Acta Mech
https://doi.org/10.1007/s00707-026-04653-1

ORIGINAL PAPER

M. Moriche · M. García-Villalba · M. Uhlmann

Particle-resolved simulations of settling particles: a methodology for long time-integration intervals

Received: 21 October 2025 / Revised: 28 January 2026 / Accepted: 2 February 2026

Abstract We present a methodology for simulating dilute suspensions of particles settling under gravity, with the main purpose of overcoming limitations of triply periodic configurations, mainly the strong vertical correlation that hinders the study of cluster dynamics. The current approach removes vertical periodicity and employs a moving reference frame, enabling efficient simulations of both single- and many-particle cases. We illustrate the method with two examples of increasing complexity: a single particle in the steady vertical regime and a many-particle case at a parametric point where collective effects were previously observed and recovered here. A converged, free-of-corrections time interval of approximately 600 D/U_g is simulated in the many-particle case, representing the first simulation of this kind to date. New physical insights can be explored thanks to this new configuration, for example, the effect of still fluid on the first layer of particles encountered by the fluid, or the turbulent character of the flow after a swarm of particles has passed by. Finally, the method only requires parameter tuning, allowing implementation within existing solvers without changes to their core formulation: for a standard configuration with an imposed free stream velocity at the inlet, only the input velocity (or the viscosity of the fluid) and the time step need to be updated.

1 Introduction

Particle-laden flows, in which solid particles interact with a surrounding fluid, play critical roles in both natural and industrial systems [15,19]. Such flows are challenging because both experiments and simulations are hard to carry out. Experimental characterization requires high-resolution three-dimensional tracking of particle positions and orientations in opaque suspensions [24]. Numerical methods using particle-resolved direct numerical simulations (PR-DNS) face also a complex task. They must capture detailed hydrodynamic interactions at the scale of individual particles, while also accounting for large-scale collective behavior [6,21]. This multiscale challenge requires enormous computational resources. Even state-of-the-art simulations in dilute systems typically handle only $O(10^4)$ particles at moderate Reynolds numbers, limiting statistical significance.

One particular problem of interest in this field involves gravity-driven, particle-laden flows in the dilute regime, where collective effects significantly modify the hydrodynamic behavior compared to isolated particles. These collective effects manifest themselves as spatial heterogeneities in particle concentration, with experimental observations revealing regions of higher particle concentration (clusters) and lower (voids) relative to the mixture's average concentration [8]. The onset of clustering fundamentally alters the flow dynamics, resulting in enhanced settling velocities, increased collision frequencies, and amplified fluid perturbations, among other modified transport properties.

Numerical simulations of such systems are typically conducted in triply periodic configurations, analyzing particle behavior in the absence of walls or other geometric constraints beyond the fluid–particle interfaces.

Institute of Fluid Mechanics and Heat Transfer, TU Wien, Vienna, Austria · Institute for Water and Environment, Karlsruhe Institute of Technology, Karlsruhe, Germany

Published online: 28 May 2026

This computational setup has yielded valuable insights into clustering phenomena for spherical particles [5,11,22], cubes [18], oblate spheroids [4,16], and prolate spheroids [10,14]. However, the triply periodic approach presents inherent limitations for studying cluster dynamics. As clusters grow to sizes comparable to the computational domain, the results become strongly correlated along the vertical direction. This constrains our ability to analyze cluster evolution and stability beyond their initial formation stages. To address these limitations, this work introduces a methodology that enables an efficient usage of inflow–outflow conditions, thereby removing the need for vertical periodicity. This non-homogeneous vertical setup overcomes the challenge that the terminal settling velocity of the particle ensemble is not known a priori. The method uses an iterative approach, adding corrections to determine this a-priori-unknown velocity, or what is equivalent, the Reynolds number of the particle ensemble. In this way, we enable long time-integration intervals in reasonably small computational domains and without the need to consider non-inertial effects in the governing equations.

The article is organized as follows. First, in §2 we describe in detail the problem under study, including the governing equations, the relevant non-dimensional parameters, and the definition of the reference frames that will be used throughout the paper. Then, in §3 we present the proposed methodology, where we explain how the iterative correction of the Reynolds number is implemented, together with the mapping operators and the details regarding the numerical integration. In §4 and §5, two representative examples are shown, first for the case of a single particle in the steady vertical regime, and later for the case of many particles where collective effects appear and the performance of the method is evaluated. Finally, the main conclusions and remarks are summarized in §6, where we also discuss the potential of the approach to be applied to other problems and the possible directions for future work.

2 Problem description

The problem under study is the settling of particles under the effect of gravity in an unbounded domain. The fluid is assumed to be Newtonian with density ρ_f and kinematic viscosity ν, and the particles are assumed to be rigid spheres with uniform density ρ_p, volume V and moment of inertia per unit mass $\boldsymbol{J}$. Note that the extension to non-spherical particles does not affect the methodology, but it is not considered here to simplify notation. We further assume the flow to be incompressible. Figure 1a shows a sketch of the problem including the gravitational acceleration vector, $\mathbf{g}$, the particles at three different instants (assuming $\rho_p > \rho_f$), and the laboratory frame, O^L_{xyz}. The laboratory frame is defined as the reference frame in which the fluid is at rest in the absence of particles. The vertical direction is defined as the direction opposite to gravitational acceleration, with unit vector $\mathbf{e}_z = -\mathbf{g}/|\mathbf{g}|$. In the following, we assume particles to be heavy and, for simplicity of notation, the governing equations are discussed for the case of a single particle: the change in formulation for lighter-than-the-fluid particles and the extension to an ensemble of particles being straightforward.

In the laboratory frame O^L_{xyz}, the governing equations for the fluid velocity, $\mathbf{u}$, and pressure, p, are the Navier–Stokes equations for an incompressible flow

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\,\mathbf{u} = -\frac{\nabla p}{\rho_f} + \nu\nabla^2\mathbf{u}\,, \tag{1}$$

$$\nabla\cdot\mathbf{u} = 0\,, \tag{2}$$

and for the particle velocity, $\mathbf{u}_p = \left(u_p, v_p, w_p\right)$, and angular velocity, $\boldsymbol{\omega}$, the Newton–Euler equations for a rigid body

$$\rho_p V\frac{\mathrm{d}\mathbf{u}_p}{\mathrm{d}t} = \int_S \boldsymbol{\tau}\cdot\mathbf{n}\mathrm{d}\sigma + V\left(\rho_p - \rho_f\right)\mathbf{g}\,, \tag{3}$$

$$\rho_p V\frac{\mathrm{d}(\boldsymbol{J}\boldsymbol{\omega})}{\mathrm{d}t} = \int_S \mathbf{r}\times(\boldsymbol{\tau}\cdot\mathbf{n})\,\mathrm{d}\sigma\,, \tag{4}$$

where S represents the surface of the particle, $\mathbf{n}$ is a unit vector normal to the surface of the particle pointing toward the fluid and $\mathbf{r}$ is a position vector with respect to the center of gravity. The fluid stress tensor, $\boldsymbol{\tau}$, is defined in terms of the fluid pressure and velocity as $\boldsymbol{\tau} = -p\boldsymbol{I} + \rho_f\nu\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)$, where $\boldsymbol{I}$ is the identity matrix. Please note that the term involving gravity is explicitly added in (3) because the gravity term has been eliminated from the fluid momentum Eq. (1), as is customarily done by subtracting the hydrostatic part from the pressure field. Note also that for an ensemble of particles, a collision term has to be added to the right hand side of (3). For a discussion of the various collisions models, we refer to the literature [2,7,12,13].

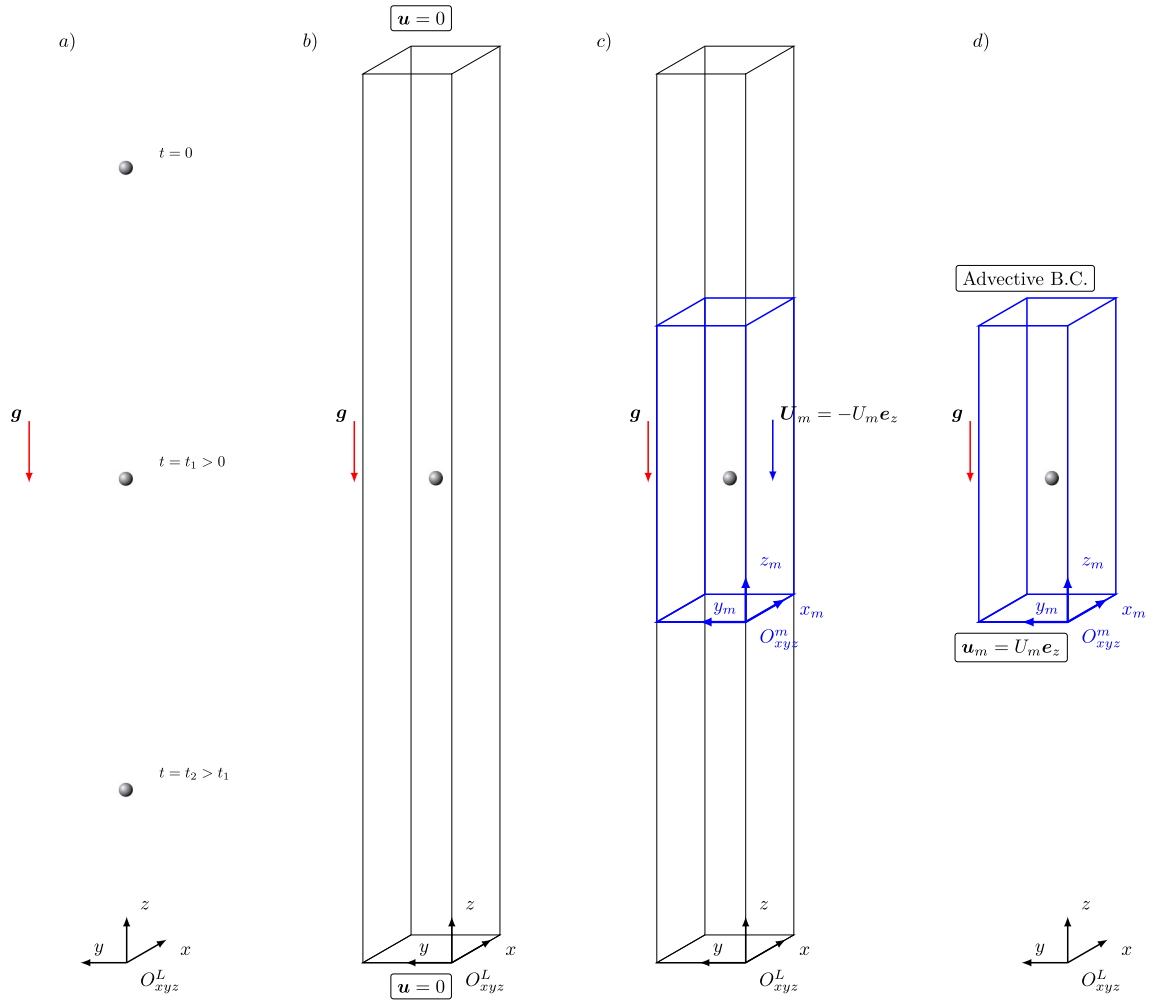

Fig. 1 a) Sketch of the particles settling in the unbounded domain and **b)** representation of a computational domain in the laboratory frame, O^L_{xyz}. **c)** Sketch of the relation of the laboratory (O^L_{xyz}) and moving (O^m_{xyz}) frames. **d)** Representation of the computational domain in the moving frame, O^m_{xyz}. The gravitational acceleration **g** is included in all the panels, and the boundary conditions in the bottom and top boundaries are indicated in panels b and d

Dimensional analysis shows that, in addition to the solid volume fraction, the problem is governed by two non-dimensional parameters. Among the possible combinations, we select the density ratio between the particle and the fluid, ϱ, and the Galileo number, Ga, defined as

$$\varrho = \frac{\rho_p}{\rho_f}, \tag{5}$$

$$Ga = \frac{U_g D}{\nu}, \tag{6}$$

where D is the diameter and $U_g = \sqrt{(\varrho - 1)\,|\mathbf{g}|\,D}$ is a gravitationally scaled velocity.

