Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

A LAKE CYANOBACTERIA COLONY DYNAMICS SIMULATION SUPPORTED BY SPH

Samuel Ferrero-Losada[1], José L. Risco-Martín[1], José A. López-Orozco[1], and Eva Besada-Portas[1]

[1]Dept. of Computer Architecture and Automation, Universidad Complutense de Madrid, Madrid, Spain

ABSTRACT

Addressing the prediction of Harmful Cyanobacterial Blooms (CyanoHABs) is critical due to the increasing strain on global water resources from climate change and overexploitation. This paper introduces a combined physical and biological modeling approach for simulating the 3D migration, growth, and decay of cyanobacteria colonies in water bodies. Utilizing the Smoothed Particle Hydrodynamics (SPH) method, our model accommodates complex geometries with high accuracy. This adaptable open-source framework significantly enhances the simulation of cyanobacteria migration in three dimensions, a capability often lacking in existing environmental lake simulators. Moreover, it is being integrated into an advanced early warning system, representing a digital twin of the water body. This integration aims to improve the prediction and management of CyanoHABs, contributing to safeguard water quality and ecosystem health.

1 INTRODUCTION

The quality of waters in lakes, rivers and reservoirs has become an increasing concern in recent years, due to, among other factors, the frequent occurrence of Harmful Cyanobacterial Blooms (CyanoHABs), which endanger the health of people and local ecosystems. Consequently, both national and international entities have begun to regulate how to take action through monitoring and preventive action programs, which aim to predict the onset of CyanoHABs with enough lead time to enable the planning and execution of effective responses (United States Environmental Protection Agency 2023, European Comission 2023)

Current water monitoring systems often provide incomplete information about the actual conditions of water bodies, which is collected by manually taking samples from boats at few locations or automatically acquiring data from stationary probes. The former approach typically lacks temporal resolution, while the latter only surveys a fixed point of the environment. This situation requires the development of more innovative, automated and high-performance solutions. Among the existing challenges, the optimization of measurement points can be addressed using models to predict the location of cyanobacteria. These models, in turn, should account, at least, for the water body flow and the cyanobacteria behavior. For this problem, the hydrodynamic model must have an adequate grade of granularity to deal with different complex bathymetric geometries and support the definition of dissolved substances and suspended particles.

Specific aquatic body simulators (e.g Delft3D , EEMS , UNTRIM) and multiphysics simulators (e.g. COMSOL Multiphysics® 1998) could be used for this case. But, as it will ultimately be integrated in the advanced early warning system DEVS-BLOOM (Risco-Martín et al. 2023), it should focus on the required task to better adjust the invested resources. The previous examples are more complex than what is required for our case, including more features and parameters than needed, not focusing on the hydrodynamic aspects of the ecosystem, or working only in one or two dimensions. The ideal simulator for obtaining the stationary lake flow would be one that accounts for three dimensions, has enough granularity to define any type of water body geometry, and enough resolution in the vertical axis to account for vertical flow and mixing. Ideally, it should also be easily configurable for different scenarios through a graphic interface.

Mainstream mesh-based simulation methods such as Raynolds averaged Navier-Stokes by Reynolds (1895) and Lattice-Boltzmann by Chen and Doolen (1998) have a hard time dealing with the complex geometries, free surfaces and interfaces appearing during our studies. Moreover, they harden the process

979-8-3315-3420-2/24/$31.00 ©2024

as they require defining different complex meshes for different scenarios. Alternatively, Smoothed Particle Hydrodynamics (SPH, Gingold and Monaghan 1977), a particle-based fluid simulation method, presents itself as a better (although somehow immature) option to deal with these problems. It has undergone various adaptations, additions and corrections, and currently supports different options to deal with recognized instabilities and problematic phenomena. Many of them are available in the open-source software DualSPHysics (Domínguez et al. 2022), which combines mathematical simulation tools with fast Graphical Processing Unit (GPU) computing capabilities, resulting in a promising candidate for our case, as it provides easy to use, adaptable and adequate three-dimensional granularity simulation capabilities.

In this paper, we present how to use this software in combination with the biological model of CyanoHAB colonies by Carazo-Barbero et al. (2023) in order to compute the movement, growth and decay of these organisms within a lake. This biological model accounts for the density variation of the colonies (which produces their vertical migrations) and for their growth/decay, all of them dependent on the hourly changes of light radiance. The paper also presents a practical case of the integration of all the models, preceded by detailed descriptions of the formulation and scenario build process. Additionally, it reports abnormal behaviors of our large time-scale SPH simulations, which, to the best of the authors knowledge, have not been previously presented in the literature, usually focused on shorter simulations (Pilloton et al. 2022).

2 FLUID DYNAMICS MODELING

In this section a brief discussion of simulation models introduces the reader to the weakly compressible SPH model, which is then explained in its original version. Afterwards, several additions (such as density field corrections or boundary and time-step definition schemes), necessary for the model to be fully functional in our case, are described.

The large variety of scenarios where the fluid dynamics equations by Navier-Stokes (Fernández Cara 2005) are applicable has led to the development of different models that make assumptions on the boundary conditions to simplify the problem, improving simulation efficiency. Generally, the most popular models use a mesh to discretize the space, which eases the definition of basic boundary conditions and guarantees mass and flux conservation by nature. However, these grid-based methods raise problems when dealing with complex spatial geometries (e.g. the bed, inlets and outlets of real water bodies) or any moving boundary, as they tend to reduce the granularity and cannot easily evolve dynamically (Liu and Liu 2010).

As our objective is to simulate the dynamics of CyanoHABs in small real-world lakes given their shapes (provided by bathymetric data), water sources and drains; the granularity requirements are important to deal with complex boundary geometries. Furthermore, as our advanced monitoring system (Risco-Martín et al. 2023) also collects measurements of dissolved substances in water such as oxygen, nitrates and phosphates, it is also preferable to use a fluid simulation capable of also dealing with the diffusion and transport mechanisms of this substance. In order to face these challenges, it is better to use some mesh-less Computational Fluid Dynamics (CFD) methodologies, such as the ones supported by SPH.

2.1 Smoothed Particle Hydrodynamics (SPH)

SPH is a fluids simulation method that was originally developed for astrophysics (Gingold and Monaghan 1977) and that is currently used in many other fields. Its application in CFD is currently under development and improvement, having appeared many new papers published in the last decade about model refinements and different versions for compressible, incompressible, viscous or inviscid flows. Hence, it presents itself as a valid, yet somehow immature, solution to flow simulation (Violeau and Rogers 2016).

