The application of Buckingham π theorem to Lattice-Boltzmann modelling of sewage sludge digestion

Davide Dapelo [a,*], Robin Trunk [c], Mathias J. Krause [c], Nigel Cassidy [b], John Bridgeman [a]

[a] *Faculty of Engineering and Informatics, University of Bradford, B2.20, Chesham Building, Bradford BD7 1DP, United Kingdom*
[b] *Department of Civil Engineering, University of Birmingham, Birmingham, United Kingdom*
[c] *Lattice Boltzmann Research Group, Karlsruhe Institute of Technology, Karlsruhe, Germany*

ARTICLE INFO

ABSTRACT

For the first time, a set of Lattice-Boltzmann two-way coupling pointwise Euler-Lagrange models is applied to gas mixing of sludge for anaerobic digestion. The set comprises a local model, a "first-neighbour" (viz., back-coupling occurs to the voxel where a particle sits, plus its first neighbours) and a "smoothing-kernel" (forward- and back-coupling occur through a smoothed-kernel averaging procedure).

Laboratory-scale tests display grid-independence problems due to bubble diameter being larger than voxel size, thereby breaking the pointwise Euler-Lagrange assumption of negligible particle size. To tackle this problem and thereby have grid-independent results, a novel data-scaling approach to pointwise Euler-Lagrange grid independence evaluation, based on an application of the Buckingham π theorem, is proposed.

Evaluation of laboratory-scale flow patterns and comparison to experimental data show only marginal differences in between the models, and between numerical modelling and experimental data. Pilot-scale simulations show that all the models produce grid-independent, coherent data if the Euler-Lagrange assumption of negligible (or at least, small) particle size is recovered. In both cases, a second-order convergence was achieved.

A discussion follows on the opportunity of applying the proposed data-scaling approach rather than the smoothing-kernel model.

1. Introduction

Wastewater process is an energy-intensive operation. EU figures [8] for Member States show that wastewater treatment works (WwTWs)'s energy consumption exceeds 23,800 GWh per annum, with further increase of 60% in the next 10–15 years being expected due to tightened regulation of effluent discharge (Water Framework Directive, 14). [46] forecasted a global need of water increase by 30% and food and energy by 50% in the next decades, and hence, it is urgent to address the water-energy link in order to tackle climate change.

UK WwTWs product 1.5M tonnes of sludge per annum [43]. Sludge, the principal by-product of wastewater treatment, is usually treated through mesophilic gas-mixed anaerobic digestion: anaerobic bacteria degrade sludge at 22–41 °C into more stable compounds and methane-rich biogas, which in turn is harnessed as a renewable source of energy through combined heat and power technology.

Mixing is key for stable digestion process. Despite mixing being responsible for 17–73% of digester's energy consumption [32], mixing optimization is still far from being optimal: [23]'s experimental work demonstrated that input mixing energy can be halved without impacting nutrient distribution, and hence, biogas yield. It is therefore imperative to orient mixing design and operation towards a more modern approach where stress is given to input mixing / output biogas energy balance, rather than merely to digestate quality.

Numerical modelling is an economic alternative to lengthy and expansive tracer experiments; furthermore, whilst tracer methods can only provide "black box" representations, numerical modelling allows analysis of flow patterns in details. A considerable amount of numerical modelling has been devoted to improve mixing in anaerobic digestion [6,9,10,12,19,20,26,28,29,36,38,41,44,47]. On the

* Corresponding author.
E-mail address: ddapelo@bradford.ac.uk (D. Dapelo).

Nomenclature

Γ_K	Colour function for the Kth Lagrangian pointwise particle
Δt_{inj}	Time interval between the injection of two bubbles, s
Δ^+	Maximum extremal of $\mathcal{I}_K^3(\boldsymbol{x})$, m
Δ^-	Minimum extremal of $\mathcal{I}_K^3(\boldsymbol{x})$, m
Θ_x	Vortex's x coordinate, m
Φ	Sourcing term, kg m^3
Ξ	Collision step, kg m^3
α_j^i	Power exponent expressing the contribution of the jth dimensional parameter to the ith dimensionless group
$|\dot{\gamma}|$	shear rate magnitude, s^{-1}
$\delta_{\delta x}$	Discrete delta function of width δx
δx	Lattice cell size, m
δt	Lattice timestep, s
$\varepsilon_{\ell,K}$	Smoothed volumetric liquid phase fraction at the location of the Kth Lagrangian pointwise particle
μ	Apparent dynamics viscosity, Pa s
μ_{max}	Apparent dynamics viscosity, maximum range value, Pa s
μ_{min}	Apparent dynamics viscosity, minimum range value, Pa s
μ_{PL}	Apparent dynamics viscosity before turbulence correction, Pa s
ν	Kinematic viscosity, m^2 s^{-1}
ξ	Generic dimensionless variable
ν_{eff}	Effective kinematic viscosity after turbulence correction, m^2 s^{-1}
π_i	i th independent dimensionless parameter ("group") describing a generic physical process
π_i^{P}	i th "Physically-meaningful" independent dimensionless group describing a generic physical process
π_1	Forth dimensionless group
π_2	Fifth dimensionless group
π_3	Sixth dimensionless group
π_4	Seventh dimensionless group
$\hat{\pi}_i$	i th arbitrarily-modified independent dimensionless parameter group describing a generic physical process
$\bar{\pi}_i$	i th tuned independent dimensionless parameter group describing a generic physical process
ρ	Liquid-phase density, Kg m^{-3}
ρ_{L}	Physical value of the liquid phase density, Kg m^{-3}
ρ_{p}	Physical value of the Lagrangian dispersed phase density, Kg m^{-3}
σ	Liquid-phase shear stress, Pa
τ	Lattice relaxation time, s
τ_{eff}	Effective lattice relaxation time after turbulence correction, s
τ_{PL}	Lattice relaxation time before turbulence correction, s
φ	Map extracting the "physically-meaningful" dimensionless group from the set of all the independent dimensionless groups describing a generic physical process
χ	Map mapping a set of arbitrarily-modified to a set of tuned dimensionless groups describing a generic physical process
$\boldsymbol{A}_K$	Acceleration of the Kth Lagrangian pointwise particle, m s^{-2}
$\mathcal{B}_K^{\square}$	Bounding cube around the sphere centered at $\boldsymbol{X}_K$ of radius $\mathcal{L}_K$
$\mathscr{C}$	Collision operator, kg m^3
C_{d}	Drag coefficient
$\mathcal{C}_K$	Normalization factor of the Kth Lagrangian pointwise particle at a given timestep
C_{Smago}	Smagorinsky constant
D	Tank diameter, m
$\boldsymbol{F}_K$	Total force acting on the Kth Lagrangian pointwise particle, N
$\boldsymbol{F}_K^{\text{a}}$	Added-mass force acting on the Kth Lagrangian pointwise particle, N
$\boldsymbol{F}_K^{\text{b}}$	Buoyancy force acting on the Kth Lagrangian pointwise particle, N
$\boldsymbol{F}_K^{\text{d}}$	Drag force acting on the Kth Lagrangian pointwise particle, N
D	Tank height, m
$\mathcal{I}^3(\boldsymbol{x})$	Cube of side δx centered at $\boldsymbol{x}$
$\mathcal{I}_\ell^3(\boldsymbol{x})$	Parellelepiped where to evaluate e_K
$\mathcal{L}$	Chosen value for the smoothed kernel radius, m
$\mathcal{L}_K$	Smoothed kernel radius of the Kth Lagrangian pointwise particle, m
K	Power-law consistency coefficient, Pa s^n
K	(as a subscript) Generic label to a Lagrangian pointwise particle
$\mathcal{K}_K$	Kernel radius of the Kth Lagrangian pointwise particle, m
M_i	i th unit of measure connected to the description of a generic physical process
M_K	Mass of the Kth Lagrangian pointwise particle, Kg
Ma	Mach number
N	Number of points along a coordinate direction to fill $\mathcal{I}_K^3(\boldsymbol{x})$
N	Number of lattice points along the tank diameter
N^0	Reference number of lattice points along the tank diameter
M_{bubbles}	Total number of bubbles at a given timestep
P_{d}	Total drag power dissipated at each timestep, W
$\mathscr{P}_K$	Tuple representing the Kth Lagrangian pointwise particle
PNu	First physically-meaningful dimensionless group
PNuU	Second physically-meaningful dimensionless group
PU	Third physically-meaningful dimensionless group
Q	Forth physically-meaningful dimensionless group
$\mathcal{R}$	Ratio between nominal particle diameter and lattice cell size
R_K	Nominal radius of the Kth Lagrangian pointwise particle, m
Re	Reynolds number (fifth physically-meaningful dimensionless group)
Re$_{\text{p}}$	Particle Reynolds number
$\boldsymbol{S}$	Shear tensor, s^{-1}
$\mathscr{S}$	Source term operator, kg m^3
$\boldsymbol{U}_K$	Spatial coordinate of the Kth Lagrangian pointwise particle, m s^{-1}
W	Wendland function
$\boldsymbol{X}_K$	Spatial coordinate of the Kth Lagrangian pointwise particle, m
$\boldsymbol{X}_K^{\text{next}}$	Spatial coordinate of the Kth Lagrangian pointwise particle, approximated at the nearest lattice node, m
$\boldsymbol{c}_i$	i th discretized lattice velocity, m s^{-1}
c_{s}	Lattice speed velocity, m s^{-1}

d	Experimental bubble diameter, m
e_K	Contribution to $\varepsilon_{\ell,K}$
err	Error on the vortex's x coordinate, m
f	One-particle density function, kg m^{-3}
$f^{(eq)}$	Equilibrium one-particle density function, kg m^{-3}
$\mathbf{g}$	Acceleration of gravity, m s^2
g_H	First dimensionless group
g_K	Smoothing-kernel function of the Kth Lagrangian pointwise particle
h_{noz}	Distance between the tank bottom and the initial position of the bubble centre of mass, m
g_{noz}	Second dimensionless group
k	(as a subscript) Number of independent units of measure connected to the description of a generic physical process
n	Power-law index
n	(as a subscript) Number of physical parameters describing a generic physical process
n	Third dimensionless group
$\mathbf{n}$	Vector of indices labelling each of the points filling $\mathcal{I}_K^3(\mathbf{x})$
p	Pressure, Pa
$\mathbf{p}_K^{(a)}$	Coordinates of the points contributing to e_K, external set
$\mathbf{p}_K^{(b)}$	Coordinates of the points contributing to e_K, internal set
q	Gas injection flow rate, m^3 s^{-1}
q_i	i-th dimensional parameter describing a generic physical process
$\bar{q}_i$	i-th Scaled dimensional parameter describing a generic physical process
r	Generic length variable, m
t	(as a subscript) Number of "physically-meaningful" independent dimensionless groups describing a generic physical process
$\mathbf{u}$	Liquid-phase velocity, m s^{-1}
U_{LB}	Lattice velocity
U_{LB}^0	Reference lattice velocity
w_i	Weight of the ith component of the equilibrium particle distribution
$\mathbf{x}$	Discretized lattice spatial coordinate, m
$[\cdot]$	(if not used as functional argument) Extraction of the units of measure
EU	European Union
HLBM	Homogenized Lattice-Boltzmann Method
TS	Total solid content
UK	United Kingdom
WWAP	World Water Assessment Programme
WwTW	Wastewater treatment work

top of the above-mentioned difficulties to perform full-scale experiments, sludge is biologically and chemically hazardous; moreover, it is opaque, which makes it difficult to perform imaging experiments. For these reasons, an approach previously followed in the literature towards numerical modelling validation is to validate a model in the laboratory-scale first, and then, to apply the validated model to full-scale scenarios [11,12]. Recently, the Lattice-Boltzmann numerical method has been used for the first time to reproduce laboratory-scale, gas-mixed anaerobic digestion [13]. The above-mentioned approach towards validation was followed, and in fact the work provided the preliminary step of laboratory-scale validation.

The Lattice-Boltzmann algorithm is time-dependent, and solves the one-particle density function in the phase space. Density and velocity fields are recovered from the one-particle density function, and for low Mach numbers (viz., Ma $\longrightarrow$ 0), they are proved to obey the incompressible Navier-Stokes equations. This approach presents important advantages over other CFD methods in terms of numerical efficiency [37]. First, it is fully explicit—no iterative loop is required to solve velocity and pressure Navier-Stokes equations in a segregated way. Then, the non-linear and non-local parts of the Lattice-Boltzmann equation are decoupled—no differentiating scheme is required and thus parallel algorithms can be implemented for hundreds of cores with negligible loss of efficiency. Finally, a straightforward structure allows relatively simple extension towards a large variety of phenomena. [13] showed that a Lattice-Boltzmann model for laboratory-scale mixing in anaerobic digestion performed around 180 times better than finite-volume analogues.

