SIMULATION OF CYANOBACTERIA BEHAVIOR IN WATER BODIES WITH CELL-DEVS AND LATTICE-BOLTZMANN METHODS

Samuel Ferrero-Losada[a], Román Cárdenas[b], José A. López-Orozco[a], and José L. Risco-Martín[a]

[a]Dept. of Computer Architecture and Automation, Universidad Complutense de Madrid, Madrid, Spain
[b]Integrated Systems Laboratory, Universidad Politécnica de Madrid, Spain

ABSTRACT

Cyanobacteria are essential to aquatic ecosystems, but their excessive growth can result in harmful blooms, causing significant environmental and health risks. This paper presents a comprehensive modeling approach using Cell-DEVS to simulate the behavior of cyanobacteria in aquatic environments. The model integrates various factors influencing cyanobacteria dynamics, including vertical migration, growth, and decay, as well as environmental conditions such as solar irradiance and nutrient availability. Using a Lattice-Boltzmann method for fluid simulation, the model represents the transport mechanisms within the water column. The simulation results indicate the model's potential to predict cyanobacteria distribution evolution and behavior under different environmental scenarios. This work contributes to the development of early warning systems for harmful cyanobacteria blooms, presenting a potentially valuable tool for water monitoring and management. It also highlights the importance of integrating multiple environmental factors in modeling efforts to enhance the predictive accuracy of cyanobacteria dynamics.

Keywords: Cyanobacteria, Cell-DEVS, Lattice-Boltzmann Method, Aquatic Ecosystems, Environmental Simulation.

1 CONTEXT AND MOTIVATION

Global warming has generated numerous challenges that threaten the ecological stability of ecosystems, posing risks to both local wildlife and human health. Among these challenges, the proliferation of cyanobacteria has become a significant concern subject within the research community. These organisms thrive in warmer water temperatures and environments enriched by human activities, leading to increasingly frequent harmful cyanobacterial blooms. These blooms release toxins into the aquatic systems on death, posing environmental and health risks [1].

To address this threat, governments worldwide have initiated efforts to monitor and study water quality, with a particular focus on drinking water sources [2, 3]. However, practical studies of these environments face challenges due to both general and cyanobacteria-specific conditions. Issues related to data acquisition, quality, quantity, and process automation make modeling and simulation valuable tools for developing intelligent and agile solutions. The integration of system design, automation, and control algorithms [4] with environmental biophysical models [5] can lead to complex and adaptable management systems [6].

There are numerous methodologies and software tools available that can fulfill many roles required by such systems. For instance, several multi-physics fluid simulation programs (both commercial [7, 8] and open-source [9]) can handle the environmental aspects of the system. However, they often require numerous, sometimes obscure, input parameters, which may not be relevant for the intended application. Furthermore,

Proc. of the 2025 Annual Modeling and Simulation Conference (ANNSIM'25)
May 26-29, 2025, Universidad Complutense de Madrid, Madrid, Spain
J.L. Risco-Martín, G. Rabadi, D. Cetinkaya, R. Cárdenas, S. Ferrero-Losada, and A. Bany Abdelnabi, eds.

embedding the simulation within an early warning system (DEVS-BLOOM, [6]) based on the Discrete Event System Specification (DEVS) [10] formalism can be challenging due to the monolithic nature of commercial software, which limits flexibility and adaptability.

Instead, the modularity of DEVS naturally aligns with layered architectures. By selecting a combination of specific, small, and independent open-source simulators or custom models, we can maintain detailed control over each element, adapting them to specific needs. This approach allows for modifications, deactivations, or replacements of models without significantly affecting the overall simulation. Additionally, models can be logically separated within the larger system based on computational requirements. For example, computationally intensive simulations (e.g., hydrodynamics) can be moved to a cloud layer, while lighter models (e.g., biological processes) can run on local, less powerful hardware such as mobile probes.

With these considerations in mind, Cell-DEVS is a suitable choice for the environmental simulation component of our proposed early warning system. Cell-DEVS, an extension of the DEVS formalism, is designed to model complex systems using cellular automata. It enhances traditional cellular automata by incorporating timing delays, enabling more accurate and efficient simulations of spatially distributed systems. This approach is particularly useful for modeling phenomena where local interactions and time-dependent behaviors are critical, such as biological and environmental processes [11, 12]. When searching the literature, no cyanobacteria-related discrete event simulation proposals have been found. Regarding water management the main discrete event simulation topics tend to revolve around policy planning and assesing their potential effect on the environment [13]. Agent and physics-based models [14, 15], as well as cellular automata simulations [16] examples can already be found. However, none have been found to give a comprehensive view of the spatial cyanobacteria distribution and lake state, instead focusing on general behavioral patterns or isolating the physical spatial problem from the biological cycle of the organisms of study.

In this paper, we use Cell-DEVS to simulate the behavior of cyanobacteria in aquatic environments. The choice of Cell-DEVS is justified by its ability to integrate multiple environmental factors and capture dynamic interactions. Using Cell-DEVS, the model can effectively represent the spatial distribution and temporal evolution of cyanobacteria populations, considering factors like nutrient availability, solar irradiance, and fluid dynamics. This capability may allow us to build predictive models that can inform early warning systems of harmful cyanobacteria blooms, ultimately aiding in water quality monitoring and management efforts.

