2. Materials and methods

This section describes the implementation of the model, including its mathematical formulation, numerical approach, and computation setup. The model is based on the extended photo-anaerobic (ePAnM) and employs a two-scalar model method with a diffuse interface to accurately represent mass transfer between liquid and gas phases. Additionally, all materials used in this work, as well as the mesh generation process and numerical leakage analysis, are detailed.

2.1 Model description: two scalar method and diffuse interface

The development of the model was carried out using the extended photo-anaerobic model (ePAnM) equations as a baseline. Considering the two-phase flow between water and gas in the reactor, the volume of fluid model uses the volume fraction of water, represented by the phase-field variable α, to create continuous fields that gradually transition between phases (Qin and Bhadeshia, 2010). In this way, the values $\alpha = 1$, $\alpha = 0$ represent cells entirely occupied by water and gas respectively, while $0 < \alpha < 1$ corresponds to the interface between these phases. Fig X illustrates a diffuse interface applying the method described above, where the gas phase is represented by the symbol $\psi = 1 - \alpha$.

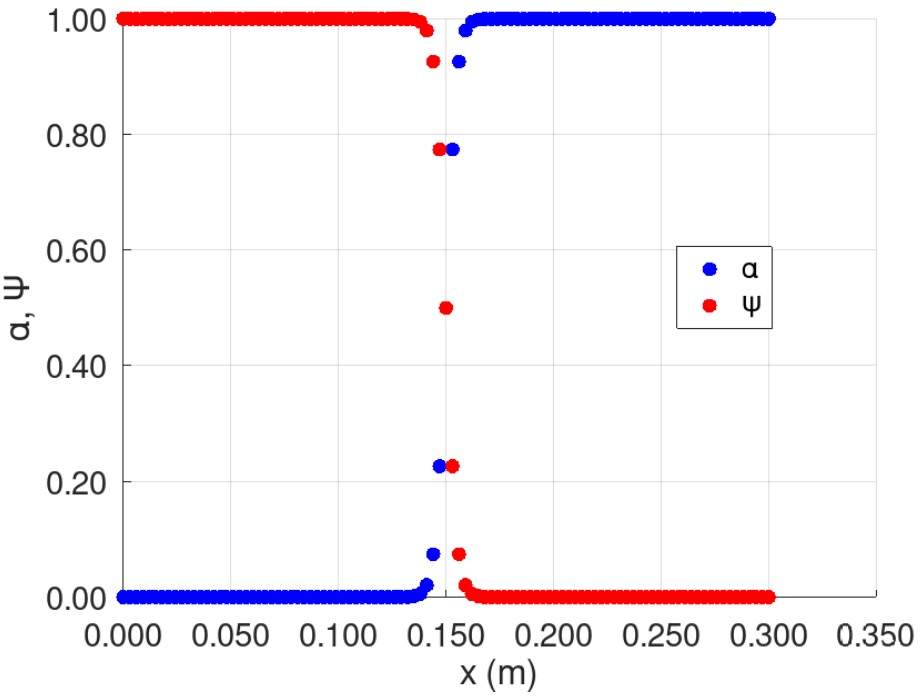

Fig X: Water and gas phases, having a diffuse interface.

The formulation used above, confines state variables and reaction source terms strictly to their respective phases. To address species that can exist in both phases, the two scalar model is employed. This model provides a numerical strategy for treating scalars by defining two separates scalar variables: one for the species in the water phase and another for the species in the gas phase. For example, hydrogen can be present in the liquid phase as dissolved hydrogen, and in the gas phase as H_2. This will be named, respectively S_{H2}, which exists only in the regions where $\alpha > 0$, and $[\mathrm{H_2}]_\mathrm{g}$, which exists only in the regions where $\psi > 0$. This approach ensures that each scalar is accurately associated with its corresponding phase, as illustrated in Figure Y.

