Fully-coupled pressure-based two-fluid solver for the solution of turbulent fluid-particle systems

M. Riella[a,*], R. Kahraman[b], G. R. Tabor[a]

[a] *College of Engineering, Mathematics and Physical Sciences, University of Exeter, North Park Road, Exeter EX4 4QF, UK*
[b] *HiETA Technologies Ltd, Bristol & Bath Science Park, Dirac Crescent, Emersons Green, Bristol, BS16 7FR, UK*

Abstract

A fully-coupled pressure-based two-fluid solver for the solution of turbulent fluid-particle flows is presented. The numerical framework details several crucial aspects: implicit treatment of the phase-velocity-pressure coupling, the implicit treatment of inter-phase momentum transfer and finally the solution algorithm. The two-fluid solver is implemented within the open source tool-box `foam-extend` which is a community driven fork of OpenFOAM. The coupled solver is verified against a standard segregated implementation of the two-fluid solution algorithm and validated against benchmark experimental data. The coupled solver shows marked improvements in convergence, stability and solution time. The coupled implementation is capable of solving to a tolerance that is six orders of magnitude smaller in residual error and 1.7 times quicker than the segregated solver. Additionally, the sequentially solved system of phase-energies experienced performance improvements when solved in conjunction with the coupled solver.

*Corresponding author
Email address: `maatthew.riella@polito.it` (M. Riella)

Keywords: `Two-fluid`, `Pressure-based`, `Fully-coupled`, `Turbulence`

1. Introduction

In Computational Fluid Dynamics (CFD), the coupling of pressure and velocity has proven to be one of the major challenges when solving the Naiver-Stokes equations (NSE) [14, 25]. Traditionally, this problem has been solved by solving the NSE in a decoupled manner [1, 14] by employing a pressure-correction approach. First, an estimate for the velocity field is found by the momentum equation using an initial guess of the pressure field. Then a Poisson equation for pressure is solved for by taking the divergence of the momentum equation. After its solution, the velocity field is corrected to ensure continuity is satisfied.

This pressure-based (meaning a pressure-correction equation is formulated) approach make up two of the most widely used algorithms in CFD; SIMPLE [25] and PISO [17]. Typically, these system of equations are solved in a segregated manner. The velocity and pressure are decoupled within the matrix with each variable being solved separately. The unknowns in each respective equation i.e. pressure in the momentum equation and momentum in the pressure equation, are treated explicitly. The computational overhead required to store and operate on a single variable matrix is cheap and the emphasis is thus put on the time taken to do repeated operations. Two of the major drawbacks of these algorithms is the use of arbitrary under-relaxation factors, due to high rates of change in dependent variables and the slow convergence rates, due to the decoupling of velocity and pressure. In transient flows, the time-step size is used as an effective under-relaxation method in order to

cope with abrupt rates of change in the dependant variable.

Recently, pressure-based approach has been employed within a Finite-Volume-Method (FVM) block-coupled framework [6, 11, 12, 39] although other coupling approaches have been developed: including the direct methods of [2, 5, 20, 23, 24, 35] i.e. solved in their primitive form and the control volume finite element method (CVFEM) [16, 21, 26, 37, 36]. Within the block-coupled framework the system of equations are solved within one single block-matrix in which the influence of velocity-pressure coupling can be introduced through inter-equation coupling terms. This ensures that the system of equations are solved implicitly using the current iteration values. As a single block-matrix of coefficients needs to be solved for the computational overhead is high, unlike in the segregated approach. Due to the current computational power available, these methods are receiving increased attention as they do not require under-relaxation and show major improvements in convergence, stability and robustness.

The pressure-based approach was first applied to co-located grids in the CVFEM framework by Webster [36, 37]. The approach shows dramatic improvements in convergence on both structured and unstructured grids in comparison to the SIMPLE algorithm and shows superior performance on denser meshes. This framework [16, 26, 36, 37] has since been extended to multiphase applications - namely the two-fluid model [3]. In this approach the entire system is coupled i.e. two momentum equations and a pressure field leading to tighter inter-equation coupling. The coupled solver showed far superior performance over its segregated counterpart with improvements in the number of iterations and computational time.

The two-fluid model is particularly well placed for such an extension due to the formulation of the governing equations. Two phase-momentum equations are coupled via a shared pressure field with inter-phase coupling through drag. When solved within a segregated framework the system of equations are solved in a decoupled manner in which the decoupled phase-velocity-pressure and inter-phase drag terms are treated explicitly putting a computational constraint on the solution time and adversely affecting convergence.

In a FVM framework, Darwish et al. [10] has recently proposed a two-fluid fully-coupled pressure-based solver in which their single-phase framework [11, 12] is extended to a multiphase framework. The governing equations are solved within a fully conservative formulation i.e. the volume fraction and density are left in the momentum equations, typically used to capture compressibility effects. They derived their model in a 2D framework and verify their results on 1D laminar test cases showing a rate of solution acceleration between 1.3 and 4.6.

More recently, Ferreira et al. [13] proposed a fully-coupled pressure-based multi-fluid framework. In their work they solve the phase-intensive formulation i.e. dividing out by volume fraction and density and employing the Compact Momentum Interpolation (CMI) practice of Cubero et al. [9] and guess-and-correct procedure shown in Darwish et al. [10]. Overall, this treatment was shown to enhance stability and convergence through the correct treatment of the temporal, drag and body force interpolation especially when a large drag force was present. The multi-fluid solver is verified on 2D laminar test cases showing superior performance when compared to the segregated solver reporting computational speedups from 4.6 to 9.3 times.

In this work we propose a fully-coupled pressure-based two-fluid solver for turbulent fluid-particle flows. The two-fluid model [29] is implicitly coupled in phase-velocity-pressure and inter-phase drag and employing the CMI practice of Cubero et al. [9]. The framework is implemented within the open-source tool-box `foam-extend` which is a community driven fork of OpenFOAM. The fully-coupled two-fluid pressure-based solver for turbulent fluid-particle flows is verified against a segregated implementation and validated against benchmark validation data. Additionally, the performance of the coupled and segregated solvers are compared and contrasted.

2. RA-TFM governing equations

We begin with a simplified set of equations from the Reynolds-Averaged Two-Fluid model (RA-TFM) of Fox [15]. The continuity and momentum equations of the particle- and fluid-phases are as follows:

$$\frac{\partial(\alpha_p\rho_p)}{\partial t} + \nabla\cdot(\alpha_p\rho_p\mathbf{u}_p) = 0, \tag{1}$$

$$\frac{\partial(\alpha_f\rho_f)}{\partial t} + \nabla\cdot(\alpha_f\rho_f\mathbf{u}_f) = 0, \tag{2}$$

$$\begin{aligned}\frac{\partial(\alpha_p\rho_p\mathbf{u}_p)}{\partial t} + \nabla\cdot(\alpha_p\rho_p\mathbf{u}_p\mathbf{u}_p) = \nabla\cdot(\alpha_p\rho_p\overline{\mathbf{R}}_{\mathrm{eff,p}}) + \beta(\mathbf{u}_f-\mathbf{u}_p) - \beta\frac{\nu_{ft}}{\mathrm{Sc}_{fs}\alpha_p\alpha_f}\nabla\alpha_p \\ -\nabla p_p - \alpha_p\nabla p_f + \alpha_p\rho_p\mathbf{g},\end{aligned} \tag{3}$$

$$\begin{aligned}\frac{\partial(\alpha_f\rho_f\mathbf{u}_f)}{\partial t} + \nabla\cdot(\alpha_f\rho_f\mathbf{u}_f\mathbf{u}_f) = \nabla\cdot(\alpha_f\rho_f\overline{\mathbf{R}}_{\mathrm{eff,f}}) + \beta(\mathbf{u}_p-\mathbf{u}_f) + \beta\frac{\nu_{ft}}{\mathrm{Sc}_{fs}\alpha_p\alpha_f}\nabla\alpha_p \\ -\alpha_f\nabla p_f + \alpha_f\rho_f\mathbf{g}.\end{aligned} \tag{4}$$

The accompanying phase-energy transport equations that make up the complete RA-TFM can be found in Table 1. In this work they are treated sequentially and are therefore not given special treatment here. Definitions of the aforementioned equations can be found in Tables 4 & 5.

Table 1: RA-TFM phase-energy equations.

The particle-phase energy transport equations:

$$\frac{\partial(\alpha_p\rho_p k_p)}{\partial t} + \nabla\cdot(\alpha_p\rho_p k_p\mathbf{u}_p) = \nabla\cdot\Big(\mu_p + \frac{\mu_{pt}}{\sigma_{pk}}\Big)\nabla k_p + \alpha_p\rho_p\Pi_p - \alpha_p\rho_p\varepsilon_p \\ +\beta(k_{fp} - k_p) \tag{5}$$

$$\frac{\partial(\alpha_p\rho_p\varepsilon_p)}{\partial t} + \nabla\cdot(\alpha_p\rho_p\varepsilon_p\mathbf{u}_p) = \nabla\cdot\Big(\mu_p + \frac{\mu_{pt}}{\sigma_{pk}}\Big)\nabla\varepsilon_p + \frac{\varepsilon_p}{k_p}(C_1\alpha_p\rho_p\Pi_p - C_2\alpha_p\rho_p\varepsilon_p) \\ +\beta(\varepsilon_{fp} - \varepsilon_p) \tag{6}$$

$$\frac{3}{2}\Big[\frac{\partial(\alpha_p\rho_p\Theta_p)}{\partial t} + \nabla\cdot(\alpha_p\rho_p\Theta_p\mathbf{u}_p)\Big] = \nabla\cdot\Big(\kappa_\Theta + \frac{3\mu_{pt}}{2Pr_{pt}}\Big)\nabla\Theta_p + 2\mu_p\overline{\mathbf{S}}_\mathbf{p}:\overline{\mathbf{S}}_\mathbf{p} \\ -p_p\nabla\cdot\mathbf{u}_p + \alpha_p\rho_p\varepsilon_p - 3\beta\Theta_p \tag{7}$$

The fluid-phase energy transport equations:

$$\frac{\partial(\alpha_f\rho_f k_f)}{\partial t} + \nabla\cdot(\alpha_f\rho_f k_f\mathbf{u}_f) = \nabla\cdot\Big(\mu_t + \frac{\mu_{ft}}{\sigma_{fk}}\Big)\nabla k_f + \alpha_f\rho_f\Pi_f - \alpha_f\rho_f\varepsilon_f \\ +\beta(k_{fp} - k_f) \tag{8}$$

$$\frac{\partial(\alpha_f\rho_f\varepsilon_f)}{\partial t} + \nabla\cdot(\alpha_f\rho_f\varepsilon_f\mathbf{u}_f) = \nabla\cdot\Big(\mu_t + \frac{\mu_{ft}}{\sigma_{fk}}\Big)\nabla\varepsilon_f + \frac{\varepsilon_f}{k_f}\Big[C_1\alpha_f\Pi_f - C_2\alpha_f\rho_f\varepsilon_f\Big] \\ +C_3\beta(\varepsilon_{fp} - \varepsilon_f) \tag{9}$$

3. Segregated solution algorithm

3.1. Phase intensive momentum equations

Here we follow the phase intensive formulation of Rusche [32], Weller [38]. For simplicity the turbulent dispersion term is now denoted as $\mathscr{D}$, separating the drag contributions into explicit and implicit terms and dividing by both the phase fraction and density we are left with:

$$\begin{aligned}\frac{\partial \mathbf{u}_p}{\partial t} + \nabla \cdot (\mathbf{u}_p \mathbf{u}_p) - \mathbf{u}_p \nabla \cdot \mathbf{u}_p + \frac{\nabla \alpha_p}{\alpha_p^*} \cdot \overline{\mathbf{R}}^c_{\mathrm{eff,p}} + \nabla \cdot \overline{\mathbf{R}}^c_{\mathrm{eff,p}} - \nabla \cdot (\nu_{\mathrm{eff,p}} \nabla \mathbf{u}_p) + \frac{\beta \mathbf{u}_p}{\alpha_p \rho_p} \\ -\nabla \cdot \left(\nu_{\mathrm{eff,p}} \frac{\nabla \alpha_p}{\alpha_p^*} \mathbf{u}_p\right) + \mathbf{u}_p \nabla \cdot \left(\nu_{\mathrm{eff,p}} \frac{\nabla \alpha_p}{\alpha_p^*}\right) \\ = \frac{\beta \mathbf{u}_f}{\alpha_p \rho_p} - \frac{\beta \mathscr{D} \nabla \alpha_p}{\alpha_p \rho_p} - \frac{\nabla p_p}{\alpha_p^* \rho_p} - \frac{\nabla p_f}{\rho_p} + \mathbf{g},\end{aligned} \tag{10}$$

$$\begin{aligned}\frac{\partial \mathbf{u}_f}{\partial t} + \nabla \cdot (\mathbf{u}_f \mathbf{u}_f) - \mathbf{u}_f \nabla \cdot \mathbf{u}_f + \frac{\nabla \alpha_f}{\alpha_f^*} \cdot \overline{\mathbf{R}}^c_{\mathrm{eff,f}} + \nabla \cdot \overline{\mathbf{R}}^c_{\mathrm{eff,f}} - \nabla \cdot (\nu_{\mathrm{eff,f}} \nabla \mathbf{u}_f) + \frac{\beta \mathbf{u}_f}{\alpha_f \rho_f} \\ -\nabla \cdot \left(\nu_{\mathrm{eff,f}} \frac{\nabla \alpha_f}{\alpha_f^*} \mathbf{u}_f\right) + \mathbf{u}_f \nabla \cdot \left(\nu_{\mathrm{eff,f}} \frac{\nabla \alpha_f}{\alpha_f^*}\right) \\ = \frac{\beta \mathbf{u}_p}{\alpha_f \rho_f} + \frac{\beta \mathscr{D} \nabla \alpha_p}{\alpha_f \rho_f} - \frac{\nabla p_f}{\rho_f} + \mathbf{g},\end{aligned} \tag{11}$$

where $\alpha_p^* = \alpha_p + \delta$ and $\alpha_f^* = \alpha_f + \delta$, and δ is introduced to avoid a division by zero and is $\mathcal{O}(10^{-6})$. It is important to clarify the behaviour of terms with the volume fraction in their denominator. The drag terms containing the phase-velocities i.e. β in which the numerator contains $\alpha_p \alpha_f$ (see Table 4) which ensures the correct behavior of the function as $\alpha_p \to 0$. The turbulent dispersion term contains the gradient of volume fraction which in the limit $\alpha_p \to 0$ means that the ratio approaches zero.

