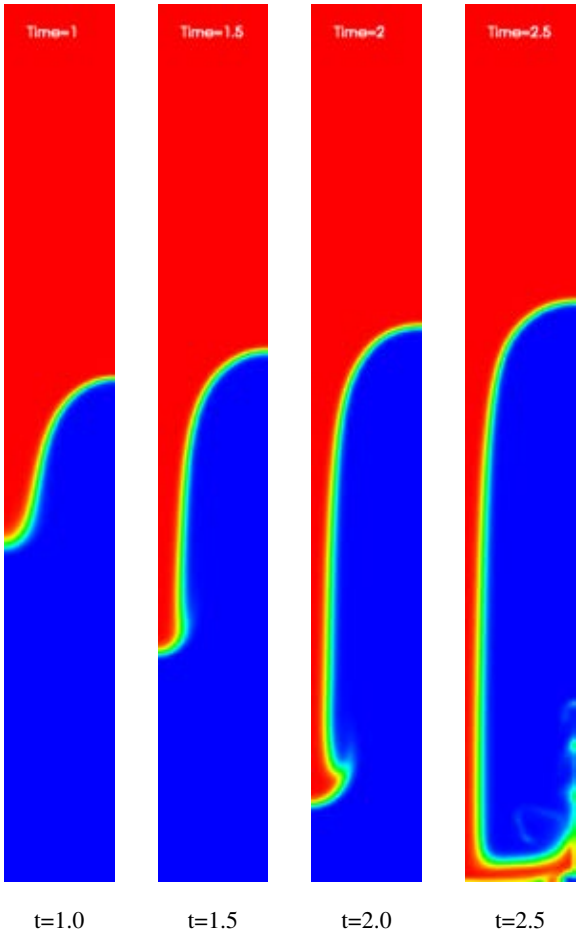



**Figure 2:** The Rayleigh-Taylor instability with density ratio of 100.

The time is also scaled using the Atwood number in Tryggvason as $t_{\text{Tryg}} = t\sqrt{A_t}$

$$A_t = \frac{\rho_0^{\max} - \rho_0^{\min}}{\rho_0^{\max} + \rho_0^{\min}}, \tag{39}$$

where $\rho_0^{\max} := \max_{\boldsymbol{x}\in\Omega}\rho_0(\boldsymbol{x})$ and $\rho_0^{\min} := \min_{\boldsymbol{x}\in\Omega}\rho_0(\boldsymbol{x})$. As the system progresses at $t > 0$, the heavy fluid will fall into the lighter fluid as a result of having the momentum equation gravity source term is $\rho\boldsymbol{g}$

We non-dimensionalize the equations as follows. We divide by: $\rho_0^{\min}$ for the density $\rho$, $d$ for length, and $d^{1/2}/|\boldsymbol{g}|^{1/2}$ for time. Consequently, $d^{1/2}|\boldsymbol{g}|^{1/2}$ is the velocity reference and the Reynolds number is $Re = \rho_0^{\min} d^{1/2}|\boldsymbol{g}|^{1/2} d/\mu$. We will restrict ourselves to the domain $(0, d/2)\times(-2d, 2d)$ because we assume that the symmetry of the initial setup continues as time progresses. The top and bottom parts have no-slip boundary conditions and the left and right sides have $\boldsymbol{u}\cdot\boldsymbol{n} = 0$, $(I - \boldsymbol{n}\otimes\boldsymbol{n})\nu\nabla\boldsymbol{u} = 0$ boundary conditions (known as symmetry or free boundary conditions).

*Remark.* Note that we must integrate the pressure term by parts in the weak form for $p$ to be in $L^2$. In this experiment, we tested both integrating by parts and leaving the pressure term as is. This leads to different boundary conditions for each case: $(I - \boldsymbol{n}\otimes\boldsymbol{n})(\nu\nabla\boldsymbol{u} - Ip) = 0$, and $(I - \boldsymbol{n}\otimes\boldsymbol{n})\nu\nabla\boldsymbol{u} = 0$ respectively. In this experiment, both were numerically stable and gave almost exactly the same results when compared to previous papers. By not integrating by parts, $p$ will be in $H^1$ and we have to answer the question: Is the discrete Ladyzhenskaya-Babuska-Brezzi (LBB) condition satisfied for the space pair $H^1, H^1$? In this experiment specifically, the numerical scheme seems to be stable but we cannot generalize to all possible cases without a rigorous proof.

As hyperbolic equations need stabilization, we do so with the nonlinear entropy viscosity (Guermond *et al.*, 2011a) using the entropy function $E(x) = -\log|\rho(1-\rho) + 10^{-14}|$. In figure 1, the evolution of the density field of ratio 3 at times 1, 1.5, 2, and 2.5 in Tryggvason time scale $t_{\text{Tryg}} = t\sqrt{A_t}$ with $Re = 1000$. The same times are shown in figure 2 with density ratio of 100. The are 8484 $\mathbb{Q}_2$ degrees of freedom for $\rho$ with uniform mesh size of 2048 cells. The time stepping is variable and maintains a maximum CFL of 0.4.

