Numerical Heat Transfer, Part B, 67: 47–74, 2015

ISSN: 1040-7790 print/1521-0626 online
DOI: 10.1080/10407790.2014.949500

Taylor & Francis
Taylor & Francis Group

A COUPLED PRESSURE-BASED FINITE-VOLUME SOLVER FOR INCOMPRESSIBLE TWO-PHASE FLOW

M. Darwish, A. Abdel Aziz, and F. Moukalled
Department of Mechanical Engineering, American University of Beirut, Beirut, Lebanon

This article reports on the formulation and testing of a coupled pressure-based algorithm for the solution of steady incompressible disperse two-phase-flow problems. The method is formulated within a Eulerian-Eulerian framework in the context of a collocated finite-volume scheme. An equation for pressure is derived from overall mass conservation following the segregated mass conservation–based algorithm (MCBA) approach and using an extended two-phase flow form of the Rhie-Chow interpolation technique. The newly developed pressure-based coupled solver differs from pressure-based segregated solvers in that it accounts implicitly for the pressure–velocity and the interphase drag couplings that are present in disperse multiphase flows to yield a system of coupled equations linking the velocity and pressure fields. The performance and accuracy of the coupled multiphase algorithm are assessed by solving eight one-dimensional two-phase flow problems spanning the spectrum from dilute bubbly to dense gas–solid flows. Each problem is solved over three grid systems with sizes of 10,000, 30,000, and 50,000 control volumes, respectively. Results are compared in terms of iterations and CPU time with similar ones generated using the segregated MCBA-SIMPLE algorithm. The newly developed coupled solver is shown to yield substantial decrease in the required number of iterations and CPU time, with the rate of solution acceleration varying between 1.3 and 4.6.

INTRODUCTION

The pressure-based approach for solving fluid flow problems has been at the heart of research interest in computational fluid dynamics (CFD) since the development of the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) by the research group at Imperial College [1]. The SIMPLE approach has been preferred by the CFD community due to the simplicity of its implementation and low memory requirement. These facts boosted developments following the segregated approach and resulted in a family of SIMPLE-like algorithms, which were proven by the authors [2, 3] to share similar behaviors. The iterative nature of these algorithms rendered them very dependent on the grid density and the value of the underrelaxation factor used. In general, the denser the mesh, the more iterations are needed to reach convergence. In addition, high underrelaxation is usually required to promote convergence.

Received 26 February 2014; accepted 28 May 2014.
Address correspondence to F. Moukalled, Department of Mechanical Engineering, American University of Beirut, P.O. Box 11-0236, Riad El Solh, Beirut 1107 2020, Lebanon. E-mail: fmukalled@aub.edu.lb

NOMENCLATURE

$a_P^{u^k u^k}, \ldots$	coefficients in the discretized equations
$b_P^{\phi^k}$	source term in the discretized equations
B	source term in the momentum equation
C_D	drag coefficient
$\mathbf{d}_{PF}$	vector joining the grid points P and F
D	operator used in the pressure equation
E	component of the surface vector in the direction of $\mathbf{d}_{PF}$
F	neighbor of the P grid point
g	gravitational acceleration
g^{km}	interfluid drag term coefficient
I	interphase source term
$\dot{m}_f$	mass flow rate at control-volume face f
$\dot{M}^k$	rate at which mass is gained/lost by phase k
p	pressure
P	main grid point
Q	general source term
r	geometric interpolation factor
RES	residuals
S	surface vector
T	component of the surface vector normal to **S**
u, v	velocity components in the x and y directions, respectively
v	velocity vector
V	cell volume
α	volume fraction
Γ	diffusion coefficient in general equation
μ	dynamic viscosity
υ	kinematic viscosity
ρ	fluid density
ϕ	general variable

Subscripts

f	refers to control-volume face
F	refers to the F grid point
inlet	refers to conditions at inlet
nb	refers to values at the faces obtained by interpolation between P and its neighbors
NB	refers to the neighbors of the P grid point
P	refers to main grid point
slip	refers to slip velocity

Superscripts

c	refers to continuous phase
d	refers to disperse phase
k	refers to phase k
p	refers to pressure
u	refers to the u-velocity component
v	refers to the v-velocity component
ϕ	refers to general variable
$-$	refers to an interpolated value

Efforts to devise velocity–pressure algorithms that are more robust and efficient have resulted in a better understanding of the numerical issues affecting their performance, such as the choice of primitive variables (density-based [4] versus pressure-based [2, 5]), the type of variable arrangement (staggered grid [6] versus collocated grid [7]), and the chosen solution approach (coupled [8] versus segregated [6]), to cite a few. These issues and their effects are now much better understood by the research community. This acquired knowledge has led to a renewed interest in pressure–velocity coupled solvers [9–11], partly because of the tremendous increase in computer memory and partly due to the convergence problems experienced by segregated solvers when used in large computational domains [12]. Even though the last problem has been addressed successfully through parallel processing and domain decomposition, the convergence issue has not itself been resolved. It is worthwhile in this respect to point out that density-based Euler solvers have been using the coupled approach quite successfully.

While early attempts to develop coupled solvers were unsuccessful due to the huge memory requirements at the time, such algorithms proved to be competitive with the segregated formulation [13–16]. Recently, the authors reported, in two separate articles [10, 11], on a coupled solver capable of accelerating the solution

of single-phase-flow problems by an order of magnitude as compared to the segregated solver. The method was nearly grid- and underrelaxation-independent.

For multiphase flow problems the complexity is further magnified by coupling of the pressure field to several velocity fields [17–23] and from the interfluid drag terms coupling the velocity fields together. In addition, the increased number of coupled equations adds to the intricacy of the problem. This dual layer of nonlinearity has resulted in large computational time needed to solve even small-scale multiphase flow problems.

In this article, the single-phase coupled solver, previously developed by the authors [10, 11], is extended to multiphase flows. The phasic momentum equations and the global pressure equation are grouped into a single matrix that is solved implicitly for all dependent variables simultaneously. The pressure–velocity coupling is formulated within a collocated grid system following the mass conservation–based algorithm (MCBA) developed in [17].

In what follows, the governing equations for multiphase flow are introduced and their discretization outlined. The coupled algorithm is then presented and its performance compared to the segregated multiphase-flow finite-volume solver [22, 23] in terms of number of iterations and computation time to convergence, by solving eight one-dimensional two-phase-flow problems.

GOVERNING EQUATIONS

The equations governing steady, incompressible, multiphase flows are the phasic mass conservation equations, the phasic momentum equations, the global continuity equation, and the geometric conservation equation given, respectively, as

$$\nabla \cdot \left(\rho^k \alpha^k \mathbf{v}^k\right) = \alpha^k \dot{M}^k \tag{1}$$

$$\nabla \cdot (\rho^k \alpha^k \mathbf{v}^k \mathbf{v}^k) = \nabla \cdot (\mu^k \alpha^k \nabla \mathbf{v}^k) + \alpha^k (\mathbf{B}^k - \nabla p) + \sum_{m \neq k} g^{km} (\mathbf{v}^m - \mathbf{v}^k) \tag{2}$$

$$\sum_k \nabla \cdot (\rho^k \alpha^k \mathbf{v}^k) = \sum_k \alpha^k \dot{M}^k = 0 \tag{3}$$

$$\sum_k \alpha^k = 1 \tag{4}$$

In these equations, k represents a phase whose density, velocity, dynamic viscosity, volume fraction, and body force are denoted by ρ^k, $\mathbf{v}^k$, μ^k, α^k, and $\mathbf{B}^k$, respectively. Moreover, p is the common pressure field, g^{km} the interfluid drag term coefficient, and $\dot{M}^k$ the mass source rate.

Generally, the dependent variables in any multiphase flow problem are the velocity, volume fraction, and pressure fields. The above equations have to be arranged in a way to solve for these variables. In this work, the mass conservation–based algorithm (MCBA) [17, 18] is used to solve the resulting system of equations. If the number of available phases is N, $(N-1)$ volume fraction fields are found using $(N-1)$ phasic mass conservation equations, and the last volume fraction field is found by applying

the geometric conservation law. The N velocity field vectors are found by solving the N available momentum equations. The remaining volume-fraction equation can be used to calculate the pressure field. Instead, the global conservation equation [Eq. (3)] (i.e., the sum of the individual mass conservation equations) is employed to derive a pressure-correction equation via an extended two-phase-flow version of the Rhie-Chow interpolation technique [7]. The sequence of events in the MCBA is as follows:

1. Solve the phasic momentum equations for velocities.
2. Solve the pressure-correction equation based on global mass conservation.
3. Correct velocities and pressure.
4. Solve the phasic mass conservation equations for volume fractions.
5. Return to the first step and repeat until convergence.

FINITE-VOLUME DISCRETIZATION

Any of the conservation equations presented above can be represented via the following general generic equation:

$$\nabla \cdot \left(\alpha^k \rho^k \mathbf{v}^k \phi^k\right) = \nabla \cdot \left(\alpha^k \Gamma^{\phi^k} \nabla \phi^k\right) + \alpha^k Q^{\phi^k} \tag{5}$$

in which the meanings of ϕ^k, Γ^{ϕ^k}, and Q^{ϕ^k} differ depending on the modeled equation. In the finite-volume method, the domain is discretized by dividing it into a number of control volumes each associated with a main grid point placed at its centroid. The discretization of the governing conservation equations is accomplished by integrating the general transport equation over the control volume shown in Figure 1 to yield

$$\int_V \nabla \cdot \left(\alpha^k \rho^k \mathbf{v}^k \phi^k\right) \, dV = \int_V \nabla \cdot \left(\alpha^k \Gamma^{\phi^k} \nabla \phi^k\right) \, dV + \int_V \alpha^k Q^{\phi^k} \, dV \tag{6}$$

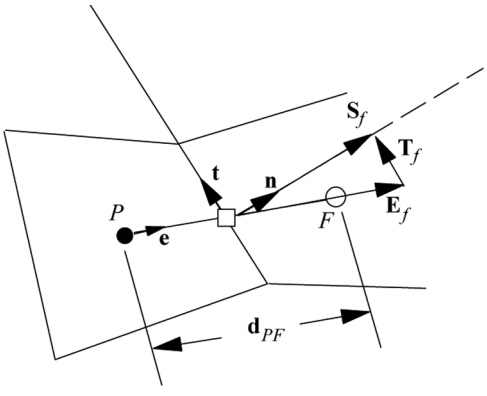

Figure 1. A schematic of control volume P, its neighbor control volume F, and related geometric quantities.

Then, using the divergence theorem, the volume integral of the flux terms in Eq. (6) are transformed into surface integrals as

$$\int_{\partial V} \left(\alpha^k \rho^k \mathbf{v}^k \phi^k\right) \cdot d\mathbf{S} = \int_{\partial V} \left(\alpha^k \Gamma^{\phi^k} \nabla \phi^k\right) \cdot d\mathbf{S} + \int \int_V \alpha^k Q^{\phi^k} \, dV \tag{7}$$

The semidiscrete form is obtained by replacing the surface integrals of the fluxes by discrete summations over the faces of the control volume, and the volume integral by the product of the integrand at the cell centroid and the cell volume. Upon substitution into Eq. (7), the semidiscrete form of the equation is obtained as

$$\sum_{f=\mathrm{nb}(P)} \left(\alpha^k \rho^k \mathbf{v}^k \phi^k - \alpha^k \Gamma^{\phi^k} \nabla \phi^k\right)_f \cdot \mathbf{S}_f = \alpha^k Q_P^{\phi^k} V_P \tag{8}$$

Using appropriate interpolation schemes, the above semidiscretized equation is transformed to a fully discrete algebraic equation of the form

$$a_P^{\phi^k} \phi_P^k + \sum_{F=\mathrm{NB}(P)} a_F^{\phi^k} \phi_F^k = b_P^{\phi^k} \tag{9}$$

In this work, the convection terms are approximated using the first-order upwind scheme, while the diffusion terms are calculated via a second-order central difference scheme.

THE DISCRETIZED EQUATIONS FOR THE COUPLED ALGORITHM

Regardless of the solution procedure followed, segregated or coupled, the adopted solution method roughly proceeds in the following two steps:

1. Update the velocity and pressure fields using volume-fraction fields from the previous iteration.
2. Use the updated mass flow rate field to find the new volume-fraction fields.