If we choose ρ_f, D and U_g as reference quantities, we can rewrite Eqs. (1)-(4) in non-dimensional form as

$$\frac{\partial\left(\frac{\mathbf{u}}{U_g}\right)}{\partial\left(\frac{t\,U_g}{D}\right)}+\left(\frac{\mathbf{u}}{U_g}\cdot\tilde{\nabla}\right)\frac{\mathbf{u}}{U_g}=-\tilde{\nabla}\frac{p}{\rho_f U_g^2}+\frac{1}{Ga}\tilde{\nabla}^2\frac{\mathbf{u}}{U_g}\,, \tag{7}$$

$$\tilde{\nabla}\cdot\frac{\mathbf{u}}{U_g}=0\,, \tag{8}$$

$$\varrho\tilde{V}\frac{\mathrm{d}\left(\frac{\mathbf{u}_p}{U_g}\right)}{\mathrm{d}\left(\frac{t\,U_g}{D}\right)}=\int_S\left(-\frac{p}{\rho_f U_g^2}\boldsymbol{I}+\frac{1}{Ga}\left(\tilde{\nabla}\frac{\mathbf{u}}{U_g}+\left(\tilde{\nabla}\frac{\mathbf{u}}{U_g}\right)^T\right)\right)\cdot\mathbf{n}\mathrm{d}\tilde{\sigma}-\tilde{V}\mathbf{e}_z\,, \tag{9}$$

$$\varrho\tilde{V}\frac{\mathrm{d}\left(\tilde{\boldsymbol{J}}\frac{\boldsymbol{\omega}D}{U_g}\right)}{\mathrm{d}\left(\frac{t\,U_g}{D}\right)}=\int_S\tilde{\mathbf{r}}\times\left(-\frac{p}{\rho_f U_g^2}\boldsymbol{I}+\frac{1}{Ga}\left(\tilde{\nabla}\frac{\mathbf{u}}{U_g}+\left(\tilde{\nabla}\frac{\mathbf{u}}{U_g}\right)^T\right)\right)\cdot\mathbf{n}\mathrm{d}\tilde{\sigma}\,, \tag{10}$$

where the particle parameters ($\tilde{V}=V/D^3$ and $\tilde{\boldsymbol{J}}=\boldsymbol{J}/D^2$), the differential element of area ($\mathrm{d}\tilde{\sigma}=\mathrm{d}\sigma/D^2$) and the position vector ($\tilde{\mathbf{r}}=\mathbf{r}/D$) are also expressed in non-dimensional form. We also introduce the non-dimensional nabla operator normalized with the particle diameter $\tilde{\nabla}=\left(\frac{\partial}{\partial(x/D)},\frac{\partial}{\partial(y/D)},\frac{\partial}{\partial(z/D)}\right)$. Given the values of ϱ and Ga, Eqs. (7)–(10) can be solved numerically with proper boundary conditions. For a Cartesian domain like the one shown in Fig. 1b, the velocity at the bottom and top boundaries is set to zero ($\mathbf{u}=0$). In this work, periodicity is applied as a lateral boundary condition to approximate an unbounded domain; nevertheless, the method we present is not limited to periodic boundaries in the lateral directions.

In the following sections, we will rewrite the equations with respect to moving reference frames. To maintain clarity, we express non-dimensional quantities for both fluid and particles as ratios of dimensional quantities to their corresponding reference values. While this approach makes Eqs. (7)–(10) appear more complex, it reduces the number of variable definitions and prevents reader confusion when we present the proposed methodology. For quantities whose non-dimensional form remains constant across all reference frames in this work, such as particle geometric properties, we denote the non-dimensional version by adding a tilde to the symbol.

2.1 The need of a moving reference frame

Figure 1b shows a sketch of the computational domain used to integrate Eqs. (7)–(10) numerically. Any combination of a finite Galileo number ($Ga>0$) and a non-neutrally buoyant particle ($\varrho>1$) will produce a relative motion of the particle with respect to the fluid. As illustrated in Fig. 1a, after transients have been discarded, heavy particles will settle with an average terminal velocity. In order to accommodate the initial transient and a sufficiently long time interval of interest, the domain must be elongated in the vertical direction. As an alternative, it is possible to use a non-periodic domain (in the vertical direction) in conjunction with a moving frame, O^m_{xyz}, which travels at a constant velocity $\mathbf{U}_m=-U_m\mathbf{e}_z$ (with $U_m=|\mathbf{U}_m|$), with respect to the laboratory frame [1,17,23]. This is sketched in Fig. 1c. The velocity of the fluid and the particle with respect to the moving frame are defined as $\mathbf{u}_m=\mathbf{u}-\mathbf{U}_m$ and $\mathbf{u}_{pm}=\mathbf{u}_p-\mathbf{U}_m$, respectively. Since the velocity $\mathbf{U}_m$ is constant, we can write the Navier–Stokes equations for the pressure and fluid velocity with respect to the moving frame, p and $\mathbf{u}_m$, respectively, as

$$\frac{\partial\mathbf{u}_m}{\partial t}+(\mathbf{u}_m\cdot\nabla)\,\mathbf{u}_m=-\frac{\nabla p}{\rho_f}+\nu\nabla^2\mathbf{u}_m\,, \tag{11}$$

$$\nabla\cdot\mathbf{u}_m=0\,. \tag{12}$$

Similarly, we can write the Newton–Euler equations for the particle relative motion

$$\rho_p V\frac{\mathrm{d}\mathbf{u}_{pm}}{\mathrm{d}t}=\int_S\boldsymbol{\tau}\cdot\mathbf{n}\mathrm{d}\sigma+V\left(\rho_p-\rho_f\right)\mathbf{g}\,, \tag{13}$$

$$\rho_p V\frac{\mathrm{d}\left(\boldsymbol{J}\boldsymbol{\omega}\right)}{\mathrm{d}t}=\int_S\mathbf{r}\times\left(\boldsymbol{\tau}\cdot\mathbf{n}\right)\mathrm{d}\sigma\,. \tag{14}$$

Note that the pressure p, the angular velocity $\boldsymbol{\omega}$, and the fluid stresses are invariant to Galilean transformation, therefore we do not need to define new symbols for these quantities in the moving frame. Equation (14) is the same as Eq.(4), and it is repeated for readability. The change in reference frame implies that the boundary condition at the bottom boundary of the computational domain is $\mathbf{u}_m = U_m \mathbf{e}_z$ and a typical advective or stress-free condition type should be imposed at the top boundary (see Fig. 1d).

If we choose ρ_f, D, and U_m as reference quantities, we can rewrite Eqs. (11)-(14) in non-dimensional form as

$$\frac{\partial\left(\frac{\mathbf{u}_m}{U_m}\right)}{\partial\left(\frac{t\,U_m}{D}\right)} + \left(\frac{\mathbf{u}_m}{U_m}\cdot\tilde{\nabla}\right)\frac{\mathbf{u}_m}{U_m} = -\tilde{\nabla}\frac{p}{\rho_f U_m^2} + \frac{1}{Re_m}\tilde{\nabla}^2\frac{\mathbf{u}_m}{U_m}\,, \tag{15}$$

$$\tilde{\nabla}\cdot\frac{\mathbf{u}_m}{U_m} = 0\,, \tag{16}$$

$$\varrho\tilde{V}\frac{\mathrm{d}\left(\frac{\mathbf{u}_{pm}}{U_m}\right)}{\mathrm{d}\left(\frac{t\,U_m}{D}\right)} = \int_S\left(-\frac{p}{\rho_f U_m^2}\boldsymbol{I} + \frac{1}{Re_m}\left(\tilde{\nabla}\frac{\mathbf{u}_m}{U_m} + \left(\tilde{\nabla}\frac{\mathbf{u}}{U_g}\right)^T\right)\right)\cdot\mathbf{n}\mathrm{d}\tilde{\sigma} - \tilde{V}\left(\frac{Ga}{Re_m}\right)^2\mathbf{e}_z\,, \tag{17}$$

$$\varrho\tilde{V}\frac{\mathrm{d}\left(\tilde{\boldsymbol{J}}\frac{\omega D}{U_m}\right)}{\mathrm{d}\left(\frac{t\,U_m}{D}\right)} = \int_S\tilde{\mathbf{r}}\times\left(-\frac{p}{\rho_f U_m^2}\boldsymbol{I} + \frac{1}{Re_m}\left(\tilde{\nabla}\frac{\mathbf{u}_m}{U_m} + \left(\tilde{\nabla}\frac{\mathbf{u}_m}{U_m}\right)^T\right)\right)\cdot\mathbf{n}\mathrm{d}\tilde{\sigma}\,. \tag{18}$$

where we have introduced the Reynolds number based on the velocity of the moving frame with respect to the laboratory frame U_m,

$$Re_m = \frac{U_m D}{\nu}\,. \tag{19}$$

Given the values of ϱ, Ga, and Re_m, we could solve numerically Eqs. (15)–(18). The aim of integrating Eqs. (15)–(18) instead of (7)–(10) is twofold: i) to enable long time-integration intervals in relatively small computational domains, as sketched in Fig. 1c, leading to an affordable computational cost, and ii) to use existing numerical tools without having to implement ad hoc terms to account for non-inertial effects. In order to achieve these goals, it is essential to properly select the value of Re_m, which should correspond to the particle Reynolds number based on the terminal velocity of the particle (or particle ensemble), $\langle w_p\rangle_t$,

$$Re_p = \frac{-\langle w_p\rangle_t D}{\nu}\,, \tag{20}$$

where the operator $\langle\cdot\rangle_t$ represents time averaging.

