RESEARCH ARTICLE

NUFEB: A massively parallel simulator for individual-based modelling of microbial communities

Bowen Li[1,2], Denis Taniguchi[1], Jayathilake Pahala Gedara[3¤a], Valentina Gogulancea[3,4], Rebeca Gonzalez-Cabaleiro[3¤b], Jinju Chen[3], Andrew Stephen McGough[1], Irina Dana Ofiteru[3], Thomas P. Curtis[3]*, Paolo Zuliani[1,2]*

1 School of Computing, Newcastle University, Newcastle upon Tyne, United Kingdom, **2** Interdisciplinary Computing and Complex bioSystems (ICOS) Research Group, Newcastle University, Newcastle upon Tyne, United Kingdom, **3** School of Engineering, Newcastle University, Newcastle upon Tyne, United Kingdom, **4** Chemical and Biochemical Engineering Department, University Politehnica of Bucharest, Bucharest, Romania

¤a Current address: Department of Oncology, University of Oxford, Oxford, United Kingdom
¤b Current address: School of Engineering, University of Glasgow, Glasgow, United Kingdom
* tom.curtis@ncl.ac.uk (TC); paolo.zuliani@ncl.ac.uk (PZ)

🔓 OPEN ACCESS

Abstract

We present NUFEB (Newcastle University Frontiers in Engineering Biology), a flexible, efficient, and open source software for simulating the 3D dynamics of microbial communities. The tool is based on the Individual-based Modelling (IbM) approach, where microbes are represented as discrete units and their behaviour changes over time due to a variety of processes. This approach allows us to study population behaviours that emerge from the interaction between individuals and their environment. NUFEB is built on top of the classical molecular dynamics simulator LAMMPS (Large-scale Atomic/Molecular Massively Parallel Simulator), which we extended with IbM features. A wide range of biological, physical and chemical processes are implemented to explicitly model microbial systems, with particular emphasis on biofilms. NUFEB is fully parallelised and allows for the simulation of large numbers of microbes (10^7 individuals and beyond). The parallelisation is based on a domain decomposition scheme that divides the domain into multiple sub-domains which are distributed to different processors. NUFEB also offers a collection of post-processing routines for the visualisation and analysis of simulation output. In this article, we give an overview of NUFEB's functionalities and implementation details. We provide examples that illustrate the type of microbial systems NUFEB can be used to model and simulate.

Author summary

Individual-based Models (IbM) are one of the most promising frameworks to study microbial communities, as they can explicitly describe the behaviour of each cell. The development of a general-purpose IbM solver should focus on efficiency and flexibility due to the unique characteristics of microbial systems. However, available tools for these

purposes present significant limitations. Most of them only facilitate serial computing for single simulation, or only focus on biological processes, but do not model mechanical and chemical processes in detail. In this work, we introduce the IbM solver NUFEB that addresses some of these shortcomings. The tool facilitates the modelling of much needed biological, chemical, physical and individual microbes in detail, and offers the flexibility of model extension and customisation. NUFEB is also fully parallelised and allows for the simulation of large complex microbial system. In this paper, we first give an overview of NUFEB's functionalities and implementation details. Then, we use NUFEB to model and simulate a biofilm system with fluid dynamics, and a large and complex biofilm system with multiple microbial functional groups and multiple nutrients.

This is a *PLOS Computational Biology* Software paper.

Introduction

Microbial communities are groups of microbes that live together in a contiguous environment and interact with each other. The presence of microbial communities on the planet plays an important role in natural processes, as well as in environmental engineering applications such as wastewater treatment [1], waste recycling [2] and the production of alternative energy source [3]. Therefore, studies on how these communities form and behave have become increasingly important over the past few decades [4, 5].

Work on microbial communities has revealed that the community's emergent behaviour arises from a variety of interactions between microbes and their local environment. *In vitro* experiments offer a way to gain insights into these complex interactions, but at great expense in time and resources. On the other hand, *in silico* computational models and numerical simulations could help researchers to investigate and predict how complex processes affect the behaviour of biological systems in an explicit and efficient way. Different approaches have been developed for modelling microbial communities [6–8]. One of the most promising strategies is to develop a mathematical model from the description of the characteristics and behaviour of the individual microbes, usually referred to as Individual-based Models (IbM) [9, 10]. In conventional IbM, the microbes are represented as rigid particles, each of which is associated with a set of properties such as mass, position, and velocity. These properties are affected by internal or external processes (e.g., diffusion), resulting in microbial growth, decay, motility, etc. Therefore, IbM are particularly useful when one is interested in understanding how individual heterogeneity and local interactions influence an emergent behaviour.

The development of an IbM solver should focus on the following aspects. First, the solver needs to be flexible. Depending on the purpose of the model, IbM may involve multiple microbial functional groups, nutrients and sophisticated biological, chemical and physical processes, or sometimes it may be a simple model that describes mono-functional group or focuses on a few processes. Thus, it is important for the solver to be highly customisable (for building IbM) and extendible (with new IbM features). Second, the solver should be scalable. Simulation of large microbial communities is difficult since they contain a very high number of individuals. Different modelling strategies have been proposed to overcome this limitation, including using super-individuals and statistically representative volume elements [11]. However, there is very little work on the development of a scalable IbM solver. Parallel computing can help scalability by using multiple computer resources to simulate many individuals simultaneously. This is accomplished by breaking the problem domain into discrete sub-domains which

separate out the individuals as much as possible, allowing each processing element to simulate the local interactions between individuals whilst minimizing the interactions between sub-domains. In this way each sub-domain can largely run concurrently with the others.

In this work, we present a three-dimensional, open-source, and massively parallel IbM solver called NUFEB that addresses these desired features above. The purpose of NUFEB is to offer a flexible and efficient framework for simulating microbial communities at the micro-scale, with an emphasis on biofilms. A comprehensive IbM is implemented in the solver which explicitly models biological, chemical and physical processes, as well as individual microbes. The present solver supports parallel computing and allows flexible extension and customisation of the model. NUFEB is based on the state-of-the-art software LAMMPS (Large-scale Atomic Molecular Massively Parallel Simulator) [12]. We selected LAMMPS because of its open-source, parallel, and extendible nature. There are several open-source IbM solvers that have been developed over the past decade and widely applied to microbiology research, such as iDynoMiCS [13], SimBiotics [14], BioDynaMo [15], and CellModeller [16]. However, most of them only facilitate serial computing for single simulation, or focus only on biological processes, but do not model mechanical and chemical processes in detail. The NUFEB simulator instead includes all of these features. A prototype NUFEB implementation was used in [17] to study physical behaviour of microbial communities. In this manuscript we focus on a major improvement of the tool, in which new features and enhancements have been developed including three-way coupling with fluid dynamics (two-way coupled fluid-particle interactions plus particle-particle interactions), code parallelisation, chemical processes (pH dynamics, thermodynamics, and gas-liquid transfer), and post-processing routines.

Model description

In this section, we describe and review the IbM implemented in NUFEB. The model is described through the ODD protocol (Overview, Design concepts and Details) which is a standard structure for clearly and efficiently describing IbM [18]. We build on the ideas described previously in [17] and extend them to cope with hydrodynamics and chemical processes. For the sake of accuracy, in the following we use the term "microbe" when describing biological and chemical processes, and use the term "particle" when describing physical process. However, the two terms effectively mean the same thing—"individual" of the IbM.

Model overview

The purpose of the IbM implemented in NUFEB is to model a range of microbial systems (including biofilm and flocs) at the micro-scale, in order to study and predict their population-level properties and behaviours emerged from the interactions between individuals and their environment.

In the model, microbes are described as soft spheres, with each individual having a set of state variables including position, density, velocity, force, mass, diameter, outer-mass, outer-diameter, growth rate, yield, etc. These attributes vary among individuals and can change through time. Outer-diameter and outer-mass are used to represent an EPS (Extracellular Polymeric Substances) shell: in some microbes EPS is initially accumulated as an extra shell around the particle. Microbial functional groups (types) are groups of one or more individual microbes that share same characteristics or parameters (such as maximum growth rate), which are constant throughout a simulation. The separation of individuals into different functional groups is based on their specific metabolism.