In a segregated solver such as SIMPLE (which is the segregated algorithm used in this work for comparison), the continuity and momentum equations are decoupled. The momentum equations are used to find the velocity fields assuming known pressure field, and then the continuity equation is solved for the pressure field using the last updated values of velocity. Note that the momentum equations are themselves decoupled from each other in SIMPLE by solving each phasic momentum equation independently for the corresponding phase.

In the coupled algorithm, however, both the velocity and the pressure fields are the main variables in the whole set of equations. The momentum equations are solved together with the pressure equation simultaneously and for all phases.

The solution procedure of a multiphase problem using the developed coupled algorithm can be summarized as

1. Start with the most updated values of velocity, pressure, and volume-fraction fields for each phase.

2. Simultaneously solve the system of momentum and pressure equations to update the velocity and pressure fields for all the phases.
3. Update the mass flow rate of each phase at the faces using the Rhie-Chow approximation.
4. Solve the phasic mass conservation equations and update the volume-fraction fields.
5. Repeat until convergence.

The detailed discretization for solving single-phase-flow problems using the coupled algorithm was reported in [10, 11]. In this article, the differences between the single and multiphase problems are pointed out. Further, the current formulation does not couple the phasic mass conservation equations (volume-fraction equations) to the velocity and pressure fields. The tight coupling of all equations together is expected to further enhance convergence and reduce computation time. However, this necessitates deriving a pressure equation following the geometric conservation–based algorithm (GCBA) [17], which will be considered in future works.

DISCRETIZATION OF THE MOMENTUM EQUATIONS

Starting with the momentum equation [Eq. (2)], there are two main differences between the multiphase and single-phase cases. First, all the terms in the momentum equation are now multiplied by the phasic volume fraction. Second, an interphase drag term appears in the equation. Integrating this term over a control volume assuming that the velocity of the fluid prevails over the control volume, its discretized form is found as

$$\int_V \sum_{m\neq k} g^{km}\left(\mathbf{v}^m - \mathbf{v}^k\right)\, dV = V_P \sum_{m\neq k} g^{km}\left(\mathbf{v}_P^m - \mathbf{v}_P^k\right) \tag{10}$$

Whereas in a segregated formulation this term is split into two parts with the part containing $\mathbf{v}_P^k$ moved to the left-hand side and treated implicitly, in the coupled formulation the whole term is added to the coefficients on the left-hand side, as all the fields are treated implicitly. Performing this step, the final forms of the u^k- and v^k-momentum equations can be written as

$$\begin{cases} a_P^{u^k u^k} u_P^k + a_P^{u^k v^k} v_P^k + a_P^{u^k p} p_P + \displaystyle\sum_{F=\mathrm{NB}(P)} a_F^{u^k u^k} u_F^k \\ \quad + \displaystyle\sum_{\mathrm{NB}=\mathrm{NB}(P)} a_F^{u^k v^k} v_F^k + \sum_{\mathrm{NB}=\mathrm{NB}(P)} a_F^{u^k p} p_F + \sum_{m\neq k} a_P^{u^k u^m} u_P^m = b_P^{u^k} \\ a_P^{v^k u^k} u_P^k + a_P^{v^k v^k} v_P^k + a_P^{v^k p} p_P + \displaystyle\sum_{F=\mathrm{NB}(P)} a_F^{v^k u^k} u_F^k \\ \quad + \displaystyle\sum_{\mathrm{NB}=\mathrm{NB}(P)} a_F^{v^k v^k} v_F^k + \sum_{\mathrm{NB}=\mathrm{NB}(P)} a_F^{v^k p} p_F + \sum_{m\neq k} a_P^{v^k v^m} v_P^m = b_P^{v^k} \end{cases} \tag{11}$$

The coefficients appearing in Eq. (11) are given by

$$a_P^{u^k u^k} = a_P^{v^k v^k} = \sum_{f=\mathrm{nb}(P)} \left\| \dot{m}_f^k,\ 0 \right\| \left(\alpha_f^k\right)^* + \sum_{f=\mathrm{nb}(P)} \left[\left(\alpha_f^k\right)^* \mu_f^k \frac{\mathbf{S}_f \cdot \mathbf{S}_f}{\mathbf{S}_f \cdot \mathbf{d}_{PF}} \right] + \sum_{m \neq k} g^{km} V_P \quad (12a)$$

$$a_F^{u^k u^k} = a_F^{v^k v^k} = -\left\| -\dot{m}_f^k,\ 0 \right\| \left(\alpha_f^k\right)^* - \left(\alpha_f^k\right)^* \mu_f^k \frac{\mathbf{S}_f \cdot \mathbf{S}_f}{\mathbf{S}_f \cdot \mathbf{d}_{PF}} \quad a_P^{u^k u^m} = a_P^{v^k v^m} = g^{km} V_P \quad (12b)$$

$$a_P^{u^k p} = \sum_{f=\mathrm{nb}(P)} \left(\alpha_f^k\right)^* r S_f^x \quad a_F^{u^k p} = \left(\alpha_f^k\right)^* (1-r) S_f^x \quad (12c)$$

$$a_P^{v^k p} = \sum_{f=\mathrm{nb}(P)} \left(\alpha_f^k\right)^* r S_f^y \quad a_F^{v^k p} = \left(\alpha_f^k\right)^* (1-r) S_f^y \quad (12d)$$

$$b_P^{u^k} = \sum_{f=\mathrm{nb}(P)} \left[\left(\alpha_f^k\right)^* \mu_f^k \nabla u_f^k \cdot \mathbf{T}_f \right] + \left(\alpha_P^k\right)^* B_P^{k,x} V_P \quad (12e)$$

$$b_P^{v^k} = \sum_{f=\mathrm{nb}(P)} \left[\left(\alpha_f^k\right)^* \mu_f^k \nabla v_f^k \cdot \mathbf{T}_f \right] + \left(\alpha_P^k\right)^* B_P^{k,y} V_P \quad (12f)$$

As mentioned above, the coefficients in Eq. (12) are obtained using a first-order upwind formulation of the convection terms. This choice does not present any limitation for the new solver. The use of higher-order schemes amounts to an additional source term in the algebraic equation when implemented using the deferred correction procedure of Rubin and Khosla [24]. The effect of this additional source term on the solution acceleration of the coupled solver for incompressible single-phase flows was studied by Moukalled et al. in [25]. An increase in computation cost was reported. Nonetheless, the computation time of the segregated solution also increases with the use of high-resolution schemes, and the relative performance of the coupled algorithm remains more or less unaffected. Similar increase in computation cost is also expected with multiphase flows. However, this will not be discussed here and will form the subject of a future article.

THE TWO-PHASE RHIE-CHOW INTERPOLATION

The Rhie-Chow interpolation procedure [7] for the calculation of the velocity components at a control-volume face enables the formulation of the single-phase SIMPLE algorithm on collocated grids. A multifluid version of this procedure is required for multiphase flows. In either case, the purpose is to construct a pseudo-momentum equation for the interface velocity vector at the control-volume face from the momentum equations at the control volumes straddling the interface.

For single-fluid flows, the computation of the Rhie-Chow coefficient is relatively simple [7, 11, 26]. The derivation starts by writing the x-momentum equation for an element P as

$$a_P^u u_P = -\sum_{NP=\mathrm{NB}(P)} a_{NP}^u u_{NP} - V_P \left(\frac{\partial p}{\partial x}\right)_P + b_P^u \quad (13)$$

and for a neighboring element F to P as

$$a_F^u u_F = -\sum_{NF=\mathrm{NB}(F)} a_{NF}^u u_{NF} - V_F \left(\frac{\partial p}{\partial x}\right)_F + b_F^u \tag{14}$$

The velocity at the interface f between P and F should satisfy

$$\bar{a}_f^u u_f = -\overline{\sum_{n=nb(f)} a_n^u u_n} - \bar{V}_f \left(\frac{\partial p}{\partial x}\right)_f + \bar{b}_f^u \tag{15}$$

while the interpolated velocity to the common face f obtained from the two equations (13) and (14) is given by

$$\bar{a}_f^u \bar{u}_f = -\overline{\sum_{n=\mathrm{nb}(f)} a_n^u u_n} - \bar{V}_f \overline{\left(\frac{\partial p}{\partial x}\right)}_f + \bar{b}_f^u \tag{16}$$

Combining Eqs. (15) and (16), the Rhie-Chow interpolation is obtained as

$$u_f = \bar{u}_f - \bar{D}_f^u \left[\left(\frac{\partial p}{\partial x}\right)_f - \overline{\left(\frac{\partial p}{\partial x}\right)_f}\right] \tag{17}$$

where

$$\bar{D}_f^u = \frac{\bar{V}_f}{\bar{a}_f^u} \tag{18}$$

and the average value is given by

$$\overline{\square_f} = r_P \square_P + r_F \square_F \quad r_P = \frac{V_F}{V_p + V_F} \quad r_F = 1 - r_P \tag{19}$$

where f denotes the face common to elements P and F, r is the geometric interpolation factor, a_P^u and a_F^u are the main coefficients of the P and F grid points of the u-momentum equation, and V is the element volume.

For a two-phase flow simulation with interphase drag, the x-momentum equations for the velocities u^k and u^m of the two fluids at element P can be written as

$$\begin{bmatrix} a_P^{u^k u^k} & a_P^{u^k u^m} \\ a_P^{u^m u^k} & a_P^{u^m u^m} \end{bmatrix} \begin{bmatrix} u_P^k \\ u_P^m \end{bmatrix} = -\sum_{NP=\mathrm{NB}(P)} \begin{bmatrix} a_{NP}^{u^k u^k} & 0 \\ 0 & a_{NP}^{u^m u^m} \end{bmatrix} \begin{bmatrix} u_{NP}^k \\ u_{NP}^m \end{bmatrix} - \begin{bmatrix} \alpha_P^k \\ \alpha_P^m \end{bmatrix} \left(\frac{\partial p}{\partial x}\right)_P V_P + \begin{bmatrix} b_P^k \\ b_P^m \end{bmatrix} \tag{20}$$

A similar equation can be written for the element F, which is a neighbor of P and is given by

$$\begin{bmatrix} a_F^{u^k u^k} & a_F^{u^k u^m} \\ a_F^{u^m u^k} & a_F^{u^m u^m} \end{bmatrix} \begin{bmatrix} u_F^k \\ u_F^m \end{bmatrix} = -\sum_{NF=\mathrm{NB}(F)} \begin{bmatrix} a_{NF}^{u^k u^k} & 0 \\ 0 & a_{NF}^{u^m u^m} \end{bmatrix} \begin{bmatrix} u_{NF}^k \\ u_{NF}^m \end{bmatrix} - \begin{bmatrix} \alpha_F^k \\ \alpha_F^m \end{bmatrix} \left(\frac{\partial p}{\partial x}\right)_F V_F + \begin{bmatrix} b_F^k \\ b_F^m \end{bmatrix} \tag{21}$$

At the control-volume face f, velocity equations similar to the ones given by Eqs. (20) and (21), with the pressure gradient linked to the local neighboring pressure values, will have the form

$$\overline{\begin{bmatrix} a_f^{u^k u^k} & a_f^{u^k u^m} \\ a_f^{u^m u^k} & a_f^{u^m u^m} \end{bmatrix}} \begin{bmatrix} u_f^k \\ u_f^m \end{bmatrix} = -\overline{\sum_{n=\mathrm{nb}(f)} \begin{bmatrix} a_n^{u^k u^k} & 0 \\ 0 & a_n^{u^m u^m} \end{bmatrix} \begin{bmatrix} u_n^k \\ u_n^m \end{bmatrix}} - \overline{\begin{bmatrix} \alpha_f^k \\ \alpha_f^m \end{bmatrix}} \bar{V}_f \left(\frac{\partial p}{\partial x}\right)_f + \overline{\begin{bmatrix} b_f^k \\ b_f^m \end{bmatrix}} \tag{22}$$

Since with a collocated grid the coefficients cannot be computed directly, they are approximated by interpolation from the coefficients of the neighboring nodes. Adopting a linear interpolation profile, the coefficients at the common face f are obtained from Eqs. (20) and (21) as

$$\begin{aligned} \overline{\begin{bmatrix} a_f^{u^k u^k} & a_f^{u^k u^m} \\ a_f^{u^m u^k} & a_f^{u^m u^m} \end{bmatrix}} \begin{bmatrix} \bar{u}_f^k \\ \bar{u}_f^m \end{bmatrix} = & -\overline{\sum_{n=\mathrm{nb}(f)} \begin{bmatrix} a_n^{u^k u^k} & 0 \\ 0 & a_n^{u^m u^m} \end{bmatrix} \begin{bmatrix} u_n^k \\ u_n^m \end{bmatrix}} \\ & - \overline{\begin{bmatrix} \alpha_f^k \\ \alpha_f^m \end{bmatrix}} \bar{V}_f \overline{\left(\frac{\partial p}{\partial x}\right)_f} + \overline{\begin{bmatrix} b_f^k \\ b_f^m \end{bmatrix}} \end{aligned} \tag{23}$$