This ensures that the momentum equations are able to be solved everywhere within the domain despite diminishing particle volume fractions.

As it can be seen from the system of equations in Eqs. 3 & 4 no diffusive flux exists that can be treated implicitly. This can have advantages when solving the equations i.e enhanced matrix positively and diagonal dominance. Therefore, following Weller [38], Rusche [32] the Reynolds stress term can be rewritten into a diffusive and corrective component:

$$\begin{aligned}
\overline{\mathbf{R}}_{\mathrm{eff,i}} &= \overline{\mathbf{R}}_{\mathrm{eff,i}} + \nu_{\mathrm{eff,i}}\nabla\mathbf{u}_i - \nu_{\mathrm{eff,i}}\nabla\mathbf{u}_i \\
&= -\nu_{\mathrm{eff,i}}(\nabla\mathbf{u}_i + \nabla^T\mathbf{u}_i) + \frac{2}{3}\nu_{\mathrm{eff,i}}\mathbf{I}\nabla\cdot\mathbf{u}_i \\
&+ \nu_{\mathrm{eff,i}}\nabla\mathbf{u}_i - \nu_{\mathrm{eff,i}}\nabla\mathbf{u}_i \\
&= (-\nu_{\mathrm{eff,i}}\nabla^T\mathbf{u}_i + \frac{2}{3}\nu_{\mathrm{eff,i}}\mathbf{I}\nabla\cdot\mathbf{u}_i) - \nu_{\mathrm{eff,i}}\nabla\mathbf{u}_i \\
&= \overline{\mathbf{R}}^c_{\mathrm{eff,i}} - \nu_{\mathrm{eff,i}}\nabla\mathbf{u}_i.
\end{aligned} \tag{12}$$

3.2. Discretisation of the intensive momentum equations

First, we discretise the left hand side of the equation which contains the convective, diffusive and implicit-drag transport terms:

$$\begin{aligned}
\mathcal{T}_p := \left[\!\left[\frac{\partial[\![\mathbf{u}_p]\!]}{\partial t}\right]\!\right] + \left[\!\left[\nabla\cdot(\mathbf{u}_p[\![\mathbf{u}_p]\!])\right]\!\right] - \left[\!\left[(\nabla\cdot\mathbf{u}_p)[\![\mathbf{u}_p]\!]\right]\!\right] + \frac{\nabla\alpha_p}{\alpha_p^*}\cdot\overline{\mathbf{R}}^c_{\mathrm{eff,p}} + \nabla\cdot\overline{\mathbf{R}}^c_{\mathrm{eff,p}} \\
-\left[\!\left[\nabla\cdot(\nu_{\mathrm{eff,p}}\nabla[\![\mathbf{u}_p]\!])\right]\!\right] - \left[\!\left[\nabla\cdot(\nu_{\mathrm{eff,p}}\frac{\nabla\alpha_p}{\alpha_p^*}[\![\mathbf{u}_p]\!])\right]\!\right] \\
-\left[\!\left[\nabla\cdot(\nu_{\mathrm{eff,p}}\frac{\nabla\alpha_p}{\alpha_p^*})[\![\mathbf{u}_p]\!])\right]\!\right] + \left[\!\left[\frac{\beta[\![\mathbf{u}_p]\!]}{\alpha_p\rho_p}\right]\!\right],
\end{aligned} \tag{13}$$

$$\mathcal{T}_f := \left[\!\left[\frac{\partial[\![\mathbf{u}_f]\!]}{\partial t}\right]\!\right] + \left[\!\left[\nabla\cdot(\mathbf{u}_f[\![\mathbf{u}_f]\!])\right]\!\right] - \left[\!\left[(\nabla\cdot\mathbf{u}_f)[\![\mathbf{u}_f]\!]\right]\!\right] + \frac{\nabla\alpha_f}{\alpha_f^*}\cdot\overline{\mathbf{R}}^c_{\text{eff,f}} + \nabla\cdot\overline{\mathbf{R}}^c_{\text{eff,f}}$$
$$-\left[\!\left[\nabla\cdot(\nu_{\text{eff,f}}\nabla[\![\mathbf{u}_f]\!])\right]\!\right] - \left[\!\left[\nabla\cdot(\nu_{\text{eff,f}}\frac{\nabla\alpha_f}{\alpha_f^*}[\![\mathbf{u}_f]\!])\right]\!\right] \quad (14)$$
$$-\left[\!\left[\nabla\cdot(\nu_{\text{eff,f}}\frac{\nabla\alpha_f}{\alpha_f^*})[\![\mathbf{u}_f]\!])\right]\!\right] + \left[\!\left[\frac{\beta[\![\mathbf{u}_f]\!]}{\alpha_f\rho_f}\right]\!\right],$$

where $[\![\cdot]\!]$ is the implicit dicretisation of the term, $\mathcal{T}_p$ & $\mathcal{T}_f$ represents the numerical coefficients of each respective algebraic system given by the discretisation. The second and third terms on the RHS represent convection and have been split up into a convection term minus a divergence terms as it enhances boundedness of the solution.

The discretised momentum equations, $\mathcal{T}_p$ & $\mathcal{T}_f$ represents the system of algebraic equations which are written in the form,

$$(\mathcal{T}_p)_{coeffs}\mathbf{u}_p = (\mathcal{T}_p)_s, \quad (15a)$$
$$(\mathcal{T}_f)_{coeffs}\mathbf{u}_f = (\mathcal{T}_f)_s, \quad (15b)$$

where $()_{coeffs}$ represents the off-diagonal and diagonal coefficients and $()_s$ represents the source terms i.e. explicit terms. This discretised form of the momentum equations will be revisited once the source terms on the RHS have been addressed.

Now addressing the RHS of Eq. 10 & 11 which reads as

$$\ldots = \frac{\beta\mathbf{u}_f}{\alpha_p\rho_p} - \frac{\beta\mathscr{D}\nabla\alpha_p}{\alpha_p\rho_p} - \frac{\nabla p_f}{\rho_p} - \frac{\nabla p_p}{\alpha_p^*\rho_p} + \mathbf{g}, \quad (16a)$$
$$\ldots = \frac{\beta\mathbf{u}_p}{\alpha_f\rho_f} + \frac{\beta\mathscr{D}\nabla\alpha_p}{\alpha_f\rho_f} - \frac{\nabla p_f}{\rho_f} + \mathbf{g}. \quad (16b)$$

Following the solution procedure of Weller [38] all terms on the RHS are evaluated

at cell faces. In order to avoid checker-boarding in the solution, which is a prevalent problem on collocated grids due to the storage of values at cell centres and interpolating onto the face, the group of terms on the RHS are treated in a Rhie-Chow like manner [27].

3.3. Phase momentum flux correction equations

Now a semi-discretised formulation of both the particle- and fluid-phase can be written. Invoking Eqs. 15 and splitting up the total coefficients appearing in each system into a diagonal, $\mathbf{A}_i$ and an explicit, $\mathbf{H}_i$ [19] contribution. The latter consisting of two parts, the neighbouring coefficients, $()_N$ multiplied by its respective phase velocity and the source terms, $\mathbf{H}_i = -(\mathbf{A}_i)_N\mathbf{u}_i + (\mathbf{A}_i)_S$. The equations can then be written as:

$$\mathbf{A}_p\mathbf{u}_p = \mathbf{H}_p + \frac{\beta\mathbf{u}_f}{\alpha_p\rho_p} - \frac{\beta\mathscr{D}\nabla\alpha_p}{\alpha_p\rho_p} - \frac{\nabla p_f}{\rho_p} - \frac{\nabla p_p}{\alpha_p^*\rho_p} + \mathbf{g}, \tag{17a}$$

$$\mathbf{A}_f\mathbf{u}_f = \mathbf{H}_f + \frac{\beta\mathbf{u}_p}{\alpha_f\rho_f} + \frac{\beta\mathscr{D}\nabla\alpha_p}{\alpha_f\rho_f} - \frac{\nabla p_f}{\rho_f} + \mathbf{g}. \tag{17b}$$

Rearranging Eqs. 17 gives the phase momentum correction equations, note these equations are not used in the solution algorithm, but are required to derive a flux predictor and corrector:

$$\mathbf{u}_p = \frac{\mathbf{H}_p}{\mathbf{A}_p} + \frac{\beta\mathbf{u}_f}{\alpha_p\rho_p\mathbf{A}_p} - \frac{\beta\mathscr{D}\nabla\alpha_p}{\alpha_p\rho_p\mathbf{A}_p} - \frac{\nabla p_f}{\rho_p\mathbf{A}_p} - \frac{\nabla p_p}{\alpha_p^*\rho_p\mathbf{A}_p} + \frac{\mathbf{g}}{\mathbf{A}_p}, \tag{18a}$$

$$\mathbf{u}_f = \frac{\mathbf{H}_f}{\mathbf{A}_f} + \frac{\beta\mathbf{u}_p}{\alpha_f\rho_f\mathbf{A}_f} + \frac{\beta\mathscr{D}\nabla\alpha_p}{\alpha_f\rho_f\mathbf{A}_f} - \frac{\nabla p_f}{\rho_f\mathbf{A}_f} + \frac{\mathbf{g}}{\mathbf{A}_f}. \tag{18b}$$

3.4. Construction of the pressure equation

In order to derive a pressure equation the continuity equation is enforced globally. The global continuity equation thus reads:

$$\nabla \cdot [(\alpha_p)_f \phi_p + (\alpha_f)_f \phi_f] = 0, \tag{19}$$

where the subscript $()_f$ denotes the face value which is found through linear interpolation i.e. central differencing and $\phi_i = \mathbf{u}_{i,f} \cdot \mathbf{S}_f$ is the volumetric face flux where subscript f is used to denote variables that are evaluated at the control volume's face. From here the face fluxes are found by interpolating the momentum correction equation (Eqs. 18) onto face centres using Rhie-Chow interpolation [27]. The interpolation increases pressure-velocity coupling by introducing cell-to-cell pressure coupling by evaluating the gradient of pressure on cell faces using the neighbouring cell centre contribution. Using central differencing and denoting the gradient at a face as, ∇_f, we can write

$$\phi_p = \phi_p^* - \frac{1}{\rho_p (\mathbf{A}_p)_f} \nabla_f^\perp p_f |\mathbf{S}_f|, \tag{20a}$$

$$\phi_f = \phi_f^* - \frac{1}{\rho_f (\mathbf{A}_f)_f} \nabla_f^\perp p_f |\mathbf{S}_f|, \tag{20b}$$

where $\nabla_f^\perp \phi_i$ is the face normal gradient which is the inner product of the face gradient, $\mathbf{n} \cdot \nabla_f \phi_i$. The flux prediction terms, ϕ_p^* & ϕ_f^* are written as:

$$\begin{aligned} \phi_p^* = \left(\frac{\mathbf{H}_p}{\mathbf{A}_p} \right)_f \cdot \mathbf{S}_f + \frac{(\beta)_f}{(\alpha_p)_f \rho_p (\mathbf{A}_p)_f} \phi_f - \frac{(\beta)_f (\mathscr{D})_f}{(\alpha_p)_f \rho_p (\mathbf{A}_p)_f} \nabla_f^\perp \alpha_p |\mathbf{S}_f| \\ - \frac{1}{(\alpha_p^*)_f \rho_p (\mathbf{A}_p)_f} \nabla_f^\perp p_p |\mathbf{S}_f| + \frac{\mathbf{g}}{(\mathbf{A}_p)_f} \cdot \mathbf{S}_f, \end{aligned} \tag{21}$$

$$\phi_f^* = \left(\frac{\mathbf{H}_f}{\mathbf{A}_f}\right)_f \cdot \mathbf{S}_f + \frac{(\beta)_f}{(\alpha_f)_f \rho_f (\mathbf{A}_f)_f}\phi_p - \frac{(\beta)_f (\mathscr{D})_f}{(\alpha_f)_f \rho_f (\mathbf{A}_f)_f}\nabla_f^{\perp}\alpha_p |\mathbf{S}_f| + \frac{\mathbf{g}}{(\mathbf{A}_f)_f} \cdot \mathbf{S}_f. \tag{22}$$

Now the pressure equation can be constructed by substituting Eqs. 20 into Eq. 19 which reads:

$$\left[\!\left[\nabla \cdot \left(D_p \nabla_f [\![p_f]\!] \cdot \mathbf{S}_f\right)\right]\!\right] = \nabla \cdot \left((\alpha_p)_f \phi_p^* + (\alpha_f)_f \phi_f^*\right), \tag{23}$$

where

$$D_p = \frac{(\alpha_p)_f}{\rho_p (\mathbf{A}_p)_f} + \frac{(\alpha_f)_f}{\rho_f (\mathbf{A}_f)_f}, \tag{24}$$

is the pressure diffusivity matrix and the pressure gradient has been discretised implicitly on the LHS as a diffusion term i.e. Laplacian. Essentially a shared or mixture pressure field is solved for, this ensures that continuity is obeyed throughout as the coupling is provided through the pressure equation.

