eCSE1002 Technical Report

Paul Bartholomew, Sylvain Lazet

November 29, 2018

Abstract

This technical report details the work performed as part of EPCC grant eCSE1002: "*An adjoint solver for variable-density flows in the low Mach number limit*" to implement a highly scalable, high-order variable-density solver in the Incompact3D framework and an adjoint solver for use with Incompact3D. The project was split into three work packages: 1) add additional variables and their associated transport equations to Incompact3D and implement a variable-density solver based on the constant-coefficient Poisson solver; 2) add variable transport properties to Incompact3D and a variable-coefficient Poisson solver; 3) implement and test an adjoint solver.

This work has produced the variable-density flow solver "QuasIncompact3D" which is freely available at https://github.com/ptb0890/Quasincompact3d. It has been validated for incompressible and variable-density flows and its scalability demonstrated on several TIER-1/0 supercomputers, covering both x86-64 and accelerator based architectures. A continuous adjoint solver has been implemented and is currently being tested.

QuasIncompact3D has been introduced to the scientific community through talks and a paper submitted for peer review. QuasIncompact3D will be used to conduct detailed studies of non-Boussinesq gravity currents and as part of a collaboration within Imperial College to investigate mixing in variable-density flows. Furthermore, QuasIncompact3D will be used as the basis for another eCSE project (eCSE1303) to extend its capabilities to simulate free surface flows with even higher density (air/water) than considered in this project.

1 Introduction

The principal aim of this project was to develop Incompact3D for use in variable-density flow cases, providing for the ARCHER community a highly scalable, high-order, variable-density flow solver. This was done in the context of low speed flows using the Low Mach Number (LMN) approximation, with governing equations summarised as

$$p^{(0)} = \rho T \ , \tag{1}$$

$$\frac{D\rho}{Dt} = -\rho \boldsymbol{\nabla} \cdot \boldsymbol{u} \ , \tag{2}$$

$$\boldsymbol{\nabla} p^{(0)} = 0 \ , \tag{3}$$

$$\rho \frac{DT}{Dt} = \frac{1}{RePr} \boldsymbol{\nabla}^2 T \ , \tag{4}$$

$$\rho \frac{D\boldsymbol{u}}{Dt} = -\boldsymbol{\nabla} p^{(1)} + \frac{1}{Re} \boldsymbol{\nabla} \cdot \boldsymbol{\tau} \ , \tag{5}$$

where ρ, $\boldsymbol{u}$ and T are the density, velocity vector and temperature respectively; $p^{(0)}$ and $p^{(1)}$ are the thermodynamic and mechanical pressure as arise in the LMN decomposition; $\boldsymbol{\tau}$ is the viscous stress tensor; and Re and Pr are the Reynolds and Prandtl numbers respectively.

In addition an adjoint solver was implemented for Incompact3D. The adjoint can be used for efficiently performing design/sensitivity studies where there are a large number of parameters to optimise over.

Incompact3d is an open-source high-order flow solver that can undertake turbulence-resolving simulations of fluid flow phenonema on Tier-0/Tier-1 systems. Since 2012, it is freely available to scientific community (`www.incompact3d.com` and `https://github.com/xcompact3d`) and it has a community of more than 250 active users worldwide. Publications made with this solver can be found at `http://www.incompact3d.com/impact.html`.

Incompact3d is a Fortran 90/95 finite-difference code which solves the incompressible Navier-Stokes equations (and a transport equation for passive scalars) on a Cartesian mesh using sixth order schemes for the spatial discretisation while Runge-Kutta or Adams-Bashforth schemes are used for the time advancement. To treat the incompressibility condition, a fractional step method requires to solve a Poisson equation, fully solved in spectral space via the use of relevant 3D Fast Fourier transforms. Combined with the concept of the modified wave number (Lele, 1992), this direct (i.e. non-iterative) technique allows the implementation of the divergence-free condition up to machine accuracy. A partially staggered mesh is used where the pressure mesh is shifted by a half-mesh from the velocity mesh in each direction. This type of mesh organization leads to more physically realistic pressure fields with no spurious oscillations. More details about the code can be found in (Laizet and Lamballais, 2009).

