UNIVERSITY OF TWENTE.

Faculty of Science and Technology
Faculty of Electrical Engineering, Mathematics and Computer Science

Physics of Fluids
Multiscale Modelling and Simulation

Drag reduction in two-phase Taylor-Couette flow

Author:
Arnout Franken

Supervisors:
Prof. Dr. Ir. Bernard Geurts (MMS)
Prof. Dr. Detlef Lohse (PoF)
Prof. Dr. Roberto Verzicco
Dr. Paolo Cifani
Dr. Ir. Menno Bokdam

July 17, 2020

Preface

This thesis is the conclusion of my work during the past 14 months. The first four months were spent at the Technische Universität Dresden, and for the remainder of the time, I was situated at the University of Twente. It marks the end of my six years of studying in Enschede; a time that I have thoroughly enjoyed. Besides having developed academically, it has allowed me to gradually evolve into the person I am today. I am thankful for all the friends I have met, and all the amazing experiences we had that I can proudly look back on.

I would like to take this opportunity to express gratitude toward the people that have made this graduation project possible. First of all, I want to thank my supervisors, Bernard and Detlef, who offered me the opportunity to work on this interesting topic. Especially the weekly discussions with Bernard were incredibly useful as he managed to keep me on track, always see the bigger picture and motivate me after every setback. During the last few months, the meetings were joined by Paolo, whose knowledge of the code and insight in the research topics was invaluable.

I also want to thank prof. Fröhlich and Bernhard Ott for hosting me at the Fluid Mechanics group in Dresden, where they guided me through the first part of this research. The discussions were very insightful for my research. It has been a great experience to work in their group.

Needless to say, I would like to thank my family and friends for all the support throughout the years. It has been great to make music, and drink coffee and beer with you. This gratitude extends to my roommates, who kept my spirits up during the last four months when I was forced to work from home during the Corona crisis.

Finally, I want to thank the members of my graduation committee for taking their time and effort to read and evaluate my thesis.

Abstract

In this thesis, we present a solver for direct numerical simulation (DNS) of two-phase Taylor-Couette (TC) flow. It is an extension of the TBFsolver; a highly scalable and efficient solver for dispersed gas-liquid channel flow (Cifani et al., 2018). We describe the thorough adaptation of the numerical methods to cylindrical coordinates, required for TC flow. The two-phase system is solved in the one-fluid formulation. We employ the fully conservative finite difference scheme for incompressible flow developed by (Morinishi et al., 2004) as well as a variable coefficient Poisson equation solver (Cifani, 2019). Advection of the bubble interface is carried out using a geometric volume-of-fluid (VOF) approach in the computational domain. The generalized height-function method (GHF) was adopted for the curvature computations. The surface tension force is modelled using the continuous-surface-force method (CSF).

By simulating carefully selected test cases described in literature, we observed close agreement with the reference results. Moreover, the computational performance of the VOF method achieved with the existing TBFsolver could be retained at a high level. Furthermore, by suitably varying the curvature of the domain, we observe convergence to results obtained previously on a rectangular domain. Simulations of single-phase TC flow measuring the torque on the cylinders near the critical Reynolds number show an agreement within 0.05% with literature data. At a higher Reynolds number (Re), the torque is determined via the angular velocity flux and shows agreement within 5% with literature data.

The computational capabilities of the extended computational platform are illustrated by a simulation of 120 highly deformable bubbles at $\mathrm{Re} = 5 \cdot 10^3$ and $We = 8$. Starting from developed single-phase flow, we inject bubbles into the flow arranged initially in a regular pattern. We observe transient strong migration of bubbles to the inner cylinder due to the large scale plumes present in the single-phase flow. After some time the flow has developed more fully and this plume structure is disrupted by the bubbles. This is clearly expressed by the PDF of the bubble locations. Measurements of the torque based on shear stress at the walls show no conclusive results yet on drag reduction as longer simulation time is required. The development of this code facilitates future research into drag reduction and flow structuring in two-phase flows at high fluid density ratios and high bubble deformability.

Contents

1 Introduction

1.1 Drag reduction in multiphase flows

Frictional energy drains have long been a major issue in fluid transport, which is ubiquitous in naval transportation, process technology, transportation of liquified natural gas and other industries. Historically, many ideas have been proposed to reduce this energy drain, both passive and active. Examples of passive methods are the addition of roughness and structures to the surface and shaping a surface to maintain laminar flow in the downstream direction as much as possible (Sanders et al., 2006). Active methods include introducing transverse wall oscillations or even electro-kinetic forcing of the near-wall flow for unsteady energy addition. In the middle ground of these techniques lies the injection of a second phase into the flow to reduce skin friction.

The idea of introducing gas bubbles to fluid flow in order to reduce drag has been around for at least a century. Latorre reports that patents on using air lubrication to reduce the drag on ship hulls were filed as early as 1880 (Latorre, 1997). Early implementations of this idea focused mostly on the addition of polymers to flows, which showed promising results in turbulent pipe flow (Virk, 1975). During the last four decades, however, many theoretical, numerical and experimental studies have been performed on the lubricating effect of air bubbles in liquid flows, which show promising results in reducing the frictional energy loss in various flow configurations (Madavan et al., 1984; Sanders et al., 2006; Ceccio, 2010; Van Gils et al., 2013; Verschoof et al., 2016).

The magnitude of drag reduction or of reduction of the driving force in two-phase flows compared to single-phase flows can be quite substantial. It is shown that adding dispersed bubbles at a gas volume concentration of just 4% can lead to drag reduction of up to 40% in Taylor-Couette flows (Van Gils et al., 2013). Early work on water tunnels even shows an 80% reduction in skin friction at high Reynolds numbers (Madavan et al., 1984). The relevance of this research to naval applications was shown by Kodama *et al.*, as they observed a skin friction reduction of up to 40% under realistic conditions for ships (Kodama et al., 2000). These results show the significance of this research area to the naval industry, as drag reduction can lead to a significant decrease in the fuel consumption for ships. Also in many other application areas, the reduction of skin friction can improve the performance of key processes, e.g., in the transport of chemicals in industrial equipment.

Much research on the mechanism behind drag reduction is focussed on the study of Taylor-Couette (TC) flow, the flow between two concentric cylinders. This flow configuration has several advantages over many other systems. Firstly, it is a closed flow system which allows us to derive exact global balances. The flow is usually driven by the externally controlled rotation of the cylinders, which enables the measurement of the external energy input via the torque on these cylinders. Secondly, statistically stationary states can be achieved in the Taylor-Couette system, which allows an easier evaluation of the energy balances (van den Berg et al., 2005). There are also more practical considerations, such as the relatively simple geometry and implementation of boundary conditions in TC flow (Spandan et al., 2016).

Despite all research efforts in the past few decades, there is still no solid understanding of the physical mechanisms behind this drag reduction in turbulent flows (Grossmann et al., 2016; Lohse, 2018). Various theories have been proposed to explain the origin of this effect. Among these mechanisms, bubble compressibility and deformability are shown to influence the amount of drag reduction for small bubbles (Lo et al., 2006; van den Berg et al., 2005).

Recently, it has been established that bubble size (Verschoof et al., 2016) plays an important role in the realisation of drag reduction. This indicates the important role of bubble deformability in hindering the exchange of angular momentum between the boundary layer and bulk in TC flow (Spandan et al., 2017a). In light of these findings, multiple different physical effects have been shown to contribute to drag reduction with the use of fully resolved numerical simulations (Spandan et al., 2018). In two-phase TC flow, bubbles weaken the large-scale structures that are responsible for momentum transport. Compared to single-phase flow, turbulence near the inner cylinder is enhanced, while it is attenuated in the bulk. Furthermore, the intensity of dissipative structures decreases with increasing bubble deformability, affirming its relevance to drag reduction.

1.2 VOF methods for two-phase flows

The importance of deformability of finite-sized bubbles calls for the employment of direct numerical simulations (DNS) of two-phase flows, as it uniquely enables us to resolve to flow up to the smallest physical scales of the system (Pope, 2000). This section provides a short description of the choice of DNS method for this research.

Two-phase flows are a broad category of systems in which two different fluid phases, or fluid and solid phases, are simultaneously present (Prosperetti and Tryggvason, 2007). An example of this is a liquid carrier flow that transports gas bubbles; the topic of this thesis. A comprehensive overview of DNS methods is provided in (Tryggvason et al., 2013).

Most models use the so-called one-fluid approach, where the whole flow field is simulated by solving a single set of equations (Prosperetti and Tryggvason, 2007). The differences in material properties and interface phenomena are accounted for by introducing the appropriate surface terms. A further distinction can be made between interface tracking and interface capturing methods (Mirjalili et al., 2017). In the first instance, the fluid is simulated in an Eulerian framework, whereas the interface is tracked using Lagrangian markers. In the latter method, the distinction between phases is made using an indicator function on the grid cells. Several choices of transporting this indicator function are found in literature, such as the volume of fluid (VOF) method (Popinet, 2009) and the level-set method (Sussman and Puckett, 2000). In this research, a VOF method is implemented for reasons listed below.

Even though the VOF method can be used in general two-phase flows, the following description is based on the current setup with dispersed gas bubbles in a fluid. In the VOF method, an indicator function is constructed that takes the value of 1 whenever the considered location is inside the gas phase, and 0 when inside the fluid phase. The value of the indicator function for a fluid particle is assumed not to change, which leads to the advection equation for the indicator function.

In this research, we utilize the VOF method developed and implemented by (Cifani et al., 2018), which is in part based on earlier research by (Popinet, 2009). In this method, the indicator function is used to perform a geometrical interface reconstruction based on piece-wise linear interface calculation (PLIC) (Puckett et al., 1997). The subsequent advection fluxes can be calculated based on geometrical considerations of the local velocity field. The method does not guarantee exact conservation of mass, leading to an error in the volume fraction advection. However, in most cases, this error is small compared to other sources of errors such as discretisation errors, and the method has been shown to converge to second-order accuracy with grid refinement (Rider and Kothe, 1998; Scardovelli and Zaleski, 2000).

One of the biggest challenges for this method is the computation of surface tension. This relies on the accurate determination of the curvature field describing details of the bubble interfaces. Numerically, a key method is to approximate the curvature using derivatives of the indicator function. This is done using the Generalized Height Function (GHF) method. Height-function methods usually perform well in terms of accuracy, convergence and conservation properties, but are prone to perform less well at a low spatial resolution of the interface (Popinet, 2018). The GHF method employs a parabolic reconstruction of the interface for the calculation of curvature when the standard height-function fails at low resolution. This hybrid method improves the performance at a low resolution of the interface and high curvature (Popinet, 2009). The resulting method is shown to be both accurate and robust, with limited additional computational costs, but at the expense of a significant increase in code complexity.

A complete DNS solver for turbulent two-phase channel flow that implements the GHF method was developed by Cifani (Cifani et al., 2018). This code, called the TBFsolver, was used to simulate $O(10^4)$ bubbles in turbulent flow, which showed excellent parallel performance with linear scaling up to $\approx$ 18000 computational cores. Its computational performance, coupled with the accuracy and robustness of the employed numerical methods, makes the TBFsolver a viable tool in researching the mechanisms behind drag reduction. In this work, we adopted this solver as the point of departure for the extension to cylindrical coordinates, capable of simulation of turbulent TC flow.

1.3 Goal and structure of this thesis

In this study, we aim to extend and employ the TBFsolver for the study of drag reduction in TC flow. The accuracy and robustness of the TBFsolver at high curvature enables the study of highly deformable bubbles. The main challenge in this research is the extension of the method to the cylindrical geometry of TC flow. The code will then be used to study the energy balance of bubble-laden TC flow, which can provide insight into the physical mechanisms behind the observed drag reduction. Moreover, as we adhere to first principles throughout, the results can also serve as a point of reference for other methods and simplified models.

The thesis is structured into two parts. In the first part, we focus exclusively on the numerical methods for simulating the carrier fluid, which results in a solver for single-phase in TC flow. In section 2, the TBFsolver is introduced is more detail, and the adaptation of the single-phase solver to the cylindrical geometry is discussed. In the second part of the thesis, we discuss the implementation of the complete two-phase solver for TC flow. Section 3 describes the VOF method in more detail and shows its implementation in cylindrical coordinates. In section 4, several test cases are described to test the code performance. Next, the energy balance in bubble-laden Taylor-Couette flow is analysed in section 5.

The main conclusions are summarised in section 6, where we also give an overview of possible future research topics. The reader is referred to appendix B for a list of dimensionless numbers and variables.

Part I
Single phase Taylor-Couette flow

2 Single phase Taylor-Couette flow

In this chapter, the adaptation part of the TBFsolver that deals with single phase flow to the cylindrical geometry is described.

The chapter is organised as follows. In section 2.1, the mathematical formulation of the problem is given, along with some important properties of TC flow. The numerical methods that are to be implemented in the flow solver are shown in 2.2, which concludes with an overview of all the necessary modifications to the original code. Section 2.3 describes the simulations that are used for validation of the code. Finally, in section 2.4 the results for the single-phase solver are summarised.

2.1 Problem description

Taylor-Couette (TC) flow is the flow between two concentric and independently rotating cylinders. The flow can exhibit a large variety of states, from laminar to turbulent states, including states with co-existence of laminar and turbulent regions (Andereck et al., 1986). This, and the fact that it is a closed system, contributes to TC flow being one of the most studied objects of physics of fluid (Ostilla-Mónico et al., 2014).

One of the first rigorous analyses on the system was done by Taylor (Taylor, 1923), who studied the transition from laminar Couette flow to a laminar flow state consisting of pairs of counter-rotating vortices, which is now referred to as Taylor-vortex motion. This transition has also been studied extensively by linear stability analysis (Drazin and Reid, 2004).

Experiments by (Wendt, 1933) found power laws relating the torque on the inner cylinder to the velocity of that cylinder. This scaling behaviour has since become the main focus for many studies, both experimental (Lathrop et al., 1992; Echeverry, 2014) and numerical (Bilson and Bremhorst, 2007; Dong, 2008).

2.1.1 Description of Taylor-Couette flow

The geometry for the flow is shown in figure 1. The domain is bounded radially by two co-axial cylinders with radius r_i and r_o respectively for the inner and outer walls. These wall can rotate at an externally controlled angular velocity ω_i and ω_o respectively. The geometric parameters of the TC domain are the ratio of the cylinder radii η given by:

$$\eta = \frac{r_i}{r_o}, \tag{1}$$

and the ratio Γ of the axial extent of the cylinders and the gap width between the cylinders given by:

$$\Gamma = \frac{L_z}{d}, \tag{2}$$

where $d \equiv r_o - r_i$ is the gap size. The walls can be modelled using no-slip conditions, meaning that the fluid is co-moving with the walls. In experimental studies, the domain is generally bounded in the axial direction by fixed or rotating end plates. It was shown that this can have significant impact on the stability of the flow in the laminar regime (Coles and Atta, 1966). For this reason, many numerical studies use periodic boundary conditions in the axial direction. It was shown that an accurate value of the torque can be obtained using periodic boundary conditions, even at small aspect ratios (Brauckmann and Eckhardt, 2013; Ostilla-Mónico et al., 2015).

TC flow is usually characterised by the dimensionless parameters Re (Reynolds number) and Ta (Taylor number), based on the geometrical parameters, the cylinder velocities and the kinematic viscosity ν of the fluid. For TC flow, two Reynolds numbers can be defined, based on the velocity of the inner and outer walls respectively:

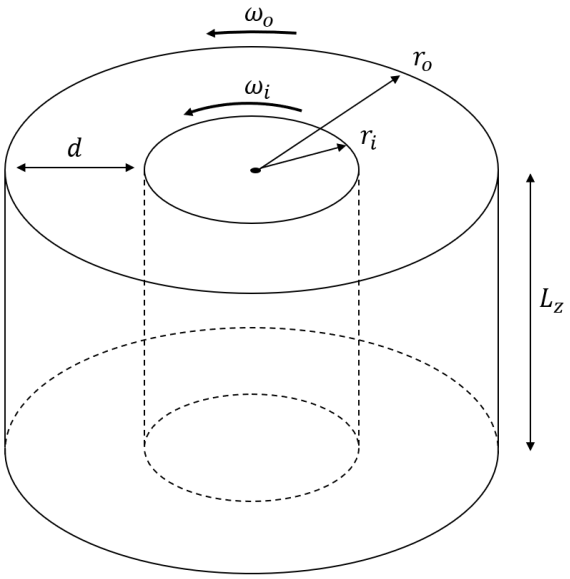


Figure 1: Domain for the Taylor-Couette flow.

$$Re_{i/o} = \frac{\omega_{i/o} r_{i/o} d}{\nu}. \tag{3}$$

In this study, only Re_i will be used, as the outer cylinder will be stationary during all simulations. The Taylor number characterises the ratio between the inertial forces due to rotation and the viscous forces. The definition of the Taylor number may differ between different studies, depending on the specific goal of a study. We define it as follows:

$$Ta = \frac{r_a^6 d^2}{r_o^2 r_i^2 \nu^2} (\omega_o - \omega_i)^2 = \frac{r_a^4}{r_o^2 r_i^2} \left(\frac{r_a (\omega_o - \omega_i) d}{\nu} \right)^2, \tag{4}$$

where $r_a = \frac{1}{2}(r_i + r_o)$ is the arithmetic mean radius. Here, $\frac{r_a^4}{r_o^2 r_i^2}$ is a measure for the curvature of the domain, while the second factor in equation 4 relates the mid-gap inertial forces to the viscous forces. This definition will be useful in the evaluation of the energy balance of the system (Eckhardt et al., 2007).

2.1.2 Torque balance

In TC flow, the fluid motion is driven by a rotation of the inner and outer cylinder. Typically, the cylinders have no-slip walls and are rotated at an externally controlled constant angular velocity. This causes the fluid to exert a torque T on the cylinders, based on the fluid shear stress at the wall. At the inner wall, the torque T_i is given by:

$$T_i = r_i \int_A \tau_{r\theta} dA = r_i (2\pi r_i) L_z \langle \tau_{r\theta} \rangle_A , \tag{5}$$

where T_i is the torque, r_i the radius of the inner cylinder, L_z is the height of the system, $\tau_{r\theta} = \mu \left(\frac{\partial u_\theta}{\partial r} + \frac{1}{r} \frac{\partial u_r}{\partial \theta} - \frac{u_\theta}{r} \right)$ is the shear stress at the wall and $\langle \ldots \rangle_A$ represents an average over the (θ, z)-plane. The torque on the outer cylinder is defined similarly. The torque can be expressed in the non-dimensional quantity:

$$G = \frac{T}{2\pi L_z \rho \nu^2}, \tag{6}$$

where ρ is the fluid mass density. When the angular velocities of the inner and outer wall differ, angular momentum is transported from one wall to the other. In a statistically stationary flow, the torque on the inner and outer cylinder wall must be equal and of opposite sign, which results in a balance equation for torque, since $T_i = -T_o$ must be satisfied. Eckhardt, Grossmann and Lohse (Eckhardt et al., 2007) showed that this torque balance can be generalised. They showed that in developed flow, there must be a constant flux of angular momentum through the domain from one wall to the other. This angular momentum flux J^ω is defined by:

$$J^\omega = r^3 \left[\langle u_r \omega \rangle_{A,t} - \nu \frac{\partial}{\partial r} \langle \omega \rangle_{A,t} \right], \tag{7}$$

where $\omega = u_\theta / r$ is the angular velocity, $\langle \ldots \rangle_{A,t} = \int \frac{dz}{Lz} \int \frac{d\theta}{L_\theta} \int \frac{dt}{T_s} (\ldots)$ for some time-averaging period T_s and ν is the dynamic viscosity of the fluid.

The angular momentum flux consists of two parts. The first term represents the convective transport of angular velocity, while the second term represents molecular transport, which is proportional to the kinematic viscosity of the fluid. In a statistically stationary flow, J^ω is independent of the radius r. At the wall, the convective term of equation (7) is zero due to the boundary condition on u_r, so we recover the torque on the wall up to a factor $2\pi L_z \rho$, resulting in the following relations between torque and angular momentum flux:

$$T = 2\pi L_z \rho J^\omega, \qquad G = \nu^{-2} J^\omega. \tag{8}$$

The angular momentum flux is very similar to its counterpart in the thermally driven Rayleigh-Bénard convection. In that case the flow is driven by the convection that is induced by a temperature difference between the top and bottom wall in a rectangular domain. This results in a constant heat flux J^θ from the hot to the cold end of the domain, given by:

$$J^\omega = \langle u_z \theta \rangle_{A,t} - \kappa \frac{\partial}{\partial z} \langle \theta \rangle_{A,t}, \tag{9}$$

where the temperature θ is transported along the wall-normal direction instead of the angular velocity ω. Using this analogy, we can define a Nusselt number for Taylor-Couette flow, which relates to the transport of angular momentum. We define the Nusselt number Nu_ω using the analytical solution for the angular velocity flux in the case of laminar Couette flow:

$$Nu_\omega = \frac{J^\omega}{J_0^\omega}, \tag{10}$$

where J_0^ω is the angular momentum flux in the case of laminar Couette flow. In that case, the velocity profile is given by:

$$\mathbf{u} = u_\theta(r)\hat{\theta} = Ar + \frac{B}{r}, \quad \text{where} \quad A = \frac{\omega_o - \eta^2 \omega_i}{1 - \eta^2}, \qquad B = \frac{(\omega_i - \omega_o) r_i^2}{1 - \eta^2}, \tag{11}$$

where ω_i and ω_o are the fixed angular velocities of the inner and outer wall respectively and η is the cylinder radius ratio (equation 1). This profile has the following angular momentum flux:

$$J_0^\omega = 2\nu B = 2\nu \frac{(\omega_o - \omega_i) r_i^2}{1 - \eta^2} = -\nu \frac{2}{\eta(1+\eta)} r_i^3 \left(\frac{\omega_i - \omega_o}{d} \right). \tag{12}$$

This generalises the torque balance to the conservation of angular momentum flux, since we require Nu_ω to be independent of the radial coordinate.

2.1.3 Energy balance

Exploiting the similarity between Taylor-Couette flow and Rayleigh-Bénard convection, we can define an analogue to the Prandtl number for TC flow. In RB convection, the Prandtl number represents the ratio between momentum diffusivity and thermal diffusivity. In a similar fashion we can define a quasi-Prandtl number Φ based on the geometry of the domain by (Eckhardt et al., 2007):

$$\Phi = \frac{r_a^4}{r_i^2 r_o^2} = \left(\frac{1+\eta}{2\sqrt{\eta}}\right)^4. \tag{13}$$

In Rayleigh-Bénard convection, the flow is driven by the temperature difference between two plates. This driving is captured in the Rayleigh number Ra. In Taylor-Couette flow, the flow is driven by the relative angular velocity of the two walls and the centrifugal pseudo-forces due to the curvature of the domain. This is captured by the Taylor number Ta, as defined in equation 4, which can also be written in terms of the quasi-Prandtl number and the average angular velocity gradient as:

$$Ta = \Phi \frac{d^4 r_a^2}{\nu^2} \left(\frac{\omega_i - \omega_o}{d}\right)^2. \tag{14}$$

One of the aspects of Taylor-Couette flow that makes it an interesting object for study, is the fact that it is a closed system. This means that an exact balance relation can be found between the driving forces and the energy dissipation, expressing the conservation of energy. In a statistially stationary flow, the global energy balance is given by: (Eckhardt et al., 2007)

$$\epsilon_u - \epsilon_{u,0} = \frac{\nu^3}{d^4} \Phi^{-2}\, Ta(Nu_\omega - 1). \tag{15}$$

Here ϵ_u is the volume-averaged energy dissipation rate, given by the following expression (Pope, 2000):

$$\epsilon_u \equiv 2\nu \left\langle s_{ij} s_{ij} \right\rangle_{V,t} = \nu \left\langle \frac{\partial u_i}{\partial x_j}\frac{\partial u_i}{\partial x_j} + \frac{\partial u_i}{\partial x_j}\frac{\partial u_j}{\partial x_i} \right\rangle_{V,t}, \tag{16}$$

where s_{ij} are the components of the strain rate tensor $\mathbf{S} = \frac{1}{2}\left(\nabla \mathbf{u} + \nabla \mathbf{u}^T\right)$ and $\langle \ldots \rangle_{V,t}$ expresses an average over the entire domain and over time.