Once this equation has been solved the phase fluxes need to be updated to satisfy continuity, as in the predictor step the influence of the pressure gradient is removed, this can be achieved by solving Eq. 20. From this stage the solution does not completely satisfy continuity as the velocities, which are stored at the cell centres, need to be corrected with the influence of the pressure gradient.

This is achieved by invoking:

$$\mathbf{u}_p = \frac{\mathbf{H}_p}{\mathbf{A}_p} + \left[\phi_p^* - \frac{1}{\rho_p (\mathbf{A}_p)_f} \nabla_f^{\perp} p_f |\mathbf{S}_f| \right]_{f \to c}, \tag{25a}$$

$$\mathbf{u}_f = \frac{\mathbf{H}_f}{\mathbf{A}_f} + \left[\phi_f^* - \frac{1}{\rho_f (\mathbf{A}_f)_f} \nabla_f^{\perp} p_f |\mathbf{S}_f| \right]_{f \to c}, \tag{25b}$$

where the subscript $f \to c$ denotes a vector field reconstruction from face flux values to cell centre values. The influence of the gradient of pressure is incorporated into the reconstruction of the phase velocity - this ensures the phase velocity obeys continuity.

3.5. Solution of the phase-mixed continuity equation

In practice the phase-mixed continuity equation is solved first based on the initial conditions but for the sake of logical progression is presented now. Following Weller [38] the particle phase continuity equation Eq. 19 can be reformulated as:

$$\frac{\partial \alpha_p}{\partial t} + \nabla \cdot (\mathbf{u}_T \alpha_p) + \nabla \cdot (\mathbf{u}_r \alpha_p \alpha_f) = 0, \tag{26}$$

where $\mathbf{u}_T = \alpha_p \mathbf{u}_p + \alpha_f \mathbf{u}_f$ is the mixture velocity and $\mathbf{u}_r = \mathbf{u}_p - \mathbf{u}_f$ is the relative velocity. This equation can then be discretised as

$$\left[\!\left[\frac{\partial [\![\alpha_p]\!]}{\partial t} \right]\!\right] + \left[\!\left[\nabla \cdot \left(\phi [\![\alpha_p]\!] \right) \right]\!\right] + \left[\!\left[\nabla \cdot \left(\phi_{r,p} [\![\alpha_p]\!] \right) \right]\!\right] = 0, \tag{27}$$

where $\phi_{r,p} = (\alpha_f)_f \phi_r$ and $\phi_r = \phi_p - \phi_f$. The second term on the LHS is ensured to be bounded between 0 and 1 due to the mixture flux, $\phi = \mathbf{u}_{p,f} \cdot \mathbf{S}_f + \mathbf{u}_{f,f} \cdot \mathbf{S}_f$ satisfying the mixture continuity equation. The third term is now non-linear and requires a

Total Variation Diminishing (TVD) scheme to ensure the term is bounded between 0 and 1. As an aside the particles volume fraction should be bounded at a much lower value i.e. its maximum packing limit ≈ 0.62. This can be achieved by including the particle pressure calculation directly in the continuity equation. Interested readers are referred to **?**].

An overview of the numerical procedure can be found below:

The numerical procedure adopted in the segregated algorithm:

1. Solve the volume fraction (Eq. 27).
2. Construct $\mathbf{A}_i$ in each phase (Eqs. 15).
3. Enter PISO-Loop:
   (a) Predict fluxes using Eqs. 21 & 22.
   (b) Construct and solve the pressure equation (Eq. 23).
   (c) Correct the phase fluxes using Eqs. 20.
   (d) Reconstruct the phase velocities using Eqs. 25.
4. Solve the system of phase energy equations.
5. Advance in time.

4. Coupled solution algorithm

4.1. Semi-discretised momentum equations

The phase-intensive formulation of the momentum equations are implemented in an analogous manner to Ferreira et al. [13]. First, we start at the semi-discretised equations as presented above:

$$\mathbf{A}_p\mathbf{u}_p = \mathbf{H}_p + \frac{\beta\mathbf{u}_f}{\alpha_p\rho_p} - \frac{\beta\mathscr{D}\nabla\alpha_p}{\alpha_p\rho_p} - \frac{\nabla p_f}{\rho_p} - \frac{\nabla p_p}{\alpha_p^*\rho_p} + \mathbf{g}, \tag{28a}$$

$$\mathbf{A}_f\mathbf{u}_f = \mathbf{H}_f + \frac{\beta\mathbf{u}_p}{\alpha_f\rho_f} + \frac{\beta\mathscr{D}\nabla\alpha_p}{\alpha_f\rho_f} - \frac{\nabla p_f}{\rho_f} + \mathbf{g}. \tag{28b}$$

From here we follow Cubero et al. [9] and separate out the temporal and drag coefficients from the semi-discretised equations. Additionally, the turbulent dispersion and gravity are absorbed into the $\mathbf{H}_i$ operator as well as the particle pressure for the particle phase. Which now reads as:

$$[\mathbf{A}_p + \mathbf{A}_{Tp} + \mathbf{A}_{Dp}]\mathbf{u}_p = \mathbf{H}_p + \mathbf{H}_{Tp} + \frac{\beta\mathbf{u}_f}{\alpha_p\rho_p} - \frac{\nabla p_f}{\rho_p}, \tag{29a}$$

$$[\mathbf{A}_f + \mathbf{A}_{Tf} + \mathbf{A}_{Df}]\mathbf{u}_f = \mathbf{H}_f + \mathbf{H}_{Tf} + \frac{\beta\mathbf{u}_p}{\alpha_f\rho_f} - \frac{\nabla p_f}{\rho_f}, \tag{29b}$$

where the time coefficient for each phase, considering a first-order Euler scheme with a fixed time step, and the drag coefficient is defined as:

$$\mathbf{A}_{Tp} = \frac{\mathbf{H}_{Tp}}{\mathbf{u}_p^{t-1}}, \qquad \mathbf{A}_{Dp} = \frac{\beta}{\alpha_p\rho_p}, \tag{30a}$$

$$\mathbf{A}_{Tf} = \frac{\mathbf{H}_{Tf}}{\mathbf{u}_f^{t-1}}, \qquad \mathbf{A}_{Df} = \frac{\beta}{\alpha_f\rho_f}. \tag{30b}$$

Now we divide each side of Eq. 29 by the diagonal coefficient, $\mathbf{A}_i$ that contains the advection and the implicit contribution of the shear stress terms, which now reads:

$$[1 + \mathrm{d}_{Tp} + \mathrm{d}_{Dp}]\mathbf{u}_p = \tilde{\mathbf{u}}_p + \mathrm{d}_{Tp}\mathbf{u}_p^{t-1} + \mathrm{d}_{Dp}\mathbf{u}_f - \frac{\nabla p_f}{\rho_p \mathbf{A}_p}, \tag{31a}$$

$$[1 + \mathrm{d}_{Tf} + \mathrm{d}_{Df}]\mathbf{u}_f = \tilde{\mathbf{u}}_f + \mathrm{d}_{Tf}\mathbf{u}_f^{t-1} + \mathrm{d}_{Df}\mathbf{u}_p - \frac{\nabla p_f}{\rho_f \mathbf{A}_f}, \tag{31b}$$

with the pseudo-velocities defined as:

$$\tilde{\mathbf{u}}_p = \frac{\mathbf{H}_p}{\mathbf{A}_p}, \qquad \tilde{\mathbf{u}}_f = \frac{\mathbf{H}_f}{\mathbf{A}_f}, \tag{32}$$

and the coefficients for time, which give a ratio of the temporal to steady coefficients, are defined as:

$$\mathrm{d}_{Tp} = \frac{\mathbf{A}_{Tp}}{\mathbf{A}_p}, \qquad \mathrm{d}_{Tf} = \frac{\mathbf{A}_{Tf}}{\mathbf{A}_f}, \tag{33}$$

and the coefficients for drag, which give a ratio of the drag to steady coefficients, are defined as:

$$\mathrm{d}_{Dp} = \frac{\mathbf{A}_{Dp}}{\mathbf{A}_p}, \qquad \mathrm{d}_{Df} = \frac{\mathbf{A}_{Df}}{\mathbf{A}_f}. \tag{34}$$

Then the approximations for each phase velocity can be obtained as:

$$\mathbf{u}_p = \frac{1}{1 + \mathrm{d}_{Tp} + \mathrm{d}_{Dp}} \left[\tilde{\mathbf{u}}_p + \mathrm{d}_{Tp}\mathbf{u}_p^{t-1} + \mathrm{d}_{Dp}\mathbf{u}_f - \frac{\nabla p_f}{\rho_p \mathbf{A}_p} \right], \tag{35a}$$

$$\mathbf{u}_f = \frac{1}{1 + \mathrm{d}_{Tf} + \mathrm{d}_{Df}} \left[\tilde{\mathbf{u}}_f + \mathrm{d}_{Tf}\mathbf{u}_f^{t-1} + \mathrm{d}_{Df}\mathbf{u}_p - \frac{\nabla p_f}{\rho_f \mathbf{A}_f} \right]. \tag{35b}$$

4.2. Momentum interpolation

Following Cubero and Fueyo [8], Cubero et al. [9], the velocities at the cell faces can be written as:

$$\mathbf{u}_{i,f} = (\mathbf{u}_i)_f + \langle \mathbf{u}_i \rangle, \tag{36}$$

where $(\mathbf{u}_i)_f$ is the linearly interpolated velocity at the face and $\langle \mathbf{u}_i \rangle$ is the velocity correction term. The correction term can be obtained by rewriting Eq. 36 as:

$$\langle \mathbf{u}_i \rangle = \mathbf{u}_{i,f} - (\mathbf{u}_i)_f. \tag{37}$$

From here, Eqs. 35 can be substituted into the above equation to give:

$$\begin{aligned}
\langle \mathbf{u}_p \rangle = & \frac{\tilde{\mathbf{u}}_{p,f}}{1 + \mathrm{d}_{Tp,f} + \mathrm{d}_{Dp,f}} - \left(\frac{\tilde{\mathbf{u}}_p}{1 + \mathrm{d}_{Tp} + \mathrm{d}_{Dp}} \right)_f \\
& + \frac{\mathrm{d}_{Tp,f} \mathbf{u}_{p,f}^{t-1}}{1 + \mathrm{d}_{Tp,f} + \mathrm{d}_{Dp,f}} - \left(\frac{\mathrm{d}_{Tp} \mathbf{u}_p^{t-1}}{1 + \mathrm{d}_{Tp} + \mathrm{d}_{Dp}} \right)_f \\
& + \frac{\mathrm{d}_{Dp,f} \mathbf{u}_{f,f}}{1 + \mathrm{d}_{Tp,f} + \mathrm{d}_{Dp,f}} - \left(\frac{\mathrm{d}_{Dp} \mathbf{u}_f}{1 + \mathrm{d}_{Tp} + \mathrm{d}_{Dp}} \right)_f \\
& - \frac{\nabla p_{f,f}}{[1 + \mathrm{d}_{Tp,f} + \mathrm{d}_{Dp,f}] \rho_p \mathbf{A}_{p,f}} + \left(\frac{\nabla p_f}{[1 + \mathrm{d}_{Tp} + \mathrm{d}_{Dp}] \rho_p \mathbf{A}_p} \right)_f,
\end{aligned} \tag{38}$$