The main characteristic of SPH is that it is a meshless method which uses material particles to discretise the space. The positions of these particles, $\vec{r}_i$, are then used as location for locally integrating the Navier-Stokes equations, taking into account the surrounding particles physical properties. The contribution of each particle to their neighbors is defined by a positive kernel function, $W(\cdot)$, with the form of the quintic spline defined in Wendland (1995), which determines the influence domain of each particle up to twice the associated smoothing length, h. Particle's physical related variables such as density, speed or acceleration are interpolated using (1), where $A(\vec{r})$ stands for the interpolated variable at location $\vec{r}$, and ρ_j and m_j for the density and mass of the corresponding j closer-than-$2h$ particles.

$$A(\vec{r}) = \sum_j \frac{m_j}{\rho_j} A(\vec{r}_j) W(|\vec{r}_i - \vec{r}_j|, h) \quad ; \quad W(r,h) = \frac{21}{16}\pi h^3 \left(1 - \frac{r}{2h}\right)^4 \left(\frac{2r}{h} + 1\right) \quad 0 \leq r \leq 2h \tag{1}$$

Afterwards, the particles speed, $\vec{v}_i$, is defined by a momentum conservation equation where gravity and dissipative terms are taken into account. The latter follows the artificial viscosity scheme given in (2), commonly used with SPH due to its simplicity, where the subindex *i-j* stands for the difference of the given parameter/variable (e.g. $\vec{r}_{i-j} = \vec{r}_i - \vec{r}_j$), $\mu_{ij} = h\vec{v}_{i-j} \cdot \vec{r}_{i-j} / (|\vec{r}_{i-j}|^2 + 0.01h^2)$ for an artificial viscosity, $\overline{c_{ij}} = 0.5(c_i + c_j)$ for the mean speed of sound of both particles, α for the tuning parameter for dissipation (typically set to 0.01), P_j for each particle's pressure, and $W_{ij} = W(|\vec{r}_{i-j}|, h)$.

$$\frac{d\vec{v}_i}{dt} = \sum_j \left(\frac{P_j + P_i}{\rho_j \cdot \rho_i} + \Pi_{ij}\right) \vec{\nabla}_i W_{ij} + \vec{g} \quad ; \quad \Pi_{ij} = \begin{cases} \dfrac{-\alpha \overline{c_{ij}} \mu_{ij}}{\overline{\rho_{ij}}} & \vec{v_{i-j}} \cdot \vec{r}_{i-j} < 0 \\ 0 & \vec{v}_{i-j} \cdot \vec{r}_{i-j} > 0 \end{cases} \tag{2}$$

Regarding the density field evolution, the continuity equation in (3) defines the density distribution evolution along the continuous space while the mass of each particle remains constant throughout the simulation. However, as our objective is to obtain incompressible fluxes (since water behaves this way), we introduce (4) in order to make the fluid weakly compressible; adjusting compressibility to artificially lower the speed of sound, limiting it to up to ten times the maximum fluid velocity; keeping density variations to less than 1%, and preventing further deviations from the incompressible regime.

$$\frac{d\rho_i}{dt} = \rho_i \sum_j \frac{m_j}{\rho_j} \vec{v}_{i-j} \cdot \vec{\nabla}_i W_{ij} \tag{3}$$

$$P_i = b\left[\left(\frac{\rho_i}{\rho_0}\right)^\gamma - 1\right] \; with \; \begin{cases} b = c_0^2 \rho_0 / \gamma \quad ; \quad c_0 = \sqrt{(\delta P_i / \delta \rho_i)}|_{\rho_0} \\ \rho_0 = 1000 kg/m^3, \; \gamma = 7 \end{cases} \tag{4}$$

2.2 Necessary adjustments to weakly compressible SPH

Since gravity is essential in our problem, it is necessary to ensure that this external force is correctly computed to avoid generating abnormal behaviors in the fluid. Among other issues, the base SPH model has trouble computing the density column correctly, generating abnormal behaviors and abrupt density oscillations due to the disordering of particles. We solve this situation as in Molteni and Colagrossi (2009), adding a new term to (3) in (5), where Ψ_{ij} is taken from the work of Fourtakas et al. (2020) as a correction for cases where gravity is dominant, and δ_Φ typically adopts the value of 0.1.

$$\begin{gathered} \frac{d\rho_i}{dt} = \rho_i \sum_j \frac{m_j}{\rho_j} \vec{v}_{i-j} \cdot \vec{\nabla}_i W_{ij} + \delta_\Phi h c_0 \sum_j \vec{\Psi}_{ij} \cdot \vec{\nabla}_i W_{ij} \frac{m_j}{\rho_j} \quad ; \quad \vec{\Psi}_{ij} = 2\left(\rho_{i-j} - \rho_{ij}^H\right) \frac{\vec{r}_{i-j}}{|\vec{r}_{i-j}|^2} \\ \rho_{ij}^H = \rho_0 \left(\sqrt[\gamma]{\frac{(\rho_0 \vec{g} \cdot \vec{r}_{i-j} + 1)\gamma}{c_0 \rho_0}} - 1\right) \end{gathered} \tag{5}$$

Besides, due to its low computation cost, we will use the Verlet time stepping scheme in (6), combined with a variable time step. This step results from the evaluation of the Courant-Friedichs-Lewy condition (Monaghan and Kos 1999) stated in (7), with either the forcing term (Δt_f, with f_i being the total force per unit mass) or the viscous diffusion one (Δt_{cv}). However, due to time integration being over a staggered time interval, the density and velocity equations are decoupled, which may lead to divergence. To prevent this, an intermediate step is inserted every 40 iterations, following (6) but eliminating the factor 2 in the density and velocity expressions.