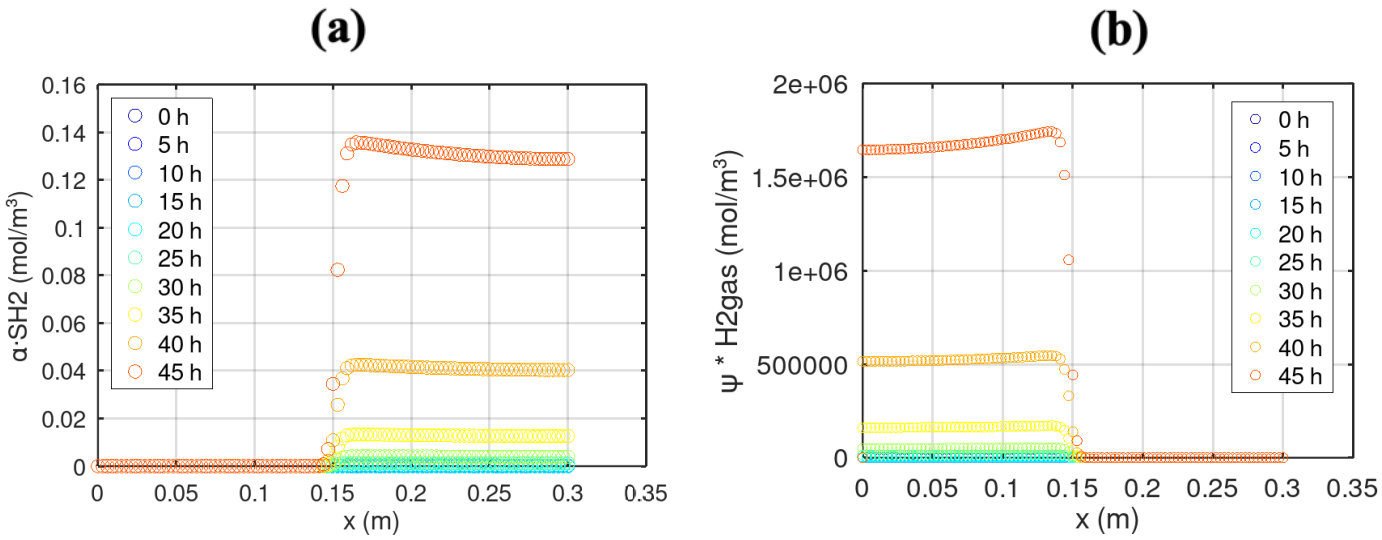

Fig Y: Confinement of the scalars in their corresponding phase using the two scalar model. (a) Hydrogen in liquid phase. (b) Hydrogen in gas phase.

Therefore, the proposed model in ePAnM, based on the two scalar model and diffusive interface, enables accurate confinement of state variables, efficient handling of species in multiple phases (e.g., hydrogen), and reduces numerical errors and enhancing computational stability by eliminating the need for explicit interface tracking.

In the context of gas transfer through bubble interfaces, the ePAnM model addresses the mass transfer of hydrogen between bubbles (gas phase) and surrounding liquid (Fig Z). Although high partial pressure in the gas phase would normally promote the solubilization of hydrogen into the liquid, the effective transfer is governed by the available gas-liquid interface area. This can be expressed by the following equation, regarding the hydrogen:

$$r_{H2} = k_{La,H2} \cdot A_{int} \cdot (SH_2^{sat} - \alpha \widehat{SH_2}) \quad (1)$$

Where Aint is the gas-liquid interfacial area, $k_{La,H2}$ is the mass transfer coefficient and SH_2^{sat} is concentration of soluble hydrogen in the liquid phase if the latter was saturated.

Thus, this equation indicates the overall mass transfer rate of hydrogen depends on both the available interfacial area and the driving force between the saturation concentration and the current concentration. In practice, this means that even if the gas phase exhibits a high hydrogen partial pressure (which would tend to increase SH_2^{sat}), significant mass transfer will not occur unless a sufficient interfacial area is present.