The high level of parallelisation (good scalability with more than one million MPI-processes) in Incompact3d is achieved thanks to a highly scalable 2D domain decomposition library and a distributed Fast Fourier Transform (FFT) interface (Laizet and Li, 2011). This open-source library is available at `http://www.2decomp.org/`. The 3D computational domain is divided in pencils with three different options referred as X-pencil, Y-pencil and Z-pencil (*fig.*1 from left to right). The derivatives and interpolations in the x-direction (y-direction, z-direction) are performed in X-pencil (Y-pencil, Z-pencil), respectively. The 3D FFTs required by the Poisson solver are also broken down as series of 1D FFTs computed in one direction at a time. The global transpositions to switch from one pencil decomposition to another are performed with the MPI command MPI_ALLTOALL(V). Incompact3d can scale well up to one million MPI-processes for simulations with several billion mesh nodes (Laizet and Li, 2011).

Incompact3d has been used recently for a variety of projects ranging from fractal-generated turbulence, gravity currents, mixing layers, boundary layers, impinging jets on a heated plate, active flow control of a turbulent jet and simulations of plasma actuators (see `http://www.incompact3d.com/impact.html` for all the Incompact3d-based papers published in the last few years by different research groups worldwide).

This report is organised as follows: sections 2, 3 and 4 summarise the aims of work packages 1-3, the work done and the achievement of the goals; section 5 details the outputs of this work in terms of interacting with the community; and the report is concluded in section 6.

2 Work Package 1 (WP1)

The main objectives for WP1 were:

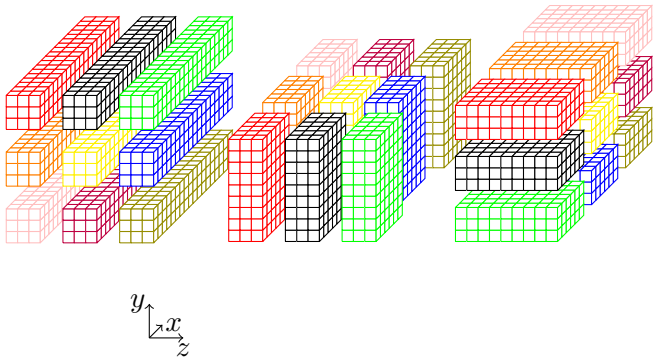


Figure 1: Schematic representation of the 2D domain decomposition showing from left to right the X-, Y- and Z-pencils. Colour indicates one processor/MPI rank.

- Adding additional variables required for LMN to Incompact3D (ρ, T)
- Adding additional transport equations for LMN ((2), (4)) - noting that only one needs to be solved, the unsolved for variable can be obtained through the equation of state (1) - and (3) is enforced by the Poisson solver
- Conversion of existing subroutines to LMN versions for example adding density to the momentum equations (5)
- Implementation of the constant-coefficient Poisson equation solver

The main task was to convert the current Poisson solver to one solving the constant-coefficient Poisson equation for a variable density flow:

$$\boldsymbol{\nabla}^2 p^{(1)} = \frac{1}{\Delta t}\left(\boldsymbol{\nabla}\cdot(\rho\boldsymbol{u})^{\star} - \boldsymbol{\nabla}\cdot(\rho\boldsymbol{u})^{n+1}\right) \ , \tag{6}$$