The computational domain is the environment where microbes reside and the biological, physical and chemical processes take place. It is defined as a micro-scale 3D rectangular
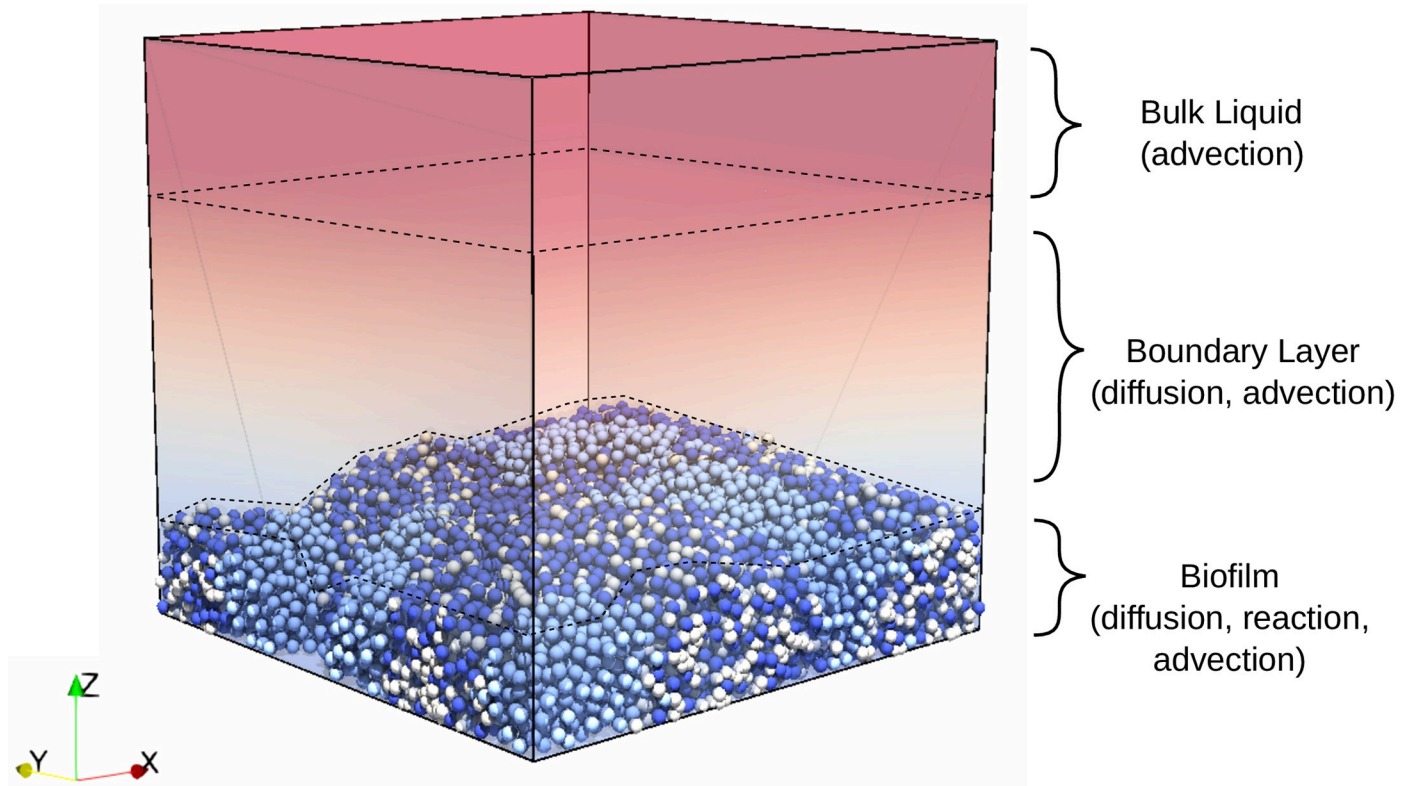
Fig 1. The computational domain with different regions. The top part is the well mixed bulk liquid region, the middle part is the diffusion boundary layer region, and the bottom part represents the biofilm region.

box with dimensions $L_X \times L_Y \times L_Z$. The size of the domain normally ranges from hundreds to thousands micrometers for micro-scale simulation. Therefore, the computational domain is considered as a sub-space of a macro-scale bioreactor or any other large-scale microbiological system. We assume that the macro-scale ecosystem is made up of replicates of the micro-scale domain if the bioreactor is perfectly mixed [17, 19]. Within the domain, chemical properties such as nutrient concentration, pH, and Gibbs free energies for microbial catabolism and anabolism are represented as continuous fields. To resolve their dynamics over time and space, the domain is discretised into Cartesian grid elements so that the values can be calculated at each discrete voxel on the meshed geometry. The style of domain boundary can be defined as either periodic or fixed. The former allows particles to cross the boundary, and reappear on the opposite side of the domain, while a fixed wall prevents particles to interact across the boundary.

The IbM allows to model a biofilm system where different regions may be defined within the computational domain [20] (Fig 1). The biofilm region is the volume occupied by microbial agents and their EPS, in which the distribution of soluble chemical species is affected by both diffusion, reaction and advection (due to biofilm porosity) processes. The boundary layer region lies over the biofilm, so that nutrient diffusion and advection is resolved in this space. The bulk liquid region is situated at the top of the boundary layer, and nutrients in this area are assumed to be perfectly mixed with the same concentration as in the macro-scale bioreactor. We also assume that the boundary layer region stretches from the maximum biofilm thickness to the bulk liquid, with the height of the region specified by the user. The boundary

between the diffusion layer and bulk liquid regions is parallel with the bottom surface (substratum), but its location needs to be updated when the biofilm thickness changes.

Design concepts

Here we address several important concepts in the NUFEB model design.

Emergence. The morphology of the microbial system, and the spatial distribution of its microbial functional groups, emerge from local interactions due to microbial growth, division, decay and mechanical interaction.

Sensing. The growth of each individual depends on a number of chemical properties within the voxel in which the individual resides, including nutrient concentration, pH, and Gibbs free energy. The motion of each individual results from local mechanical interactions and fluid flow.

Observation. At each given step, the physical and biological states of each individual (e.g., location, mass, size, type, growth rate) and the chemical state of each voxel (e.g., nutrient concentration, pH) are stored.

Interaction. The direct interactions among individuals are driven by mechanical forces, for example, contact force, EPS adhesive force, drag force, etc.

Stochasticity. The stochastic processes considered in the model include the size and location of daughter microbes after division, the location of EPS particle after EPS production, and the initial distribution of microbes.

Details of the sub-models

The processes that influence microbe activities are considered under three main sub-models: biological, physical, and chemical.

Biological processes. The NUFEB biological sub-model handles microbial metabolism, growth, decay, and reproduction (cell division and EPS formation). Details of the biological sub-model are given below.

Microbe growth and decay. An individual microbe grows and its mass increases by consuming nearby nutrients. The process of growth and decay is described by the following ordinary differential equation:

$$\frac{dm_i}{dt} = \mu_i m_i ,$$

(1)

where m_i is the biomass of the ith microbe, and μ_i is the specified growth rate. To determine μ_i, two growth models are implemented: (i) *Monod-based* and (ii) *energy-based*. The user can choose one of the growth models when configuring the simulation to run. For exemplification, the Monod-based growth model implements the work described in [17] and [19]. Three functional groups of microbes and two inert states are considered. They include: active heterotrophs (HET), ammonia oxidizing bacteria (AOB), nitrite oxidizing bacteria (NOB), and inactive EPS and dead cells. Microbial growth is based on Monod kinetics driven by the local concentration of nutrients ($S_{\text{substrate}}$, $S_{\text{NH}_4^+}$, $S_{\text{NO}_2^-}$, S_{O_2}) at the voxel in which each microbe resides [21]. The decay rate is assumed to be first order.

The *energy-based growth model* implements the work proposed in [22]. In this model, the growth rate of each microbe μ_i is determined not only by nutrient availability but also by the amount of energy available for its metabolism:

$$\mu_i = Y_i \times (q_i^{met} - m_i^{req}) ,$$

(2)

where the maximum growth yield Y_i is estimated by using the Energy Dissipation Method

[23], q_i^{met} is the metabolic rate which depends on the availability of nutrients (in particular, their dissociation forms), and m_i^{req} is the average maintenance requirement. Thus, a microbe grows if it harvests more energy than the necessary for its maintenance requirement. On the other hand, the microbe decays when the energy requirement is not met.

Microbe division and death. Microbe division is the result of biomass growth, while death is the result of biomass decay. Both are considered as instantaneous processes. Division occurs if the diameter of a microbe reaches a user-specified threshold value; the cell then divides into two daughter cells. The total mass of the two daughter cells is always conserved from the parent cell. One daughter cell is (uniformly) randomly assigned 40%–60% of the parent cell's mass, and the other gets the rest. Also, one daughter cell takes the location of the parent cell while the centre of the other daughter cell is (uniformly) randomly chosen at a distance *d* (distance between the centres of the two agents) corresponding to the sum of the diameter of the daughters.

The size of a microbe decreases when nutrients are limited or the energy available is not sufficient to meet maintenance requirements. Microbes which shrink below a user-specified minimum diameter are considered as dead. Their type then changes to the dead type. Dead cells do not perform biological activities, but their biomass will linearly convert to substrate to be consumed by other microbes. For computational efficiency, decaying cells are removed from the system when their size is sufficiently small (defined as 10 times smaller than the death threshold diameter).

EPS production. The Monod-based growth model allows active heterotrophs to secrete EPS into their neighbouring environment. The EPS play an important role in microbial aggregation by offering a protective medium. The production process follows the approach presented in [17] and [24] with the simplification that EPS are secreted by heterotrophs only. Initially, EPS is accumulated as an extra shell around a HET particle (note that EPS density is lower than microbe density). When the relative thickness of the EPS shell of the HET particle exceeds a certain threshold value, around half (uniformly random ratio between 0.4-0.6) of the EPS mass excretes as a separate EPS particle and is (uniformly) randomly placed next to the HET.

Physical processes. NUFEB's physical sub-model includes two key features and the dependencies between them: microbes (particles) and fluid. Microbes interact among themselves and with the ambient fluid. The physics of microbial motion is solved by using the discrete element method (DEM). Fluid momentum and continuity equations are solved based on computational fluid dynamics (CFD) and coupled with particle motion. The model developed in this study allows us to describe biofilm formation, detachment, and deformation based on mechanical interactions between particles. This is more realistic than many conventional IbMs where mechanistic approaches are not considered, for example, implementing detachment as probability or rate functions [24, 25].

Mechanical relaxation. When microbes grow and divide, the system may deviate from mechanical equilibrium (i.e., non-zero net force on particles) due to particle overlap or collision. Hence, mechanical relaxation is required to update the location of the particles and minimise the stored mechanical energy of the system. Mechanical relaxation is carried out using the discrete element method, and the Newtonian equations of motion are solved for each particle in a Lagrangian framework. The equation for the translational and rotational movement of particle *i* is given by:

$$m_i \frac{d\vec{v}_i}{dt} = \mathbf{F}_{c,i} + \mathbf{F}_{a,i} + \mathbf{F}_{d,i} + \dots , \qquad (3)$$

where m_i is the mass, and $\vec{v}_i$ is the velocity. The type of force acting on the particle varies

according to different biological systems. For example, the above equation takes into account three commonly used forces in microbial system. The contact force $\mathbf{F}_{c,i}$ is a pair-wise force exerted on the particles to resolve the overlap problem at the particle level. The force equation is solved based on Hooke's law, as described in [26]:

$$\mathbf{F}_{c,i} = \sum_{j=1}^{N_i} (K_n \delta \mathbf{n}_{i,j} - m_{i,j} \gamma_n \mathbf{v}_{i,j}) \,, \tag{4}$$

where N_i is the total number of neighbouring particles of i, K_n is the elastic constant for normal contact, $\delta \mathbf{n}_{i,j}$ are overlap distance between the center of particles i and its neighbour particle j, $m_{i,j}$ is the effective mass of particles i and j, γ_n is the viscoelastic damping constant for normal contact, and $\mathbf{v}_{i,j}$ is the relative velocity of the two particles.