Combining Eqs. (22) and (23), the following relations is obtained:

$$\begin{aligned} \overline{\begin{bmatrix} a_f^{u^k u^k} & a_f^{u^k u^m} \\ a_f^{u^m u^k} & a_f^{u^m u^m} \end{bmatrix}} \begin{bmatrix} u_f^k \\ u_f^m \end{bmatrix} = & \overline{\begin{bmatrix} a_f^{u^k u^k} & a_f^{u^k u^m} \\ a_f^{u^m u^k} & a_f^{u^m u^m} \end{bmatrix}} \begin{bmatrix} \bar{u}_f^k \\ \bar{u}_f^m \end{bmatrix} \\ & - \overline{\begin{bmatrix} \alpha_f^k \\ \alpha_f^m \end{bmatrix}} \bar{V}_f \left[\left(\frac{\partial p}{\partial x}\right)_f - \overline{\left(\frac{\partial p}{\partial x}\right)_f} \right] \end{aligned} \tag{24}$$

which can be simplified to

$$\begin{bmatrix} u_f^k \\ u_f^m \end{bmatrix} = \begin{bmatrix} \bar{u}_f^k \\ \bar{u}_f^m \end{bmatrix} - \left(\overline{\begin{bmatrix} a_f^{u^k u^k} & a_f^{u^k u^m} \\ a_f^{u^m u^k} & a_f^{u^m u^m} \end{bmatrix}} \right)^{-1} \overline{\begin{bmatrix} \alpha_f^k \\ \alpha_f^m \end{bmatrix}} \bar{V}_f \left[\left(\frac{\partial p}{\partial x}\right)_f - \overline{\left(\frac{\partial p}{\partial x}\right)_f} \right] \tag{25}$$

In a more compact form, Eq. (25) gives the Rhie-Chow interpolation relation as

$$\begin{bmatrix} u_f^k \\ u_f^m \end{bmatrix} = \begin{bmatrix} \overline{u_f^k} \\ \overline{u_f^m} \end{bmatrix} - \overline{\mathbf{D}}_f^u \left[\left(\frac{\partial p}{\partial x}\right)_f - \overline{\left(\frac{\partial p}{\partial x}\right)_f} \right] \tag{26}$$

where now

$$\mathbf{D}_f^u = \begin{bmatrix} \overline{\left(\frac{\alpha_f^k a_f^{u^m u^m} - \alpha_f^m a_f^{u^k u^m}}{a_f^{u^k u^k} a_f^{u^m u^m} - a_f^{u^k u^m} a_f^{u^m u^k}} \right)} \bar{V}_f \\ \overline{\left(\frac{\alpha_f^m a_f^{u^k u^k} - \alpha_f^k a_f^{u^m u^k}}{a_f^{u^k u^k} a_f^{u^m u^m} - a_f^{u^k u^m} a_f^{u^m u^k}} \right)} \bar{V}_f \end{bmatrix} = \begin{bmatrix} D_f^{u^k} \\ D_f^{u^m} \end{bmatrix} \tag{27}$$

A similar equation can be obtained in the y direction and is given by

$$\overline{\mathbf{D}}_f^y = \begin{bmatrix} \overline{\left(\frac{\alpha_f^k a_f^{v^m v^m} - \alpha_f^m a_f^{v^k v^m}}{a_f^{v^k v^k} a_f^{v^m v^m} - a_f^{v^k v^m} a_f^{v^m v^k}}\right)} \bar{V}_f \\ \overline{\left(\frac{\alpha_f^m a_f^{v^k v^k} - \alpha_f^k a_f^{v^m v^k}}{a_f^{v^k v^k} a_f^{v^m v^m} - a_f^{v^k v^m} a_f^{v^m v^k}}\right)} \bar{V}_f \end{bmatrix} = \begin{bmatrix} D_f^{v^k} \\ D_f^{v^m} \end{bmatrix} \tag{28}$$

Combining the above two equations, the two-phase flow form of the Rhie-Chow interpolation is obtained as

$$\underbrace{\begin{bmatrix} u_f^k \\ v_f^k \end{bmatrix}}_{\mathbf{v}_f^k} = \underbrace{\begin{bmatrix} \bar{u}_f^k \\ \bar{v}_f^k \end{bmatrix}}_{\bar{\mathbf{v}}_f^k} - \underbrace{\begin{bmatrix} D_f^{u^k} & 0 \\ 0 & D_f^{v^k} \end{bmatrix}}_{\overline{\mathbf{D}}_f^{v^k}} \underbrace{\begin{bmatrix} \left(\frac{\partial p}{\partial x}\right)_f - \left(\overline{\frac{\partial p}{\partial x}}\right)_f \\ \left(\frac{\partial p}{\partial y}\right)_f - \left(\overline{\frac{\partial p}{\partial y}}\right)_f \end{bmatrix}}_{\nabla p_f - \overline{\nabla p_f}} \tag{29}$$

or in a compact form, suitable for deriving the pressure equation, as

$$\mathbf{v}_f^k = \bar{\mathbf{v}}_f^k - \overline{\mathbf{D}}_f^{\mathbf{v}^k} \left(\nabla p_f - \overline{\nabla p_f}\right) \tag{30}$$

In the limit of no interphase drag the following holds:

$$a^{u^k u^m} = a^{u^m u^k} = 0 \tag{31}$$

Thus, the single-phase formulation is recovered with $\overline{\mathbf{D}}_f^{\mathbf{v}^k}$ given by

$$\overline{\mathbf{D}}_f^{\mathbf{v}^k} = \begin{bmatrix} \frac{\bar{V}}{a_f^{u^k u^k}} & 0 \\ 0 & \frac{\bar{V}}{a_f^{v^k v^k}} \end{bmatrix} = \begin{bmatrix} \frac{\bar{V}}{a_f^u} & 0 \\ 0 & \frac{\bar{V}}{a_f^v} \end{bmatrix}_f = \overline{\mathbf{D}}_f^{\mathbf{v}} \tag{32}$$

In disperse-flow problems, where the momentum coefficients are more important, the simplified (single-phase) Rhie-Chow relation can be used.

DISCRETIZATION OF THE PRESSURE EQUATION

The pressure equation is derived from the global mass conservation equation, which is the sum of all phasic mass conservation equations. Therefore the coefficients of the pressure equation represent the sum, over all available phases, of the coefficients that would arise if each phase was alone. The overall mass conservation equation is given by

$$\sum_k \nabla \cdot \left(\rho^k \alpha^k \mathbf{v}^k\right) = 0 \tag{33}$$

Integrating Eq. (33) over the control volume shown in Figure 1, it is transformed to

$$\int_V \sum_k \nabla \cdot \left(\rho^k \alpha^k \mathbf{v}^k\right) \quad dV = 0 \tag{34}$$

Using the divergence theorem and invoking the Rhie-Chow interpolation, the semidiscrete form is obtained as

$$\sum_k \sum_f \left[\rho^k \alpha^k \overline{\mathbf{v}}_f^k - \rho^k \alpha^k \overline{\mathbf{D}}_f^{\mathbf{v}^k} \left(\nabla p - \overline{\nabla p} \right) \right]_f \cdot \mathbf{S}_f = 0 \tag{35}$$

Decomposing the surface vector $\mathbf{S}_f$ (Figure 1) into a vector $\mathbf{E}_f$ in the direction of the line joining the P and F grid points and a vector $\mathbf{T}_f$ perpendicular to $\mathbf{S}_f$, then

$$\mathbf{S}_f = \mathbf{E}_f + \mathbf{T}_f \tag{36}$$

Moreover, the interpolated $\overline{\mathbf{v}}_f^k$ is written in terms of $\mathbf{v}_P^k$ and $\mathbf{v}_F^k$ values as

$$\overline{\mathbf{v}}_f^k = r_P \overline{\mathbf{v}}_P^k + (1 - r_P) \mathbf{v}_F^k \tag{37}$$

Further, the pressure gradient at the interface is discretized as

$$(\nabla p)_f \cdot \mathbf{S}_f = (\nabla p)_f \cdot \mathbf{E}_f + (\nabla p)_f \cdot \mathbf{T}_f = \frac{p_F - p_P}{d_{PF}} E_f + (\nabla p)_f \cdot \mathbf{T}_f \tag{38}$$

where E_f is the magnitude of $\mathbf{E}_f$, d_{FP} is the distance between grid points P and F, and $(\nabla p)_f \cdot \mathbf{T}_f$ is treated as a source term, in a deferred-correction manner, with its value computed explicitly using the current pressure field.

When all terms are substituted into Eq. (35), the discretized pressure equation for two-phase flow is obtained as

$$\begin{aligned} &\sum_k a_P^{pu^k} u_P^k + \sum_k a_P^{pv^k} v_P^k + a_P^{pp} p_P + \sum_k \sum_{F=\mathrm{NB}(P)} a_F^{pu^k} u_F^k + \sum_k \sum_{F=\mathrm{NB}(P)} a_F^{pv^k} v_F^k \\ &\quad + \sum_{F=\mathrm{NB}(P)} a_F^{pp} p_F = b_P^p \end{aligned} \tag{39}$$

with NB(P) denoting the neighbors of the P grid point and the coefficients given by

$$a_P^{pu^k} = \sum_{f=\mathrm{nb}(P)} \rho_f^k \left(\alpha_f^k \right)^* r_p S_f^x \quad a_F^{pu^k} = \rho_f^k \left(\alpha_f^k \right)^* (1 - r_p) S_f^x \tag{40a}$$

$$a_P^{pv^k} = \sum_{f=\mathrm{nb}(P)} \rho_f^k \left(\alpha_f^k \right)^* r_p S_f^y \quad a_F^{pv^k} = \rho_f^k \left(\alpha_f^k \right)^* (1 - r_p) S_f^y \tag{40b}$$

$$a_P^{pp} = \sum_k \sum_{f=\mathrm{nb}(P)} \rho_f^k \left[\left(\alpha_f^k \right)^* \right]^2 \frac{\bar{D}_f^{u^k} E_f^x + \bar{D}_f^{v^k} E_f^y}{d_{PF}} \tag{40c}$$

$$a_F^{pp} = - \sum_k \rho_f^k \left[\left(\alpha_f^k \right)^* \right]^2 \frac{\bar{D}_f^{u^k} E_f^x + \bar{D}_f^{v^k} E_f^y}{d_{PF}} \tag{40d}$$

$$b_P^p = \sum_k \sum_{f=\mathrm{nb}(P)} \rho_f^k \left[\left(\alpha_f^k \right)^* \right]^2 \left[\left(\overline{\mathbf{D}}_f^{\mathbf{v}^k} \nabla p_f^* \right) \cdot \mathbf{T}_f - \left(\overline{\mathbf{D}}_f^{\mathbf{v}^k} \overline{\nabla p_f^*} \right) \cdot \mathbf{S}_f \right] \tag{40e}$$