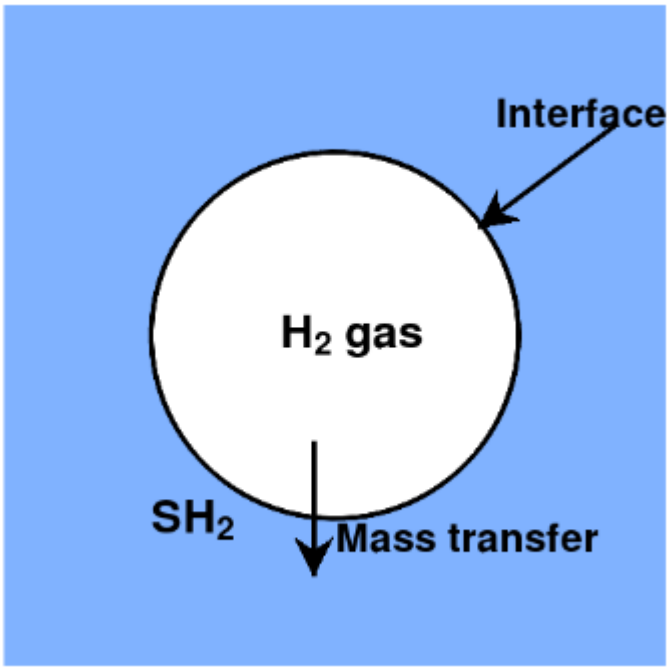

Fig Z: Gas transfer through bubble interfaces.

The ePAnM framework is built upon a set of transport equations, where the phase field variable is added to confine the state variables and source terms (e.g., reactions) to water or gas phases. Two separate transport equation are defined: one for variables restricted to the water phase (A) and another for variables limited to the gas phase (B). In this manner, an A equation is solved for the variables S_{IC}, S_{H2}, S_{IN}, X_{PB}, X_C, S_{HCO3}, S_{NH3} and S_{H2PO4}, whereas a B equation applies to the $[H_2]_g$ and the $[CO_2]_g$ equations.

$$\frac{\partial \alpha A}{\partial t} + \vec{\nabla} \cdot (\alpha \vec{u} A) - \vec{\nabla} \cdot \left(\alpha D \vec{\nabla} A\right) = St \tag{2}$$

$$\frac{\partial \psi B}{\partial t} + \vec{\nabla} \cdot (\psi \vec{u} B) - \vec{\nabla} \cdot \left(\psi D \vec{\nabla} B\right) = St \tag{3}$$

The ePAnM model assumes a steady state bubble and velocity distribution, meaning that the time required for bubbles to rise from the inlet to the headspace is negligible compared to the time needed for a significant change in gas concentration inside the bubbles. Consequently, the B equations are, in principle, not necessary.

Each transport equation incorporates an accumulation, advection, diffusion, and source term. In the first three terms, the multiplication by the phase-field variable (α or ψ) ensures that these processes are confined to the appropriate phase. Moreover, two scalar model is used for species that exists in both phases, to maintain distinct representations for the liquid and gaseous forms, which is essential for accurately modeling the mass transfer across the interface. Having hydrogen as an example, we obtain these equations:

$$\frac{\partial \alpha S_{H2}}{\partial t} + \vec{\nabla} \cdot (\alpha \vec{u} S_{H2}) - \vec{\nabla} \cdot \left(\alpha D \vec{\nabla} S_{H2}\right) = \alpha r_{H2} - \alpha f_{H2,au} r_1 \tag{4}$$

$$\frac{\partial \psi [H_2]_g}{\partial t} + \vec{\nabla} \cdot \left(\psi \vec{u} [H_2]_g\right) - \vec{\nabla} \cdot \left(\psi D \vec{\nabla} [H_2]_g\right) = -\psi r_{H2} \tag{5}$$

Where eqs 4, 5 are the transport equation for dissolved hydrogen and hydrogen in gas phase respectively.

2.2 Model Implementation

The transport equations eqs 2 and 3, applied to all the variables used in the biological model based on the two scalar model and diffuse interface approach (see Appendix), were implemented in OpenFOAM (version 8) using the finite volume method.

All computations and processes described in this work were performed on a DELL Inc. computer equipped with an Intel® Core (TM) i5-8250U CPU @ 1.60GHz, running Linux as the operating system. The system had 8 GiB of RAM and 468.8 GiB of available disk storage.