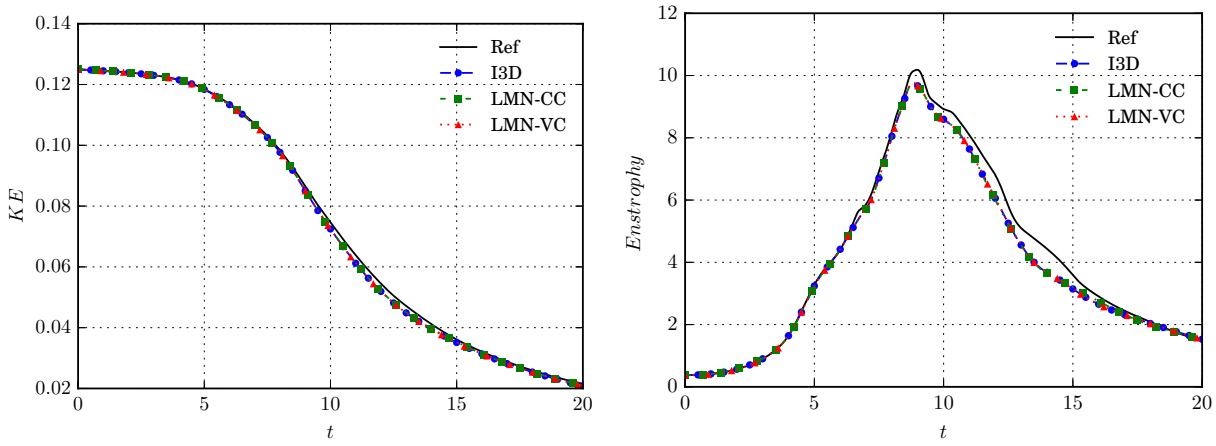
where $(\rho\boldsymbol{u})^{\star}$ is the intermediate momentum obtained from the first step of the fractional step method (Chorin, 1997) and $(\rho\boldsymbol{u})^{n+1}$ is the target divergence of momentum. As the divergence of momentum at the next time is unknown a-priori, its value is extrapolated by means of a backward finite-difference:

$$\boldsymbol{\nabla}\cdot(\rho\boldsymbol{u})^{n+1} = -\left.\frac{\partial\rho}{\partial t}\right|^{n+1} \approx -\frac{\rho^{n+1}-\rho^{n}}{\Delta t} \ , \tag{7}$$

following an approach which has been widely used in the literature (Golanski et al., 2005; McMurtry et al., 1986).

The implementation was tested by simulating the incompressible Taylor-Green Vortex (Jammy et al., 2016; Wang et al., 2013) and the non-isothermal mixing layer of Golanski et al. (2005) and comparing the results with reference data.

The Taylor-Green test case compares QuasIncompact3D with results obtained with "*classic*" Incompact3D both on a 128^3 node mesh with the reference data of Jammy et al. (2016) on a 512^3 node mesh. As *fig.*2 shows there is a good agreement with the results of QuasIncompact3D and Incompact3D - as a constant density incompressible flow this is the desired result indicating that the Navier-Stokes solver is working. The comparison with Jammy et al. (2016) also shows good agreement despite the lower resolution used here.



(a) Kinetic energy

(b) Enstrophy

Figure 2: Taylor-Green Vortex test case. In the figure legends `Ref` corresponds to data from Jammy et al. (2016), `I3D` are the results obtained with "*classic*" Incompact3D and `LMN` are those obtained with QuasIncompact3D using the constant-coefficient (`-CC`) and variable-coefficient (`-VC`) Poisson solvers.

The results of the non-isothermal mixing layer in *fig.*3 show good agreement with the reference results of Golanski et al. (2005), indicating that the LMN solver is working.

To test scalability, the triply periodic setup used for the Taylor-Green vortex was used on the TIER-0/TIER-1 supercomputers ARCHER, MARCONI and HAZEL HEN of EPCC, CINECA and HLRS respectively. Architecturally ARCHER (Cray XC30) and HAZEL HEN (Cray XC40) are similar, both based on 2×12 core Intel®Xeon®processors running at 2.7 GHz and 2.5 GHz respectively, therefore similar performance is expected. MARCONI represents an alternative architecture based on Intel®Xeon®Phi 1×68 core accelerators running at 1.4 GHz and is expected to be somewhat slower, it is however representative of potential future accelerator based machines and presents an opportunity to test QuasIncompact3D on a many core architecture.

As a particular aim of this work is to demonstrate scalability on ARCHER, scaling tests on archer were performed on $1008 \leq NCPU \leq 16128$ cores. To maintain the number of mesh nodes per core in the approximate range $50k \leq n/CPU \leq 1M$, the runs on MARCONI are split into two sets: one covering 1 to 16 nodes ($64 \leq NCPU \leq 1024$) and the other covering 16 to 93 ($1024 \leq NCPU \leq 5952$) nodes (essentially the largest available allocation). Taking a similar approach, the runs on HAZEL HEN cover $2048 \leq NCPU \leq 32768$ and $4096 \leq NCPU \leq 65536$ cores for $2048 \times 1024 \times 1204$ and 2048^3 mesh nodes respectively.