$$
\begin{aligned}
\langle \mathbf{u}_f \rangle = & \frac{\tilde{\mathbf{u}}_{f,f}}{1 + \mathrm{d}_{Tf,f} + \mathrm{d}_{Df,f}} - \left(\frac{\tilde{\mathbf{u}}_f}{1 + \mathrm{d}_{Tf} + \mathrm{d}_{Df}} \right)_f \\
& + \frac{\mathrm{d}_{Tf,f} \mathbf{u}^{t-1}_{f,f}}{1 + \mathrm{d}_{Tf,f} + \mathrm{d}_{Df,f}} - \left(\frac{\mathrm{d}_{Tf} \mathbf{u}^{t-1}_{f}}{1 + \mathrm{d}_{Tf} + \mathrm{d}_{Df}} \right)_f \\
& + \frac{\mathrm{d}_{Df,f} \mathbf{u}_{p,f}}{1 + \mathrm{d}_{Tf,f} + \mathrm{d}_{Df,f}} - \left(\frac{\mathrm{d}_{Df} \mathbf{u}_{p}}{1 + \mathrm{d}_{Tf} + \mathrm{d}_{Df}} \right)_f \\
& - \frac{\nabla p_{f,f}}{[1 + \mathrm{d}_{Tf,f} + \mathrm{d}_{Df,f}] \rho_f \mathbf{A}_{f,f}} + \left(\frac{\nabla p_f}{[1 + \mathrm{d}_{Tf} + \mathrm{d}_{Df}] \rho_f \mathbf{A}_f} \right)_f ,
\end{aligned}
\tag{39}
$$

which leads to exact corrections of each face value. However, due to the linear interpolation of many of these variables their respective face values already contain their best approximation. As shown in Cubero et al. [9] approximating the pseudo-velocities through a linear interpolation reduces them to zero. The cell face values of the momentum-weighted coefficients and the numerical coefficients can be approximated as:

$$
\mathrm{d}_{Ti,f} = (\mathrm{d}_{Ti})_f; \qquad \mathrm{d}_{Di,f} = (\mathrm{d}_{Di})_f; \qquad \mathbf{A}_{i,f} = (\mathbf{A}_i)_f. \tag{40}
$$

The face pressure is calculated from the cell centre assuming central differencing, so that

$$
\nabla p_{f,f} = \nabla^{\perp}_f p_f. \tag{41}
$$

In Finite Volume CFD codes we can simplify the face interpolation by writing $(\mathcal{A}_i \phi_i)_f = \mathcal{A}_i (\phi_i)_f$ essentially taking the independent variables outside of the interpolation and leaving the dependant variable. This is utilised throughout each correction term.

Applying the above simplifications and invoking Eq. 38 & Eq. 39, which now reads as:

$$\langle \mathbf{u}_p \rangle = \langle \mathbf{u}_p \rangle_T + \langle \mathbf{u}_p \rangle_D + \langle \mathbf{u}_p \rangle_{\nabla p_f}, \tag{42a}$$

$$\langle \mathbf{u}_f \rangle = \langle \mathbf{u}_f \rangle_T + \langle \mathbf{u}_f \rangle_D + \langle \mathbf{u}_f \rangle_{\nabla p_f}, \tag{42b}$$

where the shared coefficients in each phase are $\langle \mathbf{u}_i \rangle_T$, temporal corrections are:

$$\langle \mathbf{u}_i \rangle_T = \frac{(\mathrm{d}_{Ti})_f [\mathbf{u}_{i,f}^{t-1} - (\mathbf{u}^{t-1})_f]}{1 + (\mathrm{d}_{Ti})_f + (\mathrm{d}_{Di})_f}, \tag{43}$$

and $\langle \mathbf{u}_i \rangle_D$, drag corrections are:

$$\langle \mathbf{u}_i \rangle_D = \frac{(\mathrm{d}_{Di})_f [\mathbf{u}_{j,f} - (\mathbf{u}_j)_f]}{1 + (\mathrm{d}_{Ti})_f + (\mathrm{d}_{Di})_f}, \tag{44}$$

and $\langle \mathbf{u}_f \rangle_{\nabla p_f}$, pressure correction are:

$$\langle \mathbf{u}_i \rangle_{\nabla p_f} = \frac{-\nabla_f^{\perp} p_f + (\nabla p_f)_f}{[1 + (\mathrm{d}_{Ti})_f + (\mathrm{d}_{Di})_f] \rho_i (\mathbf{A}_i)_f}, \tag{45}$$

4.3. Construction of implicit pressure equation

Analogous to the segregated implementation an equation for the mixture pressure can be found by inserting the phase-fluxes into the continuity equation (Eq. 19). Here we introduce the velocity-corrected flux to read:

$$\phi_{i,f} = [(\mathbf{u}_i)_f + \langle \mathbf{u}_i \rangle] \cdot \mathbf{S}_f, \tag{46}$$

then inserting the relation into the continuity equation (Eq. 19) reading:

$$\nabla \cdot \Big((\alpha_p)_f [(\mathbf{u}_p)_f + \langle \mathbf{u}_p \rangle] \cdot \mathbf{S}_f + (\alpha_f)_f [(\mathbf{u}_f)_f + \langle \mathbf{u}_f \rangle] \cdot \mathbf{S}_f \Big), \tag{47}$$

then inserting the corrections velocities in Eqs. 42 results in the full pressure equation:

$$\nabla \cdot [D_p \nabla_f^{\perp} p_f |\mathbf{S}_f|] = \nabla \cdot \left(\sum_{k=1}^{P} (\alpha_k)_f (\mathbf{u}_k)_f \cdot \mathbf{S}_f \right) + \nabla \cdot [D_p (\nabla p_f)_f \cdot \mathbf{S}_f]$$
$$+\nabla \cdot \left[\sum_{k=1}^{P} (\alpha_k)_f \left(\frac{(\mathrm{d}_{Tk})_f [\phi_k^{t-1} - (\mathbf{u}_k^{t-1})_f \cdot \mathbf{S}_f]}{1 + (\mathrm{d}_{Tk})_f + (\mathrm{d}_{Dk})_f} + \frac{\sum_{m=1}^{P} (\mathrm{d}_{mi})_f [\phi_m - (\mathbf{u}_m)_f \cdot \mathbf{S}_f]}{1 + (\mathrm{d}_{Tm})_f + (\mathrm{d}_{Dm})_f} \right) \right], \tag{48}$$

where the pressure diffusivity coefficient reads as

$$D_p = \sum_{k=1}^{P} \frac{(\alpha_k)_f}{[1 + (\mathrm{d}_{Tk})_f + (\mathrm{d}_{Dk})_f] \rho_k (\mathbf{A}_k)_f}. \tag{49}$$

4.4. Discretised momentum equations

Here we present the phase-momentum equations in their implemented form as they will be referenced later when discussing the block-coefficients.

$$\begin{aligned} &\left[\!\left[\frac{\partial [\![\mathbf{u}_p]\!]}{\partial t} \right]\!\right] + \left[\!\left[\nabla \cdot (\mathbf{u}_p [\![\mathbf{u}_p]\!]) \right]\!\right] - \left[\!\left[(\nabla \cdot \mathbf{u}_p) [\![\mathbf{u}_p]\!] \right]\!\right] - \left[\!\left[\nabla \cdot (\nu_{\mathrm{eff,p}} \nabla [\![\mathbf{u}_p]\!]) \right]\!\right] \\ &- \left[\!\left[\nabla \cdot (\nu_{\mathrm{eff,p}} \frac{\nabla \alpha_p}{\alpha_p + \delta} [\![\mathbf{u}_p]\!]) \right]\!\right] - \left[\!\left[\nabla \cdot (\nu_{\mathrm{eff,p}} \frac{\nabla \alpha_p}{\alpha_p^*}) [\![\mathbf{u}_p]\!]) \right]\!\right] \\ &+ \left[\!\left[\frac{\beta [\![\mathbf{u}_p]\!]}{\alpha_p \rho_p} \right]\!\right] - \left[\!\left[\frac{\beta [\![\mathbf{u}_f]\!]}{\alpha_p \rho_p} \right]\!\right] + \left[\!\left[\frac{1}{\rho_p} [\![\nabla p_f]\!] \right]\!\right] \\ &= -\frac{\nabla \alpha_p}{\alpha_p + \delta} \cdot \overline{\mathbf{R}}^c_{\mathrm{eff,p}} - \nabla \cdot \overline{\mathbf{R}}^c_{\mathrm{eff,p}} - \frac{\beta \mathscr{D} \nabla \alpha_p}{\alpha_p \rho_p} - \frac{\nabla p_p}{\alpha_p^* \rho_p} + \mathbf{g} \end{aligned} \tag{50}$$

$$\begin{aligned} &\left[\!\left[\frac{\partial [\![\mathbf{u}_f]\!]}{\partial t} \right]\!\right] + \left[\!\left[\nabla \cdot (\mathbf{u}_f [\![\mathbf{u}_f]\!]) \right]\!\right] - \left[\!\left[(\nabla \cdot \mathbf{u}_f) [\![\mathbf{u}_f]\!] \right]\!\right] - \left[\!\left[\nabla \cdot (\nu_{\mathrm{eff,f}} \nabla [\![\mathbf{u}_f]\!]) \right]\!\right] \\ &- \left[\!\left[\nabla \cdot (\nu_{\mathrm{eff,f}} \frac{\nabla \alpha_f}{\alpha_f + \delta} [\![\mathbf{u}_f]\!]) \right]\!\right] - \left[\!\left[\nabla \cdot (\nu_{\mathrm{eff,f}} \frac{\nabla \alpha_f}{\alpha_f^*}) [\![\mathbf{u}_f]\!]) \right]\!\right] \\ &+ \left[\!\left[\frac{\beta [\![\mathbf{u}_f]\!]}{\alpha_f \rho_f} \right]\!\right] - \left[\!\left[\frac{\beta [\![\mathbf{u}_p]\!]}{\alpha_f \rho_f} \right]\!\right] + \left[\!\left[\frac{1}{\rho_f} [\![\nabla p_f]\!] \right]\!\right] \\ &= -\frac{\nabla \alpha_f}{\alpha_f + \delta} \cdot \overline{\mathbf{R}}^c_{\mathrm{eff,f}} - \nabla \cdot \overline{\mathbf{R}}^c_{\mathrm{eff,f}} + \frac{\beta \mathscr{D} \nabla \alpha_p}{\alpha_p \rho_p} + \mathbf{g} \end{aligned} \tag{51}$$

4.5. Discretised pressure equation

The final implemented pressure equation is presented below. The approach outlined in Cubero and Fueyo [8], Darwish et al. [11], Ferreira et al. [13] is followed to arrive at a pressure equation for the implicit solution of the phase-velocity-pressure coupling. The implicit divergence of the phase-velocities are corrected with the addition of the opposing drag contribution, as shown in Ferreira et al. [13]. Additionally, the whole equation is multiplied by -1 to enhance positivity of the block-coefficient matrix. The implemented pressure equation thus reads:

$$
\begin{aligned}
-\nabla \cdot [\![D_p \nabla_f^{\perp} p_f |\mathbf{S}_f|]\!] + \nabla \cdot \left[\!\!\left[(\alpha_k)_f - \sum_{m=1}^{P} \frac{(\alpha_m)_f (\mathrm{d}_{Dm})_f}{1 + (\mathrm{d}_{Tm})_f + (\mathrm{d}_{Dm})_f} ([\![\mathbf{u}_k]\!])_f \cdot \mathbf{S}_f \right]\!\!\right] \\
= \nabla \cdot [D_p (\nabla p_f)_f \cdot \mathbf{S}_f] \\
+ \nabla \cdot \left[\sum_{k=1}^{P} (\alpha_k)_f \left(\frac{(\mathrm{d}_{Tk})_f [\phi_k^{t-1} - (\mathbf{u}_k^{t-1})_f \cdot \mathbf{S}_f]}{1 + (\mathrm{d}_{Tk})_f + (\mathrm{d}_{Dk})_f} + \frac{\sum_{m=1}^{P} (\mathrm{d}_{Dm})_f \phi_m}{1 + (\mathrm{d}_{Tk})_f + (\mathrm{d}_{Dk})_f} \right) \right].
\end{aligned}
\tag{52}
$$

4.6. Correction of the phase fluxes

From the solution of the block-coupled matrix, we find new values for the phase-velocity and pressure. Then the face fluxes need to be updated by including the corrections to the phase-velocity that were added to the pressure equation.

$$
\begin{aligned}
\phi_p = (\mathbf{u}_p)_f \cdot \mathbf{S}_f + \frac{(\mathrm{d}_{Tp})_f [\phi_p^{t-1} - (\mathbf{u}_p^{t-1})_f \cdot \mathbf{S}_f]}{1 + (\mathrm{d}_{Tp})_f + (\mathrm{d}_{Dp})_f} + \frac{(\mathrm{d}_{Dp})_f [\phi_f^{n-1} - (\mathbf{u}_f^{n})_f \cdot \mathbf{S}_f]}{1 + (\mathrm{d}_{Tp})_f + (\mathrm{d}_{Dp})_f} \\
+ \frac{[-\nabla_f^{\perp} p_f^n |\mathbf{S}_f| + (\nabla p_f^{n-1})_f \cdot \mathbf{S}_f]}{[1 + (\mathrm{d}_{Tp})_f + (\mathrm{d}_{Dp})_f] \rho_p (\mathbf{A}_p)_f},
\end{aligned}
\tag{53}
$$

$$
\begin{aligned}
\phi_f = (\mathbf{u}_f)_f \cdot \mathbf{S}_f + \frac{(\mathrm{d}_{Tf})_f [\phi_f^{t-1} - (\mathbf{u}_f^{t-1})_f \cdot \mathbf{S}_f]}{1 + (\mathrm{d}_{Tf})_f + (\mathrm{d}_{Df})_f} + \frac{(\mathrm{d}_{Df})_f [\phi_p^{n-1} - (\mathbf{u}_p^{n})_f \cdot \mathbf{S}_f]}{1 + (\mathrm{d}_{Tf})_f + (\mathrm{d}_{Df})_f} \\
+ \frac{[-\nabla_f^{\perp} p_f^n |\mathbf{S}_f| + (\nabla p_f^{n-1})_f \cdot \mathbf{S}_f]}{[1 + (\mathrm{d}_{Tf})_f + (\mathrm{d}_{Df})_f] \rho_f (\mathbf{A}_f)_f},
\end{aligned}
\tag{54}
$$

where the superscripts n and $n-1$ denote the value from the present iteration and the previous iteration, respectively. The outline of the solution procedure can be found below.