2.3 Meshes

2.3.1. One-dimensional mesh

Before simulating the model implemented in the axisymmetric mesh (compartment model) of the actual photobioreactor, a one-dimensional mesh (along the x axis) was created to verify that there are no errors in the calculations during the simulation, to check if the results are physically correct, and to detect any numerical leakage.

The mesh was generated using the *blockMesh* utility in OpenFOAM. This pre-processing tool allows the creation of parametric meshes with grading and curved edges (Greenshields, 2020). The obtained mesh has a hexahedral geometry (Fig A), and the number of cells in each direction are 100 x 1 x 1. To view the mesh, we use ParaView (version 5.11.2), an open-source application for analyzing and visualizing data sets of varying sizes.

Once the mesh was generated, the next step is to introduce values of the phase-field variable (α) and the velocity as inputs for the transport equations, to ensure the confinement of the variables in their respective phases and to activate the advection term. The values of α were obtained through a GNU Octave script (see Appendix), and we used a constant velocity value of 1e-5 m/s. Once these values were obtained, we introduced them into the alpha and U dictionaries present in the 0 folder in OpenFOAM.

After these values were introduced, the simulation was started to verify the assumptions made above. (See section 3 for results).

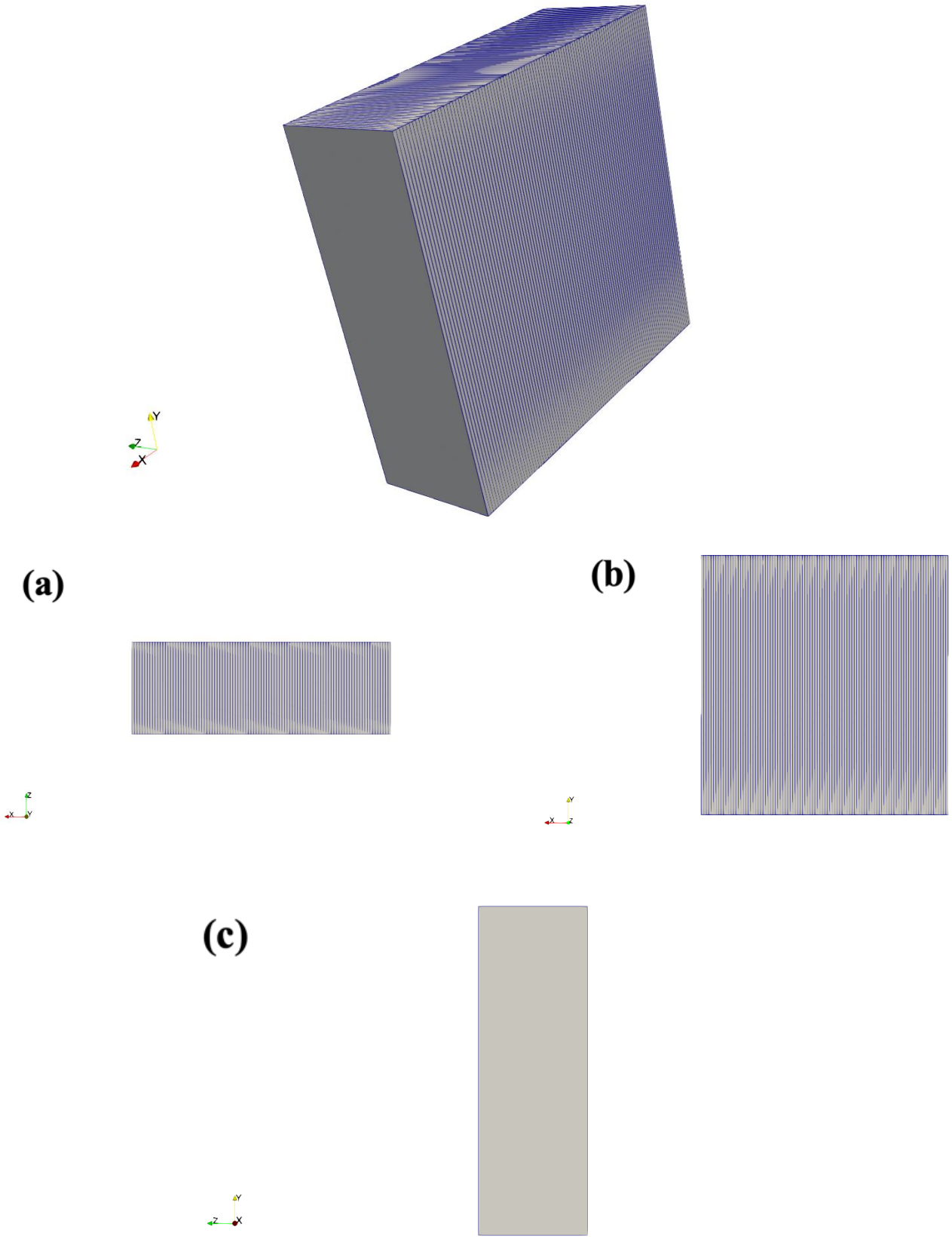

Fig A: One-dimensional mesh (along the x axis) generated using *blockMesh* and visualized in ParaView. (a) XZ-plane. (b) XY-plane. (c) YZ-plane.

2.3.2. Compartment model