The model proposed by [13] is an implementation of the Homogenized Lattice-Boltzmann Method (HLBM) [39], which is a non-pointwise Euler-Lagrangian method where the particle-fluid interface is resolved. This means that the lattice spacing is bounded to be at least 10 to 20 times smaller than particle diameter [34]:

$$2R_K > (10-20)\delta x , \qquad (1)$$

or:

$$\mathcal{R} := \frac{2R_K}{\delta x} > 10-20 , \qquad (2)$$

where R_K is the radius of a generic ("the Kth") particle (or bubble) in the system, and δx is the lattice cell size. The number of biogas bubbles is expected to be "low" for gas-mixed anaerobic digestion (with "low" meaning " $\lesssim$ 10" for laboratory-scale, Dapelo et al. [10,13], and " $\lesssim 10^4$" for full-scale, Dapelo and Bridgeman [11,12]). Under these condition, the Euler-Lagrange model is in general preferrable over other approach, i.e., the Euler-Euler [1]. In particular, the laboratory case presented a high ratio of bubble diameter over computational domain dimension (viz., ~ 0.1), and therefore the close-to-interface fluid phase flow could have exerted a non-negligible effect; under these circumstance, the HLBM is preferrable over non-interface-solving methods for the obvious reason that it resolves the particle surface and reproduces the fluid phase flow around it in greater detail. However, this ratio is bound to fall considerably when the computational domain size grows, with the consequence that the level of detail consisting of a precise reproduction of the close-to-interface fluid flow is no longer relevant. On the other hand, the requirement of 10 to 20 lattice sizes per particle diameter poses a burden in terms of computational expense: for instance, a pilot-scale model of 1 m^3 size would require around 8 billion cells in the face of around 10 million required for a laboratory-scale. This expense is likely to be unnecessary because of the low ratio of bubble diameter over computational domain in the pilot and full-scale (viz., $\lesssim 10^{-2}$), and hence, it is important to develop more economic models in order to fully take advantage of Lattice-Boltzmann's potentialities.

In the pointwise Euler-Lagrange model, each particle is treated as a material point, and particle-fluid interface is not resolved [1]; therefore, contrarily to HLBM, the pointwise Euler-Lagrange model is free from constraints on the minimum number of lattice sizes per bubble diameter. As such, the Euler-Lagrangian model is potentially a much cheaper alternative to the HLBM. In other words, the hypothesis that the particle dimension remains negligible means [42]:

$$2R_K \ll \delta x , \qquad (3)$$

or:

$$\mathcal{R} := \frac{2R_K}{\delta x} \ll 1 , \qquad (4)$$

Eqs. (4) and (2) show the existence of a gap in modelling, which is not covered by either HLBM and the pointwise Euler-Lagrange model. In fact, the hypothesis of Eq. (4) was found to be violated in [10] as well as in the laboratory-scale simulations described in this article. In [10], this problem was shown to determine a loss of grid independence in the most refined meshes, but was partially solved by generating larger cells where bubbles were expected to pass through, which means throughout the vertical axis above the nozzle. However, this solution cannot be extended to Lattice-Boltzmann modelling as only uniform cubic grids are applicable, and lattice size is bound to be constant throughout the computational domain. Two alternative solutions are proposed to resolve this specific problem and, in general, to fill the gap in modelling evidenced by Eqs. (4) and (2). The first consists of a smoothed-kernel Euler-Lagrangian model, similar to the one described in [15], which is specifically designed to simulate Lagrangian particles when the assumption in Eq. (4) does not hold. A further, novel, approach is described in this paper and consists of applying the Buckingham π theorem [7,40]. This approach consists of scaling the relevant dynamics parameters in order to keep $\mathcal{R}$ constant whilst maintaining the same physics. The underlying assumption, which was tested in the work described within this article, was that independence loss is imputable to a variable systematic error depending on $\mathcal{R}$: keeping the latter constant would thus allow to have the same systematic error in all the runs, and hence, to retrieve grid independence. An analysis of the grid independence in the scaled and unscaled cases for laboratory and pilot-scale setups of 1 m^3 follows. The laboratory-scale runs are performed due to the necessity of validating the model against experimental data and comparing it to HLBM in terms of the balance of accuracy over numerical expense. The pilot-scale runs are executed to test the behaviour of the model for setups, which are more similar to the final goal of the full-scale scenarios. A comparison with laboratory-scale data is traced, and the two approaches of smoothed-kernel model and dimensional scaling are then compared. All the simulations reported within this article were performed through the OpenLB open-source version 1.3-1 code www.openlb.net [22].

This article is structured as follows. On Section 2, the scaling mechanism based on the buckingham-π theorem is explained. Then, the Euler-Lagrangian Lattice-Boltzmann model is described on Section 3. The specifications the numerical simulations performed in the work described within this article are reported on Section 4. The presentation of the results follows on Section 5, and then, a discussion is performed on Section 6. Finally, conclusions are drawn on Section 7.

2. Buckingham-π dimensional scaling

The Buckingham theorem [7,40] states that it is possible to completely describe an arbitrary physical process through a set of opportunely-chosen dimensionless parameters (or "groups"—examples are the Reynolds number, geometrical ratios,...) in place of the usual (and generally larger) set of dimensional parameters (e.g., tank diameter, flow rate,...). As such, any scaling of the physical parameters will describe the same physical process, provided that the set of dimensionless groups remains unchanged.

As an application of this, we introduced a scaling procedure, illustrated as below. A thorough description of the mechanism and of the justifications behind it is reported in the Supplementary Material.

1. Identify a set of linearly-independent dimensional parameters $(q_1, \ldots, q_n)$;
2. Build a set of linearly-independent dimensionless groups $(\pi_1, \ldots, \pi_{n-k})$ through Rayleigh's dimensional analysis (k be-

ing the number of independent units of measure involved in the process);
3. Build another set of "physically-meaningful" (meaning that their value is important to classify the underlying physics of the process, e.g., the Reynolds number classifies a flow as laminar or turbulent) dimensionless groups $(\pi_1^{\mathrm{P}}, \ldots, \pi_t^{\mathrm{P}})$ independent from the original set of dimensionless groups $(\pi_1, \ldots, \pi_{n-k})$, not necessarily linearly-independent and not necessarily describing the physical process completely;
4. Alter a first subset of dimensionless groups $(\pi_1, \ldots, \pi_s) \subset (\pi_1, \ldots, \pi_{n-k})$ to the desired values $(\bar{\pi}_1, \ldots, \bar{\pi}_s)$ in such a way that the numerical simulation of the process becomes easier, more stable or less computationally-expensive;
5. Tune a second subset of dimensionless groups $(\pi_{s+1}, \ldots, \pi_{s+t}) \subset (\pi_1, \ldots, \pi_{n-k})$ to $(\hat{\pi}_{s+1}, \ldots, \hat{\pi}_{s+t})$ in such a way that the subset of physically-meaningful dimensionless groups $(\pi_1^{\mathrm{P}}, \ldots, \pi_t^{\mathrm{P}})$ remains unaltered;
6. Arbitrarily choose the new desired values of k dimensional parameters and compute the new values for the whole set $(\bar{q}_1, \ldots, \bar{q}_n)$ from the new set $(\bar{\pi}_1, \ldots, \bar{\pi}_s, \hat{\pi}_{s+1}, \ldots, \hat{\pi}_{s+t}, \pi_{s+t+1}, \ldots, \pi_{n-k})$ of dimensionless groups.

3. Euler-Lagrange Lattice-Boltzmann modelling of sludge

3.1. Assumptions for modelling sludge

Sludge is made of water, organic and inorganic solid fragments and, for gas mixing, biogas bubbles. Solid fragments vary from molecular size (organic polymers) to some centimeters (grit or larger agricultural debris). As such, sludge is characterized by a variety of multiphase phenomena. Gas-liquid interaction gives rise to interphase (two-way) momentum transfer; gas-gas to bubble-bubble (four-way) transfer, bubble breakup and coalescence; liquid-solid to sedimentation, flotation and complex non-Newtonian rheology (e.g., pseudoplasticity, thixotropy, shear banding and yield stress).

Considering the complexity of the problem, a number of assumptions are needed in order to effectively model sludge. (*i*) Sedimentation and flotation are ignored because their typical timescales (months if not years) are some orders of magnitude larger than the mixing timescale (15 minutes every hour), which is relevant to the problem taken into consideration. (*ii*) As solid phase is now considered in suspension to liquid phase, both are modelled together as a liquid phase, with the remaining liquid-solid interaction being modelled as non-Newtonian rheology, with thixotropy being ignored because typical digestion timescales (15-30 days) are such larger than mixing timescales (see Section 3.1.1 below). (*iii*) Bubble coalescence, breakup and in general bubble-bubble interactions were not observed experimentally [10] nor found in previous numerical work[11,12], whilst liquid-gas interaction is fundamental as momentum transfer from buoying bubbles to liquid constitute the foundation of the mixing mechanism itself; as such, gas phase is modelled as a Lagrangian phase dispersed inside a continuous Eulerian phase (Euler-Lagrange model), and two-way coupling is defined.

3.1.1. Liquid phase

A common and effective way to model sludge rheology is to adopt the pseudoplastic model [45]: apparent viscosity μ and shear rate magnitude $|\dot{\gamma}|$ obey a power-law relationship:

$$\mu = K|\dot{\gamma}|^{n-1}, \tag{5}$$

where n is the power-law coefficient which obeys the constraint $n < 1$, and K is the consistency coefficient. Divergence at zero shear rate and unphysical near-zero apparent viscosity values at high

TS (%)	K (Pa s^n)	n (–)	$\|\dot{\gamma}\|$ range (s^{-1})	$\mu_{\min}$ (Pa s)	$\mu_{\max}$ (Pa s)	Density (kg m^{-3})
2.5	0.042	0.710	226–702	0.006	0.008	1000.36
5.4	0.192	0.562	50–702	0.01	0.03	1000.78
7.5	0.525	0.533	11–399	0.03	0.17	1001.00
9.1	1.052	0.467	11–156	0.07	0.29	1001.31
12.1	5.885	0.367	3–149	0.25	2.93	1001.73

shear rates are avoided through introduction of $\mu_{\min}$ and $\mu_{\max}$, respectively a minimum and a maximum boundary for the apparent viscosity.

The power-law coefficient of Eq. (5) depend on total solid content (TS) and temperature. [25] measured sludge power-law coefficients at different TS values, and the results are reported in Table 1.

Under mesophilic conditions, the temperature is rigorously kept constant at 35 °C, and hence dependence on temperature can be ignored. As deviation of sludge density from that of water at 35 °C (994 kg m^{-3}) accounts for less than 1%, the value of 1000 kg m^{-3} is assumed for the sake of simplicity.

3.2. Lattice-Boltzmann description of the liquid phase

Whilst other CFD methods directly solve the observable fields of velocity, pressure and density, the Lattice-Boltzmann solves the one-particle density function $f(\boldsymbol{x}, \boldsymbol{c}, t)$ in the phase space. $f(\boldsymbol{x}, \boldsymbol{c}, t)$ is evaluated at the point of the phase space $(\boldsymbol{x}, \boldsymbol{c})$, and describes the probability of finding one particle of fluid within the elemental cube $(\boldsymbol{x}, \boldsymbol{x} + d\boldsymbol{x})$ with a velocity comprised within the cubic interval $(\boldsymbol{c}, \boldsymbol{c} + d\boldsymbol{c})$ at a time t. As such, the Lattice-Boltzmann is a mesoscopic model. The observable fields (viz., density, velocity and shear stress) correspond to the particle density function's zeroth, first and second moments respectively [24]:

$$\rho = \int f\, d\boldsymbol{c}\,; \tag{6}$$

$$\rho\,\boldsymbol{u} = \int f\,\boldsymbol{c}\, d\boldsymbol{c}\,; \tag{7}$$

$$\rho\,\boldsymbol{u} \otimes \boldsymbol{u} - \sigma = \int f\,\boldsymbol{c} \otimes \boldsymbol{c}\, d\boldsymbol{c}\,. \tag{8}$$

The Boltzmann equation is solved:

$$(\partial_t + \boldsymbol{c} \cdot \nabla) f = \mathscr{C}[f]\,. \tag{9}$$

The equation above is nothing else than a continuity equation for f in the phase space, plus the contribution of inter-particle collisions as a source-sink term. In the diluted gas hypothesis, the collision operator $\mathscr{C}$ models the effect of binary particle collisions. Despite being potentially extremely complicated, $\mathscr{C}$ can be drastically simplified following the Bhatnagar-Gross-Krook (BGK) hypothesis [3], which substitutes the detail of all the possible particle collisions with an isotropic average, relaxing towards the equilibrium particle density function $f^{(eq)}$:

$$\mathscr{C}[f] = -\frac{1}{\tau}\left(f - f^{(eq)}\right)\,, \tag{10}$$

where τ is the relaxation time, and $f^{(eq)}$ is the Maxwell equilibrium distribution [24]:

$$f^{(eq)}(\boldsymbol{x},\, t) := \rho(\boldsymbol{x},\, t)\left(\frac{1}{2\pi c_s^2}\right)^{3/2} \exp\left\{-\frac{[\boldsymbol{u}(\boldsymbol{x},\, t)]^2}{2c_s^2}\right\}\,, \tag{11}$$

where c_s is the sound speed, and density and velocity are evaluated through Eqs. (6) and (7).