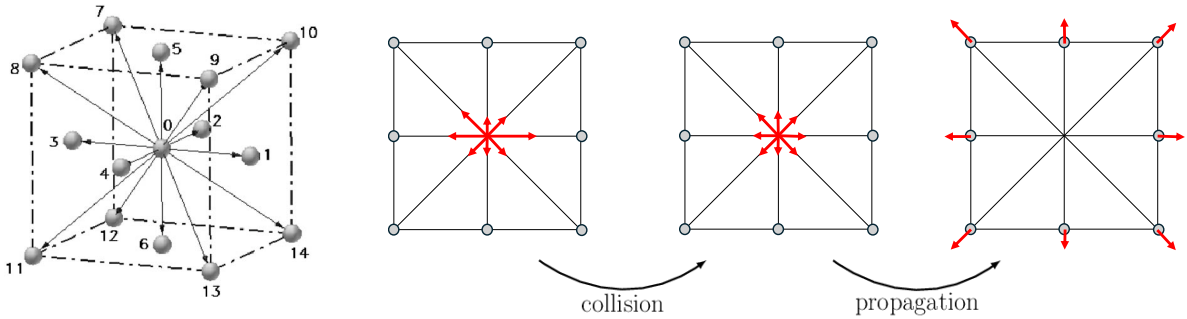
The remainder of the paper is structured as follows. Section 2 explains the formulation of each model involved in the simulation and their operation order in the Cell-DEVS cellular automata. Section 3 describes the results of a proof-of-concept study case where all model behaviors are demonstrated under realistic conditions. Finally, Section 4 presents the conclusions and potential future research directions.

2 MODELS FORMULATION AND CELL-DEVS SCHEME

In this section, we describe the fluid simulation method being used, the models present in the Cell-DEVS computations and the operations of the cellular automata during execution. These models simulate specific behaviors of cyanobacteria observed in real life. One model calculates the vertical migration of cyanobacteria in the water column, influenced by local temperature and sunlight conditions. Another simulates the growth and decay of the cyanobacteria population, based on the availability of nutrients and sunlight. The functions used to model the evolution of sunlight over time are also explained.

2.1 Fluid simulation

The flow data used as the basis for the transportation mechanism is obtained from a Lattice Boltzmann Method (LBM) simulation [17]. In this independent simulation, the composite model (method 4) proposed



(a) D3Q15 lattice model.

(b) Lattice Boltzmann method advance scheme.

Figure 1: D3Q15 structure and simulation scheme used by Lattice Boltzmann methods.

by J.M. Buick and C.A. Greated [18] is employed to incorporate the effect of gravity on the fluid. The Zou-He scheme [19] is utilized to enforce inlet and outlet velocity boundary conditions, adapted to the D3Q15 lattice structure. As shown in Figure 1a, D3Q15 refers to a three-dimensional lattice where each node is connected to 14 specific neighbors. The "Q15" notation indicates the total number of node connections, including the central node itself (neighbor 0).

This fluid simulation method operates by propagating virtual particles across the lattice nodes. When multiple particles reach a node simultaneously, a collision operation is performed to redistribute each particle into one of the 15 propagation directions with an appropriate speed. This redistribution is achieved by comparing the current particle speed distribution with a predefined equilibrium distribution function, which may include information about external forces like gravity or specific scenario conditions affecting fluid behavior. By alternating between propagation and collision operations, the fluid dynamics are simulated.

The simulation generates a high-resolution three-dimensional velocity field with a velocity value per D3Q15-mesh node. To derive the flow data between different cellular automata, the velocity components are integrated along the cell walls. This is accomplished by averaging the X, Y, and Z velocity components of the LBM lattice nodes that align with the cell interfaces and projecting them onto the normal vector of each wall. This way, the fluid simulation is run independently from the cellular automata counterpart. Then, the resulting data is integrated in the form of dynamic flows in cell-DEVs cell's interfaces.

2.2 Solar irradiance evolution and absorption functions

The behavior of cyanobacteria in this study is highly influenced by solar irradiance, which in turn is modified by water absorption relative to depth. Therefore, it is crucial to accurately model the evolution of sunlight over time and depth. Equation (1) calculates the solar irradiance at the surface level across the entire scenario. This equation incorporates time parameters such as the sunrise time, the initial simulation time, t_0, and the duration of daylight T_s, all expressed in seconds within the sine function. The parameter K_s represents the maximum possible irradiance, while I_0 denotes the minimal irradiance during nighttime.

$$I_{surf}(t) = I_o + max\left[0, K_s \cdot sin\left(\frac{\pi}{T_s}(t + t_0 - sunrise)\right)\right], \quad K_s = 800\ W/m^2; \quad I_0 = 10\ W/m^2; \quad T_s = 12\ h. \tag{1}$$

To determine the irradiance values throughout the rest of the domain, the Beer-Lambert law of absorption [20] is applied to the surface irradiance (2). In this equation, K_{abs} is the absorption constant, d_z is the cell size

along the Z-axis, N_z is the number of XY-layers, and z is the Z-axis coordinate of the cell being evaluated. The Z-axis is upwards and vertical, with its origin at the bottom of the scenario.

$$I^z(t) = I_{surf}(t) \cdot exp\left(-K_{abs} \cdot d_z(N_z - z)\right); \qquad K_{abs} = 0.3\ m^{-1}. \tag{2}$$

2.3 Vertical Migration model

A critical aspect of cyanobacteria behavior in aquatic ecosystems is their ability to migrate vertically in response to environmental factors such as light and temperature. This migration is driven by the organisms' capacity to perform photosynthesis and respiration, which are influenced by light availability. Under optimal light and nutrient conditions, cyanobacteria move upwards to maximize exposure to sunlight for photosynthesis. Conversely, in low-light conditions, they tend to sink and accumulate near the thermocline, a layer characterized by a significant temperature gradient.