When Eqs. (11) and (39) are gathered for every control volume of a two-phase-flow problem, a system of five equations results, with its solving matrix having the following form:

$$\begin{bmatrix} a_P^{u^k u^k} & a_P^{u^k v^k} & a_P^{u^k u^m} & 0 & a_P^{u^k p} \\ a_P^{v^k u^k} & a_P^{v^k v^k} & 0 & a_P^{v^k v^m} & a_P^{v^k p} \\ a_P^{u^m u^k} & 0 & a_P^{u^m u^m} & a_P^{u^m v^m} & a_P^{u^m p} \\ 0 & a_P^{u^k v^k} & a_P^{v^m u^m} & a_P^{v^m v^m} & a_P^{v^m p} \\ a_P^{pu^k} & a_P^{pv^k} & a_P^{pu^m} & a_P^{pv^m} & a_P^{pp} \end{bmatrix} \begin{bmatrix} u_P^k \\ v_P^k \\ u_P^m \\ v_P^m \\ p_P \end{bmatrix}$$

$$+ \sum_{F=\mathrm{NB}(P)} \begin{bmatrix} a_F^{u^k u^k} & a_F^{u^k v^k} & 0 & 0 & a_F^{u^k p} \\ a_F^{v^k u^k} & a_F^{v^k v^k} & 0 & 0 & a_F^{v^k p} \\ 0 & 0 & a_F^{u^m u^m} & a_F^{u_m v_m} & a_F^{u^m p} \\ 0 & 0 & a_F^{v^m u^m} & a_F^{v^m v^m} & a_F^{v^m p} \\ a_F^{pu^k} & a_F^{pv^k} & a_F^{pu^m} & a_F^{pv^m} & a_F^{pp} \end{bmatrix} \begin{bmatrix} u_F^k \\ v_F^k \\ u_F^m \\ v_F^m \\ p_F \end{bmatrix} = \begin{bmatrix} b_P^{u^k} \\ b_P^{v^k} \\ b_P^{u^m} \\ b_P^{v^m} \\ b_P^p \end{bmatrix} \tag{41}$$

These equations are assembled for the entire domain into a global matrix as

$$\begin{bmatrix} [] & [] & \cdot & [] & \cdot \\ \cdot & \cdot & & & \cdot \\ & & \cdot & \cdot & \\ \cdot & & \cdot & \cdot & \\ & \cdot & & & \cdot \end{bmatrix} \begin{bmatrix} \{\} \\ \cdot \\ \cdot \\ \cdot \\ \{\} \end{bmatrix} = \begin{bmatrix} \{\} \\ \cdot \\ \cdot \\ \cdot \\ \{\} \end{bmatrix} \tag{42}$$

where, for a two-phase, two-dimensional flow, each pair of small square brackets "[]" is a 5×5 matrix of coefficients and each pair of curly brackets "{}" is a 5×1 array of the variables or sources. The set of equations assembled in the global matrix displayed in Eq. (42) are solved simultaneously for all variables. The adopted algebraic solution algorithm is a combination of an ILU(0) [27] solver with an additive corrective multigrid method [28]. This combination is found to provide the high convergence rate of multigrid methods with the simplicity and low storage of the basic ILU algorithm. Additional details on the implementation of the algebraic solver can be found in [11].

RESULTS AND DISCUSSION

The performance of the coupled solver is assessed by solving eight one-dimensional, two-phase-flow problems and comparing results with similar ones obtained using the segregated MCBA-SIMPLE algorithm [18]. These problems can be categorized as: (i) horizontal particle transport [Figure 2*a*], and (ii) vertical particle transport [Figure 2*b*]. Results are presented in terms of convergence history plots and tabulated values of the number of iterations and CPU time needed to obtain a converged solution. For all problems and for both coupled and segregated algorithms, computations were terminated when the maximum normalized residuals over the

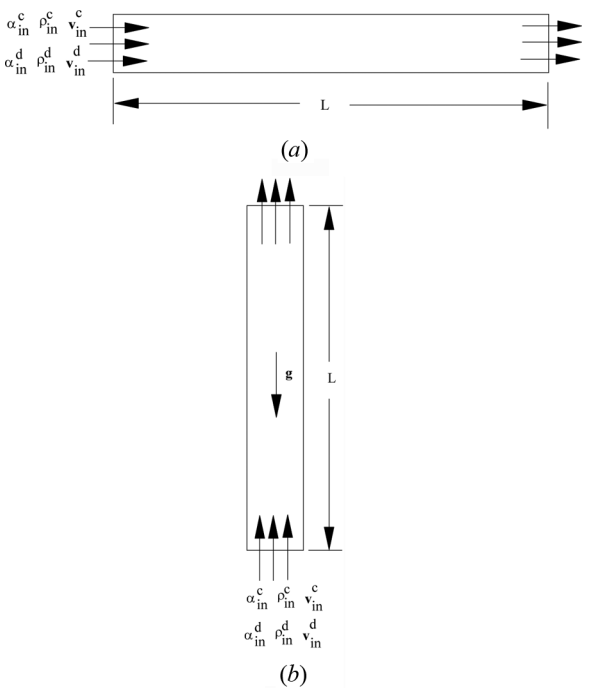

Figure 2. (*a*) Physical domain for horizontal flow. (*b*) Physical domain for vertical flow.

domain and for all phases dropped below a vanishing quantity ε_s. The maximum normalized residual (RES) is defined as

$$\mathrm{RES} = \max_{\substack{\text{over all elements} \\ \text{and over all phases}}} \frac{\left| a_P^{\phi^k} \phi_P + \sum_{\mathrm{NB}} a_{\mathrm{NB}}^{\phi^k} \phi_{NB} - b_P^{\phi^k} \right|}{a_P^{\phi^k} \phi_P} \tag{43}$$

Despite its geometric simplicity, the one-dimensional particle transport problem can represent a wide range of physical conditions. The effects of grid refinement on convergence are studied by solving the problems on three grid systems of sizes 10,000, 30,000, and 50,000, control volumes with ε_s assigned the value of 10^{-5}.

Horizontal Particle Transport

Figure 2*a* displays a schematic of the physical situation under consideration. The problem is solved for two values of the density ratio, representing either the steady flow of solid particles suspended in a free stream of air ($\rho^d/\rho^c = 2{,}000$) or the steady flow of air bubbles in a stream of water ($\rho^c/\rho^d = 1{,}000$). For each density ratio, results are generated for two values of the volume fraction, representing dilute and dense particle/bubble concentration in the suspension. The air/water and particle/bubble velocity profiles for the four cases considered are presented in

Figures 3*a*–3*d*. Reported results are obtained assuming that the interparticle/bubble forces along with diffusion within both phases are negligible. Drag between the two phases is the only force considered, and denoting the radius of the particles by r_p, it is calculated as

$$\mathbf{I}_M^c = \frac{3}{8}\frac{C_D}{r_p}\alpha^d \rho^c U_{\text{slip}}\left(\mathbf{v}^d - \mathbf{v}^c\right) \tag{44}$$

$$\mathbf{I}_M^d = -\frac{3}{8}\frac{C_D}{r_p}\alpha^d \rho^c U_{\text{slip}}\left(\mathbf{v}^d - \mathbf{v}^c\right) \tag{45}$$

$$U_{\text{slip}} = \left\|\mathbf{v}^d - \mathbf{v}^c\right\| \tag{46}$$

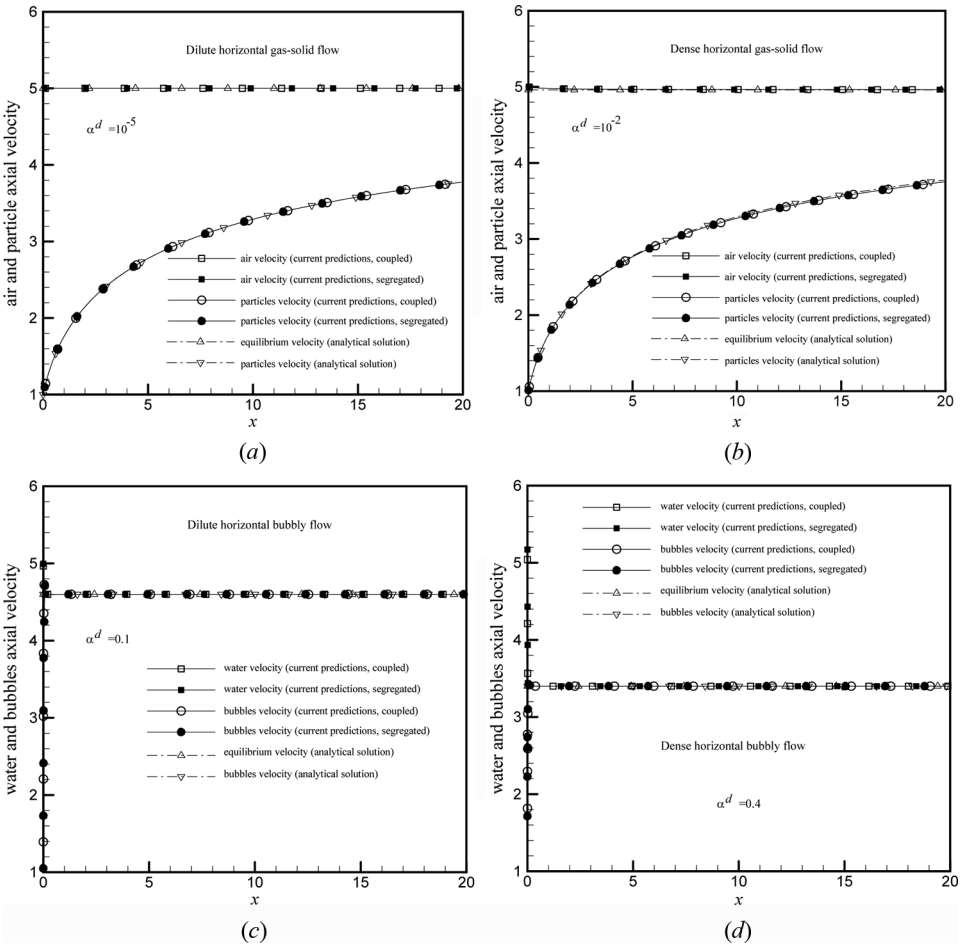

Figure 3. Axial velocity variation of (*a*) dilute air–particle, (*b*) dense air–particle, (*c*) dilute bubbly flow in a horizontal duct, and (*d*) dense bubbly flow in a horizontal duct.

The value of the drag coefficient, C_D, is set at 0.44. The task is to calculate the particle/bubble velocity distribution as a function of position. If the flow field is extended far enough (here $L = 20\,\text{m}$), the particle/bubble and fluid phases are expected to approach an equilibrium velocity given by

$$U_{\text{equilibrium}} = \alpha^c_{\text{inlet}} U^c_{\text{inlet}} + \alpha^d_{\text{inlet}} U^d_{\text{inlet}} \tag{47}$$

Problem 1: Dilute gas–solid flow. For this problem, the inlet air $\left(U^c_{\text{inlet}}\right)$ and particle $\left(U^d_{\text{inlet}}\right)$ velocities are set at 5 m/s and 1 m/s, respectively. The physical properties of the two phases are $\rho^d/\rho^c = 2{,}000$, $r_p = 1\,\text{mm}$, and $\alpha^d_{\text{inlet}} = 10^{-5}$. Due to the dilute concentration of the particles, the free-stream velocity is almost unaffected by their presence, and the equilibrium velocity is nearly equal to the inlet free-stream velocity. Based on this observation, Morsi and Alexander [29] obtained the following analytical solution for the particle velocity u^d as a function of the position x and the properties of the two phases:

$$\text{Ln}\left[U^c_{\text{inlet}} - u^d\right] + \frac{U^c_{\text{inlet}}}{U^c_{\text{inlet}} - u^d} = \frac{3}{8}\frac{\rho^c}{\rho^d}\frac{C_D}{r_p}x + \text{Ln}\left[U^c_{\text{inlet}} - U^d_{\text{inlet}}\right] + \frac{U^c_{\text{inlet}}}{U^c_{\text{inlet}} - U^d_{\text{inlet}}} \tag{48}$$

This case is of particular importance since the flow situation has an exact solution. As shown in Figure 3*a*, the predicted particle and air velocity distributions generated with both coupled and segregated solvers fall on top of the analytical solution given by Eq. (46), indicating a correct implementation of the numerical procedures.

A summary of the number of iterations, the CPU time, and their ratios for all grid sizes are presented in Table 1*a*. As depicted, the coupled solver requires a much lower number of iterations than the segregated solver. For the coupled solver the number increases from 54 iterations, for a grid with size of 10,000 control volumes, to 197 iterations for a grid system with size of 50,000 control volumes. For the segregated solver the number increases from 355 to 1,466 iterations, respectively. The ratio of the number of iterations required by the segregated solver to that required by the coupled algorithm varies between 7.4 and 9.8. The solution time required by the coupled solver increases from 355 to 7,188 s, while the computation time of the segregated solver increases from 654 to 9,856 s. The ratio of the CPU time required by the segregated solver to that required by the coupled algorithm varies between 1.4 and 1.8, indicating reduction in computation time with the coupled solver for all grid sizes used. This performance of the coupled solver is further demonstrated

Table 1*a*. Comparison of the number of iterations and CPU time required for convergence by the coupled and segregated solvers for horizontal dilute gas–solid flow

	Coupled solver (C)		Segregated solver (S)		S/C	
Mesh	Iterations	Time (s)	Iterations	Time (s)	Iterations	Time
10,000	51	355	501	654	9.8	1.8
30,000	138	3,061	959	3,890	6.9	1.3
50,000	197	7,188	1,466	9,856	7.4	1.4

by the residual history plots presented in Figures 4*a*–4*d*. These plots compare the convergence of the newly developed coupled solver with the ones obtained using the segregated MCBA-SIMPLE solver. Figures 4*a* and 4*b* are for a grid with size of 10,000 control volumes, while Figures 4*c* and 4*d* are for a grid with size of 50,000 elements. The higher rate of convergence per iteration obtained with the coupled solver can easily be inferred from the plots.