The problem with this approach is that the particle Reynolds number, Re_p, is a result of the calculation, not known a priori. As a consequence, the precise value of Re_m to achieve the goal of long integration time intervals is not known at the beginning of the simulation. Indeed, we would like to have $Re_m = Re_p$, which would imply

$$Re_m = Re_p \Leftrightarrow U_m = -\langle w_p\rangle_t \Leftrightarrow \left\langle\frac{w_{pm}}{U_m}\right\rangle_t = 0\,, \tag{21}$$

resulting in no average drift of the particle in the vertical direction in the moving reference frame. Figure 2 shows the three possible scenarios regarding the value of Re_m with respect to Re_p:

- $Re_m < Re_p$: Particles drift toward the bottom of the domain and eventually are influenced by the presence of the bottom boundary.
- $Re_m > Re_p$: Particles drift toward the top of the domain and eventually are influenced by the presence of the top boundary.
- $Re_m \approx Re_p$: Particles stay in the computational domain for a long time interval.

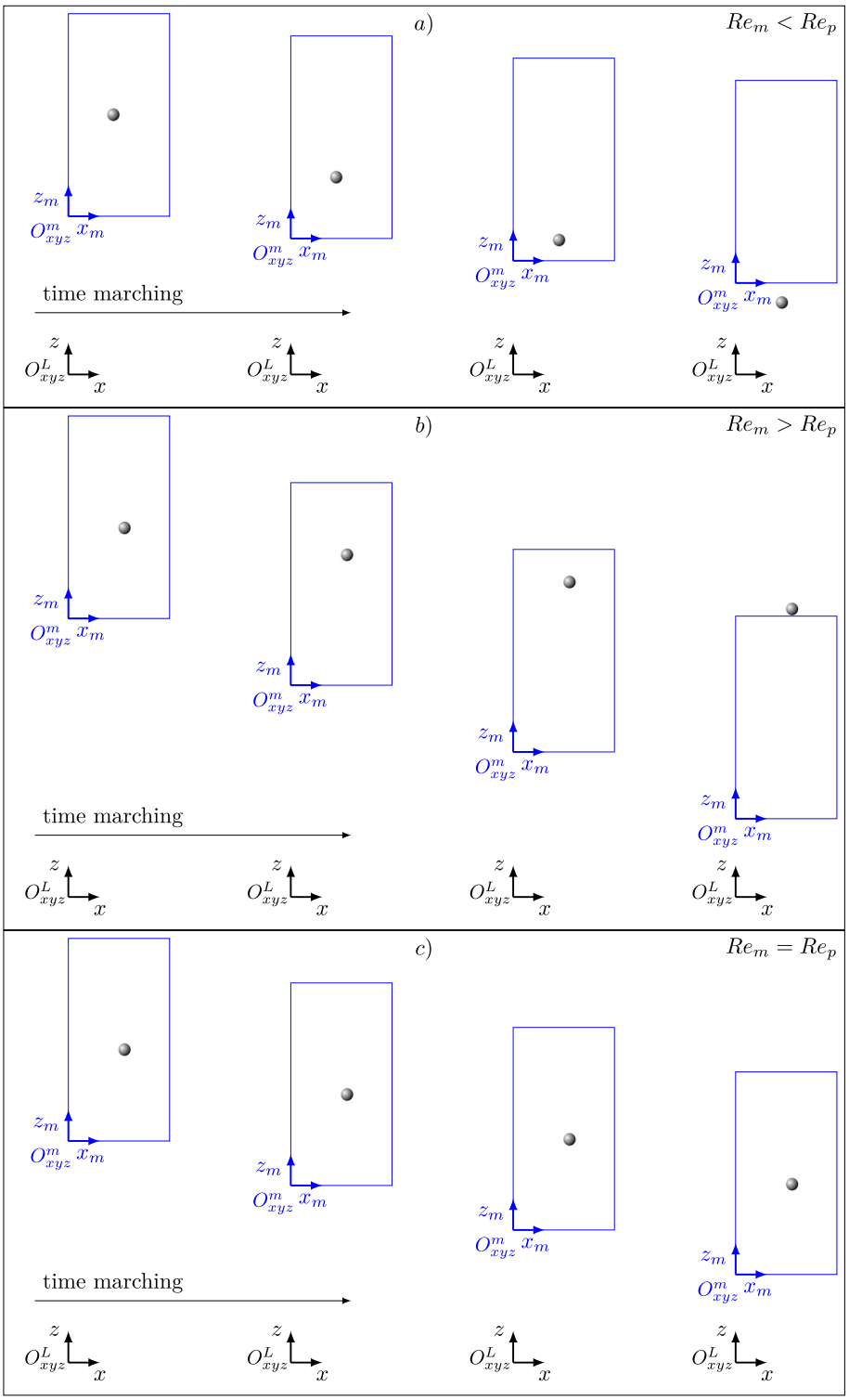

Fig. 2 Sketch of the possible scenarios for values of Re_m **a)** lower, **b)** higher, or **c)** equal to the particle Reynolds number Re_p

For a single particle, the scenarios discussed above comprise all possible cases. However, for an ensemble of particles, additional considerations may arise, particularly regarding the homogeneity of the mixture. For example, particles may drift apart from each other in certain situations. Even when the particle Reynolds number of the ensemble is properly captured, individual particles might drift and become influenced by the presence of top and/or bottom boundaries. The potential negative influence of these individual particles drifting away will depend on the total number of particles and should be evaluated with care.

In the following section, we present a method to iteratively correct the value of Re_m enabling long time-integration intervals in reasonably small computational domains (as sketched in Fig. 2c), and using a numerical tool in which Eqs. (15)–(18) are discretized, without the need of implementing ad hoc terms to account for non-inertial effects.

3 Methodology

Here, we propose an iterative method to obtain a sequence of estimates of Re_m, denoted $Re_m^{(s)}$, being s the iteration index, that, if successful, converges toward the particle (or particle ensemble) Reynolds number. This is done by letting the system evolve during consecutive, relatively short time intervals and using the resulting evolution to correct the next iterate. It will be shown that even if the initial guess is a rough estimate, the method is robust and converges efficiently. Please note that in this work, we use "iteration" to refer to a time interval during which the system evolves using the same $Re_m^{(s)}$. Each iteration consists of many individual time steps.

Let us first outline the methodology before proceeding with the technical details. We start with an initial condition, such as the fluid and particles at rest in the laboratory frame, and specify an initial value for the Reynolds number $Re_m^{(0)}$, which can be estimated using simple order-of-magnitude methods or any other suitable approach. Assuming that we have already performed $s-1$ iterations, then the following steps are to be followed for the next iteration s:

1. We change the non-dimensional state variables at the end of the previous iteration, $t_B^{(s-1)}$, from the laboratory frame O_{xyz}^L to the moving reference frame $O_{xyz}^{m,s}$. This is done using a simple mapping operation to be discussed below.
2. We then integrate Eqs. (15)–(18) in a time interval $\left[t_A^{(s)}, t_B^{(s)}\right]$ with an estimated value of the Reynolds number $Re_m^{(s)}$. The length of the time interval can be determined during run time. For example, when a predefined particle vertical displacement threshold is met, the time interval ends. During integration, one must accumulate the complete time evolution of a reduced set of particle-related statistics (compared to the full set of state variables), which will be later used to estimate the Reynolds number of the next iteration, $Re_m^{(s+1)}$.
3. The non-dimensional state variables are now transferred from the moving reference frame $O_{xyz}^{m,s}$ to the laboratory reference frame O_{xyz}^L, by using the inverse mapping operation of step 1. This is done in two substeps:
   - The reduced set of statistics is mapped for the whole time interval $\left[t_A^{(s)}, t_B^{(s)}\right]$.
   - The full set of state variables is mapped only for the last time instant $t_B^{(s)}$ to initialize the next iteration.
4. We then concatenate the reduced set of statistics obtained in this s-th interval to that of the previous intervals. In this way, at this stage the complete time evolution in the interval $[0, t_B^{(s)}]$ is available for the statistics needed to compute $Re_m^{(s+1)}$.
5. Using the complete time evolution of the reduced set of statistics, we can correct the Reynolds number and determine its value for the next iteration, $Re_m^{(s+1)}$. This involves a control/optimization strategy which is problem dependent and it is discussed in its own section below.
6. We can now return to step 1 for the next iteration

Please note that the choice of retaining intermediate fluid and/or particle states in step 2 is left to the user; we show here the steps using the minimum amount of storage resources for the algorithm to function.

3.1 Elements of the algorithm

3.1.1 Set of state variables

We define the two sets of non-dimensional variables each of which fully defines the state of the system:

$$\mathbf{\Phi}(\mathbf{x}, t; Ga, \varrho) := \left\{ \frac{\mathbf{u}}{U_g}, \frac{p}{\rho_f U_g^2}, \frac{\mathbf{u}_p}{U_g}, \frac{\boldsymbol{\omega} D}{U_g} \right\} , \tag{22}$$

associated with the laboratory frame O^L_{xyz}, and

$$\mathbf{\Phi}^{(s)}_m(\mathbf{x}, t; Ga, \varrho, Re^{(s)}_m) := \left\{ \left(\frac{\mathbf{u}_m}{U_m}\right)^{(s)}, \left(\frac{p}{\rho_f U_m^2}\right)^{(s)}, \left(\frac{\mathbf{u}_{pm}}{U_m}\right)^{(s)}, \left(\frac{\boldsymbol{\omega} D}{U_m}\right)^{(s)} \right\} , \tag{23}$$

with the s-th moving frame $O^{m,s}_{xyz}$. Ideally, one would like to integrate the state vector in the laboratory frame shown in (22) since it only depends on the governing parameters of the problem, Re_m and ϱ. However, in order to integrate the solution for long time intervals, we rely on integrating Eqs.(15)–(18) with $Re_m = Re^{(s)}_m$ to obtain the state vector $\mathbf{\Phi}^{(s)}_m$. Therefore, the main purpose of working with $\mathbf{\Phi}^{(s)}_m$ is numerical integration.

3.1.2 Mapping functions to change the reference frame

In this section, we define the mapping functions $\mathcal{M}$ and $\mathcal{M}^{-1}$ to change the reference frame from the laboratory to the moving frame and vice versa, respectively,

$$\mathbf{\Phi}_m = \mathcal{M}\left(\mathbf{\Phi}; \gamma\right) , \tag{24}$$

$$\mathbf{\Phi} = \mathcal{M}^{-1}\left(\mathbf{\Phi}_m; \gamma\right) , \tag{25}$$

where we introduce the ratio between the velocity of the moving frame and the gravitationally scaled velocity

$$\gamma = \frac{U_m}{U_g} = \frac{Re_m}{Ga} . \tag{26}$$

Assuming that the moving frame travels with respect to the laboratory frame along the vertical direction $\mathbf{U}_m = -U_m \mathbf{e}_z$ (see Fig. 1c), the resultant expressions are

$$\mathbf{\Phi}_m = \mathcal{M}\left(\mathbf{\Phi}; \gamma\right) \Rightarrow \quad \begin{cases} \frac{\mathbf{u}_m}{U_m} & = \frac{\mathbf{u}}{U_g}\,\gamma^{-1} + \mathbf{e}_z \\ \frac{p}{\rho_f U_m^2} & = \frac{p}{\rho_f U_g^2}\,\gamma^{-2} \\ \frac{\mathbf{u}_{pm}}{U_m} & = \frac{\mathbf{u}_p}{U_g}\,\gamma^{-1} + \mathbf{e}_z \\ \frac{\boldsymbol{\omega} D}{U_m} & = \frac{\boldsymbol{\omega} D}{U_g}\,\gamma^{-1} \end{cases} , \tag{27}$$

$$\mathbf{\Phi} = \mathcal{M}^{-1}\left(\mathbf{\Phi}_m; \gamma\right) \Rightarrow \quad \begin{cases} \frac{\mathbf{u}}{U_g} & = \left(\frac{\mathbf{u}_m}{U_m} - \mathbf{e}_z\right)\gamma \\ \frac{p}{\rho_f U_g^2} & = \frac{p}{\rho_f U_m^2}\,\gamma^2 \\ \frac{\mathbf{u}_p}{U_g} & = \left(\frac{\mathbf{u}_{pm}}{U_m} - \mathbf{e}_z\right)\gamma \\ \frac{\boldsymbol{\omega} D}{U_g} & = \frac{\boldsymbol{\omega} D}{U_m}\,\gamma \end{cases} . \tag{28}$$

Please note that we have dropped the iteration index (s) for clarity.