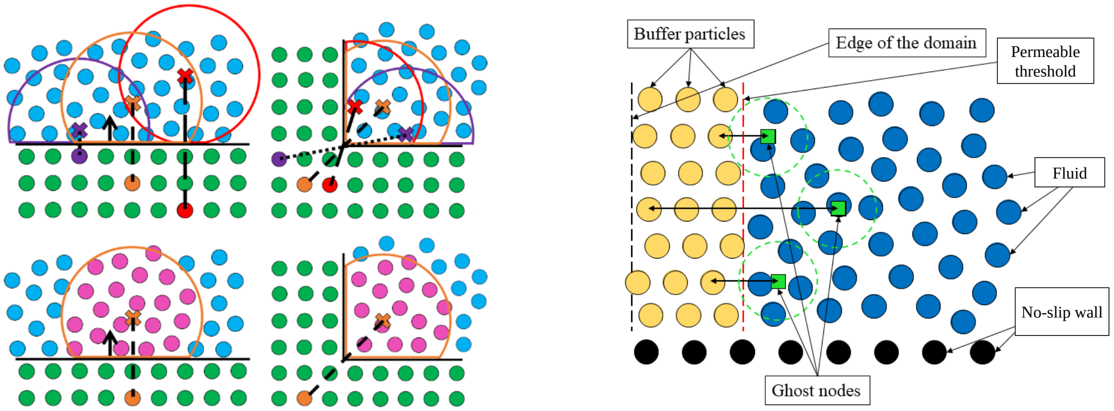
$$\vec{v}_i^{\,n+1} = \vec{v}_i^{\,n} + 2\Delta t \frac{d\vec{v}_i}{dt} \quad ; \quad \rho_i^{n+1} = \rho_i^n + 2\Delta t \frac{d\rho_i}{dt}; \quad \vec{r}_i^{\,n+1} = \vec{r}_i^{\,n} + \Delta t \vec{v}_i^{\,n} + 1/2\Delta t^2 \frac{d\vec{v}_i}{dt} \tag{6}$$

$$\Delta t = min\left(\Delta t_f, \Delta t_{cv}\right); \quad \Delta t_f = min\left(\sqrt{h/|f_i|}\ \forall i\right); \quad \Delta t_{cv} = min\left(h \Big/ \left(c_s + max\left(\frac{h\vec{v}_i \cdot \vec{r}_i}{r_{ij}^2 + 0.01h^2}\right)\right) \forall i\right) \tag{7}$$

Additionally, both open and no-slip boundaries are necessary to create the lake scenario. In SPH, while all particles are subjected to the same equations, boundary particles either remain fixed in their initial position (lake bottom) or move according to inlet/outlet conditions.

In the Dynamic Boundary Conditions (DBC, Crespo et al. 2007) method, when a fluid particle comes closer than $2h$ to a boundary, the density in the latter increases, with a consequent repulsive force on the nearby fluid. Given a short enough time step to handle the greatest velocities close to boundaries, this method is stable. Thus, considering Δt_f when calculating the time step (7) is relevant.

However, DBC produces an extensive boundary layer around walls, making the density field around boundary particles reach nonphysical values. Moreover, for certain flows, a high resolution is needed for this method to work properly, which leads to higher-than-expected velocities in the fluid close to the non-slip boundaries. To tackle these problems, the DBC modification (mDBC) by English et al. (2021) is implemented. Hence, to maintain the particle configuration, a virtual interface is inserted between fluid and wall layers and ghost nodes are situated mirroring each particle's position along the normal of the interface. In the corners, the node projection is made through the corner point (see Figure 1a).



(a) Node projection and field interpolation radius.

(b) Open boundary scheme from Tafuni et al. (2018)

Figure 1: Open and no-slip boundary definition.

The boundary particles density is extrapolated from the ghost nodes by interpolating the density field. With these values and their gradient, the total density of the boundary particles can be found through a standard first-order Taylor series expansion. This method provides smoother density fields along the walls, reducing the gap between fluid and boundary particles generated by DBC. Finally, ghost node speeds, $\vec{v}_g$, calculated through (8), are inverted to obtain the boundary particles speed value. This way, the no-slip condition enforcement is improved, efficiently reducing fluid speed along the walls.

$$\vec{v}_g = \frac{\sum_j \vec{v}_j W_{gj} m_j/\rho_j}{\sum_j W_{gj} m_j/\rho_j} \tag{8}$$

For open boundaries (water inlets and outlets), where specific flow conditions must be enforced, Tafuni et al. (2018) algorithm is used. As Figure 1b shows, a wide buffer zone of SPH particles is defined, in order to ensure full kernel support for the fluid. Boundary conditions are enforced on the buffer, predefining the density or speed values, or interpolating their fields in the nodes situated mirroring the buffer particles across the permeable threshold. However, open boundary nodes are so close to the buffer that standard particle interpolation does not work correctly. Instead, a multi-dimensional first-order Taylor series approximation of the field function times the kernel is implemented (Liu and Liu 2006). Depending on the result, buffer particles move towards or away from the threshold. When one crosses this limit, it becomes a free fluid particle, and the buffer is refilled with a new particle at the edge of the domain. If a fluid particle enters the buffer, boundary conditions are enforced. When crossing the domain edge, they are discarded.

All these improvements make SPH functional. However, we have observed that it still presents instability issues related to anisotropic particle spacing, which can increase the noise in the velocity field and create voids of particles within the water domain, decreasing the field's resolution. To prevent this situation, the particle shifting algorithm by Lind et al. (2012) is introduced, to make the particles stay uniformly spread along the domain. The shift is regulated by Fick's first law of diffusion in (9), linking the diffusion flux to the concentration gradient. The proportionality coefficient, D, and the concentration gradient, $\vec{\nabla} C_i$, are defined to make the shifting big enough to solve the described problems without adding new instabilities.

$$\delta\vec{r}_s = -D\vec{\nabla}_i C_i \quad ; \quad D = 2h|\vec{v}_i|dt \quad ; \quad \vec{\nabla}_i C_i = \sum_j \frac{m_j}{\rho_j}\vec{\nabla}_i W_{ij} \tag{9}$$

This solution heavily depends on full kernel support, raising problems when dealing with the free surface fluid particles, as they would be diffused all over the empty space. To solve it, the correction by Xu et al. (2009) is implemented, restricting the diffusion near the free surface to be tangent to it. Particle proximity to this zone is evaluated in (10) by comparing the particle number divergence $\nabla \cdot \vec{r}$ with the threshold value $A_{FST} = 2.75$. When $\nabla \cdot \vec{r} - A_{FST}$ is negative, the shifting is attenuated in (11) by A_{FSC}.

$$\nabla \cdot \vec{r}_i = \sum_j \frac{m_j}{\rho_j}\vec{r}_{i-j} \cdot \vec{\nabla}_i W_{ij} \quad ; \quad A_{FSC} = \frac{\vec{\nabla} \cdot \vec{r}_i - A_{FST}}{3 - A_{FST}} \tag{10}$$

$$\delta\vec{r}_s = \begin{cases} A_{FSC} 2h|\vec{v}_i|dt \cdot \vec{\nabla}_i C_i & \text{if } (\vec{\nabla} \cdot \vec{r}_i - A_{FST}) < 0 \\ -2h|\vec{v}_i|dt \cdot \vec{\nabla}_i C_i & \text{if } (\vec{\nabla} \cdot \vec{r}_i - A_{FST}) \geq 0 \end{cases} \tag{11}$$

All the above formulation conforms to the fluid model required for this work. However, it is unnecessary to code everything from scratch, as we can use DualSPHysics to perform the simulations for the scenarios under study. Furthermore, the FreeCAD macro DesignSPHysics (Vieira et al. 2017) supports the definition of the scenarios with a graphical 3D environment, where configuring the inlets, outlets, and walls geometry becomes easy and agile. This macro also streamlines the generation of the initial SPH simulation according to the 3D design, defining which particles are fixed boundaries, open ones, or fluid. This ready-to-run case, in turn, may be executed through this interface. Additionally, several post-processing tools are available too. We have decided to use DualSPHysics, since this open-source SPH simulator improves the efficiency of the original SPHysics FORTRAN code (Gómez-Gesteira et al. 2012) due to its careful C++ and CUDA implementation, which reduces the memory requirements of SPHysics and enables the use of advanced parallel computing strategies of GPUs.