The EPS adhesive force $\mathbf{F}_{a,i}$ is a pair-wise interaction, which is modelled as a van der Waals force [27]:

$$\mathbf{F}_{a,i} = \sum_{j=1}^{N_i} \frac{H_a r_{i,j}}{12 h_{min,i,j}^2} \mathbf{n}_{i,j} \,, \tag{5}$$

where H_a is the Hamaker coefficient, $r_{i,j}$ is the effective outer-radius of particles i and j, $h_{min,i,j}$ is the minimum separation distance of the two particles, and $\mathbf{n}_{i,j}$ is the unit vector from particle i to j.

The drag force $\mathbf{F}_{d,i}$ is the fluid-particle interaction force due to fluid flow, with direction opposite the microbe motion in a fluid. It is formulated as [27]:

$$\mathbf{F}_{d,i} = \frac{V_{p,i}}{\epsilon_{f,i}\epsilon_{s,i}} \beta_i (\mathbf{u}_{p,i} - \mathbf{U}_{f,i}) \,, \tag{6}$$

where $\epsilon_{s,i}$ is the particle volume fraction, $\epsilon_{f,i} = 1 - \epsilon_{s,i}$ is the fluid volume fraction, $V_{p,i}$ and $\mathbf{u}_{p,i}$ are volume and velocity of particle i, respectively, $\mathbf{U}_{f,i}$ is the fluid velocity imposed on particle i, and β_i is the drag correction coefficient [28]. Apart from the forces above, the LAMMPS framework offers mechanical interactions that one can apply directly to particles (see LAMMPS' user manual for more details [29]).

In the model, we assume that the mechanical equilibrium is obtained when the average pressure of the microbial community reaches the equilibrium state. The average pressure of the system due to mechanical interactions is calculated as [30]:

$$P = \frac{1}{3V} \left(\sum_{i=1}^{N} m_i \vec{v}_i \cdot \vec{v}_i + \sum_{i=1}^{N}\sum_{j>i}^{N} \vec{r}_{i,j} \cdot \vec{F}_{i,j} \right) \,, \tag{7}$$

here V is the sum of the volumes of particles. The first term in the bracket addresses the contribution from kinetic energy, where m_i is the mass of particle i, $\vec{v}_i$ is the velocity. The second term addresses the interaction energy, where $\vec{r}_{i,j}$ and $\vec{F}_{i,j}$ are the distance and force between two interacting particles i and j, respectively.

Fluid dynamics. Hydrodynamics is an important factor in microbial community modelling as many microorganisms are found in water where their behaviour is influenced by fluid flow in two ways: transport of nutrients and detachment. Nevertheless, accurate hydrodynamics has rarely been considered in 3D microbial community modelling due to its computational complexity [31]. In NUFEB, with the support of code parallelisation, hydrodynamics is introduced by using the two-way coupled CFD-DEM approach [32, 33]: the fluid flow affects the motion of particles and the particles in turn affect the motion of the fluid. DEM solves the

motion of Lagrangian particles based on Newton's second law, while CFD (computational fluid dynamics) tracks the motion of fluid based on locally averaged Navier-Stokes equations.

The particle velocity at each voxel in space is replaced by its average, and the locally averaged incompressible continuity and momentum equations for the fluid phase are given by [34]:

$$\nabla \cdot (\epsilon_s \mathbf{U}_s + \epsilon_f \mathbf{U}_f) = 0 \,, \tag{8}$$

and

$$\frac{\partial(\epsilon_f \mathbf{U}_f)}{\partial t} + \nabla \cdot \left(\epsilon_f \mathbf{U}_f \mathbf{U}_f \right) = \frac{1}{\rho_f} \left(-\nabla P + \epsilon_f \nabla \cdot \mathcal{R} + \epsilon_f \rho_f \mathbf{g} + \mathbf{F}_f \right) \,, \tag{9}$$

where ϵ_s, $\mathbf{U}_s$, and $\mathbf{F}_f$ are the fields of the solid volume fraction, velocity and fluid-particle interaction forces (e.g., drag force) of microbes, respectively. They are obtained by averaging discrete particle data in DEM [35, 36]; ϵ_f is the fluid volume fraction and $\mathbf{U}_f$ is the fluid velocity. Besides the fluid-particle interaction, the terms on the right-hand side of Eq 9 also include the fluid density ρ_f, the pressure gradient ∇P, the divergence of the stress tensor $\mathcal{R}$ and the gravitational acceleration $\mathbf{g}$. The fluid momentum equations are discretised and solved on an Eulerian grid by a finite volume method. The results, in particular the velocity field and the particle drag force, are used for solving nutrient transport and mechanical relaxation, respectively.

Chemical processes. Nutrient transport is described using the diffusion-advection-reaction equation. To improve the representation of microbial growth, NUFEB also implemented the work proposed in [37] to allow pH dynamics and gas-liquid transfer to be considered. In this section, we briefly address the main ideas of this chemical sub-model.

Nutrient consumption. The rate of nutrient consumption (or reaction rate) is calculated at each voxel. The reaction rates in the Monod-based growth model are defined according to Monod kinetics, where the stoichiometric matrix for particulate and soluble components is given in the S1 File. By contrast, in the energy-based growth model the consumption/formation rate of the ith microbe for each soluble species is driven by the microbial growth rate and the stoichiometric coefficients of the overall growth reaction, and is formalised as follows [22]:

$$R_i = \mu_i \left(\frac{1}{Y_i} Cat_i + Ana_i \right) X \,, \tag{10}$$

where Cat_i is the free energy supplied by the microbial catabolic reactions, Ana_i is the free energy required by the anabolic reactions, and X is the biomass density.

Nutrient mass balance. Nutrient concentration at each point within the computational domain is affected by different processes. For example, nutrient advection due to fluid flow, microbial growth causing nutrient consumption, or solute component transferred into gas. On the other hand, the variation of the concentration also affects biological activities. A typical example would be that aerobes such as AOB can survive and grow in aerobic regions but not in anoxic regions.

To solve the nutrient distribution for each soluble component, the following diffusion-advection-reaction equation for the solute concentration S is employed in the model:

$$\frac{\partial S}{\partial t} = \nabla \cdot (D \nabla S) - \nabla \cdot (\vec{v} S) + R \,. \tag{11}$$

On the right-hand side of the equation, the first part is the diffusion term which describes nutrient movement from an area of high concentration to an area of low concentration, where ∇ is the gradient operator and D is the diffusion coefficient. The second part is the advection

term which describes nutrient motion along the fluid flow, where $\vec{v}$ is the fluid velocity field. Finally, R is the reaction term which is governed by both biological activities and chemical activities.

Nutrient concentration in the bulk region is dynamically updated according to the following mass balance equation [20]:

$$\frac{dS_b}{dt} = \frac{Q}{V}(S_{in} - S_b) + \frac{A_f}{VL_XL_Y} \iiint R(x, y, z)dxdydz \, . \tag{12}$$

The bulk concentration S_b of each soluble component is influenced by the nutrient inflow and outflow in the bioreactor (Q is the volumetric flow rate, V is the bioreactor volume and S_{in} is the influent nutrient concentration), as well as the total consumption rate in the biofilm volume in the bioreactor (A_f is the surface area of the biofilm, and $R(x, y, z)$ is the reaction rate at each voxel).

pH dynamics. We assume the influent enters the bulk liquid with a fixed pH value. However, the pH varies in space and time due to change in nutrient concentration as a result of microbial activity. The model considers acid-base reactions as equilibrium processes. The kinetic expressions are amended to consider only the non-charged form of the nutrients (e.g., HNO_2 but not NO_2^-). The concentration of all dissociated and undissociated forms can be expressed as a function of the proton (H^+) concentration. Then, the proton concentration can be determined by finding the root between 1 and 10^{-14} of the following charge balance equation, using an implicit Newton-Raphson approximation:

$$[H^+] + \sum_{i=1}^{p} m[S_i^{m+}] = [OH^-] + \sum_{j=1}^{q} n[S_j^{n-}] \, , \tag{13}$$

where p and q are the total number of cations and anions contributing to the pH, respectively, m and n are the charges corresponding to the cations and anions considered in the dissociation equilibrium, and S is the concentrations of cations and anions calculated based on the Gibbs free energy adjusted for ambient temperature.

Gas-liquid transfer. A gas field can be defined in NUFEB to describe the rate of nutrient mass transfer from gas to liquid or vice-versa. The phase is modelled as a volume fraction inside the biofilm region. The equilibrium between liquid and gas is disturbed by the acid-base reaction and the microbial activity taking place in the biofilm and bulk liquid. On the other hand, mass transfer may also affect microbial growth due to varying nutrient concentration in the liquid phase. The reaction rate from gas to liquid $R_{G \to L}$ of a given nutrient can be expressed by the following equation:

$$R_{G \to L} = K_L a \cdot (S_{gas} - \frac{S_{liq}}{K_H}) \, . \tag{14}$$

The rate is determined by the mass transfer coefficient of chemical component $K_L a$, the gas concentration S_{gas} in the head space, the saturation liquid concentration S_{liq}, and Henry's constant K_H. Mass transfer from liquid to gas is formalised as follows:

$$R_{L \to G} = \frac{-R_{G \to L}V_{gas}}{R_g T} \, , \tag{15}$$

where V_{gas} is the volume of the reactor head space, which is considered of equal size to the computational domain, and R_g and T are the ideal gas constant and temperature, respectively.