To simulate this behavior, the model calculates the rate at which cyanobacteria move vertically based on local temperature and irradiance values. The simulation determines the fraction of cyanobacteria mass that should be transferred to neighboring cells above or below. The upward and downward migration rates, denoted as O^z_{up} and O^z_{down} are computed using (3), where I is the solar irradiance, C is the cyanobacteria concentration, and T is the local temperature. Here, $\Delta\#^{z\pm1} = \#^{z\pm1} - \#^z$, with # representing any variable and z the cell's vertical coordinate. α, β and γ are dimensionless parameters.

$$\begin{aligned} O^z_{up} &= \alpha_u \cdot log\Delta I^{z+1} - \beta_u \cdot max(0, \Delta C^{z+1}); \qquad \alpha_u = 0.16,\ \beta_u = 0.01, \\ O^z_{down} &= \alpha_d - \beta_d \cdot max(0, \Delta C^{z-1}) + \gamma_d \cdot \Delta T^{z-1}; \qquad \alpha_d = 0.2,\ \beta_d = 0.004,\ \gamma_d = 10. \end{aligned} \tag{3}$$

The logarithmic term for upward migration in (3) accounts for the exponential decay of irradiance due to absorption (2), resulting in smoother migratory flows that vary linearly with depth. To prevent excessive concentrations at the surface, a resistance term based on the concentration gradient is included. This ensures a more realistic distribution, with maximum concentrations near the surface.

For downward migration, the constant term is predominant. Similar to upward migration, a concentration gradient resistance term is included to broaden the equilibrium distribution. The temperature gradient term acts as a barrier, preventing further descent past the thermocline. This term, although positive in the equation, is typically negative due to the temperature decrease with depth, causing cyanobacteria to accumulate above the thermocline.

These rates are constrained between 0 and 1 using the first expression in (4). The second expression determines the fraction of cyanobacteria mass that migrates to each neighboring cell. Here, V^z is the cell volume, and D is the maximum fraction of mass that can migrate. In this study, the limit is set to 25% for both upward and downward migration as a preliminary value for simplicity.

$$O^z_x = max(0, min(O^z_x, 1)); \quad m^z_x = O^z_x \cdot C^z \cdot V^z \cdot D \quad \forall x \in [up, down]. \tag{4}$$

2.4 Nutrient-Cyanobacteria population model

Cyanobacteria are the primary focus of this simulation, necessitating a model that computes their population dynamics in relation to local environmental conditions. This is achieved using a modified Lotka–Volterra predator-prey model [21], where cyanobacteria interact with local nutrient levels. The model equations (5) retain most of the original terms as constants. However, the interaction term for cyanobacteria growth has

been adjusted to include a dependency on solar irradiance, which enhances reproduction in the presence of nutrients, N. This enhancement is represented by the parameter c_I, constrained between 0 and 1. Solar irradiance can potentially double the reproduction rate of cyanobacteria at an optimal irradiance level, I_{opt}. For irradiance values above this level, the parameter decreases smoothly, while for lower values, it decreases more rapidly, reaching zero at no irradiance.

$$\begin{aligned} &\frac{dN}{dt} = A_N \cdot N - B_N \cdot C \cdot N, && \frac{dC}{dt} = -A_C \cdot C + B_C \cdot C \cdot N \cdot (1 + c_I(I)), \\ &A_N = 5 \cdot 10^{-3},\ B_N = 5 \cdot 10^{-6}, && c_I(I) = \frac{e}{I_{opt}} I \cdot exp\left(-I/I_{opt}\right), \quad I_{opt} = 150\, W/m^2, \\ & && A_C = 1.9 \cdot 10^{-3},\ B_C = 2.5 \cdot 10^{-5}. \end{aligned} \tag{5}$$

This model introduces a time and space-dependent parameter, complicating the task of parameter adjustment to achieve a desired equilibrium state. Given the significant differences in environmental conditions between day and night, the equilibrium must be robust to these variations. For practical purposes, the proof-of-concept case in Section 3 uses a preliminary parameter configuration that allows for easy differentiation of each model's effects while maintaining overall population stability. Further refinement of these parameters is necessary for future work.

2.5 Cellular automata definition and operation scheme

Cell-DEVS follows the general functional structure of classical cellular automaton formulation, where a finite grid of cells updates each cell's internal state taking into account their neighbor's states. But it also takes this formalism a step further, allowing differences in cell characteristics (e.g., size and shape) and asymmetric neighbor relations (e.g., interface permeability, one-way flows, ...), advantages our simulation will exploit as explained in this section. Since we aim to build a somewhat irregular mesh of cells with special neighboring conditions, the simulation initialization process must also be explained [11].

In Cell-DEVS, each cell is represented as an atomic model, which is a fundamental building block in the DEVS formalism [10]. The atomic model defines the behavior of a cell by specifying its state variables, input and output ports, and the functions that govern state transitions. The key components of an atomic model in Cell-DEVS include: (i) *state variables*, represent the internal state of the cell, such as concentrations of cyanobacteria and nutrients, temperature, and sun irradiance; (ii) *input ports*, used to receive information from neighboring cells, like the concentration of cyanobacteria from its neighbors; (iii) *output ports*, used to send information to neighboring cells, like the updated concentration values, to its neighbors; (iv) *internal transition function*, which defines how the cell's state changes over time in the absence of external events; (v) *external transition function*, which defines how the cell's state changes in response to inputs from neighboring cells, such as changes in concentration due to migration; (vi) *output function*, which determines what information is sent to neighboring cells after a state transition; and (vii) *time advance function*, which specifies the time until the next internal transition, allowing the model to simulate asynchronous events efficiently. Each cell is defined as an atomic model. As a result, Cell-DEVS can simulate complex interactions within the cellular automata framework, capturing both local and global dynamics of the system. Formally, the cell-DEVS atomic models that compose the cellular automata are defined as follows:

$$\text{CA} = < \text{X, Y, S, NE, } \tau\text{, d, delay, config} >,$$

where:

- X = Y = S: (C, N, T, I, m_{up}, m_{down}) refer to the potential input, output and state data sets, which in our case coincide.They always contain: cyanobacteria and nutrient concentration (C,N), temperature (T), sun irradiance (I), upwards migration mass (m_{up}) and downwards migration mass (m_{down}).
- NE: ($inout$, raw_flow) is the neighboring cell ID set. Each element is accompanied by a $inout = \pm 1$ value and the current flow data, which is periodically updated. $inout$ is used to identify if positive raw_flow values correspond to inflow ($inout = 1$) or outflow ($inout = -1$) situations, since flow data is expressed in the general space coordinates.
- $\tau : S \times N \times X \rightarrow S$ is the local computation function of the cell. When triggered, it computes the new state of the cell from its previous state, the neighborhood set, and inputs.
- d is the delay function. Currently, it is the same for all cells, corresponding to an effective time step of the simulation.
- $delay$ is the delay type function, which in this case is always set to "inertial". More information in [11].
- $config$ contains different cell parameters such as cell volume (V) and initial day hour (t_0).

The scenario is defined as a theoretical unwrapped three-dimensional matrix of cells, which automatically enforces closed boundary conditions around the simulation domain. Next, the neighborhood of each cell is created. By default, a von Neumann first-order neighborhood is applied. However, this scheme must be altered for specific cells so that their neighbor sets match that of the desired geometry. This way, a matrix of 8x4x10 cells may be used as a 32x10 heterogeneous mesh. Note however, that Cell-DEVS still uses the theoretical matrix cell indexes internally. During this initial process, cell configuration and initial state parameters are also established.

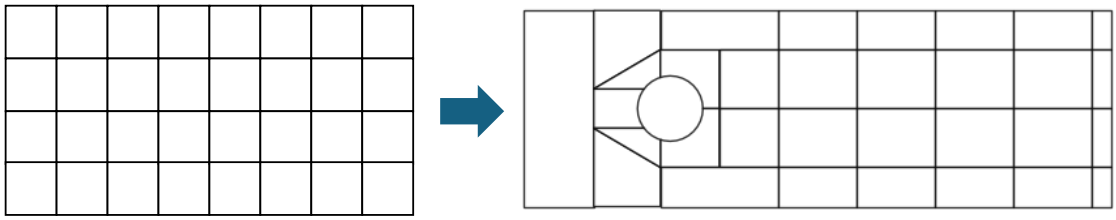


Figure 2: A computational 8x4x10 homogeneous cell matrix is used as a 32x10 inhomogeneous mesh.

Once all the domain and neighbor connections are stablished, the simulation begins. Each cell's internal transition function acts as a timer, allowing time to pass without updating any state values. When inputs from neighboring cells are received, the external transition function is triggered, which calls the local computation function, τ, initiating the operations of each model in a specific sequence. First, the influence of neighboring cells is considered by incorporating the downward (or upward) migration from the cell above (or below), if applicable. Additionally, the total cell concentration is adjusted based on the direction of the flow, f, at each neighbor interface, as shown in (6). These operations are performed on the mass values of cyanobacteria and nutrients, which are then converted back to concentrations using the cell volume, V. Here, Δt represents the output delay of the cell DEVS module, which is consistent across all cells and serves as the effective time step of the simulation.

$$C_{new} = C_{old} + \frac{\Delta t}{V}\left(\sum_{f>0} C_{neighbor} \cdot f^{neighbor} + \sum_{f<0} C_{old} \cdot f^{neighbor}\right), \qquad \text{same for } N. \tag{6}$$

Subsequently, the growth model is applied, updating both concentrations as described. The vertical migration model is then executed, producing new upward and downward migration mass values based on the output of the growth model. These new migration values are subtracted from the concentration value of the cell in the same manner they were added by equation (6). Finally, the sun irradiance model updates the local irradiance value. With all state values updated, the cell waits for the Δt before sending its state to all neighbors, thus restarting the cycle.

The reader can find a repository with the code files of the cell-DEVS and LBM fluid simulations in https://github.com/iscar-ucm/LBM-Cyano-Chanel. Note, however, that xDEVS is needed to run the cellular automata part.

3 CASE OF STUDY

Now we will show how all the explained models behave together through a simple proof-of-concept case. It consists of a 50 by 50 meter square section channel, with a length (X) of 150 meters, and a cylindrical obstacle across the vertical axis (Z). This column is centered at X = 37.5 meters and in the middle of the Y axis, with a diameter of one third of this axis extent. We have selected this scenario because it has a very well-known resulting flow field and a simple geometry in which to test every major aspect of our models.

The flow is simulated in python using the Lattice-Boltzmann method, with solid walls on all sides except X=0 and X=150. At these locations an inlet speed of 0.1 m/s and an open boundary outlet are defined respectively. As the interesting flow profile for this scenario is generated when the resulting Reynolds number is low enough, we have selected a relaxation time of 0.506, and a dynamic viscosity of $4{\cdot}10^{-3}$. The result is presented in Figure 3, where an Y-axis oscillating tail behind the column appears due to the chain of vortexes generated by the obstacle. It's worth noting, that the lateral solid bounds make the flow at the second half of the tube length be very small near them, which will be important for the cyanobacteria transportation and accumulation.