3.1.3 Integration

As discussed above, numerical integration is performed in the moving reference frame. Figure 1d shows the computational setup in which the problem is integrated: an inflow/outflow configuration. More specifically, the bottom boundary condition is of Dirichlet type, imposing a uniform free stream of velocity $U_m \mathbf{e}_z$. We apply an advective boundary condition at the top to minimize computational domain influence, though other approaches achieving this goal are possible. Similar freedom of choice applies for the lateral boundary conditions. Particularly, we assume the domain to be periodic in the lateral directions (x and y according to Fig. 1d), but the algorithm has no restrictions in the usage of other type of boundary conditions. This flexibility is an advantage: Unknown effects of certain lateral boundary conditions on particle settling pose no additional challenge.

The integration time interval of the s-th iteration is $[t_A^{(s)}, t_B^{(s)}]$, where the starting point is the end of the previous iteration $t_A^{(s)} = t_B^{(s-1)}$. There are two issues to consider when defining the duration of the time-integration interval: First, when $Re_m^{(s)} \neq Re_m^{(s-1)}$, the flow variables after restart present a small unphysical discontinuity that is damped after a few time steps. This is due to the change in reference frame and the presence of the nonlinear term in the momentum equation. Since our main interest is in reaching a statistically stationary state after having applied corrections and then keeping Re_m constant, and we are not primarily interested in the transient state, we disregard this issue in the following. Nevertheless, the length of the intervals should be sufficiently long to accommodate the damping of the small discontinuity. Based on our experience, a minimum of 10 time steps is a good rule of thumb. Second, rough estimates of Re_m, although handled by the method, may result in moderately large vertical drifts inside the computational domain. Therefore, the first set of iterations should be relatively short to avoid large displacements of the particle(s) in the moving frame. This requirement can be relaxed if, during runtime, the simulation is designed to stop when a specified condition is violated, for example, when the particle's vertical position falls below or rises above predefined thresholds.

The other aspect to consider for numerical integration is the selection of the time step. In this work, we use a fixed time step normalized with the gravitationally scaled velocity, $\Delta t\, U_g/D$, which means that the time step is updated with every correction of $Re_m^{(s)}$. Therefore, the non-dimensional time step used to integrate numerically Eqs. (7)–(18) is computed as follows

$$\left(\frac{\Delta t\, U_m}{D}\right)^{(s)} = \gamma^{(s)} \frac{\Delta t\, U_g}{D}\,. \tag{29}$$

The relationship between the time normalized with the gravitationally scaled velocity, $t\, U_g/D$, and the time normalized with the velocity of the moving frame, $t\, U_m/D$, is somewhat more elaborated than Eq. (29) and is discussed in Appendix A.

3.1.4 Correction of Reynolds number Re_m

In this section, we describe how to correct the value of the Reynolds number between successive iterations. This method component is highly dependent on the specific problem and user preferences. Therefore, we present here the simplest expression to update the ratio $\gamma = Re_m/Ga$, which for a fixed Galileo number is equivalent to obtain an expression for $Re_m = \gamma\, Ga$, but somehow more convenient since in this work we are presenting the mapping operators as functions of γ. Starting from $w_p = w_{pm} - U_m$, after simple algebra, we obtain the expression

$$\gamma = \frac{w_p/U_g}{w_{pm}/U_m - 1}, \tag{30}$$

which can be averaged in time, leading to

$$\gamma = \frac{\langle w_p/U_g \rangle_t}{\langle w_{pm}/U_m \rangle_t - 1}. \tag{31}$$

Since the objective is to maintain the particle at an approximately constant vertical position relative to the moving reference frame, we seek $\langle w_{pm} \rangle_t = 0$. From this condition, we can obtain an expression to update γ for the next iteration

$$\gamma^{(s+1)} = -\left\langle \frac{w_p}{U_g} \right\rangle_t, \tag{32}$$

and consequently $Re_m^{(s+1)} = \gamma^{(s+1)} Ga$.

To illustrate the algorithm's basic functioning, we consider an artificially generated test case mimicking single-particle settling behavior: A particle starting from rest undergoes constant acceleration during a transient period of duration t_c before reaching terminal velocity. Therefore, the vertical velocity relative to the laboratory frame as a function of time is given by

$$\frac{w_p}{U_g} = \begin{cases} -A\dfrac{t}{t_c}, \text{ if } & 0 \le t < t_c \\ -A\,, \text{ if } & t_c \le t \end{cases}, \tag{33}$$

where $A > 0$ is a known constant.

For the iterative procedure, we use fixed-duration "integration" intervals (iterations) of $T_{\text{int}} = t_c/4$, and averaging intervals of the same duration $T_{\text{avg}} = T_{\text{int}}$. Therefore, the averaged value of w_p/U_g obtained in the s-th iteration is

$$\left\langle \frac{w_p}{U_g} \right\rangle_t^{(s)} = \frac{1}{t_B^{(s)} - t_A^{(s)}} \int_{t_A^{(s)}}^{t_B^{(s)}} \frac{w_p}{U_g}(\tau)\mathrm{d}\tau\,, \tag{34}$$

where $t_B^{(s)} - t_A^{(s)} = T_{\text{avg}} = T_{\text{int}}$. The term "integration" is marked by quotes because, in this particular example, we do not integrate numerically, but use the information given in equation (33) (in the laboratory frame) and construct the "integrated" solution in the s-th moving frame with the relation

$$\left(\frac{w_{pm}}{U_m}\right)^{(s)} = \frac{w_p}{U_g}\,\gamma^{(s)^{-1}} + 1\,. \tag{35}$$

In a realistic case, this procedure is reversed because the settling velocity w_p/U_g is not known a priori. One first integrates numerically in the s-th moving frame to obtain the particle-related statistics, in this case $(w_{pm}/U_m)^{(s)}$, and then applies the appropriate mapping functions to obtain w_p/U_g. In the following, we will present the procedure as if it would be done in a realistic case, using the term "integration."

Figure 3 shows the evolution of the iterative procedure. The problem starts from rest ($w_p/U_g = 0 \Leftrightarrow w_{pm}/U_m = 1$), and we initialize γ with an underestimated value $\gamma^{(0)} = 0.8\,A$. (Simple algebra shows that the converged value of γ is A.) After the first iteration, the particle is still at the early stages of the transient, and therefore, the averaged value of the settling velocity (see Fig. 3b) is small compared to the terminal velocity. We update the value of γ according to Eq. (32) and obtain $\gamma^{(1)} = 0.13\,A$. Because during the first iteration the particle is accelerating downward from rest, the averaged value of w_p/U_g used to compute $\gamma^{(1)}$ underestimates the instantaneous velocity at the end of the interval; therefore, the particle has a negative initial velocity in the moving frame of the second iteration, $(w_{pm}/U_m)^{(1)}(t_A^{(1)}) < 0$. Furthermore, it keeps on accelerating downward. As commented above, this behavior deviates from our target ($\langle w_{pm}/U_m \rangle_t = 0$), but it only exposes the lack of knowledge of the solution, which is overcome by using relatively short integration intervals, especially at the early stages. In the consecutive iterations, we obtain $\gamma^{(2)} = 0.38A$, $\gamma^{(3)} = 0.63A$, and $\gamma^{(4)} = 0.88A$, which shows how the algorithm progressively reduces the difference between $\gamma^{(s)}$ and its target $\gamma = A$. This can be seen in Fig. 3a, where the vertical downward motion of the particle is smaller and smaller as the iterative procedure advances.

The success of the algorithm starts to be visible when the transient comes to an end. This can be seen in Fig. 3b, where the selected signal to compute the average of w_p/U_g is free from the influence of the transient. This results in $\gamma^{(5)} = A$, which, because of the simplicity of the current example, is exactly our target value and results in $\langle w_{pm} \rangle_t = 0$. From this point on, we can conclude that the iterative process is ended and the rest of the problem is simply evolved without further modifying the value of γ.

Please note that although this seems an oversimplified example, it does include the main features of realistic cases, especially the initial transient behavior. The main differences with respect to a realistic case are that more, and (probably longer) integration intervals are needed after the transient has ended in order to ensure that: i) the statistically stationary regime is fully attained, and ii) the averaged values involved in updating γ are robust enough.

To conclude the section, we follow the notation from § 3.1.1 and introduce two vectors of non-dimensional quantities that contain the reduced set of particle-related statistics used by the correction algorithm: $\mathbf{\Pi}$, associated with the laboratory frame, and $\mathbf{\Pi}_m$, with the moving frame. We also define two mapping functions to

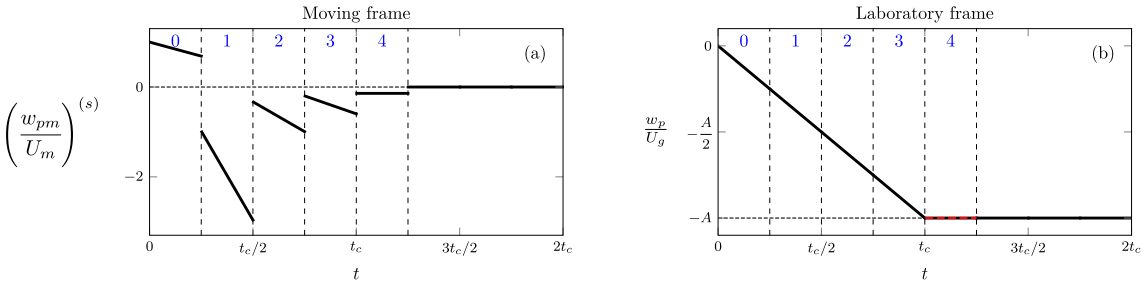

Fig. 3 Evolution of the solution of the illustrative example in the **a)** moving and **b)** laboratory frame. In panel b, the portion of signal used to correct γ after the last iteration ($s = 4$) is shown with a red, dashed line. Iterations are identified by the vertical dashed lines, with the iteration index in blue on top of each panel