Design and implementations

NUFEB is developed in C++ as a user package within the LAMMPS platform. In this section, we summarise the NUFEB functionalities and give some implementation details.

IbM in LAMMPS

NUFEB is built on top of LAMMPS and extends it with IbM features. LAMMPS is a classical molecular dynamics simulator and primarily solves particle physics, including a wide range of inter-particle interactions and potentials [12]. In NUFEB, a new sphere-like particle type is defined with additional attributes (e.g., outer-diameter, outer-mass) to represent a microbe. Microbes are grouped into different functional groups, and members of each group share the same biological parameters. The computational domain is restricted to a 3D rectangular box with user-specified dimensions, Cartesian grid size and boundary conditions. In LAMMPS, a "fix" is any operation that applies to the system during time integration. Examples include the updating of particle locations, velocities, forces. NUFEB defines a series of new fixes to perform the IbM related processes previously described. The user can also extend the tool with other processes by implementing new fix commands. During the simulation, fixes are invoked at a user-defined frequency to update field quantities and microbe attributes. For the purpose of computational efficiency and different modelling systems, NUFEB allows users to customise a simulation by enabling/disabling any of the fix commands from the input setting.

In order to execute a NUFEB simulation, an input script (a text file) is prepared with certain commands and parameters. NUFEB will read those commands and parameters, one line at a time. Each command causes NUFEB to take some action, such as setting initial conditions, performing IbM processes, exporting results, or running a simulation. In Fig 2, we show an excerpt of an input script used in this work (Case Study 2). The details of the input format are described in S1 Manual.

```
....
# define variables used in fix commands
variable EPSdens equal 30
variable etaHET equal 0.6
variable divDia equal 1.3e-6

# define IbM processes
fix k1 all kinetics/growth/monod v_EPSdens v_etaHET
fix k2 all kinetics/diffusion v_tol pp pp nd kg
fix d1 all divide 1 v_EPSdens v_divDia 41341
fix e1 HET eps_extract 1 v_EPSratio v_EPSdens 5234
....
```

Fig 2. An example of (partial) input script for NUFEB simulation. A list of "fix" commands defines IbM processes that apply to the simulation, which includes Monod-based growth (`k1`), nutrient mass balance (`k2`), cell division (`d1`), and EPS production (`e1`). Each fix command may require one or more parameters for the model specification, such as EPS density (`EPSdens`), division diameter (`divDia`), and HET reduction factor in anoxic condition (`etaHET`). The last parameter is introduced due to the fact that the observed growth rate of HET in anoxic condition was always smaller than that in aerobic condition.

https://doi.org/10.1371/journal.pcbi.1007125.g002

The procedure of a classical IbM simulation in NUFEB is presented in Algorithm 1. Note that when solving nutrient mass balance (step 5), if energy-growth model is applied, the computation is accomplished with pH dynamics and gas-liquid transfer to update the concentration of dissociation forms, pH and reaction rate. The Monod-based model, however, does not couple with pH and gas-liquid transfer in the current implementation. The model's processes can be operated sequentially as they are on different timescales. The mechanical timestep is typically of the order of 10^{-7}s; the diffusion timestep is of the order of 10^{-4}s, while the biological timestep is much larger ranging from minutes to hours. The coupling between multiple timescales relies on the pseudo steady-state approximation and the frozen state [31]. For example, when a steady state solute concentration is reached at each biological timestep, the concentration is assumed to remain unchanged (frozen state) until the next biological step. In this way, the computational load for solving fast dynamic processes can be significantly reduced.

Algorithm 1 (IbM simulation procedure)

```
1: input: initial states of computation domain, microbes and fields
2: output: states of all microbes and fields at each output time step
3: while biological time step t_bio < t_end do
4:    solve fluid dynamics to update velocity field and particle
      (drag) force
5:    solve nutrient mass balance to update solute concentration field
6:    update nutrient concentration in bulk based on the total consump-
      tion rate in biofilm region
7:    perform microbe growth to update biomass and size
8:    perform microbe division, death and EPS production
9:    perform mechanical relaxation to update microbe position and
      velocity
10:   update boundary layer location and neighbour list
11:   t_bio = t_bio + Δt
12: end while
```

Mechanical relaxation is resolved by using the Verlet algorithm provided by LAMMPS [38]. The computation of interaction forces between particles, such as contact force, relies on LAMMPS' neighbour lists. During the Verlet integration, instead of iterating through every other particle, which would result in a quadratic time complexity algorithm, LAMMPS maintains a list of neighbours for each particle. The computation of the interaction force is only performed between a particle and its corresponding neighbours. The neighbour lists must be updated from time to time depending on the microbe's displacement and division.

The nutrient mass balance equation is discretised on a Marker-And-Cell (MAC) uniform grid and the concentration scalar is defined at the centre of the cubic voxel. The temporal and spatial derivatives of the transport equation are discretised by Forward Euler and Central Finite Differences, respectively. Further details about the calculations are given in the S1 File. Depending on the physical situation, different boundary conditions can be chosen when solving the equation. For instance, a biofilm system would normally uses non-flux Neumann conditions through the bottom surface to model impermeable support material, Dirichlet boundary conditions at the top surface as it is assumed to connect with bulk environment, and periodic boundary conditions for the rest of the four surfaces [20].

Coupling with fluid dynamics

NUFEB employs and extends the existing CFD-DEM solver SediFoam for the simulation of hydrodynamics [39]. SediFoam provides a flexible interface between the two open-source solvers LAMMPS and OpenFOAM. LAMMPS aims to simulate particle motions, while
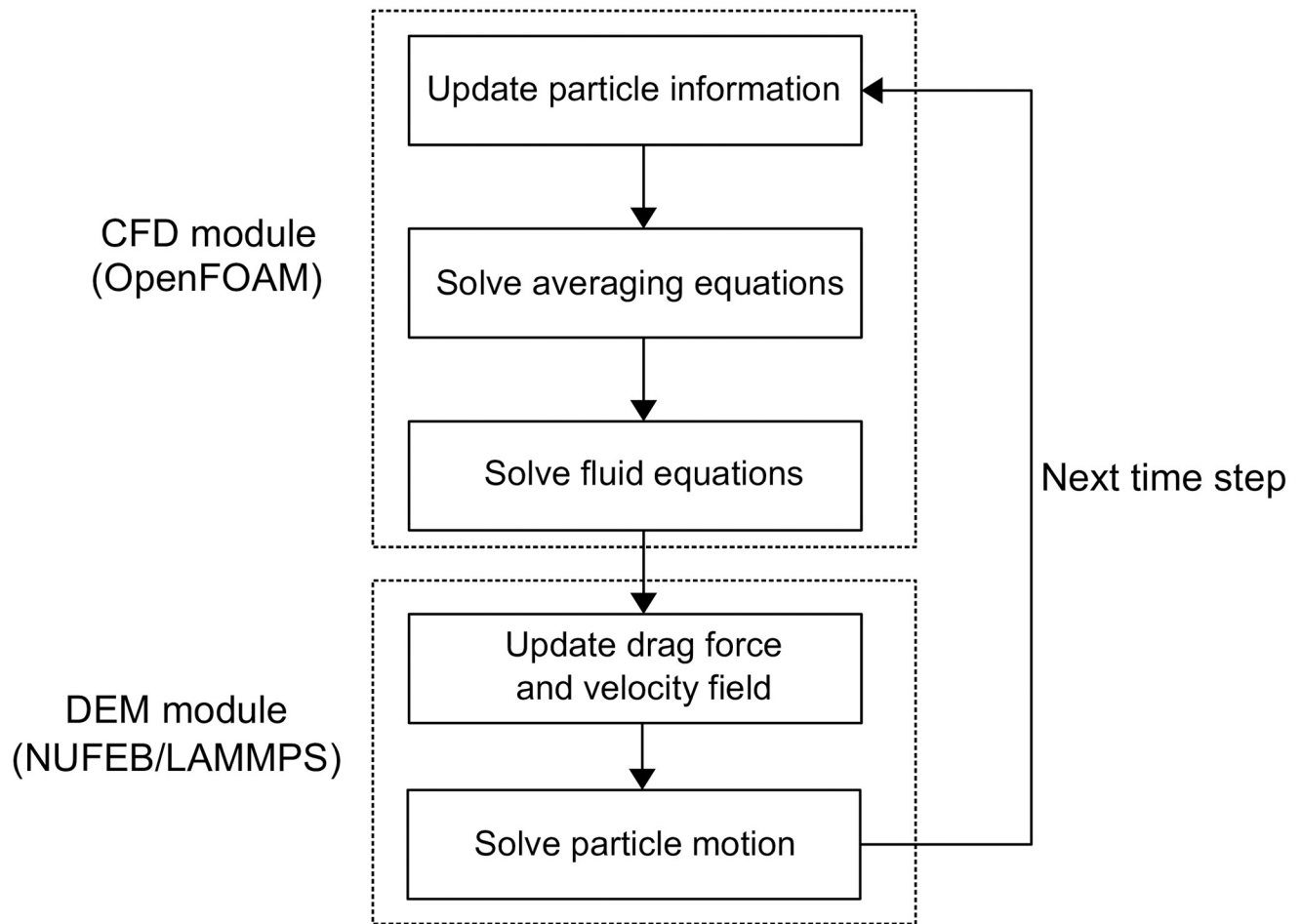
Fig 3. Block diagram of CFD-DEM coupling. Diagram adopted from [39]. Fluid dynamics is solved in the CFD module and particle motion is solved in the DEM module. Particle and field information are transferred between the two modules based on an averaging procedure.

OpenFOAM (Open Field Operation and Manipulation) is a parallel CFD solver that can perform three-dimensional fluid flow simulations [40]. Inbetween them, SediFoam offers efficient parallel algorithms that transfers and maps the properties of Lagrangian particles to an Eulerian mesh, and vice versa. In this work, we also extend SediFoam for compatibility with IbM features, in particular, to transfer and map the information of new divided particles and velocity field between the solvers.