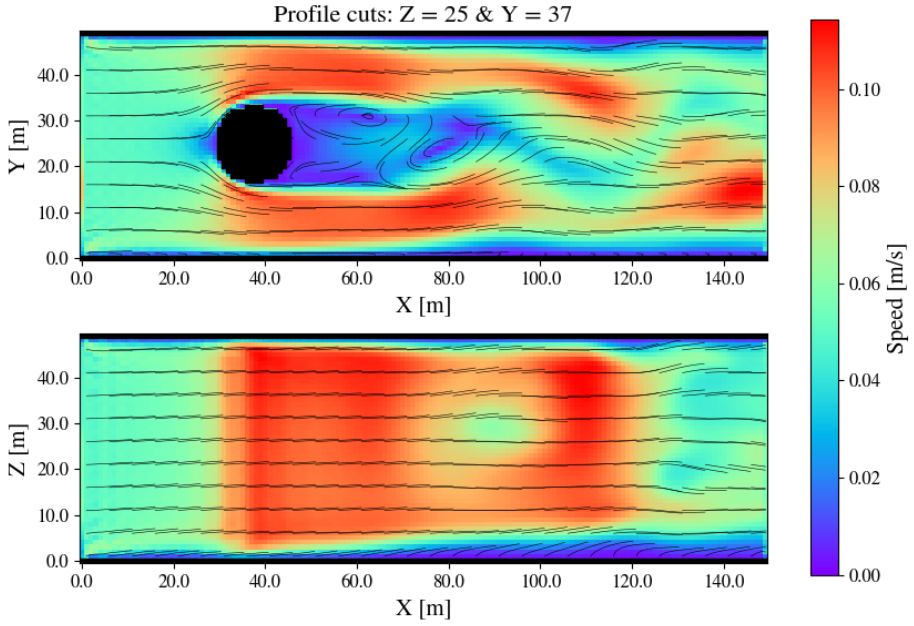


Figure 3: Horizontal and vertical stationary flow profiles.

To use this velocity field in Cell-DEVS, the cell's walls must be defined so that fluxes (in m^3/s) may be calculated for each of them. As explained, Cell-DEVS allows for the definition of heterogeneous cell ge-

ometries and distributions. We have taken advantage of this feature to create the mesh in Figure 4. This division generates 32 cells in the horizontal plane, which are then homogeneously divided in the vertical axis in 10 identical 5 meter-high planes. Consequently, the mesh will contain cells with 4,5 or 6 active walls (1 or 2 along the Z axis + 3 or 4 depending on the Figure 4 cell shape). Understanding by active walls those that separate defined cells from each other.

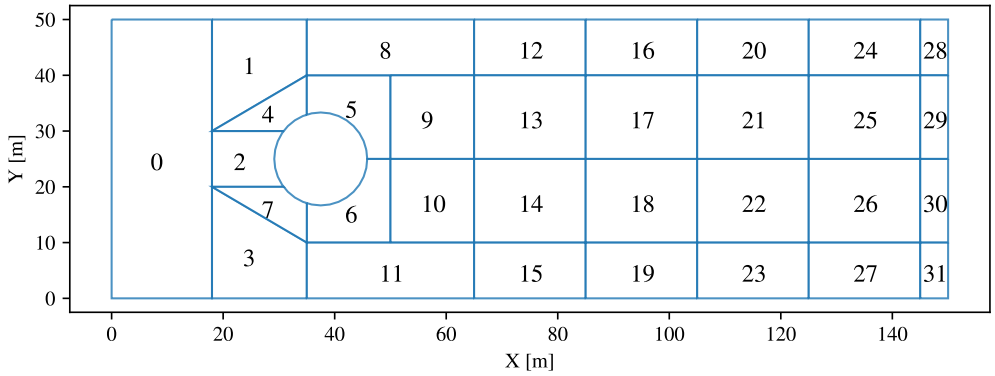


Figure 4: Horizontal sector division of the Cell-DEVS simulation domain in 32 cells.

With this mesh set, the initial and boundary conditions for the biological models remain to be defined. As we aim to show that each model works as intended, we have assigned an special behavior for cell 0, isolating the biological models from flux transportation. This cell works as a bioreactor, generating dynamic vertical concentration profiles through the vertical migration and growth models. These concentrations, then, are transported to the forward cells without altering the values of the initial cell. This way the starting cell can not empty itself of cyanobacteria unless the predator-prey equilibrium breaks, leading to the extinction of the organism's population. In every other cell, all models coexist and evolve as one.

Regarding the different physical quantities involved in the models calculations, temperature will be set as a constant field of 10ºC for the three deepest XY-layers of cells and 15ºC for the rest. This configuration establishes a hard boundary for the vertical migration model emulating the real life thermocline, over which cyanobacteria tend to accumulate, avoiding deeper lake regions. Initial nutrient and cyanobacteria concentrations are set respectively to 80 and 90 in all cell 0 layers above the thermocline. Although this distribution is not realistic, this distribution is chosen for its simplicity and the easiness of the models to quickly find an equilibrium for the whole column. This way, the simulation only needs a relatively small warm up time before more dynamic events happen.

As this is a proof-of-concept scenario to present the models and methodology, the result analysis in this section will focus in those cases where each model's behavior is easily distinguishable from each other. These are the dawn and dusk hours, as the vertical migration pattern of cyanobacteria is greatly dependent on sun radiation. Provided that both cases show the same behavioral aspects, for conciseness, only examples of the dawn hours will be shown. Consequently, the simulation is set to start at 5:30 AM and compute the following 7 hours of the day. Sunrise starts at 6:30 AM, with a 12-hour long daylight cycle. The Cell-DEVS simulation time step is 10 s, established based on the flow data update period, which is 50 s. These values are set so that an adequate time resolution is employed to simulate each phenomenom. In this case, since the flow is stable with slow oscillations, it is possible to update the flow data less often. On the other hand, since local concentration depends on more volatile values (such as concentration itself), a smaller update period translates into more accurate results. Note that different model parameter configurations may allow for a different range of suitable step sizes.