The runtimes are shown in *fig.*4. Good scalability of the code is shown both on ARCHER and the other machines considered. HAZEL HEN represents a similar architecture to ARCHER being based on x86-64 whereas MARCONI is a KNL accelerator based machine indicating that the code is suitable, or could be adapted, for use on upcoming machine architectures.

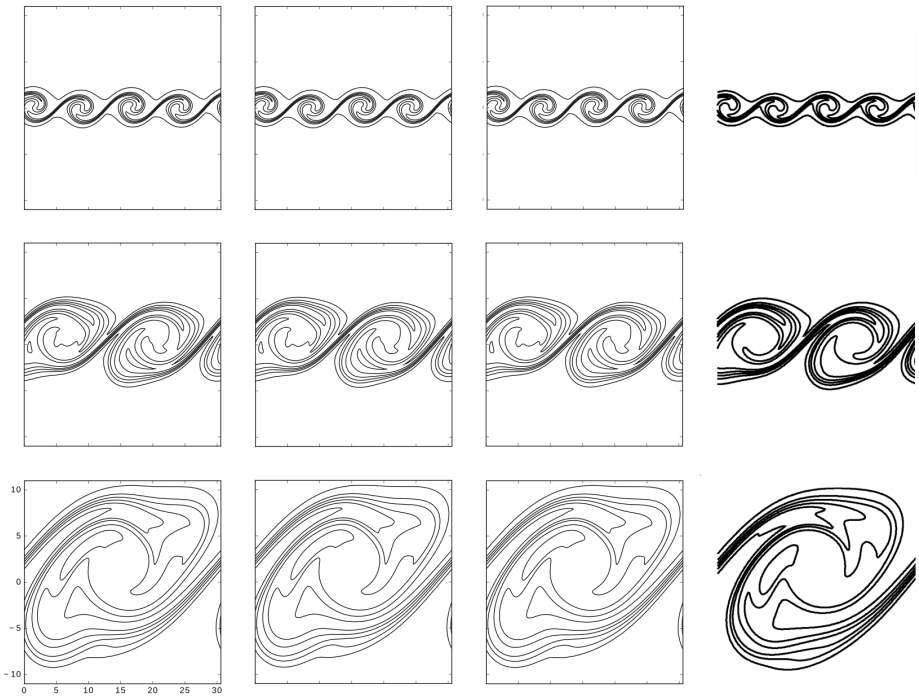


Figure 3: Non-isothermal mixing layer test case showing results from QuasIncompact3D using constant-coefficient and variable-coefficient Poisson equations ($\widetilde{\rho} = \rho_0, \rho^h$ respectively) and reference data of Golanski et al. (2005) respectively.

2.1 WP1 Success

The constant-coefficient algorithm was implemented and shown to agree well with reference data. Good scaling was demonstrated up to 16k cores on ARCHER and additionally on MARCONI and HAZEL HEN with 65k cores.

3 Work Package 2 (WP2)

In WP2 the code was extended to add variable transport properties (viscosity, diffusivity) following the approach of Motheau and Abraham (2016): in the general case the diffusive terms are treated conservatively *i.e.*

$$\boldsymbol{\nabla} \cdot \boldsymbol{q} = \boldsymbol{\nabla} \cdot k \boldsymbol{\nabla} T \ , \tag{8}$$

to guarantee conservation of variables; whereas in the case of the viscous stress it is implemented non-conservatively as

$$\boldsymbol{\nabla} \cdot \boldsymbol{\tau} = \mu \boldsymbol{\nabla} \cdot \left(\frac{\boldsymbol{\tau}}{\mu} \right) + (\boldsymbol{\nabla} \mu) \cdot \left(\frac{\boldsymbol{\tau}}{\mu} \right) \ , \tag{9}$$

to prevent oscillations in the velocity field.