3 CYANOBACTERIA DYNAMIC MODEL

The resulting velocity field from the fluid simulation is combined with the dynamic model of cyanobacteria colonies presented in Carazo-Barbero et al. (2023) in order to obtain the 3D temporal evolution of their concentration. The model takes into account the vertical penetration of the sun irradiance and the colony density to compute the natural vertical migration of the cyanobacteria. This is done by emulating the density dynamics produced by the biophysical processes that occur inside their cells as a response to different stimulus.

To calculate sun irradiance at the position of each cyanobacteria particle (i.e. cell), we assume that the whole lake receives the same amount of light on every surface point. This value is defined by a homebrew light function consisting of a sine, with frequency given by user-defined sunrise and sunset hours, multiplied by a maximum irradiance constant. Under these assumptions, Beer-Lambert law (Bouguer 1923) is used to evaluate the exponentially attenuated irradiance at each cell depth and time $I(z,t)$.

The rate of change of the density of every cell $\rho^k_{cell}(t)$ is calculated according to two different regimes of received light that (12) considers: on brighter conditions the density generally increases (according to the normative factor β^+, the photo-inhibition limit I_0, and the dark regime mean rate of growth γ) as long as the received light is not too high; on darker conditions the change rate is negative (proportional to the density decrease rate, β^-, and current cell density) and limited by the Heaviside function $H(\cdot)$ to ensure that the density does not decrease under its minimum value C_p.

$$\frac{\delta}{\delta t}\rho^k_{cell}(t) = \begin{cases} \beta I(z^k,t)exp\left(-I(z^k,t)/I_0\right)+\gamma & I(z,t)\geq I_{lim} \quad \text{(on brighter conditions)} \\ -\beta^-(\rho^k_{cell}-C_p)H(\rho^k_{cell}-C_p) & I(z,t)<I_{lim} \quad \text{(on darker conditions)} \end{cases} \tag{12}$$

The results of integrating (12) are used in (13) to update the cells density in the colony $\rho^k_p(t)$, where n_{cell} and n_{gas} refers to proportion of cell and gas in the colony, and ρ_{muc} to the mucilage density.

$$\rho^k_p(t) = \rho^k_{cell}(t)\cdot n_{cell}(1-n_{gas})+\rho_{muc}(1-n_{cell}) \tag{13}$$

The colony growth model decides whether particles are created or removed. The time-step growth rate is obtained in (14, where K_*, σ_{gr}, and n_{max} are tuning parameters) combining a growth term, a linear death rate term and a normal distortion to give some variability between colonies. If n_{grow} is positive, n_{grow} particles are generated in a normal distribution around the parent, with its same velocity and density. If negative, the closest $|n_{grow}|$ particles to the parent are discarded.

$$n^k_{grow} = \frac{n_{max}\left(1+K_{gr}\frac{I(z^k,t)}{I_0}\right)}{\left(\frac{I(z^k,t)}{I_0}\right)^2+2K_{gr}\frac{I(z^k,t)}{I_0}+1}+\left(K_d I(z^k,t)+K_{d0}\right)+N(0,\sigma_{gr}) \tag{14}$$

Finally, the colony transport speed is computed with (15) by combining the water velocity field ($\vec{v}_f$) and the current density of each colony with a random walk diffusion term . Here, $\vec{g}$ is gravity acceleration, ρ_f the fluid density, ν is the fluid's kinematic viscosity, d_p the particle diameter, D_0 the diffusion coefficient, and $\vec{r_D}$ a random 3D vector sampled from a standard Gaussian distribution.

$$\left(1+\frac{\rho_f}{2\rho^k_p}\right)\frac{d\vec{v}^k_p}{dt} = \left(1+\frac{\rho_f}{\rho^k_p}\right)\vec{g}+\frac{18\nu\rho_f}{d^2_p\rho^k_p}\left(\vec{v}_f-\vec{v}^k_p\right)+\frac{3\rho_f}{2\rho^k_p}\left(\frac{\delta\vec{v}_f}{\delta t}+(\vec{v}_f\cdot\nabla)\vec{v}_f\right)+D_0\vec{r}_D \tag{15}$$

As the density, speed and acceleration fields are defined only in the SPH particles positions, to implement this model these fields must be interpolated so that values can be found for any position in the domain. Said interpolation consist in a weighted mean where each weight is the inverse of the distance between the colony position and the fluid particle.

4 STUDY CASE

The objective of this work is to simulate CyanoHABs evolution in real-world lakes. We have chosen the Campillo lake-bed, situated in Madrid (Spain), for our simulations, since we have its bathymetric profile and it is frequently used by our project members to conduct part of their experiments. However, we have modified its water inlet and outlet configuration: while water enters and exits only through soil filtration in the actual lake (as it is an old gravel quarry flooded by nearby Jarama river), in the simulation, the inlets and outlets are treated as channels (or rivers) that flow into or out of the lake. This change is made because our project primarily targets larger lakes with a dam interrupting the river's flow. However, flow speed scale is adapted to emulate the one observed during practical experiments at these locations.