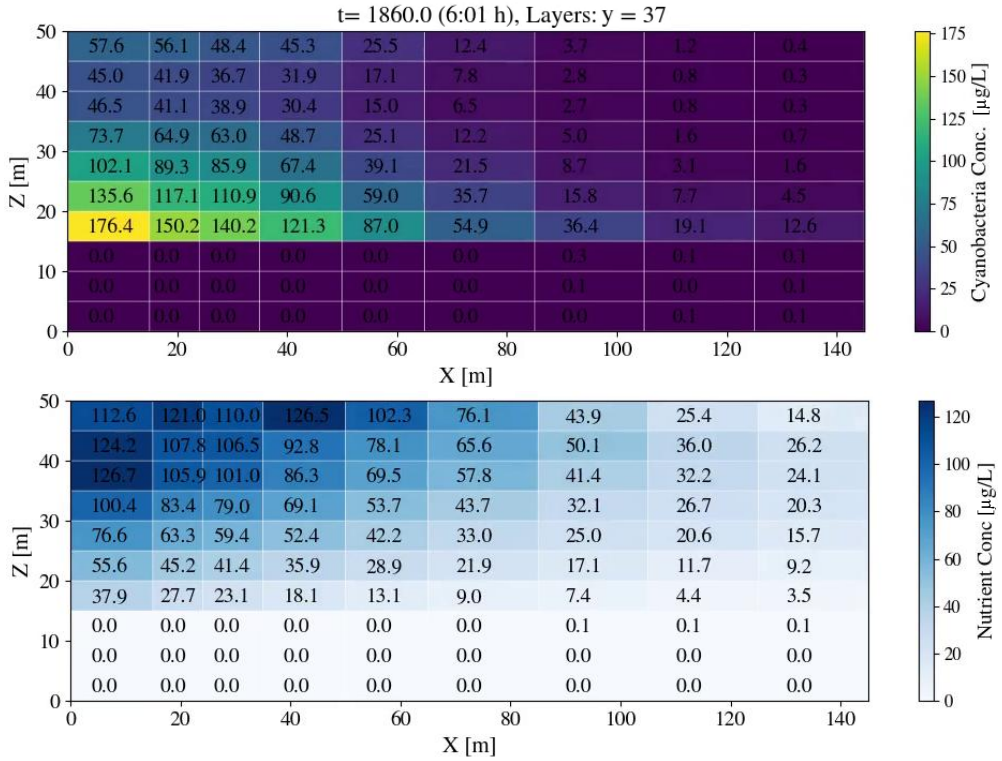


Figure 5: Vertical cyanobacteria concentration and nutrient distribution at night time.

As shown in Figure 5, the initial conditions rapidly evolve towards the expected night-time state where most of the cyanobacteria accumulates above the thermocline. However, this accumulation is not complete due to the presence of nutrients in shallower layers and the concentration-dependent term in the vertical migration model. In parallel, nutrient distribution evolves as expected resulting in the opposite distribution of that of the cyanobacteria. Note also that this relation is kept throughout the simulation domain, with the exception of certain cells, where the flow alters it accumulating both nutrients and organisms, generating local sources of colony growth. Due to the flow profile this zones tend to be close to the Z^+ channel wall.

Forward in simulation time, as the day starts to dawn, cyanobacteria start to migrate upwards, accumulating at the more superficial layers to the point of turning upside down the initial distribution. As seen in Figure 6, this transitory period lasts about an hour. Additionally, observing the evolution of the numerical values, the reader can observe an overall increase in concentrations throughout the column. This occurs due to the deeper upwards-migrating cyanobacteria entering sectors with higher concentration of nutrients. This high nutrient and cyanobacteria concentrations, combined with the increasing radiation of the sun, produce a very big growth rate until the nutrients fall off. Following this proliferation, as expected in this kind of growth models, cyanobacteria concentration strongly decreases due to the lack of nutrients to sustain such population. However, while the explained migratory changes correspond in form and broad timescale to the real life observations, the general population oscillations around an equilibrium point do not. This is due to the broadly adjusted parameter configuration of the growth model, whose optimization is out of the scope of this study. Taking this into account, the model works as expected, having oscillating populations around a stable equilibrium point.

Regarding the horizontal concentration evolution, a stronger dependence on the flows is expected, as there is bigger differences and more variation in fluid speeds along the Y axis than within the fluid column (see Figure 3). The alternating fast and slow flows behind the obstacle generate a similar pattern in the concen-

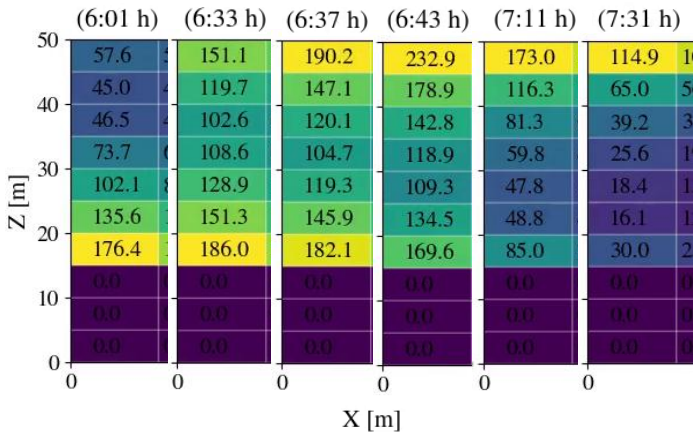


Figure 6: Vertical migration of sector 0 column cyanobacteria population during the first hour of daylight.

tration field. In Figure 7 two opposite states of the flow-induced cycle are shown. As it happens with the flow, the pattern is clearer the further it moves from the obstacle. Note that each color map uses a different scale to show with suitable contrast the explained features. This change in concentration ranges is produced by the migration and growth models.