Also in WP2 a variable-coefficient Poisson equation was implemented (selectable at runtime) of the form

$$\boldsymbol{\nabla} \cdot \frac{1}{\rho^{n+1}} \boldsymbol{\nabla} p^{(1)} = \frac{1}{\Delta t} \left(\boldsymbol{\nabla} \cdot \boldsymbol{u}^\star - \boldsymbol{\nabla} \cdot \boldsymbol{u}^{n+1} \right) \ , \tag{10}$$

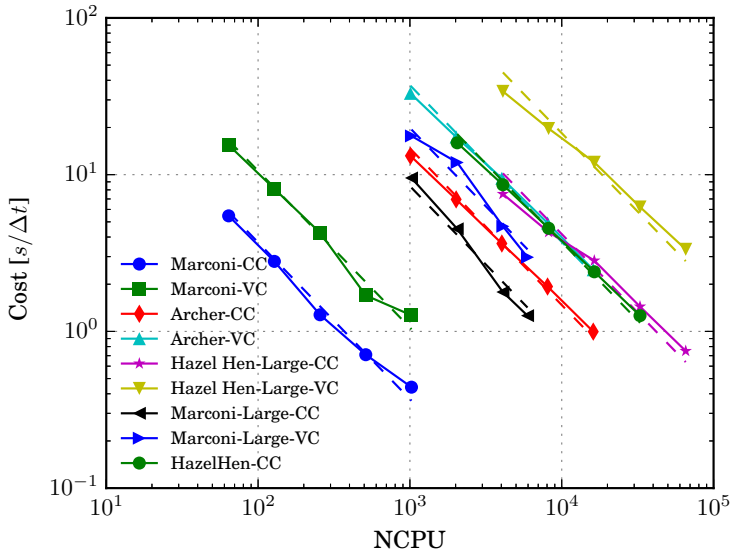


Figure 4: Scaling of QuasIncompact3D on ARCHER, MARCONI and HAZEL HEN. Following the nomenclature in *fig.*2, `-CC` and `-VC` indicates either the constant- or variable-coefficient Poisson equation solver being run on the named machine.

which has the advantage over the constant-coefficient Poisson equation that $\boldsymbol{\nabla} \cdot \boldsymbol{u}^{n+1}$ is calculated as part of the solution process and thus does not require extrapolation - as shown by Nicoud (2000) only this approach enforces the velocity divergence constraint. The variable-coefficient Poisson equation cannot however be solved directly using fast Poisson solvers and is recast in iterative form

$$\boldsymbol{\nabla}^2 p^{(1)^{\nu+1}} = \widetilde{\rho}\left(\frac{1}{\Delta t}\left(\boldsymbol{\nabla}\cdot\boldsymbol{u}^{\star} - \boldsymbol{\nabla}\cdot\boldsymbol{u}^{n+1}\right) - \boldsymbol{\nabla}\cdot\frac{1}{\rho^{n+1}}\boldsymbol{\nabla}p^{(1)^{\nu}}\right) + \boldsymbol{\nabla}^2 p^{(1)^{\nu}} \tag{11}$$

where superscript ν is an iteration counter and $\widetilde{\rho}$ is a freely chosen density scale. This implementation was also tested on the Taylor-Green Vortex and non-isothermal mixing layer, shown in *fig.*2 and *fig.*3. As for the constant-coefficient Poisson solver, the variable-coefficient Poisson solver shows good agreement with the reference data.

In *fig.*3 two sets of results are shown for the variable-coefficient Poisson solver relating to different choices of $\widetilde{\rho}$. The usual choice for $\widetilde{\rho}$ is the minimum of the density field, denoted ρ_0, chosen for stability purposes (Motheau and Abraham, 2016). During implementation of the variable-coefficient solver an alternative choice was developed, based on harmonic averages. Nothing in (11) requires that $\widetilde{\rho}$ be a constant - it simply needs to be evaluated at the same point as the Laplacian of pressure term. In the Incompact3D framework a partially staggered mesh arrangement is employed with transported variables collocated at mesh nodes and pressure stored at the centre of control volumes defined by these nodes. Interpolating density to these control volume centres directly to evaluate $\widetilde{\rho}$ tends towards the maximum value in the interpolation stencil - if we compare this with the previous choice of $\widetilde{\rho} = \rho_0$, we can see that the minimum density has a stabilising effect by minimising the