Problem 2: Dense gas–solid flow. The only difference between this case and the previous one is in the concentration of particles, which is set at $\alpha^d_{\text{inlet}} = 10^{-2}$. For this value of the particle volume fraction, the ratio of disperse–phase and continuous-phase mass loadings is large, $\alpha^d\rho^d/\alpha^c\rho^c = 20$, indicating that the particle phase carries most of the inertia of the mixture. The equilibrium velocity in this case, as obtained from Eq. (47) is 4.96 m/s, which is still close to the inlet air velocity.

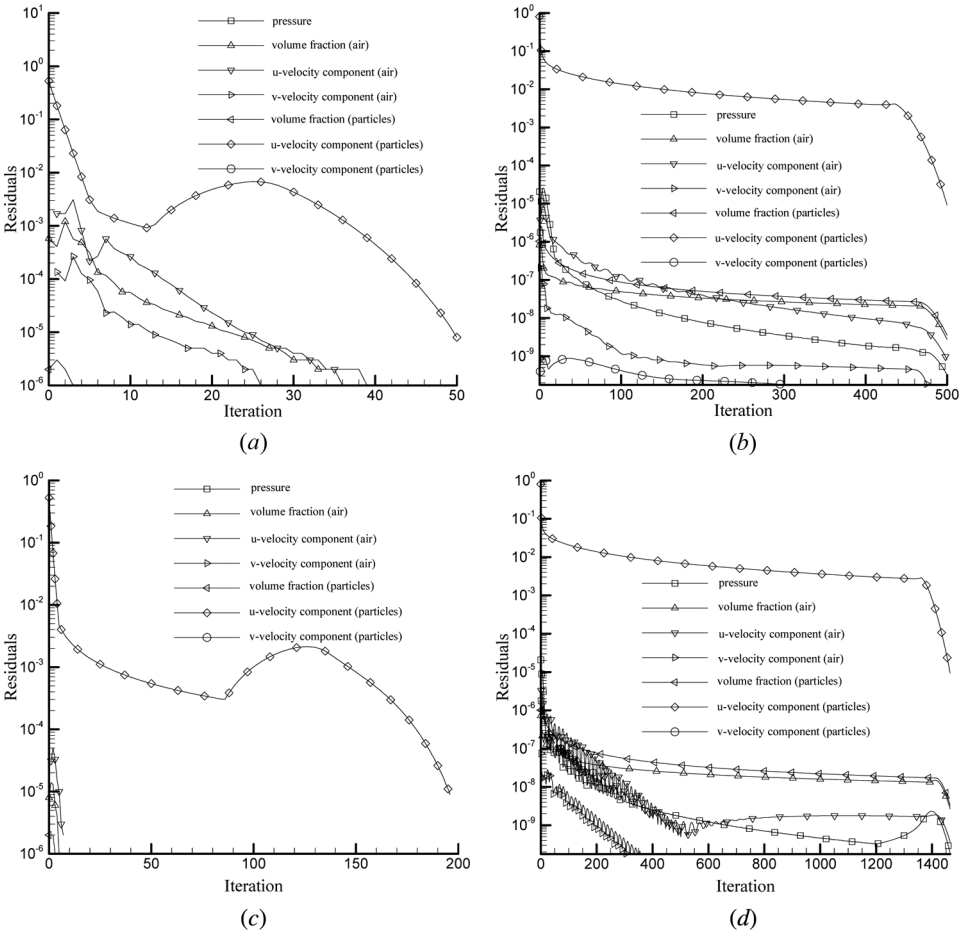

Figure 4. Comparison of residual history plots for grid systems with sizes of 10,000 [(*a*) and (*b*)] and 50,000 [(*c*) and (*d*)] control volumes obtained by the coupled [(*a*) and (*c*)] and segregated [(*b*) and (*d*)] algorithms for the dilute particle flow in a horizontal duct problem.

Therefore the variation in particle velocity can be obtained again from Eq. (48). The predicted air and particle velocity distributions are displayed in Figure 3*b*. The numerical (coupled and segregated) and analytical particle velocity profiles are indistinguishable and fall on top of each other. Moreover, the slight decrease in the air velocity can be easily depicted.

The number of iterations, CPU time, and their ratios that are presented in Table 1*b* in addition to the convergence history plots displayed in Figures 5*a*–5*d*, indicate behavior similar to that of the dilute case, with the acceleration rates for the different grid sizes being respectively equal.

Problem 3: Dilute bubbly flow. For this problem, the continuous phase is considered to be water and the disperse phase to be air. The resulting flow is denoted in the literature by bubbly flow. The differences in physical properties between this case and the previous ones are only in the values of the density ratio (ρ^d/ρ^c) and the disperse-phase volume fraction at the inlet $\left(\alpha^d_{\text{inlet}}\right)$, which are set at 10^{-3} and 0.1, respectively. This is a strongly coupled problem and represents a good test for the coupled solution algorithm. The correct physical solution is that the bubble and continuous-phase velocities both reach the equilibrium velocity of 4.6 m/s [Eq. (47)] in a very short distance. The axial velocity distribution for both water and air are displayed in Figure 3*c*. As expected, both phases reach the equilibrium velocity of 4.6 m/s over a very short distance from the inlet section and remain constant afterward.

Results displayed in Table 1*c* indicate that the number of iterations required by the coupled solver, varies between 35 and 39 as the grid size increases from 10,000 elements to 50,000 elements. For the segregated solver however, the number increases from 450 to 640 iterations, respectively, with the ratio of the number of iterations to convergence required by the segregated solver to that required by the coupled algorithm varying between 12.9 and 18.7 with the grid size. As compared to the previous cases, a larger reduction in computation time is obtained with the ratio of the time needed by the segregated solver to that needed by the coupled solver increasing from 2.3 to 3.6 as the grid size increases. This performance of the coupled algorithm is further revealed by the residuals' reduction rates, depicted in Figures 6*a*–6*d*. As shown, the rate of convergence of the coupled algorithm is much higher than the rate at which convergence is achieved with the segregated algorithm. It further reveals the higher stability and robustness of the coupled approach.

Problem 4: Dense bubbly flow. This problem is similar to the previous one except for the concentration of bubbles, which is set at $\alpha^d_{\text{inlet}} = 0.4$. Due to the high

Table 1*b*. Comparison of the number of iterations and CPU time required for convergence by the coupled and segregated solvers for horizontal dense gas–solid flow

	Coupled solver (C)		Segregated solver (S)		S/C	
Mesh	Iterations	Time (s)	Iterations	Time (s)	Iterations	Time
10,000	54	373	500	659	9.3	1.8
30,000	136	2,982	958	3,860	7.0	1.3
50,000	195	7,159	1,465	9,783	7.5	1.4

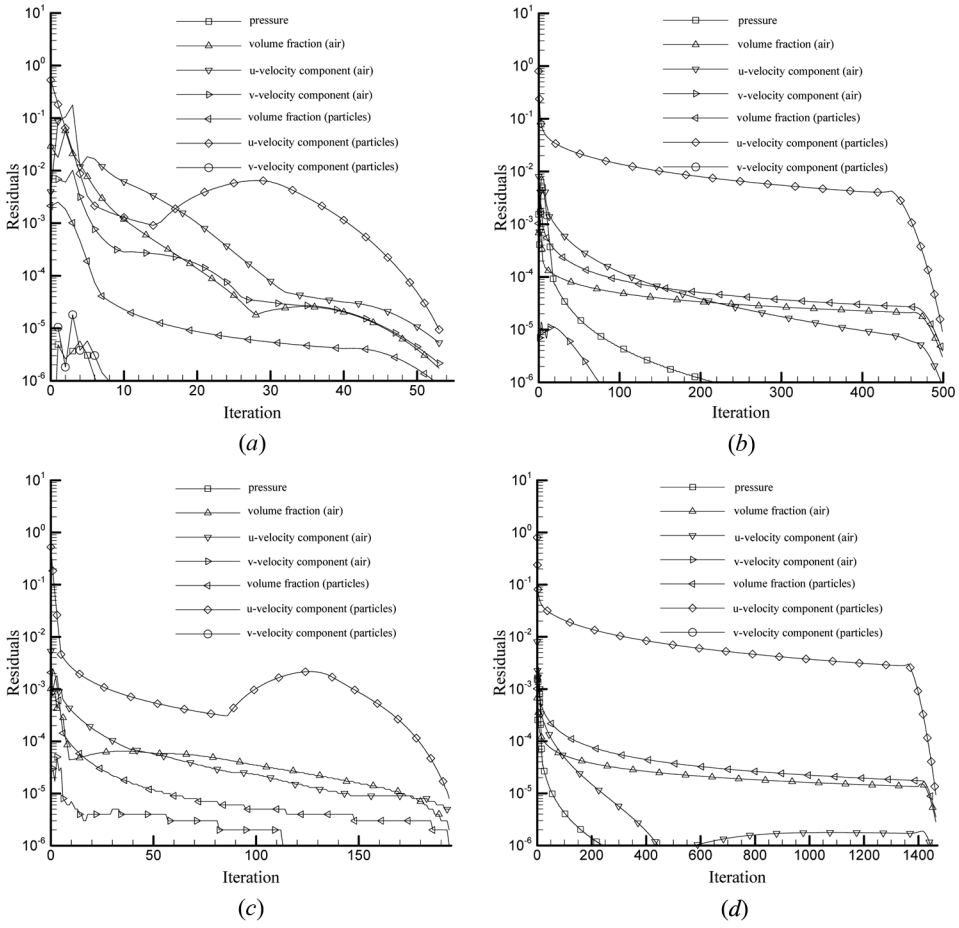

Figure 5. Comparison of residual history plots for grid systems with sizes of 10,000 [(*a*) and (*b*)] and 50,000 [(*c*) and (*d*)] control volumes obtained by the coupled [(*a*) and (*c*)] and segregated [(*b*) and (*d*)] algorithms for the dense particle flow in a horizontal duct problem.

value of void fraction, the likelihood of bubble coalescence is high. However, this is not accounted for here. The analytical solution is the same as in the previous case with the equilibrium velocity, as computed from Eq. (47), being 3.4 m/s. As depicted in Figure 3*d*, the equilibrium velocity obtained numerically is exact.