Space is discretized as a cubic lattice of lattice size δx, time is subdivided into timesteps of duration δt, and the three-dimensional velocity space is discretized as a finite set of vectors $\{\boldsymbol{c}_0, \ldots, \boldsymbol{c}_{q-1}\}$ pointing to the zeroth, first, second and third neighbours of a generic lattice site, with moduli respectively $\sqrt{0}$, $\sqrt{1}$, $\sqrt{2}$ and $\sqrt{3}$ times $\delta x/\delta t$. A given discretization is conventionally identified by a tag DdQq, with d the dimensionality of the problem and q the dimension of the discrete velocity space. The one-particle distribution function is now defined over the discretized lattice and timesteps, and rewritten as a set $f_i(\boldsymbol{x}, t)$, each f_i indicating the probability of finding a particle with the corresponding discretized velocity $\boldsymbol{c}_i$. The integrals in Eqs. (6), (7) and (8) are substituted with summations over the velocity set:

$$\rho = \sum_i f_i\,; \tag{12}$$

$$\rho\,\boldsymbol{u} = \sum_i f_i\,\boldsymbol{c}_i\,; \tag{13}$$

$$\rho\,\boldsymbol{u} \otimes \boldsymbol{u} - \sigma = \sum_i f_i\,\boldsymbol{c}_i \otimes \boldsymbol{c}_i\,. \tag{14}$$

The error arising from the discretization of the velocity set is cancelled by writing the Maxwell equilibrium function (Eq. (11)) in terms of orthonormal Hermite polynomials up to the second order:

$$f_i^{(eq)} = w_i\,\rho\left[1 + \frac{\boldsymbol{u} \cdot \boldsymbol{c}_i}{c_s^2} + \frac{(\boldsymbol{u} \cdot \boldsymbol{c}_i)^2 - c_s^2 u^2}{2c_s^4}\right]\,, \tag{15}$$

where the weights w_i and the sound speed are defined in a standard way for the specific DdQp lattice, with the latter usually being defined as:

$$c_s := \frac{1}{\sqrt{3}}\frac{\delta x}{\delta t}\,. \tag{16}$$

Density and velocity are evaluated through the discretized version of Eqs. (6) and (7) [24]. In this way, the adiabatic dynamics with a compressibility error of Ma^2 with Ma being the Mach number, is recovered. The Boltzmann Eq. (9), considering the BGK assumption (Eq. (10)), becomes the Lattice-Boltzmann Equation:

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i\,\delta t,\, t + \delta t) = f_i(\boldsymbol{x},\, t) - \frac{1}{\tau}\left[f_i(\boldsymbol{x},\, t) - f_i^{(eq)}(\boldsymbol{x},\, t)\right]\,. \tag{17}$$

The updating process for each lattice site at a given timestep comprises two steps. First: a local, non-linear *collision*:

$$\Xi_i(\boldsymbol{x},\, t) = f_i(\boldsymbol{x},\, t) - \frac{1}{\tau}\left[f_i(\boldsymbol{x},\, t) - f_i^{(eq)}(\boldsymbol{x},\, t)\right]\,. \tag{18}$$

Then: a linear, non-local *streaming*:

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i,\, t + 1) = \Xi_i(\boldsymbol{x},\, t)\,. \tag{19}$$

In the limit Ma $\ll$ 1, the Lattice-Boltzmann Eq. (17) reproduces the incompressible Navier-Stokes equations [24], with pressure and kinematic viscosity defined as:

$$p := \rho c_s^2\,, \qquad \nu := c_s^2\left(\tau - \frac{\delta t}{2}\right)\,. \tag{20}$$

3.2.1. Non-dimensionalization and stability

While humans understand measurement units, computers deal only with dimensionless quantities. In Lattice-Boltzmann, the non-dimensionalization procedure should be performed accurately in order to strike a convenient balance between stability, accuracy and computational expense [24]. The standard way to perform non-dimensionalization in OpenFOAM is through using δx, δt and a reference value of density as the conversion factors for length, time

and density respectively. Therefore, viscosity and reference velocity are converted to:

$$\nu_{\text{LB}} = \nu \frac{\delta t}{\delta x^2}, \qquad U_{\text{LB}} = U \frac{\delta t}{\delta x}. \tag{21}$$

Combining Eqs. (16) and (21), we also find the following relation between Mach number and lattice velocity:

$$\text{Ma} \equiv \frac{U}{c_s} = \sqrt{3}\, U_{\text{LB}}, \tag{22}$$

which shows that the determination of the Mach number depends on the choices of δx and δt through Eq. (21). Finally, Eqs. (20) and (21) allow the determination of the values of τ and $\tau_{\text{LB}} \equiv \tau/\delta t$ from the values of ν and the choices of δx and δt:

$$\tau_{\text{LB}} = \sqrt{3}\, \nu \frac{\delta t}{\delta x^2} + \frac{1}{2}. \tag{23}$$

The value of τ_{LB} consequently decreases if ν decreases, until the limit case of $\tau_{\text{LB}} \longrightarrow 1/2$ for $\nu \longrightarrow 0$.

Stability depends on the values of U_{LB} (or, equivalently, Ma), and τ_{LB}. In particular, stability deteriorates when τ_{LB} approaches the limit value of 1/2, thereby showing that viscosity's dissipative effect improves Lattice-Boltzmann stability. A stability criterion is not available in Lattice-Boltzmann; however, the following recommendations for τ_{LB} and U_{LB} are given for the case of the BGK collision [24]:

$$\tau_{\text{LB}} > \frac{1}{2}; \tag{24}$$

$$U_{\text{LB}} \leq \begin{cases} 8(\tau_{\text{LB}} - 1/2), & \tau_{\text{LB}} \leq 0.55; \\ 0.4, & \tau_{\text{LB}} > 0.55. \end{cases} \tag{25}$$

Furthermore, the BGK collision has been found to be unstable in under-resolved turbulent flow [31]; therefore, the choice of a too large value of δx can negatively affect stability.

Accuracy depends on grid resolution (therefore, δx) and compressibility error (which is $\sim \text{Ma}^2$). Therefore, the desired level of grid resolution (δx) must be tuned together with a choice of δt in order to prevent the compressibility error (viz., Ma or, equivalently, U_{LB}) from increasing, and to maintain stability (viz., preventing τ_{LB} from becoming too small, and from U_{LB} from diverging from the recommendations of Eq. (25)). Of course, the computational expense is inversely proportional to both δx and δt.

Finally, an appropriate scaling of δt should be defined when the grid is coarsened or refined in order to control stability and the behaviour of the compressibility error. A common choice of scaling, which is followed in this work, is $\delta t \sim \delta x^2$, or "diffusive scaling". This ensures that τ_{LB} remains constant, thereby leaving the stability properties of the system unaltered. Furthermore, the spatial discretization error in Lattice-Boltzmann is $\sim \delta x^2$: the choice of diffusive scaling means that Ma $\sim \delta x$, thereby imposing the incompressibility error to be $\sim \delta x^2$, as the spatial discretization error. Therefore, the Lattice-Boltzmann algorithm is theoretically second-order accurate in space and first-order accurate in time when choosing diffusive scaling [24].

3.2.2. Lattice-Boltzmann rheology and turbulence modelling

Following [13], rheology is modelled as in [5], and turbulence through a Smagorinsky closure [18] Each timestep, at each lattice node, before the collision step:

1. The rate of shear tensor $S_{\alpha\beta} \equiv \frac{1}{2}(\partial_\alpha u_\beta + \partial_\beta u_\alpha)$ is evaluated through its relationship with the second momentum of the non-equilibrium term of the particle distribution function $f_i^{(1)}$ [24]:

$$\boldsymbol{S} = -\frac{1}{2\tau\rho c_s^2} \sum_i f_i^{(1)} \boldsymbol{c}_i \otimes \boldsymbol{c}_i, \tag{26}$$

and the dynamic viscosity $\mu_{\text{PL}}(\boldsymbol{x}, t)$ is calculated through Eq. (5) with the substitution:

$$|\dot{\gamma}| \equiv \sqrt{2\boldsymbol{S}:\boldsymbol{S}}; \tag{27}$$

2. The relaxation time is locally altered a first time $\tau \longrightarrow \tau_{\text{PL}}$ through inverting the second of Eqs. (20) with $\nu_{\text{PL}} \equiv \mu_{\text{PL}}/\rho$ in place of ν;

3. Rate of shear tensor $\boldsymbol{S}$ and shear rate magnitude $|\dot{\gamma}|$ are computed once again through re-applying Eq. (26) with τ_{PL} in place of τ, and Eq. (27) respectively, and the turbulent kinematic viscosity is computed through the Smagorinsky closure:

$$\nu_{\text{eff}} = \nu_{\text{PL}} + C_{\text{Smago}}|\dot{\gamma}|, \tag{28}$$

with the Smagorinsky constant C_{Smago} being set to 0.14;

4. The relaxation time is locally altered a second time $\tau_{\text{PL}} \longrightarrow \tau_{\text{eff}}$ through inverting the second of Eqs. (20) with ν_{eff} in place of ν.

3.3. Euler-Lagrange modelling of pointwise particles in Lattice-Boltzmann

Within the Euler-Lagrange models, the dispersed phase is treated as an ensemble of pointwise elements, which are conventionally called "particles" (in the case of the work reported in this article, each "particle" corresponds to a gas bubble) [42]. Each (spherical) particle $\mathscr{P}_K$ present inside the computational domain at a given time is formally represented as a tuple composed of coordinate $\boldsymbol{X}_K$, velocity $\boldsymbol{U}_K$, acceleration $\boldsymbol{A}_K$, radius R_K and mass M_K:

$$\mathscr{P}_K \equiv (\boldsymbol{X}_K, \boldsymbol{U}_K, \boldsymbol{A}_K, R_K, M_K). \tag{29}$$

Further quantities may be added to the list depending on the complexity of the problem—for instance, moment of inertia, angular coordinate, velocity and acceleration may be added if rotational effects (e.g., lift) are important, or aspect ratios if significant deviations from sphericiy are present. However, rotational effects were found to be negligible for the application taken into consideration in the work described here, and deviations from sphericity were ignored [10,13].

Each particle was set to evolve separately through verlet integration of Newton's second law:

$$\boldsymbol{F}_K = M_K \boldsymbol{A}_K \tag{30}$$

(no Einstein summation rule here!), where $\boldsymbol{F}_K$ is the resultant of the forces acting on $\mathscr{P}_K$ (*forward-coupling*). For the problem described in this article, we have:

$$\boldsymbol{F}_K = \boldsymbol{F}_K^{\text{b}} + \boldsymbol{F}_K^{\text{a}} + \boldsymbol{F}_K^{\text{d}}, \tag{31}$$

where $\boldsymbol{F}_K^{\text{b}}$ is buoyancy:

$$\boldsymbol{F}_K^{\text{b}} = -\left(\frac{4}{3}\pi R_K^3 \rho - M_K\right)\boldsymbol{g} \tag{32}$$

with $\boldsymbol{g}$ being the acceleration of gravity, and $\boldsymbol{F}_K^{\text{a}}$ and $\boldsymbol{F}_K^{\text{d}}$ are the added mass and the drag force respectively. The last two are modelled according to the specific models described below (see Sections 3.3.1, 3.3.2 and 3.3.3). In general, they depend on both $\mathscr{P}_K$ and the observables from the liquid phase ρ, $\boldsymbol{u}$ and ν.

The effect of $\mathscr{P}_K$ on the surrounding liquid phase consists of a body force $-\boldsymbol{F}_K$ acting on the continuous phase (*back-coupling*). A body force can be included in the Lattice-Boltzmann framework in a standard way [17] through the addition of a source term in the Lattice-Boltzmann Eq. (17):

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i, t+1) = f_i(\boldsymbol{x}, t) - \frac{1}{\tau}\left[f_i(\boldsymbol{x}, t) - f_i^{(\text{eq})}(\boldsymbol{x}, t)\right]$$
$$+ \left(1 - \frac{1}{2\tau}\right)\mathscr{S}_i(\boldsymbol{x}, t), \tag{33}$$

where the source term $\mathscr{S}_i$ is defined as:

$$\mathscr{S}_i := w_i \left[\frac{\boldsymbol{c}_i - \boldsymbol{u}}{c_s^2} + \frac{(\boldsymbol{c}_i \cdot \boldsymbol{u})\boldsymbol{c}_i}{c_s^4} \right] \cdot \boldsymbol{\Phi} \ . \tag{34}$$

The forcing term $\boldsymbol{\Phi} \equiv \Sigma_K \boldsymbol{\Phi}_K$ ($\boldsymbol{\Phi}_K$ being the specific contribution from particle $\mathscr{P}_K$) depends on the specific model (see Sections 3.3.1, 3.3.2 and 3.3.3). The equilibrium velocity (Eq. (13)) is redefined as:

$$\rho \, \boldsymbol{u} = \sum_i f_i \boldsymbol{c}_i + \frac{1}{2} \sum_i \mathscr{S}_i \boldsymbol{c}_i \ . \tag{35}$$

The forcing model described above presents the advantage of avoiding spurious $\mathcal{O}(\text{Ma}^2)$ terms in the macroscopic equations of motion whilst keeping a nominal order of convergence of two [16].

3.3.1. Local model

Within this model, forward-coupling (viz., the computation of $\boldsymbol{F}_K$, Eq. (31)) occurs locally: liquid phase density $\rho(\boldsymbol{X}_K)$ and velocity $\boldsymbol{u}(\boldsymbol{X}_K)$ are interpolated linearly across the cells surrounding $\mathscr{P}_K$ position $\boldsymbol{X}_K$, while kinematic viscosity $\nu(\boldsymbol{X}_K^{\text{next}})$ is computed at the nearest lattice node coordinate $\boldsymbol{X}_K^{\text{next}}$ for the sake of simplicity. The added mass in Eq. (31) is modelled as:

$$\boldsymbol{F}_K^{\text{a}} = -\frac{1}{2} \rho(\boldsymbol{X}_K) \frac{4}{3} \pi R_K^3 \frac{\mathrm{d}}{\mathrm{d}t} [\boldsymbol{U}_K - \boldsymbol{u}(\boldsymbol{X}_K)] \simeq -\frac{1}{2} \rho(\boldsymbol{X}_K) \frac{\pi}{6} d_K^3 A_K \ , \tag{36}$$

and the drag force as:

$$\boldsymbol{F}_K^{\text{d}} = -\frac{1}{2} \rho(\boldsymbol{X}_K) C_{\text{d}}(\text{Re}_{\text{p}}) \pi R_K^2 \, |\boldsymbol{U}_K - \boldsymbol{u}(\boldsymbol{X}_K)| [\boldsymbol{U}_K - \boldsymbol{u}(\boldsymbol{X}_K)] \ . \tag{37}$$

The drag coefficient C_{d} is modelled according to [30]:

$$C_{\text{d}}(\xi) = a + \frac{b}{\xi} + \frac{c}{\xi^2}$$

$$(a, b, c) = \begin{cases} (0.0, 24.0, 0.0), & \xi \leq 0.1 \ ; \\ (3.69, 22.73, 0.0903), & 0.1 < \xi \leq 1.0 \ ; \\ (1.222, 29.16667, 3.8889), & 1.0 < \xi \leq 10 \ ; \\ (0.6167, 46.5, 116.67), & 10 < \xi \leq 100 \ ; \\ (0.4644, 98.33, -2778.0), & 100 < \xi \leq 1000 \ ; \\ (0.357, 148.62, -4.75e4), & 1000 < \xi \leq 5000 \ ; \\ (0.46, -490.546, 57.87e4), & 5000 < \xi \leq 10,000 \ ; \\ (0.5191, -1662.5, 5.4167e6), & \xi > 10,000 \ . \end{cases} \tag{38}$$

The particle Reynolds number Re_{p} is evaluated as:

$$\text{Re}_{\text{p}} = \frac{2R_K |\boldsymbol{U}_k - \boldsymbol{u}(\boldsymbol{X}_K)|}{\nu_{\text{PL}}(\boldsymbol{X}_K^{\text{next}})} \ , \tag{39}$$

where ν_{PL} is computed as per in Section 3.2.2.