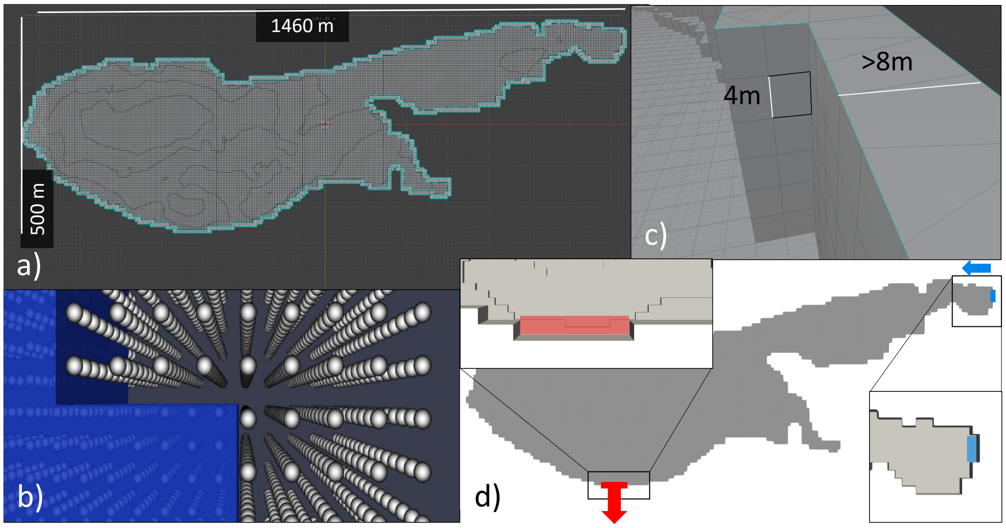


Figure 2: a) Lake walls figure dimensions. b) mDBC particle-surface separation mismatch. c) Cubic re-mesh and wall width. d) Lake inlet (blue) and outlet (red) location.

4.1 Building the scenario

Boundary definition is a critical aspect of SPH simulations. As explained in section 2, this method requires not only the definition of a boundary region (to be filled with particles) but also a continuous surface near the inner boundary layer. Furthermore, DualSPHysics software generates particles based on a cubic mesh, requiring a careful adaptation of the irregular bathymetry geometry to this mesh. To do it, the original bathymetry file was pre-processed. First, high vertical walls were added to enclose the lake. Next, to ensure that the boundaries have the correct particle thickness, the entire figure is enveloped in another surface, replicating the lake. Subsequently, the 3D mesh was reorganized into a cubic one, as Figure 2 shows, using the *remesh* tool in Blender. This new mesh has a cube side of 4 meters, and a wall thickness of no less than 8 meters. These dimensions are determined by the inter-particle distance, dp, which is set to 2 meters.

To generate the inner surface required for implementing mDBC, a process similar to the one used for external boundaries was followed, but without enclosing the geometry within an external wall. However, achieving a uniform spacing between the geometry and the boundary particles throughout the domain proved to be challenging due to discrepancies at corners and edges, where the surface and the particle boundaries meet. Figure 2.b illustrates this issue, showing that the corner of the continuous surface (in blue) is not evenly spaced from the boundary particles (in white). However, once changes were made, despite ensuring that the separation was bigger than $dp/2$, we observed no significant impact on the fluid behavior. Consequently, for the sake of simplicity, the boundary configuration was maintained as originally defined, with particles close to the inner surface. It is conceivable that this heterogeneity could have significantly affected the wall's particles, but since the density field of the walls is irrelevant to this study, their effect was not considered. Regarding the no-slip condition, each wall is constructed with a minimum of 4 layers of particles to ensure full kernel support, whose radius is $4dp$, since $h = 2dp$.

Remeshing the lake bed into a cubic mesh significantly alters its geometry. However, it is important to note that most features removed by this process are smaller than the 4-meter cubes comprising the new mesh. Since SPH movement is determined by the cumulative influence of all neighboring particles, any obstacle must be composed of many particles to be noticeable. This makes enforcing no-slip conditions on small wall features very challenging. If we were to use the original lake bed contour for mDBC, the system would struggle to conform the cubic mesh to such shape, potentially placing particles outside the boundary, within the fluid domain. This would exacerbate the issue described earlier, with particles not only too close to the boundary interface (the blue one in Figure 2.b) but also on the fluid domain side.

As previously mentioned, the fluid inlet and outlet are implemented as channels, positioned as shown in Figure 2.d). The inlet channel has dimensions 25x10m, while the outlet measures 68x10m. These dimensions have been chosen to match the size of the wall pockets they occupy. The channels are defined with at least 4 layers of buffer particles to prevent wall particles from influencing the fluid across the buffer.

Both channels have fixed velocities to prevent fluid from exiting through the inlet or entering through the outlet. The density of the buffer particles is extrapolated from the adjacent fluid to ensure a smooth transition. The velocities will not exceed 1m/s to maintain typical lentic waters flow rates. As our goal is to capture the general steady flow in the lake, we must avoid significant water level fluctuations during the simulation. To achieve this, the velocities at the inlets and outlets are adjusted to 0.1 m/s and 0.03 m/s respectively, according to the channel cross-sections to balance the inflow and outflow.

Regarding the parameters used in the subsequent simulations and tests presented, all adjustable parameters provided by DualSPHysics have been left at their default values, except for the maximum simulation time and the time step between data saves.

4.2 Impact of Modified Dynamic Boundary Conditions and Density Diffusion on Water Level

This paper has so far outlined the methodology for obtaining results, and the features and tools employed. Initially, for simplicity, the model did not incorporate mDBC or the advanced density diffusion term. However, during the water level stabilization phase, both improvements became necessary. An anomalous rise of several meters in the water level, irrespective of the number of fluid particles, was found. This subsection graphically demonstrates how both improvements rectified this issue, providing a valuable study case for future DualSPHysics users on the benefits of their use for improving simulation outcomes.

The study of the particle dynamics revealed that the water level rise was more pronounced in the deepest areas of the lake, where the water column is taller. Another irregular behavior was detected at the bottom, among particles in contact with the boundary walls, where the fluid's deepest layers flowed horizontally towards even deeper layers along the wall, gliding over the cubic mesh steps. The flow velocity was higher where more uneven terrain was found. To investigate these issues, we created a test scenario, featuring a box with a stepped bottom and several fluid layers between the high and low points. The box dimensions ensured that the fluid's central section remains unaffected by the walls, while the simulations (with the particle shifting algorithm enabled) were conducted without external forces (except for gravity), inlets, or outlets. The first simulation of the section used the density diffusion term from Molteni and Colagrossi (2009) and DBC (to replicate the misleading conditions), while the others have the correction enabled too.

As Figure 3.a, 3.b and 3.c show, the first simulation results in an unusual density field. Although there was appropriate progressive stratification, the deepest layer of particles acquired a very low density at the start of the simulation. Additionally, on the right side of the step, the maximum density exceeded that at the same height on the left side, leading to instability. This condition generated a vertical flow along the step, where the low-density layer near the walls moves upwards. Simultaneously, the denser fluid on the right side moved towards the left and downwards, creating a vortex over the step corner, which induced movement in particles situated further from the bottom. Regarding the water level, Figure 3.d shows that after 30 minutes, the water surface rose 20 meters, twice the inter-particle distance in this test, which is a concerning amount for what should be a hydrostatic scenario.