The schema of the CFD-DEM coupling is shown in Fig 3. At each fluid timestep, particle information (e.g., mass, force, velocity) is transferred from the DEM module to the CFD module. A particles list maintained by OpenFOAM is updated based on the obtained information. An averaging procedure is then performed to convert the properties of discrete particles to the Eulerian CFD mesh. The fluid momentum equations are discretised and solved on the Eulerian mesh by the finite volume method. The PISO (Pressure-Implicit with Splitting of Operators) algorithm is followed to solve the equations [41]. In the DEM module, the solutions of drag force and velocity field obtained from OpenFOAM are assigned to each particle and corresponding mesh grid, respectively. The drag force will be taken into account in evolving the motion of the particles. The velocity field is used for solving the advection-diffusion-reaction equation (Eq 11).

Code parallelisation

The IbM implementation in NUFEB involves particle-based functions (e.g., contact force, division, and EPS excretion) and continuum-based functions (e.g., nutrient mass balance, and pH). The parallelisation of the former function group is based on LAMMPS' parallel mechanism. The computational domain is spatially decomposed into multiple MPI (Message Passing Interface) processes (sub-domains). Each sub-domain contains local particles as well as ghost particles. Local particles are those residing in the owned sub-domain, and each process is responsible for updating the status of their local particles. Ghost particles are copies of particles owned by neighbouring processes. During the simulation, the local particles obtain information from their ghost (and neighbour) counterparts for calculating and updating their physical properties (e.g., forces). The neighbour lists require updating when particles are moved due to mechanical interactions, and created/deleted due to microbe division, decay, etc.

Continuum-based functions have their variables computed using a uniform grid. Parallelism is achieved by spatially decomposing the computational domain into sub-domains. Computations such as diffusion and advection require solute information from adjacent cells. Therefore, like particle-based functions, grid cells on the boundary of each sub-domain need to be communicated to neighbouring sub-domains and are treated locally as ghost cells. The implementation of grid decomposition also considers the sub-domain box to be always conforming to the uniform grid boundary.

Automatic vectorisation was employed to further optimise computation intensive routines, such as pH calculation. The loops in the routines are vectorised and can be performed simultaneously. The vectorisation is achieved together with the use of control directives (i.e., `#pragmas`) to instruct the compiler on how to handle data dependencies within a loop.

Other features

During simulation, NUFEB allows to output any state variable of microbes or voxels based on the keywords given in the input script. The output results can be stored into various formats for visualisation or analysis. The supported formats are VTK, POVray and HDF5. The VTK binary format is readable by the VTK visualisation toolkit or other visualisation tools, such as ParaView (all biofilm figures shown in the Results section were produced from ParaView). A post-processing routine is implemented to convert the LAMMPS default format to POVray image, which supports high quality rendering of particles. HDF5 is a hierarchical, filesystem-like data format supported by a number of popular software platforms, including Java, MATLAB and Python. This allows the user to directly import simulation data to any of the platforms for further analysis.

To understand the morphological dynamics of microbial systems, characteristics such as biofilm average height, biofilm surface roughness, floc equivalent diameter and floc fractal dimension can be measured and exported during simulation. These aggregated characteristics are essential factors to study and design microbial system [42]. Parallelisation of the characteristics measurements is supported by NUFEB and based on domain decomposition.

NUFEB also supports most of the LAMMPS default commands, which can be useful in microbial simulations. For example, the *restart* and *read restart* functions write out the current state of a simulation as a binary file, and then start a new simulation with the previously saved system configuration; the *lattice* and *create atoms* functions allow to automatically create large numbers of initial microbes based on a user-defined lattice structure; *load balance* is performed with the objective of maintaining the same number of particles in each sub-domain in parallel runs. During the simulation, this function adjusts the size and shape of the sub-domains to balance the computational cost in the processors.

Results

We have successfully validated NUFEB against two biofilm benchmark problems BM2 and BM3 proposed by the International Water Association (IWA) task group on biofilm modelling [43, 44]. The validation results can be found in the S1 File. This section shows two further examples implemented using NUFEB. The case studies are based on biofilm systems, which are microbial communities of single or multiple microbial functional groups in which cells stick together and attach to a substratum by means of EPS.

Case study 1: Biofilm deformation and detachment

One of the outstanding questions in biofilm research is understanding how fluid-biofilm interactions affect the mechanical properties of biofilms. In this section, we describe how to use NUFEB to simulate a biofilm system with fluid dynamics by using three-way fluid and particle coupling. This is different from previous studies, where the coupling was from flow field to biofilm structure but not the other way [17], or assumes biofilm as a collection of 1D springs [45, 46].

To simulate a hydrodynamic biofilm, we apply fluid flow to a pre-grown biofilm that consists of heterotrophs and their EPS production. The biofilm is grown from 40 microbes inoculated on the substratum to a pre-determined height (80 μm) without flow and in an oxygen-limited condition (1×10^{-4} kg m^{-3}). In this way, a mushroom-shaped biofilm structure can be developed to model liquid filled voids and channels (see Fig 4(a) and S1 Video). Then, we impose a fluid flow to the biofilm. During the fluid stage, any biological process is considered to be in the frozen state due to the small time scale of hydrodynamic calculations, and nutrient mass balance is omitted for the sake of simplicity. The motion of microbes is driven by both particle-particle and particle-fluid interactions, including EPS adhesion, contact force and drag force, as described in Eqs (4)–(6). Note that the EPS adhesive force also exists between HETs due to their EPS shells. The physical model parameters are kept constant throughout the simulation and can be found in the S1 File. For boundary conditions, we impose a fixed
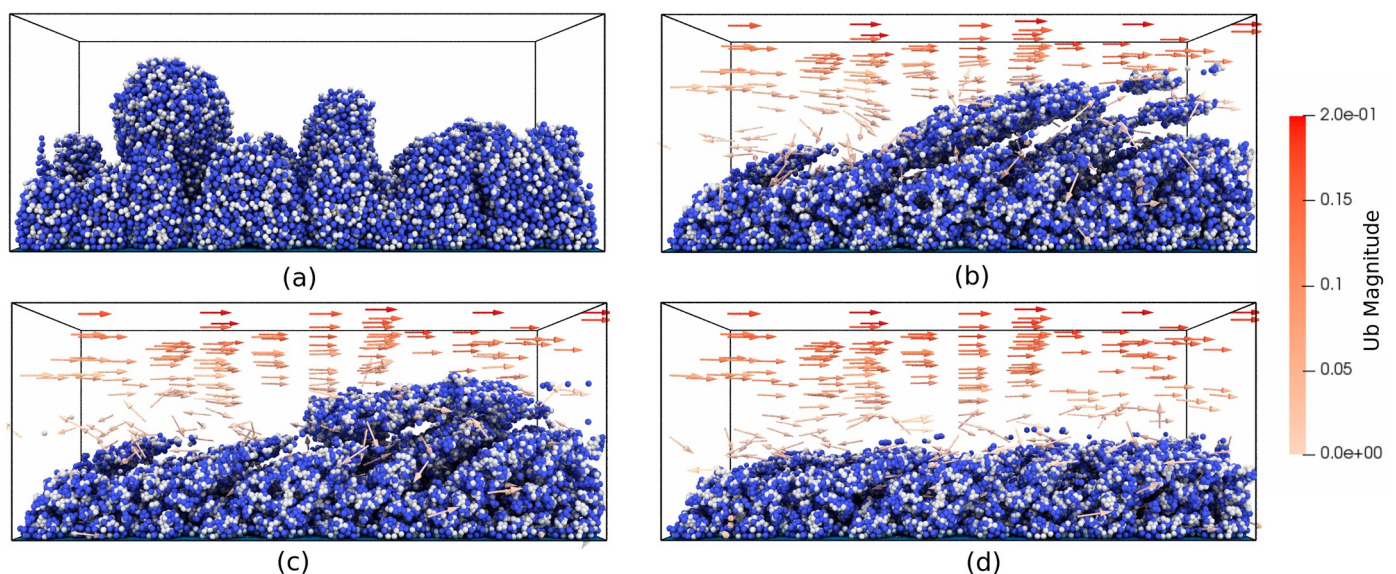

Fig 4. Biofilm deformation and detachment at U$_f$ = 0.2 m s^{-1}. (a) Time = 0; (b) Time = 0.0015s; (c) Time = 0.003s; (d) Time = 0.01s. The model simulates 4×10^4 particles. Particles crossing the domain boundary will be removed from the system. Particle colours are blue for heterotrophs and grey for EPS. CPU time is 8 hours with dual processors, and initial particle number is 41210.

https://doi.org/10.1371/journal.pcbi.1007125.g004

velocity $\mathbf{U}_f$ at the top surface with direction along the x-axis as inlet velocity, no-slip condition at the bottom surface and periodic conditions at other four surfaces. The pressure are enforced as zero gradient at the top and bottom surfaces.

Fig 4(b)–4(d) and S2 Video. show the biofilm deformation and detachment at $\mathbf{U}_f$ = 0.2 m s^{-1} (Reynolds number = 20). The biofilm deforms and microbes detach along with the flow direction. In the early stage of the detachment process, the top of the biofilm is highly elongated and forms filamentous streamers. However, most of the microbes are still connected together with cohesion, and there is only a small number of clusters detached from the head of the streamers due to cohesive failures (Fig 4(b)). As the fluid continues to flow, large chunks of microbes detach from the biofilm surface. These detached microbe chunks may also break-up again, re-agglomerate with other clusters or re-attach to the biofilm surface (Fig 4(c)). Such deformation and detachment events observed from our NUFEB simulation show qualitative agreement with both experimental results [47] and other numerical simulations using different methods [17, 46]. The deformation reaches a pseudo-steady-state when the mushroom-shaped biofilm protrusions are removed from the system. As a result, the biofilm morphology changes dramatically from a rough to a flat surface (Fig 4(d)). During the deformation, the fluid, represented as red arrows, travels around the biofilm. Due to the irregular shape of the biofilm and the high fluid velocity, small vortexes can be observed at the biofilm surface on both the upstream and downstream sides. This phenomenon has been observed in previous studies [46].