As discussed in the first section of this work, the compartment model is a mesh generated from the quarter-cylinder representation of the photobioreactor (full geometry). This approach reduces the number of calculations and shortness the simulation time. Unlike the previous section, where pre-processing utilities were used, the compartment model is obtained solely through post-processing utilities.

Initially, the compartment model was created using the *TopoSet* and *splitMeshRegions* utilities in OpanFOAM (Fig C). *Toposet* is a post-processing utility used to manipulate cell, face, and point sets and zones to define regions (Greenshields, 2020), while *splitMeshRegions* divides the mesh into multiple independent regions, allowing each region to solve different equations and interact with others.

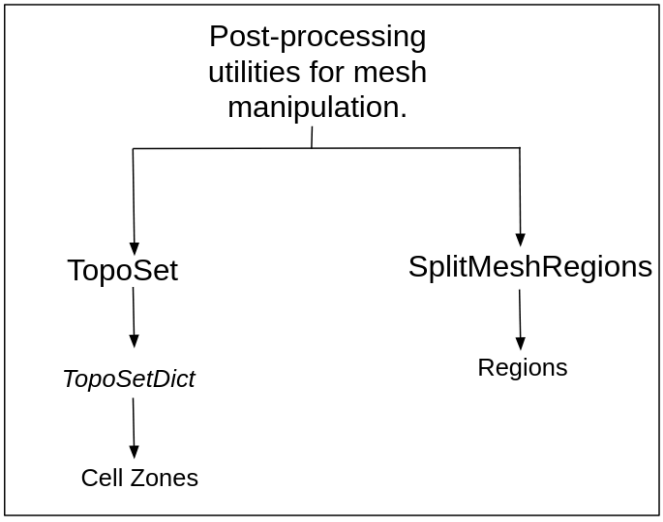

Fig C: Functionality of the utilities for mesh manipulation.

To use *Toposet*, a *TopoSetDict* dictionary was created to specify the different regions dividing the full geometry. Within this dictionary, specific functions were defined. We used the `boxToCell` function, designed to generate a parallelepiped into the full geometry where all cells whose centers are inside the parallelepiped were selected. After configuring the dictionary, *TopoSet* was executed. Subsequently, we applied the *splitMeshRegions* utility to divide the full geometry mesh into different regions. This tool guarantees that each created region retains assigned properties, such as the average velocity and distribution of water and bubbles. If *splitMeshRegions* is not specified, the generated regions will not contain this data.

In Fig D, we observe the functionality of the *TopoSet* and *splitMeshRegions* utilities, showing the transformation of the quarter-cylinder mesh into an axisymmetric mesh. The

process reduces the number of cells from 5243400 to 88200 while maintaining the numerical values of the original mesh.