As the walls were not behaving correctly, inducing unrealistic densities in nearby particles, mDBC was implemented. With mDBC, the results improved significantly. No vortex formed and the density stratification was complete, with the lower densities disappearing in the deepest layer. However, as Figure 3.e shows, the water level continued to rise, but with only the upper layers of fluid exhibiting a positive vertical velocity, v_z, which diminishes as density increases. As this suggests a relationship between these vertical movements and density, the Fourtakas et al. (2020) density diffusion term correction is implemented.

The final test show that the previously outlined issues were corrected, yielding a realistic and stable solution with no significant flows and a stable water level, with variations well below the inter-particle distance. Figure 3.f displays the stratified density field, which is very similar if not identical to the initial

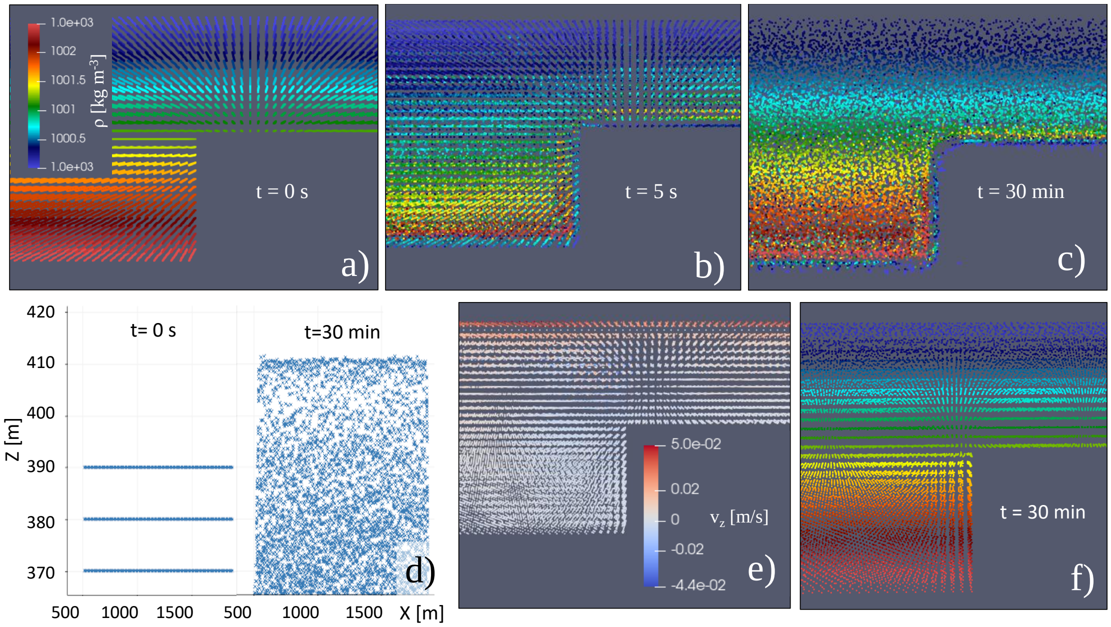


Figure 3: Column stability test. a), b), c) Particle density at different timesteps: initial, first registered and 30 minutes. d) Water level change in 30 minutes. e) Vertical velocity field with mDBC implemented. f) Particle density after 30 minutes with mDBC and Fourtakas et al. (2020) density diffusion term.

one in 3.a, with particles close to their initial positions. After half an hour, the starting cubic mesh remains intact without flow. Following these results mDBC and Fourtakas' density term were applied to the lake simulation. However, as the simulation progressed over several hours, the rise in water level reappeared, as the previous solutions only diminished the effect, significantly slowing the phenomenon.

The parameter δ_ϕ, which controls the intensity of the density diffusion term, was adjusted to enhance the simulation outcomes. We experimented with values between 0.1 (default setting) and 0.01, and observed that the fluid's behavior exhibits significant changes over this narrow range. Specifically, density variations within the fluid became markedly abrupt when the parameter value was reduced below 0.052. Conversely, a setting above 0.053 resulted in a gradual increase in the water level, consistent with our previous observations. Considering that the SPH fluid models are not typically employed for extended simulations (spanning several hours), the issues we encountered may be attributed to the formulation of the density term, which seems to exhibit inherent instability over prolonged time. To mitigate this issue and prevent the observed phenomena from affecting the flow field's integrity, we will utilize a δ_ϕ value of 0.053, constrain our simulations to a maximum of 6 hours, and disregard data from the surface regions in our analysis.

5 RESULTS

With all the adjustments, a 6.5-hour fluid simulation was run to obtain the flow field of Campillo lake, which is afterwards applied to two different simulations of the CyanoHABs migration model: one starting at 12:00 and the other starting at 23:00, to observe the behavioural differences between day and night hours. This model's parameters have the same values used by Carazo-Barbero et al. (2023). The fluid simulation data was registered every 5 minutes, containing each particle's density, position, and speed values. The CyanoHABs simulation time step is 1 minute. Consequently, the flow data will be linearly interpolated each time step. Since the fluid SPH particles are homogeneously scattered, k-d trees are used to find the closest ones from each colony's position, limiting the calculations to the closer-than-$2dp$ neighbors.

Figures 4.a and 4.b show the shallow flow evolution where each particle has a fixed color and new ones generated at the inlet are showed in a progressively darker red. Several red patches of particles have

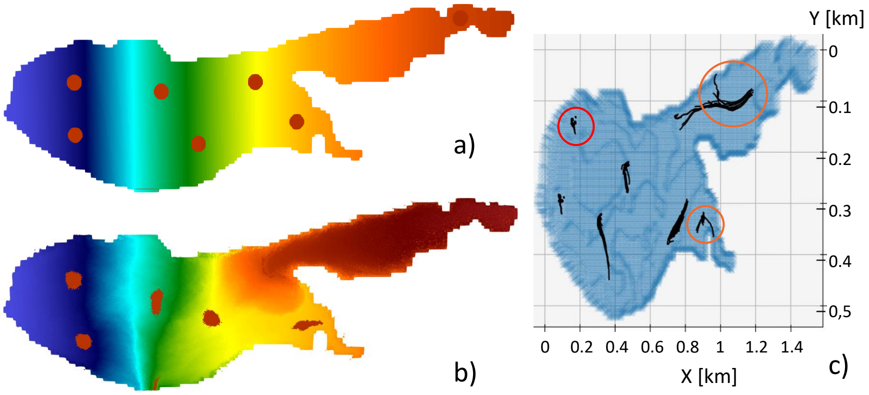


Figure 4: Fluid particles evolution: a) t = 0, b) t= 5h 30 min. c) 3D colony trajectories viewed from above.

been added for a more clear view of particle displacement. Figure 4.c presents the resulting colonies' trajectories. Orange circles indicate trajectories that somehow have already been trapped in a vortex or pushed towards shallow, calmer areas, where big colonies tend to accumulate in real-life scenarios.