Back-coupling occurs through applying $-\boldsymbol{F}_K$ to the most near cell from $\mathscr{P}_K$. This is done trough defining the forcing $\boldsymbol{\Phi}_K$ in Eq. (34) as (see Krüger et al. [24]):

$$\boldsymbol{\Phi}_K(\boldsymbol{x}) = \begin{cases} -\boldsymbol{F}_K \ , & \boldsymbol{x} = \boldsymbol{X}_K^{\text{next}} \ ; \\ \boldsymbol{0} \ , & \text{otherwise.} \end{cases} \tag{40}$$

3.3.2. First-neighbour model

Within this model, forward-coupling occurs exactly as per in the local (Section 3.3.1). Back-coupling occurs through applying $-\boldsymbol{F}_K$ to the most near cell from $\mathscr{P}_K$, and the first neighbour cells of the latter, with weights normalized to 1, as in [27]:

$$\boldsymbol{\Phi}_K(\boldsymbol{x}) = -\boldsymbol{F}_K \, \delta_{\delta x}(\boldsymbol{x} - \boldsymbol{X}_K) \ . \tag{41}$$

$\delta_{\delta x}$ is a discrete delta function of width δx (Peskin [33], Eq. (6.28)).

3.3.3. Smoothing-kernel model

The presence of Lagrangian particles of non-negligible size was accounted for by [15] in a finite-volume CFD model for atomization process through a smoothing-kernel procedure, which in turn produced: (i) the field equations for the continuous phase (viz., the Navier-Stokes) were weighted over volumetric phase fraction; (ii) a modified formula for the drag coefficient, dependent on the volumetric phase fraction, was used in place of the one of Eq. (37); and (iii) back-coupling occurred through smooth transfer of momentum from the particle to the cells inside one smoothing-kernel length from particle's centre of mass. Conversely, forward-coupling remained as per in the local case (see Section 3.3.1).

The approach described within this article is a translation of the model presented by [15] from the finite-volume to the Lattice-Boltzmann framework, with the following differences: (i) the field equations for the continuous phase (viz., the Lattice-Boltzmann Eq. (17)) is not weighted over volumetric phase fraction because such arrangement is not yet available for the Lattice-Boltzmann at the time the work is carried out; and (ii) the discretized smoothing procedure is different because, in [15], the back-coupling momentum term becomes unphysically negligible and dependent on lattice size for $\mathcal{R} < 1$ after smoothing.

The smoothing-kernel model used in this work is defined as follows. A smoothing-kernel function is defined for every $\mathscr{P}_K$ as a compact-support polynomial normalized to 1, as in [15] apart from the bespoke normalizing factor:

$$g_K(r) = \mathcal{C}_K \, W\!\left(\frac{r}{\mathcal{L}_K}\right) \ , \tag{42}$$

where $\mathcal{L}_K$ is the kernel radius (in practice, a multiple of R_K), $\mathcal{C}_K$ the normalized factor to be recalculated at every timestep in order to keep g_K normalized to 1, and $W(\xi)$ is the Wendland function:

$$W(\xi) = \begin{cases} (4\xi + 1)(1 - \xi)^4 \ , & 0 \leq \xi \leq 1; \\ 0 \ , & \xi > 1. \end{cases} \tag{43}$$

The choice of adopting two-way coupling was underpinned by the fact that bubbles were observed to be well-distanced each other [10]: the observed space between the bubbles also justifies the following hypothesis, that bubbles' centres of mass never come closer to each other than the sum of the respective kernel lengths at any time. Under this hypothesis, volumetric phase fraction at any lattice point is determined by the presence (or the absence) of one particle at most. As such, the (smoothed) liquid phase fraction at $\mathscr{P}_K$'s location $\varepsilon_{\ell,K}$ is computed as the summation of contributions e_K coming from each node $\boldsymbol{x}$, weighted by g_K:

$$\varepsilon_{\ell,K} = \sum_{\boldsymbol{x}} e_K(\boldsymbol{x}) \, g_K(|\boldsymbol{x} - \boldsymbol{X}_K|) \ . \tag{44}$$

The idea behind the computation of each contribution e_K is to consider a number of points within the cube $\mathcal{I}^3(\boldsymbol{x})$ of side δx centered at $\boldsymbol{x}$, and determine the value of e_K depending on how many points fall within one particle radius R_K from $\boldsymbol{X}_K$. In order to do so, we proceed as follows. Let be $\mathcal{B}_K^{\square}$ the bounding cube around the sphere centered at $\boldsymbol{X}_K$ of radius $\mathcal{L}_K$; then, let the parallelepiped $\mathcal{I}_K^3(\boldsymbol{x})$ be defined as the intersection $\mathcal{I}_K^3(\boldsymbol{x}) := \mathcal{I}^3(\boldsymbol{x}) \cap \mathcal{B}_K^{\square}$ (see Fig. 1):

$$\mathcal{I}_K^3(\boldsymbol{x}) = \left[\Delta_x^-, \, \Delta_x^+ \right] \times \left[\Delta_y^-, \, \Delta_y^+ \right] \times \left[\Delta_z^-, \, \Delta_z^+ \right] \ , \tag{45}$$

where $\boldsymbol{\Delta}^- \equiv \left(\Delta_x^-, \, \Delta_y^-, \, \Delta_z^- \right)$ and $\boldsymbol{\Delta}^+ \equiv \left(\Delta_x^+, \, \Delta_y^+, \, \Delta_z^+ \right)$ are respectively defined as:

$$\boldsymbol{\Delta}^- = \left(\max\left(X_K - \mathcal{L}_K, \, x - \frac{\delta x}{2} \right), \ \max\left(Y_K - \mathcal{L}_K, \, y - \frac{\delta x}{2} \right), \right.$$
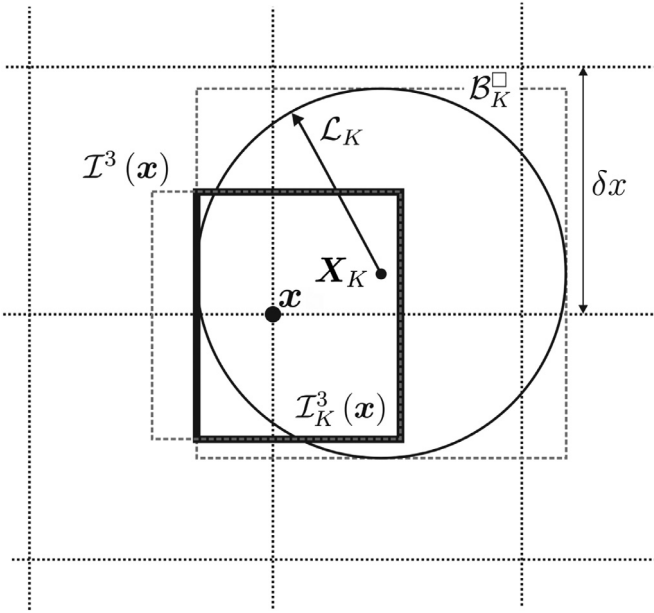$$\left. \max\left(Z_K - \mathcal{L}_K, \, z - \frac{\delta x}{2} \right) \right) \ ; \tag{46}$$

Fig. 1. Construction of $\mathcal{I}^3_K(\boldsymbol{x})$.
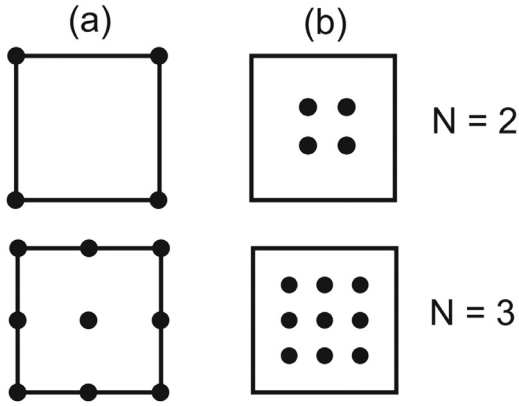
Fig. 2. Distributing N points throughout $\mathcal{I}^3_K(\boldsymbol{x})$.

$$\boldsymbol{\Delta}^+ = \left(\min\left(X_K + \mathcal{L}_K, x + \frac{\delta x}{2} \right), \ \min\left(Y_K + \mathcal{L}_K, y + \frac{\delta x}{2} \right), \right.$$
$$\left. \min\left(Z_K + \mathcal{L}_K, z + \frac{\delta x}{2} \right) \right) . \qquad (47)$$

$\mathcal{I}^3_K(\boldsymbol{x})$ is filled regularly and isotropically with N^3 points, N per co-ordinate direction. As shown in Fig. 2, there are two ways to do so, which are labelled as "(a)" and "(b)" respectively. As such, let the integer vector $\boldsymbol{n}$ be defined as $\boldsymbol{n} := (i,j,k)$ with i, j, $k \in 1, \ldots, N$. Then, two sets of points, $\left\{ \boldsymbol{p}^{(a)}_K(\boldsymbol{n}, \boldsymbol{x}) \right\}$ and $\left\{ \boldsymbol{p}^{(a)}_K(\boldsymbol{n}, \boldsymbol{x}) \right\}$, are defined as:

$$\boldsymbol{p}^{(a)}_K(\boldsymbol{n}, \boldsymbol{x}) := \boldsymbol{\Delta}^- + \frac{1}{N-1}(\boldsymbol{n} - (1, 1, 1)) \cdot \left(\boldsymbol{\Delta}^+ - \boldsymbol{\Delta}^- \right), \qquad (48)$$

and:

$$\boldsymbol{p}^{(b)}_K(\boldsymbol{n}, \boldsymbol{x}) := \boldsymbol{\Delta}^- + \frac{1}{N+1}\boldsymbol{n} \cdot \left(\boldsymbol{\Delta}^+ - \boldsymbol{\Delta}^- \right). \qquad (49)$$

Then, after defining a colour function Γ_K as:

$$\Gamma_K(\boldsymbol{\xi}) := \begin{cases} 1, & |\boldsymbol{\xi} - \boldsymbol{X}_K| \leq R_K ; \\ 0, & \text{otherwise}, \end{cases} \qquad (50)$$

e_K is defined as:

$$e_K(\boldsymbol{x}) = \frac{1}{2} \sum_{\{\boldsymbol{n}\}} \left\{ \Gamma_K\left[\boldsymbol{p}^{(a)}_K(\boldsymbol{n}, \boldsymbol{x}) \right] + \Gamma_K\left[\boldsymbol{p}^{(b)}_K(\boldsymbol{n}, \boldsymbol{x}) \right] \right\} . \qquad (51)$$

The drag force is then defined as in [15]:

$$\boldsymbol{F}^d_K = -\frac{1}{2\varepsilon^{1.65}_{\ell,K}} \rho(\boldsymbol{X}_K) C_d(\varepsilon_{\ell,K} \text{Re}_p) \pi R^2_K |\boldsymbol{U}_K - \boldsymbol{u}(\boldsymbol{X}_K)|[\boldsymbol{U}_K - \boldsymbol{u}(\boldsymbol{X}_K)] . \qquad (52)$$

The added mass force is defined as in the previous models, as in Eq. (36).

The back-coupling by applying $-\boldsymbol{F}_K$ to the cells inside $\mathscr{P}_K$'s kernel length:

$$\boldsymbol{\Phi}_K(\boldsymbol{x}) = -\boldsymbol{F}_K \, g_K(|\boldsymbol{x} - \boldsymbol{X}_K|) . \qquad (53)$$

4. Numerical simulations

4.1. Transparent sludge substitute

The employment of a transparent sludge substitute has been a common strategy to reproduce sludge's hydrodynamic behaviour while maintaining transparency and thus the ability of performing optical measurements [45]. Accordingly, in [10], sludge was modelled through a 2 g ℓ^{-1} water solution of Sigma-Aldrich 419,338 sodium carboxymetyl cellulose (CMC) with average molar weight of 700,000. The power-law coefficient and consistency index were measured through a rheometer within the shear rate interval 100-500 s^{-1}, and resulted to be 0.805 and 0.054 Pa s^n respectively. As such, transparent sludge's rheological characteristics resulted to be intermediate between 2.5 and 5.4% TS sludge (see Table 1).

Following what concluded in Section 3.1.1, sludge density was set to $\rho_L = 1000$ kg m^{-3}, while bubble density was set to the density of air at atmospheric pressure $\rho_P = 1$ kg m^{-3}.