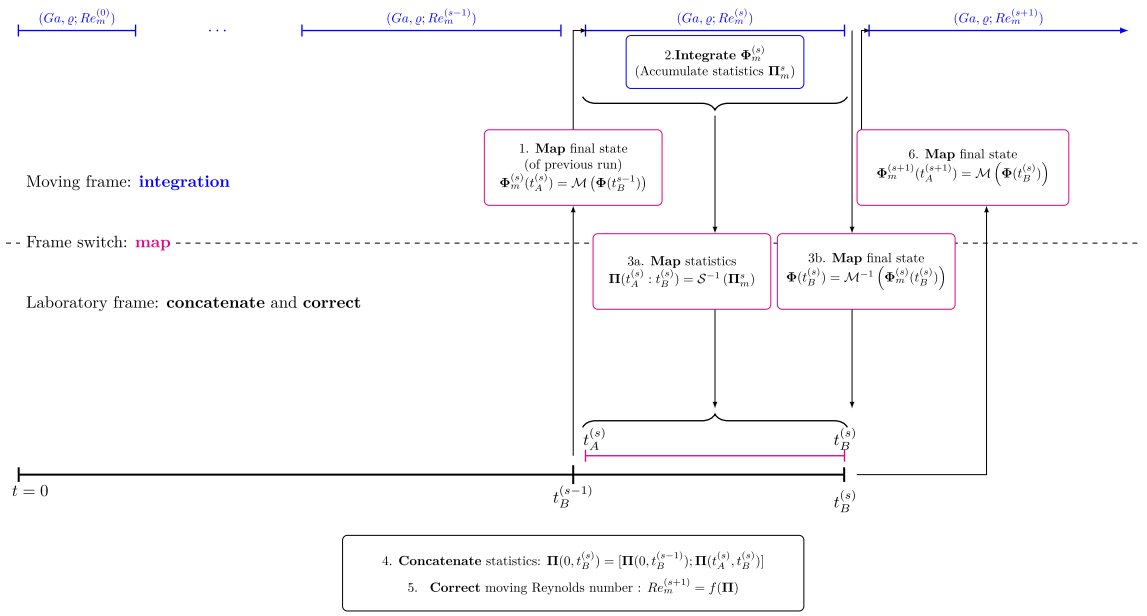

Fig. 4 Diagram of the workflow of the algorithm

transform between reference frames: $\mathcal{S}$ maps from the laboratory to the moving frame, and $\mathcal{S}^{-1}$ maps from the moving to the laboratory frame. In the example discussed above, this would have been

$$\mathbf{\Pi}_m = \left\{\frac{w_{pm}}{U_m}\right\} = \mathcal{S}\left(\mathbf{\Pi}; \gamma\right) = \frac{w_p}{U_g}\,\gamma^{-1} + 1\,, \tag{36}$$

$$\mathbf{\Pi} = \left\{\frac{w_p}{U_g}\right\} = \mathcal{S}^{-1}\left(\mathbf{\Phi}_m; \gamma\right) = \left(\frac{w_{pm}}{U_m} - 1\right)\gamma\,. \tag{37}$$

The user should include in $\mathbf{\Pi}$ (and $\mathbf{\Pi}_m$) the quantities needed for the correction algorithm employed to successfully work. This will become clearer in the following sections, where we present examples with single and multiple particles. Finally, the whole methodology is summarized in the diagram shown in Fig. 4.

4 Settling of a single particle

In this section, we show the performance of the method for a single sphere settling under gravity in the steady vertical regime. We set the Galileo number and the density ratio to $Ga = 121$ and $\varrho = 1.5$, based on the works of [9] and [22]. Following [23], we use a cuboidal computational domain of dimensions $[16/3 \times 16/3 \times 16]D^3$ and a spatial resolution of $D/\Delta x = 24$ in all directions, resulting in a number of grid cells of $[128 \times 128 \times 384]$. We

impose an inflow boundary condition with constant velocity $U_m \mathbf{e}_z$ at the inlet (bottom boundary), an advective condition at the outlet (top boundary), and periodicity in the lateral directions, as sketched in Fig. 1a.

Because of the simplicity of the steady vertical regime, and the fact that we are dealing only with one particle, the correction algorithm is very similar to the example shown in § 3.1.4. Thus, the reduced sets of statistics are $\mathbf{\Pi} = \{w_p/U_g\}$ and $\mathbf{\Pi}_m = \{w_{pm}/U_m\}$, with the corresponding mapping functions already presented in Eqs. (36) and (37). We use a time step of $\Delta t\, U_g/D = 8 \cdot 10^{-3}$ and fixed-duration integration intervals of $T_{\text{int}} U_g/D = 50$ (6250 time steps). As mentioned in § 3.1.3, the time step normalized with the velocity of the moving frame of the s-th iteration, $(\Delta t\, U_m/D)^{(s)}$, is updated according to Eq. (29).

We initialize the problem with both fluid and particle at rest

$$\frac{\mathbf{u}}{U_g} = (0,0,0) \Leftrightarrow \frac{\mathbf{u}_m}{U_m} = (0,0,1)\ , \tag{38}$$

$$\frac{\mathbf{u}_p}{U_g} = (0,0,0) \Leftrightarrow \frac{\mathbf{u}_{pm}}{U_m} = (0,0,1)\ . \tag{39}$$

The iteration procedure on Re_m is initialized using a point-particle model in which the drag coefficient is given by

$$C_D = f(Re_p)\,\frac{24}{Re_p}, \tag{40}$$

where $f(Re_p) = 1 + 0.15 Re_p^{0.672}$ is the factor proposed by Schiller and Naumann (Crowe et al., [3], their equation 4.15), which is reasonably good for Reynolds number up to 800 ($f = 1$ recovers Stokes drag). After some algebra and setting the drag force equal to the submerged weight of a sphere, one obtains

$$Re_p\, f\left(Re_p\right) = \frac{Ga^2}{18}\,. \tag{41}$$

The initial Reynolds number is set to the numerically determined solution of this equation, resulting in $Re_m^{(0)} = 145.5$ for the present case.

Table 1 shows the Reynolds number and the time step used to integrate Eqs. (15)–(18) in every iteration. It can be seen that the Reynolds number converges steadily to the value $Re_m = 139.97$. This value is then the computed particle Reynolds number Re_p, which only differs from the value of the point-particle model by about 4%. For the sake of simplicity, iterations and integration intervals are analogous. The value of γ is updated according to $\gamma^{(s+1)} = -\left\langle (w_p/U_g)^{(s)} \right\rangle_t$ (see Eq. (32)), but the time window to compute the averaged vertical velocity is now dynamically updated

$$\left\langle \frac{w_p}{U_g} \right\rangle_t^{(s)} = \frac{1}{t_B^{(s)}/2} \int_{t_B^{(s)}/2}^{t_B^{(s)}} \frac{w_p}{U_g}(\tau)\mathrm{d}\tau\ , \tag{42}$$

which implies that we used the last half of the simulated time to compute the next iterate of $Re_m^{(s+1)} = \gamma^{(s+1)} Ga$. During the initial transient, accelerations are large and the shorter the averaging window, the better the algorithm will compensate large drifts in the computational domain. As the case evolves, and approaches convergence, the averaging window increases, resulting in more stable predictions for next iterates. This is particularly useful to handle cases in which the particle starts from rest and shows unsteadiness in the converged state.

Figure 5 shows the evolution of the vertical velocity of the particle in the laboratory frame normalized with the fluid viscosity, equivalent to an instantaneous particle Reynolds number (panel a) and in the moving frames for every iteration (panel b). The insets include an amplified view of the transition from one iteration to the next one, showing a very smooth update and small oscillations with respect to the averaged value. It should be noted that although, strictly speaking, the converged state of this case is stationary, the numerical approximation can show small oscillations when using immersed boundary methods (IBMs) (as we do in this example). This is a consequence of the relative motion between the fluid and particle discretization inherent of IBMs (see Section 5.1.2 of Uhlmann, [20]). In the moving frame, panel b, it can be seen how the initial rest condition of the particle ($w_p = 0$) in the laboratory frame corresponds to an upward drift in the moving frame ($w_{pm}/U_m = 1$) that results in a vertical displacement of the particle inside the computational domain of approximately $5D$. The vertical motion in the computational domain (moving frame) is negligible from the fourth iteration.

Table 1 Integration parameters Re_m and $\Delta t\, U_m/D$ corresponding to each iteration of the single particle settling in the steady vertical regime ($Ga = 121$, $\varrho = 1.5$)

Iteration (s)	$Re_m^{(s)}$	$\left(\frac{\Delta t U_m}{D}\right)^{(s)}$
0	145.500	$9.6198 \cdot 10^{-3}$
1	138.856	$9.1805 \cdot 10^{-3}$
2	140.120	$9.2642 \cdot 10^{-3}$
3	139.940	$9.2522 \cdot 10^{-3}$
4	139.973	$9.2544 \cdot 10^{-3}$
5	139.967	$9.2540 \cdot 10^{-3}$

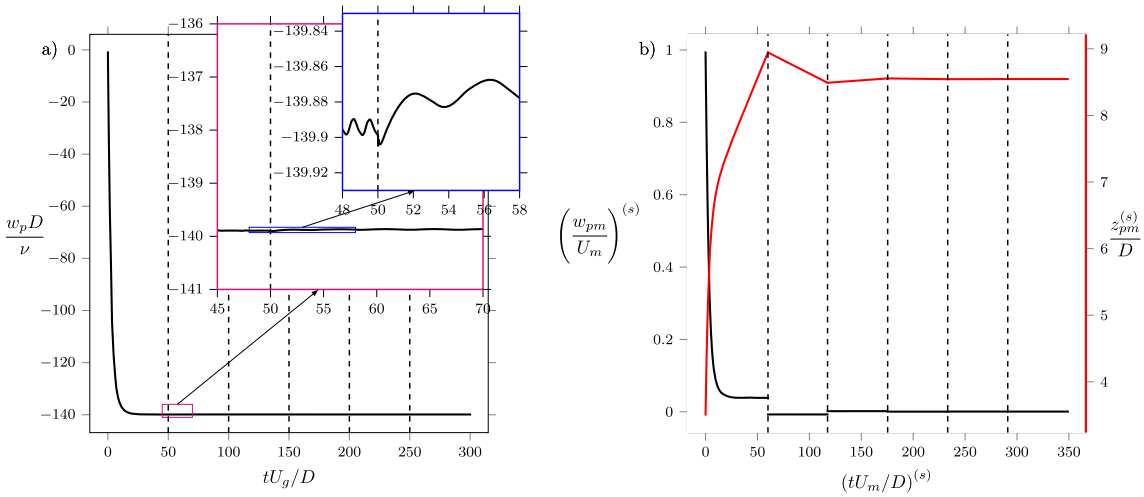

Fig. 5 Time evolution of **a)** vertical velocity expressed as a particle Reynolds number (laboratory frame) and **b)** vertical velocity (black) and position (red) in the moving frame corresponding to each iteration. Insets in **a)** are zooms of the same quantities

5 Settling of multiple particles

5.1 Problem description and computational setup

To demonstrate the capabilities of the proposed method, we now extend the analysis from a single particle to a suspension of many spherical particles, in the dilute regime. As in previous studies for dilute systems [16,22], for the contact modeling we use the method proposed by [7]. This example highlights the main advantage of the present approach, its efficiency, and robustness in handling multi-particle systems.