For a more quantitative measurement of the deforming biofilm, we evaluated the area density and surface roughness of the biofilm at different fluid velocities. The biofilm surface roughness is calculated by [17]:

$$roughness = \left(\frac{1}{L_X L_Y} \iint \left(h(x,y) - \bar{h}\right)^2 dxdy\right)^{1/2} , \tag{16}$$

where $h(x,y)$ is the biofilm height in the z direction at location (x, y) on the substratum, and $\bar{h}$ is the average biofilm height:

$$average\ height = \frac{1}{L_X L_Y} \iint h(x,y)dxdy , \tag{17}$$

As expected, when the fluid velocity increases the removed biomass also increases. For example, when $\mathbf{U}_f$ = 0.4 m s^{-1} is applied, the area density reaches steady-state after 0.004s, and the value decreases by 13% (Fig 5(a)). Note that the decrease does not take into account the change of basis. By contrast, the area density decreases less than 1% if $\mathbf{U}_f$ = 0.1 m s^{-1} is applied. The biofilm surface roughness shows a similar trend: the roughness decreases with increasing velocities (Fig 5(b)), indicating that biofilm morphology tends to be more flat in high-velocity fluid conditions as most of the mushroom protrusions can be removed.

Case study 2: Biofilm growth with 10^7 particles

In this case study, we first show the development of a large and complex biofilm system and then focus on the parallel efficiency of the simulations. The aim of this case study is to demonstrate the capability and performance of NUFEB in the simulation of larger biological systems.

Biofilm development.   The system is defined as a multi-functional group and multi-nutrient biofilm. In order to represent a more realistic biofilm, we explicitly consider nitrification as a two-step oxidation process that is performed by different groups of microbes: ammonia oxidizing bacteria (AOB) and nitrite oxidizing bacteria (NOB). In addition, the biofilm includes heterotrophs (HET) and their EPS production. The reaction model contains five
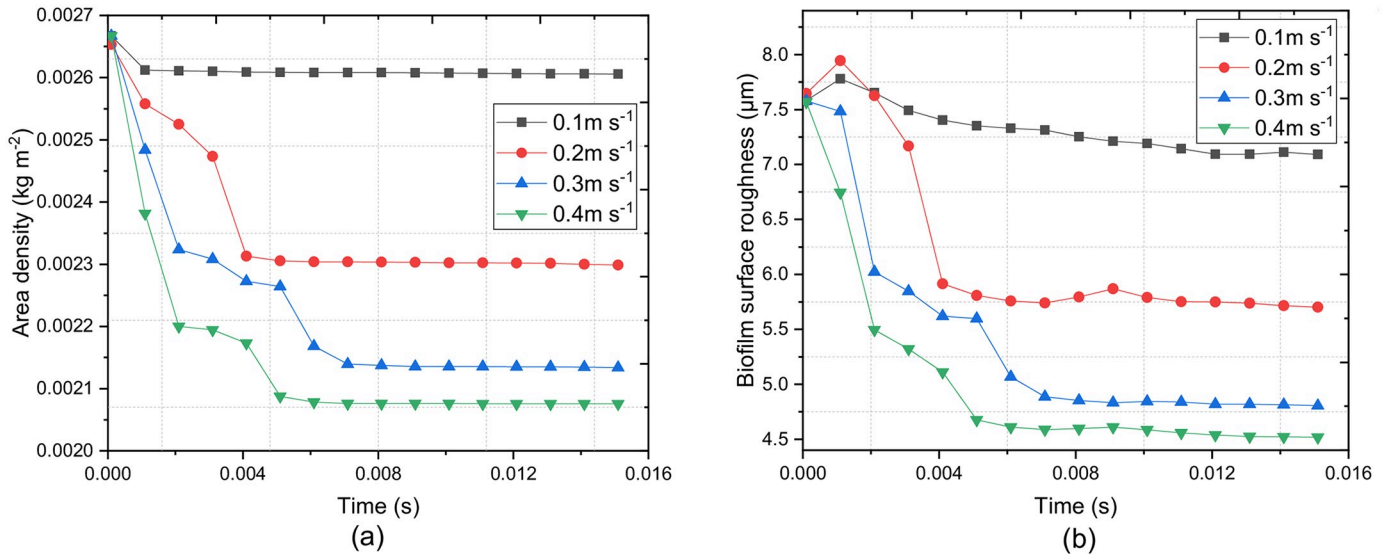
Fig 5. Effect of emergent properties on biofilm detachment. (a) biomasss area density, and (b) biofilm surface roughness.

soluble species, nutrients and products during microbial metabolism. The catabolic reactions include oxidation of ammonium NH_4^+ to nitrite NO_2^- by AOB, oxidation of nitrite to nitrate NO_3^- by NOB, and HET aerobic and anaerobic growth by consuming organic substrate in oxygenated conditions or nitrate in anoxic denitrifying conditions. The kinetics and reaction stoichiometry of the modelled processes and their corresponding parameters are detailed in the S1 File. The computational domain is divided into three regions. In the bulk region, nutrients are assumed to be completely mixed and their concentration is updated dynamically at each biological timestep, except for oxygen. We also assume that there is sufficient O_2 and NH_4^+ but no NO_2^- and NO_3^- in the reactor influent. So the concentration of the two N compounds can only come from the transformation of NH_4^+. In the boundary layer region, a 20 μm distance from the maximum biofilm thickness to the bulk liquid is defined for solving the nutrient gradient. In the biofilm region, instead of using the super particle method [20], a small division diameter (1.3 μm) is chosen to represent real microbe sizes (on average 1 μm [48]).

The simulation is run on an in-house HPC system at Newcastle University. In Fig 6 and S5 Video. we present the biofilm development over time. The system reaches 2.3×10^7 particles after 160 hours (CPU time = 30 hours). The initial particles are randomly placed on the substratum. In the early stages of biofilm formation, due to high growth rate and sufficient supply of substrate from bulk liquid, heterotrophs grow faster than nitrifiers (0, 60 and 120 hours). As the biomass grows, the system turns to substrate-limited condition for the HET group, while there is still sufficient NH_4^+ due to its high initial concentration that favours nitrifier growth. As a result, biofilm surface coverage of heterotrophs becomes smaller than nitrifiers (160 hours). This phenomenon matches previous experimental results where the nitrifying population can be significantly higher than heterotrophs in a substrate-limited reactor [49]. The biofilm geometry forms a wavy structure after 160 hours. This is because of a self-enhancing process from the non-uniform initial microbial distribution [50]. The spatial distribution of NO_2^- concentration is shown in Fig 7. It is clear that the NO_2^- concentration field differs according to the AOB distribution, where areas with high NO_2^- concentration are the locations where AOB clusters are present.
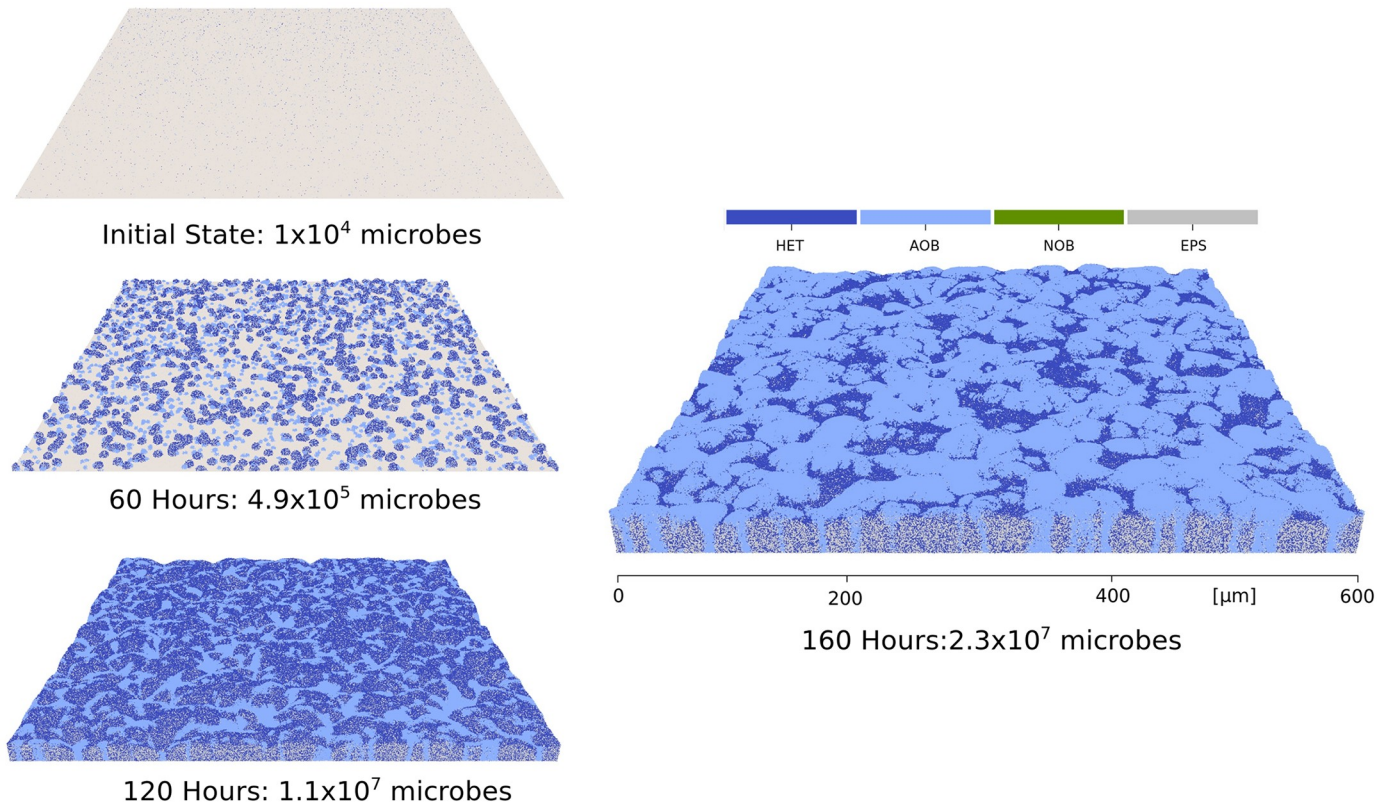
Fig 6. Biofilm development after 0, 60, 120, and 160 hours. The simulation uses 100 processors and 30 hours CPU time to reach 2.3×10^7 particles. The biological timestep is 0.25 hour. Particle colours are blue for heterotrophs, grey for EPS, light blue for AOB, and green for NOB.