For laminar Couette flow, the energy dissipation rate $\epsilon_{u,0}$ is analytically given by:

$$\epsilon_{u,0} = \nu \frac{r_i^2 r_o^2}{r_a^2} \left(\frac{\omega_i - \omega_o}{d}\right)^2 = \frac{\nu^3}{d^4} Ta \Phi. \tag{17}$$

Since the right and left hand sides of expression (15) can be calculated independently from the velocity field, we can define a quantity to measure the accuracy with which this identity is approximated numerically. In line with (Ostilla et al., 2013), we define the quantity Δ_ϵ as:

$$\Delta_\epsilon = \frac{\nu^3 d^{-4} \Phi^{-2}\, Ta(Nu_\omega - 1) + \epsilon_{u,0} - \epsilon_u}{\epsilon_u}. \tag{18}$$

The quantity Δ_ϵ describes the imbalance between energy input and energy dissipation in the system. Following the energy balance from expression (15), Δ_ϵ is analytically equal to zero due to energy conservation, but numerically it can deviate from the exact solution.

A summary of the formal relation between Taylor-Couette flow and Rayleigh-Bénard convection is shown in table 1.

	Rayleigh-Bénard convection	Taylor-Couette flow
Conserved quantity	$J = \langle u_z \Theta \rangle_{A,t} - \kappa \frac{\partial}{\partial z} \langle \Theta \rangle_{A,t}$	$J^\omega = r^3 \left[\langle u_r \omega \rangle_{A,t} - \nu \frac{\partial}{\partial r} \langle \omega \rangle \right]$
Nusselt number	$Nu = \frac{J}{J_0}$,	$Nu_\omega = \frac{J^\omega}{J_0^\omega}$,
	$J_0 = \frac{\kappa \Delta}{L}$	$J_0^\omega = -\nu \frac{2}{\eta(1+\eta)} r_i^3 \left(\frac{\omega_i - \omega_o}{d} \right)$
Prandtl number	$Pr = \frac{\nu}{\kappa}$	$\Phi = \left(\frac{1+\eta}{2\sqrt{\eta}} \right)^4$
Driving parameter	$Ra = \frac{\beta g \Delta L^3}{\kappa \nu}$	$Ta = \Phi \frac{d^4 r_a^2}{\nu^2} \left(\frac{\omega_i - \omega_o}{d} \right)^2$
Energy balance	$\epsilon_u - \epsilon_{u,0} = \frac{\nu^3}{L^4} RaPr^{-2}(Nu - 1)$,	$\epsilon_u - \epsilon_{u,0} = \frac{\nu^3}{d^4} Ta\Phi^{-2}(Nu_\omega - 1)$,
	$\epsilon_{u,0} = 0$	$\epsilon_{u,0} = \frac{r_i^2 r_o^2}{r_a^2} \nu \left(\frac{\omega_i - \omega_o}{d} \right)^2$

Table 1: Relation between Rayleigh-Bénard and Taylor-Couette flow, as developed by (Eckhardt et al., 2007). In RB flow, the temperature difference is given by Δ, with the distance between the plates given by L. The flow is driven by the Rayleigh number, where β is the thermal expansion coefficient, g the gravitional acceleration and κ the thermal diffusivity.

2.2 Numerical methods

As described in the introduction, the goal of this research is to adapt the TBFsolver by Cifani (Cifani et al., 2018) from a Cartesian geometry to a cylindrical geometry. This section focusses on the numerical methods that are used to solve the equations governing single-phase flow between two concentric cylinders and its implementation in the current TBFsolver.

The section is organised as follows. In subsection 2.2.1, the governing equations are presented, along with the general approach that will be used to solve the equations. The discretisations for the relevant equations are given in subsection 2.2.2. Subsection 2.2.4 gives an overview of the structure of the TBFsolver and describes the implementation of the numerical methods.

2.2.1 Mathematical formulation

The flow is simulated by solving the Navier-Stokes equation along with the incompressibility condition on the whole domain. The equations for a viscous, incompressible fluid can be written in conservative form as (Cifani et al., 2018):

$$\rho \left[\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) \right] = -\nabla p + \rho \mathbf{g} + \nabla \cdot (2\mu \mathbf{S}) + \mathbf{f} \tag{19a}$$

$$\nabla \cdot \mathbf{u} = 0. \tag{19b}$$

Here, $\mathbf{u} = (u_\theta, u_r, u_z)$ is the velocity field, p the pressure, t the time, ρ the fluid mass-density and μ the fluid viscosity. $\mathbf{S}$ is the strain rate tensor given by $\mathbf{S} = \frac{1}{2} \left(\nabla \mathbf{u} + \nabla \mathbf{u}^T \right)$, $\mathbf{g}$ is the gravitational acceleration and $\mathbf{f}$ is the sum of other external forces.

Time-marching algorithm

At high Reynolds numbers, it is often practical to employ a fractional time-marching algorithm such as the third-order Runge-Kutta method with implicit treatment of the viscous terms, which

reduces the restriction on the time step Δt compared to fully explicit methods. In two-phase flows, surface tension is added to the momentum equation. It can be shown that this term leads to the most severe restriction on the time step compared to the viscous and convective terms (Cifani et al., 2018). Since the aim of this research is to eventually extend this TC flow solver to a two-phase solver, the viscous and convective terms will be treated fully explicitly using the second order Adams-Bashfort scheme (AB2) (Wesseling, 2001). The AB2 scheme for solving a differential equation $\frac{dy}{dt} = f(y,t)$ is given by:

$$y_{n+1} = y_n + \frac{\Delta t}{2}(3f(y_n,t_n) - f(y_{n-1},t_{n-1})), \tag{20}$$

where y_n is the numerical solution at time t_n and Δt is the time step size. Since the two-phase problem will be modelled using the one-fluid formulation (Prosperetti and Tryggvason, 2007), it is important to develop a method for solving the Navier-Stokes equation with variable coefficients, as the density and viscosity will be highly non-uniform. We will therefore generalise the method to non-uniform and time-dependent density and viscosity fields.

At stage n of the discrete time marching, at time t_n, we start by calculation a provisional velocity $\mathbf{u}^*$ given by:

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + \gamma^n(C^n - V^n)\mathbf{u}^n - \xi_n(C^{n-1} - V^{n-1})\mathbf{u}^{n-1} = \alpha_n\left[-\frac{Gp^n}{\rho^{n+1/2}} + \frac{\hat{\mathbf{g}}}{\mathrm{Fr}^2}\right], \tag{21}$$

where $\Delta t \equiv t_{n+1} - t_n$ is the time step, $\gamma_n = 3/2$ and $\xi_n = -1/2$ are the AB2 coefficients, $\alpha_n \equiv \gamma_n + \xi_n$, C^n is the discrete operator $u \cdot \nabla$, V^n is the discrete viscosity operator $\frac{1}{\rho^n}\nabla \cdot (2\mu^n \nabla)$ and G is the discrete gradient operator ∇.

If we write the pressure at the next time level as $p^{n+1} = p^n + \phi$, the continuity equation for $\mathbf{u}^*$ requires the solution of the following discrete Poisson equation:

$$D\left(\frac{1}{\rho^{n+1/2}}G\phi\right) = D\left(\frac{\mathbf{u}^*}{\alpha_n \Delta t}\right), \tag{22}$$

where D is the discrete divergence operator $(\nabla\cdot)$

The solution of this equation allows us to correct the velocity and pressure fields according to:

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \alpha_n \Delta t\left(\frac{1}{\rho^{n+1/2}}G\phi\right) \tag{23}$$

$$p^{n+1} = p^n + \phi \tag{24}$$

By construction, the velocity at time level t^{n+1} is divergence-free, and the method remains second order accurate in time (Lee et al., 2001). For generality, we describe the numerical methods taking into account differences in the material properties so that the description is readily applicable to two-phase flow which will be discussed in the next chapter.

Spatial discretisation

The fields are discretised in cylindrical coordinates given by $\mathbf{x} = (\theta, r, z)$, where θ is the azimuthal direction, r is the radial direction and z is the axial direction. The flow field is solved in a cylindrical annular section, i.e. a section of the region between two cylinders that are both coaxial with the coordinate system.

Since the goal is to simulate wall-bounded turbulent Taylor-Couette flow, a natural choice is to have uniform grid spacing in the azimuthal and axial directions. In order to accurately resolve the turbulent structures near the wall, we use a non-uniform grid in the radial direction as described in (Vreman, 2014) in order to better resolve the turbulent boundary layer. This stretched grid

is smooth in the entire domain, with a smaller grid spacing near the walls. We employ periodic boundary conditions in the axial direction in order to reduce the effect of end plates in the axial direction (Ostilla-Mónico et al., 2014).

The fields are stored in a staggered arrangement. The vector fields (e.g. velocity and velocity flux) are stored on the respective cell faces, while the scalar fields (e.g. pressure, viscosity and density) are stored at the cell centers. This ensures a strong coupling between velocity and pressure, as well as an easier energy-conserving discretisation (Harlow and Welch, 1965; Meier et al., 1999). The grid configuration is shown schematically in figure 2.

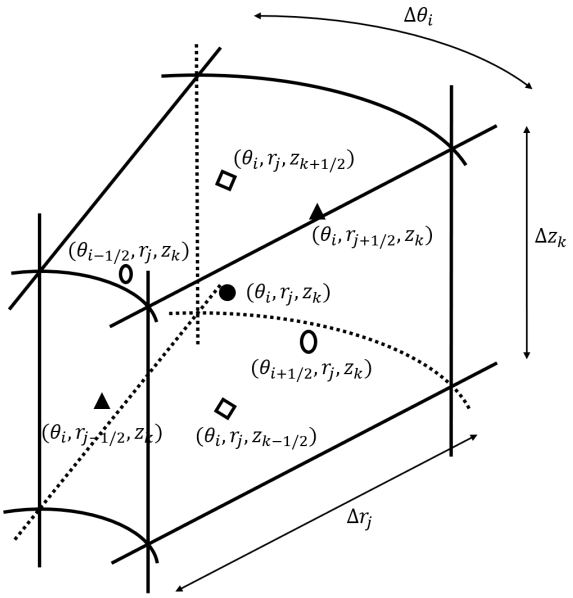


Figure 2: One grid cell in a staggered grid arrangement. Scalar quantities are defined at the cell centers (●), while the θ-, r- and z-components of vector quantities are defined at the respective face centers (○, ▲, □).

The following notation is used for discrete quantities:

$$\phi_{i,j,k} = \phi(\theta_i, r_j, z_k) \tag{25a}$$

$$\Delta\theta_i = \theta_{i+\frac{1}{2}} - \theta_{i-\frac{1}{2}}, \qquad i = 0, \frac{1}{2}, 1, \frac{3}{2}, \ldots, n_\theta \tag{25b}$$

$$\Delta r_j = r_{j+\frac{1}{2}} - r_{j-\frac{1}{2}}, \qquad j = 0, \frac{1}{2}, 1, \frac{3}{2}, \ldots, n_r \tag{25c}$$

$$\Delta z_k = z_{k+\frac{1}{2}} - z_{k-\frac{1}{2}}, \qquad k = 0, \frac{1}{2}, 1, \frac{3}{2}, \ldots, n_z \tag{25d}$$

2.2.2 Discretisation of convective and diffusive terms

In cylindrical coordinates, the incompressible Navier-Stokes equations (19a) along with the continuity equation are given by:

$$\rho\left[\frac{\partial u_\theta}{\partial t}+\frac{1}{r}\frac{\partial(u_\theta u_\theta)}{\partial\theta}+\frac{\partial(ru_r u_\theta)}{\partial r}+\frac{\partial(u_z u_\theta)}{\partial z}+\frac{u_r u_\theta}{r}\right] = -\frac{1}{r}\frac{\partial p}{\partial\theta}+\frac{1}{r}\frac{\partial\tau_{\theta\theta}}{\partial\theta}+\frac{\partial\tau_{\theta r}}{\partial r}+\frac{\partial\tau_{\theta z}}{\partial z}+2\frac{\tau_{\theta r}}{r}+f_\theta, \tag{26a}$$

$$\rho\left[\frac{\partial u_r}{\partial t}+\frac{1}{r}\frac{\partial(u_\theta u_r)}{\partial\theta}+\frac{\partial(ru_r u_r)}{\partial r}+\frac{\partial(u_z u_r)}{\partial z}-\frac{u_\theta u_\theta}{r}\right] = -\frac{\partial p}{\partial r}+\frac{1}{r}\frac{\partial\tau_{\theta r}}{\partial\theta}+\frac{\partial\tau_{rr}}{\partial r}+\frac{\partial\tau_{rz}}{\partial z}+\frac{(\tau_{rr}-\tau_{\theta\theta})}{r}+f_r, \tag{26b}$$

$$\rho\left[\frac{\partial u_z}{\partial t}+\frac{1}{r}\frac{\partial(u_\theta u_z)}{\partial\theta}+\frac{\partial(ru_r u_z)}{\partial r}+\frac{\partial(u_z u_z)}{\partial z}\right] = -\frac{\partial p}{\partial z}+\frac{1}{r}\frac{\partial\tau_{\theta z}}{\partial\theta}+\frac{\partial\tau_{rz}}{\partial r}+\frac{\partial\tau_{zz}}{\partial z}+\frac{\tau_{rz}}{r}+f_z. \tag{26c}$$

$$\frac{1}{r}\frac{\partial(ru_r)}{\partial r}+\frac{1}{r}\frac{\partial u_\theta}{\partial\theta}+\frac{\partial u_z}{\partial z}=0 \tag{26d}$$

Here, u_θ, u_r and u_z are the velocity components in cylindrical coordinates, f_θ, f_r and f_θ are the body forces, ρ is the density of the fluid and p is the pressure. The viscous stress tensor components τ_{ij}, $(i,j=\theta,r,z)$ are given by:

$$\tau_{\theta\theta}=2\mu\left(\frac{1}{r}\frac{\partial u_\theta}{\partial\theta}+\frac{u_r}{r}\right), \tag{27a}$$

$$\tau_{rr}=2\mu\left(\frac{\partial u_r}{\partial r}\right), \tag{27b}$$

$$\tau_{zz}=2\mu\left(\frac{\partial u_z}{\partial z}\right), \tag{27c}$$

$$\tau_{\theta r}=\tau_{r\theta}=\mu\left(\frac{\partial u_\theta}{\partial r}+\frac{1}{r}\frac{\partial u_r}{\partial\theta}-\frac{u_\theta}{r}\right), \tag{27d}$$

$$\tau_{\theta z}=\tau_{z\theta}=\mu\left(\frac{\partial u_\theta}{\partial z}+\frac{1}{r}\frac{\partial u_z}{\partial\theta}\right), \tag{27e}$$

$$\tau_{rz}=\tau_{zr}=\mu\left(\frac{\partial u_r}{\partial z}+\frac{\partial u_z}{\partial r}\right), \tag{27f}$$

where μ is the viscosity of the fluid. For the discretisation of equations 26a-26c, we employ the scheme developped by (Morinishi et al., 2004), which is a fully conservative method. This method is developed for cylindrical coordinates, where the grid can be non uniform in each direction. The discretisations are shown in the following sections.

Convective terms

The convective terms, given by $\nabla \cdot (\mathbf{uu})$, are discretised in the following way:

$$(\nabla \cdot (\mathbf{uu}))_\theta \rightarrow \left\{ \frac{1}{r}\frac{\delta}{\delta\theta}\left[\overline{(u_\theta)}^{\theta}\ \overline{(u_\theta)}^{\theta}\right] + \frac{1}{r}\frac{\delta}{\delta r}\left[r\,\overline{(u_r)}^{\theta}\ \overline{(u_\theta)}^{r}\right] + \frac{\delta}{\delta z}\left[\overline{(u_z)}^{\theta}\ \overline{(u_\theta)}^{z}\right]\right\} + \frac{1}{r}\overline{\left(\overline{(u_\theta)}^{\theta}\ \overline{(u_r)}^{r}\right)}^{\theta}, \tag{28a}$$

$$(\nabla \cdot (\mathbf{uu}))_r \rightarrow \frac{1}{\overline{(r\Delta r)}^{r}}\left\{ \frac{\delta}{\delta\theta}\left[\overline{(\Delta r u_\theta)}^{r}\ \overline{(u_r)}^{\theta}\right] + \frac{\delta}{\delta r}\left[\overline{(r u_r)}^{r}\ \overline{(u_r)}^{r}\right] + \frac{\delta}{\delta z}\left[\overline{(r\Delta r u_z)}^{r}\ \overline{(u_r)}^{z}\right]\right\} - \frac{1}{\overline{(r\Delta r)}^{r}}\overline{\left(\Delta r\,\overline{(u_\theta)}^{\theta}\ \overline{(u_\theta)}^{\theta}\right)}^{r}, \tag{28b}$$

$$(\nabla \cdot (\mathbf{uu}))_z \rightarrow \left\{ \frac{1}{r}\frac{\delta}{\delta\theta}\left[\overline{(u_\theta)}^{z}\ \overline{(u_z)}^{\theta}\right] + \frac{1}{r}\frac{\delta}{\delta r}\left[r\,\overline{(u_r)}^{z}\ \overline{(u_z)}^{r}\right] + \frac{\delta}{\delta z}\left[\overline{(u_z)}^{z}\ \overline{(u_z)}^{z}\right]\right\}. \tag{28c}$$

The discretisation is very similar to its Cartesian counterpart, with the addition of the terms in blue. These terms arise from the curvature of the domain, which follow directly form the ∇-operator in cylindrical coordinates. In the equations above, we use a second order approximation of the derivative using the notation $\frac{\delta}{\delta x}$. In the θ-direction, it is defined as the following:

$$\left.\frac{\delta\phi}{\delta\theta}\right|_{i,j,k} = \frac{\phi_{i+1/2,j,k} - \phi_{i-1/2,j,k}}{\theta_{i+1/2} - \theta_{i-1/2}}, \tag{29}$$

and similar definitions are used for the derivatives in the r- and z-directions. Since the convective terms are calculated at the face centers of the grid, the discrete derivatives require the velocity at the cell centers and edges. These are calculated using a second order symmetric interpolation in computational space, denoted by the $\overline{(\ldots)}^{x}$ operation. In the θ-direction, the interpolation of a quantity ϕ at $(\theta, r, z) = (\theta_i, r_j, z_k)$ is given as follows:

$$\left.\overline{(\phi)}^{\theta}\right|_{i,j,k} = \frac{1}{2}\left(\phi_{i-1/2,j,k} + \phi_{i+1/2,j,k}\right). \tag{30}$$

This method of discretisation of the convective terms in the Navier-Stokes equations is fully energy conservative and second order accurate in space (Morinishi et al., 2004).

Diffusive terms

Substituting the stress tensor components from equations (27a-27f) into the Navier Stokes equations (equations (26a-26c)) gives the following expressions for the viscous terms:

$$(\nabla \cdot 2\mu\mathbf{S})_\theta = \frac{1}{r}\frac{\partial}{\partial\theta}\left[2\mu\left(\frac{1}{r}\frac{\partial u_\theta}{\partial\theta} + \frac{u_r}{r}\right)\right] + \frac{1}{r}\frac{\partial}{\partial r}\left[\mu\left(\frac{\partial u_r}{\partial\theta} + r\frac{\partial u_\theta}{\partial r} - u_\theta\right)\right] + \frac{\partial}{\partial z}\left[\mu\left(\frac{\partial u_\theta}{\partial z} + \frac{1}{r}\frac{\partial u_z}{\partial\theta}\right)\right] + \frac{1}{r}\left[\mu\left(\frac{1}{r}\frac{\partial u_r}{\partial\theta} + \frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r}\right)\right], \tag{31a}$$

$$(\nabla \cdot 2\mu\mathbf{S})_r = \frac{1}{r}\frac{\partial}{\partial\theta}\left[\mu\left(\frac{1}{r}\frac{\partial u_r}{\partial\theta} + \frac{\partial u_\theta}{\partial r} - \frac{u_\theta}{r}\right)\right] + \frac{2}{r}\frac{\partial}{\partial r}\left[r\mu\frac{\partial u_r}{\partial r}\right] + \frac{\partial}{\partial z}\left[\mu\left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r}\right)\right] - 2\mu\left[\frac{1}{r^2}\frac{\partial u_\theta}{\partial\theta} + \frac{u_r}{r^2}\right], \tag{31b}$$

$$(\nabla \cdot 2\mu\mathbf{S})_z = \frac{1}{r}\frac{\partial}{\partial\theta}\left[\mu\left(\frac{\partial u_\theta}{\partial z} + \frac{1}{r}\frac{\partial u_z}{\partial\theta}\right)\right] + \frac{1}{r}\frac{\partial}{\partial r}\left[r\mu\left(\frac{\partial u_r}{\partial z} + \frac{\partial u_z}{\partial r}\right)\right] + 2\frac{\partial}{\partial z}\left[\mu\frac{\partial u_z}{\partial z}\right]. \tag{31c}$$

These expressions are very similar to their Cartesian counterparts. The only major difference is the addition of the terms in blue, which arise due to the curvature of the coordinate directions. The viscous terms are discretised according to the scheme developped by (Morinishi et al., 2004), and adapted by (Desjardins et al., 2008). The latter is implemented[1] and shown below for completeness:

$$
\begin{aligned}
(\nabla \cdot 2\mu \mathbf{S})_\theta \rightarrow & \frac{1}{r}\frac{\delta}{\delta\theta}\left[2\mu\left(\frac{1}{r}\frac{\delta u_\theta}{\delta\theta} + \frac{1}{r}\overline{(u_r)}^r\right)\right] + \frac{1}{r}\frac{\delta}{\delta r}\left[r\overline{\overline{\mu}^r}^\theta\left(\frac{\delta u_\theta}{\delta r} + \frac{1}{r}\frac{\delta u_r}{\delta\theta} - \frac{1}{r}\overline{(u_\theta)}^r\right)\right] \\
& + \frac{\delta}{\delta z}\left[\overline{\overline{\mu}^z}^\theta\left(\frac{\delta u_\theta}{\delta z} + \frac{1}{r}\frac{\delta u_z}{\delta\theta}\right)\right] + \frac{1}{r}\overline{\left[\overline{\overline{\mu}^r}^\theta\left(\frac{\delta u_\theta}{\delta r} + \frac{1}{r}\frac{\delta u_r}{\delta\theta} - \frac{1}{r}\overline{(u_\theta)}^r\right)\right]}^r,
\end{aligned} \tag{32a}
$$

$$
\begin{aligned}
(\nabla \cdot 2\mu \mathbf{S})_r \rightarrow & \frac{1}{r}\frac{\delta}{\delta\theta}\left[\overline{\overline{\mu}^r}^\theta\left(\frac{1}{r}\frac{\delta u_r}{\delta\theta} + \frac{\delta u_\theta}{\delta r} - \frac{1}{r}\overline{(u_\theta)}^r\right)\right] + \frac{1}{r}\frac{\delta}{\delta r}\left[2r\mu\frac{\delta u_r}{\delta r}\right] \\
& + \frac{\delta}{\delta z}\left[\overline{\overline{\mu}^z}^r\left(\frac{\delta u_r}{\delta z} + \frac{\delta u_z}{\delta r}\right)\right] - \frac{1}{r}\overline{\left[2\mu\left(\frac{1}{r}\frac{\delta u_\theta}{\delta\theta} + \frac{1}{r}\overline{(u_r)}^r\right)\right]}^r,
\end{aligned} \tag{32b}
$$

$$
\begin{aligned}
(\nabla \cdot 2\mu \mathbf{S})_z \rightarrow & \frac{1}{r}\frac{\delta}{\delta\theta}\left[\overline{\overline{\mu}^z}^\theta\left(\frac{1}{r}\frac{\delta u_z}{\delta\theta} + \frac{\delta u_\theta}{\delta z}\right)\right] + \frac{1}{r}\frac{\delta}{\delta r}\left[r\overline{\overline{\mu}^z}^r\left(\frac{\delta u_z}{\delta r} + \frac{\delta u_r}{\delta z}\right)\right] \\
& + \frac{\delta}{\delta z}\left[2\mu\frac{\delta u_z}{\delta z}\right].
\end{aligned} \tag{32c}
$$

Note that all the geometrical terms (in blue) tend to zero inversely proportional the local radius of curvature of the coordinate system, so we recover the Cartesian discretisation in the limit of vanishing curvature.