While this is happening mainly in the internal cells of the XY plane, the external sectors maintain higher and more stable concentration values, as their flow field is calmer than their counterparts. Figure 7 numerical values also show a slow decay in concentration in this regions, provoked by the decreasing nutrient abundance as the cyanobacteria advance through the channel. Also, bear in mind that both prey and predator are transported horizontally by the flow in the same manner, but not vertically. Since these data corresponds to the first hours of the day, cyanobacteria migrates to upper layers when possible. As the concentrations are transported forward they find new regions with less cyanobacteria concentration in which to migrate upwards. As it is defined right now, vertical migration is faster the closer to the surface. This makes mid depth layers (such as the one shown) slowly lose concentration downstream, since the amount of mass coming from the depths is smaller than the one leaving towards the surface.

4 CONCLUSIONS AND FUTURE WORK

This study successfully demonstrates the feasibility of integrating multiple biological behavior models into a Cell-DEVS simulation. This approach allows for the flexible definition of three-dimensional sectors within the scenario and the implementation of flow-dependent transport processes between them, all while maitaining the efficiency of discrete event simulation. This combination of features makes it a promising framework for simulating geometrically complex environments, such as small lakes and reservoirs.

The results indicate that simple models can effectively emulate key events in cyanobacterial development, such as dawn and dusk migration in thermally stratified environments. However, these models may not provide the level of detail needed to predict extreme events like harmful cyanobacterial blooms. Each model, while implemented independently, interacts with and is influenced by cyanobacteria concentration. This interdependence highlights the importance of parameter adjustment, as each model must be robust enough to handle the constant imbalances caused by other environmental mechanisms, which may be competing or even opposing.

This methodology is a viable option for studying cyanobacteria behavior in the context of water quality management. The architecture's flexibility allows users to add new behaviors or replace existing ones with

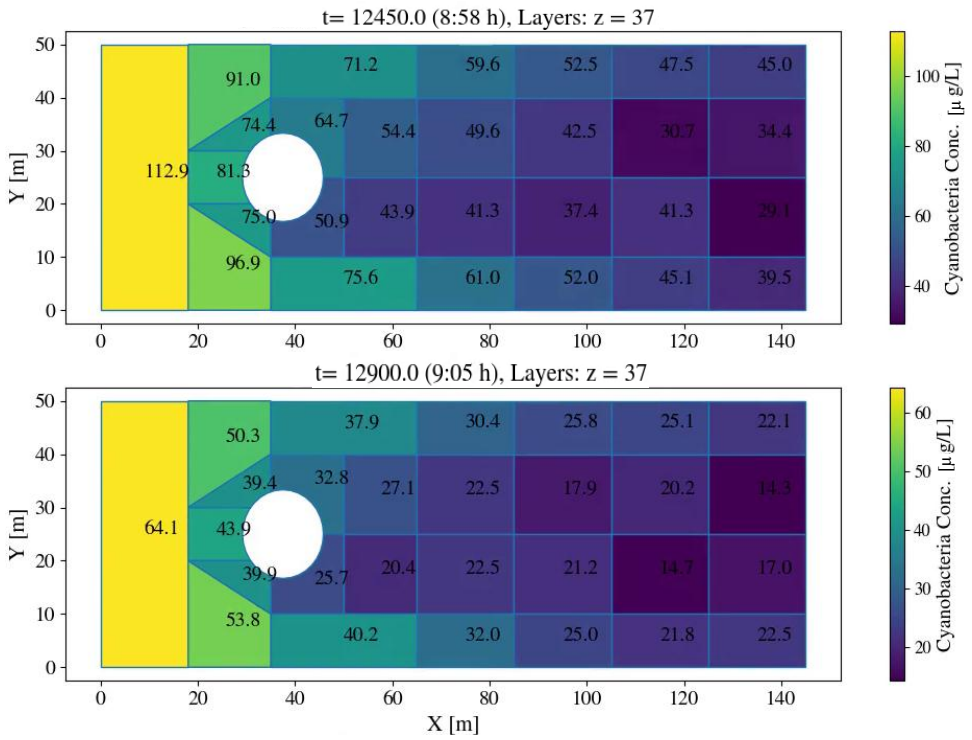


Figure 7: XY-layer (Z = 37) view of flow induced concentration oscillations due to the obstacle's presence.

more complex models. While the simulation's hydrodynamic component is handled outside of Cell-DEVS, there is ample opportunity to explore the various mechanisms that affect cyanobacteria.

Several potential avenues for future work are available, depending on the aspect of the system to be enhanced. Regarding the Cell-DEVS specification, a more event-driven time advance scheme could be developed. Currently, the system relies solely on cellular external transitions to modify cell states. Moving the growth model to the internal function could be a worthwhile exploration. Additionally, since the current behavior includes two distinct stationary global states for day and night, implementing two different cell configurations, each with its own equilibrium state, could simplify parameter adjustment and facilitate the exploration of more optimized configurations.

In terms of the implemented models, there are clear areas for improvement:

- With the methodology established and the basics tested, it is time to apply the model to real lake geographies, where more comprehensive scenarios may arise. With these changes, finer meshes should also be tested, given the current low resource demand of the simulation.
- Since temperature significantly influences biological processes, thermal dependencies should be further developed, and the thermocline effect softened to allow subtle population interactions in the surrounding region. In the future, the thermocline may have its own evolutionary model related to the global thermodynamic conditions of the scenario.
- The early warning system considers various dissolved substances in water, such as oxygen and nitrates, so the nutrient variable should be replaced with these. This change will necessitate nutrient-specific production models, which may increase the importance of deeper layers down to the lake bed.