4.2. Laboratory-scale simulations

The experimental apparatus used in [10] and reproduced in the work presented within this article consisted of a 4 l cylindrical tank of diameter $D = 20$ cm, filled with CMC solution up to a height of $H = 13$ cm. Bubbles of diameter $d = 7.01$ mm were created one at a time, with the centre of mass being $h_{noz} = 1$ cm from the centre of the domain's bottom face, at time intervals chosen to match a gas inlet flow rate of $q = 5.30$ ml s^{-1} (see Eq. (54))

$$\Delta t_{inj} = \frac{\pi d^3}{6 \, q} . \qquad (54)$$

The bubbles reaching the top face were removed from the system. A schematics of the computational domain is reported in Fig. 3.

The boundary conditions were defined as follows: zero-velocity Bouzidi [4] at the bottom and the lateral walls; free-slip [2,35] at the top. The initial conditions were set to zero velocity and constant density throughout the computational domain. For the smoothed-kernel runs, the kernel length was set to $\mathcal{L} = 1.5 \, d$. The dimensionless velocity U_{LB} was set according to diffusive scaling:

$$U_{LB} = U^0_{LB} \frac{N^0}{N} , \qquad (55)$$

with $U^0_{LB} = 0.3$ and $N^0 = 40$. Grids with a number of lattice nodes N along the diameter spanning between 40 and 200, corresponding to 36,936 and 4,515,701 overall nodes respectively, were generated. The simulations were run up to a simulated time of 21 s. The numerical work was performed on the same hardware of [13], viz., 18-core (36-thread) Intel Xeon E5-2695 v4 "Broadwell" (2.20 GHz clock, 8 GT/s QPI bus) processors, but on two instead of eight processors.
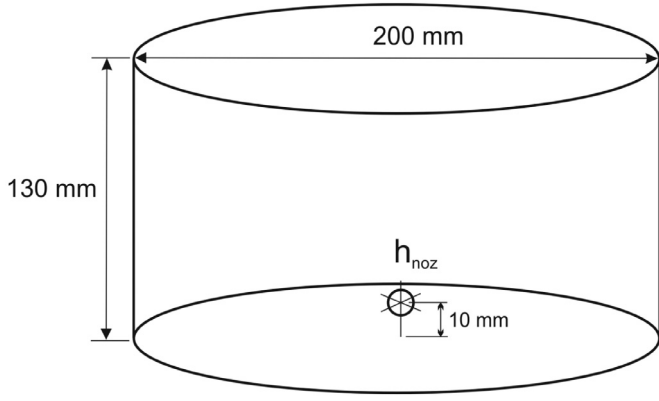
Fig. 3. Laboratory-scale computational domain [13].

4.3. Pilot-scale simulations

The setup was the same of the laboratory-scale case (Section 4.2 and Fig. 3), with the following differences: $D = 1$ m, $H = 1$ m, $h_{\text{noz}} = 10$ cm, $U_{\text{LB}}^0 = 0.1$, maximum value for N set to 120 corresponding to 976,835 cells, maximum simulated time of 121 s.

4.4. Current setting of the scaling mechanism

As emerged in Sections 4.2 and 4.3, the relevant dimensional parameter were ten:

$$(g, H, D, h_{\text{noz}}, q, d, K, n, \rho_{\text{L}}, \rho_{\text{P}}), \tag{56}$$

while the units of measure involved were three:

$$([L], [T], [M]). \tag{57}$$

Accordingly, following the Rayleigh's dimensional analysis (Point 2 in Section 2, also see Section A.2 in the Supplementary Material), the following repeating variables were chosen:

$$(g, q, d), \tag{58}$$

and the following seven dimensionless groups were defined:

$$g_{\text{H}} := \frac{H}{D}; \tag{59}$$

$$g_{\text{noz}} := \frac{h_{\text{noz}}}{D}; \tag{60}$$

$$n := n; \tag{61}$$

$$\pi_1 := \frac{d^5 g}{q^2}; \tag{62}$$

$$\pi_2 := \frac{d^{n+2}}{g^{n-2} K^2}; \tag{63}$$

$$\pi_3 := \frac{\rho_{\text{L}}}{\rho_{\text{P}}}; \tag{64}$$

$$\pi_4 := \frac{D}{d}. \tag{65}$$

According to Point 3 of Section 2, the following physically-meaningful dimensionless groups were defined:

$$\text{PNu} := \frac{P_{\text{d}} d}{\rho_{\text{L}} \langle \nu \rangle^3}; \tag{66}$$

$$\text{PNuU} := \frac{P_{\text{d}}}{\rho_{\text{L}} \langle \nu \rangle \langle u \rangle^2 d}; \tag{67}$$

$$\text{PU} := \frac{P_{\text{d}}}{\rho_{\text{L}} \langle u \rangle^3 d^2}; \tag{68}$$

Table 2
Types of scaling.

Label	$\hat{\pi}_i$	π_i^{P}
0	—	—
1PNu	π_1	PNu
2PNu	π_2	PNu
2PU	π_2	PU
2Re	π_2	Re
12PNu-Q	π_1, π_2	PNu, Q
12PNu-Re	π_1, π_2	PNu, Re
12PNuU-Re	π_1, π_2	PNuU, Re
12PU-Q	π_1, π_2	PU, Q
12Q-Re	π_1, π_2	Q, Re

$$\text{Q} := \frac{q}{\langle \nu \rangle d}; \tag{69}$$

$$\text{Re} := \frac{\langle u \rangle D}{\langle \nu \rangle}. \tag{70}$$

$< u >$ is the characteristic velocity—in this case, the approximated average velocity, which was computed by approximating the velocity flow patterns as constant rising velocity along a central column of diameter d, and zero elsewhere:

$$\langle u \rangle = u_{\text{d}} \frac{d^2}{D^2}, \tag{71}$$

where u_{d} is the approximated asymptotic liquid phase drag velocity, and was computed by solving numerically Eq. (30) for $\boldsymbol{A}_K = \boldsymbol{0}$.

$< \nu >$ is the characteristic kinematic viscosity, which was computed substituting the characteristic shear rate $\langle |\dot{\gamma}| \rangle$ into Eq. (5), with the former being defined as:

$$\langle |\dot{\gamma}| \rangle = \frac{u_{\text{d}} - 0}{2d}. \tag{72}$$

P_{d} is the total drag power being dissipated at a given timestep, which was computed as:

$$P_{\text{d}} = N_{\text{bubbles}} |\boldsymbol{F}^{\text{d}}| u_{\text{d}}, \tag{73}$$

where N_{bubbles} is the total number of bubbles present within the computational domain at a given timestep, and was evaluated as:

$$N_{\text{bubbles}} = \frac{q}{\pi d^3 / 6} \frac{H - h_{\text{noz}}}{u_{\text{d}}}. \tag{74}$$

Point 4 if Section 2 was implemented as follows. In order to keep $\mathcal{R}$ constant throughout the different runs, d was adjusted in the following way:

$$d \longrightarrow \bar{d} = d \frac{N^0}{N}, \tag{75}$$

while the other repeating variables (Eq. (58)) were kept unaltered. At the same time, the dimensionless group π_4 (Eq. (65)) was altered as:

$$\pi_4 \longrightarrow \bar{\pi}_4 = \pi_4 \frac{N}{N^0}. \tag{76}$$

Finally (Points 5 and 6 of Section 2, the map χ of Eq. (A.7) was applied in ten possible ways with the related systems of equations being solved numerically, as illustrated on Table 2.

4.5. Numerical setting

All the simulations were performed on a D3Q27 lattice. The value of ν_{LB} was computed from $< \nu >$. The non-dimensionalization defined from the choices of U_{LB} and N met the recommendation of Eq. (24), but not the one of Eq. (25). However, the introduction of a Smagorinsky term as in Eq. (28) resulted in an increase of the local values of ν_{LB} and therefore exerted a stabilising effect. No stability problems were observed in the numerical runs.
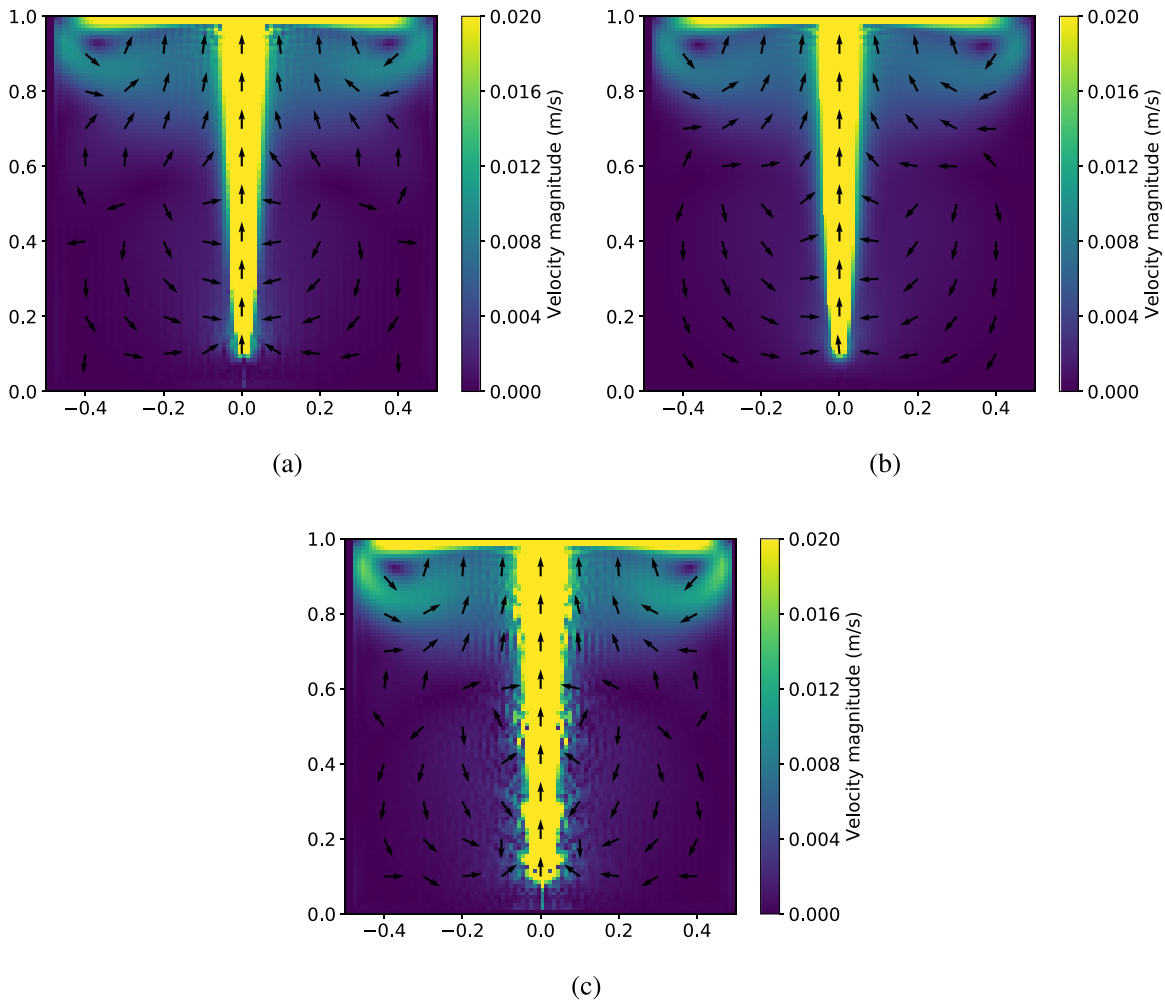
(a)

(b)

(c)

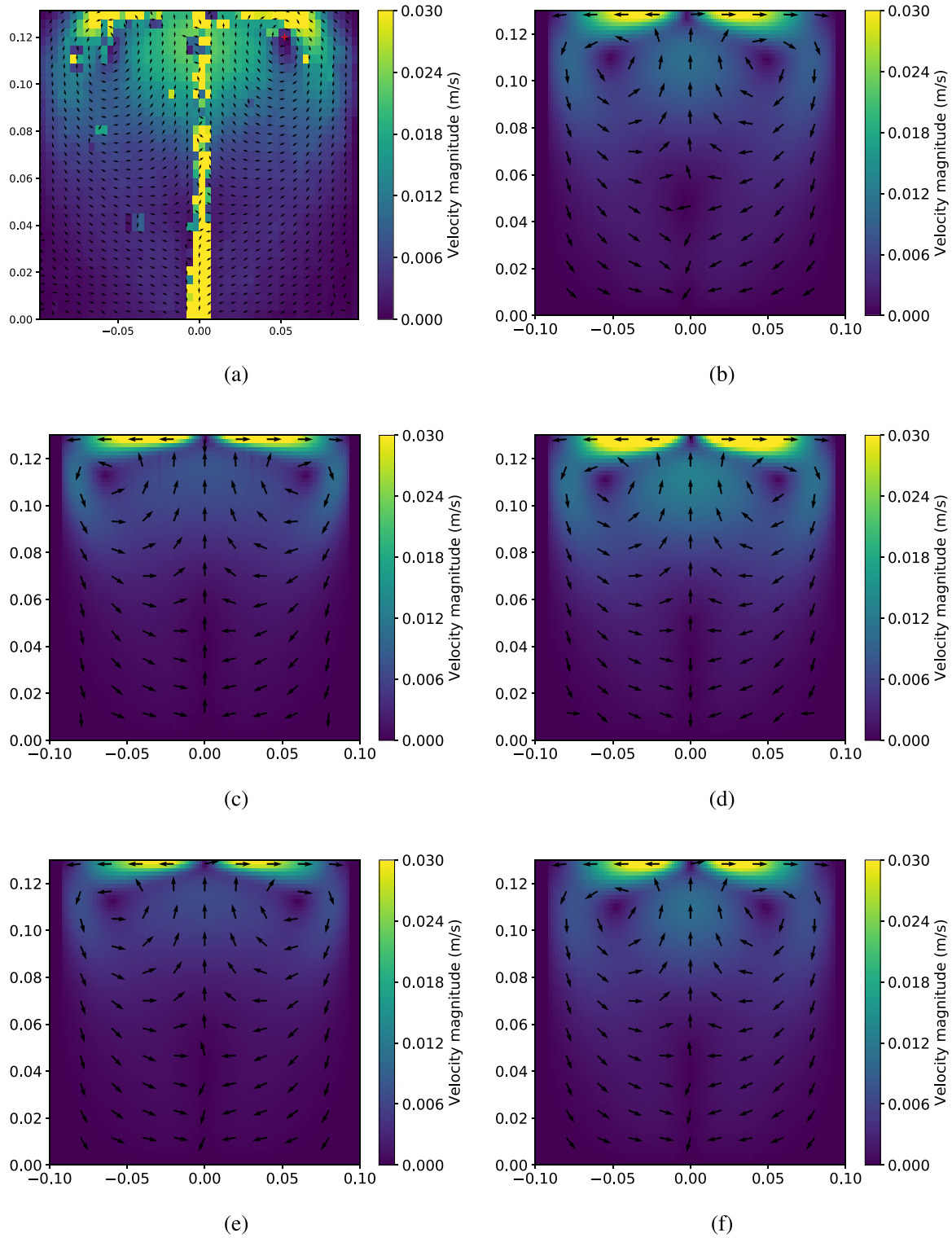**Fig. 4.** Flow patterns, pilot-scale, axial plane. Numerical data are displayed at their last timestep, and computational expense is reported. (a) Local model. 3143 CPUs. (b) First-neighbour model. 3334 CPUs. (c) Smoothed-kernel model. 2762 CPUs.