2.2.3 Fast Poisson solver

The next important part of the solver is the solution method for the Poisson equation (22). Since the domain is periodic in two directions, a natural choice is to utilize a 2D Fourier-based solver. However, this method requires a constant coeffient on the left-hand side of equation (22) while in the case of two-phase flow, the density may be highly non-uniform (Dodd and Ferrante, 2014). Following the ideas from Dong and Shen (Dong and Shen, 2012), the pressure gradient is split into two terms in order to create a constant-coefficient Poisson equation:

$$
\frac{1}{\rho^{n+1}}\nabla p^{n+1} = \frac{1}{\rho_0}\nabla p^{n+1} + \left(\frac{1}{\rho^{n+1}} - \frac{1}{\rho_0}\right)\nabla \hat{p}, \tag{33}
$$

where ρ_0 is a chosen constant reference mass density and the pressure gradient $\nabla\hat{p}$ is a suitable approximation of ∇p^{n+1}. Substituting this expression into equation (22) leads to the modified Poisson equation:

$$
\nabla^2 p^{n+1} = \nabla\cdot\left[\left(1 - \frac{\rho_0}{\rho^{n+\frac{1}{2}}}\right)\nabla\hat{p}\right] + \nabla\cdot\left(\rho_0\frac{\mathbf{u}^*}{\Delta t}\right). \tag{34}
$$

This modified pressure equation allows for the use of a Fast-Poisson solver. It is worth mentioning that particular care has to be taken in the computation of the pressure gradients in the presence of the jump due to surface tension. Here we follow the strategy developed in Cifani (2019), where the pressure gradient operator is combined with a corresponding volume fraction gradient at the interface. This allows to retain a grid convergent solution at high density ratios.

Since the domain is periodic the θ- and z-directions, we can transform equation (34) as follows:

$$
\left(\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial}{\partial r}\right) - \frac{k_\theta^2}{r^2} - k_z^2\right)\mathcal{F}(\phi) = \mathcal{F}(s), \tag{35}
$$

[1] A small typo appeared in (Desjardins et al., 2008) on the last line of equation 57 in that paper, where the term β directly after [should be omitted.

where $\mathcal{F}$ is the 2D Fourier transform operator in the (x, z)-plane, and s is the source term given by the right-hand side of equation (34).

The discrete wave numbers k are given by (van der Poel et al., 2015):

$$k_{x,i}^2 = \frac{2}{(\Delta x)^2}\left[1 - \cos\left(\frac{2\pi(i-1)}{N_x}\right)\right], \qquad i = 1, \ldots, N_x \tag{36}$$

for $x = \theta, z$. By discretising the term $\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial}{\partial r}\right)$ using a second-order approximation, equation (35) is reduced to a system of linear equations that can be solved using a tridiagonal matrix solver. The following discretisation is implemented:

$$\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial}{\partial r}\right)F_j \rightarrow \frac{1}{r_j}\frac{1}{\Delta r_j}\left(r_{j+\frac{1}{2}}\frac{F_{j+1}-F_j}{\Delta r_{j+\frac{1}{2}}} - r_{j-\frac{1}{2}}\frac{F_j - F_{j-1}}{\Delta r_{j-\frac{1}{2}}}\right) \tag{37a}$$

$$= a_m F_{j-1} + (-a_m - a_p)F_j + a_p F_{j+1}, \tag{37b}$$

where $a_m = \frac{r_{j-\frac{1}{2}}}{r_j}\frac{1}{\Delta r_j \Delta r_{j-\frac{1}{2}}}$ and $a_p = \frac{r_{j+\frac{1}{2}}}{r_j}\frac{1}{\Delta r_j \Delta r_{j+\frac{1}{2}}}$.

The transformed Poisson equation (equation (35)) can now be written as the linear system of equations $A\phi = b$ for every pair (i, k). The matrix A is constructed using the coefficients given by equation (37b) and the wave numbers from equation (36):

$$A_{m,n} = \begin{cases} a_p & \text{if } n = m+1 \\ -a_p - a_m - \frac{k_{x,i}^2}{r_n^2} - k_{z,k}^2 & \text{if } n = m \\ a_m & \text{if } n = m-1 \\ 0 & \text{otherwise} \end{cases}, \qquad m, n = 1, \ldots, N_y. \tag{38}$$

The system of equations is solved using an LU decomposition. After transforming the solution of equation (35) back to the computational space using an inverse 2D Fourier transformation, the velocity can be made divergence free using:

$$\mathbf{u}^{n+1} = \mathbf{u}^* - \alpha\Delta t\left(\frac{Gp^{n+1}}{\rho_0} + G\hat{p}\left(\frac{1}{\rho} - \frac{1}{\rho_0}\right)\right). \tag{39}$$

2.2.4 Implementation in TBFsolver

The TBFsolver is a Fortran-based code developed by Paolo Cifani. It is designed as a highly scalable solver for turbulent two-phase flow. However, there is an option to simulate single-phase flow. Since the modifications only regard the single-phase solver, we continue with this version of the TBFsolver. The general structure of the code is shown in algorithm 1. The structure for the modified TBFsolver is identical to the original solver. However, the relevant subroutines have been modified to account for cylindrical coordinates.

Before the start of the time marching loop, all the fields and structures are initialized. Depending on the settings in the specification files, the fields are either read from an input folder, or initialized to a specified value. After initialization, the time loop is started. At each time step, the time marching is executed. Additionally, the time step size can be updated based on the CFL-condition

Algorithm 1 TBFsolver - main

```
Initialize grid, fields and data types
while t < T do
    Update time
    Calculate provisional velocity (equation 21)
        Calculate body forces
        Solve momentum equation
    Solve Poisson equation (equation 22)
    Make velocity divergence free (equation 23)
    Update pressure (equation 24)
    Output statistics
end while
```

and the time restriction given by the viscous term. These conditions are given by:

$$(\Delta t)_{convective} \leq CFL \left[\max_{i,j,k} \left(\frac{u_r(\theta_i, r_j, z_k)}{\Delta r_j} + \frac{u_\theta(\theta_i, r_j, z_k)}{r_j \Delta \theta_i} + \frac{u_z(\theta_i, r_j, z_k)}{\Delta z_k} \right) \right]^{-1} \tag{40a}$$

$$(\Delta t)_{\text{viscous}} \leq \frac{1}{6\nu} \left(\min_{i,j,k} \Big(\min\{r_j \Delta \theta_i, \Delta r_j, \Delta z_k\} \Big) \right)^2, \tag{40b}$$

where the maximum allowed CFL number CFL can be set as a parameter. At the end of the time step, important statistics can be calculated and exported, such as the torque and the Nusselt number, as well as average velocity profiles.

Implementation of Navier-Stokes equations

The main changes in the code are made in the subroutines for the momentum and pressure equations. The calculation of the provisional velocity $\mathbf{u}^*$ from equation 21 is done using the operators in cylindrical coordinates. For the convection terms, the discretisation from equations 28a-28c are used. Compared to the original code, this meant introducing an occasional factor r or r^{-1}, as well as the addition of the terms in blue. For example, for the grid cell with indices (i, j, k), the term in blue in equation 28a is defined on the cell face in the θ-direction at $(i + \frac{1}{2}, j, k)$, and is written as:

$$\frac{1}{2r_j} \left[\frac{u^\theta_{i+\frac{3}{2},j,k} + u^\theta_{i+\frac{1}{2},j,k}}{2} \frac{u^r_{i+1,j+\frac{1}{2},k} + u^r_{i+1,j-\frac{1}{2},k}}{2} + \frac{u^\theta_{i+\frac{1}{2},j,k} + u^\theta_{i-\frac{1}{2},j,k}}{2} \frac{u^r_{i,j+\frac{1}{2},k} + u^r_{i,j-\frac{1}{2},k}}{2} \right], \tag{41}$$

where the abbreviation $u_{i,j,k} = u(\theta_i, r_j, z_k)$ has been used. Other terms are discretised following the same principles.

In the original TBFsolver, it was possible to select different discretisation schemes for the convective terms, namely central differencing, upwind differencing and QUICK. This is still possible in the θ- and z-directions, since the grid is uniform in these directions. However, in the radial direction, only the energy conserving scheme is implemented.

The Poisson solver requires less modification. Besides the modification of the gradient and divergence operators, only the coefficients for the tri-diagonal matrix solver have to be adjusted. Since the Poisson equation has constant coeffiencts, these can be pre-calculated at the start of the simulation, and therefore induce no further computational costs.

Calculation of angular momentum flux

An extra routine is added to the statistics module in order to calculate J^ω. Since the angular momentum flux (equation 7) is closely related to the torque on the walls, and the calculation

requires the radial derivative of the azimuthal velocity, it seems natural to define the quantity numerically on the cell edges in these directions. The discretisation is therefore chosen as:

$$(J^\omega)_{j+\frac{1}{2}} = r^3_{j+\frac{1}{2}} \left[\left\langle \overline{(u_r)}^{\theta}_{i,j+\frac{1}{2},k} \overline{\left(\frac{u_\theta}{r}\right)}^{r}_{i,j+\frac{1}{2},k} \right\rangle_{A,t} - \overline{\nu}^{r}_{i,j+\frac{1}{2},k} \left\langle \left(\frac{\delta\left(\frac{u_\theta}{r}\right)}{\delta r} \right)_{i,j+\frac{1}{2},k} \right\rangle_{A,t} \right]. \tag{42}$$

However, it is also possible to define this quantity on the cell centres, which is where the average velocity profiles are defined. This means that J^ω can be calculated a posteriori from the velocity profiles, using:

$$\widehat{(u_\theta)}_j = \left\langle \overline{(u_\theta)}^{\theta}_{i,j,k} \right\rangle_{A,t} \tag{43a}$$

$$\widehat{(u_r u_\theta)}_j = \left\langle \overline{(u_r)}^{r}_{i,j,k} \overline{(u_\theta)}^{\theta}_{i,j,k} \right\rangle_{A,t} \tag{43b}$$

$$(J^\omega)_j = r^3_j \left[\frac{\widehat{(u_r u_\theta)}_j}{r_j} - \nu_{i,j,k} \frac{\frac{\widehat{(u_\theta)}_{j+1}}{r_{j+1}} - \frac{\widehat{(u_\theta)}_{j-1}}{r_{j-1}}}{r_{j+1} - r_{j-1}} \right] \tag{43c}$$

These two methods will be compared later on in the next section.

Next to these large changes, many minor changes have been made in order to simulate in cylindrical coordinates. These changes include:

- change all occurences of discrete gradient operator;
- change all occurences of discrete divergence operator.
- offset of radial coordinates by r_i;
- change in volume of grid cells;
- addition of laminar Couette flow to initial condition options;
- addition of body forces in z-direction for gravity;

All of the above mentioned changes to the code do not change the structure of the solver. It is therefore to be expected that the solver will show a similar scalability with respect to the number of computational cores. The discretisation of the viscous and convective terms of the momentum equation require a larger amount of terms to be calculated. At large Reynolds numbers, most computational time is however spent on the Poisson equation, which is expected to have similar performance to the Cartesian solver. For all remaining discretisation schemes, the size of the computational stencil is the same size as in the original code, so a similar scalability is to be expected.

2.3 Torque measurements in low *Re* flows

Several test cases have been considered in order to asses the validity of the new Taylor-Couette flow solver. The first test cases consist of several laminar flow cases that have analytical solutions available for comparison. In the second set of test cases TC flow is simulated at Reynolds numbers near the transition from laminar flow to Taylor-vortex flow in order to verify the energy balance in a non-trivial flow field. Finally the convergence of the numerical results with respect to grid size is investigated in the case of Taylor-vortex flow at a Reynolds number of $Re = 1120$.

2.3.1 Constant velocity fields

The first step in validating the new Taylor-Couette flow solver is to simulate flows with analytical solutions for the velocity field. Three of such cases are selected in order to verify the solver along the three coordinate directions. Firstly, the flow is initialized with zero velocity, and non-moving walls. Secondly, solid body rotation is simulated, where both walls move at the same angular velocity, which is stable at low Reynolds numbers. Thirdly, a constant velocity in the axial direction is initialized. The results are given below.

Zero velocity field

For the first test, we adopt a domain with $\eta = 1/2$ and $\Gamma = 2$. The flow is simulated on a grid with resolution $(128 \times 64 \times 64)$. The grid is uniform in the azimuthal and axial direction, and non-uniform in the radial direction. The velocity and pressure fields are initialised at a uniform value of zero, with homogeneous boundary conditions. The viscosity is set to $\mu = 1/50$, which sets the maximum time step for the viscous term to $dt \approx 1.3 \cdot 10^{-4}$.

After simulating the flow for approximately 50,000 time steps, the velocity field is evaluated. The magnitude of the velocity was still equal to zero up to machine precision. This validation indicates that there are no unwanted source terms in the code.

Solid body rotation at low *Re*

The third test is to examine the stability of solid body rotation of the fluid. Using the same domain and grid sizes as in the previous test cases, we set the velocity of the inner wall to $V_i = 1$ and the velocity of the outer wall to $V_o = 2$, which gives both walls the same angular velocity. The velocity field is initialized to zero in the axial and radial directions, and in the azimuthal direction to:

$$u_\theta(r) = r, \qquad 1 \leq r \leq 2. \tag{44}$$

The viscosity is set to $\mu = 1/50$, for which the flow should remain laminar. The flow is simulated for one full rotation.

After one rotation, the velocity in the axial direction remained zero. In the radial direction, velocity fluctuations of $O(10^{-16})$ developed. Similarly sized deviations from the laminar profile developed in the azimuthal direction.

Homogeneous axial flow

In order to verify the stability of the axial velocity component, the flow is initialized with a nonzero uniform velocity U in the axial direction. In order for the flow to remain stationary, we set the fluid velocity at the boundary to $U\hat{\mathbf{z}}$. This can be seen as solid body translation in the z-direction.

Using the same grid and parameters as in the previous test case, the flow is simulated for 6 seconds approximately 50,000 time steps at $u_z = 1$, after which the velocity field is compared with the analytical solution.

The velocity in the azimuthal and radial directions have remained equal to zero up to machine precision. In the axial direction, the velocity is still equal to $u_z = 1$ up to machine precision.

2.3.2 Validation at low Reynolds number

The second step in validating the new TC solver is to compare it with the torque measurements from (Pirrò and Quadrio, 2008). This procedure is also used in (Ostilla et al., 2013). The goal of the test is to find the correct Nusselt number scaling with Reynolds number near the critical Reynolds number, which is the point where the transition from laminar flow to Taylor-vortex flow occurs. We use a stationary outer cylinder, and a fixed non-zero velocity of the inner cylinder. For the current geometry, where $\eta = 0.5$, the critical Reynolds number is $Re_c \approx 68.2$ (Fasel and Booz, 1984).

The comparison is done using measurements of $Nu_\omega(r)$ at different low values of Re_i near the critical Reynolds number. Similar to (Pirrò and Quadrio, 2008) and (Ostilla et al., 2013), the

aspect ratio is set to $\Gamma = 2$, the ratio of the radii to $\eta = 1/2$, and a rotational symmetry of order four is assumed to limit the size of the simulations. The flow is simulated at a resolution of $(N_\theta \times N_r \times N_z) = (32 \times 64 \times 64)$, with uniform grid spacing along the θ- and z-directions, and hyperbolic-tangent-type clustering along the radial direction. The domain is decomposed into two parts along the θ-direction and four along the z-direction, using a total of eight MPI processes.

For all the low-Re simulations, the initial velocity field is set to laminar Couette flow with an added disturbance of $O(10^{-12})$. The development of Taylor vortices is accelerated by starting at a lower viscosity, and gradually lowering it to the set value. The flow is then simulated for about 100 rotations of the inner cylinder in order to reach a statistically stationary flow. After reaching that state, the relevant statistics are calculated using a time-average of around eight rotations.

The results for the Nusselt numbers are shown in table 2. The values from the present study agree with previous results up to at least three digits. The values recorded in table 2 correspond to the average values of the $Nu_\omega(r)$ profiles, given by $Nu_\omega = \langle Nu_\omega(r) \rangle_r$. The values for Δ_ϵ are also reported in the table. This discrepancy in the energy balance is larger at increasing Reynolds number, but remains below the requirement $\Delta_\epsilon < 1\%$ for numerical convergence (Ostilla et al., 2013).

Re_i	Nu_ω (present study)	Δ_ϵ (present study)	Nu_ω (Ostilla et al., 2013)	Nu_ω (Pirrò and Quadrio, 2008)	Nu_ω (Fasel and Booz, 1984)
60	1.0000	$1.12 \cdot 10^{-4}$	1.0005	1.0000	1.0000
68	1.0000	$1.12 \cdot 10^{-4}$	1.0006	1.0000	1.0000
70	1.0240	$1.94 \cdot 10^{-4}$	1.0235	1.0237	1.0238
75	1.0840	$3.94 \cdot 10^{-4}$	1.0835	1.0833	1.0834
80	1.1371	$5.66 \cdot 10^{-4}$	1.1375	1.1371	1.1372

Table 2: Comparison of Nu_ω at low Re_i and $Re_o = 0$.

In figure 3, the $Nu_\omega(r)$ profiles are shown, normalised by their average values. It shows that there is a slight variation in $Nu_\omega(r)$, however, these variations are limited to approximately 0.01%. The profiles for $Re = 60$ and $Re = 68$, corresponding to laminar TC flow, collapse onto the same curve. This indicates that there is a very small error in the simulation which causes the Nusselt number to fluctuate around the analytical solution of $Nu_\omega(r) = 1$ for the laminar cases.

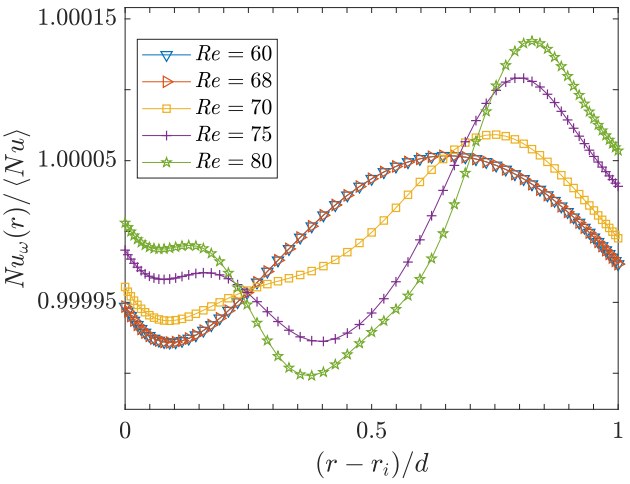


Figure 3: Variation in Nusselt number as function of radial distance at different values of Re_i. The Nusselt numbers are normalized by their average values.

For all cases, the flow was initialized by introducing Taylor vortices, and then letting the flow develop for some time. In the case with $Re = 68$, the flow was not yet developed after 100 rotations, as the flow is very close to the instability, but the normalised variations as a function of

the radial coordinate in $Nu_\omega(r)$ were still an order of magnitude higher than for the other cases. After around 100 more rotations however, the $Nu_\omega(r)$ had collapsed to the $Re = 60$ case, indicating that the flow has sufficiently converged to laminar Couette flow.

The calculation of Nu is based on a time averaging over a sufficiently long period. In figure 4 the results for Nu_ω are shown using different time periods for the time averaging. The convergence rate can be calculated using Richardson extrapolation, from which it follows that Nu_ω converges proportional to the inverse of the time averaging length. This is a higher rate than the expected convergence proportional to $\sqrt{N}$ for a sequence of N samples, which might be caused by the fact that we consider a laminar Taylor-vortex flow.

Remarkably, by averaging only over half the time that it takes the inner cylinder to complete a full rotation, the result is already within 0.01% of its value after 10 rotations of time-averaging.

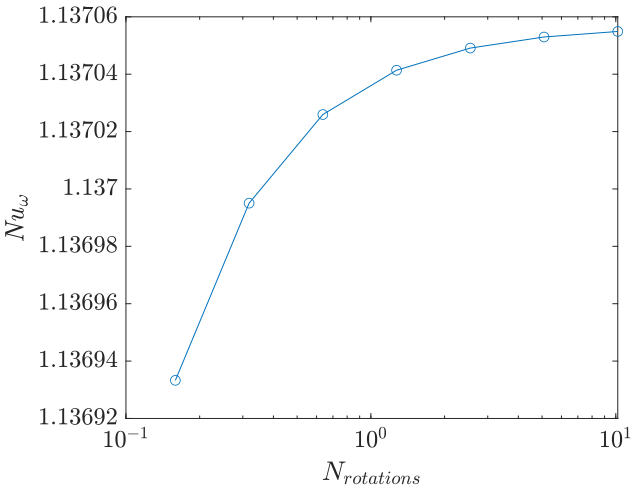


Figure 4: Dependence of Nu_ω on the length of the time averaging for the $Re = 80$ test case. The horizontal axis shows the number of rotations of the inner cylinder during the time averaging process.

2.3.3 Resolution tests

The next step in the validation of the code is to compare the results at different grid resolutions. For this test, the value of Δ_ϵ will be used to determine the degree to which a the simulation has converged, following the approach from (Ostilla et al., 2013). We simulate a flow at a Reynolds number of $Re_i = 1120$ and a stationary outer cylinder. The domain size is set to $\eta = 5/7$ and $\Gamma = 2\pi$ to create a case identical to the case reported in (Ostilla et al., 2013).

For this large gap geometry, i.e., small value of η, the wavelength of a Taylor-vortex pair is around $\lambda \approx 1.09d$ (Chouippe et al., 2014). This means that three Taylor-vortex pairs can form over the axial extent of the domain. However, depending on the initial conditions, it is also possible for four vortex pairs with a smaller wavelength to develop in this domain (Ostilla-Mónico et al., 2015). The actual size and number of the vortices has a large influence on the Nusselt number as shown extensively in (Brauckmann and Eckhardt, 2013). They showed that an extra vortex pair can increase the torque by around 20% at low Taylor numbers.

In order to force the emergence of only three vortex pairs, the flow is initialized as laminar Taylor-Couette flow, with an added sinusoidal disturbance of order $O(10^{-2})$ with a wavelength of $\frac{2}{3}\pi$ in the z-direction.

For the grid resolution test, four different grid sizes are used, as shown in table 3. After initialisation, the flow is simulated for around 60 rotations in order for the flow to develop. The Nusselt number is calculated using a time-average of around 6 rotations.

Using this approach, the results for grids A and B were as expected, with three Taylor-vortex pairs

developing and small fluctuations in the $Nu_\omega(r)$ profile. However, despite the initial condition, four Taylor-vortex pairs emerged in cases C and D, as is shown in figure 6. This led to overestimations of Nu_ω of about 6%, consistent with the results from (Ostilla et al., 2013). In order to generate three Taylor-vortex pairs, the developed flow state from case B was used as an initial condition for cases C* and D*, after which the flow was simulated for approximately 20 rotations of the inner cylinder. The final results for the Nusselt numbers are shown in table 3.