Now, we want to conduct a more challenging test. Specifically, we will test with density ratio 100 to check the robustness of the scheme – see, for example, (Sussman *et al.*, 1994). As figure 2 shows, the simulation holds nicely. Also, when figure 1 is visually compared with the results in (Guermond *et al.*, 2015), they are visually almost identical.

## CONCLUSION

The Navier-Stokes equations were solved using a code developed based on finite element method to accurately simulate immiscible two phase flows. A proprietary massively parallel Navier-Stokes solver code based on the open source software deal.II was successfully implemented to simulate the interface between the two liquid phases with good accuracy. The utilization of both the level set method with a third order strong stability property Runge-Kutta (SSPRK) time integrator and a second-order projection method for the momentum equation was deemed successful. Numerical val-

idations of the transport and momentum equations were presented; they confirmed the code accuracy. The convergence rate of the numerical schemes selected for modeling the non-conforming manufactured solutions were shown to be within expected convergence rate values. Classical Rayleigh-Taylor instability results were shown to be in good agreement with previously published work.

## ACKNOWLEDGEMENT

The authors would like to thank Saudi Aramco for technical and financial support which made this study possible.

## REFERENCES

GOTTLIEB, S. (2005). "On high order strong stability preserving runge–kutta and multi step time discretizations". *Journal of Scientific Computing*, **25(1)**, 105–128.

GUERMOND, J.L. and PASQUETTI, R. (2011). "Entropy viscosity method for high-order approximations of conservation laws". J.S. Hesthaven and E.M. Rønquist (eds.), *Spectral and High Order Methods for Partial Differential Equations*, vol. 76 of *Lecture Notes in Computational Science and Engineering*, 411–418. Springer Berlin Heidelberg. URL http://dx.doi.org/10.1007/978-3-642-15337-2_39.

GUERMOND, J.L. and SALGADO, A.J. (2011). "Error analysis of a fractional time-stepping technique for incompressible flows with variable density". *SIAM J. Numer. Anal.*, **49(3)**, 917–944. URL http://dx.doi.org/10.1137/090768758.

GUERMOND, J.L., PASQUETTI, R. and POPOV, B. (2011a). "Entropy viscosity method for nonlinear conservation laws". *Journal of Computational Physics*, **230(11)**, 4248–4267.

GUERMOND, J.L., PASQUETTI, R. and POPOV, B. (2011b). "Entropy viscosity method for nonlinear conservation laws". *Journal of Computational Physics*, **230(11)**, 4248 – 4267. Special issue High Order Methods for CFD Problems, URL http://www.sciencedirect.com/science/article/pii/S0021999110006583.

GUERMOND, J.L., PASQUETTI, R. and POPOV, B. (2011c). "From suitable weak solutions to entropy viscosity". *Journal of Scientific Computing*, **49(1)**, 35–50. URL http://dx.doi.org/10.1007/s10915-010-9445-3.

GUERMOND, J.L., SALGADO, A.J. and SHEN, J. (2015). "Splitting for variable density flows". *in preparation*.

JOHN, V. (2004). *Large Eddy Simulation of Turbulent Incompressible Flows: Analytical and Numerical Results for a Class of Les Models*. Lecture Notes in Computational Science and Engineering. Springer Berlin Heidelberg. URL http://books.google.com/books?id=Y-Z0qMEiwdUC.

OSHER, S. and SETHIAN, J.A. (1988). "Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations". *Journal of Computational Physics*, **79(1)**, 12 – 49. URL http://www.sciencedirect.com/science/article/pii/0021999188900022.

SMAGORINSKY, J. (1963). "General circulation experiments with the primitive equations". *Monthly Weather Review*, **91(3)**, 99–164. URL http://dx.doi.org/10.1175/1520-0493(1963)091<0099:GCEWTP>2.3.CO;2.

SUSSMAN, M. and FATEMI, E. (1999). "An efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow". *SIAM Journal on scientific computing*, **20(4)**, 1165–1191.

SUSSMAN, M., SMEREKA, P. and OSHER, S. (1994). "A level set approach for computing solutions to incompressible two-phase flow". *Journal of Computational Physics*, **114(1)**, 146 – 159. URL http://www.sciencedirect.com/science/article/pii/S0021999184711557.

TRYGGVASON, G. (1988). "Numerical simulations of the rayleigh-taylor instability". *J. Comput. Phys.*, **75(2)**, 253–282. URL http://dx.doi.org/10.1016/0021-9991(88)90112-X.