We focus on a configuration for which collective effects have been reported in the literature. Specifically, we select the parameters of case M178 from [22]: $Ga = 178$, $\varrho = 1.5$, and solid volume fraction $\Phi^* = 5 \cdot 10^{-3}$, corresponding to a dilute regime. For these flow parameters, a single particle follows a steady oblique trajectory; this is known as the steady oblique regime [23]. In the triply periodic computations performed by [22], the solid volume fraction was defined as

$$\Phi^* = \frac{N_p V}{L_x\, L_y\, L_z}\,, \tag{43}$$

where N_p is the number of particles and L_x, L_y, and L_z are the dimensions of the computational domain. The superscript * highlights that this definition applies to triply periodic configurations.

In the present work, we employ a smaller computational domain than [22], $(L_x\, L_y\, L_z) = [64/3 \times 64/3 \times 64]D^3$, to reduce the computational cost. Recall that the aim here is to show the feasibility of the method. The spatial resolution is $D/\Delta x = 24$ in every direction, resulting in a number of grid cells equal to $[512 \times 512 \times 1536]$. The initial condition consists of a uniform distribution of particles within a subregion of the computational domain of height $H = 39D$; the regions above and below are filled with fluid only (see Fig. 6). The particles are distributed following a random Poisson process.

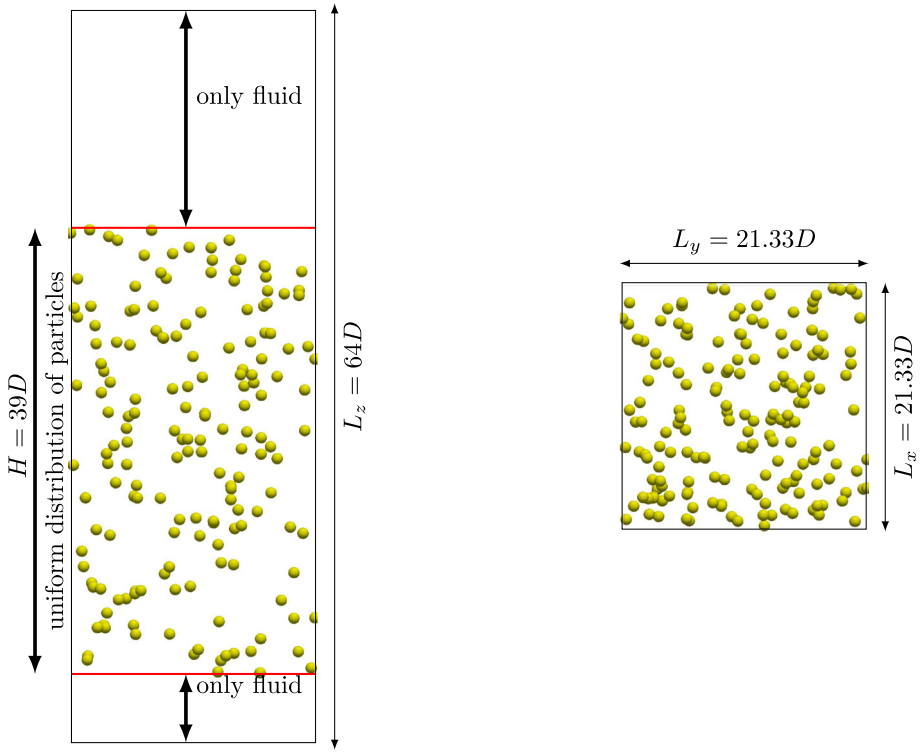

Fig. 6 **a)** Lateral and **b)** top view of the computational domain and the initial position of the particles. Sketch-like annotations are included for clarity, but both the representation of the computational domain and the rendering of the particles are shown to scale

Replacing L_z by H in Eq. (43), and using the same value of the volume fraction as [22],

$$\Phi = \frac{N_p V}{L_x\, L_y\, H} = 5 \cdot 10^{-3}, \tag{44}$$

we can solve for N_p and obtain $N_p = 169$. Note that we have discarded the superscript *. This is because in the inflow–outflow configuration, the initially uniform particle concentration becomes non-uniform as the simulation progresses. It develops a spatial variation that depends on the vertical coordinate, $\phi(z)$, as will be shown below.

Finally, we use a time step of $\Delta t\, U_g/D = 8 \cdot 10^{-3}$ and variable-size integration intervals of minimum 200 steps and, as mentioned in § 3.1.3, the time step normalized with the velocity of the moving frame will be updated according to Eq. (29).

5.2 Correction algorithm

In the multi-particle case, the correction algorithm requires more statistical information than for a single particle. We monitor four quantities: the mean vertical position of the particle ensemble, its standard deviation, the lowest particle position, and the highest particle position. The switch from velocities to positions allows us to account for the spatial extent of the particle ensemble, which determines when corrections to the Reynolds number Re_m should be applied. In the single-particle case, this was not needed due to the short transients involved. We still require the mean settling velocity to correct Re_m, but this is easily obtained from the time derivative of the ensemble-averaged position z_p^{avg}. The sets of reduced statistics are now $\mathbf{\Pi} = \left\{z_p^{\text{avg}}/D, z_p^{\text{min}}/D, z_p^{\text{max}}/D, z_p^{\text{std}}/D\right\}$, and $\mathbf{\Pi}_m = \left\{z_{pm}^{\text{avg}}/D, z_{pm}^{\text{min}}/D, z_{pm}^{\text{max}}/D, z_{pm}^{\text{std}}/D\right\}$.

Each particle position in the laboratory frame is given by

$$z_p^{(i)} = z_b + z_{pm}^{(i)}\,, \tag{45}$$

where z_b is the position of the lower boundary of the computational domain in the laboratory frame (see Appendix A for its expression). Then, the four quantities which appear in $\mathbf{\Pi}$ are defined as

$$z_p^{\text{avg}} = \frac{\sum_{N_p} z_p^{(i)}}{N_p}, \tag{46}$$

$$z_p^{\text{min}} = \min\left(z_p^{(i)}\right), \tag{47}$$

$$z_p^{\text{max}} = \max\left(z_p^{(i)}\right), \tag{48}$$

$$z_p^{\text{std}} = \sqrt{\frac{\sum_{N_p}\left(z_p^{(i)} - z_p^{\text{avg}}\right)^2}{N_p}}. \tag{49}$$

Thus, the mean, minimum, and maximum position in both frames differs only by the shift z_b, whereas the standard deviation remains identical.

Figure 7a and b shows the evolution of the metrics during the simulation. The correction algorithm includes the following features:

- If z_p^{min} or z_p^{max} crosses a predefined lower or upper limit, the simulation is paused and a new estimate of γ is computed.
- A threshold on the mean vertical velocity is also enforced. If exceeded, the simulation is paused and a new estimate of γ is computed.
- A vertical drift, prescribed by the user, may be imposed to ensure the particles remain well inside the computational domain. For this, the equation for the next iteration is modified with respect to (32) as follows

$$\gamma^{(s+1)} = \frac{\langle w_p/U_g\rangle_t}{L_{\text{drift}}/T_{\text{drift}} - 1} \tag{50}$$

  where L_{drift} and T_{drift} denote the desired vertical drift length and drift duration, respectively, for the next iteration. These are target quantities that may not be exactly achieved during the subsequent time evolution. The closer the system is to a statistically steady state, characterized by reduced unsteadiness in the mean settling velocity and by being further from the initial transient, the more accurately these desired quantities will be reproduced in the next iteration.
- The thresholds are dynamically updated based on the observed behavior of the system.

The goal of simulating the settling of many particles over a long period was successfully achieved. Figure 7 shows the entire time history of the simulation, including the long initial transient during which corrections were applied and in which the thresholds were gradually updated, as indicated by the various horizontal lines. The total interval without corrections spans approximately 600 D/U_g. Although only this portion is used for analysis, the complete time history remains useful for feeding the correction algorithm and examining the system s overall behavior.

In the following section, time-averaged quantities are computed over the window $1129.9 < t\,U_g/D < 1718.4$, where no further corrections are applied and the solution exhibits a reasonable degree of statistical stationarity. During this time interval, the mean vertical position of the particle ensemble presents a tiny drift. As a consequence, the Reynolds number imposed during this last interval, $Re_m = 233.52$, is slightly different from the actual particle Reynolds number $Re_p = 234.4$. This difference is, however, smaller than 0.4%. The success of the method is reflected in how small this difference is, which allows for long time-integration intervals. This is the case even when the particle distribution size shows oscillatory behavior of moderate amplitude ($3D$) and large timescale ($300D/U_g$), as seen in the evolution of z_p^{std} in Fig. 7b.

5.3 Results

Now, let us briefly discuss some of the results that are obtained in this new configuration. First, for illustration, we show a 3D visualization of a slab of the domain, representing the particles and the vortex structure identified using the second invariant of the velocity gradient tensor, Q (Fig. 8a), and the vertical velocity of the fluid phase (Fig. 8b). It can be seen how the wake around the particles located at the bottom part of the distribution