Table 1*c*. Comparison of the number of iterations and CPU time required for convergence by the coupled and segregated solvers for horizontal dilute bubbly flow

	Coupled solver (C)		Segregated solver (S)		S/C	
Mesh	Iterations	Time (s)	Iterations	Time (s)	Iterations	Time
10,000	35	262	450	593	12.9	2.3
30,000	34	843	637	2,581	18.7	3.1
50,000	39	1,361	640	4,593	16.4	3.6

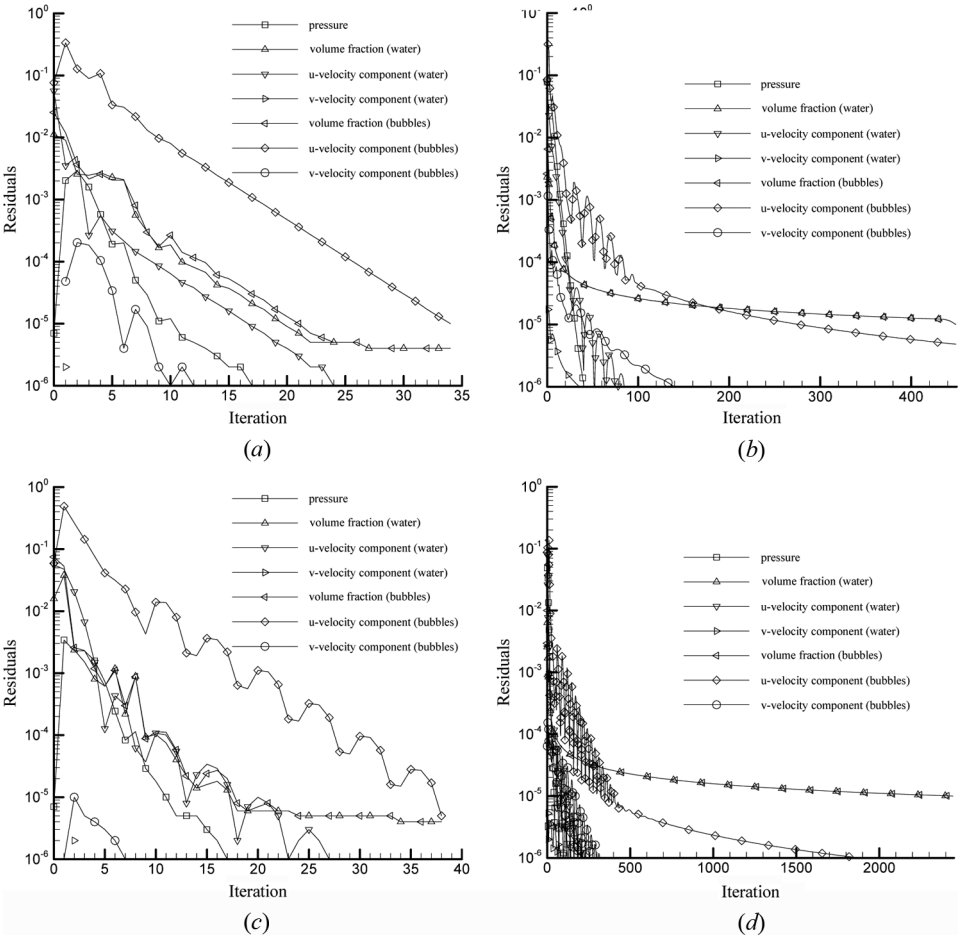

Figure 6. Comparison of residual history plots for grid systems with sizes of 10,000 [(*a*) and (*b*)] and 50,000 [(*c*) and (*d*)] control volumes obtained by the coupled [(*a*) and (*c*)] and segregated [(*b*) and (*d*)] algorithms for the dilute bubbly flow in a horizontal duct problem.

The number of iterations, computational time, and convergence history plots for the problem are presented in Table 1*d* and Figures 7*a*–7*d*, which show that the required computational effort for the problem is higher than in the previous case.

Table 1*d*. Comparison of the number of iterations and CPU time required for convergence by the coupled and segregated solvers for horizontal dense bubbly flow

	Coupled solver (C)		Segregated solver (S)		S/C	
Mesh	Iterations	Time (s)	Iterations	Time (s)	Iterations	Time
10,000	49	282	510	671	10.4	2.4
30,000	51	899	959	3,891	18.8	4.3
50,000	93	3,411	1,445	9,799	15.5	2.9

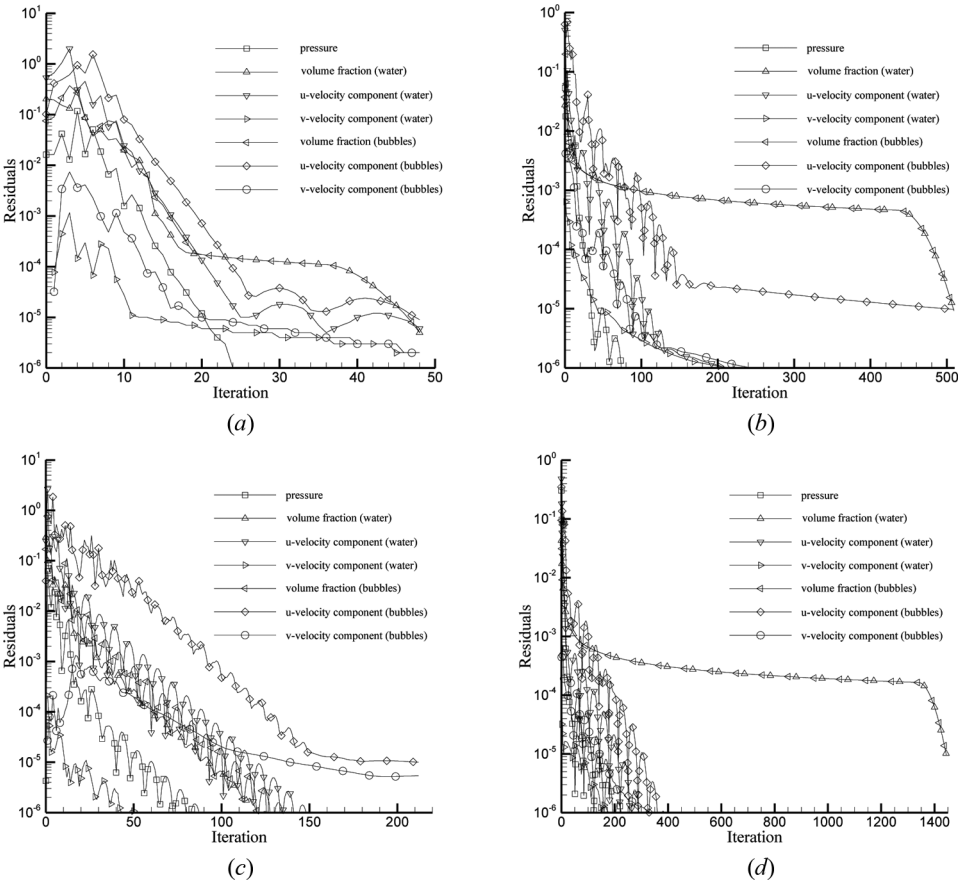

Figure 7. Comparison of residual history plots for grid systems with sizes of 10,000 [(*a*) and (*b*)] and 50,000 [(*c*) and (*d*)] control volumes obtained by the coupled [(*a*) and (*c*)] and segregated [(*b*) and (*d*)] algorithms for the dense bubbly flow in a horizontal duct problem.

The number of iterations for the coupled solver varies between 49 and 93 iterations, whereas for the segregated algorithm it varies between 510 and 1445 iterations. Therefore, as for the previous cases, converged solutions are achieved at lower computation cost with the coupled algorithm. The ratio of the time needed by the segregated solver to that needed by the coupled solver varies between 2.4 and 4.3 as the grid size varies between 10,000 and 50,000 control volumes. This is further demonstrated by the convergence history plots displayed in Figures 7*a*–7*d*.

Vertical Particle Transport

As shown in Figure 2*b*, for the same geometric parameters used above, the pipe is placed in a vertical direction and the gravitational acceleration is assigned a value of $g = 10\,\mathrm{m/s^2}$ in the negative y direction. The interparticle/bubble forces are again neglected, but diffusion in the continuous phase is retained. Moreover, the interphase

drag force is calculated using Eqs. (44–46), and the drag coefficient, C_D, is considered to be particle Reynolds number–dependent and calculated as

$$C_D = \frac{24}{\mathrm{Re}_p} + 0.44 \quad \mathrm{Re}_p = \frac{2r_p U_{\mathrm{slip}}}{\upsilon^c} \tag{49}$$

As for horizontal particle transport, the problem is solved for four different combinations of the physical quantities using three grid systems with sizes of 10,000, 30,000, and 50,000 control volumes. The generated air/water and particle/bubble velocity profiles are presented in Figures 8*a*–8*d*.

Problem 5: Dilute gas–solid flow. The density ratio (ρ^d/ρ^c) is set at 10^3, the kinematic viscosity (υ^c) at 10^{-3}, the particle diameter (r_p) at 1 mm, the disperse-phase inlet volume fraction $\left(\alpha^d_{\mathrm{inlet}}\right)$ at 10^{-6}, and the inlet air $\left(U^c_{\mathrm{inlet}}\right)$ and particle $\left(U^d_{\mathrm{inlet}}\right)$

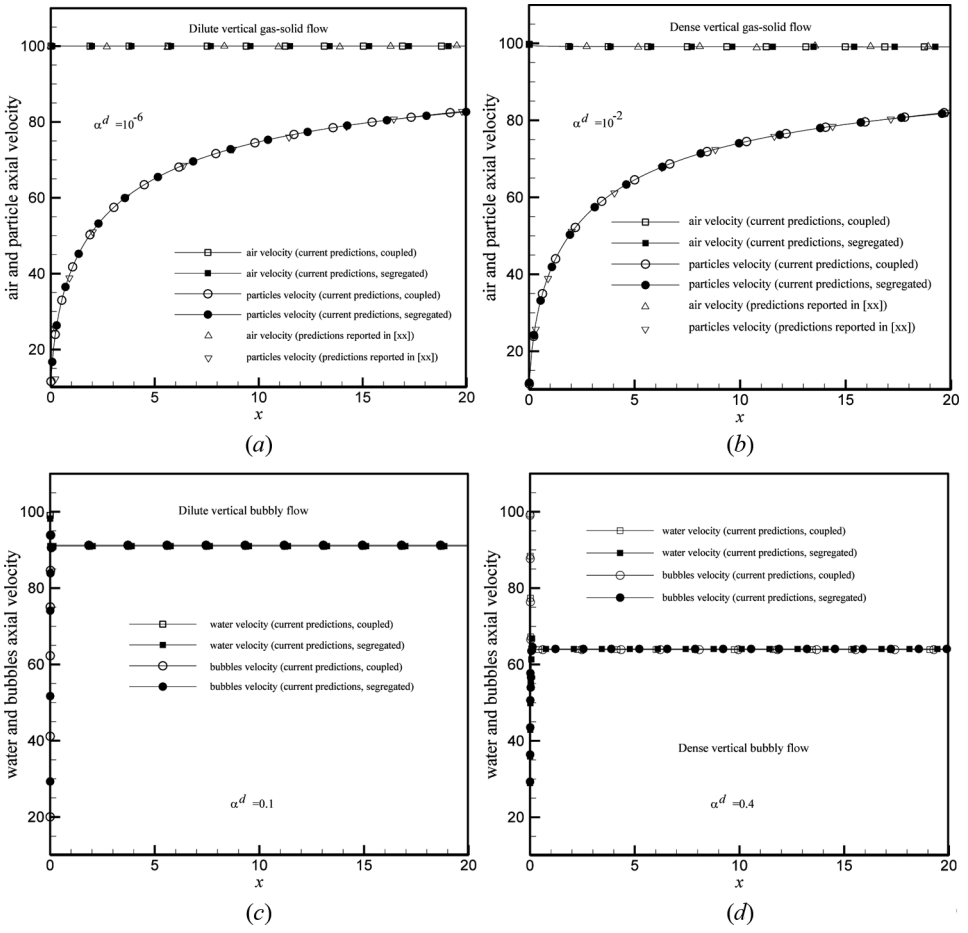

Figure 8. Axial velocity variation of (*a*) dilute air-particle, (*b*) dense air–particle, (*c*) dilute bubbly, and (*d*) dense bubbly flow in a vertical duct.

velocities at 100 m/s and 10 m/s, respectively. The large velocity boundary condition is used to ensure that the solid phase does not exit the inlet. As depicted in Figure 8*a*, the predicted air and particle velocity distributions using both the coupled and segregated solution procedures are on top of each other and in excellent agreement with similar predictions reported in [30].

The number of iterations and computation cost for all grid sizes are presented in Table 2*a*. As shown, the number of iterations required by the coupled solver is much lower than the number required by the segregated solver. For the coupled algorithm it increases from 41 to 144 iterations, as the grid size increases from 10,000 to 50,000 control volumes. For the segregated solver the number of iterations increases from 465 to 1,397 iterations. The ratio of the number of iterations required by the segregated solver to that required by the coupled algorithm varies between 9.2 and 11.3. The CPU time consumed by the coupled solver increases from 283 to 5,357 s, while it increases from 588 to 9,198 s with the segregated solver. The coupled solver requires less computational time compared to the segregated solver over all grid sizes, with the ratio of times (S/C) varying between 1.6 and 2.1. This superiority in performance of the coupled solver is further demonstrated by the residual history plots presented in Figures 9*a*–9*d*, which clearly show the higher rate of convergence per iteration.