The numerical procedure adopted in the coupled algorithm:

1. Solve the volume fraction (Eq. 27).
2. Construct $\mathbf{A}_i$ and $\mathbf{H}_i$ in each phase.
3. Update the temporal and drag coefficients in Eqs. 30.
4. Update the momentum-weighted coefficients in Eq. 33 & 34.
5. Update the correction velocities in each phase using Eq. 42.
6. Assemble and solve the 7x7 block-coupled matrix.
   (a) Feed in the phase momentum equations.
   (b) Feed in the pressure equation.
   (c) Remove cross-coupling source and place in implicit off-diagonal.
7. Apply the flux update using Eqs. 53 & 54.
8. Solve the system of phase energy equations
9. Advance in time.

4.7. *Implicitly coupled phase-velocity-pressure, $\mathbf{u}_i - p_f$ system*

The system of linear algebraic equations discretised in a Finite Volume framework can be written as:

$$A_{i,j} x_i = b_i, \tag{55}$$

where $A_{i,j}$ is the matrix representing diagonal and off-diagonal coefficients. x_i is the solution variable and finally b_i is the source vector. This discretisation, within a block-coupled solution, can be easier expressed in two steps [34]: the first level, represents the spatial coupling across the computational domain (Eq. 56) and the second level, which represents the inter-equation coupling i.e. phase-velocity-pressure coupling. Expressing a system with N unknowns, in which N is denoted as the number of cells, Eq. 55 can be written as:

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,N} \\ a_{2,1} & a_{2,2} & \dots & a_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N,1} & a_{N,2} & \dots & a_{N,N} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}, \tag{56}$$

where $a_{i,j}$ represents a block-coefficient and is spatially coupled between cells i and j, the solution vector, x_i contains the unknowns at cell i and finally, b_i is the source vector for cell i. As mentioned previously, a second level of discretisation is present within a block-coupled matrix.

The solution vector, at cell i now reads:

$$x_i = \begin{bmatrix} u_{p,xi} \\ u_{p,yi} \\ u_{p,zi} \\ u_{f,xi} \\ u_{f,yi} \\ u_{f,zi} \\ p_{f,i} \end{bmatrix}, \qquad b_i = \begin{bmatrix} b_{u_{p,xi}} \\ b_{u_{p,yi}} \\ b_{u_{p,zi}} \\ b_{u_{f,xi}} \\ b_{u_{f,yi}} \\ b_{u_{f,zi}} \\ b_{p_{f,i}} \end{bmatrix}, \tag{57}$$

where $u_{k,xi}$ represents the x component of the respective phase velocity in cell i, similarly, $u_{k,yi}$ and $u_{k,zi}$ represents the y and z components of the respective phase velocity and finally $p_{f,i}$ represents the pressure field in cell i. In the source vector, the source term for each variable is found following the same notation as the solution vector.

Each block-matrix coefficient, $a_{i,j}$ is a sub-matrix representing the coupling between the phase-velocity-pressure components which comprises of a 7x7 block matrix which can be written as follows:

$$a_{i,j} = \begin{bmatrix} a_{u_{p,xi},u_{p,xj}} & 0 & 0 & a_{u_{p,xi},u_{f,xj}} & 0 & 0 & a_{u_{p,xi},p_{f,j}} \\ 0 & a_{u_{p,yi},u_{p,yj}} & 0 & 0 & a_{u_{p,yi},u_{f,yj}} & 0 & a_{u_{p,yi},p_{f,j}} \\ 0 & 0 & a_{u_{p,zi},u_{p,zj}} & 0 & 0 & a_{u_{p,zi},u_{f,zj}} & a_{u_{p,zi},p_{f,j}} \\ a_{u_{f,xi},u_{p,xj}} & 0 & 0 & a_{u_{f,xi},u_{f,xj}} & 0 & 0 & a_{u_{f,xi},p_{f,j}} \\ 0 & a_{u_{f,yi},u_{p,yj}} & 0 & 0 & a_{u_{f,yi},u_{f,yj}} & 0 & a_{u_{f,yi},p_{f,j}} \\ 0 & 0 & a_{u_{f,zi},u_{p,zj}} & 0 & 0 & a_{u_{f,zi},u_{f,zj}} & a_{u_{f,zi},p_{f,zj}} \\ a_{p_{f,i},u_{p,xj}} & a_{p_{f,i},u_{p,yj}} & a_{p_{f,i},u_{p,zj}} & a_{p_{f,i},u_{f,xj}} & a_{p_{f,i},u_{f,yj}} & a_{p_{f,i},u_{f,zj}} & a_{p_{f,i},p_{f,j}} \end{bmatrix}. \tag{58}$$

Given the structure of the block-coefficient, a description of the coefficients and their physical meaning is required. We will now focus our attention on four sections of the block-coefficient and for brevity concern ourselves with the particle-phase coefficients.

$$\begin{bmatrix} a_{u_{p,xi},u_{p,xj}} & 0 & 0 \\ 0 & a_{u_{p,yi},u_{p,yj}} & 0 \\ 0 & 0 & a_{u_{p,zi},u_{p,zj}} \end{bmatrix}; \begin{bmatrix} a_{u_{p,xi},u_{f,xj}} & 0 & 0 \\ 0 & a_{u_{p,yi},u_{f,yj}} & 0 \\ 0 & 0 & a_{u_{p,zi},u_{f,zj}} \end{bmatrix}. \quad (59)$$

The first 3x3 block represents the coupling between the velocity components. These coefficients are filled by the implicitly discretised directional momentum equations pertaining to the time derivative, convection, diffusion and drag in Eqs. 50 & 51. The explicitly discretised terms are moved to the source vector, Eq. 57 i.e. the terms found on the RHS of Eqs. 50 & 51. The second 3x3 block introduces the cross-coupling coefficients. These represent the implicit treatment of drag which includes the phase-velocity of the opposing phase and is the eighth term on the LHS of Eqs. 50 & 51.

$$\begin{bmatrix} a_{u_{p,xi},p_{f,j}} \\ a_{u_{p,yi},p_{f,j}} \\ a_{u_{p,zi},p_{f,j}} \end{bmatrix}; \quad \begin{bmatrix} a_{p_{f,i},u_{p,xj}} & a_{p_{f,i},u_{p,yj}} & a_{p_{f,i},u_{p,zj}} \end{bmatrix}. \quad (60)$$

The first 1x3 block in Eq. 60 denotes the phase-velocity-pressure coupling in the momentum equation. This pertains to the implicit treatment of the pressure gradient, the ninth term in Eqs. 50 & 51. The second block, 3x1 denotes the phase-velocity-pressure coupling in the pressure equation. This pertains to the implicit

treatment of the terms inside in the divergence operator in Eq. 52 i.e. the second term, with explicit boundary contributions populating the source vector, $b_{p_{f,i}}$.

The implicitly treated pressure on the LHS of Eq. 52 is fed into the coefficient $a_{p_{f,i},p_{f,j}}$ with the explicit boundary contributions being fed into the corresponding source vector $b_{p_{f,i}}$. Finally, the explicit terms on the RHS of Eq. 52 are fed into the source vector, $b_{p_{f,i}}$.

Within the community driven branch of OpenFOAM called `foam-extend` several numerical tools have been developed to house coupled solvers [4, 7, 18]. The block-matrix machinery is extended in order to construct a 7x7 block matrix and ensure the correct populating of matrix coefficients. The matrix solvers are then used to solve the phase-velocity-pressure coupled system.

4.8. Simulation set-up and geometry

The geometry used in the experiment of Tsuji et al. [33] comprises of a vertically facing pipe with a diameter (D) of 0.035m and can be seen schematically in Fig. 1. The length of the pipe (L) including the development section is, L = 5.2m. The mesh size is 50 cells in the x direction and 20 in the y direction with adequate spacing to ensure a $y^+ > 30$ criterion can be prescribed for the wall function. Due to the computational power available, and the amount of coefficients that need to be stored for each cell in the coupled solver, the mesh size had to be limited.

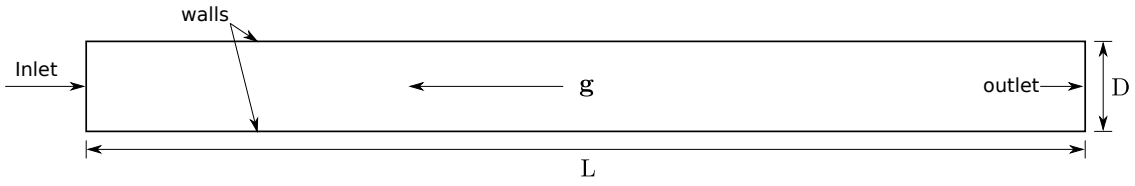

Figure 1: Schematic of the computational domain.

At the inlet a Dirichlet boundary condition is prescribed for both phase velocities and a Neumann condition for pressure. At the outlet a Dirichlet boundary condition is prescribed for pressure and a Neumann condition for both phase velocities. For the particulate phase wall boundary conditions a Neumann boundary condition is prescribed for the velocity and turbulence statistics. For the fluid-phase, the no slip wall condition is prescribed for velocity and the standard wall functions are employed for the turbulence statistics. Both k_p and ε_p are initialised as 1/3rd of their fluid counterpart with $\Theta_p = 1.0$ x $10^{-8}\mathrm{m}^2\mathrm{s}^{-2}$.

Table 2 details the cases simulated in this work. For the majority of the cases the centreline velocities were not recorded therefore the bulk velocities have been used. The mean velocity ($\mathrm{U}_i^+ = \mathrm{u}_x/\mathrm{u}_m$) and turbulence intensity ($\mathrm{u}_i^+ = (0.5k_i)^{1/2}/\mathrm{u}_m$) are normalised by the bulk velocity, u_m which is taken from the simulation due to the lack of recorded values in the experiment.

Table 2: Table of simulated cases

Case	Mass loading	d_p [μm]	Density [$kg\mathrm{m}^{-3}$]	U_m [m/s]
1	1	0.2	1020	15.6
2	2.1	-	-	15.3
3	1.3	0.5	-	10.8
4	2.9	-	-	10.8

Both the coupled and segregated solvers solve the phase-energy system of equations in a sequential manner using generic relaxation factors of 0.7 and a PGiCG solver. The coupled solver employs the ILU preconditioner and the biconjugate gradient stabilised solver (BiCGSTAB) with no relaxation factors. For the segregated system of equations the pressure equation was solved using the generalised alge-

braic multi-grid (GAMG) with a relaxation factor of 0.3. The volume fraction is solved using Multi-dimensional Universal Limiter with Explicit Solution (MULES) [40] which is a flux-corrected transport algorithm which ensures robustness, stability and convergence. Time derivative terms are discretised using the first order accurate implcit Euler scheme, gradients are discretised using the Gauss linear scheme, convective terms are discretised using the first order upwind scheme. Finally, Laplacians are discretised with the second order accurate central differencing scheme. All simulations were run on a Dell XPS 13 - Intel Core i7 with 8GB of RAM.