5. Results

5.1. Pilot-scale runs

In Fig. 4, example results of the unscaled pilot-scale runs are reported for $N = 100$. All the three Lagrangian models described in Section 3.3 are tested. The flow patterns displayed very scarce dependence on the Lagrangian model, apart from a very slight increase in intensity for the smoothed-kernel model. The computational expense was found to undertake scarce variations, being comprised between 2700 and 3400 CPUs. The fact that the smoothed-kernel required slightly less computational expense than the other two model was surprising: it would have been expected the contrary because the smoothed-kernel required an additional cycle per particle over the cells comprised within the smoothing-kernel length. However, for $\mathcal{R} \leq 1$ the number of cells comprised within the smoothed-kernel length tend to be around the unity, and hence the additional contribution of such cycles on the computational expense tend to become negligible. A possible untackled optimization problem in the code is probably responsible for the increased computational expense on the local and first-neighbour models.

Fig. 5 reports the results of the grid independence test under diffusive scaling for the pilot-scale runs, comprising all the La-grangian models. The x coordinate of the vortex, which is evident in all the plots of Fig. 4, was plotted against N. All the models displayed a similar behaviour, and convergence is evidently achieved for $N \geq 80$. This was expectable, as $\mathcal{R} < 1$ for all the runs. Remarkably, no data points are reported from the smoothed-kernel runs for $N < 49$: this happened because, for $d < \delta x$, configurations can occur where no grid points are comprised between one smoothed-kernel length from a bubble centre, thus raising an ir-resolvable divide-by-zero error. Variations in the vortex's x coordinate between contiguous values of N resembled oscillations, and this is especially true for the first-neighbour and the smoothed-kernel model. It was not possible to ascertain what caused them.

5.2. Laboratory-scale runs: comparison of different Lagrangian models

In Fig. 6, example results of the laboratory-scale runs are reported for $N = 120$. All the three Lagrangian models are tested. On the left, scaled flow patterns following the "0" method are reported, on the right the unscaled. Experimental flow patterns from [10] are reported for comparison. The flow patterns reproduced the experimental data well, apart from a mild overall underestimation of the intensity, which is more pronounced in the scaled runs. A slight inwards horizontal displacement of the vortex position for

(a)

(b)

(c)
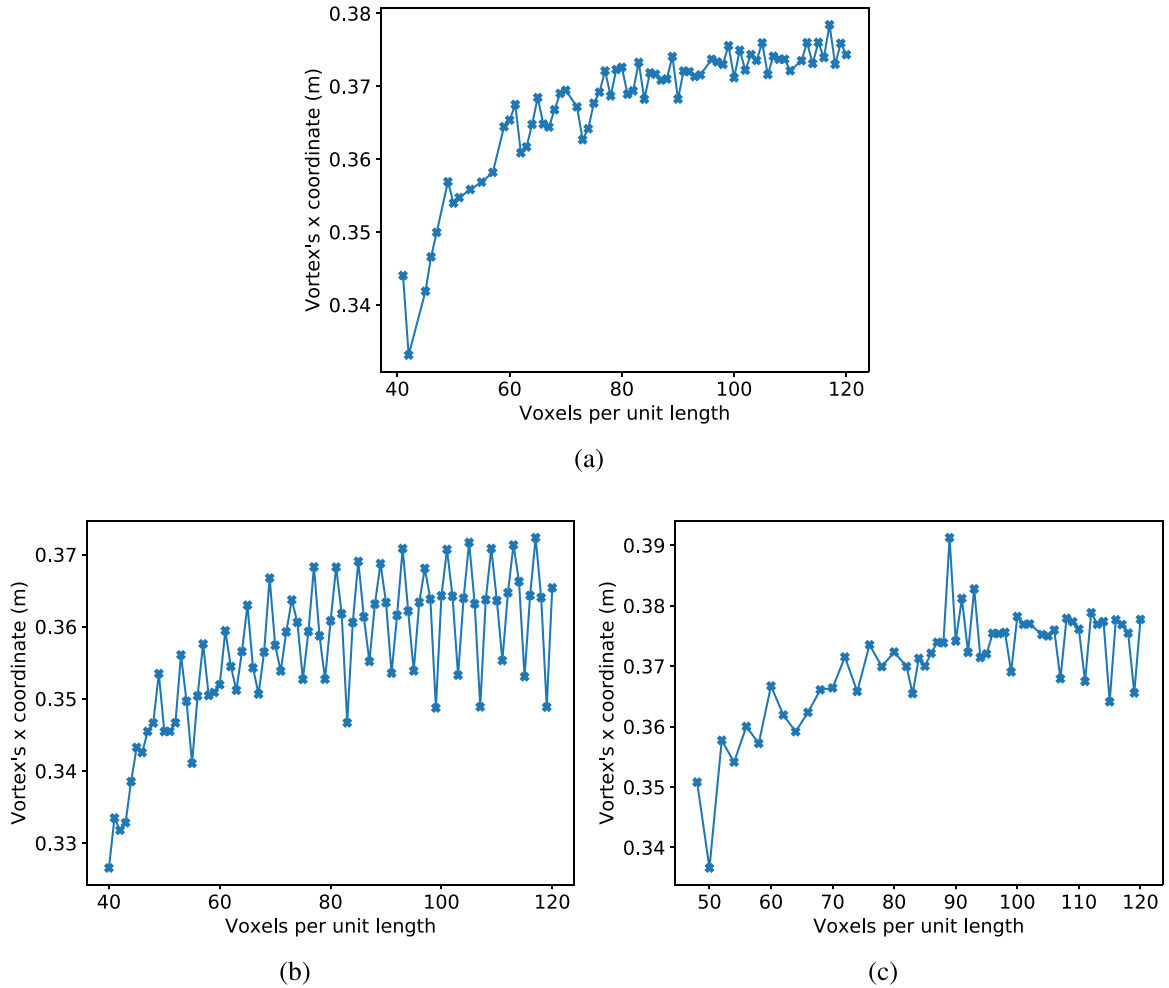
Fig. 5. Mesh independence, diffusive scaling, pilot-scale. (a) Local model. (b) First-neighbour model. (c) Smoothed-kernel model.

the unscaled runs, and outwards for the scaled, was also present. The computational expense was comprised between 850 and 8500 CPUs, with the smoothing-kernel model being remarkably more expensive (up to ten times) than the other models. This was due to the fact that the number of cells comprised within the smoothing-kernel length grows as N^3 for $\mathcal{R} > 1$, and the same holds for the computational expense related to the smoothing-kernel internal cycles.

Fig. 7 reports the results of the grid independence test under diffusive scaling for the laboratory-scale runs, comprising all the Lagrangian models. On the right, scaled flow patterns following the "0" method are reported, on the left the unscaled. The local unscaled model clearly does not converge, while the judgement is more difficult for the other unscaled models. This was expectable, as we had $\mathcal{R} > 1$. On the contrary, convergence was evident in all the scaled runs, and grid independence could be considered to be achieved for $N = 120$ − at the cost, however, of a more pronounced discrepancy from experimental data (12-26% against 8-12%). Oscillations are clearly visible.

5.3. Laboratory-scale runs: comparison of different types of scaling

In Fig. 8, example results of the laboratory-scale runs for all the scaling types reported in Table 2 are provided for $N = 120$. Only the local Lagrangian was used. Experimental flow patterns from [10] are reported for comparison. As expectable, not all the scaling types resulted to produce flow patterns compatible with the ex-

perimental data. In particular, they did so only the "0", the "2PNu" and the "12PNu-Q".

Fig. 9 reports the results of the grid independence test under diffusive scaling for the laboratory-scale runs, comprising all the scaling types reported in Table 2, and the local Lagrangian model. Convergence was found only under the scaling types "0"and "12PNu-Q'. This despite the presence of some off-trend values in "12PNu-Q" following a quasi-periodical recurrence. This was probably due to an unidentified error in a floating-point-to-integer conversion. Performing a large number of runs helped identifying the entries suffering of this problem, and excluding them. A comparison with Fig. 8 shows that these scaling types produce grid-independent results which are compatible with experimental data, with the "12PNu-Q", with the last one being the best choice.

6. Discussion

6.1. Convergence and scaling

A correlation between value and variation of $\mathcal{R}$, and convergent (or non-convergent) behaviour, is evident. In the laboratory-scale simulations (Section 5.1), we had $0.3176 \leq \mathcal{R} \leq 0.9528 < 1$ and convergent behaviour; in pilot-scale, unscaled simulations (Section 5.2), we had $1 < 1.588 \leq \mathcal{R} \leq 7.94$ and non-convergent behaviour; finally, in laboratory-scale, scaled simulations (Sections 5.2 and 5.3), we had $1 < 1.588 \equiv \mathcal{R}$ and convergent

Fig. 6. Flow patterns, lab-scale, PIV plane. Numerical data are displayed at their last timestep, and computational expense is reported. (a) PIV experimental data, averaged. (b) Unscaled, smoothed-kernel model. 8526 CPUs. (c) "0", local model. 847 CPUs. (d) Unscaled, local model. 1516 CPUs. (e) "0", first-neighbour model. 863 CPUs. (f) Unscaled, first-neighbour model. 1329 CPUs.

behaviour (depending on the choice of scaling type). This allowed us to conclude that:

1. For $\mathcal{R} < 1$, convergence occurs and is unaffected by the value of $\mathcal{R}$;
2. For $\mathcal{R} < 1$, the value of $\mathcal{R}$ does affect convergence, and in particular:

(a) for variable $\mathcal{R}$, non-convergent behaviour occurs;
(b) for constant $\mathcal{R}$, it is possible to retrieve convergent behaviour.

In synthesis, convergence can occur if $\mathcal{R} < 1$ or $\mathcal{R} = $ const.

The introduction of a scaling mechanism has the effect of constraining $\mathcal{R}$ to a constant value, hence enabling convergence de-
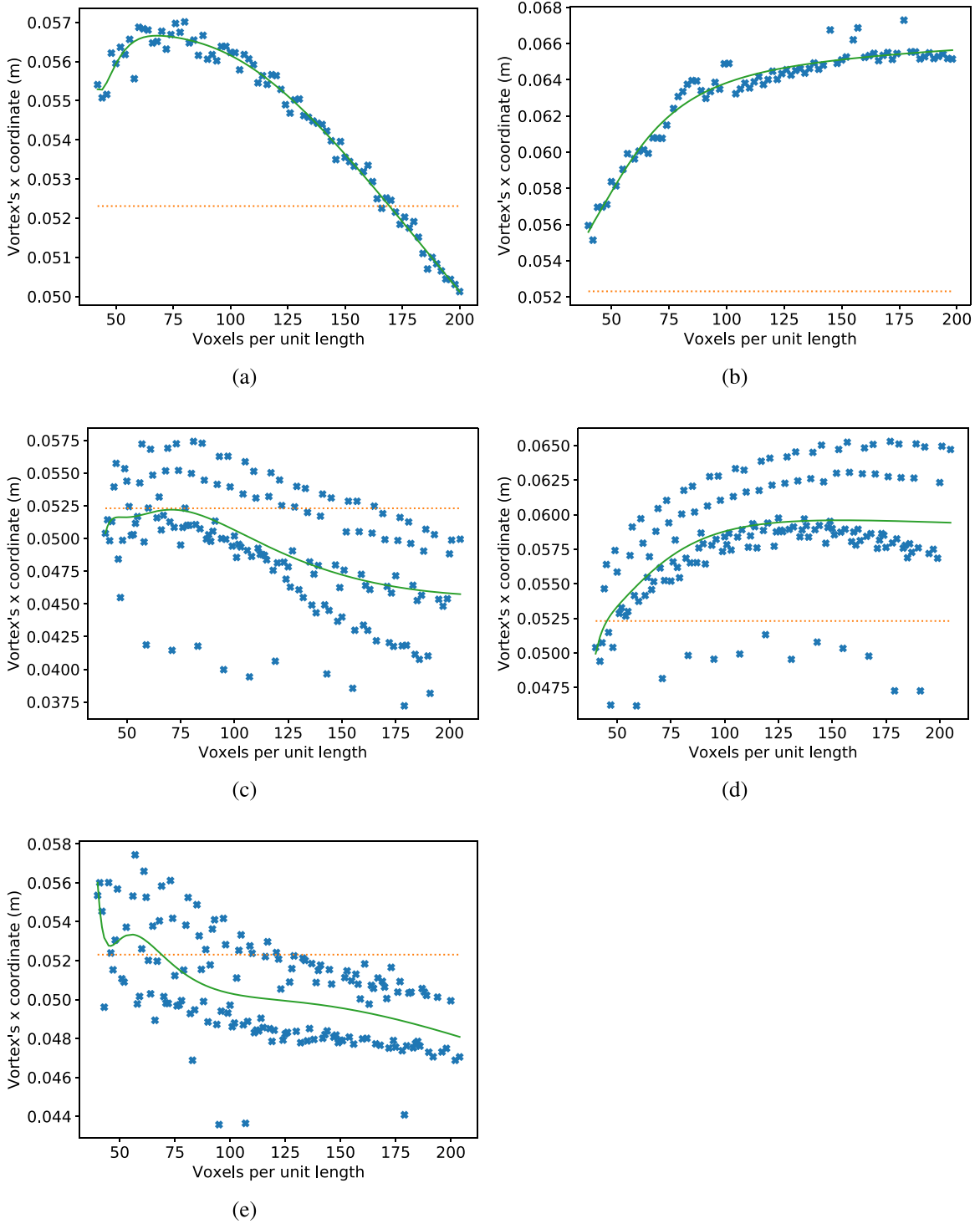
Fig. 7. Mesh independence, diffusive scaling, lab-scale. "X" marks: numerical data Continuous line: best fit over 5th order polynomial. Dotted line: experimental data. (a) Local model, unscaled. (b) Local model, "0". (c) First-neighbour model, unscaled. (d) First-neighbour model, "0". (e) Smoothed-kernel model, unscaled.