Case	$n_\theta \times n_r \times n_z$	Nu_ω (present study)	Δ_ϵ (present study)	Nu_ω (Ostilla et al., 2013)	Δ_ϵ (Ostilla et al., 2013)
A	$128 \times 64 \times 64$	5.0915	0.0467		
B	$192 \times 96 \times 96$	4.7576	0.0232	4.83540	0.0949
C*	$256 \times 128 \times 128$	4.7329	0.0146	4.46000	0.0174
D*	$384 \times 192 \times 192$	4.7122	0.0072	4.47765	0.0065

Table 3: Resolution test at $\eta = 5/7$, $\Gamma = 2\pi$ and $L_x = 2\pi$, where $Re = 1120$. The star attached to the case name indicates that a different initial condition is used.

Similar to the results from (Ostilla et al., 2013), a coarse grid results in an overestimation of the Nusselt number. Table 3 also shows Δ_ϵ, calculated from equation 18. This quantity is analytically equal to zero, and checks the relative difference between energy transport (via Nu_ω) and energy dissipation (via ϵ_u). This value is positive for all cases, further indicating an overestimation of Nu_ω. However, both Nu_ω and Δ_ϵ show convergence with grid resolution.

As mentioned in the previous section, two methods can be used for calculating the Nusselt number. The first method is to calculate the angular momentum flux at the cell corners (equation 42), and the second method calculates this quantity a posteriori from the average velocity profiles (equation 43c). Even though the choice of discretisation has hardly any influence on the value of Nu_ω, the first method shows significantly less dependence on r. This is shown in figure 5, where the profiles of $Nu_\omega(r)$ are shown for the four test cases, using both formulations. The results show that the radial dependence of $Nu_\omega(r)$ is approximately one order of magnitude lower when using method 1 compared to using method 2.

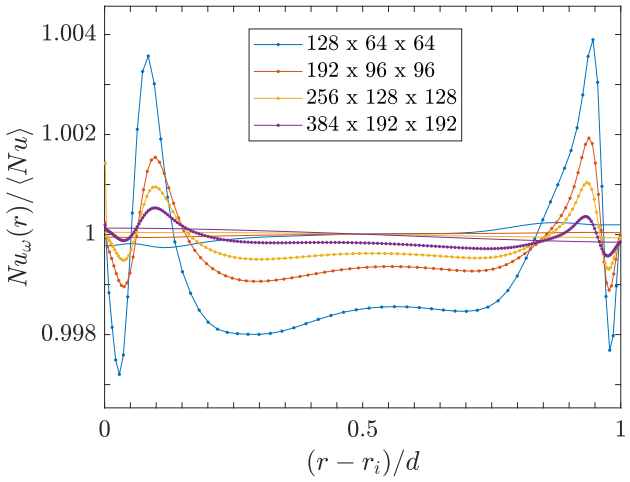
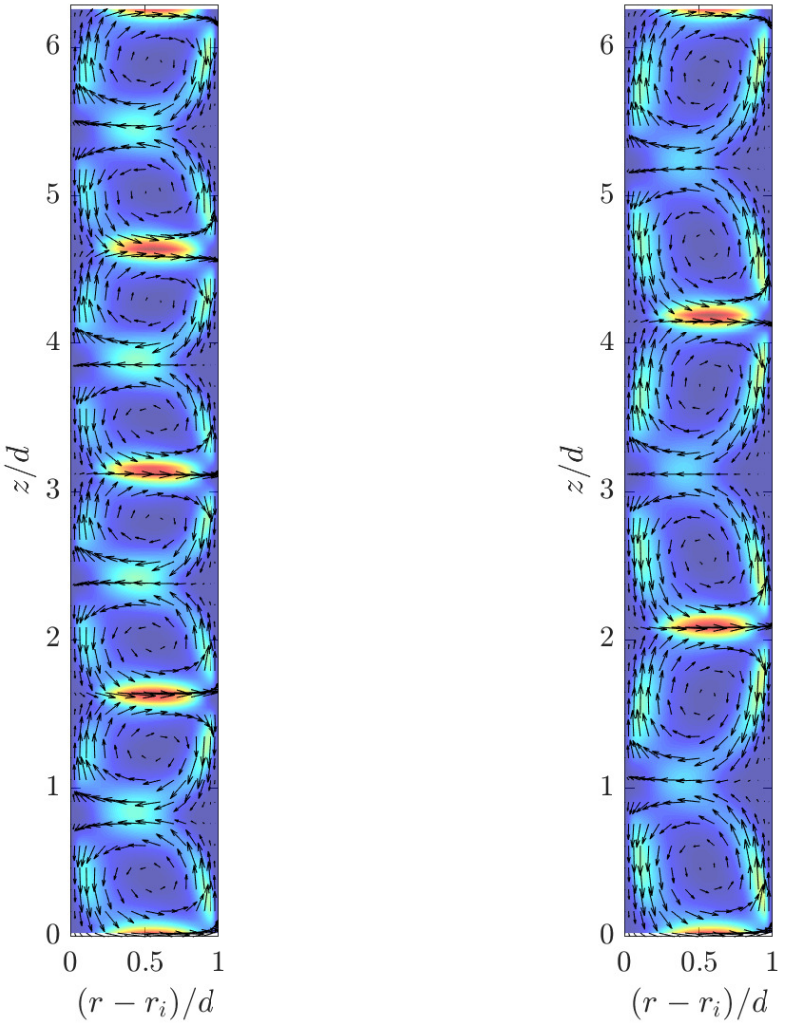


Figure 5: Radial dependence of $Nu_\omega(r)$ at different grid resolutions. Results are shown for method 1 (using equation 42, shown as solid lines) and for method 2 (using equation 43c, shown as dotted lines).

According to (Ostilla et al., 2013), case C and D represent fully resolved flows. However, for these cases there is still a discrepancy in Nu_ω of approximately 5% with the present study. This might be attributed to the different numerical method used, since Ostilla uses the discretisation given by (Verzicco and Orlandi, 1996). However, the main difference between the two methods is the difference in initial conditions. It might be the case that the flow in cases C* and D* are not

strictly stationary, but in a transition between a flow state with 3 and 4 vortex pairs respectively. A longer simulation time for these cases might give insight into this development, which is subject of further study.



(a) (b)

Figure 6: Cross-section of the domain in the (r, z)-plane. The color indicates the magnitude of the 2D velocity field, while the arrows give the magnitude and local direction of the velocity field. (a) Shows the velocity field for case C after 60 rotation using laminar TC flow with sinusoidal disturbance as initial condition. (b) Shows the velocity field of case C after 20 rotations using the developed flow from case B as initial condition.

2.4 Conclusion

In this chapter, an extension has been made to the TBFsolver, which makes it possible to simulate single-phase flows in cylindrical coordinates. The Navier-Stokes equations are solved using the fully conservative scheme developed by Morinishi (Morinishi et al., 2004) for the convective and diffusive fluxes. The pressure is solved using the original spectral method, with modifications to the wavenumbers and to the coefficients of the transformed Poisson equation.

As a first validation step, the flow solver is tested for rigid body flow between two concentric

cylinders at a low Reynolds number, as well as solid body rotation. In all test cases, the resulting velocity field was equal to the analytical solutions up to machine precision.

Secondly, the flow solver was tested for Taylor-Couette flow near the critical Reynolds number, which is defined as the transition between laminar Couette flow, and Taylor-Vortex flow. At several values of the inner cylinder Reynolds number near the critical point (with a stationary outer cylinder), the angular momentum flux was measured, expressed in the non-dimensional Nusselt number. This study shows an agreement in this quantity with previous literature up to around 0.05%.

The Nusselt number is based on the time-averaged velocity field. For the $Re_i = 80$ case, the convergence of the Nusselt number with respect to the length of time averaging was investigated. It was found that the Nusselt number converges linearly proportional to the inverse of the time-averaging length, and a sufficient convergence using a time averaging window of one revolution of the inner cylinder.

At a higher Reynolds number, the spatial convergence was investigated. With a stationary outer cylinder, and an inner cylinder Reynolds number of $Re_i = 1120$, the Taylor-vortex motion was simulated at four different grid sizes. It was shown that calculating the angular momentum flux at the cell corners decreases its radial dependence by approximately an order of magnitude compared to an a posteriori calculation from the cell-centered average velocity fields. At larger grid sizes, both the radial dependence of the angular momentum flux and the discrepancy in the energy balance decrease. However, the Nusselt number does not seem to converge to the value reported Ostilla et al (Ostilla et al., 2013), with a 5% difference at the largest grid size. Further investigation must be done into this discrepancy. The difference in Nusselt number might be attributed to a gradual transition between a flow state with 3 and 4 vortex pairs respectively. Another test for the new TC flow solver would be to investigate the scalability of the code, as we expect the code to perform similar to the original code, that is, linear scaling up to $O(10^4)$ computational cores.

The framework of the developed flow solver is general enough to allow for the simulation of single- and two-phase flows, with necessary changes only at the level of the input parameters. Compared to the single-phase flow, the addition of a gas phase only requires an interface reconstruction and advection method, and the addition of surface tension. As illustrated above, the Poisson equation solver is, in fact, already suited for inhomogeneous density and viscosity fields. Currently, the TBFsolver is equipped with a VOF module, combined with the Generalized Height Function method for Cartesian systems. The next step in the development of the code will be to map these methods to the cylindrical geometry such that two-phase TC flow can be simulated with the TBFsolver.

Part II

Two-phase Taylor-Couette flow

3 Numerical methods

The next part of this thesis is dedicated to the extension of the code from the previous part to two-phase flows. In this section, we introduce the methods for simulating bubbles in a cylindrical domain. In section 3.1, the governing equations are introduced. An outline of the numerical implementation is given in section 3.2. In section 3.3 the interface reconstruction and advection method is described. Finally, the numerical method for calculating surface tension is shown in section 3.4.

3.1 Mathematical model

The two-phase flow consists of two fluids that can have different intrinsic parameters such as mass density and dynamic viscosity. Due to surface tension between the two fluids, a sharp interface between the two emerges. The system is modelled using the one-fluid formulation, which means that both fluids are simulated using one set of equations. For the simulation of this flow, the non-dimensional incompressible Navier-Stokes equations are used:

$$\nabla \cdot \mathbf{u} = 0 \tag{45a}$$

$$\rho \left[\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u}\mathbf{u}) \right] = -\nabla p + \frac{1}{\mathrm{Fr}^2} \rho \hat{\mathbf{g}} + \frac{1}{\mathrm{Re}} \nabla \cdot (2\mu \mathbf{D}) + \frac{1}{\mathrm{We}} \sigma \kappa \mathbf{n} \delta(n). \tag{45b}$$

Here, $\mathbf{u} = (u_\theta, u_r, u_z)$ is the velocity field, ρ is the fluid mass density, μ is the dynamic viscosity, $\mathbf{D}$ is the deformation tensor, σ is the surface tension coefficient for the fluid interface, κ is the curvature of the interface, $\mathbf{n}$ is the unit normal to the interface and $\delta(n)$ is the Dirac delta function, with n being the local coordinate to the interface in the direction of $\mathbf{n}$. This localizes the surface tension at the fluid interface. In the current setup for TC flow, gravity points down in the axial direction. Furthermore, Fr is the Froude number, Re is the Reynolds number and We is the Weber number.

Note that these equations describe one fluid with highly non-homogeneous intrinsic parameters. In both phases individually, density and viscosity are assumed to be constant in time. The distribution of the phases is given by the indicator function $f(\mathbf{r}, t)$. This function f is equal to zero if $(\mathbf{r}, t)$ lies in one phase, and is equal to one if it is in the other. Therefore, f acts as a Heaviside function.

In this study, we simulate dispersed bubbles in a carrier fluid, so the two phases will be referred to as the carrier phase, characterized by material properties (ρ_f, μ_f), and gas phase, characterized by (ρ_b, μ_b). The indicator function f is defined to equal 1 inside a bubble, and 0 outside the bubble. This allows us to define the density and viscosity fields in the following way:

$$\rho(\mathbf{r}, t) = f(\mathbf{r}, t)\rho_g + (1 - f(\mathbf{r}, t))\rho_l \tag{46}$$

$$\mu(\mathbf{r}, t) = f(\mathbf{r}, t)\mu_g + (1 - f(\mathbf{r}, t))\mu_l, \tag{47}$$

where ρ_g, ρ_l and μ_g, μ_l are the mass density and dynamic viscosity of the dispersed phase and the carrier phase respectively.

The value of the indicator function at a specific location represents the phase of the fluid particle at that location. Since we assume that no phase transitions take place and that the two phases are immiscible, the value of f should remain constant for each fluid particle. Furthermore, in order to simulate meaningful interactions between bubbles, we employ the multiple-marker formulation (Coyajee and Boersma, 2009). An individual indicator function f_i is assigned to each bubble. This prevents automatic numerical coalescence when bubbles approach within one grid cell of each other (Cifani et al., 2018). In a simulation with N bubbles, this leads to the following advection equations:

$$\frac{\partial f_i}{\partial t} + \mathbf{u} \cdot \nabla f_i = 0, \qquad i = 1, \ldots, N \tag{48}$$

From these individual indicator functions we can reconstruct the common volume fraction field f as:

$$f = \max_{i=1,\dots,N} f_i \tag{49}$$

which ensures $0 \leq f \leq 1$. Similarly, since surface tension forces are derived from the indicator function, we can calculate the total surface tension field from the contributions of the individual bubbles as:

$$\mathbf{f}_\sigma = \sum_{i=1}^{N} \sigma \kappa_i \mathbf{n}_i \delta(n_i) \tag{50}$$

In the next section, the numerical methods are described for solving this set of governing equations for bubble-laden flow.

3.2 Numerical methods for two-phase flow

Since we employ a one-fluid formulation, the governing equations for two-phase flow are similar to those of single-phase flow as described in section 2.2.1. However, two-phase flow requires us to include a surface tension term in the Navier-Stokes equations. In the TBFsolver, time-marching is carried out using a fractional-step method. For reasons that are discussed later in this section, we employ the second-order Adams-Bashfort scheme (AB2) for the diffusion and convection terms in equation 45b, while the Crank-Nicolson scheme is used for surface tension. This implies that every time step consists of only one stage.

This problem setting is very similar to the single-phase problem described in the previous chapters of this thesis. The added challenge is to compute the surface tension field as described by the last term in equation 45b. This requires the calculation of the curvature field as well as the position of the interface.

At every time step n, the goal is to solve the governing equations on t_{n+1} given the full state at t_n. At the start of the time step, the advection of the indicator function is carried out, which requires the solution of the N advection equations (48). This is a computationally increasingly expensive procedure as the number of bubbles N increases. However, since the support of each of the functions f_i is localized to the interface of the bubble, each advection equation needs only be solved on a subdomain containing the bubble interface as equations 48 reduce to $f_i^n = f_i^{n+1}$ away from the interface. Therefore, each indicator function f_i is discretized only on a box surrounding the bubble that is co-moving with the bubble. More details on this can be found in (Cifani et al., 2018).

After finding f^{n+1}, a provisional velocity field $\mathbf{u}^*$ can be calculated similar to the single-phase method from equation 21, adding the surface tension term:

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + \frac{3}{2}(C^n - V^n)\mathbf{u}^n - \frac{1}{2}(C^{n-1} - V^{n-1})\mathbf{u}^{n-1} = -\frac{Gp^n}{\rho^{n+1/2}} + \frac{\hat{\mathbf{g}}}{\mathrm{Fr}^2} + \left(\frac{\kappa \mathbf{n} \delta(n)}{\rho \mathrm{We}}\right)^{n+\frac{1}{2}}, \tag{51}$$

where $\Delta t \equiv t_{n+1} - t_n$ is the time step, C^n is the discrete convection operator $u \cdot \nabla$, V^n is the discrete diffusion operator $\frac{1}{\rho^n}\nabla \cdot (2\mu^n \nabla)$, G is the discrete gradient operator ∇ and $\hat{\mathbf{g}}$ is the direction of gravity. The surface tension term and the mass density $\rho^{n+\frac{1}{2}}$ are calculated using a linear interpolation between f^n and f^{n+1} according to the Crank-Nicolson scheme. Note that the AB2 scheme is used for the time stepping of the convective and diffusive terms. The velocity at time $(n+1)$ can be found by imposing the incompressibility condition on the provisional velocity. This method has been described in section 2.2.3.

Due to the cylindrical geometry of the problem at hand, the domain is discretised on a structured grid in cylindrical coordinates. Since the TC flow is a type of wall-bounded flow, it is natural to employ uniform grid spacing along the streamwise and axial directions. Along the radial coordinate, the grid spacing is refined near the walls using a hyperbolic-tangent-type clustering.

In section 2.2.4, the time step restrictions due to the explicit treatment of the convective and diffusive terms were discussed. The presence of surface tension introduces another physical time into the system (Brackbill et al., 1992). This time scale is based on the wave velocity of capillary waves which leads to a CFL condition, as the advection algorithm is only shown to be accurate for CFL numbers approximately up to 1. The time step restriction due to capillary waves is given by:

$$(\Delta t)_{capillary} = \sqrt{\frac{d^3\rho}{4\pi\sigma}}, \tag{52}$$

where d is the smallest grid cell size. For high Reynolds numbers, this is the most restricting limitation on the numerical procedure. This observation leads to the choice of the AB2 scheme rather than a multi-staged time-marching procedure. Since the time step is dictated by the capillary waves instead of the convective and diffusive terms, the computational cost of a multi-stage method no longer outweighs its enlarged numerical stability region.

The full procedure for solving the two-phase flow is shown in the pseudo-algorithm below.

Algorithm 2 Outline of two-phase TC flow solver

```
Initialize grid, fields and data types
Calculate initial VOF field f based on initial bubble distribution
Construct blocks around each bubble
while t < T do
    for all blocks do
        Solve VOF field advection equation
        Compute curvature
        Compute surface tension
    end for
    Construct global VOF field and surface tension
    Update material properties
    Compute surface tension
    Calculate provisional velocity u*
    Solve Poisson equation
    Update velocity and pressure
end while
```

Compared to the single-phase flow, we are tasked with several extra steps in the numerical algorithm. Firstly, we have to solve the advection equations 48 in order to provide us with $f^{n+\frac{1}{2}}$. Secondly, the surface tension term in equation 51 must be approximated. The next two sections detail these two challenges. The advection algorithm is described in section 3.3. Next, the method for calculating the surface tension force is described in section 3.4.

3.3 Interface reconstruction and advection

On a rectangular domain, the advection equation (48) can be solved by using a geometrical VOF scheme. In this method, the interface between the fluids is first approximated using pre-defined shapes. Based on this shape, the flux of the interface can be calculated which advects the interface.

The TBFsolver employs the most widely used geometrical VOF method, the piecewise linear interface calculation (PLIC). In this method, we can approximate the interface in a grid cell by

$$\mathbf{n} \cdot \mathbf{x} = \alpha, \tag{53}$$

where $\mathbf{x}$ is the position vector in the grid cell, $\mathbf{n}$ is the normal vector to the interface and α is a parameter related to the intersection points of the interface with the coordinate axes. This equation describes a 3D linear surface, which divides the interface into two regions, associated with the two fluids respectively. The parameter α is chosen such that the volume of these regions matches the known gas volume fraction in that cell. This leads to an algebraic procedure, detailed in (Scardovelli and Zaleski, 2000).

Once the equation for the plane is known, the flux can be evaluated. This method is illustrated in figure 7. During a time Δt, the shaded area inside the dashed lines will advect to the neighbouring grid cell due to the fluid velocity at the interface. The advected volume can be found algebraically from the equations for the plane. In the figure, the shaded area represents the flux $(uf)_{i+1/2,j}$.

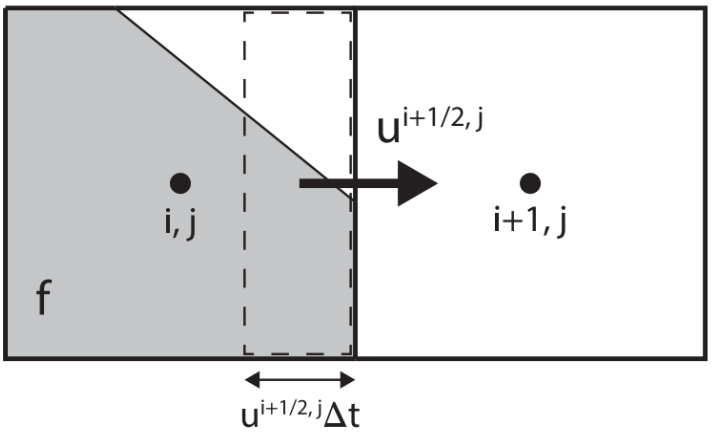


Figure 7: Illustration of geometrical flux in 2D. The shaded area represents the area occupied by one phase in cell (i, j). During a time Δt, the shaded area inside the dashed line will advect to cell $(i + 1, j)$ by the velocity at the interface $u_{i+1/2,j}$. Figure taken from (Cifani et al., 2016).

The full 3D advection is carried out by the split-operator method proposed by (Puckett et al., 1997). With this method, advection is performed sequentially in the three directions. A combination of correction terms and a permutation of the order in which the advection directions are chosen lead to a minimal error in the conservation of mass.

The goal is now to employ the VOF-PLIC method in the case of cylindrical grid cells. The geometrical method hinges on the fact that there is an algebraic relation between the equation for a planar interface, and the volume it cuts a grid cell into. It is clear that we cannot use this method to reconstruct the interface with planes in the physical space, where we have cylindrical cells.

We can follow the approach from (Wang et al., 2012), which suggests reconstructing the interface on the computational domain. Since we use a structured grid in cylindrical coordinates, computational cells are in fact rectangular. Similar to the Cartesian method, we can reconstruct the interface in each grid cell. The challenge remains to calculate the flux of the indicator function based on the interface flux in the computational space.

By definition, we can write the divergence operator as follows:

$$\nabla \cdot (\mathbf{u}f) = \lim_{\Delta V \to 0} \frac{1}{\Delta V} \oint_S (\mathbf{u}f) \cdot \mathbf{n} dS, \tag{54}$$

for any control volume V that is bounded by the surface S. We denote the volume fraction flux $\mathbf{u}f$ as $\mathbf{q}$. A discrete approximation of equation 54 in cylindrical coordinates reads:

$$\frac{1}{r_j \Delta\theta_i \Delta r_j \Delta z_k} \left(\Delta F_\theta + \Delta F_r + \Delta F_z\right), \tag{55}$$

where the fluxes ΔF_θ, ΔF_r and ΔF_z are given by:

$$\Delta F_\theta = q_\theta(\theta_{i+\frac{1}{2}}, r_j, z_k)\Delta r_j \Delta z_k - q_\theta(\theta_{i-\frac{1}{2}}, r_j, z_k)\Delta r_j \Delta z_k \tag{56a}$$
$$\Delta F_r = q_r(\theta_i, r_{j+\frac{1}{2}}, z_k) r_{j+\frac{1}{2}} \Delta\theta_i \Delta z_k - q_r(\theta_i, r_{j-\frac{1}{2}}, z_k) r_{j-\frac{1}{2}} \Delta\theta_i \Delta z_k \tag{56b}$$
$$\Delta F_z = q_z(\theta_i, r_j, z_{k+\frac{1}{2}}) r_j \Delta\theta_i \Delta r_j - q_z(\theta_i, r_j, z_{k-\frac{1}{2}}) r_j \Delta\theta_i \Delta r_j \tag{56c}$$

Taking the limit $\Delta V \to 0$ in equation 55 gives the well known divergence formula in cylindrical coordinates:

$$\nabla \cdot \mathbf{q} = \frac{1}{r}\frac{\partial r q_r}{\partial r} + \frac{1}{r}\frac{\partial q_\theta}{\partial \theta} + \frac{\partial q_z}{\partial z} \tag{57}$$

Equations 55-56 allow us to translate the volume fraction flux in computational space (q_θ, q_r, q_z) to physical space. The full 3D discretisation of the advection equation using the split-operator method now reads:

$$f^* = f^n - \Delta t \frac{1}{r}\frac{\partial (u_\theta f^n)}{\partial \theta} + \Delta t f^* \frac{\partial u_\theta}{\partial \theta} \tag{58a}$$
$$f^{**} = f^* - \Delta t \frac{1}{r}\frac{\partial (r u_r f^*)}{\partial r} + \Delta t f^{**} \frac{1}{r}\frac{\partial (r u_r)}{\partial r} \tag{58b}$$
$$f^{***} = f^{**} - \Delta t \frac{\partial (u_z f)}{\partial z} + \Delta t f^{***} \frac{\partial u_z}{\partial z} \tag{58c}$$
$$f^{n+1} = f^{***} - \Delta t \left(f^* \frac{1}{r}\frac{\partial u_\theta}{\partial \theta} + f^{**} \frac{1}{r}\frac{\partial (r u_r)}{\partial r} + f^{***} \frac{\partial u_z}{\partial z} \right) \tag{58d}$$

where f^*, f^{**} and f^{***} represent the volume fraction field after each advection step in one direction, and in the last step f^{n+1} is calculated by subtracting the correction terms. This scheme does not guarantee the conservation of mass, as each intermediate advection step can cause f to go below zero or exceed one (Rider and Kothe, 1998). In practice, this error has not turned out to be an issue, as it is small compared to other sources (Popinet, 2009).