Problem 6: Dense gas–solid flows. All physical and material properties are similar to the previous case with the exception of the particles' volume fraction $\left(\alpha_{\text{inlet}}^{(d)}\right)$, which is set at 10^{-2}. The predicted coupled and segregated air and particle velocity profiles displayed in Figure 8*b* fall on top of those reported by Baghdadi [26]. The number of iterations and CPU times (Table 2*b*) required by both the coupled and segregated solvers over all grid systems are almost the same as those obtained for the dilute case. This is also true for the rate of acceleration obtained with the coupled solver. The convergence history plots displayed in Figures 10*a*–10*d* show also a behavior similar to that of the dilute case.

Problem 7: Dilute bubbly flows. Considering the continuous phase to be water and the disperse phase to be air, the problem is solved for a density ratio (ρ^d/ρ^c) of 10^{-3}, inlet water and air velocities of 100 m/s and 10 m/s, respectively, and an inlet void fraction of 0.1.

As depicted in Figure 8*c*, the continuous- and disperse-phase velocity profiles obtained using the segregated and coupled solvers fall on top of each other. The number of iterations and CPU time for both algorithms are compared in Table 2*c*

Table 2*a*. Comparison of the number of iterations and CPU time required for convergence by the coupled and segregated solvers for vertical dilute gas–solid flow

	Coupled solver (C)		Segregated solver (S)		S/C	
Mesh	Iterations	Time (s)	Iterations	Time (s)	Iterations	Time
10,000	41	283	465	588	11.3	2.1
30,000	99	2,182	906	3,595	9.2	1.6
50,000	144	5,357	1,397	9,198	9.7	1.7

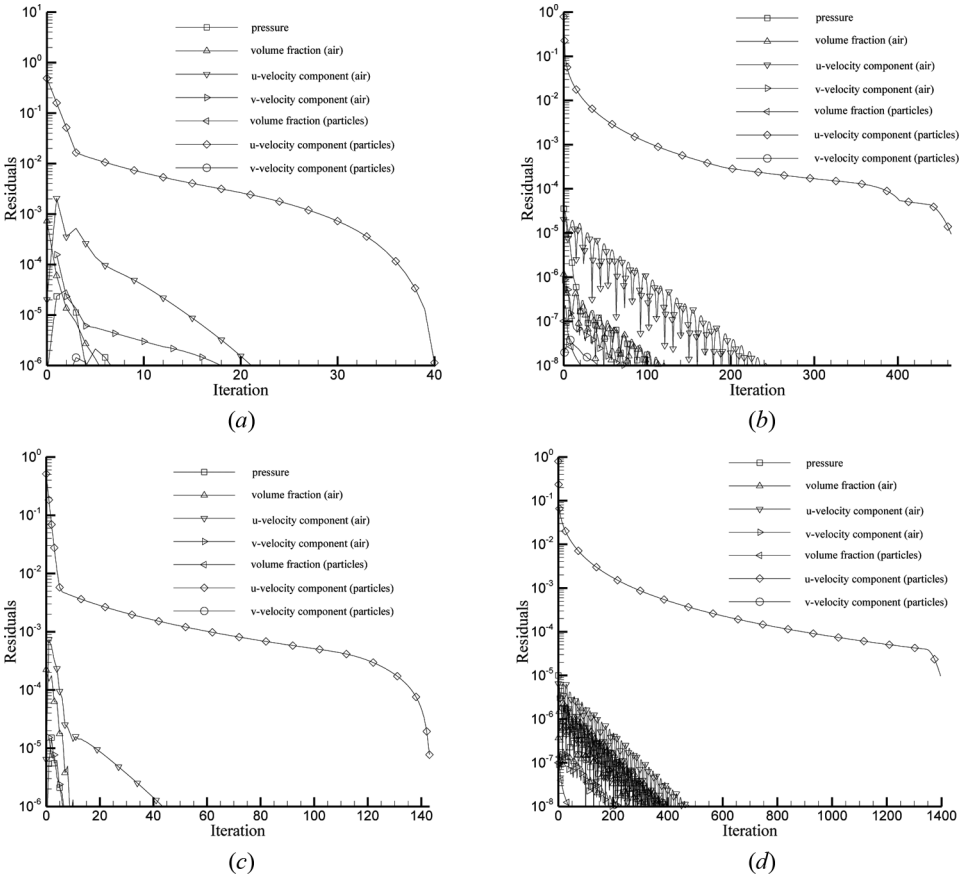

Figure 9. Comparison of residual history plots for grid systems with sizes of 10,000 [(*a*) and (*b*)] and 50,000 [(*c*) and (*d*)] control volumes obtained by the coupled [(*a*) and (*c*)] and segregated [(*b*) and (*d*)] algorithms for the dilute particle flow in a vertical duct problem.

for the various grid systems used. The number of iterations varies between 33 and 59 for the coupled solver and between 476 and 1,400 for the segregated solver as the grid size increases from 10,000 to 50,000 control volumes. The time required by the coupled and segregated solvers increases from 196 to 2,070 s and from 604 to 9,454 s, respectively, as the mesh size increases. This indicates that the ratio of the

Table 2*b*. Comparison of the number of iterations and CPU time required for convergence by the coupled and segregated solvers for vertical dense gas–solid flow

	Coupled solver (C)		Segregated solver (S)		S/C	
Mesh	Iterations	Time (s)	Iterations	Time (s)	Iterations	Time
10,000	41	286	468	613	11.4	2.1
30,000	99	2,167	912	3,691	9.2	1.7
50,000	145	5,330	1,396	9,462	9.6	1.8

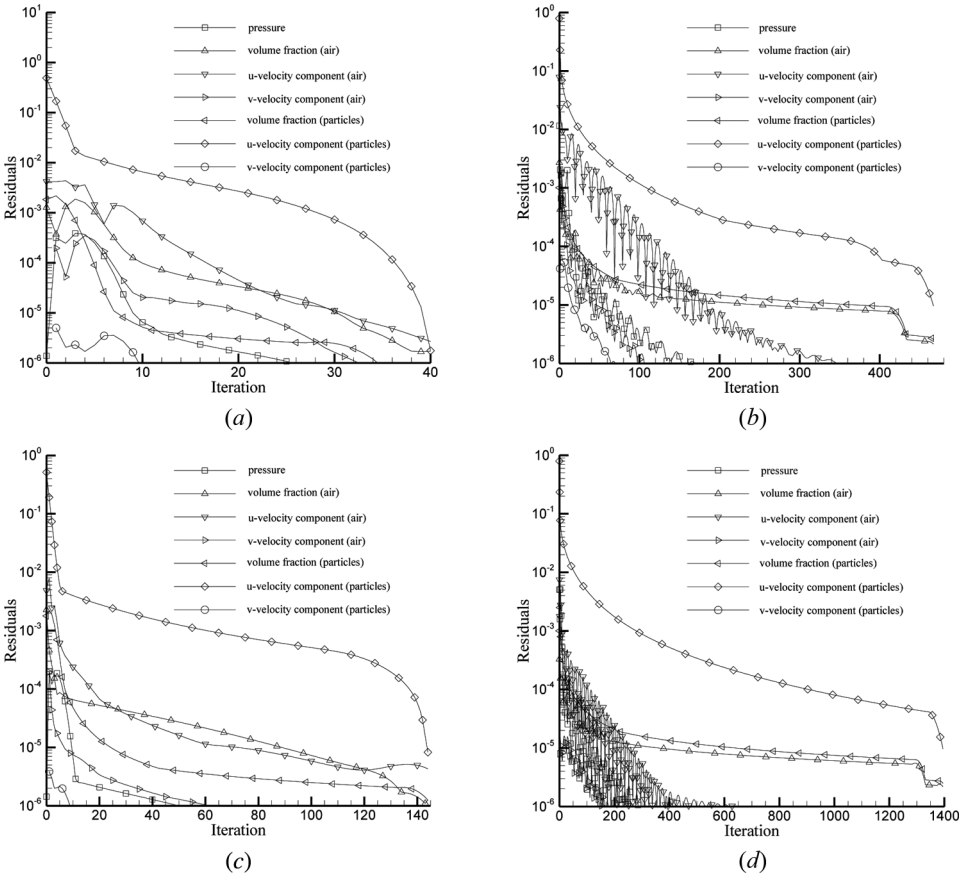

Figure 10. Comparison of residual history plots for grid systems with sizes of 10,000 [(*a*) and (*b*)] and 50,000 [(*c*) and (*d*)] control volumes obtained by the coupled [(*a*) and (*c*)] and segregated [(*b*) and (*d*)] algorithms for the dense particle flow in a vertical duct problem.

time required by the segregated solver to that needed by the coupled algorithm (S/C) varies between 3.1 and 4.6. Therefore a substantial reduction in computation cost is achieved with the coupled solver. These savings can also be inferred from the convergence history plots displayed in Figures 11*a*–11*d*, which also demonstrate the higher stability and robustness of the newly developed coupled algorithm.

Table 2*c*. Comparison of the number of iterations and CPU time required for convergence by the coupled and segregated solvers for vertical dilute bubbly flow

	Coupled solver (C)		Segregated solver (S)		S/C	
Mesh	Iterations	Time (s)	Iterations	Time (s)	Iterations	Time
10,000	33	196	476	604	14.4	3.1
30,000	53	1,087	914	3,758	17.2	3.5
50,000	59	2,070	1,400	9,454	23.7	4.6

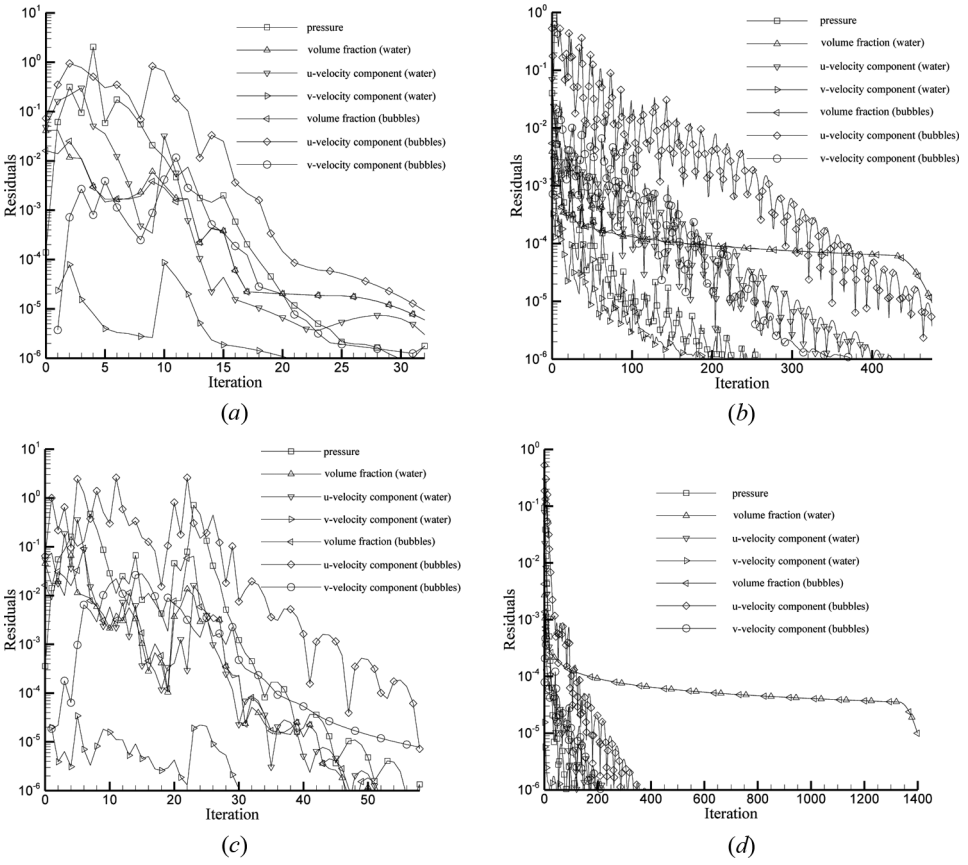

Figure 11. Comparison of residual history plots for grid systems with sizes of 10,000 [(*a*) and (*b*)] and 50,000 [(*c*) and (*d*)] control volumes obtained by the coupled [(*a*) and (*c*)] and segregated [(*b*) and (*d*)] algorithms for the dilute bubbly flow in a vertical duct problem.