Fig 8(a) shows a quantitative evaluation of the total biomass accumulation over time. The trend shows linear biomass increase which indicates that the total microbial growth rate is not yet balanced by the decay rate. Therefore, a biomass steady state is not achieved after 160 hours. This is due to the high-oxygen environment (1×10^{-2} kg m^{-3}) and the thin
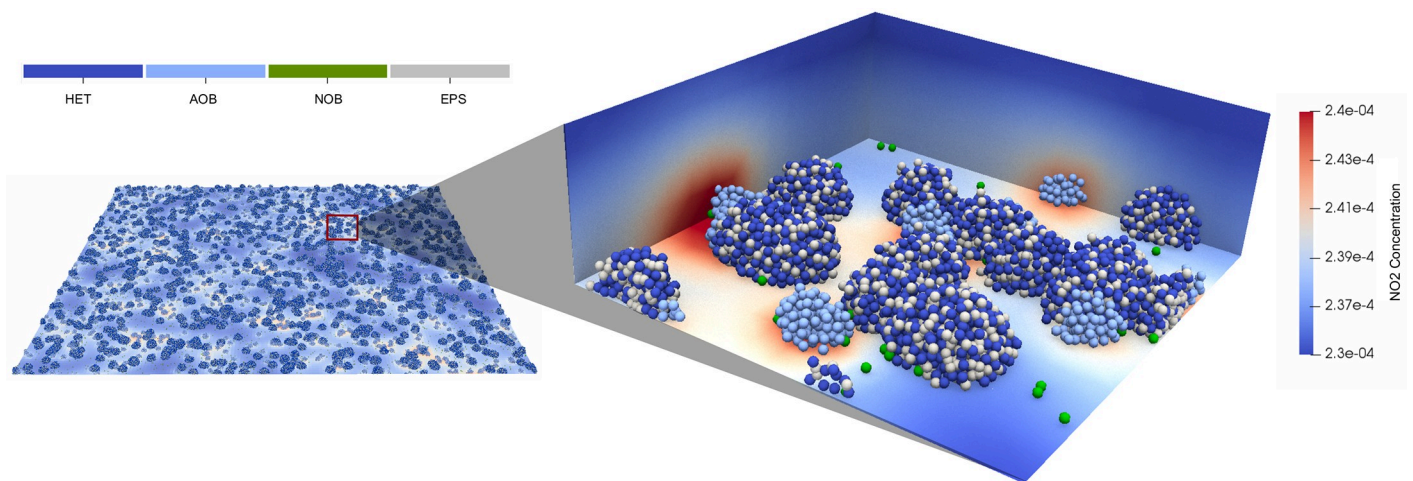

Fig 7. Nitrite concentration field at a small part of the simulated domain after 60 hours. The spatial distribution of NO_2^- concentration follows the nitrifier distribution. The areas where NO_2^- accumulates are due to production by AOB.
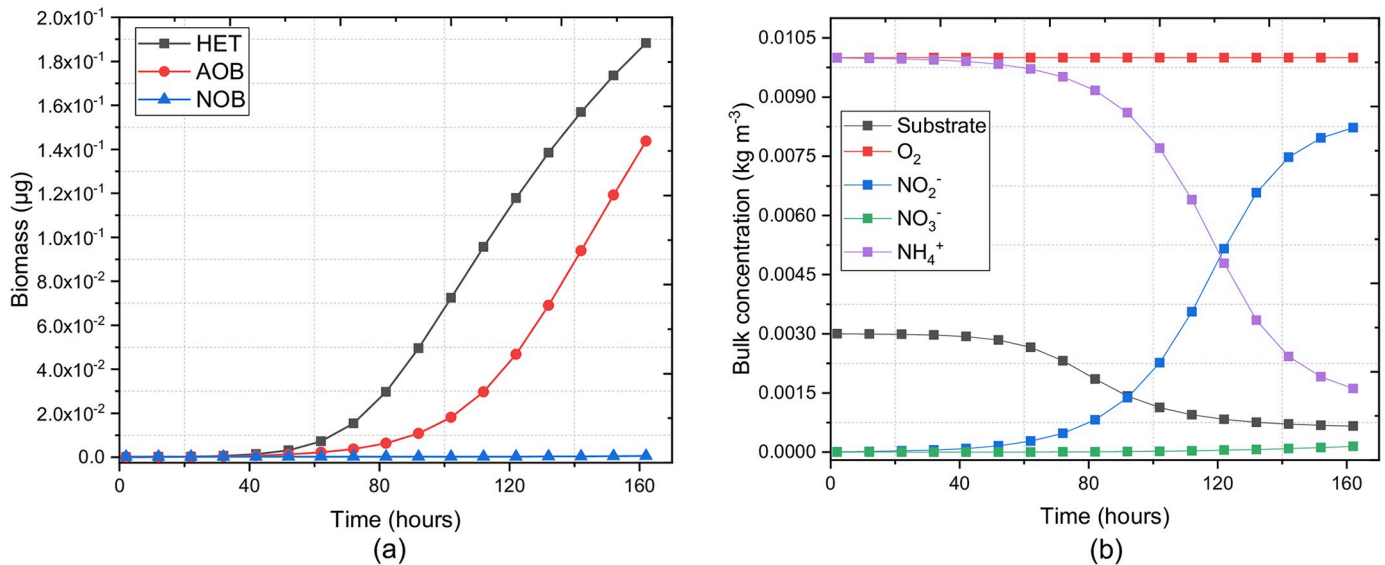
Fig 8. Quantitative evaluation of Case Study 2. (a) Total biomass of active functional groups over time, and (b) nutrients concentration in bulk liquid. Note that we assume the oxygen concentration in bulk liquid is kept constant by aeration. The slow growth of NOBs is due to the nutrient limitation, their low growth rate, and spatial distribution. At the early stage of the biofilm formation, the system is in NO_2^- limited condition. As the biofilm growth, the bottom area where NOBs reside turns to oxygen-limited condition. Both conditions inhibit NOBs growth.

biofilm which nutrients can penetrate. However, the concentration of substrate and NH_4^+ in the bulk liquid relax to steady state before the total biomass concentration relaxes (Fig 8(b)), as bulk concentrations are mainly determined by biomass in the top biofilm layers, which ensures a high growth rate [20]. The NO_2^- profile is influenced by both AOB synthesis and NOB consumption. Thus, the bulk concentration decreases with the increase of NOB populations.

Parallel performance. Parallel performance is crucial to IbM solvers for simulating large and complex problems. To investigate the performance of NUFEB, a scalability test was performed based on Case Study 2. The test aims at evaluating how simulation time varies with the number of processors for a fixed problem size. The *speed-up* of the scalability test is defined as t_{p0}/t_{pn}, where t_{p0} and t_{pn} are the actual CPU times spent on the baseline case and the test case respectively. Then the *efficiency* is the ratio between the speed-up of the baseline and the test cases obtained when using a given number of processors, i.e., $N_{p0}t_{p0}/N_{pn}t_{pn}$, where N_{p0} and N_{pn} are the number of processors employed in these cases.