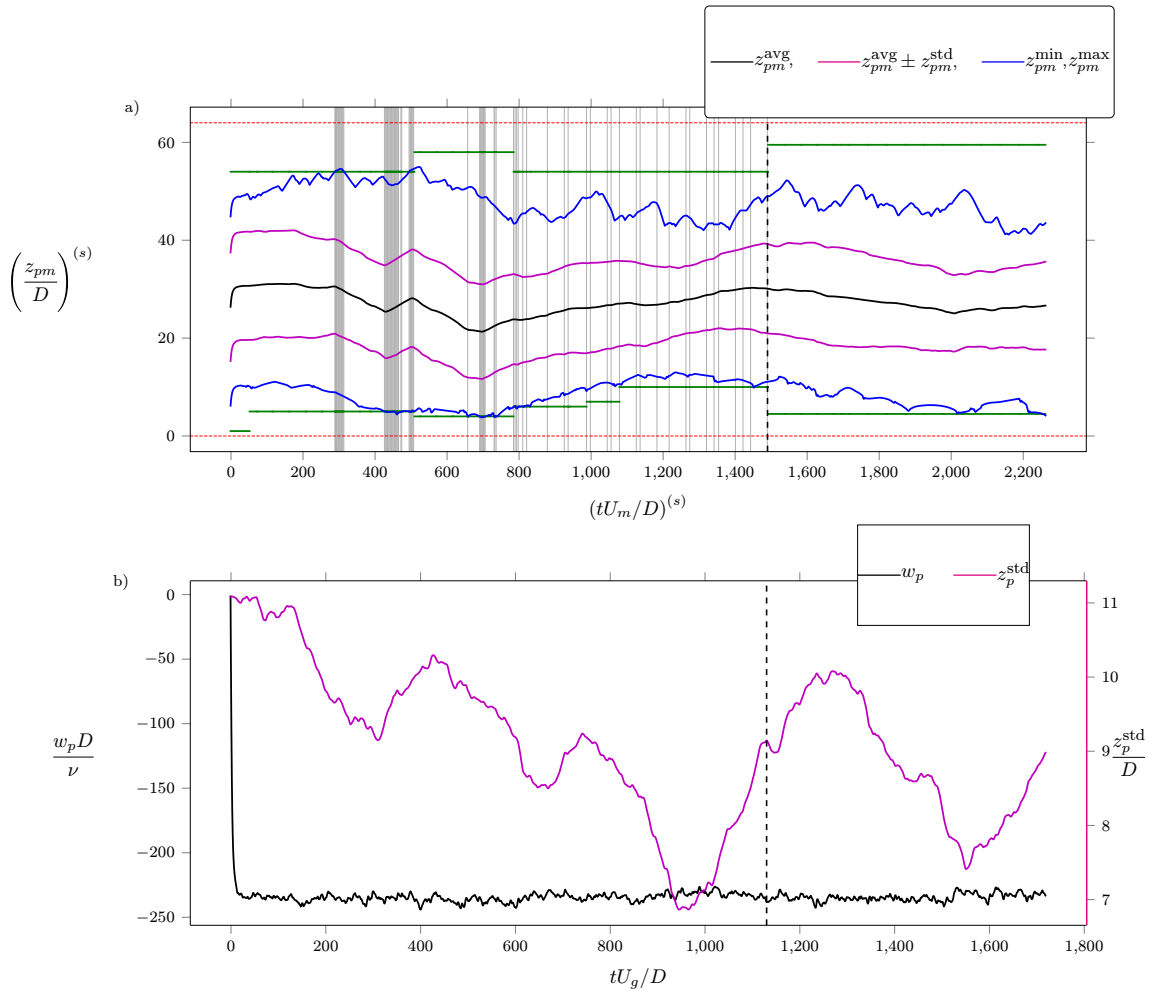

Fig. 7 Time history of **a)** Π_m^s as they are computed (moving frame) and **b)** vertical velocity in the laboratory frame and width of the distribution. In **a)**, the thin gray lines indicate change of moving frame, the black dashed line the last correction, the red dashed lines the limits of the computational domain and the green lines the minimum and maximum thresholds used in the correction algorithm. The vertical dash line in both panels indicates the beginning of the converged part of the simulation, in which no correction is applied, which corresponds to $(t\ U_m/D)^{(s)} = 1489.9$ and $t\ U_g/D = 1129.9$

shows a toroidal vortex (see panel a), and a clearly tilted, straight, and elongated wake (see panel b), resembling the solution of an isolated particle in the steady oblique regime. This is the regime that corresponds to the selected Galileo number and density ratio for an isolated particle [9,23], as mentioned above. Differently, particles located downstream of the first front of particles show unsteady wakes with the characteristic hairpin-like vortex structures, whose formation is triggered by the hydrodynamic interactions with the surrounding particles.

Another phenomenon that can be examined in this configuration is the perturbed flow field that develops within and downstream of the particle swarm. For example, in the region of fluid surrounding the zone of higher particle concentration located near the center of the domain, the downward velocity perturbations reduce the drag experienced by the particles located further downstream. In contrast, regions with few or no particles are also slightly perturbed (although not visible in the figure): Because of conservation of mass, a weak upward flow develops that forms a vertically aligned region of increased drag.

Figure 9 shows a contour plot of the particle concentration as a function of time and of the vertical coordinate, $\left(z - z_p^{\mathrm{avg}}\right)/D$. The plot includes the whole time evolution for illustration, but we analyze only the time interval where no corrections are applied, beyond the vertical dashed line in the figure. The figure shows that there are time intervals during which the particle ensemble is more compressed, indicated by the red patches in the figure. One such area is located between the two auxiliary lines toward the end of the simulation,

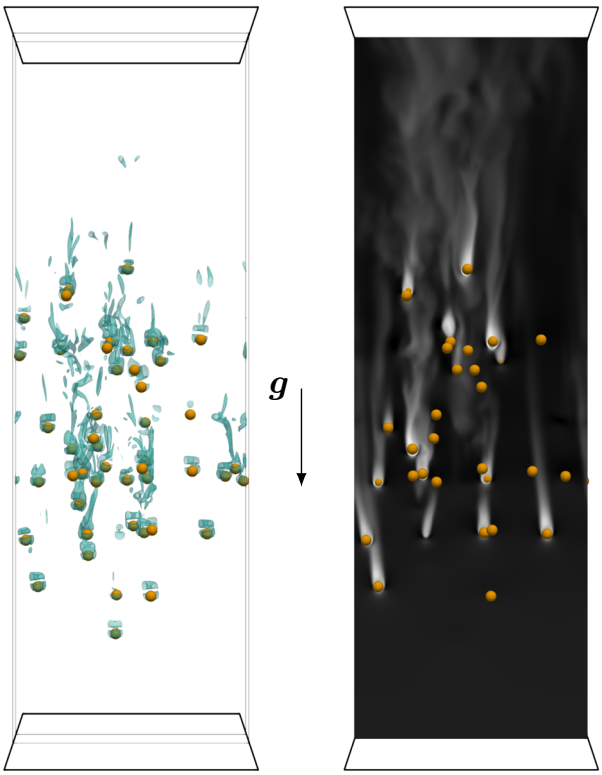

Fig. 8 Visualization of a slab of the problem of **a)** the Q-criterion and **b)** the vertical velocity

$t\,U_g/D = [1480, 1650]$. At earlier times, $t\,U_g/D < 1480$, the particle ensemble occupies a broader area. This is clearly visible in Fig. 10a that shows the time evolution of a measure of the width of the particle distribution (z_p^{std}). This quantity presents an oscillatory behavior with maximum and minimum at times which are consistent with Fig. 9.

Figure 10a shows also the evolution of the vertical velocity of the particle ensemble. With the given non-dimensionalization, the quantity can be understood as an instantaneous particle Reynolds number. The time evolution of this quantity presents high-frequency oscillations, probably due to the small amount of particles considered here. Interestingly, in the interval where the particle ensemble is more compressed, the vertical velocity of the particle ensemble is somewhat lower. This reduction in velocity may be attributed to a hindrance effect: When particles are more closely packed, the fluid displaced by their motion generates upward disturbances that interfere with neighboring particles, thereby reducing the overall settling speed of the ensemble. However, the number of particles considered in this study is too small for statistical convergence and these observations need to be confirmed in simulations with a larger number of particles.

Finally, Fig. 10b shows the vertical profile of the mean particle concentration, averaged over the time interval without corrections, together with the initial distribution for comparison. Because of the small number of particles in this dilute regime, and since statistics are computed separately at each height, the mean concentration profile exhibits poor convergence. To obtain a smoother representation, the figure also includes a smoothed version, ϕ_B, obtained by applying a box-filter with a filter width of $10D$. The results indicate that the particle distribution tends to adopt a Gaussian-like shape, with a maximum concentration near the center and two nearly symmetric tails extending toward the top and bottom. In future simulations with a larger number of particles, it will be interesting to examine whether this Gaussian shape persists, how the peak concentration compares with the initial one, and how far the distribution tails extend beyond the initial profile, among other questions.

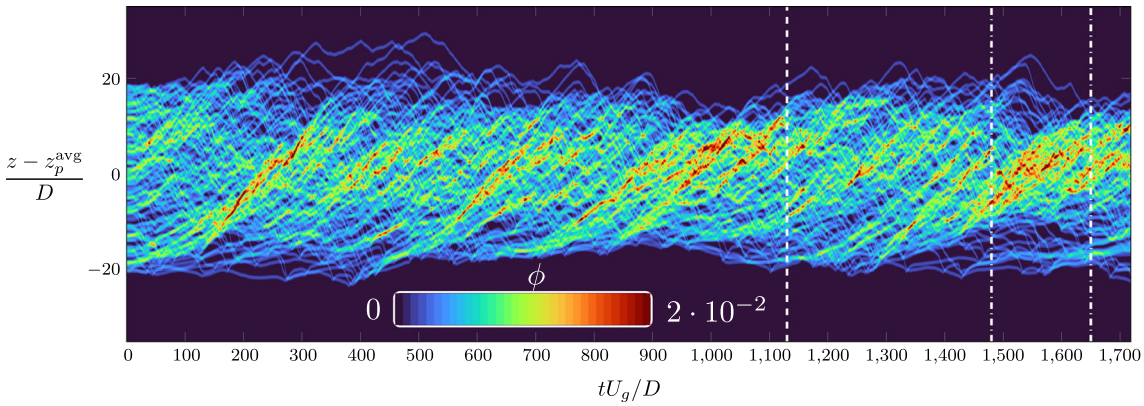

Fig. 9 Contour plot of particle concentration as a function of time and the vertical coordinate. The vertical dash line indicates the beginning of the converged part of the simulation, and the dash-dotted lines are auxiliary lines ($t\,U_g/D = 1480, 1650$) also represented in Fig. 10a to support the discussion

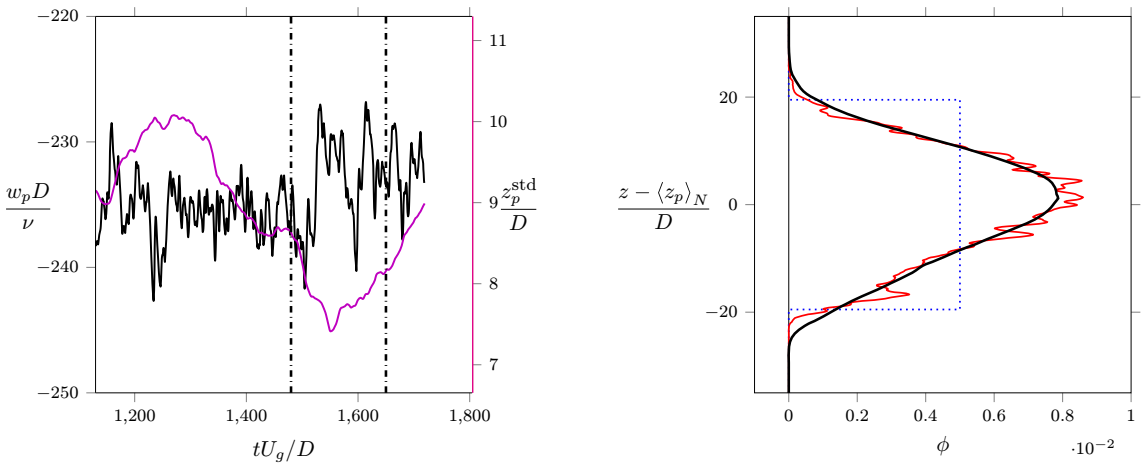

Fig. 10 a) Vertical velocity of the particle ensemble (black line) and standard deviation of particles' vertical position (magenta line) as a function of time. **b)** Solid volume fraction averaged in the lateral directions and time ($\langle\phi\rangle_t$, red curve) and box-averaged with a filter width of $10D$ ($\langle\phi_B\rangle_t$, black curve). In a), the vertical dash-dotted lines are auxiliary lines ($t\,U_g/D = 1480, 1650$) also represented in Fig. 9 to support the discussion. In b), the initial particle concentration distribution is represented with a blue dotted line

6 Conclusions

In this work, we advance the simulation of particle suspensions settling under gravity. The most common setup for this type of problem is a triply periodic computational domain. However, such configurations suffer from strong vertical correlations, which hinder the study of phenomena such as cluster dynamics. Here, we show that it is possible to remove the vertical periodicity and propose a methodology to overcome the associated numerical challenges.