Problem 8: Dense bubbly flows. The only difference between this problem and the previous one is in the value of the void fraction $\left(\alpha^{d}_{\text{inlet}}\right)$ which is assigned the value 0.4. Results for the problem are generated using both the coupled and MCBA-SIMPLE algorithm over the three grid systems with sizes of 10,000, 30,000, and 50,000 elements. The predicted liquid and gas velocity distributions obtained via the coupled and segregated algorithms [Figure 8*d*], are on top of each other. As shown in Table 2*d*, a larger number of iterations (especially at higher grid sizes) is required as compared to the dilute case. The coupled solution requires 35 iterations on a grid with size of 10,000 control volumes. This number increases to 128 iterations and then 175 iterations as the grid size increases to 30,000 and 50,000 elements, respectively. The corresponding numbers of iterations needed by the segregated solver are 477, 904, and 1,389, respectively. The CPU time consumed by the coupled solver increases from 266 s, to 2,150 s, to 6,418 s as the grid size increases. The corresponding times required by the segregated solver are 627 s, 3,838 s, and 9,277 s, respectively. Reduction in the computational effort is achieved with the coupled solver on all grid

Table 2*d*. Comparison of the number of iterations and CPU time required for convergence by the coupled and segregated solvers for vertical dense bubbly flow

	Coupled solver (C)		Segregated solver (S)		S/C	
Mesh	Iterations	Time (s)	Iterations	Time (s)	Iterations	Time
10,000	35	266	477	627	13.6	2.4
30,000	128	2,150	904	3,838	7.1	1.8
50,000	175	6,418	1,389	9,277	7.9	1.4

sizes, with the ratio of the time required by the segregated algorithm to that needed by the coupled solver (S/C) varying between 1.4 and 2.4. This behavior is further revealed by the convergence history plots reported in Figures 12*a*–12*d*, which clearly show the higher rate of convergence achieved by the coupled solution procedure.

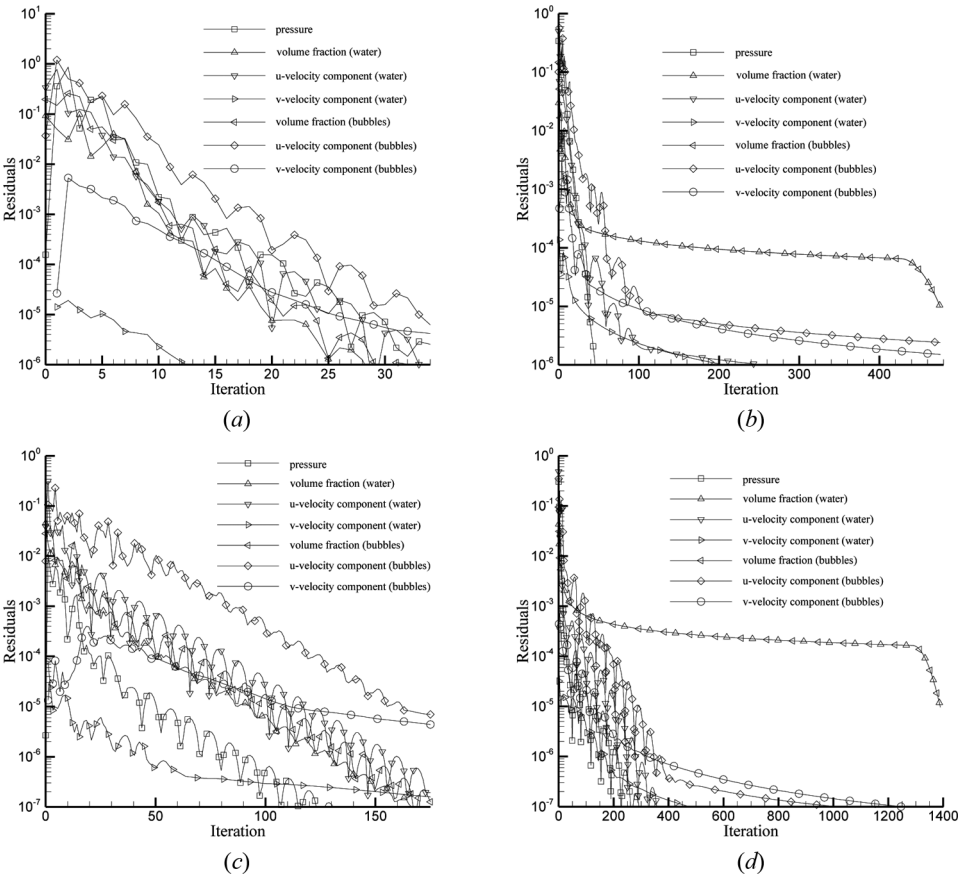

Figure 12. Comparison of residual history plots for grid systems with sizes of 10,000 [(*a*) and (*b*)] and 50,000 [(*c*) and (*d*)] control volumes obtained by the coupled [(*a*) and (*c*)] and segregated [(*b*) and (*d*)] algorithms for the dense bubbly flow in a vertical duct problem.

CLOSING REMARKS

A coupled pressure-based algorithm for the solution of disperse two-phase flows was developed. The solver was implemented within a collocated finite-volume method and used to solve eight one-dimensional, two-phase-flow problems spanning the spectrum from dilute bubbly to dense gas–solid flows. The coupled solver was demonstrated to accelerate convergence over all grid sizes employed. When compared to MCBA-SIMPLE, the rate of solution acceleration was found to vary between 1.3 and 4.6.

All test cases used in this article were one-dimensional. Therefore, future work will concentrate on testing the method for multidimensional two-phase-flow problems in order to assess the ability of the coupled solver to solve such problems. Moreover, the phases in the problems presented were all incompressible. Future work will also concentrate on extending the capabilities of the coupled solver to simulate compressible multiphase flows. Furthermore, the MCBA approach was followed to derive the pressure equation. This caused the volume-fraction field to remain decoupled from the pressure and velocity fields. Work on developing a fully coupled multiphase-flow solver is still needed.

REFERENCES

1. S. V. Patankar and D. B. Spalding, A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows, *Int. J. Heat Mass Transfer*, vol. 15, pp. 1787–1806, 1972.
2. F. Moukalled and M. Darwish, A Unified Formulation of the Segregated Class of Algorithms for Fluid Flow at All Speeds, *Numer. Heat Transfer B*, vol. 37, pp. 103–139, 2000.
3. M. Darwish, D. Asmar, and F. Moukalled, A Comparative Assessment within a Multigrid Environment of Segregated Pressure-Based Algorithms for Fluid Flow at all Speeds, *Numer. Heat Transfer B*, vol. 45, pp. 49–74, 2004.
4. A. Rizzi and L. E. Eriksson, Computation of Inviscid Incompressible Flow with Rotation, *J. Fluid Mech.*, vol. 153, pp. 275–312, 1985.
5. F. S. Lien and M. A. Leschziner, A General Non-orthogonal Collocated Finite Volume Algorithm for Turbulent Flow at All Speeds Incorporating Second-Moment Turbulence-Transport Closure, Part 1: Computational Implementation, *Comput. Meth. Appl. Mech. Eng.*, vol. 114, pp. 123–148, 1994.
6. S. V. Patankar, *Numerical Heat Transfer and Fluid Flow*, Hemisphere, New Yorks, 1981.
7. C. M. Rhie and W. L. Chow, A Numerical Study of the Turbulent Flow past an Isolated Airfoil with Trailing Edge Separation, *AIAA J.*, vol. 21, pp. 1525–1532, 1983.
8. L. S. Caretto, R. M. Curr, and D. B. Spalding, Two Numerical Methods for Three-Dimensional Boundary Layers, *Comput. Meth. Appl. Mech. Eng.*, vol. 1, pp. 39–57, 1972.
9. R. Webster, An Algebraic Multigrid Solver for Navier-Stokes Problems in the Discrete Second-Order Approximation, *Int. J. Numer. Meth. Fluids*, vol. 22, pp. 1103–1123, 1996.
10. M. Darwish, I. Sraj, and F. Moukalled, A Coupled Incompressible Flow Solver on Structured Grids, *Numer. Heat Transfer B*, vol. 52, pp. 353–371, 2007.
11. M. Darwish, I. Sraj, and F. Moukalled, A Coupled Incompressible Flow Solver for the Solution of Incompressible Flows on Unstructured Grids, *J. Comput. Phys.*, vol. 228, pp. 180–201, 2009.
12. J. P. Van Doormal and G. D. Raithby, An Evaluation of the Segregated Approach for Predicting Incompressible Fluid Flows, ASME Paper 85-HT-9, National Heat Transfer Conference, Denver, CO, 1985.

13. S. P. Vanka, Block-Implicit Multigrid Solution of Navier-Stokes Equations in Primitive Variables, *J. Comput. Phys.*, vol. 65, pp. 138–158, 1986.
14. K. C. Karki and H. C. Mongia, Evaluation of a Coupled Solution Approach for Fluid Flow Calculations in Body Fitted Co-ordinates, *Int. J. Numer. Meth. Fluids*, vol. 11, pp. 1–20, 1990.
15. M. E. Braaten, Development and Evaluation of Iterative and Direct Methods for the Solution of the Equations Governing Recirculating Flows, Ph.D. thesis, University of Minnesota, Minneapolis, MN, 1985.
16. Z. Mazhar, A Procedure for the Treatment of the Velocity-Pressure Coupling Problem in Incompressible Fluid Flow, *Numer. Heat Transfer B*, vol. 39, pp. 91–100, 2001.
17. M. Darwish, F. Moukalled, and B. Sekar, A Unified Formulation of the Segregated Class of Algorithms for Multi-fluid Flow at All Speeds, *Numer. Heat Transfer B*, vol. 40, pp. 99–137, 2001.
18. F. Moukalled and M. Darwish, A Comparative Assessment of the Performance of Mass Conservation Based Algorithms for Incompressible Multi-phase flows, *Numer. Heat Transfer B*, vol. 42, pp. 259–283, 2002.
19. F. Moukalled, M. Darwish, and B. Sekar, A Robust Multi-grid Algorithm for Multifluid Flow at All Speeds, *Int. J. Numer. Meth. Fluids*, vol. 41, pp. 1221–1251, 2003.
20. F. Moukalled, M. Darwish, and B. Sekar, A High Resolution Pressure-Based Algorithm for Multi-phase Flow at All Speeds, *J. Comput. Phys.*, vol. 190, pp. 550–571, 2003.
21. F. Moukalled and M. Darwish, The Performance of Geometric Conservation Based Algorithms for Incompressible Multi-fluid Flow, *Numer. Heat Transfer B*, vol. 45, pp. 343–368, 2004.
22. F. Moukalled and M. Darwish, Pressure-Based Algorithms for Multifluid Flow At All Speeds—Part I: Mass Conservation Formulation, *Numer. Heat Transfer B*, vol. 45, pp. 495–522, 2004.
23. F. Moukalled and M. Darwish, Pressure-Based Algorithms for Multifluid Flow at All Speeds—Part II: Geometric Conservation Formulation, *Numer. Heat Transfer B*, vol. 45, pp. 523–540, 2004.
24. S. G. Rubin and P. K. Khosla, Polynomial Interpolation Method for Viscous Flow Calculations, *J. Comput. Phys.*, vol. 27, pp. 153–168, 1982.
25. F. Moukalled, A. Abdel Aziz, and M. Darwish, Performance Comparison of the NWF, and DC Methods for Implementing High-Resolution Schemes in a Fully Coupled Incompressible Flow Solver, *Appl. Math. Comput.*, vol. 217, pp. 5041–5054, 2011.
26. F. Moukalled and M. Darwish, Pressure Based Algorithms for Single and Multifluid Flows, in W. J. Minkowycz, E. M. Sparrow, and J. Y. Murthy (eds.), *Handbook of Numerical Heat Transfer*, 2nd ed., pp. 325–367, John Wiley, New York, 2006.
27. C. Pommerell and W. Fichtner, Memory Aspects and Performance of Iterative Solvers, *SIAM J. Sci. Stat. Comput.*, vol. 15, pp. 460–473, 1994.
28. B. R. Hutchinson and G. D. Raithby, A Multigrid Method Based on the Additive Correction Strategy, *Numer. Heat Transfer*, vol. 9, pp. 511–537, 1986.
29. S. A. Morsi and A. J. Alexander, An Investigation of Particle Trajectories in Two-Phase Flow Ststem, *J. Fluid Mech.*, vol. 55, part 2, pp. 193–208, 1972.
30. A. H. A. Baghdadi, Numerical Modelling of Two-Phase Flow with Inter-phase Slip, Ph.D. thesis, Imperial College, University of London, UK, 1979.