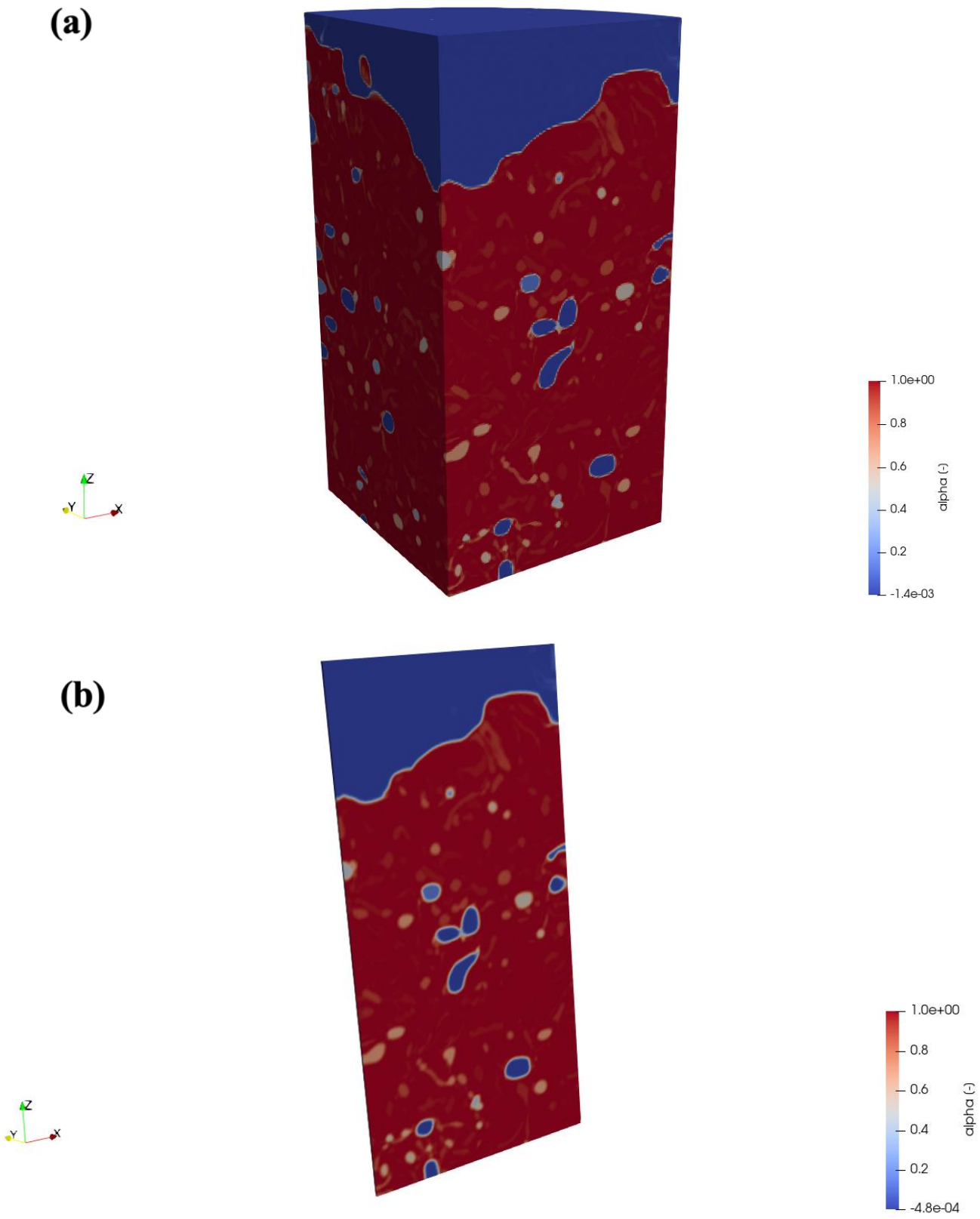

Fig D: Bubble distribution. (a) Quarter-cylinder mesh (full geometry). (b) Resulting mesh (compartment model) generated using the *TopoSet* and *spliMeshRegions* utilities.