pending on the choice of the physically-meaningful dimensionless parameter(s). On the other side, however, introduces a systematic error, which may affect the model's predictive power.

In both cases, the model was verified (see Fig. 10) to be second-order accurate, as the Lattice-Boltzmann general framework under diffusive scaling, and as the HLBM [21]. The error was defined as err$(N) := |\Theta_x(N) - \Theta_x(N^\star)|$, where Θ_x is the vortex's x coordinate. $N^\star$ is the superior cutoff value for N, and it was chosen as 120 for the pilot-scale (because it was the highest number for N tested

in the pilot-scale), and 128 for the laboratory-scale (because close to the value of $N = 120$, for which convergence was found to be achieved in Section 5.3).

6.2. Choosing a simulation strategy

The considerations above allow us to trace a two-step strategy for Lagrangian modelling in the $\mathcal{R} > 1$ case.
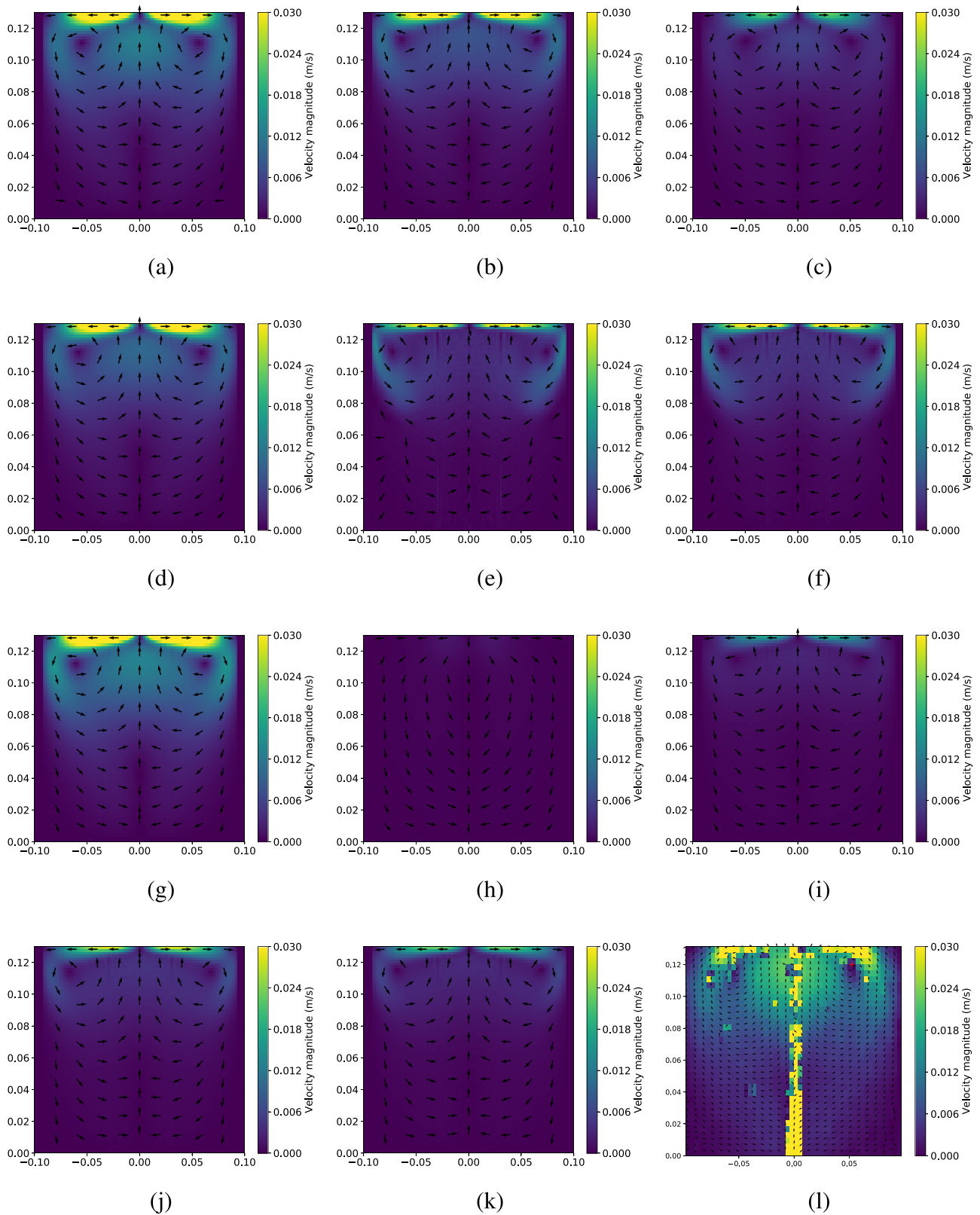
Fig. 8. Flow patterns, lab-scale, PIV plane. Numerical data are produced through the local model, and are displayed at their last timestep. Computational expense is reported. (a) Unscaled, 1516 CPUs. (b) "0", 847 CPUs. (c) "1PNu", 749 CPUs. (d) "2PNu", 705 CPUs. (e) "2PU", 1468 CPUs. (f) "2Re", 1006 CPUs. (g) "12PNu-Q", 879 CPUs. (h) "12PNu-Re", 936 CPUs. (i) "12PNuU-Re", 1311 CPUs. (j) "12PU-Q", 903 CPUs. (k) "12Q-Re", 1161 CPUs. (l) PIV experimental data, averaged.
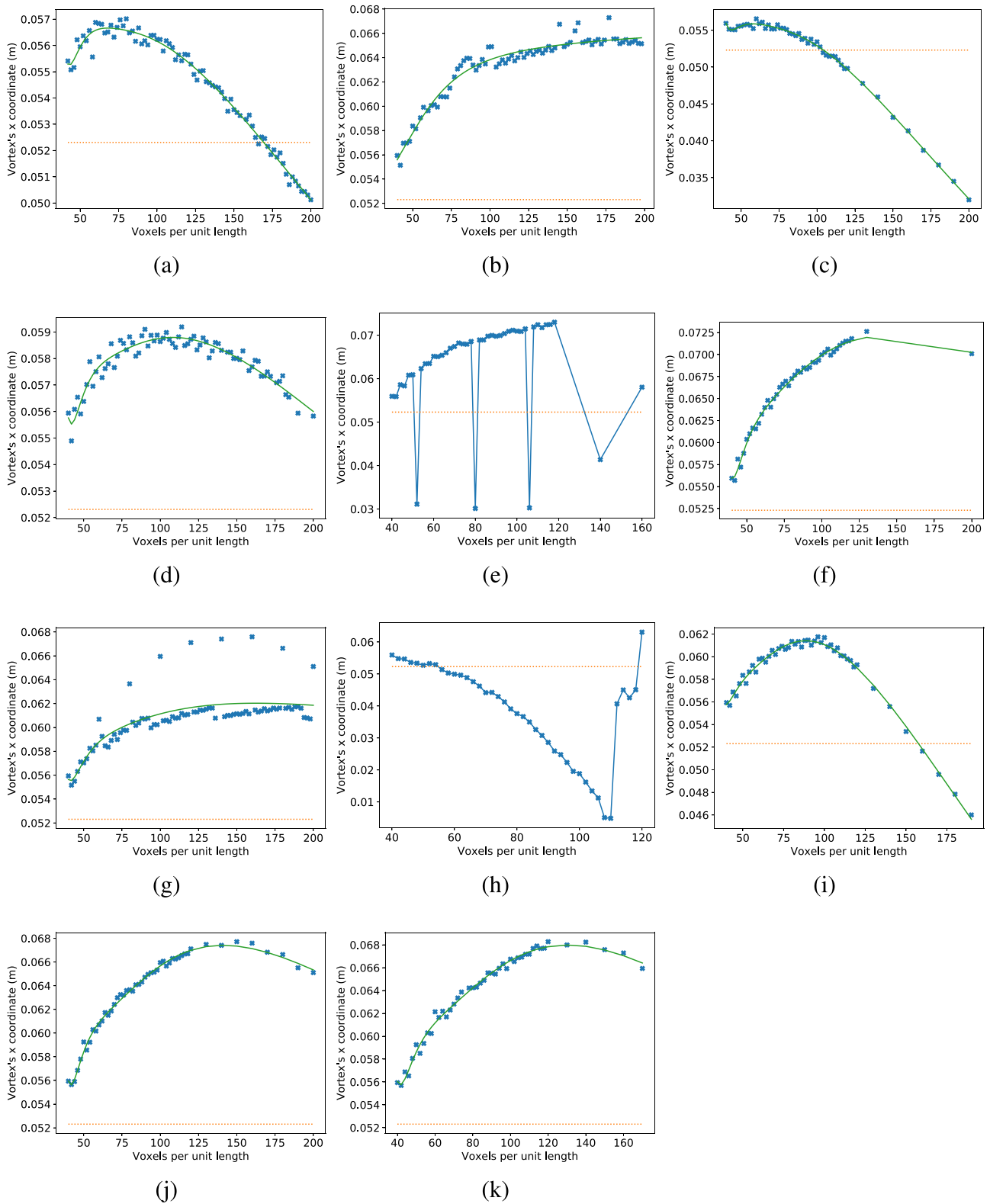
Fig. 9. Mesh independence, diffusive scaling, lab-scale. Numerical data are produced through the local model. "X" marks: numerical data Continuous line: best fit over 5th order polynomial. Dotted line: experimental data. (a) Unscaled. (b) "0". (c) "1PNu". (d) "2PNu". (e) "2PU". (f) "2Re". (g) "12PNu-Q". (h) "12PNu-Re". (i) "12PNuU-Re". (j) "12PU-Q". (k) "12Q-Re".
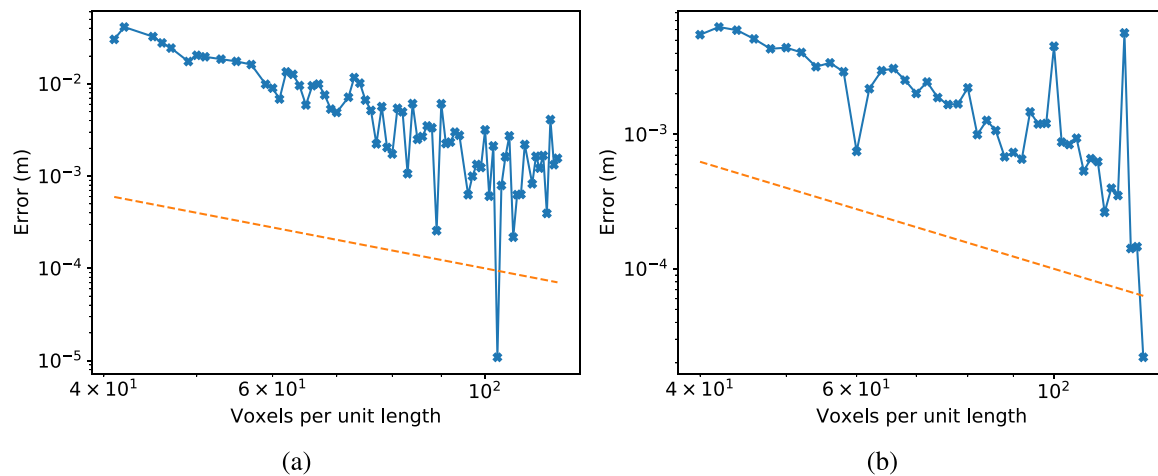
Fig. 10. Order of convergence. Solid dashed lines: numerical result. Dashed lines: Second-order example. (a) Pilot scale, local model. (b) Laboratory scale, "12PNu-Q".

1. Grid independence should be evaluated through scaling, and the best compromise between grid independence and computational expense should be determined. Different scaling types should be tested in order to find the most appropriate physically-meaningful dimensionless parameter(s). For the specific example of the laboratory-scale computational work reported within this article (Sections 5.2 and 5.3), grid-independent behaviour was shown, and the best compromise between grid independence and computational expense was evaluated to be $N = 120$.