5. Results and discussion

5.1. Verification of the coupled solver

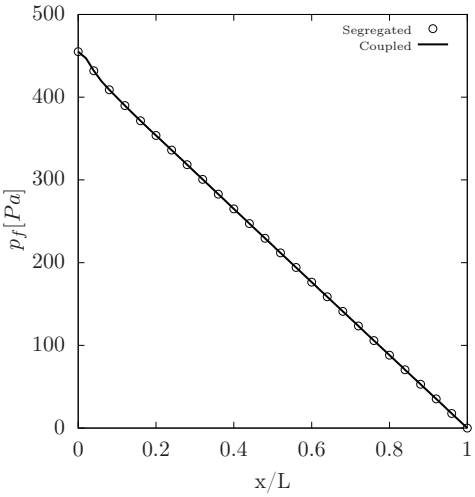

Figure 2: Distribution of pressure across the horizontal midsection of the pipe.

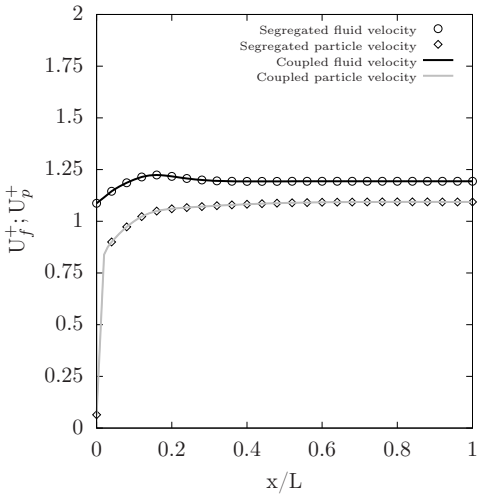

Figure 3: Distribution of the phase-velocities across the horizontal midsection of the pipe.

Fig. 2 shows a comparison of the pressure drop across the pipe in both the coupled and segregated solver. Both solvers show identical behaviour with a linear drop across the length of the pipe. From the authors experience, this was greatly influenced by the momentum interpolation technique of Cubero et al. [9] and implicit treatment of the drag correction in the divergence operator (Eq. 52). In particular the behaviour of the pressure drop in cells close to the inlet proved particularly challenging and could not be realised without the the CMI of Cubero and Fueyo [8].

.

Fig. 3 shows a comparison of the phase-velocities across the pipe in both the coupled and segregated solver. To highlight the influence of the inter-phase momentum

transfer the inlet velocity for the particle phase is a $\approx 10\%$ of the fluid phase. Again identical behaviour between solvers is demonstrated.

5.2. Validation of the coupled solver

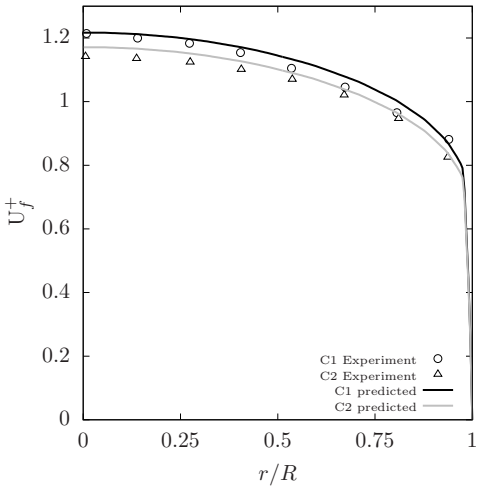

Figure 4: Mean fluid velocity. Symbols Tsuji et al. [33]; curves are predictions for Case 1 & 2.

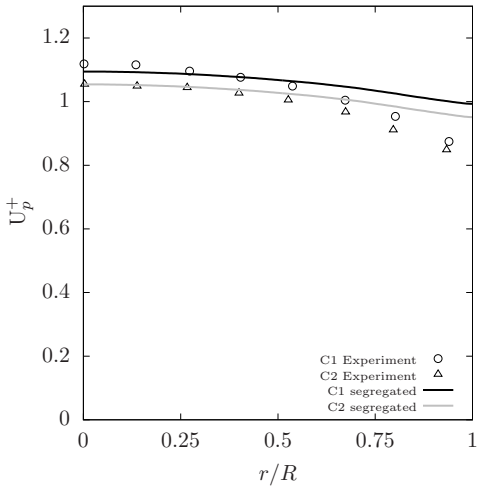

Figure 5: Mean particle velocity. Symbols Tsuji et al. [33]; curves are predictions for Case 1 & 2.

Fig. 4 shows the mean fluid velocity profiles. Overall, the trend of the fluid behaviour is captured, with the increase in mass loading resulting in a global reduction of fluid velocity (due to the direction of the body force) in both experimental and numerical predictions. In Case 1, there is an almost global over-prediction of the mean velocity albeit small. In the near-wall region ($r/R > 0.75$) the momentum loss is difficult to capture correctly. The experimental results suggest that the numerical model is not producing enough mean shear. This would result in a higher rate of change in the near-wall region thus falling in line with the experimental data. This lack of momentum loss can also be affected by the co-variance coupling term. As the

particles are tightly coupled with the fluid phase the main mechanism for momentum transfer is drag. An under-prediction in the co-variance term will reduce momentum loss - which would result an over-prediction of mean velocity.

For Case 2 this over-prediction is exacerbated and with an increased mass loading, in particular across ($r/R < 0.5$). In the region ($r/R > 0.75$) a substantial relative velocity between Case 1 and Case 2 was observed in the experimental data. Throughout the simulations this behaviour was qualitatively predicted showing reasonable agreement. It should be noted that the instrument used to measure the flow statistics, namely the laser Doppler Velocimeter (LDV) can be an intrusive way of measuring velocity and turbulence statistics. Additionally, in the near-wall region it becomes particularly challenging to record reliably.

Fig. 5 shows the mean particle velocity predictions for Case 1 & 2. In the near wall region the slip condition enables a relative velocity between both phases to develop. Experimentally this resulted in a negative, $\mathbf{u}_r = \mathbf{u}_f - \mathbf{u}_p$ in the region ($r/R > 0.75$) and a positive $\mathbf{u}_r$ in the ($r/R < 0.75$) region. The slip boundary condition exhorts its influence over a quarter of the pipe - a finding that is consistent with the numerical prediction. The main discrepancy between the experimental and numerical results is across the near-wall region. The experimental results indicate that the particles remain largely correlated with the fluid-phases boundary layer. This is expected due to their tight coupling through drag and can be partly predicted by the model as the influence of the fluid phase is felt across the particle velocities across ($r/R > 0.75$).

Two explanations for this lack of momentum loss can be offered. Firstly, this behaviour indicates that the turbophoresis force that is responsible for wall-normal

migration of particles is being under predicted. Without the redistribution of particles across the width of the pipe a more uniform velocity distribution is seen [30]. Secondly, the wall boundary condition was taking as slip assuming smooth walls. This is a speculative assumption and with the inclusion of boundary conditions that incorporate the effect of wall roughness [29] the momentum loss in the boundary layer would be enhanced resulting in a closer prediction.

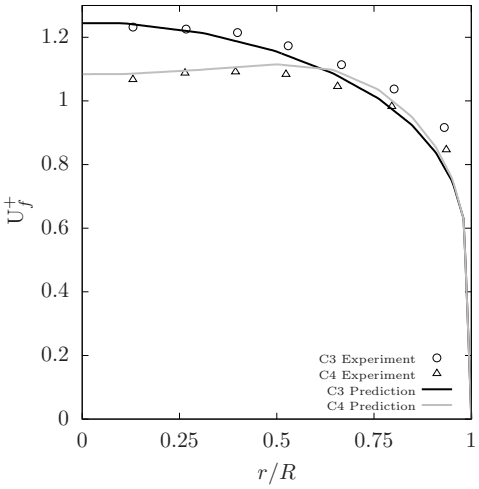

Figure 6: Mean fluid velocity. Symbols Tsuji et al. [33]; curves are predictions for Case 3 & 4.

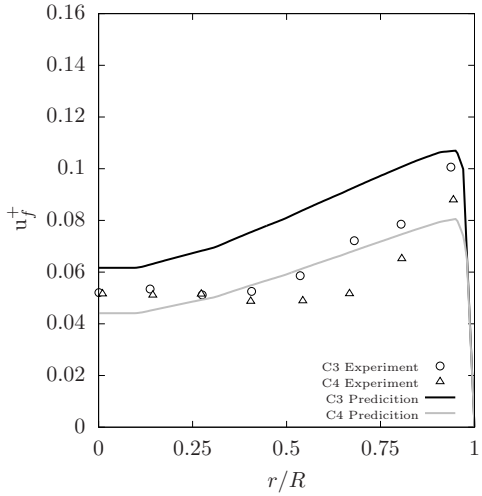

Figure 7: Mean fluid turbulent kinetic energy. Symbols Tsuji et al. [33]; curves are predictions for Case 3 & 4.

Figs 6 & 7 show the results from Cases 3 & 4. In the former, the predicted mean fluid-velocities are in relatively good agreement with the experimental data with the main discrepancies being seen in the near-wall region. With increased mass loading the difficult to capture [22] reduction of fluid velocities in the core region ($r/R < 0.5$) is reproduced. The increase in particle diameter and mass loading results in an accumulation of particles within the core of the pipe which are being dragged down

by gravity. Due to the increase in Stokes number and increased likelihood of particle collisions - the uncorrelated energy, Θ_p experiences an increase in the core of the flow. This ensures that the particles are no longer closely correlated with the carrier flow, i.e. increased dissipation in the correlated energy equations $k_p - \varepsilon_p$. Through the co-variance coupling terms (see Table 5), as well as the inter-phase momentum transfer term, this behaviour can be captured. This results in the fluid phase velocity being "dragged" by the particle phase - a complex two-way coupled mechanism that is apparent in the numerical prediction and in the experimental observation. Due to the conservation of momentum across the pipe this results in an increase in the velocities in the $(0.5 > r/R > 0.75)$.

For Case 3, a good agreement is found with the centreline velocity but the main bulk of the velocities leading up to the near-wall region are under-predicted. This behaviour can be better explained by looking at Fig. 7. The fluid intensity result for Case 3 illuminates the situation. The over-prediction of the intensity across the centreline would manifest itself in an over-prediction in the turbulent viscosity calculation resulting in the predicted behaviour. Due to the non-linear profile of the experimental turbulence intensity the behaviour is difficult to capture within a Reynolds-Averaged methodology, in particular the use of the wall function also limits the situation further. To this end a near-wall pressure-velocity model has been recently proposed that can circumvent these problems in two-fluid simulations [30, 31, 28].

5.3. Performance of the coupled solver vs segregated solver

In this section both the coupled and segregated solvers are run for 30 seconds of actual flow time on Case 1 under identical conditions with the CFL number kept constant at 0.5.

In order to ascertain the magnitude of the estimated error, the normalised residual error estimate is calculated according to Jasak [19]. The residual is normalised by the dominant diagonal coefficients in order to ascertain the behaviour of each variable more readily. This enables the formulation of a relative error.

$$\varepsilon_r(\phi) = \frac{|b_i - A_{i,j}x_i|}{|A^n_{i,j}x^k_i - A^{n-1}_{i,j}x^{n-1}_i| + |b^k_i - A^{n-1}_{i,j}x^{n-1}_i|}. \tag{61}$$

A convergence criterion can be set as:

$$\varepsilon_r(\phi) \leqslant \varepsilon_{res}. \tag{62}$$

Although we do not set a stop criterion in this study it should be noted that conventionally residuals are set between $\varepsilon_r < 10^{-3} - 10^{-6}$. If we take the latter value as our convergence criterion the two-fluid coupled solver converges in 161s whereas the segregated solver fails to reach values near $\varepsilon_r(\phi) = 10^{-6}$ and oscillate in the order of $\varepsilon_r(\phi) = 10^{-3} - 10^{-4}$.

Throughout we have only shown the residual behaviour for Case 1, although from the author's experience, this was representative of the typical behaviour seen across all four cases. As there are relatively small increases in mass loading the overall residual behaviour remained similar.

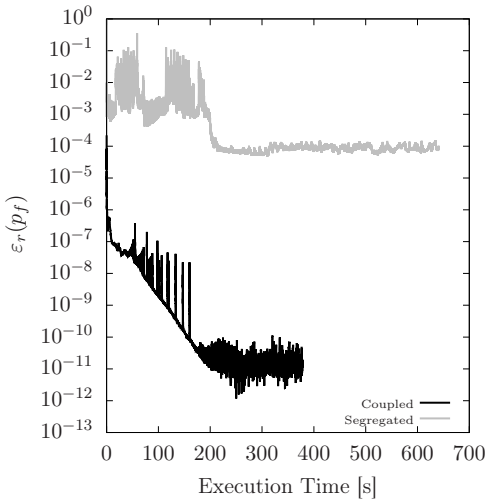

Figure 8: Pressure residual behaviour for coupled and segregated solver.