- As discussed in Section 3, model parameters must be further optimized to achieve more robust equilibrium points for each model, especially the growth model. Longer simulations and extended time ranges are needed to adjust the long-term behavior of the cyanobacteria population.

ACKNOWLEDGMENTS

This work has been supported by the Research Projects SMART-BLOOMS (TED2021-130123B-I00) by MCIN/AEI/10.13039/501100011033 and the European Union NextGenerationEU/PRTR, IA-GES-BLOOM-CM (Y2020/TCS-6420) of the Synergic program of the Comunidad Autónoma de Madrid, and INSERTION (PID2021-127648OB-C33) by the Knowledge Generation program of the Spanish Ministry of Science and Innovation.

REFERENCES

[1] H. W. Paerl and T. G. Otten, "Harmful cyanobacterial blooms: Causes, consequences, and controls," *Microbial Ecology*, vol. 65, pp. 995–1010, 2013. [Online]. Available: https://doi.org/10.1007/s00248-012-0159-y

[2] United States Environmental Protection Agency, "Drinking water requirements for states and public water systems," https://www.epa.gov/dwreginfo/drinking-water-regulations, 2023, accessed on January 2023.

[3] European Comission, "European Commission water related directives," https://ec.europa.eu/environment/water/index_en.htm, 2023, accessed on January 2023.

[4] G. Carazo-Barbero, E. Besada-Portas, J. L. Risco-Martín, and J. A. López-Orozco, "Ea-based asv trajectory planner for detecting cyanobacterial blooms in freshwater," in *Proceedings of the Genetic and Evolutionary Computation Conference*, ser. GECCO '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 1321–1329. [Online]. Available: https://doi.org/10.1145/3583131.3590484

[5] C. Overman and S. Wells, "Modeling cyanobacteria vertical migration," *Water*, vol. 14, p. 953, 03 2022.

[6] J. L. Risco-Martín, S. Esteban, J. Chacón, G. Carazo-Barbero, E. Besada-Portas, and J. A. López-Orozco, "Simulation-driven engineering for the management of harmful algal and cyanobacterial blooms," *SIMULATION*, vol. 99, no. 10, pp. 1041–1055, 2023. [Online]. Available: https://doi.org/10.1177/00375497231184246

[7] C. Multiphysics®, "Introduction to comsol multiphysics®," *COMSOL Multiphysics, Burlington, MA, accessed Feb*, vol. 9, p. 2018, 1998.

[8] EEMS, "EE Modeling System wepbage," https://www.eemodelingsystem.com, accessed 5th Abril 2023.

[9] Delft3D, "Delft3D wepbage," https://oss.deltares.nl/web/delft3d, accessed 5th Abril 2023.

[10] B. Zeigler, A. Muzy, and E. Kofman, *Theory of Modeling and Simulation: Discrete Event & Iterative System Computational Foundations.* Academic Press, 12 2018.

[11] R. Cárdenas and G. Wainer, "Asymmetric cell-devs models with the cadmium simulator," *Simulation Modelling Practice and Theory*, vol. 121, p. 102649, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569190X22001198

[12] R. Cárdenas, C. R. Martin, G. Wainer, P. Dobias, and M. Rempel, "Studying the spread of diseases using geographical data and irregular topologies with cell-devs," in *2021 Annual Modeling and Simulation Conference (ANNSIM)*, 2021, pp. 1–12.

[13] J. L. Romeu, "Design and evaluation of aquatic ecosystems via discrete event simulation," 2004. [Online]. Available: https://api.semanticscholar.org/CorpusID:9544838

[14] S. Shrivastava, S. Barat, S. Kausley, V. Kulkarni, and B. Rai, "River digital twin for water quality prediction," 12 2024, pp. 608–619.

[15]

[16] F. Zhao, S. Zhang, R. Chen, L. Xiao, G. Luan, S. Feng, and Z. Xie, "A modified cyanobacteria prediction model based on cellular automata model using n and p concentration reverse data: a case study in taihu lake," *Environmental Science and Pollution Research*, vol. 29, pp. 34 546–34 557, 2022. [Online]. Available: https://doi.org/10.1007/s11356-022-18612-5

[17] S. Chen and G. D. Doolen, "Lattice boltzmann method for fluid flows," *Annual Review of Fluid Mechanics*, vol. 30, no. 1, pp. 329–364, 1998.

[18] J. M. Buick and C. A. Greated, "Gravity in a lattice boltzmann model," *Phys. Rev. E*, vol. 61, pp. 5307–5320, May 2000.

[19] M. Hecht and J. Harting, "Implementation of on-site velocity boundary conditions for d3q19 lattice boltzmann simulations," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2010, no. 01, p. P01018, jan 2010. [Online]. Available: https://dx.doi.org/10.1088/1742-5468/2010/01/P01018

[20] D. F. Swinehart, "The beer-lambert law," *Journal of Chemical Education*, vol. 39, no. 7, p. 333, 1962. [Online]. Available: https://doi.org/10.1021/ed039p333

[21] N. Bacaër, *Lotka, Volterra and the predator–prey system (1920–1926)*. London: Springer London, 2011, pp. 71–76. [Online]. Available: https://doi.org/10.1007/978-0-85729-115-8_13

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 fluids and aquatic organism behavior modeling and simulation. His email address is saferr03@ucm.es.

ROMAN CÁRDENAS is an Assistant Professor in the Department of Electronic Engineering at Universidad Politécnica de Madrid (UPM), Spain, where he obtained a Ph.D. in Electronic Systems Engineering in Cotutelle with Carleton University (CU). His research interests include modeling and simulation with applications in the IoT domain. His email address is r.cardenas@upm.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.

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.