size of the explicit term in (11). By using the harmonic average to evaluate $\widetilde{\rho}$, *i.e.*

$$\widetilde{\rho} = \rho^h = \left(\overline{1/\rho}\right)^{-1}, \tag{12}$$

where the overline indicates interpolation, it can be shown that the harmonic average tends towards the minimum value in the stencil. Consequently the stabilising effect of $\widetilde{\rho} = \rho_0$ is obtained in areas where density changes and in areas of uniform density $\widetilde{\rho}$ takes the local value, giving better scaling of the explicit term. Testing on the non-isothermal mixing layer case showed that using harmonic average, instead of minimum value, to calculate $\widetilde{\rho}$ lead to faster convergence, as shown in table 1. These results also demonstrate the importance of efficiently solving the Poisson equation, with $\gtrsim 90\%$ of time spent in the Poisson solver for both variable-coefficient versions.

Table 1: Cost analysis of non-isothermal mixing layer simulations showing: CPU time per timestep; average number of iterations of Poisson solver per timestep; cost in terms of iterations relative to constant-coefficient solver; and percentage of time spent in Poisson solver.

Solver	**CPU / Δt [s]**	**Iterations**	**Cost**	**[%] in Poisson**
CC	4.48×10^{-2}	1	1	23.4
VC ρ_0	6.16×10^{-1}	34.1	13.75	94.2
VC ρ_h	3.53×10^{-1}	18.8	7.88	89.9

The variable-coefficient solver's scaling was also tested on ARCHER, MARCONI and HAZEL HEN, using the setup described in §2 with results shown in *fig.*4. For the scaling tests the number of iterations of the variable-coefficient Poisson solver was fixed at 5 per timestep and as can be seen the scalability remains close to ideal across architectures, the cost per timestep simply increases due to the cost of the iterations.

3.1 WP2 Success

The scaling of the variable-coefficient algorithm was successfully tested on ARCHER with 16k cores, demonstrating good scaling in comparison with the constant-coefficient algorithm, the additional cost of the algorithm is due to the cost of the iterations, but does not affect the scaling performance. This scaling was demonstrated on additional machines MARCONI and HAZEL HEN up to 65k cores.

In simulating both incompressible and variable-density flows, the variable-coefficient algorithm shows good agreement with the constant-coefficient algorithm and the reference data.

4 Work Package 3 (WP3)

An adjoint solver has been implemented for Incompact3D using the continuous (optimise-then-discretise) approach. This is outlined in *fig.*5 showing the flow of information through the flow solver forward in time for $t_0 \leq t \leq t_{end}$, at the end of the simulation period the adjoint solver is initialised based on the final time state, the adjoint solver then evolves the adjoint variables back in time to t_0 to determine the gradient and hence change in initial conditions required. This iteration is repeated in a process known as Direct Adjoint Looping (Kerswell, 2018) until an optimal condition at t_{end} is achieved. The adjoint solver is currently being tested in collaboration with Dr.

Ubaid Ali Qadri at Cambridge and it is planned to take the preliminary results to form the basis of an application for EPSRC funding.

5 Outputs

As a result of this work, two talks were given: one at the Incompact3D User Group at Imperial College and one at the UKFluids2018 conference at Manchester University. These talks, focusing on the development of QuasIncompact3D, allowed to communicate directly with current users of Incompact3D and the wider fluid dynamics community in the UK of the new capabilities that QuasIncompact3D enables for fluid dynamics research.

In addition to the above talks, a manuscript has been prepared and submitted for peer review introducing QuasIncompact3D with a preliminary investigation of non-Boussinesq gravity currents. Additional papers related to this work are under preparation: one to investigate non-Boussinesq gravity currents in more detail at high Reynolds number and one focusing on the adjoint solver in collaboration with Dr. Ubaid Ali Qadri at Cambridge.

Finally, in future work it is proposed to use QuasIncompact3D to investigate non-Boussinesq gravity currents in greater detail and a collaboration has been proposed within Imperial College to use QuasIncompact3D to investigate mixing in variable-density flows. This work also forms the basis of a newly funded eCSE project (eCSE1303) to extend QuasIncompact3D for use to simulate free-surface flows with considerably higher density ratios (that of air/water) than considered here.