3.4 Curvature and surface tension

In the TBFsolver, surface tension is modelled using the continuum-surface-force (CSF) approach (Brackbill et al., 1992). It uses an approximation to the delta function in the surface tension term of equation 45b:

$$\sigma\kappa\delta(n) \approx \sigma\kappa\nabla f, \tag{59}$$

which allows us to determine the surface tension using the volume fraction field f.

Analytically, the mean curvature κ of the interface is given by:

$$\kappa = -\nabla \cdot \mathbf{n} \tag{60}$$

We will now introduce the height-function (HF) method to calculate the normal of the interface from the volume fraction field. We define a height function h which describes the local interface position with respect to some coordinate. For example, when the normal to the interface points mainly in the vertical direction z, we can define the interface as the surface:

$$z = h(r, \theta), \tag{61}$$

where h describes the position of the interface on the (θ, r)-plane. Numerically, the height function can be approximated by evaluating the volume fractions in the grid cells along the z-direction. The total volume in such a column is the sum of the volume fractions. We find the height function as the height of a column with the same total volume. This procedure is illustrated in figure 8.

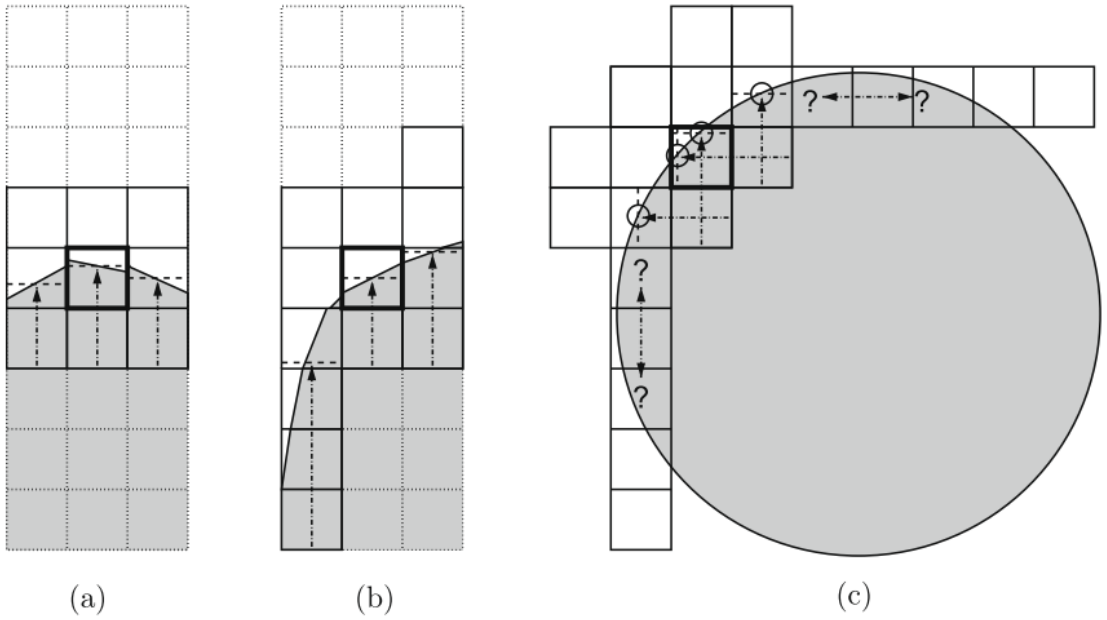


Figure 8: Illustration of the HF method to find a local approximate the position of the interface, adapted from (Popinet, 2009). The height function is calculated around the highlighted grid cell. In case (a) and (b) an adaptive stencil size is used to determine the height function, indicated by the dashed lines. For highly curved surfaces (c), the HF method can fail.

Let us define the surface of the fluid interface on the coordinate system (x_1, x_2, x_3). Without loss of generality, we can locally represent the surface using a height function along the x_3 axis. The surface is then defined as:

$$x_3 - h(x_1, x_2) = 0 \tag{62}$$

The height function in a column along the x_3-direction is calculated as follows:

$$h((x_1)_i, (x_2)_j) \approx h_{i,j} = \sum_{l=k-N}^{k+N} f_{i,j,l} \cdot (\Delta x_3)_l, \tag{63}$$

where $2N + 1$ is the stencil size around the interface. In practice however, the summation is done iteratively starting at the center cell $l = 0$, and then moving up and down the column until an empty ($f_{i,j,k} = 0$) or full ($f_{i,j,k} = 1$) cell is encountered; that is, when the next cell is a not part of the interface. This procedure is shown in figure 8b, where the stencil extends to grid cells that satisfy $f = 1$ or $f = 0$. Using the HF at several grid points, we can calculate the normal vector $\mathbf{n}$ to the interface using the local gradient of the height function following the procedure developed in (Brackbill et al., 1992):

$$\mathbf{n} = \begin{bmatrix} -\frac{\partial h}{\partial x_1} & -\frac{\partial h}{\partial x_2} & 1 \end{bmatrix}^T \tag{64}$$

We can find an expression for the mean curvature by substituting this into equation 60. The mean curvature is given by:

$$\kappa = -\nabla \cdot \mathbf{n} = -\nabla \cdot \left(\frac{\nabla(x_3 - h(x_1, x_2))}{|\nabla(x_3 - h(x_1, x_2))|} \right) \tag{65}$$

Using the quotient rule, we find:

$$\kappa = \frac{|\nabla x_3 - \nabla h(x_1, x_2)| \left(\nabla \cdot (\nabla h - \nabla x_3)\right) - \nabla \left(|\nabla x_3 - \nabla h(x_1, x_2)|\right) \cdot (\nabla h - \nabla x_3)}{|\nabla x_3 - \nabla h(x_1, x_2)|^2} \tag{66}$$

In curvilinear orthogonal coordinates, the unit vectors are orthogonal, so we can write $\nabla\phi = \sum_i \frac{1}{d_i}\frac{\partial \phi}{\partial x_i}\hat{\mathbf{e}}_i$ for a scalar field ϕ, where d_i are the Lamé coefficients. In the case of cylindrical coordinates, we have $d_\theta = r$ and $d_r = d_z = 1$, while for Cartesian coordinates, $d_x = d_y = d_z = 1$. Furthermore, we use the notation $\frac{\partial \phi}{\partial x_i} = \phi_{x_i}$. Substituting these expression gives:

$$\kappa = \frac{\sqrt{\frac{1}{d_3^2} + |\nabla h|^2}\ (\nabla^2 h - \nabla^2 x_3) - \nabla\left(\sqrt{\frac{1}{d_3^2} + |\nabla h|^2}\right) \cdot (\nabla h - \nabla x_3)}{\frac{1}{d_3^2} + |\nabla h|^2} \tag{67a}$$

$$= \frac{\left(\frac{1}{d_3^2} + |\nabla h|^2\right)(\nabla^2 h - \nabla^2 x_3) - \frac{1}{2}\nabla\left(\frac{1}{d_3^2} + |\nabla h|^2\right) \cdot \left(\nabla h - \frac{1}{d_3}\hat{\mathbf{e}}_3\right)}{\left(\frac{1}{d_3^2} + |\nabla h|^2\right)^{3/2}}, \tag{67b}$$

where we use the following identities for general orthogonal coordinate systems:

$$|\nabla h|^2 = \frac{1}{d_1^2}(h_{x_1})^2 + \frac{1}{d_2^2}(h_{x_2})^2 \tag{68a}$$

$$\nabla^2 h = \frac{1}{d_1 d_2 d_3}\left[\left(\frac{d_2 d_3}{d_1}h_{x_1}\right)_{x_1} + \left(\frac{d_1 d_3}{d_2}h_{x_2}\right)_{x_2} + \left(\frac{d_1 d_2}{d_3}h_{x_3}\right)_{x_3}\right] \tag{68b}$$

In Cartesian coordinates, the expression for the mean curvature simplifies significantly. Substituting $d_1 = d_2 = d_3 = 1$, we find the following equations:

$$\kappa = \frac{h_{11}(1 + h_2^2) + h_{22}(1 + h_1^2) - 2h_1 h_2 h_{12}}{(1 + h_1^2 + h_2^2)^{3/2}}. \tag{69}$$

Due to the symmetry of the coordinate systems, this expression is valid for height functions along all three directions (Evrard et al., 2020).

When we evaluate the mean curvature for the cylindrical coordinate system, we find three expressions, since the equation depends on the direction along which the height function is defined. The expressions are shown below:

$$\kappa_{\theta = h(r,z)} = \frac{h_{rr}\left(\frac{1}{r^2} + h_z^2\right) + h_{zz}\left(\frac{1}{r^2} + h_r^2\right) - 2h_r h_z h_{rz} + \frac{h_r}{r}\left(\frac{2}{r^2} + h_r^2 + h_z^2\right)}{\left(\frac{1}{r^2} + h_r^2 + h_z^2\right)^{3/2}}, \tag{70a}$$

$$\kappa_{r = h(\theta,z)} = \frac{h_{\phi\phi}(1 + h_z^2) + h_{zz}(1 + h_\phi^2) - 2h_\phi h_z h_{\phi z} - \frac{1}{r}(1 + 2h_\phi^2 + h_z^2)}{(1 + h_\phi^2 + h_z^2)^{3/2}}, \tag{70b}$$

$$\kappa_{z = h(\theta,r)} = \frac{h_{\phi\phi}(1 + h_r^2) + h_{rr}(1 + h_\phi^2) - 2h_\phi h_r h_{\phi r} + \frac{h_r}{r}(1 + 2h_\phi^2 + h_r^2)}{(1 + h_\phi^2 + h_r^2)^{3/2}}, \tag{70c}$$

where we have introduced the short-hand notation $h_\theta = rh_\phi$, $h_{\theta,r} = rh_{\phi,r}$ and $h_{\theta\theta} = r^2 h_{\phi\phi}$. The derivation can be found in appendix A. The expressions above are used to calculate the curvature at a grid cell on the interface. The derivatives of the height function are calculated using central difference formulas.

The standard height function method allows a relatively simple implementation compared to other higher-order methods such as parabolic fitting (PROST) (Renardy and Renardy, 2002). However, as seen in figure 8c, the standard height function method can fail to find a valid height along a column when the interface is highly curved. In cases where the height function method fails, Popinet has suggested implementing a hierarchical structure that allows us to fall back to a higher-order method, which leads to the generalised height function method (GHF) (Popinet, 2009).

The TBFsolver employs PROST in cases where the standard HF method fails. In this case, the valid heights from the HF method are collected. As long as there are at least six valid interface positions, a paraboloid can be fitted to the interface positions using a minimization function (Renardy and Renardy, 2002). The curvature of a paraboloid can be calculated directly from the coefficients of its equation in Cartesian coordinates. Therefore, we construct the paraboloid using the interface positions in Cartesian coordinates, which can be calculated using coordinate transform on the numerical height function.

The GHF method is studied extensively in (Cifani et al., 2018). It is shown that for well-resolved bubbles with 15 to 20 grid points per diameter, the standard HF method fails in only approximately 2% of the interface cells, in which cases the computationally more expensive PROST method is employed. In the next section, the accuracy and stability of the GHF method in cylindrical coordinates are tested.

4 Single bubble dynamics

In this section, the numerical methods described in section 3 are tested using several test cases. These cases are split into two groups. The first set of tests cover the VOF advection algorithm by analyzing bubbles transport on an imposed velocity field. In the second group of cases, we simulate the full dynamics of the two-phase flow by studying a single bubble rising in an initially stationary flow due to buoyancy.

4.1 Pure advection test cases

We verify the advection algorithm by analysing the advection of a spherical bubble on an assumed velocity field. In all these cases, the velocity field is chosen such that the advected body will return to its initial shape after some fixed time T, which allows us to define the numerical error that is introduced by the advection algorithm. Similar to the existing literature, we define three error measures that can be evaluated at any time t_n during the simulation:

$$E_1(t_n) = \sum_i \left| f_i(t_n) - \overline{f}_i(t_n) \right| V_i \tag{71a}$$

$$E_2(t_n) = \frac{\sum_i \left| f_i(t_n) - \overline{f}_i(t_n) \right|}{\sum_i \overline{f}_i(t_n)} \tag{71b}$$

$$E_3(t_n) = \frac{\sum_i (f_i(t_n) - \overline{f}_i(t_n))}{\sum_i \overline{f}_i(t_n)} \tag{71c}$$

where the summation is over all grid cells, and V_i is the physical volume represented by grid cell i. In these definitions, $f(t_n)$ is the numerical volume fraction field at time level t_n, and $\overline{f}(t_n)$ is the exact solution at time level t_n.

4.1.1 Solid body rotation

In the first test case, we study solid body rotation of a bubble. The case is sketched in figure 9. The domain consists of two concentric cylinders with radii $(R_i, R_o) = (R_i, R_i + 1)$. A bubble with diameter $d_b = 0.6$ is initialized, centered at $(\theta, r, z) = (0, R_i + \frac{1}{2}, 0)$, and is subjected tot he following velocity field:

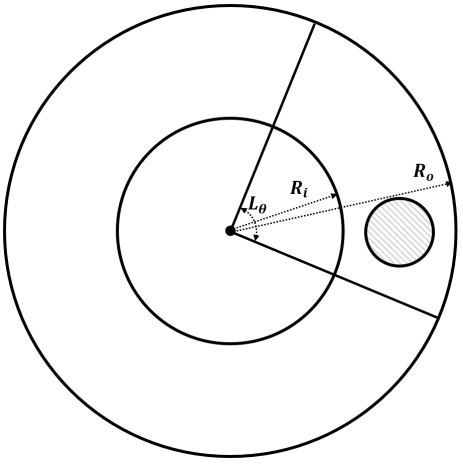


Figure 9: Sketch for solid body rotation (chapter 2.1)

$$\mathbf{u}(\mathbf{r}, t) = u_\theta(r, t)\hat{\theta} = \frac{r}{R_o}\cos(\pi t) \qquad 0 \le t \le 1 \tag{72}$$

We set the time step to a fixed value during the simulation, which is determined by the maximum CFL number. In order to guarantee numerical stability for the advection scheme wet the CFL number to 0.1. This gives a maximum time step of $\Delta t = 0.1\frac{R_o L_\theta}{n_\theta}$, where L_θ is the angular size of the domain in the azimuthal direction, and n_θ is the number of grid points along this direction.

The geometry of this test can be characterized by the ratio of between the inner and outer cylinder radius $\eta = R_i/R_o = R_i/(R_i+1)$, which represents the curvature of the domain. In the limit $\eta \to 1$, we obtain a Cartesian grid. The bubble is simulated at four different values of η, and additionally, we perform the test in the original Cartesian TBFsolver.

A summary of the parameters in each simulation is shown in table 4. In each case, the domain size in the r- and z-directions are $L_r = L_z = 1$, while the angle L_θ depends on the cylinder radii. It is chosen such that the arc length along the outer cylinder is 2. The grid points are uniformly spaced along each direction. The cases are named A_n such that $R_o = 2^n$. The case A_∞ refers to the simulation in the original Cartesian formulation of the TBFsolver. Additionally, we perform grid refinement on case A_1.

Case	R_i	η	$(n_\theta \times n_r \times n_z)$	$[\theta_{min}, \theta_{max}]$	Δt $(\cdot 10^{-3})$
A_1	1	$\frac{1}{2}$	$(64 \times 32 \times 32)$	$[-\frac{1}{4}, \frac{3}{4}]$	3.125
A_2	3	$\frac{3}{4}$	$(64 \times 32 \times 32)$	$[-\frac{1}{8}, \frac{3}{8}]$	3.125
A_3	7	$\frac{7}{8}$	$(64 \times 32 \times 32)$	$[-\frac{1}{16}, \frac{3}{16}]$	3.125
A_4	15	$\frac{15}{16}$	$(64 \times 32 \times 32)$	$[-\frac{1}{32}, \frac{3}{32}]$	3.125
A_∞	-	1	$(64 \times 32 \times 32)$	$[x_{min}, x_{max}] = [-\frac{1}{2}, \frac{3}{2}]$	3.125
$A_{1,b}$	1	$\frac{1}{2}$	$(128 \times 64 \times 64)$	$[-\frac{1}{4}, \frac{3}{4}]$	1.563
$A_{1,c}$	1	$\frac{1}{2}$	$(256 \times 128 \times 128)$	$[-\frac{1}{4}, \frac{3}{4}]$	0.781

Table 4: Description of test cases for solid body rotation.

During the simulations, the error values are tracked over time. The reference field $\overline{f}(t_n)$ are calculated by constructing a volume of fluid field based on a spherical bubble centred at the exact solution for the centre of the bubble.

The results of the simulations with varying η are shown in figure 10. This figure shows an increasing error $E_2(t)$. Looking at the results for cases A_1, A_2, A_3 and A_4, we note that the results converge to the result on a Cartesian grid A_∞ as expected.

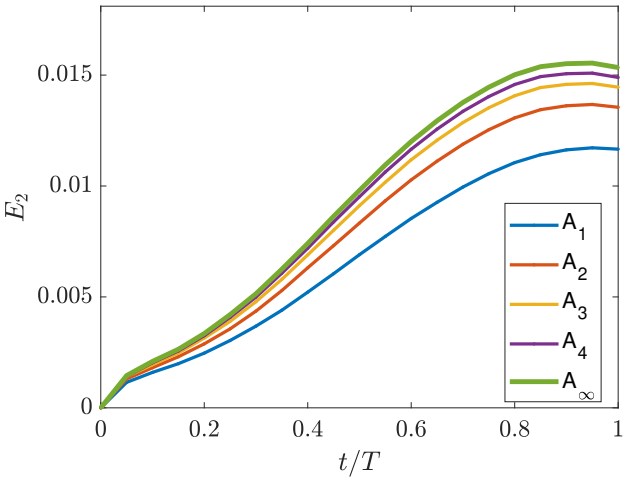


Figure 10: Results of error $E_2(t)$ for cases A_1, A_2, A_3, A_4 and the Cartesian result A_10.

It is interesting to note that the error in the volume fraction field is smaller for the cases with higher curvature. A possible explanation is that the velocity of the centre of the bubble is lower for the cases with a smaller R_i, since velocity increases linearly with distance to the origin, and is equal to $\cos(\pi t)$ at the outer wall in all cases. With an overall lower velocity, we find a lower error in these cases.

The results for the error E_1 are qualitatively the same as E_2 since they are very similar. The error E_3 measures the relative mass loss during the advection. For all test cases, this error is $E_3(t) < 10^{-9}$, which emerges from the initial numerical approximation of the indicator function. This indicates that mass is preserved up to a large precision. Note however that the advection algorithm does not guarantee exact mass conservation.

Using multiple grid refinements, we can study the convergence of the advection algorithm. Figure 11 shows the result for $\eta = 1/2$ at three different grid sizes as mentioned in table 4. At a constant CFL number, the time step is also decreased proportionally to the grid size. However, despite a smaller time step and smaller grid size, the results only show linear convergence with the grid cell length. Theoretically, we expect linear convergence with time as well as at least linear convergence in space.

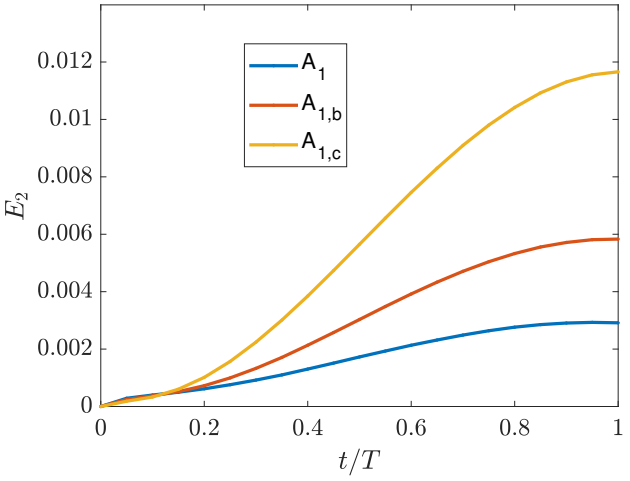


Figure 11: Time-evolution of error E_2 at $\eta = 1/2$ for different grid sizes as described in table 4.

The error is dominated by the error in the time-stepping method. This is shown in figure 12, where the case A_1 has been simulated at different CFL numbers, which give different time steps. As expected, it shows linear convergence, with only a slower convergence at a CFL number of 0.025. This indicates that the errors in figure 11 are also dominated by the time-stepping error.

As shown by these test cases, the error in the volume fraction field is around 1% for bubbles with 20 grid cells across at a CFL number of 0.1. However, the main cause for the error is the explicit Euler time-stepping method for the velocity field. Furthermore, the results for the cylindrical cases converge to the results for the Cartesian case when considering a smaller curvature of the domain.

4.1.2 Radial translation

As a second test case, we impose a divergence-free velocity field in the radial direction, given by:

$$\mathbf{u}(\mathbf{r}, t) = u_r(r, t)\hat{\mathbf{r}} = \frac{R_i}{2r}\cos(\pi t) \qquad 0 \leq t \leq 1 \tag{73}$$

This velocity field can be interpreted as the flow originating from a source at the line $r = 0$. The velocity at the inner cylinder is equal to $u = 1/2$. We initialize a bubble with diameter 0.3 at $(\theta, r, z) = (0, R_i + 0.25, 0)$. Similar to the previous cases, the domain is bounded by concentric

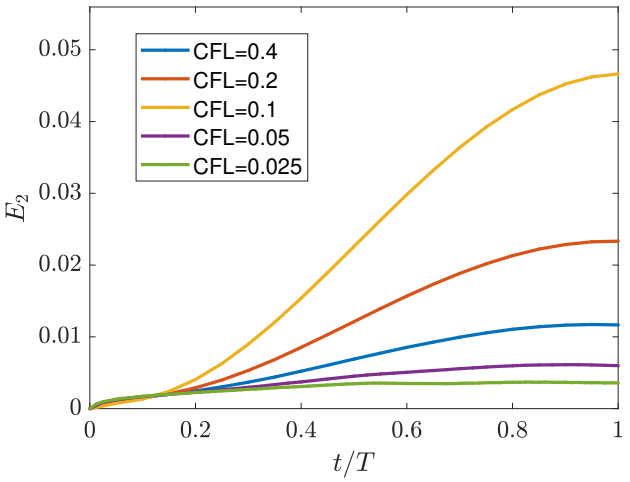


Figure 12: Time-evolution of error E_2 for case A_1 at varying CFL numbers.

cylinders with radii R_i and $R_o = R_i + 1$. Furthermore $L_z = 1/2$ and L_θ depends on the value of R_i such that the arc length along the outer wall is 1. Again, we include results from the original TBFsolver, denoted by case B_∞. The different test cases are shown in table 5.