2.4 Leakage

The scalar quantities usually possess different properties in two phases, resulting in scalars being effectively confined to one of the phases. The confinement of scalars in one of the phases leads to the formation of sharp gradients in scalar concentration values, causing numerical leakage (artificial numerical diffusion) at the interface (Jain and Mani, 2023). For example, if dissolved hydrogen is present in the gas phase, this constitutes numerical leakage, which can lead to the failure of the calculations or even cause them to diverge.

To avoid that, we will check for numerical leakage in our implemented model for the meshes generated in the section 2.3. Hence, the following equation has been integrated:

$$Excess\ percentage = \frac{|Z_t - Z_{max}|}{Z_{max}} \tag{6}$$

Where Z_t represents the current value of the biological model variable at time t, and Z_{max} represents the maximal value of the biological variable over the entire domain.

For a better understanding, we will use hydrogen as an example for the two scalar model. Hydrogen, in addition to the equations presented in section 2.1, also follows these:

$$p_{H2} = \psi\widehat{[H_2]}_g RT \tag{7}$$

$$SH_2^{sat} = \alpha K_{H,H2} p_{H2} \tag{8}$$

In liquid phase, the maximum quantity of dissolved hydrogen (SH_2^{sat}) is determined at every spatial interval. Thus, we use the values of hydrogen in the gas phase at every step to determine the partial pressure (eq. 7), and with this partial pressure, we obtain the saturated dissolved hydrogen (SH_2^{sat}) for each spatial interval and every time step (eq. 8). This allows us to determine our parameter Z_{max} in eq. 6, leading to the equation applied to hydrogen in the liquid phase:

$$Excess\ percentage = \frac{\left|\widehat{\alpha S_{H2,t}} - SH_2^{sat}\right|}{SH_2^{sat}} \quad (9)$$

A similar equation is applied to all other biological model variables in both liquid and gas phases of our model (See section 3 for results).

REFERENCES

Janke, D., Caiazzo, A., Ahmed, N., Alia, N., Knoth, O., Moreau, B., Wilbrandt, U., Willink, D., Amon, T., John, V., 2020. On the feasibility of using open source solvers for the simulation of a turbulent air flow in a dairy barn. Computers and Electronics in Agriculture 175, 105546. https://doi.org/10.1016/j.compag.2020.105546

(Janke et al., 2020)

Islas, A., Rodríguez-Fernández, A., Betegón, C., Martínez-Pañeda, E., Pandal, A., 2022. CFD simulations of turbulent dust dispersion in the 20 L vessel using OpenFOAM. Powder Technology 397, 117033. https://doi.org/10.1016/j.powtec.2021.117033

(Islas et al., 2022)

He, Z., You, J., Kang, D., Zou, Q., Zhang, W., Zhang, Z., 2024. Overall numerical simulation of chemical-thermal-electric conversion for an all-in-one thermoelectric generator based on micro scale combustion. Energy 292, 130307. https://doi.org/10.1016/j.energy.2024.130307

(He et al., 2024)

Capson-Tojo, G., Batstone, D.J., Hülsen, T., 2023. Expanding mechanistic models to represent purple phototrophic bacteria enriched cultures growing outdoors. Water Research 229, 119401. https://doi.org/10.1016/j.watres.2022.119401

(Capson-Tojo et al., 2023)

Puyol, D., Barry, E.M., Hülsen, T., Batstone, D.J., 2017. A mechanistic model for anaerobic phototrophs in domestic wastewater applications: Photo-anaerobic model (PAnM). Water Research 116, 241–253. https://doi.org/10.1016/j.watres.2017.03.022

(Puyol et al., 2017)

Greenshields, C., 2020. OpenFOAM v8 User Guide [WWW Document]. CFD Direct. URL https://doc.cfd.direct/openfoam/user-guide-v8/index/ (accessed 3.7.25).

(Greenshields, 2020)

Qin, R., Bhadeshia, H., 2010. Phase field method. Materials Science and Technology 26, 803–811. https://doi.org/10.1179/174328409X453190

(Qin and Bhadeshia, 2010)

Jain, S.S., Mani, A., 2023. A computational model for transport of immiscible scalars in two-phase flows. Journal of Computational Physics 476, 111843. https://doi.org/10.1016/j.jcp.2022.111843

(Jain and Mani, 2023)