6 Conclusion

This work has met its targets of developing a variable-density flow solver in the Xcompact3D framework with algorithms for solving both the constant- and variable-coefficient Poisson equations and good agreement shown with reference data. It was shown to maintain good scaling on ARCHER up to 16k cores and on the MARCONI and HAZEL HEN supercomputers with up to 65k cores on HAZEL HEN. An adjoint solver has been implemented and is currently being tested in collaboration with Dr. Ubaid Ali Qadri at Cambridge.

QuasIncompact3D has been introduced to both the Incompact3D user group and the wider UK fluid dynamics community by way of presentations. It is already having an impact in the scientific community, with a paper submitted for peer review and with future work in the form of an internal collaboration being proposed.

7 Acknowledgements

This work was funded under the embedded CSE programme of the ARCHER UK National Supercomputing Service (http://www.archer.ac.uk). Computing time was provided on ARCHER by EPCC grant eCSE1002 and the UK turbulence consortium grant EP/R029321/1 and on MARCONI and HAZEL HEN by PRACE grants 2016 163847 and 2018 184381. Additional computing time was provided by the Imperial College Research Computing Service.

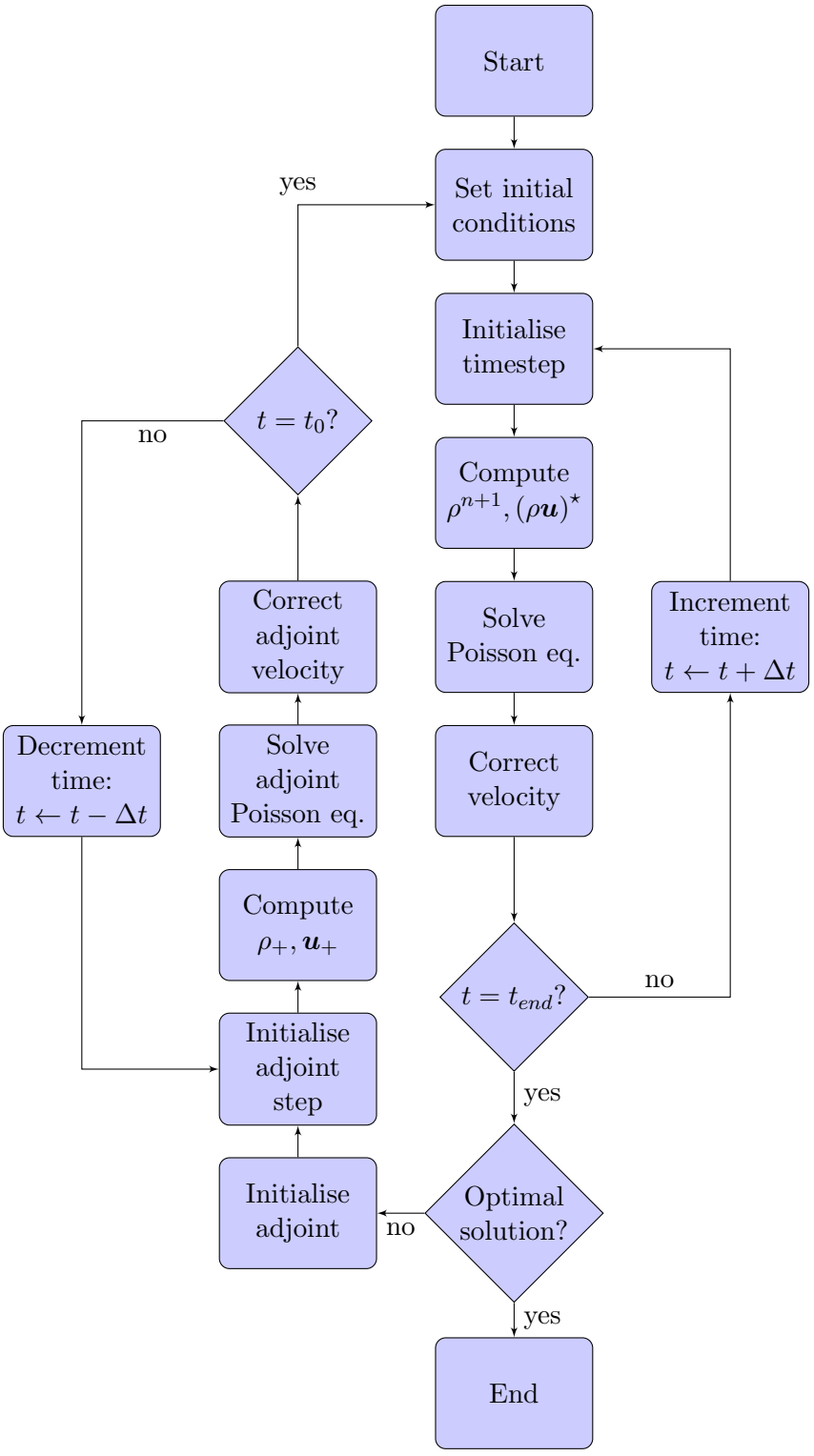


Figure 5: Schematic showing the Direct Adjoint Looping solver.

References

Alexandre Joel Chorin. A numerical method for solving incompressible viscous flow problems. *Journal of computational physics*, 135(2):118–125, 1997.

François Golanski, Véronique Fortuné, and Eric Lamballais. Noise radiated by a non-isothermal, temporal mixing layer. Part II: Prediction using DNS in the framework of low Mach number approximation. *Theoretical and Computational Fluid Dynamics*, 19(6):391–416, December 2005. ISSN 0935-4964, 1432-2250. doi: 10.1007/s00162-005-0002-x.

Satya P. Jammy, Christian T. Jacobs, and Neil D. Sandham. Performance evaluation of explicit finite difference algorithms with varying amounts of computational and memory intensity. *Journal of Computational Science*, November 2016. ISSN 18777503. doi: 10.1016/j.jocs.2016.10.015.