Case	r_i	η	$(n_\theta \times n_r \times n_z)$	$[\theta_{min}, \theta_{max}]$	dt $(\cdot 10^{-3})$
B_1	1	$\frac{1}{2}$	$(64 \times 64 \times 32)$	$[-\frac{1}{4}, \frac{1}{4}]$	3.125
B_2	3	$\frac{1}{2}$	$(64 \times 64 \times 32)$	$[-\frac{1}{8}, \frac{1}{8}]$	3.125
B_3	7	$\frac{1}{2}$	$(64 \times 64 \times 32)$	$[-\frac{1}{16}, \frac{1}{16}]$	3.125
B_4	15	$\frac{1}{2}$	$(64 \times 64 \times 32)$	$[-\frac{1}{32}, \frac{1}{32}]$	3.125
B_∞	-	1	$(32 \times 64 \times 32)$	$[x_{min}, x_{max}] = [-\frac{1}{4}, \frac{1}{4}]$	3.125
B_2	1	$\frac{1}{2}$	$(128 \times 128 \times 64)$	$[-\frac{1}{4}, \frac{1}{4}]$	1.563
B_2	1	$\frac{1}{2}$	$(256 \times 256 \times 128)$	$[-\frac{1}{4}, \frac{1}{4}]$	0.781

Table 5: Description of test cases for radial translation.

For these simulations, there is no reference volume fraction field at all time steps. Instead, we can only calculate the error at $t = T$. The velocity has a component $\cos(\pi t)$, which causes the bubble to return exactly to its initial position. A snapshot of case B_1 at $t = 0.3$ is shown in figure 13 as an illustration of the bubble deformation. The bubble is compressed in the radial direction due to the decreasing velocity in this direction. The results for $E_2(T)$ as a function of the curvature η are shown in figure 14. We see a clear convergence of the error to that of the Cartesian grid when we increase the distance to the origin. Since the advection velocity is twice as low as in the previous test case, we observe an overall lower error, since the error in the velocity due to the explicit time-stepping is also lower.

4.1.3 Laminar Couette flow

In order to further analyze the behaviour of deformable bubbles, we impose laminar circular Couette flow, which is given by

$$\mathbf{u} = u_\theta(r)\hat{\theta} = Ar + \frac{B}{r}, \quad \text{where} \quad A = \frac{\omega_o - \eta^2\omega_i}{1-\eta^2}, \qquad B = \frac{(\omega_i - \omega_o)r_i^2}{1-\eta^2}, \tag{74}$$

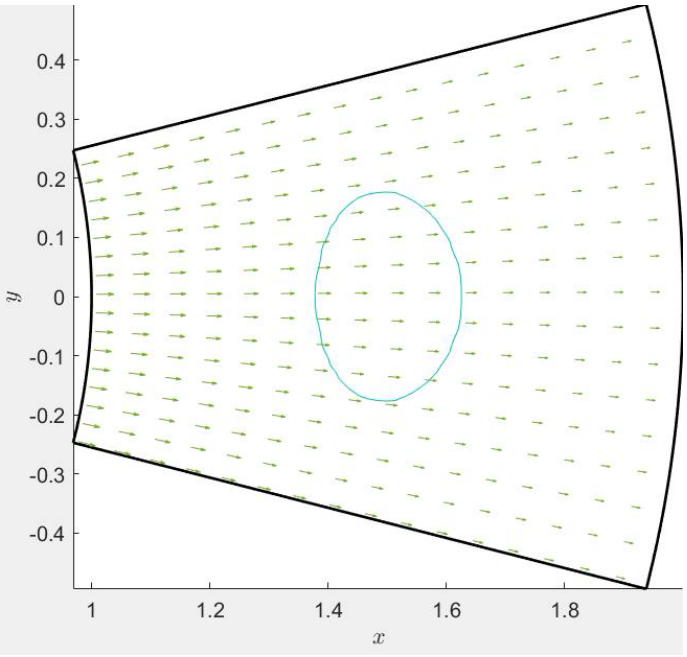


Figure 13: Snapshot of case B_1 at $t = 0.3$. The bubble is deformed by the imposed velocity field.

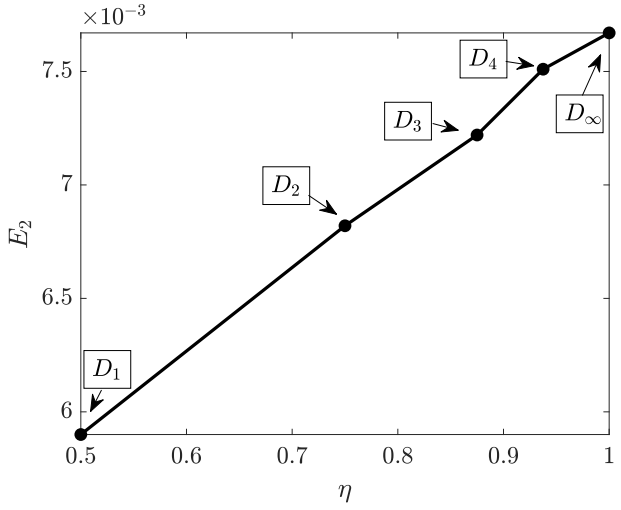


Figure 14: Error $E_2(T)$ for test cases B_1, B_2, B_3, B_4 and the Cartesian solver B_∞.

For the next cases, we take $\omega_o = -\omega_i = \cos\left(\frac{\pi t}{T}\right)/r_o$, where t is the time, which represents counter-rotating cylinders. such that it returns to its initial position at $t = T$.

We initialize a bubble with diameter 0.6 at $(\theta, r, z) = (0, R_i + \frac{1}{2}, 0)$. The CFL number is fixed at 0.2, which restricts the time step to $\Delta t = 0.2 \cdot \frac{R_i L_\theta}{n_\theta}$. Similar to the first advection test, we set $L_z = 1$ and L_θ such that the arc length along the outer cylinder is 3. The distance between the cylinder walls is $L_r = 1$.

A snapshot of the solution for case C_1 at $t = 0.25$ is shown in figure 15. The blue line indicates the level set $f = 0.5$ in the plane $z = 0$. This is an approximation to the cross-section of the bubble in this plane. The bubble is sheared by the motion of the two cylinder walls, and reversed to its initial position after $t = T = 1$.

The error in the volume fraction field $E_2(T)$ is shown in figure 16. As before, the error converges

Case	r_i	η	$(n_\theta \times n_r \times n_z)$	$[\theta_{min}, \theta_{max}]$	dt $(\cdot 10^{-3})$
C_1	1	$\frac{1}{2}$	$(96 \times 32 \times 32)$	$[-\frac{3}{4}, \frac{3}{4}]$	3.125
C_2	3	$\frac{1}{2}$	$(96 \times 32 \times 32)$	$[-\frac{3}{8}, \frac{3}{8}]$	3.125
C_3	7	$\frac{1}{2}$	$(96 \times 32 \times 32)$	$[-\frac{3}{16}, \frac{3}{16}]$	3.125
C_4	15	$\frac{1}{2}$	$(96 \times 32 \times 32)$	$[-\frac{3}{32}, \frac{3}{32}]$	3.125
C_∞	-	1	$(96 \times 32 \times 32)$	$[x_{min}, x_{max}] = [-\frac{3}{2}, \frac{3}{2}]$	3.125
C_2	1	$\frac{1}{2}$	$(192 \times 64 \times 64)$	$[-\frac{3}{4}, \frac{3}{4}]$	1.563
C_2	1	$\frac{1}{2}$	$(384 \times 128 \times 128)$	$[-\frac{3}{4}, \frac{3}{4}]$	0.781

Table 6: Description of test cases for laminar Couette flow.

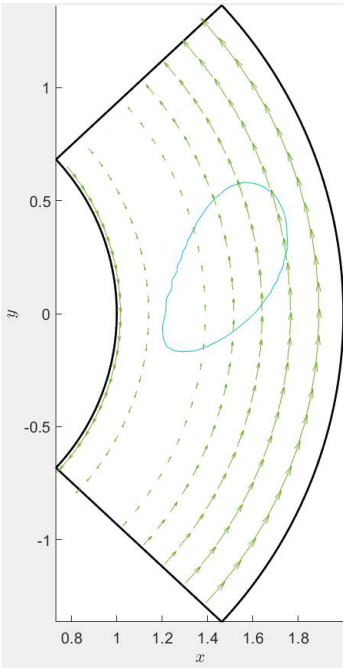


Figure 15: Snapshot of case C_1 at $t = 0.25$.

approximately to the result on the original solver on a Cartesian domain as η approaches 1. However, the error is now higher for the domains with higher curvature. The reason for this is that the bubble experiences the largest deformations on these grids, as illustrated by the bubble in figure 15 in the bottom left corner, whereas deformations in previous test cases were very relatively small. The error is also higher in absolute terms for this test case compared to test cases B. This is partly because of the higher CFL number in these cases, and partly due to the large deformations.

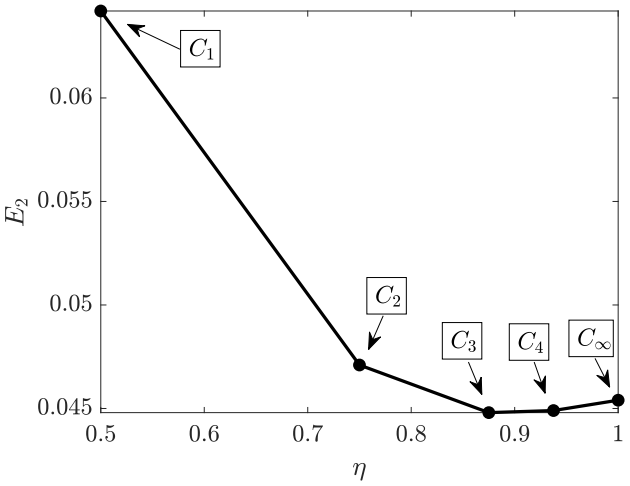


Figure 16: Error $E_2(T)$ for test cases C_1, C_2, C_3, C_4 and the Cartesian solver C_∞.

4.2 Single rising bubble

The previous test cases showed the performance of the interface advection algorithm on an imposed velocity. In this section, we study the performance of the fully coupled two-phase problem by introducing surface tension. We follow the full algorithm as described in chapter 3.

We study the behaviour of a 3D bubble in an initially stationary fluid with buoyancy. This benchmark is described in detail in (Hysing et al., 2009), and there is reference data available from several different methods (Klostermann et al., 2013; Zainali et al., 2013; Adelsberger et al., 2014; Zhang et al., 2015; Cifani et al., 2016). The aim is to reproduce the first 3D rising bubble test case described in the benchmark.

The rising bubble dynamics are characterized by the Eötvös number and the Reynolds number:

$$E\ddot{o} = \frac{g\Delta\rho d_e^2}{\sigma}, \qquad Re = \frac{\rho_l U_\infty d_e}{\mu_l} \tag{75}$$

where U_∞ is the terminal velocity of the bubble, ρ_l and μ_l the mass density and the dynamic viscosity of the liquid phase. The effective diameter d_e is defined as the diameter of a spherical bubble with the same volume as the actual bubble.

Following (Hysing et al., 2009), these parameters are set to $E\ddot{o}= 10$, $Re = 35$, with a density and viscosity ratio of $\rho_b/\rho_l = \mu_b/\mu_l = 1/10$. A spherical bubble with diameter 0.5 is initialized in a domain of dimensions $L_x \times L_y \times L_z = 1 \times 1 \times 2$, with gravity pointing in the negative z-direction. Figure 17 shows the initial condition for the 3D rising bubble case.

In this benchmark, the domain is bounded by non-slip walls in all directions. These boundary conditions cannot be reproduced in our current numerical setup in cylindrical coordinates, as the Poisson equation solver requires two periodic directions (θ and z) and one direction (r) with von Neumann conditions on the pressure. We will therefore first investigate the effect of periodic boundary conditions on the flow.

First, we perform the rising bubble benchmark case using the original TBFsolver in a rectangular domain with solid walls on all sides. The domain uniformly discretized using a $64 \times 64 \times 128$ grid. The relevant parameters are shown in table 7. The same parameter values are used in (Zhang et al., 2015).

In order to compare the results with reference data (Adelsberger et al., 2014), we define the rising velocity of the bubble as follows:

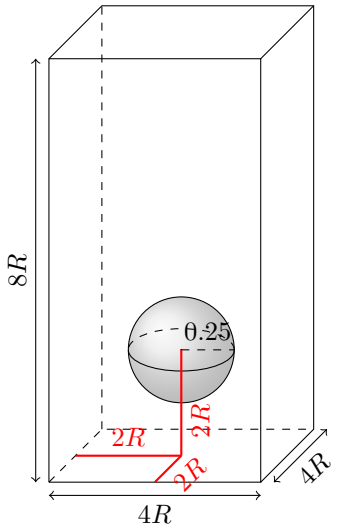


Figure 17: Initial configuration for the 3D rising bubble case in a rectangular domain.

ρ_l	ρ_g	μ_l	μ_g	g	σ
10^3	10^2	10	1	0.98	24.5

Table 7: Simulation parameters for the 3D rising bubble case.

$$u_{bubble} = \frac{\sum_i f_i (u_z)_i V_i}{\sum_i f_i V_i}, \tag{76}$$

where the summation is over all grid cells i, with f_i the volume fraction in cell i, $(u_z)_i$ is the vertical velocity at the cell center and V_i is the volume of cell i. Additionally, we denote the maximum extension of the bubble in each coordinate direction as the diameter $\mathbf{d}_b(t) = [d_x(t), d_y(t), d_z(t)]$, where $d_n(t)$ is given by the maximum distance between interface positions along direction n for the whole bubble interface.

The final shape of the bubble is shown in figure 18. The bubbles start as spheres with a diameter of 0.5, and is gradually compressed in the vertical direction, which causes the bubble to expand symmetrically in the horizontal directions. The final shape is near an ellipsoid, which is consistent with the position of the point (Re,$E\ddot{o}$)= $(35, 10)$ in the Grace diagram (Grace, 1973).

Figure 19 shows the results for this case compared with results from (Adelsberger et al., 2014). On the left graph, the initial vertical acceleration of the bubble is shown, which gradually settles to its terminal velocity. The right graph shows the different components of $\mathbf{d}_b(t)$. The upper branch of the graph represents the horizontal directions, while the lower branch represents the z-direction. The results for the bubble diameter agree within 1% with the reference data, which corresponds to a distance of less than one grid cell. Better accuracy might therefor be achieved using grid refinements.

Next, we investigate the effect of periodic boundary conditions instead of non-slip walls on the bubble dynamics. Similar to the cylindrical setup, we set the boundary conditions in the x- and z-directions to be periodic, while non-slip walls are maintained in the y-direction. All other parameters are fixed at the same values.

The results for this comparison are shown in figure 20. The simulation with the modified boundary conditions is shown in blue, while the previous results are shown in red. With two periodic boundary conditions, the rising velocity of the bubble is greater as shown in the left graph, since the vertical velocity is no longer restricted by the non-slip walls. This effect was also shown in (Zhang et al.,

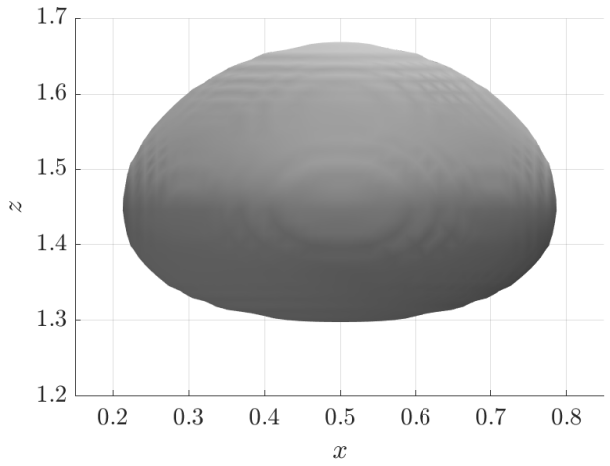


Figure 18: Isosurface $f = 0.5$ of 3D rising bubble at $t = 3$.

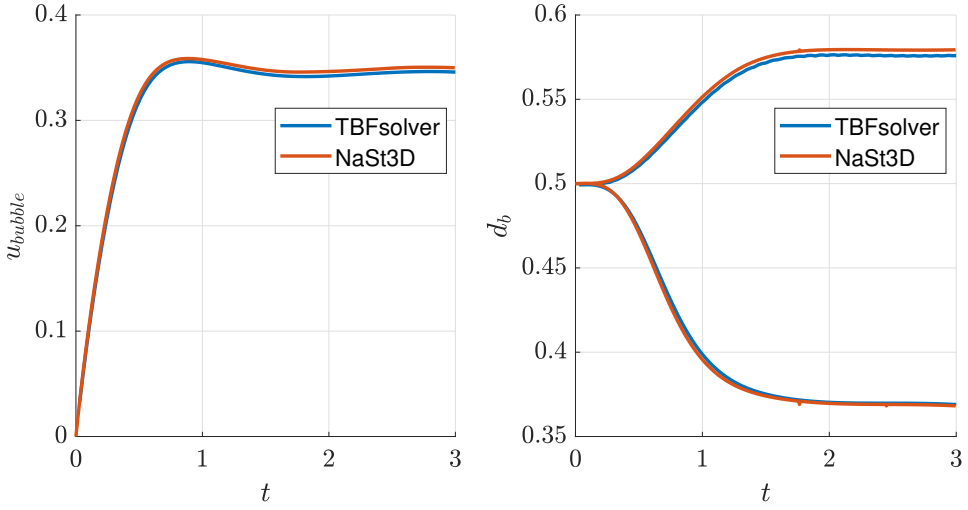


Figure 19: Single rising bubble in rectangular domain with solid walls. Left: rise velocity in z-direction. Right: bubble diameter in the x- and y-direction (upper branch) and z-direction (lower branch). Results from the TBFsolver are compared to data from (Adelsberger et al., 2014) (NaSt3D).

2015), where the domain size was increased in order to remove the influence of the boundary conditions.

Zhang et al. also showed that removing boundary effects in the horizontal directions lead to a larger bubble deformation, as the bubble can more freely expand in the horizontal directions. This effect is also seen in the right graph of figure 20. Compared to the non-slip wall boundary condition, the bubble expands significantly more in the horizontal x-direction with periodic boundaries. In the y-direction, the boundary conditions are the same, and the bubble size in this direction remains the same.

Next, the results for the rising bubble in a rectangular domain will be used as a reference for a rising bubble in cylindrical coordinates. Similar to previous tests, the rising bubble case will be simulated in cylindrical domains with different values of the curvature $\eta = R_i/R_o = R_i/(R_i + 1)$. The domain spans an angle of $L_\theta = 1/R_o$, such that the arc length along the outer cylinder is 1.

Figure 21 shows the results for the cylindrical cases, as well as the results for the Cartesian domain,

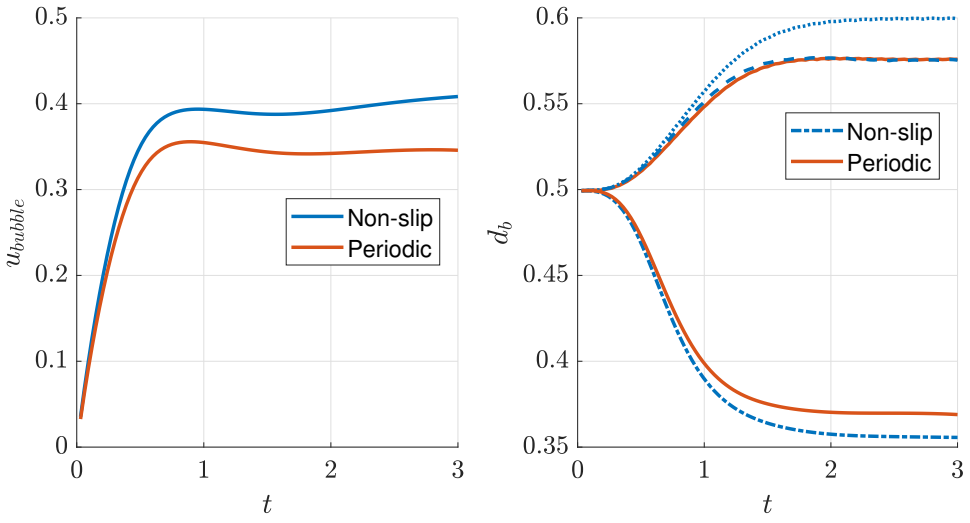


Figure 20: Single rising bubble in rectangular domain with solid walls (red) and two periodic directions (blue). Left: rise velocity in z-direction. Right: bubble diameter in the x-direction (dotted line), y-direction (dashed line) and z-direction (dash-dotted line).

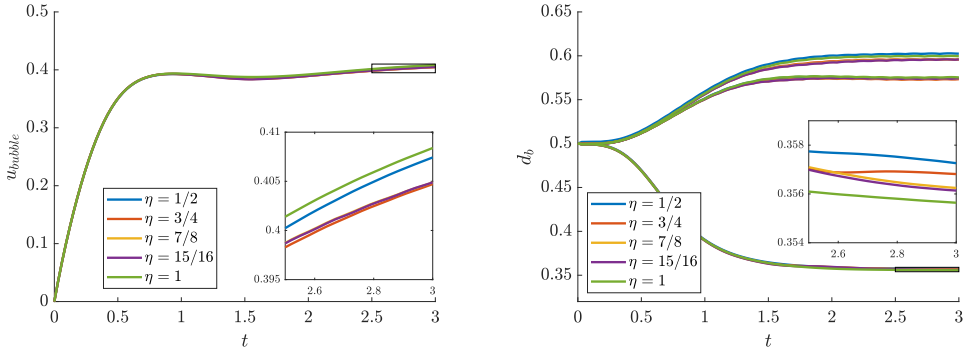


Figure 21: Single rising bubble in cylindrical domains at different values of η. The results for a single rising bubble in a rectangular domain is denoted by $\eta = 1$. Left: rise velocity in z-direction. Right: bubble diameter in the vertical direction (bottom branch), wall-normal direction (middle branch) and horizontal periodic direction (top branch).

denoted by $\eta = 1$. The left plot shows a consistent rising velocity of the bubble for the different cases. The inset shows the final rising velocity. There is no clear convergence visible for $\eta \to 1$, however, the results are within 1% from the Cartesian case.

The graph on the right of figure 21 shows the diameter of the bubble. The diameter is split into three branches, corresponding to the x-, y- and $z-$directions from top to bottom. There are slight variations in bubble diameter for the different cases. This can be expected since the shape of the domain is dependent on η. The inset of figure 21 shows convergence of the bubble diameter for $\eta \to \infty$.

In conclusion, these results show that we can accurately model the dynamics of a rising bubble in a cylindrical. By reducing the curvature of the cylindrical domain, we find with results in a rectangular domain for the rise velocity and bubble diameter within 1%.

5 Drag reduction in Taylor-Couette flow

The numerical methods derived in the previous sections can be used to investigate the behaviour of bubbles in Taylor-Couette flow and determine the influence on the overall drag. We start in subsection 5.1 with a definition of the drag reduction. In subsection 5.2, we detail the parameters that are used in the simulations. Finally, the results are discussed in subsection 5.3.

5.1 Definition of drag reduction

Drag reduction in TC flow can be defined in terms of the torque on the cylinders (Sugiyama et al., 2008):

$$DR = 100 \times \frac{T_0 - T_\alpha}{T_0}, \tag{77}$$

where T_0 and T_α are the torques on the cylinders obtained, respectively, at 0 and α volume fraction of the dispersed bubbly gas phase. Drag reduction hence measures the torque alteration due to the presence of a gas volume fraction α compared to single-phase flow as a percentage.