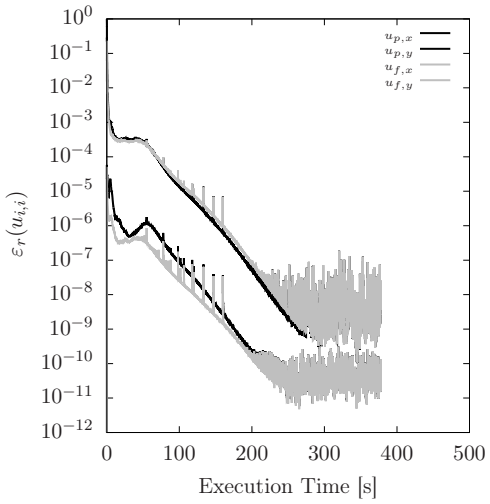

Figure 9: Velocity component behaviour for the coupled solver.

Figs. 8 and 9 show the normalised residual behaviour for pressure and phase velocity components. Due to the segregated solution algorithm used the phase velocity components are not explicitly solved for and are instead used to predict and correct, hence no data is available for a comparison. Fig. 8 reveals some quite striking behaviour about the residual behaviour. The coupled two-fluid solver's initial residual, due to the implicit treatment of the pressure correction, starts at the $\varepsilon_r(p_f) = \mathcal{O}(10^{-4})$ - as the flow is driven by inlet condition for velocity, the pressure coefficients do not contain a substantial source. This residual error is driven down by several orders of magnitude within the first few iterations before reaching an oscillatory steady state at $\varepsilon_r(p_f) = \mathcal{O}(10^{-11})$.

In the segregated solver typical residual behaviour is observed, showing saw-toothed behaviour, due to the relaxation factor. After some time, similar to the coupled solver, the solution reaches a steady-state with the residual remaining oscil-

latory until the simulation ends. The main contributor to the extension in time is the explicit calculation of the pressure equation. In the segregated solver crucial terms, drag and gravity, are moved to the pressure calculation - this increases the stability of the solution but puts a penalty on the computational time. This often results in a hefty amount of iterations to drive the pressure residual down to its prescribed tolerance before advancing the solution in time.

Figure 9 shows the four phase velocity components. It can be seen how the normalised residual behaviour follows the same qualitative behaviour of the pressure residual - a natural consequence of the block-coupled solution. Throughout the solution small spikes and oscillatory behaviour is experienced a feature that was also present in Uroić and Jasak [34] and was shown to be an artifact of the linear solver BiCGSTAB. The two largest residuals are the momentum variables in the flow direction, this is expected due to their diagonal coefficients containing the dominant momentum flux and implicit drag correction. It is evident that the implicit treatment of the phase-velocity-pressure has positive benefits on the normalised residual error showing substantial improvements over the explicit treatment.

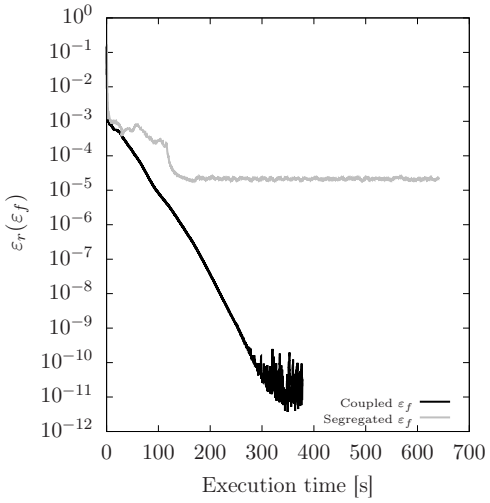

Figure 10: Fluid turbulent dissipation residual behaviour.

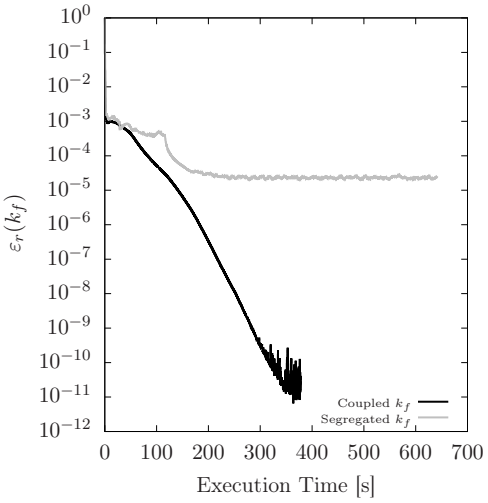

Figure 11: Fluid turbulent kinetic energy residual behaviour.

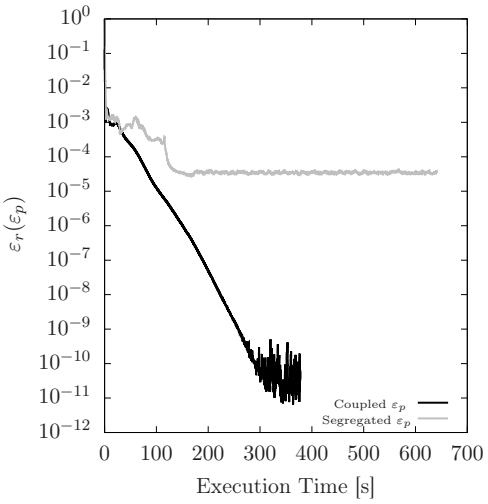

Figure 12: Particle turbulent kinetic energy execution time.

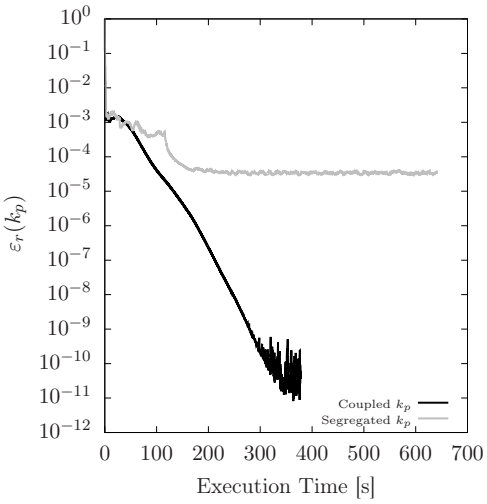

Figure 13: Particle turbulent kinetic energy dissipation convergence.

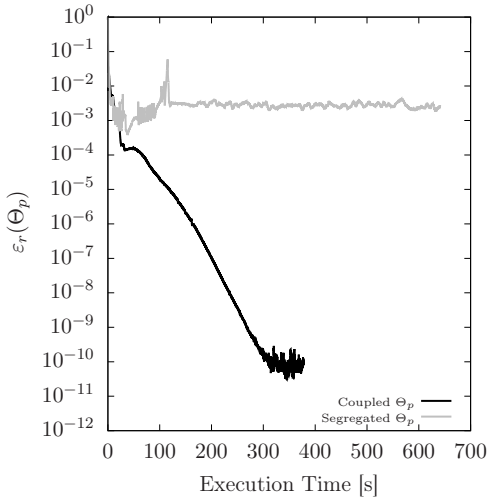

Figure 14: Granular temperature convergence.

Figures 10 - 14 show the residual behaviour for the phase-energy system. Overall, it can be seen that the coupled solver reduces the residual error across all turbulence variables resulting in a comparative drop of several orders of magnitude. The benefits of the implicit treatment of the phase-velocity-pressure coupling is carried over into

the solution of phase-energies despite them being solved using a segregated solution algorithm.

The segregated solution, on the other hand, displays similar residual error behaviour across each turbulence variable. After an initial reduction the error tends to stall and oscillate around an unsatisfactory value, behaviour that is similar to that seen in the previous section. This is symptomatic of the segregated solution algorithm and further (minor) improvements in the residual error would require arbitrary tweaking of relaxation factors.

In the Two-Fluid model employed in this work the phase-energy equations are coupled through inter-phase momentum transfer and the term is treated semi-implicitly. These system of equations, particle- and fluid-phase energy, are also suitable candidates for a block coupled solution as they can be coupled through: turbulent kinetic energy production, dissipation and inter-phase drag. This could provide further enhancements in solution time and residual error. Moreover, this would enhance the coupling within the energy system resulting in a more robust and stable solution algorithm. In particular in flow regimes with large drag values e.g. small particle diameters.

Finally, this methodology can be readily extended to more coupling mechanisms e.g. buoyancy, lift or virtual mass, and more sophisticated turbulence modelling e.g. LES, and more complicated physical process e.g. chemical reactions or heat transfer. The inclusion of which would certainly enhance the performance of the solution algorithm.

5.4. CFL number variation

One additional benefit of an implicitly coupled phase-velocity-pressure solution is that the solution can be accelerated due to the implicit treatment of hitherto explicit terms, unlike in the segregated solver. The implicit treatment of the phase-velocity-pressure coupling and the inter-phase momentum transfer in particular enables the CFL number to be increased beyond conventional limits. In this section the simulations are rerun with incrementally increasing CFL number to ascertain the performance of both solvers.

Courant No.	Coupled Exe. [s]	Segregated Exe. [s]
0.25	541	1022
0.5	377	641
1	235	320
1.5	216	255
2	176	N/A
2.5	149	N/A

Table 3: Total execution time of the coupled and segregated solvers under increasing CFL Number.

Table 3 details the solution execution time of each solver under increasing CFL number. Overall, it can be seen that the coupled solver out performs the segregated solver across each increment of CFL number. In addition, the coupled solver is able to achieve higher CFL numbers due to its implicit solution. This results in the coupled solver being 1.7 times quicker than than the segregated solver. Above CFL numbers of 1.5 the segregated solution becomes unstable and the solution is compromised. This is due to the explicit treatment of the phase-velocity-pressure coupling and the semi-implicit implementation of the inter-phase momentum transfer.

For the solution of the block-matrix a fairly conventional matrix solver is employed i.e. ILU preconditioner with BiCGSTAB. Recently, a more sophisticated approach has been developed: a block-selective algebraic multigrid algorithm [34]. We note here that an aggregative algebraic multigrid algorithm exists within `foam-extend` but its performance was found to be unsatisfactory in comparison to BiCGSTAB. The block-selective algorithm has shown to provide substantial increases in the performance of the linear solver. In some cases completing the solution within half the time of the BiCGSTAB algorithm. This could further improve the results of the coupled solver with a further reduction in execution time.

6. Conclusions

In this work a fully-coupled pressure-based two-fluid framework for the solution of turbulent fluid-particle flows is presented. The numerical framework detailed several crucial aspects: implicit treatment of the phase-velocity-pressure coupling, the implicit treatment of inter-phase momentum transfer and finally the solution algorithm. The approach is directly contrasted with the segregated approach in order to compare key differences in the solution algorithm. The coupled two-fluid solver is verified and validated against the segregated solver and benchmark experimental data respectively, showing good agreement throughout. The performance of both the coupled and segregated solvers are also evaluated.

The papers main contributions can be summarised as follows:

- A fully-coupled pressure-based two-fluid solver for fluid-particle flow is derived and implemented within `foam-extend`.
- The solver is validated against benchmark experimental data showing good agreement throughout.
- The coupled solver, in general, provides superior performance:
  - Solving to a tolerance that is six orders of magnitude smaller in residual error.
  - Completing the simulation 1.7 times quicker than the segregated solver.
  - Able to increase the CFL number to 2.5 further accelerating the simulation as opposed to 1.5 in the segregated solver.

- As an auxiliary benefit to the implicit treatment of the phase-velocity-pressure coupling the system of phase-energy equations, of which are solved sequentially, are solved to a tolerance that is seven times smaller in magnitude.