R.R. Kerswell. Nonlinear Nonmodal Stability Theory. *Annual Review of Fluid Mechanics*, 50(1): 319–345, January 2018. ISSN 0066-4189, 1545-4479. doi: 10.1146/annurev-fluid-122316-045042.

Sylvain Laizet and Eric Lamballais. High-order compact schemes for incompressible flows: A simple and efficient method with quasi-spectral accuracy. *Journal of Computational Physics*, 228(16): 5989–6015, September 2009. ISSN 00219991. doi: 10.1016/j.jcp.2009.05.010.

Sylvain Laizet and Ning Li. Incompact3d: A powerful tool to tackle turbulence problems with up to O(105) computational cores. *International Journal for Numerical Methods in Fluids*, 67(11): 1735–1757, December 2011. ISSN 02712091. doi: 10.1002/fld.2480.

Sanjiva K. Lele. Compact finite difference schemes with spectral-like resolution. *Journal of Computational Physics*, 103(1):16–42, 1992.

P. A. McMurtry, W.-H. Jou, J. Riley, and R. W. Metcalfe. Direct numerical simulations of a reacting mixing layer with chemical heat release. *AIAA Journal*, 24(6):962–970, June 1986. ISSN 0001-1452, 1533-385X. doi: 10.2514/3.9371.

E. Motheau and J. Abraham. A high-order numerical algorithm for DNS of low-Mach-number reactive flows with detailed chemistry and quasi-spectral accuracy. *Journal of Computational Physics*, 313:430–454, May 2016. ISSN 00219991. doi: 10.1016/j.jcp.2016.02.059.

F. Nicoud. Conservative High-Order Finite-Difference Schemes for Low-Mach Number Flows. *Journal of Computational Physics*, 158(1):71–97, February 2000. ISSN 00219991. doi: 10.1006/jcph.1999.6408.

Z.J. Wang, Krzysztof Fidkowski, Rémi Abgrall, Francesco Bassi, Doru Caraeni, Andrew Cary, Herman Deconinck, Ralf Hartmann, Koen Hillewaert, H. T. Huynh, Norbert Kroll, Georg May, Per-Olof Persson, Bram van Leer, and Miguel Visbal. High-order CFD methods: Current status and perspective. *International Journal for Numerical Methods in Fluids*, 72(8):811–845, July 2013. ISSN 02712091. doi: 10.1002/fld.3767.