The colony cloud circled in red in Figure 4.c is presented in more detail on the left side of Figure 5, which shows at its right the vertical radiance profile during the day hours (using a red curve to indicate the limit between light and dark growth regimes, and green to show the optimal radiance for growth). In the trajectory graph, the initial position of each colony is marked with a red dot. Black trajectories are produced when the simulation starts at 11:00 PM, while colorful ones when starting at 12:00 AM. The shallower starting points result in a more significant difference between night and day migration, produced by increased colony density when enough light reaches them. Deeper in the lake, less light creates progressively more similar paths. Deep enough colonies (e.g. the dark blue one) are in both cases in the dark regime, following trajectories that only differ by the Gaussian term displacements.

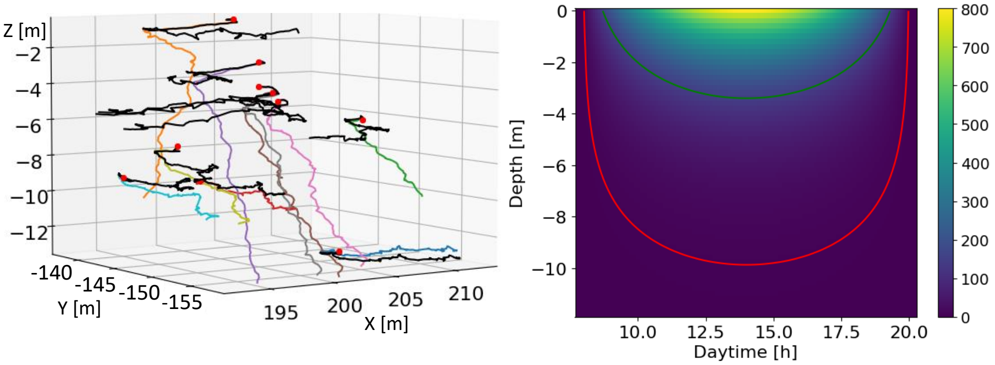


Figure 5: Left: 3D migration of colonies at night (black) and day (color). Right: Sun irradiance curb.

6 CONCLUSION AND FUTURE WORK

This study successfully demonstrates the feasibility of integrating SPH fluid simulation methods with dynamic models of cyanobacteria to predict the trajectories of cyanobacterial colonies in aquatic environments. Our approach allows for the identification of potential accumulation zones and the origins of colonies, providing valuable insights for water quality management and bloom prediction efforts. Despite encountering challenges related to the SPH formulation, particularly in managing the density field at free surfaces over extended simulation periods, our findings underscore the potential of this method in environmental modeling.

The integration of physical and biological models is a significant step forward in our understanding of CyanoHABs and their dynamics. By simulating the 3D migration, growth, and decay of cyanobacteria colonies, we can better predict bloom formation and develop strategies to mitigate their impact on water bodies. While the current SPH model has limitations in long-duration simulations, it offers a promising foundation for future research and development. The adaptability and granularity of SPH, combined with its ability to handle complex geometries, make it a valuable tool in the simulation of environmental phenomena.

Looking ahead, our research will focus on refining and enhancing the fluid dynamics model to address the challenges identified in this study. Real data combined with other fluid simulators, such as EEMS (EEMS), must be used to compare and validate the present methodology and to assess its robustness and efficiency. Additionally, we aim to develop more sophisticated biological models that incorporate a wider range of environmental factors, such as temperature gradients or nitrate and phosphate diffusion, which play a crucial role in cyanobacteria behavior and bloom formation.

The ultimate goal of our research is to contribute to the development of an advanced early warning system for cyanobacterial blooms. By integrating our simulation framework with real-time data and predictive analytics, we can create a powerful tool for water resource managers, and researchers (Risco-Martín et al. 2023). This system will not only improve our ability to predict and respond to CyanoHABs but will also enhance our understanding of their underlying mechanisms and impact.

Future work will also involve collaboration with interdisciplinary teams to validate and refine our models using field data. Through these collaborations, we aim to ensure that our simulation framework is robust, accurate, and applicable to a wide range of aquatic environments. By continuously improving our models and integrating them with emerging technologies, we can make significant strides in safeguarding water quality and ecosystem health.

ACKNOWLEDGMENTS

This work has been supported by the Research Projects IA-GESBLOOM-CM (Y2020/TCS-6420) funded by the Synergic program of the Comunidad Autónoma de Madrid (CAM), and SMART-BLOOMS (TED2021-130123B-I00) by MCIN/AEI/10.13039/501100011033 and the European Union NextGenerationEU/PRTR. We also want to thank Parques Regionales de la CAM for their support and access to Laguna del Campillo.

REFERENCES

Bouguer, P. 1923. “Essai d’optique sur la gradation de la lumière”. *Nature* 111:320 https://doi.org/10.1038/111320b0.

Carazo-Barbero, G., E. Besada-Portas, J. L. Risco-Martín, and J. A. López-Orozco. 2023. “EA-Based ASV Trajectory Planner for Detecting Cyanobacterial Blooms in Freshwater”. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO ’23, 1321–1329. New York, USA: Association for Computing Machinery https://doi.org/10.1145/3583131.3590484.

Chen, S. and G. D. Doolen. 1998. “LATTICE BOLTZMANN METHOD FOR FLUID FLOWS”. *Annual Review of Fluid Mechanics* 30(1):329–364 https://doi.org/10.1146/annurev.fluid.30.1.329.

Crespo, A. J. C., M. Gómez-Gesteira, and R. A. Dalrymple. 2007. “Boundary Conditions Generated by Dynamic Particles in SPH Methods”. *Computers, Materials & Continua* 5(3):173–184 https://doi.org/10.3970/cmc.2007.005.173.

Delft3D. “Delft3D Wepbage”. https://oss.deltares.nl/web/delft3d. Accessed 5th Abril 2023.

Domínguez, J., G. Fourtakas, C. Altomare, R. Canelas and et al.. 2022. “DualSPHysics: from fluid dynamics to multiphysics problems”. *Computational Particle Mechanics* 9(5):867–895 https://doi.org/10.1007/s40571-021-00404-2.