With the aid of two examples of increasing complexity—a single particle in the steady vertical regime and a many-particle case exhibiting collective effects—we have demonstrated the robustness and efficiency of the method. We measure the success and feasibility of the method through two aspects: First, the approach is largely systematic; apart from the correcting algorithm, which is problem dependent, all components of the method are simple and well structured. Furthermore, some indications have been given to deal with the openness of the correcting algorithm. The second aspect that demonstrates the feasibility of the method is that a converged, free-of-corrections interval of approximately $600D/U_g$ has been obtained in a many-particle case. This constitutes the first simulation of this kind presented to date (to the best of our knowledge).

The effectiveness of the method follows from two aspects. Even simulations initiated with coarse estimates of the moving-frame Reynolds number provide valuable information on the system's settling dynamics. Each

iteration refines these estimates, progressively approaching the converged state. In addition, the consistent update of flow and particle variables during reference-frame adjustments minimizes discontinuities in the solution when simulations are restarted.

The proven feasibility of the method opens new possibilities in the analysis of particle-laden flows. By eliminating vertical periodicity, the long-term dynamics of clusters can now be examined under physically uncorrelated conditions. A further observation is that the final particle concentration profile emerges as a result of the simulation rather than as an imposed input. The relevant control parameter becomes the total particle weight per unit area, yielding a more realistic configuration in which the suspension naturally compresses or expands according to the underlying hydrodynamic interactions.

This framework also allows investigation of the influence of still fluid below the settling region on the lower particle layers. For example, in this work we find that the enhancement of the settling velocity is significantly lower compared to the triply periodic case. This observation raises important questions about whether an unperturbed fluid layer acts effectively as a wall that suppresses collective effects, or whether the result reflects finite-domain limitations, among other possibilities. The approach also permits detailed examination of the fluctuating flow induced by particle interactions: Whether clusters can grow to scales large enough to excite broad flow structures and recover canonical turbulence, or whether the energy injected by particle wakes remains confined to pseudo-turbulent regimes.

Finally, we emphasize that the proposed methodology can be implemented in any numerical solver that integrates the Navier–Stokes equations in an inertial reference frame. Since this includes the majority of current codes, the present method broadens the accessibility of this class of problems to the wider research community.

Acknowledgements The authors acknowledge TU Wien Bibliothek for financial support through its Open Access Funding Programme.

Funding Open access funding provided by TU Wien (TUW).

Declarations

Author contribution M.M. wrote the main manuscript text. All authors reviewed the manuscript.

Funding information This work was supported by the German Research Foundation (DFG) under Project UH 242/11−1 and by the Austrian Science Fund (FWF) 10.5576/PAT1793625. For open access purposes, the author has applied a CC BY public copyright license to any author-accepted manuscript version arising from this submission.

Data availability Data sets generated during the current study are available from the corresponding author on reasonable request.

Conflict of interest The authors declare no conflict of interest.

A Technical details on the treatment of time

Although time is treated in a simple way, the relationship between the time normalized by gravitationally scaling, $t\,U_g/D$, and the time normalized by the moving-frame velocity, $t\,U_m/D$, requires some discussion. In some cases, this distinction can be ignored, for example, when analyzing a single particle in § 4. However, when determining the position of the bottom boundary of the moving domain in the laboratory frame, z_b, as in § 5.2 or in more complex situations such as imposing continuous turbulence at the inlet, this issue must be addressed. Therefore, in this appendix, we establish the relation between the time in the s-th moving frame and the time in the laboratory frame.

Assuming that the origin of time ($t = 0$) is the beginning of the simulation, the times normalized by U_g and U_m in the first iteration ($s = 0$) are simply related by

$$\left(\frac{t\,U_m}{D}\right)^{(0)} = \gamma^{(0)}\frac{t\,U_g}{D}\,; \tag{51}$$

recall that $\gamma^{(s)} = Re_m^{(s)}/Ga$. Since the moving frame velocity changes from iteration to iteration, this needs to be taken into account leading to, for the s-th iteration,

$$\left(\frac{t\,U_m}{D}\right)^{(s)} = \sum_{i=0}^{s-1}\left(\gamma^{(i)}\frac{T^{(i)}\,U_g}{D}\right) + \gamma^{(s)}\frac{\left(t - t_A^{(s)}\right)U_g}{D}\,, \tag{52}$$

where $T^{(i)} = t_B^{(i)} - t_A^{(i)}$ is the interval duration of the i-th iteration.
The position of the lower boundary in the laboratory frame, z_b, is easily obtained since during each iteration the reference frame travels at constant speed. Therefore, for the s-th iteration it is given by

$$\frac{z_b^{(s)}}{D}(t) = -\sum_{i=0}^{s-1}\left(\frac{T^{(i)}U_m^{(i)}}{D}\right) - \frac{\left(t - t_A^{(s)}\right)U_m^{(s)}}{D}\,, \tag{53}$$

where we have assumed that $z_b(t = 0) = 0$. Introducing in this expression, the factor $\gamma^{(s)} = Re_m^{(s)}/Ga = U_m^{(s)}/U_g$, leads to

$$\frac{z_b^{(s)}}{D}(t) = -\sum_{i=0}^{s-1}\left(\gamma^{(i)}\frac{T^{(i)}U_g}{D}\right) - \gamma^{(s)}\frac{\left(t - t_A^{(s)}U_g\right)}{D} = -\left(\frac{tU_m}{D}\right)^{(s)}\,. \tag{54}$$

In the last equality, Eq. (52) has been used.

References

1. Catalán, J.M., Moriche, M., Flores, O., García-Villalba, M.: On the settling of a spherical particle in slightly perturbed ambient fluid. Acta Mech. **235**(4), 2479–2493 (2024)
2. Costa, P., Boersma, B.J., Westerweel, J., Breugem, W.-P.: Collision model for fully resolved simulations of flows laden with finite-size particles. Phys. Rev. E **92**(5), 053012 (2015)
3. Crowe, C.T., Schwarzkopf, J.D., Sommerfeld, M., Tsuji, Y.: Multiphase flows with droplets and particles. CRC Press (2011)
4. Fornari, W., Ardekani, M.N., Brandt, L.: Clustering and increased settling speed of oblate particles at finite Reynolds number. J. Fluid Mech. **848**, 696–721 (2018)
5. Fornari, W., Picano, F., Brandt, L.: Sedimentation of finite-size spheres in quiescent and turbulent environments. J. Fluid Mech. **788**, 640–669 (2016)
6. Garcia-Villalba, M., Colonius, T., Desjardins, O., Lucas, D., Mani, A., Marchisio, D., Matar, O.K., Picano, F., Zaleski, S.: Numerical methods for multiphase flows. Int. J. Multiph. Flow **191**, 105285 (2025)
7. Glowinski, R., Pan, T.-W., Hesla, T.I., Joseph, D.D.: A distributed Lagrange multiplier/fictitious domain method for particulate flows. Int. J. Multiph. Flow **25**(5), 755–794 (1999)
8. Huisman, S.G., Barois, T., Bourgoin, M., Chouippe, A., Doychev, T., Huck, P., Morales, C.E.B., Uhlmann, M., Volk, R.: Columnar structure formation of a dilute suspension of settling spherical particles in a quiescent fluid. Phys. Rev. Fluids **1**, 074204 (2016)
9. Jenny, M., Dušek, J., Bouchet, G.: Instabilities and transition of a sphere falling or ascending freely in a Newtonian fluid. J. Fluid Mech. **508**, 201–239 (2004)
10. Jiang, X., Xu, C., Zhao, L.: Prolate spheroids settling in a quiescent fluid: clustering, microstructures and collisions. J. Fluid Mech. **1000**, A49 (2024)
11. Kajishima, T., Takiguchi, S.: Interaction between particle clusters and particle-induced turbulence. Int. J. Heat Fluid Flow **23**(5), 639–646 (2002)
12. Kempe, T., Fröhlich, J.: Collision modelling for the interface-resolved simulation of spherical particles in viscous fluids. J. Fluid Mech. **709**, 445–489 (2012)
13. Kidanemariam, A.G., Uhlmann, M.: Interface-resolved direct numerical simulation of the erosion of a sediment bed sheared by laminar channel flow. Int. J. Multiph. Flow **67**, 174–188 (2014)
14. Lu, J., Xu, X., Zhong, S., Ni, R., Tryggvason, G.: The dynamics of suspensions of prolate spheroidal particles - Effects of volume fraction. Int. J. Multiph. Flow **165**, 104469 (2023)

15. Marchioli, C., Bourgoin, M., Coletti, F., Fox, R., Magnaudet, J., Reeks, M., Simonin, O., Sommerfeld, M., Toschi, F., Wang, L.-P., Balachandar, S.: Particle-laden flows. Int. J. Multiph. Flow **191**, 105291 (2025)
16. Moriche, M., Hettmann, D., García-Villalba, M., Uhlmann, M.: On the clustering of low-aspect-ratio oblate spheroids settling in ambient fluid. J. Fluid Mech. **963**, A1 (2023)
17. Moriche, M., Uhlmann, M., Dušek, J.: A single oblate spheroid settling in unbounded ambient fluid: a benchmark for simulations in steady and unsteady wake regimes. Int. J. Multiph. Flow **136**, 103519 (2021)
18. Seyed-Ahmadi, A., Wachs, A.: Sedimentation of inertial monodisperse suspensions of cubes and spheres. Phys. Rev. Fluids **6**, 044306 (2021)
19. Subramaniam, S., Balachandar, S.: (eds.). *Modeling approaches and computational methods for particle-laden turbulent flows*. Academic Press (2023)
20. Uhlmann, M.: An immersed boundary method with direct forcing for the simulation of particulate flows. J. Comput. Phys. **209**(2), 448–476 (2005)
21. Uhlmann, M., Derksen, J., Wachs, A., Wang, L.-P., Moriche, M.: Efficient methods for particle-resolved direct numerical simulation. In *Modelling approaches and computational methods for particle-laden turbulent flows* (ed. S. Subramaniam & S. Balachandar), pp. 147–184. Academic press (2023)
22. Uhlmann, M., Doychev, T.: Sedimentation of a dilute suspension of rigid spheres at intermediate Galileo numbers: the effect of clustering upon the particle motion. J. Fluid Mech. **752**, 310–348 (2014)
23. Uhlmann, M., Dušek, J.: The motion of a single heavy sphere in ambient fluid: a benchmark for interface-resolved particulate flow simulations with significant relative velocities. Int. J. Multiph. Flow **59**, 221–243 (2014)
24. Villafañe, L., Aliseda, A., Ceccio, S., DiMarco, P., Machicoane, N., Heindel, T.J.: 50 years of International Journal of Multiphase Flow: Experimental methods for dispersed multiphase flows. Int. J. Multiph. Flow **189**, 105239 (2025)