Nomenclature

$(\cdot)_f$	cell to face interpolation
$\mathbf{A}_i$	main diagonal of coefficients obtained from the discretisation procedure, [s^{-1}]
C_D	drag coefficient, [−]
CFL	Courant-Friedrichs-Lewy number
D	pipe diameter, [m]
D_p	pressure diffusivity matrix, [$kg^{-1}sm^3$]
d_p	particle diameter, [m]
$d_{,i}$	numerical coefficient ratio
g_0	radial distribution coefficient, [−]
$\mathbf{g}$	gravity, [ms^{-2}]
$\mathbf{H}_i$	off-diagonal of coefficients obtained from the discretisation procedure, [ms^{-2}]
k_i	turbulent kinetic energy, [m^2s^{-2}]
L	pipe length, [m]
P	number of phases
p_i	phase-pressure, [Pa]
Re_p	particle Reynolds number, [−]
$\mathbf{S}_f$	surface area vector, [m^2]
t	time, [s]
$\mathbf{u}_i$	phase-velocity, [ms^{-1}]
u_i	phase-velocity component, [ms^{-1}]

Greek letters

α_i	volume fraction, [−]
β	momentum exchange coefficient, [$\mathrm{kgm^{-3}s^{-1}}$]
Γ	generic diffusion coefficient
ε_i	turbulent kinetic energy dissipation, [$\mathrm{m^2s^{-3}}$]
Θ_p	granular temperature, [$\mathrm{m^2s^{-2}}$]
κ_p	particle fluctuation energy, [$\mathrm{m^2s^{-2}}$]
$\kappa_{\Theta s}$	diffusion coefficient for granular energy, [$\mathrm{kgm^{-1}s^{-1}}$]
μ_i	shear viscosity, [$\mathrm{kgm^{-1}s^{-1}}$]
$\mu_{i,t}$	turbulent shear viscosity, [$\mathrm{kgm^{-1}s^{-1}}$]
ν_i	kinematic viscosity, [$\mathrm{m^2s^{-1}}$]
$\nu_{i,t}$	turbulent kinematic viscosity, [$\mathrm{m^2s^{-1}}$]
ρ_i	density, [$\mathrm{kgm^{-3}}$]
τ_d	particle relaxation time, [s]

Subscripts

f	fluid
i	cell i
j	cell j
k	general index denoting a phase
m	1- k
p	particle
r	relative
T	total
x	x direction
y	y direction
z	z direction
f	face interpolated value

Superscripts

$*$	predicted
$\perp$	surface normal gradient
ϕ	generic variable
k	values at current iteration
$k-1$	values at previous iteration
p	pressure
P	phases
t	current time step
$t-1$	old time step

Table 4: Model characteristics & turbulence variables.

$$\beta = \frac{\rho_p \alpha_p}{\tau_d} = \frac{3}{4}\frac{\alpha_p \alpha_f \rho_f \mathbf{u}_r}{d_p} C_d$$

$$C_d = \begin{cases} \frac{24}{Re_p}\left[1 + 0.15 Re_p^{0.287}\right] & \text{if } Re_p < 1000 \\ 0.44 & \text{if } Re_p \geqslant 1000 \end{cases}$$

$$\mathrm{Sc}_{fp} = (k_f/k_p)^{1/2}$$

$$St = \tau_d/\tau_f$$

$$\tau_f = k_f/\varepsilon_f$$

$$e = 1$$

$$\Pi_p = 2\nu_{pt}\overline{\mathbf{S}}_\mathbf{p} : \overline{\mathbf{S}}_\mathbf{p} + \frac{2}{3}k_p \nabla \cdot \mathbf{u}_p$$

$$\Pi_f = 2\nu_{ft}\overline{\mathbf{S}}_\mathbf{f} : \overline{\mathbf{S}}_\mathbf{f} + \frac{2}{3}k_f \nabla \cdot \mathbf{u}_f$$

$$\overline{\mathbf{R}}_{\mathrm{eff,p}} = -2\nu_{\mathrm{eff,p}}\overline{\mathbf{S}}_\mathbf{p}$$

$$\overline{\mathbf{R}}_{\mathrm{eff,f}} = -2\nu_{\mathrm{eff,f}}\overline{\mathbf{S}}_\mathbf{f}$$

$$\overline{\mathbf{S}}_\mathbf{p} = \frac{1}{2}[\nabla \mathbf{u}_p + (\nabla \mathbf{u}_p)^T] - \frac{1}{3}\nabla \cdot \mathbf{u}_p \mathbf{I}$$

$$\overline{\mathbf{S}}_\mathbf{f} = \frac{1}{2}[\nabla \mathbf{u}_f + (\nabla \mathbf{u}_f)^T] - \frac{1}{3}\nabla \cdot \mathbf{u}_f \mathbf{I}$$

C_1	C_2	C_3	C_4	C_5	β_k	β_ε	$C_{f\mu}$	$C_{p\mu}$
1.44	1.92	1	1	1	1	1	0.09	0.09

Table 5: Definition of variables.

$$\kappa_p = k_p + 1.5\Theta_p$$

$$\mu_f = \rho_f \nu_f$$

$$\mu_{ft} = \alpha_f \rho_f \nu_{ft} = \alpha_f \rho_f C_{fu} k_f^2 / \varepsilon_f$$

$$\mu_p = \alpha_p \rho_p \nu_p = \frac{2\mu_{p_{dil}}}{(1+e)g_0}\Big[1 + \frac{4}{5}(1+e)g_0\alpha_p\Big]^2 + \frac{4}{5}\alpha_p^2 \rho_p d_p g_0 (1+e)\Big(\frac{\Theta_p}{\pi}\Big)^{1/2}$$

$$\mu_{p_{dil}} = \frac{5\sqrt{\pi}}{96}\rho_p d_p \Theta_p^{1/2}$$

$$\mu_{pt} = \alpha_p \rho_p \nu_{pt} = \alpha_p \rho_p C_{pu} k_p^2 / \varepsilon_p$$

$$p_p = \rho_p \alpha_p \Theta_p + 2(1+e)\rho_p \alpha_p^2 g_0 \Theta_p$$

$$\gamma = \frac{12(1-e^2)g_o}{\sqrt{\pi} d_p}\alpha_p^2 \rho_p \Theta_p^{3/2}$$

$$\kappa_\Theta = \frac{2}{(1+e)g_0}\Big[1 + \frac{6}{5}(1+e)g_0\alpha_p\Big]^2 \kappa_{\Theta,dil} + 2\alpha_p^2 \rho_p d_p g_0 (1+e)\Big(\frac{\Theta_p}{\pi}\Big)^{\frac{1}{2}}$$

$$\kappa_{\Theta,dil} = \frac{75}{384}\sqrt{\pi}\rho_p d_p \Theta_p^{1/2}$$

$$g_0 = \Big[1 - \Big(\frac{\alpha_p}{\alpha_{p,max}}\Big)^{\frac{1}{3}}\Big]^{-1}$$

$$k_{fp} = \beta_k \sqrt{k_f k_p}$$

$$\varepsilon_{fp} = \beta_\varepsilon \sqrt{\varepsilon_f \varepsilon_p}$$

7. Bibliography

[1] Anderson, T. B. and Jackson, R. (1967). Fluid mechanical description of fluidized beds: Equations of Motion. *Industrial and Engineering Chemistry Fundamentals*, 6(4):527–539.

[2] Braaten, M. and Patankar, S. (1989). A block-corrected subdomain solution procedure for recirculating flow calculations. *Numerical Heat Transfer, Part B: Fundamentals*, 15(1):1–20.

[3] Burns, A., Splawski, A., Lo, S., and Guetari, C. (2001). Application of coupled solver technology to cfd modeling of multiphase flows with cfx. *WIT Transactions on Engineering Sciences*, 30.

[4] Cardiff, P., Tuković, Ž., Jasak, H., and Ivanković, A. (2016). A block-coupled finite volume methodology for linear elasticity and unstructured meshes. *Computers & structures*, 175:100–122.

[5] Caretto, L., Curr, R., and Spalding, D. (1972). Two numerical methods for three-dimensional boundary layers. *Computer Methods in Applied Mechanics and Engineering*, 1(1):39 – 57.

[6] Chen, Z. and Przekwas, A. (2010). A coupled pressure-based computational method for incompressible/compressible flows. *Journal of Computational Physics*, 229(24):9150 – 9165.

[7] Clifford, I., J. H. (2009). The application of a multi-physics toolkit to spatial

reactor dynamics. In *International Conference on Mathematics, Computational Methods Reactor Physics, Curran Associates, Saratoga Springs, New York.*

[8] Cubero, A. and Fueyo, N. (2007). A compact momentum interpolation procedure for unsteady flows and relaxation. *Numerical Heat Transfer, Part B: Fundamentals*, 52(6):507–529.

[9] Cubero, A., Sánchez-Insa, A., and Fueyo, N. (2014). A consistent momentum interpolation method for steady and unsteady multiphase flows. *Computers Chemical Engineering*, 62:96 – 107.

[10] Darwish, M., Abdel Aziz, A., and Moukalled, F. (2015). A coupled pressure-based finite-volume solver for incompressible two-phase flow. *Numerical Heat Transfer, Part B: Fundamentals*, 67(1):47–74.

[11] Darwish, M., Sraj, I., and Moukalled, F. (2007). A coupled incompressible flow solver on structured grids. *Numerical Heat Transfer, Part B: Fundamentals*, 52(4):353–371.

[12] Darwish, M., Sraj, I., and Moukalled, F. (2009). A coupled finite volume solver for the solution of incompressible flows on unstructured grids. *Journal of Computational Physics*, 228(1):180–201.

[13] Ferreira, G. G., Lage, P. L., Silva, L. F. L., and Jasak, H. (2019). Implementation of an implicit pressure-velocity coupling for the eulerian multi-fluid model. *Computers Fluids*, 181:188 – 207.

[14] Ferziger, J. H. and Peric, M. (2002). *Computational Methods for Fluid Dynamics*. Springer.

[15] Fox, R. O. (2014). On multiphase turbulence models for collisional fluid-particle flows. *Journal of Fluid Mechanics*, 742:368–424.

[16] Hutchinson, B., Galpin, P., and Raithby, G. (1988). Application of additive correction multigrid to the coupled fluid flow equations. *Numerical Heat Transfer*, 13(2):133–147.

[17] Issa, R. I. (1986). Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of Computational Physics*, 62(1):40–65.

[18] Jareteg, K., Vukčević, V., and Jasak, H. (2014). pucoupledfoam-an open source coupled incompressible pressure-velocity solver based on foam-extend. In *9th OpenFOAM Workshop*.

[19] Jasak, H. (1996). Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows. *PhD Thesis,*, Imperial College London(June).

[20] Karki, K. C. and Mongia, H. C. Evaluation of a coupled solution approach for fluid flow calculations in body-fitted co-ordinates. *International Journal for Numerical Methods in Fluids*, 11(1):1–20.

[21] Lonsdale, R, D. (1991). An algebraic multigrid scheme for solving the navier-stokes equations on unstructured meshes. In *Proc. 7th Int. Conf. on Numerical Methods in Turbulent and Laminar Flows.*

[22] Lun, C. K. K. (2000). Numerical simulation of dilute turbulent gas–solid flows. *International Journal of Multiphase Flow*, 26(10):1707–1736.

[23] Mazhar, Z. (2001). A procedure for the treatment of the velocity-pressure coupling problem in incompressible fluid flow. *Numerical Heat Transfer: Part B: Fundamentals*, 39(1):91–100.

[24] Mazhar, Z. (2016). A novel fully implicit block coupled solution strategy for the ultimate treatment of the velocity–pressure coupling problem in incompressible fluid flow. *Numerical Heat Transfer, Part B: Fundamentals*, 69(2):130–149.

[25] Patankar, S. V. and Spalding, D. B. (1983). A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. In *Numerical Prediction of Flow, Heat Transfer, Turbulence and Combustion*, pages 54–73. Elsevier.

[26] Raw, M. (1995). *A Coupled Algebraic Multigrid Method for the 3D Navier-Stokes Equations*, pages 204–215. Vieweg+Teubner Verlag, Wiesbaden.

[27] Rhie, C. M. and Chow, W. L. (1983). A numerical study of the turbulent flow past an isolated airfoil with trailing edge separation. *AIAA J.*, pages 1525–1532.

[28] Riella, M. (2019). *Turbulence modelling of fluid-particle interaction*. PhD thesis, University of Exeter. http://hdl.handle.net/10871/37927.

[29] Riella, M., Kahraman, R., and Tabor, G. (2018). Reynolds-averaged two-fluid model prediction of moderately dilute fluid-particle flow over a backward-facing step. *International Journal of Multiphase Flow*, 106:95 – 108.

[30] Riella, M., Kahraman, R., and Tabor, G. (2019a). Inhomogeneity and anisotropy in eulerian-eulerian near-wall modelling. *International Journal of Multiphase Flow*, 114:9 – 18.

[31] Riella, M., Kahraman, R., and Tabor, G. (2019b). Near-wall modelling in eulerian-eulerian simulations. *Computers Fluids*, 190:206 – 216.

[32] Rusche, H. (2002). Computational Fluid Dynamics of Dispersed Two-Phase Flows at High Phase Fractions. *PhD Thesis*, (December).

[33] Tsuji, Y., Morikawa, Y., and Shiomi, H. (1984). LDV measurements of an air-solid two-phase flow in a vertical pipe. *Journal of Fluid Mechanics*, 120:385–409.

[34] Uroić, T. and Jasak, H. (2018). Block-selective algebraic multigrid for implicitly coupled pressure-velocity system. *Computers & fluids*, 167(1):100–110.

[35] Vanka, S. (1986). Block-implicit multigrid solution of navier-stokes equations in primitive variables. *Journal of Computational Physics*, 65(1):138 – 158.

[36] Webster, R. An algebraic multigrid solver for navier-stokes problems. *International Journal for Numerical Methods in Fluids*, 18(8):761–780.

[37] Webster, R. An algebraic multigrid solver for navier-stokes problems in discrete second-order approximation. *International Journal for Numerical Methods in Fluids*, 22(11):1103–1123.

[38] Weller, H. (2002). Derivation, modelling and solution of the conditionally averaged two-phase flow equations. *Tech.Rep.OpenCFD Ltd.*

[39] Xiao, C.-N., Denner, F., and van Wachem, B. G. (2017). Fully-coupled pressure-based finite-volume framework for the simulation of fluid flows at all speeds in complex geometries. *Journal of Computational Physics*, 346:91–130.

[40] Zalesak, S. T. (1979). Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics*, 31(3):335–362.