EEMS. “EE Modeling System Wepbage”. https://www.eemodelingsystem.com. Accessed 5th Abril 2023.

English, A., J. Domínguez, R. Vacondio, A. Crespo, P. Stansby, S. Lind, *et al*. 2021, 04. “Modified dynamic boundary conditions (mDBC) for general-purpose smoothed particle hydrodynamics (SPH): application to tank sloshing, dam break and fish pass problems”. *Computational Particle Mechanics* 9 https://doi.org/10.1007/s40571-021-00403-3.

European Comission 2023. “European Commission water related directives”. https://ec.europa.eu/environment/water/index_en.htm. Accessed on January 2023.

Fernández Cara, E. 2005. “A review of basic theoretical results concerning the Navier-Stokes and other similar equations.”. *Boletín de la Sociedad Española de Matemática Aplicada* (32):45–73.

Fourtakas, G., R. Vacondio, J. Dominguez Alonso, and B. Rogers. 2020, January. “Improved density diffusion term for long duration wave propagation”. 351–357.

Gingold, R. A. and J. J. Monaghan. 1977, 12. “Smoothed particle hydrodynamics: theory and application to non-spherical stars”. *Monthly Notices of the Royal Astronomical Society* 181(3):375–389 https://doi.org/10.1093/mnras/181.3.375.

Gómez-Gesteira, M., B. D. Rogers, A. J. C. Crespo, R. A. Dalrymple, M. Narayanaswamy and J. M. Dominguez. 2012. “SPHysics - development of a free-surface fluid solver - Part 1: Theory and formulations”. *Comput. Geosci.* 48:289–299.

Lind, S., R. Xu, P. Stansby, and B. Rogers. 2012. “Incompressible smoothed particle hydrodynamics for free-surface flows: A generalised diffusion-based algorithm for stability and validations for impulsive flows and propagating waves”. *Journal of Computational Physics* 231(4):1499–1523 https://doi.org/10.1016/j.jcp.2011.10.027.

Liu, M. and G. Liu. 2006. “Restoring particle consistency in smoothed particle hydrodynamics”. *Applied Numerical Mathematics* 56(1):19–36 https://doi.org/10.1016/j.apnum.2005.02.012.

Liu, M. and G. Liu. 2010, 03. “Smoothed Particle Hydrodynamics (SPH): an Overview and Recent Developments”. *Archives of Computational Methods in Engineering* 17:25–76 https://doi.org/10.1007/s11831-010-9040-7.

Molteni, D. and A. Colagrossi. 2009. “A simple procedure to improve the pressure evaluation in hydrodynamic context using the SPH”. *Computer Physics Communications* 180(6):861–872 https://doi.org/10.1016/j.cpc.2008.12.004.

Monaghan, J. J. and A. Kos. 1999. “Solitary Waves on a Cretan Beach”. *Journal of Waterway, Port, Coastal, and Ocean Engineering* 125(3):145–155 https://doi.org/10.1061/(ASCE)0733-950X(1999)125:3(145).

Multiphysics®, C. 1998. “Introduction to COMSOL multiphysics®”. *COMSOL Multiphysics, Burlington, MA, accessed Feb* 9:2018.

Pilloton, C., A. Bardazzi, A. Colagrossi, and S. Marrone. 2022. “SPH method for long-time simulations of sloshing flows in LNG tanks”. *European Journal of Mechanics - B/Fluids* 93:65–92 https://doi.org/https://doi.org/10.1016/j.euromechflu.2022.01.002.

Reynolds, O. 1895. “IV. On the dynamical theory of incompressible viscous fluids and the determination of the criterion”. *Philosophical Transactions of the Royal Society of London. (A.)* 186:123–164 https://doi.org/10.1098/rsta.1895.0004.

Risco-Martín, J. L., S. Esteban, J. Chacón, G. Carazo-Barbero, E. Besada-Portas and J. A. López-Orozco. 2023. “Simulation-driven engineering for the management of harmful algal and cyanobacterial blooms”. *SIMULATION* 99(10):1041–1055 https://doi.org/https://doi.org/10.1177/00375497231184246.

Tafuni, A., J. Domínguez, R. Vacondio, and A. Crespo. 2018. “A versatile algorithm for the treatment of open boundary conditions in Smoothed particle hydrodynamics GPU models”. *Computer Methods in Applied Mechanics and Engineering* 342:604–624 https://doi.org/10.1016/j.cma.2018.08.004.

United States Environmental Protection Agency 2023. “Drinking Water Requirements for States and Public Water Systems”. https://www.epa.gov/dwreginfo/drinking-water-regulations. Accessed on January 2023.

UNTRIM. “UNTRIM Wepbage”. https://wiki.baw.de/en/index.php/Mathematical_Model_UNTRIM. Accessed 5th Abril 2023.

Vieira, A., O. García Feal, J. Domínguez, A. Crespo, M. Gesteira and A. English. 2017, 06. “Graphical User Interface for SPH codes: DesignSPHysics”.

Violeau, D. and B. Rogers. 2016, 01. “Smoothed particle hydrodynamics (SPH) for free-surface flows: past, present and future”. *Journal of Hydraulic Research* 54:1–26 https://doi.org/10.1080/00221686.2015.1119209.

Wendland, H. 1995, 12. “Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree.”. 389–396.

Xu, R., P. Stansby, and D. Laurence. 2009. “Accuracy and stability in incompressible SPH (ISPH) based on the projection method and a new approach”. *Journal of Computational Physics* 228(18):6703–6725 https://doi.org/10.1016/j.jcp.2009.05.032.

AUTHOR BIOGRAPHIES

SAMUEL FERRERO-LOSADA is a Ph.D. student at the University Complutense of Madrid (UCM). He holds a Master degree in Astrophysics from the same University. His research interests lie in modeling and simulation. His email address is saferr03@ucm.es.

JOSÉ L. RISCO-MARTÍN received his Ph.D. from UCM, where he currently is Full Professor in the Department of Computer Architecture and Automation. His research interests include systems modeling, simulation, and optimization. His email address is jlrisco@ucm.es.

JOSÉ A. LÓPEZ-OROZCO is a Full Professor in the UCM. He holds a Ph.D. in Physics from the same University. His research interests include multisensor data fusion, control and planning of unmanned vehicles, and robotics. His email address is jalo@ucm.es.

EVA BESADA-PORTAS is an Associate Professor of Systems Engineering and Automation at UCM. She also holds a PhD in Computer Systems from UCM. Her research interests include uncertainty modeling and simulation, optimal control and planning of unmanned vehicles. Her email address is ebesada@ucm.es.