The torque on the cylinder wall is computed from the shear stress of the fluid at the wall as shown in section 2.1.2. Here we take $T_\alpha = \langle (\mathcal{T}_{i,\alpha} + \mathcal{T}_{o,\alpha})/2 \rangle_t$ as the average of the torques on the inner and outer walls at volume fraction α, averaged over time. We recall that

$$\mathcal{T}_{i,\alpha} = 2\pi r_i^2 L_z \langle \tau_{r\theta} \rangle_{A_i,\alpha}, \tag{78}$$

where $\langle \ldots \rangle_{A_i,\alpha}$ indicates an average over the surface of the inner cylinder. A similar relation holds for the torque on the outer wall.

5.2 Setup of the numerical experiments

In this section, we describe the setup for the numerical simulation of two-phase TC flow. Two geometrical parameters describe the flow domain, i.e., the cylinder radius ratio $\eta = R_i/R_o$ and the aspect ratio $\Gamma = L_z/d$, where R_i and R_0 are the radii of the inner and outer cylinder respectively, L_z is the vertical height of the cylinders and $d \equiv R_o - R_i$ is the gap width. In these simulations we set $\eta = 10/11$ and $L_z = 2$, as was also adopted in (Spandan et al., 2018).

The flow is driven by rotation of the inner cylinder, while the outer cylinder remains stationary. The flow is therefore characterized by the inner cylinder Reynolds number $\mathrm{Re}_i = Ud/\nu$, where U is the fixed velocity of the inner cylinder, and ν is the kinematic viscosity of the carrier fluid.

Similar to (Spandan et al., 2018), we will study the flow dynamics at $\mathrm{Re}_i = 5 \times 10^3$. The focus of this study is to understand the effect of the addition of bubbles on the global energy dissipation in the flow. The deformability of bubbles also influences DR, next to α. It is controlled by the Weber number, which relates inertial forces to surface tension forces. It is defined as $\mathrm{We} = \rho_f U^2 d_p/\sigma$, where ρ_f is the density of the carrier fluid, d_p is the bubble diameter and σ is the surface tension between the gas and the carrier fluid. In this study, we consider bubbles at $We = 8$.

We simulate the flow at a grid resolution of $N_\theta \times N_r \times N_z = 768 \times 192 \times 384$. The grid is uniformly spaced in the azimuthal and axial direction and follows a hyperbolic tangent-type profile in the radial direction. We initialize 120 bubbles with a diameter of $d_b = 0.1d$, which corresponds to a global gas volume fraction of $\alpha \approx 0.8\%$. Initially, the bubbles are resolved with approximately 15-20 grid points in each direction, which indicates that the bubble shape is well resolved with the GHF method according to experience gathered in (Popinet, 2009; Cifani et al., 2018).

The influence of gravity is measured by the Froude number. For two-phase flows, this is defined as $\mathrm{Fr} = \sqrt{\tilde{\rho} U^2/((\tilde{\rho} - 1) g R_i)}$, where $\tilde{\rho} = \rho_f/\rho_b$ is the density ratio of the two bubbles. The Froude number is set to $\mathrm{Fr} = 0.64$. The density ratio of the carrier fluid to the bubbles is set to 20. This is smaller than in most physical experiments, but provides numerical stability that is sometimes lost

at realistic density ratios of 1000. The viscosity ratio is similarly set to $\tilde{\mu} = \mu_f/\mu_b = 20$. With this setup, the bubble size in comparison to the Kolmogorov scale η_K is approximately $d_p/\eta_K \sim 14$.

The simulation is carried out in two steps. First, we initialize laminar Couette flow in the domain and allow the single-phase system to develop fully. As described before, we consider the simulation to be developed into a statistically stationary state after sufficiently many flow-through times. A flow-through time is defined as the typical time it takes a fluid particle to traverse the domain in the streamwise direction, which in this simulation is determined by the rotational velocity of the inner wall and the azimuthal extent of the domain. Here, we use 500 flow-through times for this phase. Subsequently, we determine time-averaged properties and take a time-averaging window long enough so that the relative variation of Nu_ω is less than 1% along the radial direction. From this simulation, we determine the torque T_0 on the cylinders in single-phase TC flow. Next, following (Spandan et al., 2018) we initialize a regular grid of 120 bubbles into the flow and let the flow develop further. After the flow has reached a statistically stationary state, we again measure the torque T_α on the cylinder from a sufficiently long time-average. Other statistics that will be gathered are the radial distribution of bubbles, energy dissipation and average velocity profiles.

5.3 Results

To set up a bubble-laden flow simulation, we first create a reference by simulating single-phase TC flow at $\mathrm{Re}_i = 5 \times 10^3$. Starting from laminar Couette flow, the flow develops into a statistically stationary state. The flow is simulated for approximately 500 flow-through times, based on the velocity of the inner cylinder and the size of the domain. An instantaneous snapshot of the magnitude of the velocity field is shown in figure 22. The flow is dominated by a large scale plume that is ejected from the inner cylinder and extends radially to the outer cylinder. This flow organization confirms earlier observations in Spandan et al. (2018).

In single-phase TC, the torque is directly related to the Nusselt number Nu_ω, which is a measure for the angular velocity transport. The relation between the torque and the Nusselt number was discussed in section 2.1.3, and is repeated below for convenience:

$$Nu_\omega = \frac{J^\omega}{J_0^\omega}, \qquad \text{where} \quad J^\omega = \frac{T}{2\pi L_z \rho}, \quad J_0^\omega = -\nu \frac{2}{\eta(1+\eta)} \frac{r_i^2}{d} \tag{79}$$

Figure 23 shows the Nusselt number as a function of the radial position between the cylinders. As expected, angular velocity transport is constant throughout the domain. From figure 23 we find $Nu_\omega = 8.38$, which is close to the results quoted in (Spandan et al., 2018).

Next, we initialize a grid of 120 bubbles in the domain, motivated by Spandan et al. (2018). Initially, the bubbles are roughly equidistant from each other. The bubbles are placed in the developed velocity field of single-phase TC flow. Based on the bubble positions, the indicator function f is constructed, from which the initial density and viscosity field can be determined. The velocity of the fluid inside the bubbles is set equal to that in single-phase flow.

As discussed before, the presence of capillary waves in two-phase flows adds another restriction to the time step. At high Reynolds numbers, this severely restricts the time step compared to that of single-phase flow. Since the time step restriction is inversely proportional to the surface tension, we decided to initially simulate highly deformable bubbles for which the surface tension is less restrictive for the time step. In this simulation, we therefore set $\mathrm{We} = \rho_f U^2 d_p/\sigma = 8$, which leads to a maximum time step based on capillary waves of:

$$(\Delta t)_{capillary} = \sqrt{\frac{d^3\rho}{4\pi\sigma}} \approx 2 \times 10^{-4}, \tag{80}$$

which is approximately ten times lower than the required time step for single-phase flow. Given the rather tight restriction (80), the total simulated time that we can report here corresponds to 50 flow-through times. In terms of computational effort, the VOF algorithm accounts for at most

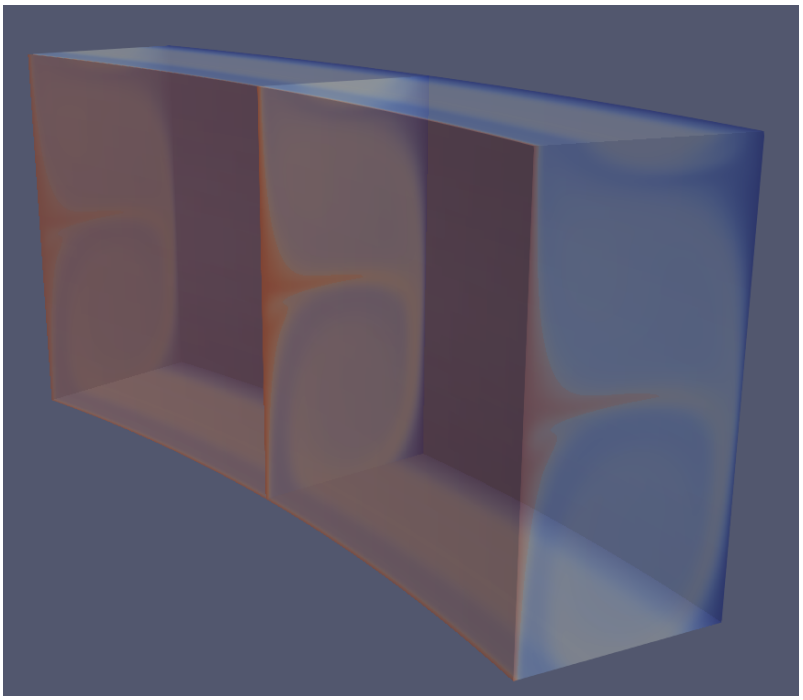


Figure 22: Instantaneous snapshot of the magnitude of the velocity field in single-phase TC flow. Red colour indicates a high velocity, blue color indicates a low velocity.

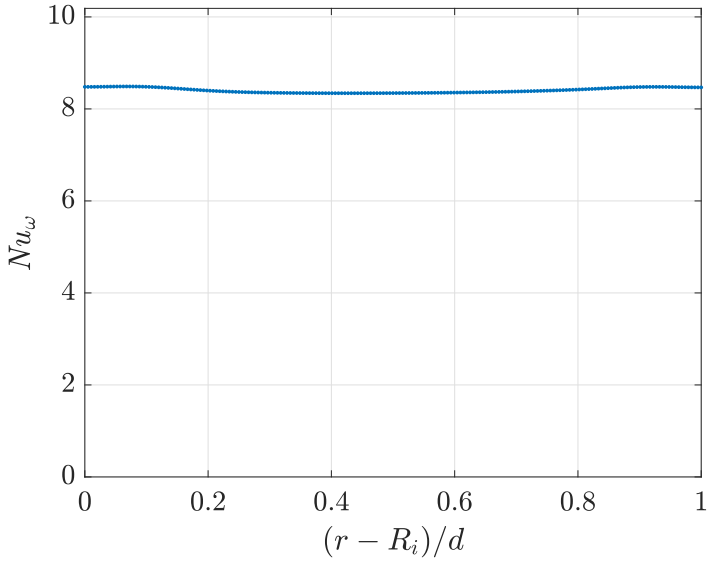


Figure 23: Non-dimensionalized angular momentum transport profile for the singe phase TC flow.

30% of the total computational time, indicating that the solution of the Poisson equation is still the most resource-intensive task.

An instantaneous snapshot of the developed flow with bubbles is shown in figure 24. The colour indicates the magnitude of the velocity field. Compared to the single-phase flow, we see a clear weakening of the plume ejected from the inner cylinder. Additionally, the plume no longer protrudes through the bulk to the outer cylinder.

The break-up of large scale plumes in two-phase Taylor-Couette flow is often attributed to the ac-

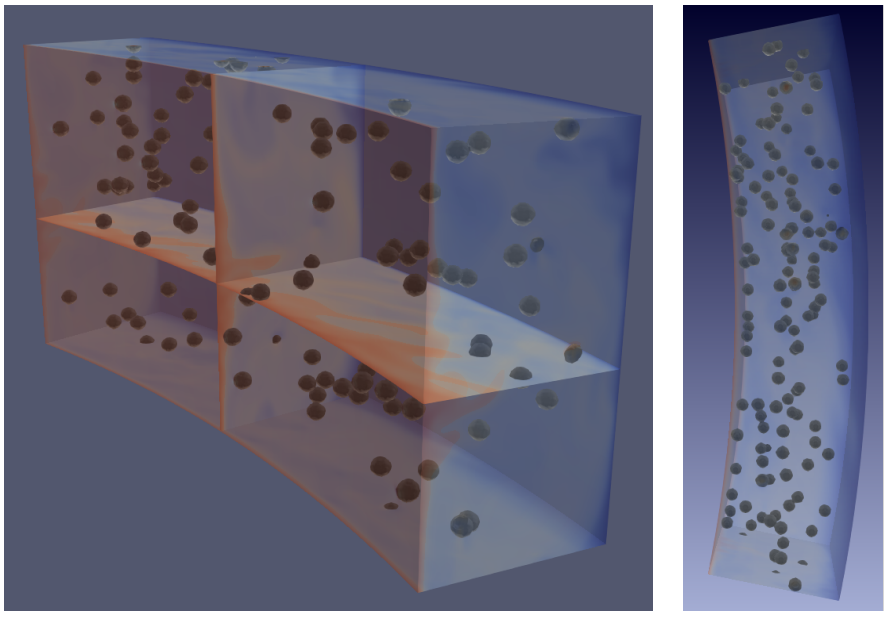


Figure 24: Instantaneous snapshot of the magnitude of the velocity field in two-phase TC flow. Red colour indicates a high velocity, blue color indicates a low velocity. The color range is the same as in figure 22. The isosurfaces $f = 0.5$ are in grey, which show the bubbles. Clustering of bubbles near the inner wall is clearly appreciated from the top-view shown on the right.

cumulation of bubbles near the inner cylinder (Spandan et al., 2017b). In figure 25, the probability distribution function of the radial position of the bubbles is shown. The PDF is calculated using a time averaging of two flow-through times. Initially, the bubbles quickly accumulate near the inner cylinder. After some time has passed, more bubbles migrate away from the inner cylinder. This behaviour was also noted in (Sugiyama et al., 2008) during the transition to a statistically stationary flow. At first, bubbles are drawn toward the vortical structures that are present in single-phase TC flow. This is clearly visible in experiments at slightly lower Reynolds numbers, where bubble strongly accumulate near the inner cylinder (Murai et al., 2005).

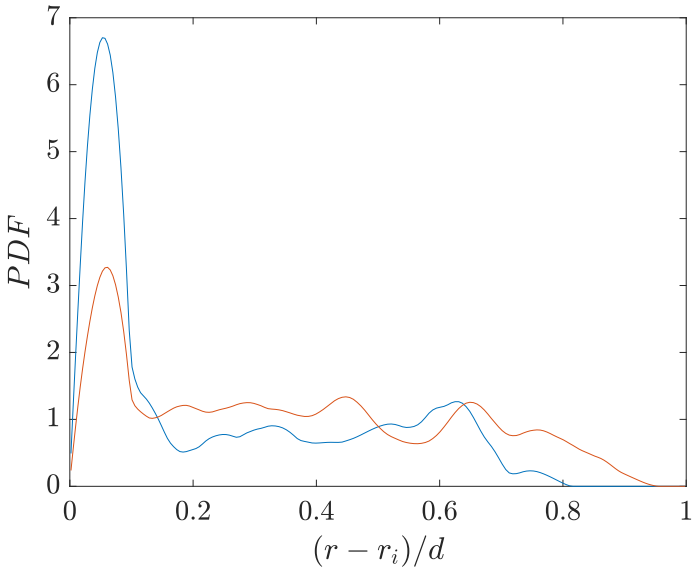


Figure 25: Probability distribution function of the radial coordinates of the gas phase. PDFs are based on averages over 2 flow-through times, taken after 18 (blue) and 48 (red) flow-through times.

Once the bubbles have accumulated near the inner cylinder, their axial (vertical) movement disrupts the vortex structure in the flow. This allows the bubbles to also move away from the inner cylinder, which can be seen in figure 25. Shortly after injecting the bubbles in the flow, most bubbles accumulate on the inner cylinder. After 30 more flow-through times, most bubbles tend to remain close to the inner cylinder around $(r - r_i)/d = 0.1$, corresponding to the initial diameter of the bubbles. There is less accumulation at the inner wall after the initial transient has passed. We observe a 'plateau' in the PDF between $(r - r_i)/d = 0.2$ and 0.8 and a gradual decrease toward 0 as we approach the outer wall.

We can now compare the velocity fields of single-phase flow and bubble-laden flow to find the change in torque. We can directly evaluate the torque on the inner and outer cylinder, by calculating the shear stress on these boundaries. The torque can be non-dimensionalized using the Nusselt number as before (equation 79). Figure 26 shows the calculated torque on both cylinders. Conservation of angular momentum states that, averaged over time, both torques must be equal in a statistically stationary state. In the figure, we can observe an oscillating behaviour with a period of approximately 2.5 flow-through times. This is on the same time-scale as the period of the vortical structure in single-phase TC flow, which indicates that the flow is still developing and has not yet reached a stationary state. We do observe a decreasing magnitude of the difference between T_i and T_o. The time-averaging over the period available to us indicates a value Nu_ω which is 1.5% higher than in the single-phase flow.

Longer simulations are required to accurately determine the possible drag reduction for this flow. Currently, the torque shows oscillations in time of more than 10%, which is much higher than the expected drag reduction. Moreover, although deformability is beneficial for drag reduction, it is not clear whether this also holds for high values of We as adopted here. In a fully developed flow, we can accurately determine the convergence of the torque and compare this to the single-phase flow result to determine drag reduction.

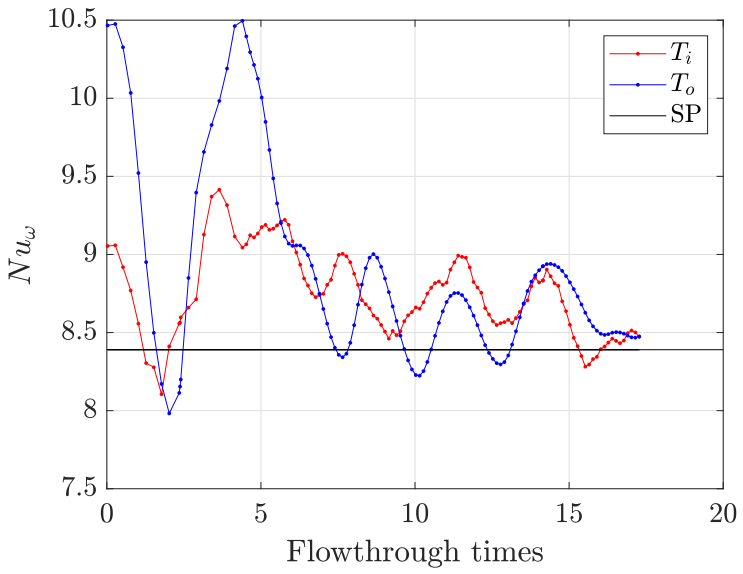


Figure 26: Non-dimensionalized torque on the inner (red) and outer (outer) cylinder evaluated on instantaneous velocity fields. For single-phase flow $Nu_\omega \approx 8.4$. As a reference, the Nusselt number for single-phase flow (SP) is shown.

6 Conclusions and outlook

6.1 Main findings and achievements

The motivation for this study was to understand the physical mechanisms behind drag reduction in two-phase TC flow through direct numerical simulation. In this research, an important step toward this long-term goal was made by creating a new code for the simulation of bubble-laden TC flow. The new simulation platform is a further development of the code TBFsolver developed by Cifani (Cifani et al., 2018). This code for channel flow has been adapted thoroughly to the cylindrical geometry of TC flow and constitutes a highly parallel algorithm for such simulations. Extensive validation against analytical findings or by comparison with reported results in literature has been presented to enhance confidence in the code. Moreover, a first full-scale simulation of turbulent multi-phase TC flow has been illustrated, underlining the potential of the new approach.

In section 2, we focused on adapting the numerical methods for single-phase flow to be able to treat flow in cylindrical coordinates. The Navier-Stokes equations for single-phase flow are solved using the fully conservative method developed by (Morinishi et al., 2004). Additionally, we solve the Poisson equation for pressure using the spectral method based on the research by (van der Poel et al., 2015) and further developed for variable-density flows (Cifani et al., 2018; Cifani, 2019). The method has been adapted to cylindrical coordinates by modifying the Fourier transformed Poisson equations.

The code has been tested for in various scenarios. First, we have shown that laminar Couette can be simulated precisely, with the resulting velocity field equal to the exact solution up to machine precision. Secondly, we measured the increase in torque near the transition from laminar Couette flow to Taylor-vortex flow. The results are found to agree with results from literature within 0.05%. Next, we simulated TC flow at a higher Reynolds number and determined the angular velocity flux. Its value is shown to be independent of the radial position between the cylinders, which shows convergence to a statistically stationary flow. From the angular velocity transport, we found the Nusselt number for TC flow. The results show an agreement with literature within 5% Ostilla et al. (2013). However, further grid refinement shows no clear convergence to the literature data hinting at the possibility of a small systematic error between the codes. In fact, the precise model reported in Ostilla et al. (2013) is not identical to our problem, which may account for the slight differences.

Section 3 contains an account of the extension of the VOF method from the TBFsolver to cylindrical coordinates. We have retained the main interface reconstruction methods. The interface is reconstructed as linear surfaces on the computational domain. The geometrical flux following from the advection of the interface is subsequently transformed to cylindrical coordinates.

The surface tension force is calculated from the curvature of the interface. We have derived the equations which relate the curvature to the interface position. The interface position is determined using the generalized height function method. Most of the structure of the TBFsolver has been kept intact, so the computational performance on a cylindrical domain is similar to that of the original code showing linear parallel speed-up for problems running on tens of thousands of CPUs.

The performance of the new code has been tested in section 4. Firstly, the advection algorithm for a single bubble has been tested by studying the bubble shape on an imposed velocity field. By comparing the final shape of the bubble to the analytical solution, it has been shown that the advection algorithm converges linearly with time. By varying the curvature of the domain, we found that the advection algorithm has a similar performance compared to test cases on a rectangular domain using the original TBFsolver.

The algorithm for calculating surface tension is tested using a single rising bubble in an initially quiescent flow. We compared the bubble shape and its eventual rise velocity to results from the original TBFsolver. We found that the bubble rise velocity and diameter are within 1% of the values obtained on a rectangular grid.

After testing and validation of the new code, a simulation of 120 bubbles in TC flow at $\mathrm{Re}_i = 5\times10^3$ is performed, which is described in section 5. Initially, we simulated the flow without the presence

of any bubbles, from which the torque on the cylinders is calculated. We observe close agreement to literature data. Subsequently, 120 bubbles were added to the flow and the torque was studied during the transition to statistically stationary two-phase flow.

A limiting factor during the simulations was the strict limit on the time step due to capillary waves. By introducing bubbles in the flow, the time step had to be severely reduced. This reduced time-step size has a marked influence on the simulated flow-time that could be amassed. Results up to this point show that large scale structures in the flow are broken up by the rising bubbles. Similar to results from experiments, we see that bubbles concentrate near the inner cylinder, which causes a break-up of the large plumes in TC flow that are responsible for angular momentum transport. After a transitional phase, bubbles migrate on average to a peaked distribution near the wall, which shows simultaneously a clear 'shoulder' toward the bulk of the flow, and the bubble distribution becomes qualitatively similar to experimental results.

As of yet, the drag reduction in TC flow due to the addition of bubbles could not be accurately determined, as more time is needed for the flow to develop into a statistically stationary state. Current results at high Weber numbers, implying high deformability, are suggestive that the drag reduction will not be significantly higher than results from literature at lower Weber numbers. Longer simulations might give more insight into the effect of bubble deformability on drag reduction when simulations are repeated also at different We. The basis for such research has been laid in the newly developed TBFsolver, which is the main result achieved in this study.

6.2 Future developments

In this study, an extension to the TBFsolver has been developed for studying two-phase Taylor-Couette flow problems. The TBFsolver can be used to simulate two-phase systems at relatively large volume fractions and with relatively high bubble deformability.

Now that all parts of the numerical method have been carefully tested and validated, the code is ready to be used for various flow studies on bubble-laden TC flow. The excellent performance of the GHF method for highly deformed bubbles enables us to investigate the Weber number dependence of drag reduction. By performing multiple simulations such as described in section 5 at different Weber numbers, we can investigate the importance of bubble deformability on drag reduction. In order to achieve lower simulation times, an extension of the time-stepping method to implicit time-integration for the surface tension terms may be fruitful.