2. The best grid determined in the point above should be used to run unscaled simulations: as they are free from the systematic error affecting the scaled runs (see Section 6.1), better physical results will be obtained.

6.3. Comparison with HLBM [13]

All the Lagrangian model apart from the smoothed-kernel display strong similarities as regards computational expense. The latter is generally several times more expensive because of the cycles over the cells comprised within the smoothing-kernel length, as explained in Section 5.2. It was also shown that this additional amount of computational expense vanishes for $\mathcal{R} \sim 1$.

Compared to the HLBM [13], the advantage of the Lagrangian methods is evident. Both the applications reported here and in [13] presented the same structure and computational expense resulted to depend on the number of cells. The HLBM constraint, reported in Section 1, of having 10-20 cells per particle diameter meant that HLBM laboratory-scale simulations were found to converge only for $N \geq 400$: as the Lagrangian laboratory-scale simulations were found to converge for $N \geq 120$ in Section 5.2, that means that the Lagrangian model requires around 37 times less resources than the HLBM to produce the laboratory-scale problem described in Section 4.2. The same constraint means that hypothetical pilot-scale HLBM simulations of the pilot-scale problem (Section 4.3) would be expected to converge for $N \geq 2000$ (in Section 5.1 it was found that pilot-scale Lagrangian simulations converge for $N \geq 100$), which would mean around 8000 times the resources needed for Lagrangian runs.

Qualitatively, both the Lagrangian and the HLBM models reproduced the experimental flow patterns, with the first slightly underpredicting them, and the latter equally overpredicting. Quantitatively, the Lagrangian model overpredicted the vortex's x position by around 19%. This is relevant larger than HLBM's error on the same quantity, which fell under 1%. However, the HLBM's good performance was obtained thanks to the expedient of introducing partial-slip boundary conditions for the top liquid surface. Such expedient was not adopted in the Lagrangian model presented within this paper. HLBM's error in the case of free-slip boundary conditions peaked to around 30%.

7. Conclusions

Euler-Lagrange Lattice-Boltzmann models were applied for the first time in gas-mixed anaerobic digestion. Laboratory-scale validation was performed, and pilot-scale scenarios were studied. The advantage of these models against previously-adopted solutions (viz., the HLBM) was evaluated as being between 37 and 8000 times less resources being necessary.

The convergence behaviour of the Euler-Lagrange models was found to be dependent on the ratio $\mathcal{R}$ between particle diameter and lattice size. In particular, convergence did not naturally occur for $\mathcal{R} > 1$, corresponding to the violation of the Euler-Lagrange's fundamental assumption of negligible particle size.

A smoothed-kernel Euler-Lagrange model was introduced in order to retrieve convergence for $\mathcal{R} > 1$, but a definitive answer could not be found. As an alternative, a novel scaling system, based on the Buckingham-pi theorem, was introduced. The scaling system was proved to be effective. Limitations of the scaling approach and best ways to employ it were discussed.

The work described within this article will have applications in the modelling of complex flow patterns in laboratory, pilot and full scale geometries; in particular, it represents a significant step forward in advancing the topic of mixing optimization of gas-mixed anaerobic digesters.

This work is part of an ambitious project (see "Acknowledgements" below), which aims to fully couple biokinetics and hydrodynamics in full-scale anaerobic digesters. Next steps will include a separated validation of a simplified biokinetic model, the validation of a coupled model against laboratory experimental data, and the application to industrial digesters.

Declaration of Competing Interest

No interests to declare.

CRediT authorship contribution statement

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.compfluid.2020. 104632.

References

[1] Andersson B, Andersson R, Håkansson L, Mortensen M, Defence N, Sudiyo R, et al. Computational fluid dynamics for engineers. Cambridge, UK: Cambridge University Press; 2012.

[2] Benzi R, Biferale L, Sbragaglia M, Succi S, Toschi F. Mesoscopic modelling of heterogeneous boundary conditions for microchannel flows. J Fluid Mech 2006;548:257–80. arXiv:0502060. doi:10.1017/S0022112005007512.

[3] Bhatnagar PL, Gross EP, Krook M. A model for collision processes in gases. I. small amplitude processes in charged and neutral one-component systems. Phys Rev 1954;94(3):511–25. doi:10.1103/PhysRev.94.511.

[4] Bouzidi M, Firdaouss M, Lallemand P. Momentum transfer of a Boltzmann-lattice fluid with boundaries. Phys Fluids 2001;13:3452–9.

[5] Boyd J, Buick J, Green S. A second-order accurate Lattice Boltzmann non-Newtonian flow model. J Phys A Math Gen 2006;39(46):14241–7. doi:10.1088/0305-4470/39/46/001.

[6] Bridgeman J. Computational fluid dynamics modelling of sewage sludge mixing in an anaerobic digester. Adv Eng Software 2012;44(1):54–62. doi:10.1016/j.advengsoft.2011.05.037.

[7] Buckingham E. Model experiments and the forms of empirical equations. Trans Am Soc MechEng 1915;37:263–96.

[8] Council of the European Union, 1991. Council Directive of 21 May 1991 concerning Urban Waste Water Treatment (91/271/EEC).

[9] Craig KJ, Nieuwoudt MN, Niemand LJ. CFD simulation of anaerobic digester with variable sewage sludge rheology. Water Res 2013;47(13):4485–97. doi:10.1016/j.watres.2013.05.011.

[10] Dapelo D, Alberini F, Bridgeman J. Euler-Lagrange CFD modelling of unconfined gas mixing in anaerobic digestion. Water Res 2015;85:497–511. doi:10.1016/j.watres.2015.08.042.

[11] Dapelo D, Bridgeman J. Assessment of mixing quality in full-scale, biogas-mixed anaerobic digestion using CFD. Bioresour Technol 2018;265(April):480–9. doi:10.1016/j.biortech.2018.06.036.

[12] Dapelo D, Bridgeman J. Euler-Lagrange computational fluid dynamics simulation of a full-scale unconfined anaerobic digester for wastewater sludge treatment. Adv Eng Software 2018;117:153–69. doi:10.1016/j.advengsoft.2017.08.009.

[13] Dapelo D, Trunk R, Krause MJ, Bridgeman J. Towards Lattice-Boltzmann modelling of unconfined gas mixing in anaerobic digestion. Comput Fluids 2019;180:11–21. doi:10.1016/j.compfluid.2018.12.008.

[14] European Environment Agency. Waterbase - UWWTD: Urban Waste Water Treatment Directive reported data. Technical Report; 2015.

[15] Evrard F, Denner F, van Wachem B. A multi-scale approach to simulate atomisation processes. Int J Multiphase Flow 2019;119:194–216. doi:10.1016/j.ijmultiphaseflow.2019.07.005.

[16] Guo Z, Shu C. Lattice Boltzmann method and its applications in engineering, 3. Singapore: World Scientific Publishing Co. Pte. Ltd.; 2013.

[17] Guo Z, Zheng C, Shi B. Discrete lattice effects on the forcing term in the lattice Boltzmann method. Phys Rev E 2002. doi:10.1103/PhysRevE.65.046308.

[18] Hou S, Sterling J, Chen S, Doolen GD. A lattice boltzmann subgrid model for high reynolds number flows. In: Lawniczak A, Kapral R, editors. Pattern formation and lattice gas automata, 6. AMS - Fields Institute Communications; 1996. arXiv:9401004.

[19] Hurtado F, Kaiser A, Zamora B. Fluid dynamic analysis of a continuous stirred tank reactor for technical optimization of wastewater digestion. Water Res 2015;71:282–93. doi:10.1016/j.watres.2014.11.053.

[20] Karim K, Thoma GJ, Al-Dahhan MH. Gas-lift digester configuration effects on mixing effectiveness. Water Res 2007;41(14):3051–60. doi:10.1016/j.watres.2007.03.042.

[21] Krause MJ, Klemens F, Henn T, Trunk R, Nirschl H. Particle flow simulations with homogenised Lattice Boltzmann methods. Particuology 2017;34:1–13. doi:10.1016/j.partic.2016.11.001.

[22] Krause MJ, Kummerländer A, Avis SJ, Kusumaatmaja H, Dapelo D, Klemens F, et al. OpenLBOpen source Lattice Boltzmann code. Comput Math Appl 2020. doi:10.1016/j.camwa.2020.04.033.

[23] Kress P, Nägele HJ, Oechsner H, Ruile S. Effect of agitation time on nutrient distribution in full-scale CSTR biogas digesters. Bioresour Technol 2018;247(September 2017):1–6. doi:10.1016/j.biortech.2017.09.054.

[24] Krüger T, Kasumaatmaja H, Kuzmim A, Shardt O, Silva G, Viggen EM. The Lattice Boltzmann method. Springer; 2017. doi:10.1007/978-3-319-44649-3.

[25] Landry H, Laguë C, Roberge M. Physical and rheological properties of manure products. Appl Eng Agric 2004;20(3):277–88. doi:10.13031/2013.16061.

[26] Lebranchu A, Delaunay S, Marchal P, Blanchard F, Pacaud S, Fick M, et al. Impact of shear stress and impeller design on the production of biogas in anaerobic digesters. Bioresour Technol 2017;245(June):1139–47. doi:10.1016/j.biortech.2017.07.113.

[27] Maier ML, Henn T, Thaeter G, Nirschl H, Krause MJ. Multiscale simulation with a two-way coupled Lattice Boltzmann method and discrete element method. Chem Eng Technol 2017;40(9):1591–8. doi:10.1002/ceat.201600547.

[28] Meister M, Rezavand M, Ebner C, Pümpel T, Rauch W. Mixing non-Newtonian flows in anaerobic digesters by impellers and pumped recirculation. Adv Eng Software 2018;115(February 2017):194–203. doi:10.1016/j.advengsoft.2017.09.015.

[29] Meroney RN, Colorado PE. CFD simulation of mechanical draft tube mixing in anaerobic digester tanks. Water Res 2009;43(4):1040–50. doi:10.1016/j.watres.2008.11.035.

[30] Morsi SA, Alexander AJ. An investigation of particle trajectories in two-phase flow systems. J Fluid Mech 1972;55(2):193–208. doi:10.1017/S0022112072001806.

[31] Nathen P, Gaudlitz D, Krause MJ, Adams NA. On the stability and accuracy of the BGK, MRT and RLB Boltzmann schemes for the simulation of turbulent flows. Commun Comput Phys 2018;23(3):846–76. doi:10.4208/cicp.oa-2016-0229.

[32] Owen WF. Anaerobic treatment processes. Energy in wastewater treatment. Englewood Cliffs, NJ: Prentice-Hall, Inc.; 1982.

[33] Peskin CS. The immersed boundary method. Acta Numerica 2002;11:479–517. doi:10.1017/S0962492902000077.

[34] Rettinger C, Rüde U. A comparative study of fluid-particle coupling methods for fully resolved lattice boltzmann simulations. Comput Fluids 2017;154:74–89. arXiv:1702.04910. doi:10.1016/j.compfluid.2017.05.033.

[35] Sbragaglia, M., Succi, S., 2004. Analytical calculation of slip flow in Lattice Boltzmann models with kinetic boundary conditions, 1–27arXiv:041003910.1063/1.2044829

[36] Sindall RC, Bridgeman J, Carliell-Marquet C. Velocity gradient as a tool to characterise the link between mixing and biogas production in anaerobic waste digesters. Water Sci Technol 2013;67(12):2800–6. doi:10.2166/wst.2013.206.

[37] Succi S. Lattice Boltzmann 2038. Europhys Lett 2015;109(5):50001. doi:10.1209/0295-5075/109/50001.

[38] Terashima M, Goel R, Komatsu K, Yasui H, Takahashi H, Li YY, et al. CFD simulation of mixing in anaerobic digesters. Bioresour Technol 2009;100(7):2228–33. doi:10.1016/j.biortech.2008.07.069.

[39] Trunk R, Marquardt J, Thäter G, Nirschl H, Krause MJ. Towards the simulation of arbitrarily shaped 3D particles using a homogenised lattice Boltzmann method. Comput Fluids 2018;172:621–31. doi:10.1016/j.compfluid.2018.02.027.

[40] Vaschy A. (Title unknown). Les Annales Telegraphiques 1892;19:25–8.

[41] Vesvikar MS, Al-Dahhan MH. Flow pattern visualization in a mimic anaerobic digester using CFD. Biotechnol Bioeng 2005;89(6):719–32. doi:10.1002/bit.20388.

[42] van Wachem BGM, Almstedt AE. Methods for multiphase computational fluid dynamics. Chem Eng J 2003;96(1–3):81–98. doi:10.1016/j.cej.2003.08.025.

[43] WaterUK. Sustainability Indicators 2010–2011. Technical Report. London: Water UK; 2012.

[44] Wu B. CFD simulation of gas and non-Newtonian fluid two-phase flow in anaerobic digesters. Water Res 2010;44(13):3861–74. doi:10.1016/j.watres.2010.04.043.

[45] Wu B, Chen S. CFD simulation of non-Newtonian fluid flow in anaerobic digesters. Biotechnol Bioeng 2008;99(3):700–11. doi:10.1002/bit.21613.

[46] WWAP (World Water Assessment Programme). The United Nations World Water Development Report 4: Managing Water under Uncertainty and Risk. Technical Report. Paris: UNESCO; 2012.

[47] Zhang Y, Yu G, Yu L, Siddhu MAH, Gao M, Abdeltawab AA, et al. Computational fluid dynamics study on mixing mode and power consumption in anaerobic mono- and co-digestion. Bioresour Technol 2016;203:166–72. doi:10.1016/j.biortech.2015.12.023.