Using the new code, we can look more closely at the mechanisms behind drag reduction by analyzing the kinetic energy dissipation rate in the flow. This may lead to more insight into the effect of bubbles on the intensity of the dissipating structures in two-phase TC flow.

Currently, the multi-marker formulation prevents the coalescence and break-up of bubbles. However, in a real system, interactions between bubbles or between a bubble and a wall may lead to coalescence or breakup. This leads to a distribution of sizes of bubbles, which may affect the flow structures. In order to account for this, a method might be implemented to determine whether coalescence or breakup occurs, after which marker functions are merged, or a new marker function is added. This will be particularly important at higher bubble volume fractions, when interactions occur more frequently.

Bibliography

Adelsberger, J., Esser, P., Griebel, M., Gross, S., Klitz, M., and Rüttgers, A. 2014. 3d incompressible two-phase flow benchmark computations for rising droplets. In Proceedings of the 11th world congress on computational mechanics (WCCM XI), Barcelona, Spain, volume 179.

Andereck, C. D., Liu, S. S., and Swinney, H. L. 1986. Flow regimes in a circular Couette system with independently rotating cylinders. *Journal of Fluid Mechanics* 164:155–183.

Bilson, M. and Bremhorst, K. 2007. Direct numerical simulation of turbulent Taylor–Couette flow. *Journal of Fluid Mechanics* 579:227–270.

Brackbill, J. U., Kothe, D. B., and Zemach, C. 1992. A continuum method for modeling surface tension. *Journal of Computational Physics* 100:335–354.

Brauckmann, H. J. and Eckhardt, B. 2013. Direct numerical simulations of local and global torque in Taylor-Couette flow up to Re = 30 000. *Journal of Fluid Mechanics* 718:398–427.

Ceccio, S. L. 2010. Friction Drag Reduction of External Flows with Bubble and Gas Injection. *Annual Review of Fluid Mechanics* 42:183–203.

Chouippe, A., Climent, E., Legendre, D., and Gabillet, C. 2014. Numerical simulation of bubble dispersion in turbulent Taylor-Couette flow. *Physics of Fluids* 26:43304.

Cifani, P. 2019. Analysis of a constant-coefficient pressure equation method for fast computations of two-phase flows at high density ratios. *Journal of Computational Physics* 398:108904.

Cifani, P., Kuerten, J. G., and Geurts, B. J. 2018. Highly scalable DNS solver for turbulent bubble-laden channel flow. *Computers and Fluids* 172:67–83.

Cifani, P., Michalek, W. R., Priems, G. J. M., Kuerten, J. G. M., van der Geld, C. W. M., and Geurts, B. J. 2016. A comparison between the surface compression method and an interface reconstruction method for the VOF approach. *Computers & Fluids* 136:421–435.

Coles, D. and Atta, C. V. 1966. Measured distortion of a laminar circular Couette flow by end effects. *Journal of Fluid Mechanics* 25:513–521.

Coyajee, E. and Boersma, B. J. 2009. Numerical simulation of drop impact on a liquid-liquid interface with a multiple marker front-capturing method. *Journal of Computational Physics* 228:4444–4467.

Desjardins, O., Blanquart, G., Balarac, G., and Pitsch, H. 2008. High order conservative finite difference scheme for variable density low Mach number turbulent flows. *Journal of Computational Physics* 227:7125–7159.

Dodd, M. S. and Ferrante, A. 2014. A fast pressure-correction method for incompressible two-fluid flows. *Journal of Computational Physics* 273:416–434.

Dong, S. 2008. Herringbone streaks in Taylor-Couette turbulence. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 77.

Dong, S. and Shen, J. 2012. A time-stepping scheme involving constant coefficient matrices for phase-field simulations of two-phase incompressible flows with large density ratios. *Journal of Computational Physics* 231:5788–5804.

Drazin, P. G. and Reid, W. H. 2004. Hydrodynamic Stability. Cambridge University Press, Cambridge.

Echeverry, D. B. 2014. Subcritical transition in turbulence in Taylor-Couette flow. Phd, Georgia Institute of Technology.

Eckhardt, B., Grossmann, S., and Peterson, W. H. 2007. Torque scaling in turbulent Taylor - Couette flow between independently rotating cylinders. *Journal of Fluid Mechanics* 581:221–250.

Evrard, F., Denner, F., and van Wachem, B. 2020. Height-function curvature estimation with arbitrary order on non-uniform Cartesian grids. *Journal of Computational Physics: X* 7:100060.

Fasel, H. and Booz, O. 1984. Numerical investigation of supercritical Taylor-vortex flow for a wide gap. *Journal of Fluid Mechanics* 138:21–52.

Grace, J. 1973. Shapes and velocities of bubbles rising in infinite liquids. *Transactions of the Institution of Chemical Engineers* 51:116–120.

Grossmann, S., Lohse, D., and Sun, C. 2016. High–Reynolds Number Taylor-Couette Turbulence. *Annual Review of Fluid Mechanics* 48:53–80.

Harlow, F. H. and Welch, J. E. 1965. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. *Physics of Fluids* 8:2182–2189.

Hysing, S., Turek, S., Kuzmin, D., Parolini, N., Burman, E., Ganesan, S., and Tobiska, L. 2009. Quantitative benchmark computations of two-dimensional bubble dynamics. *International Journal for Numerical Methods in Fluids* 60:1259–1288.

Klostermann, J., Schaake, K., and Schwarze, R. 2013. Numerical simulation of a single rising bubble by VOF with surface compression. *International Journal for Numerical Methods in Fluids* 71:960–982.

Kodama, Y., Kakugawa, A., Takahashi, T., and Kawashima, H. 2000. Experimental study on microbubbles and their applicability to ships for skin friction reduction. *International Journal of Heat and Fluid Flow* 21:582 – 588. Turbulence and Shear Flow Phenomena 1.

Lathrop, D. P., Fineberg, J., and Swinney, H. L. 1992. Turbulent flow between concentric rotating cylinders at large Reynolds number. *Physical Review Letters* 68:1515–1518.

Latorre, R. 1997. Ship hull drag reduction using bottom air injection. *Ocean Engineering* 24:161–175.

Lee, M. J., Oh, B. D., and Kim, Y. B. 2001. Canonical Fractional-Step Methods and Consistent Boundary Conditions for the Incompressible Navier-Stokes Equations. *Journal of Computational Physics* 168:73–100.

Lo, T. S., L'Vov, V. S., and Procaccia, I. 2006. Drag reduction by compressible bubbles. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics* 73.

Lohse, D. 2018. Bubble puzzles: From fundamentals to applications. *Physical Review Fluids* 3:110504.

Madavan, N. K., Deutsch, S., and Merkle, C. L. 1984. Reduction of turbulent skin friction by microbubbles. *Physics of Fluids* 27:356–363.

Meier, H. F., Alves, J. J. N., and Mori, M. 1999. Comparison between staggered and collocated grids in the finite-volume method performance for single and multi-phase flows. *Computers and Chemical Engineering* 23:247–262.

Mirjalili, S., Jain, S. S., and Dodd, M. S. 2017. Interface-capturing methods for two-phase flows : An overview and recent developments. *Center for Turbulence Research: Annual Research Briefs* pp. 117–135.

Morinishi, Y., Vasilyev, O. V., and Ogi, T. 2004. Fully conservative finite difference scheme in cylindrical coordinates for incompressible flow simulations. *Journal of Computational Physics* 197:686–710.

Murai, Y., Oiwa, H., and Takeda, Y. 2005. Bubble behavior in a vertical Taylor-Couette flow. *Journal of Physics: Conference Series* 14:143–156.

Ostilla, R., Stevens, R. J. A. M., Grossmann, S., Verzicco, R., and Lohse, D. 2013. Optimal Taylor-Couette flow: Direct numerical simulations. *Journal of Fluid Mechanics* 719:14–46.

Ostilla-Mónico, R., Van Der Poel, E. P., Verzicco, R., Grossmann, S., and Lohse, D. 2014. Exploring the phase diagram of fully turbulent Taylor-Couette flow. *Journal of Fluid Mechanics* 761:1–26.

Ostilla-Mónico, R., Verzicco, R., and Lohse, D. 2015. Effects of the computational domain size on direct numerical simulations of Taylor-Couette turbulence with stationary outer cylinder. *Physics of Fluids* 27:025110.

Pirrò, D. and Quadrio, M. 2008. Direct numerical simulation of turbulent Taylor-Couette flow. *European Journal of Mechanics, B/Fluids* 27:552–566.

Pope, S. B. 2000. Turbulent Flows. Cambridge University Press, Cambridge.

Popinet, S. 2009. An accurate adaptive solver for surface-tension-driven interfacial flows. *Journal of Computational Physics* 228:5838–5866.

Popinet, S. 2018. Numerical Models of Surface Tension. *Annual Review of Fluid Mechanics* 50:49–75.

Prosperetti, A. and Tryggvason, G. 2007. Computational methods for multiphase flow, volume 9780521847. Cambridge University Press, Cambridge.

Puckett, E. G., Almgren, A. S., Bell, J. B., Marcus, D. L., and Rider, W. J. 1997. A high-order projection method for tracking fluid interfaces in variable density incompressible flows. *Journal of Computational Physics* 130:269–282.

Renardy, Y. and Renardy, M. 2002. PROST: A parabolic reconstruction of surface tension for the volume-of-fluid method. *Journal of Computational Physics* 183:400–421.

Rider, W. J. and Kothe, D. B. 1998. Reconstructing Volume Tracking. *Journal of Computational Physics* 141:112–152.

Sanders, W. C., Winkel, E. S., Dowling, D. R., Perlin, M., and Ceccio, S. L. 2006. Bubble friction drag reduction in a high-Reynolds-number flat-plate turbulent boundary layer. *Journal of Fluid Mechanics* 552:353–380.

Scardovelli, R. and Zaleski, S. 2000. Analytical Relations Connecting Linear Interfaces and Volume Fractions in Rectangular Grids. *Journal of Computational Physics* 164:228–237.

Spandan, V., Meschini, V., Ostilla-Mónico, R., Lohse, D., Querzoli, G., de Tullio, M. D., and Verzicco, R. 2017a. A parallel interaction potential approach coupled with the immersed boundary method for fully resolved simulations of deformable interfaces and membranes. *Journal of Computational Physics* 348:567–590.

Spandan, V., Ostilla-Mónico, R., Verzicco, R., and Lohse, D. 2016. Drag reduction in numerical two-phase Taylor-Couette turbulence using an Euler-Lagrange approach. *Journal of Fluid Mechanics* 798:411–435.

Spandan, V., Verzicco, R., and Lohse, D. 2017b. Deformable ellipsoidal bubbles in Taylor-Couette flow with enhanced Euler-Lagrangian tracking. *Physical Review Fluids* 2.

Spandan, V., Verzicco, R., and Lohse, D. 2018. Physical mechanisms governing drag reduction in turbulent Taylor-Couette flow with finite-size deformable bubbles. *Journal of Fluid Mechanics* 849:R31–R313.

Sugiyama, K., Calzavarini, E., and Lohse, D. 2008. Microbubbly drag reduction in Taylor - Couette flow in the wavy vortex regime. *Journal of Fluid Mechanics* 608:21–41.

Sussman, M. and Puckett, E. G. 2000. A Coupled Level Set and Volume-of-Fluid Method for Computing 3D and Axisymmetric Incompressible Two-Phase Flows. *Journal of Computational Physics* 162:301–337.

Taylor, G. I. 1923. Stability of a Viscous Liquid Contained between Two Rotating Cylinders. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 223:289–343.

Tryggvason, G., Dabiri, S., Aboulhasanzadeh, B., and Lu, J. 2013. Multiscale considerations in direct numerical simulations of multiphase flows. *Physics of Fluids* 25:31302.

van den Berg, T. H., Luther, S., Lathrop, D. P., and Lohse, D. 2005. Drag Reduction in Bubbly Taylor-Couette Turbulence. *Physical Review Letters* 94:44501.

van der Poel, E. P., Ostilla-Mónico, R., Donners, J., and Verzicco, R. 2015. A pencil distributed finite difference code for strongly turbulent wall-bounded flows. *Computers and Fluids* 116:10–16.

Van Gils, D. P. M., Narezo Guzman, D., Sun, C., and Lohse, D. 2013. The importance of bubble deformability for strong drag reduction in bubbly turbulent Taylor–Couette flow. *Journal of Fluid Mechanics* 722:317–347.

Verschoof, R. A., van der Veen, R. C. A., Sun, C., and Lohse, D. 2016. Bubble Drag Reduction Requires Large Bubbles. *Physical Review Letters* 117:104502.

Verzicco, R. and Orlandi, P. 1996. A finite-difference scheme for three-dimensional incompressible flows in cylindrical coordinates. *Journal of Computational Physics* 123:402–414.

Virk, P. S. 1975. Drag reduction fundamentals.

Vreman, A. W. 2014. The projection method for the incompressible Navier-Stokes equations: The pressure near a no-slip wall. *Journal of Computational Physics* 263:353–374.

Wang, Z., Yang, J., and Stern, F. 2012. A new volume-of-fluid method with a constructed distance function on general structured grids. *Journal of Computational Physics* 231:3703–3722.

Wendt, F. 1933. Turbulente Strömungen zwischen zwei rotierenden konaxialen Zylindern. *Ingenieur-Archiv* 4:577–595.

Wesseling, P. 2001. Principles of Computational Fluid Dynamics, volume 29. Springer.

Zainali, A., Tofighi, N., Shadloo, M. S., and Yildiz, M. 2013. Numerical investigation of Newtonian and non-Newtonian multiphase flows using ISPH method. *Computer Methods in Applied Mechanics and Engineering* 254:99–113.

Zhang, A., Sun, P., and Ming, F. 2015. An SPH modeling of bubble rising and coalescing in three dimensions. *Computer Methods in Applied Mechanics and Engineering* 294:189–209.

Appendices

A: Derivation of curvature equations

In this appendix, we present a derivation of equations 70 from the equation for a general orthogonal coordinate system (67). We will derive the three equations separately based on the direction along which the surface is defined.

Let us first define the surface in cylindrical coordinates as $z = h(\theta, r)$. Thus, the Lamé coefficients are gives as $d_1 = r$, $d_2 = 1$ and $d_3 = 1$. Throughout these derivations, we will use the short-hand notation $h_\theta = rh_\phi$, $h_{\theta,r} = rh_{\phi,r}$ and $h_{\theta\theta} = r^2 h_{\phi\phi}$. The curvature is then given by:

$$\begin{aligned}
\kappa &= \frac{\left(1+|\nabla h|^2\right)\left(\nabla^2 h - \nabla^2 z\right) - \frac{1}{2}\nabla\left(1+|\nabla h|^2\right)\cdot(\nabla h - \hat{\mathbf{e}}_z)}{\left(1+|\nabla h|^2\right)^{3/2}} \\
&= \frac{(1+h_r^2+h_\phi^2)\left(h_{rr}+\frac{h_r}{r}+h_{\phi\phi}\right) - \frac{1}{2}\left(2(h_{rr}h_r + h_{r\phi}h_\phi - h_\phi^2/r)h_r + 2(h_{r\phi}h_r + h_{\phi\phi}h_\phi)h_\phi\right)}{(1+h_r^2+h_\phi^2)^{3/2}} \\
&= \frac{h_{rr}(1+h_\phi^2) + h_{\phi\phi}(1+h_r^2) - 2h_r h_\phi h_{r\phi} + \frac{h_r}{r}(1+h_r^2+2h_\phi^2)}{(1+h_r^2+h_\phi^2)^{3/2}}
\end{aligned}$$

As a check, we can calculate the curvature for a sphere of radius R, for which the mean curvature should equal $-2/R$ everywhere on the sphere. The surface is given as $z = h(\theta, r) = \sqrt{R^2 - r^2}$. The derivatives of the height function are given as: $h_r = -r/\sqrt{R^2-r^2}$, $h_{rr} = -R^2/(R^2-r^2)^{3/2}$, $h_\theta = h_{\theta\theta} = h_{r\theta} = 0$. This gives for the curvature:

$$\kappa = \frac{-R^2/\sqrt{R^2-r^2} + \frac{-1}{\sqrt{R^2-r^2}}\left(1+\frac{r^2}{R^2-r^2}\right)}{(1+r^2/(R^2-r^2))^{3/2}} = -\frac{2}{R} \tag{81}$$

Secondly, we can define the surface in cylindrical coordinates along the radial direction as $r = h(\theta, z)$, which implies $d_1 = r$ and $d_2 = d_3 = 1$. Substituting these into equation 67, we find the following expression for the curvature:

$$\begin{aligned}
\kappa &= \frac{\left(1+|\nabla h|^2\right)\left(\nabla^2 h - \nabla^2 r\right) - \frac{1}{2}\nabla\left(1+|\nabla h|^2\right)\cdot(\nabla h - \hat{\mathbf{e}}_r)}{\left(1+|\nabla h|^2\right)^{3/2}} \\
&= \frac{(1+h_\phi^2+h_z^2)\left(h_{\phi\phi}+h_{zz}-\frac{1}{r}\right) - \frac{1}{2}\left(2(h_\phi h_{\phi\phi} + h_z h_{z\phi})h_\phi + 2(h_\phi h_{z\phi} + h_z h_{zz})h_z - 2\frac{h_\phi^2}{r}\right)}{(1+h_\phi^2+h_z^2)^{3/2}} \\
&= \frac{h_{\phi\phi}(1+h_z^2) + h_{zz}(1+h_\phi^2) - 2h_\phi h_z h_{\phi z} - \frac{1}{r}(1+2h_\phi^2+h_z^2)}{(1+h_\phi^2+h_z^2)^{3/2}}
\end{aligned}$$

Lastly, we define the interface along the azimuthal direction as $\theta = h(r, z)$. Using the Lamé coefficients $d_1 = d_2 = 1$ and $d_3 = r$ we similarly find the curvature:

$$
\begin{aligned}
\kappa &= \frac{\left(\frac{1}{r^2}+|\nabla h|^2\right)\left(\nabla^2 h-\nabla^2\theta\right)-\frac{1}{2}\nabla\left(\frac{1}{r^2}+|\nabla h|^2\right)\cdot\left(\nabla h-\frac{1}{r}\hat{\mathbf{e}}_\theta\right)}{\left(\frac{1}{r^2}+|\nabla h|^2\right)^{3/2}} \\
&= \frac{\left(\frac{1}{r^2}+h_r^2+h_z^2\right)\left(h_{rr}+h_{zz}+\frac{h_r}{r}\right)-\frac{1}{2}\left(2\left(\frac{-1}{r^3}+h_r h_{rr}+h_z h_{rz}\right)h_r+2(h_r h_{rz}+h_z h_{zz})h_z\right)}{\left(\frac{1}{r^2}+h_r^2+h_z^2\right)^{3/2}} \\
&= \frac{h_{rr}\left(\frac{1}{r^2}+h_z^2\right)+h_{zz}\left(\frac{1}{r^2}+h_r^2\right)-2h_r h_z h_{rz}+\frac{h_r}{r}\left(\frac{2}{r^2}+h_r^2+h_z^2\right)}{\left(\frac{1}{r^2}+h_r^2+h_z^2\right)^{3/2}}
\end{aligned}
$$

Summarizing, we have derived the expressions for curvature based on height functions along the three directions θ, r and z, which are:

$$
\kappa_{\theta=h(r,z)} = \frac{h_{rr}\left(\frac{1}{r^2}+h_z^2\right)+h_{zz}\left(\frac{1}{r^2}+h_r^2\right)-2h_r h_z h_{rz}+\frac{h_r}{r}\left(\frac{2}{r^2}+h_r^2+h_z^2\right)}{\left(\frac{1}{r^2}+h_r^2+h_z^2\right)^{3/2}} \tag{82a}
$$

$$
\kappa_{r=h(\theta,z)} = \frac{h_{\phi\phi}(1+h_z^2)+h_{zz}(1+h_\phi^2)-2h_\phi h_z h_{\phi z}-\frac{1}{r}(1+2h_\phi^2+h_z^2)}{(1+h_\phi^2+h_z^2)^{3/2}} \tag{82b}
$$

$$
\kappa_{z=h(\theta,r)} = \frac{h_{\phi\phi}(1+h_r^2)+h_{rr}(1+h_\phi^2)-2h_\phi h_r h_{\phi r}+\frac{h_r}{r}(1+2h_\phi^2+h_r^2)}{(1+h_\phi^2+h_r^2)^{3/2}} \tag{82c}
$$

B: List of symbols

A list of variables and non-dimensional numbers are given below. Units are given in terms of mass M, length L and time T

Non-dimensional quantities

Symbol	Description
$Eö$	Eötvos number
Fr	Froude number
G	Dimensionless torque
Nu	Nusselt number
Nu_ω	Angular velocity-based Nusselt number
Pr	Prandtl number in convective flow
Ra	Rayleigh number
Re_i, Re_o	Reynolds number based on inner and outer cylinder velocities respectively
Ta	Taylor number
We	Weber number
Φ	Quasi-Prandtl number for TC flow
Γ	Aspect ratio of domain, $\Gamma = L_z/L_y$
η	Ratio of cylinder radii, $\eta = R_o/R_i$
Δ_ϵ	Relative error in energy balance

Greek symbols

Symbol	Description	Units
α^n	Sum of Adams-Bashfort coefficients, $\alpha^n = \gamma^n + \xi^n$	
γ^n	First Adams-Bashfort coefficient	
ϵ_u	Volume-averaged energy dissipation rate	L^2T^{-3}
$\epsilon_{u,0}$	Volume-averaged energy dissipation rate in purely diffusive flow	L^2T^{-3}
θ	Azimuthal coordinate or direction	
κ	Mean curvature	LL^{-1}
μ	Dynamic liquid phase viscosity	$ML^{-1}T^{-1}$
ν	Kinematic viscosity	L^2T^{-1}
ξ^n	Second Adams-Bashfort coefficient	
ρ	Liquid phase mass density	ML^{-3}
σ	Surface tension	MT^{-2}
τ_w	Wall normal stress	$ML^{-1}T^{-2}$
ϕ	Pressure correction term	$ML^{-1}T^{-2}$
ω	Angular velocity of the fluid	T^{-1}
ω_i, ω_o	Angular velocity of inner and outer cylinder resp.	T^{-1}

Latin symbols

Symbol	Description	Units
C^n	Discrete convection operator $u \cdot \nabla$ at time step n	
D	Discrete divergence operator	
d	Distance between inner and outer cylinder, $d \equiv r_o - r_i$	L
f	VOF indicator function	
G	Discrete gradient operator	
$\mathbf{g}$	Gravitational acceleration	LT^{-2}
h	Height function	L
J^ω	Angular velocity flux	L^4T^{-2}
J_0^ω	Angular velocity flux for laminar Couette flow	L^4T^{-2}
k_θ, k_z	Modified wave numbers for transformed Poisson equation	L^{-1}
L_θ, L_r, L_z	Domain size in the respective directions	L
$\mathbf{n}$	Normal vector of interface	
n	Time level index	
p	Fluid pressure	$ML^{-1}T^{-2}$
r_i, r_o	Radius of inner and outer cylinder respectively	L
r_a	Arithmetic mean of cylinder radii	L
r	Radial coordinate or direction	L
s	Source term for the Poisson equation	$ML^{-3}T^{-2}$
$\mathbf{S}$	Strain rate tensor, $\mathbf{S} = \frac{1}{2}(\nabla\mathbf{u} + \nabla\mathbf{u}^T)$	T^{-1}
t	Time	T
$\mathbf{u}$	Velocity vector with components $\mathbf{u} = (u_\theta, u_r, u_z)$	LT^{-1}
$\mathbf{u}^*$	Provisional velocity during time marching algorithm	LT^{-1}
V^n	Discrete viscosity operator $\nabla \cdot (2\mu\mathbf{S})$ at time step n	