In order to reach a considerable number of microbes (4 millions), the baseline case is performed using 4 processors, and the subsequent tests were performed with increasing numbers of processors, ranging from 8 to 256. As mentioned previously, NUFEB implements two distinct solutions for the parallelisation of continuum-based and particle-based processes. The performance of the two solutions are studied separately, and are presented in Fig 9. It can be observed that when employing a small number of processors, both solutions are close to the ideal speed-up (linear increase) and the parallel efficiency is over 90%. As more processor nodes are added, each processing node spends more time doing inter-processor communication than useful processing, and the parallelisation becomes less efficient. In the tests using 128 and 256 processors, the parallel efficiency decreases to around 42%, but it still shows a good scalability.
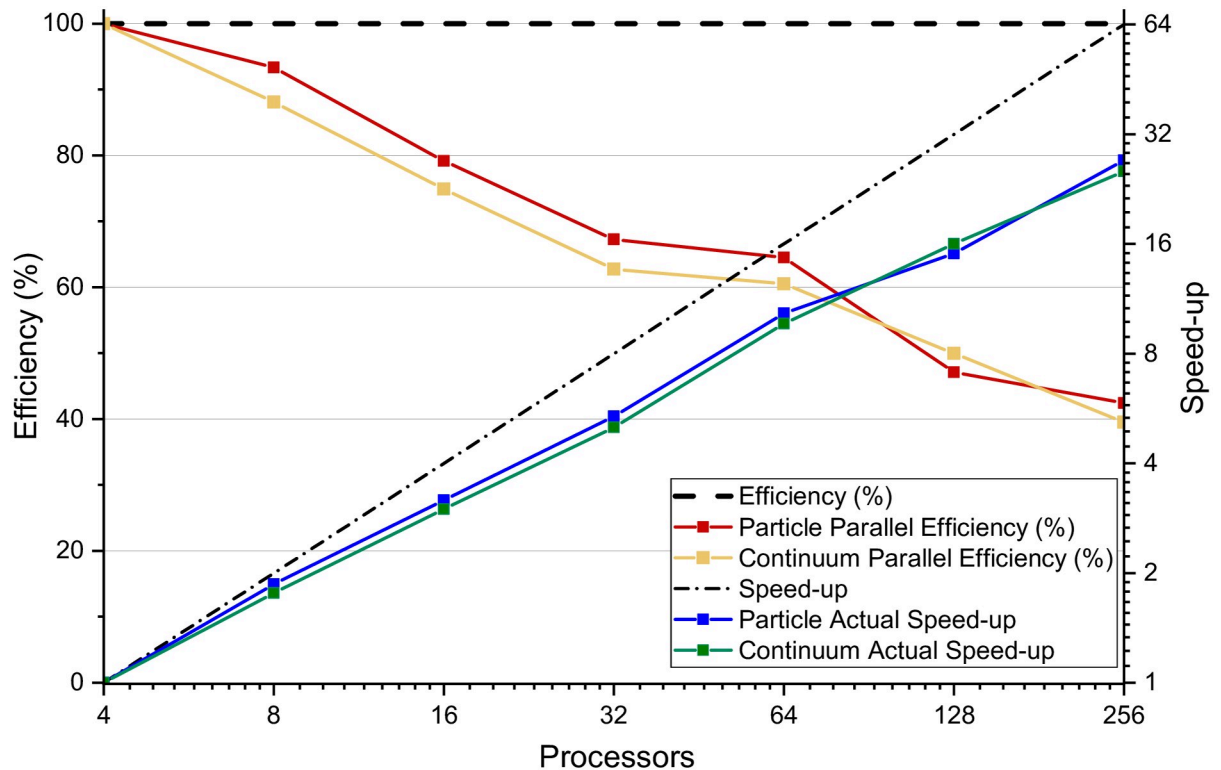
Fig 9. Performance of Case Study 2 with 4–256 processors. The initial conditions and the model parameters are kept the same in all cases. The average achieved particle number of all cases is 4.77×10^6 with 0.5% standard deviation. The difference is due to the randomness in sub-domains. The simulation with 4 processors is regarded as baseline case for computing speed-up. Doubling the numbers of processors results in almost double speed-up when small processor numbers are employed (e.g., less than 32). However, the speed-up and parallel efficiency decrease with increasing numbers of processors due to inter-processor communication.

https://doi.org/10.1371/journal.pcbi.1007125.g009

Availability and future directions

In this paper, we have presented the NUFEB tool for modelling and simulating Individual-based Models. The tool, documentation, and examples are publicly available on the GitHub repository: https://github.com/nufeb/NUFEB. To date, NUFEB has been adopted to model microbial communities in a variety of studies. In [17], we studied the influence of nutrient gradients on biofilm structure formation, and the influence of shear flow on growing biofilm. Bacteria twitching motility under fluid flow was investigated in [51]. We also successfully applied the tool to study the influence of thermodynamics and pH on microbial growth [37]. In [42] and [52], we used micro-scale NUFEB simulations to emulate the behaviour of biofilm/floc in the macro-scale. An emulation strategy for parameter calibration of NUFEB against iDynoMiCS has been applied in [53].

The main advantage of the 3D microbial system model is to study effects of spatial behaviours of microbial systems. Our ongoing work includes modelling biofilm streamer oscillation, modelling nitritation anammox system, and coupling microbial growth with fluid condition. The goal for NUFEB development in the future would be to deliver an even more general, efficient and user-friendly platform. This will include, for example, the development of an intuitive Graphical User Interface which will significantly improve the user experience. In order to make NUFEB available for power-efficient HPC architecture, we will also focus on a Kokkos port for the NUFEB code. This would allow the code to run on different kinds of hardware, such as GPUs (Graphics Processing Units), Intel Xeon Phis, or many-core CPUs.

The computational demands of IbM will always place a limit on the scale at which they can be applied. However, it is now evident that this limit can overcome by the use of statistical emulators [42]. A statistical emulator is a computationally efficient mimic of an IbM that can run thousands of times faster. In principle this new approach will allow the output of an IbM to be used at the metre scale and beyond and thus to make predictions about systems level performance in, for example, wastewater treatment or biofilm-fouling and drag on ships. This is a strategically important advance that creates a new impetus for the development of IbM that can credibly combine chemistry, mechanics, biology and hydrodynamics in a computationally efficient framework. NUFEB is, we believe, the first generation of IbM to meet that need and will help IbM transition from a research to application.

Supporting information

S1 File. Supporting information (SI).
(PDF)

S1 Manual. User manual.
(PDF)

S1 Chart. Chart data.
(XLSX)

S1 Video. Growth of a biofilm without flow. The biofilm is initially grown for 9 days without flow. It forms a mushroom-shaped structure in an oxygen-limited condition (1×10^{-4} kg m^{-3}).
(AVI)

S2 Video. Biofilm removal at U_f = 0.2 m s^{-1}. Large chunks of microbes detach from the biofilm surface and then remove from the systems. The biofilm morphology changes from a rough to a flat surface.
(MP4)

S3 Video. Biofilm removal at U_f = 0.1 m s^{-1}. The top of the biofilm is highly elongated. Small clusters erode from the deforming biofilm and the amount of biomass removed is less than the U_f = 0.2 m s^{-1} case.
(MP4)

S4 Video. Biofilm deformation and detachment with periodic domain boundary style at U_f = 0.2 m s^{-1}. Microbes crossing the boundary will re-appear on the opposite side of the domain. It can be observed that the detached clusters can re-attached to the biofilm surface or re-agglomerate with other clusters.
(MP4)

S5 Video. Growth of a large biofilm system. The multiple functional groups biofilm is grown without flow and reaches 2.3×10^7 particles after 160 hours (CPU time = 30 hours). The biofilm forms a wavy structure because of the high nutrient concentration environment and the non-uniform initial microbial distribution.
(MP4)

Acknowledgments

Author Contributions

Conceptualization: Jinju Chen, Andrew Stephen McGough, Irina Dana Ofiteru, Thomas P. Curtis, Paolo Zuliani.

Methodology: Bowen Li, Jayathilake Pahala Gedara, Valentina Gogulancea, Rebeca Gonzalez-Cabaleiro, Jinju Chen, Irina Dana Ofiteru, Thomas P. Curtis.

Software: Bowen Li, Denis Taniguchi.

Supervision: Jinju Chen, Andrew Stephen McGough, Irina Dana Ofiteru, Thomas P. Curtis, Paolo Zuliani.

Validation: Bowen Li, Jayathilake Pahala Gedara.

Visualization: Bowen Li, Denis Taniguchi.

Writing – original draft: Bowen Li.

Writing – review & editing: Denis Taniguchi, Jayathilake Pahala Gedara, Valentina Gogulancea, Rebeca Gonzalez-Cabaleiro, Jinju Chen, Andrew Stephen McGough, Irina Dana Ofiteru, Thomas P. Curtis, Paolo Zuliani.

References

1. Sheng G, Yu H, Li X. Extracellular polymeric substances (EPS) of microbial aggregates in biological wastewater treatment systems: a review. Biotechnology Advances. 2010; 28(6):882–894. https://doi.org/10.1016/j.biotechadv.2010.08.001. PMID: 20705128

2. Kaplovsky AJ. Activity of microorganisms in organic waste disposal. Applied and Environmental Microbiology. 1957; 5(3):175–179.

3. Rabaey K, Clauwaert P, Aelterman P, Verstraete W. Tubular microbial fuel cells for efficient electricity generation. Environmental Science & Technology. 2005; 39:8077–82. https://doi.org/10.1021/es050986i

4. Rodriguez-Brito B, Li L, Wegley L, Furlan M, Angly F, Breitbart M, et al. Viral and microbial community dynamics in four aquatic environments. The ISME Journal. 2010; 4:739–51. https://doi.org/10.1038/ismej.2010.1 PMID: 20147985

5. Battin T, Sloan WT, Kjelleberg S, Daims H, Head I, Curtis T, et al. OpinionMicrobial landscapes: new paths to biofilm research. Nature Reviews Microbiology. 2007; 5:76–81. https://doi.org/10.1038/nrmicro1556

6. Picioreanu C, van Loosdrecht M, Heijnen S. A new combined differential-discrete cellular automaton approach for biofilm modeling: application for growth in gel beads. Biotechnology and Bioengineering. 1998; 57:718–31. https://doi.org/10.1002/(SICI)1097-0290(19980320)57:6%3C718::AID-BIT9%3E3.0.CO;2-O PMID: 10099251

7. Laspidou C, Rittmann B. Modeling the development of biofilm density including active bacteria, inert biomass, and extracellular polymeric substances. Water Research. 2004; 38:3349–61. https://doi.org/10.1016/j.watres.2004.04.037 PMID: 15276752

8. Pizarro G, Griffeath D, Noguera D. Quantitative cellular automaton model for biofilms. Journal of Environmental Engineering-asce. 2001; 127. https://doi.org/10.1061/(ASCE)0733-9372(2001)127:9(782)

9. Kreft JU, Picioreanu C, Wimpenny JWT, van Loosdrecht M. Individual-based modelling of biofilms. Microbiology. 2001; 147:2897–2912. https://doi.org/10.1099/00221287-147-11-2897 PMID: 11700341

10. Kreft JU, Booth G, Wimpenny JWT. BacSim, a simulator for individual-based modelling of bacterial colony growth. Microbiology. 1998; 144(12):3275–3287. https://doi.org/10.1099/00221287-144-12-3275 PMID: 9884219

11. Hellweger FL, Clegg R, Clark J, Plugge C, Kreft JU. Advancing microbial sciences by individual-based modelling. Nature Reviews Microbiology. 2016; 14. https://doi.org/10.1038/nrmicro.2016.62

12. Plimpton S. Fast parallel algorithms for short-range molecular dynamics. Journal of Computational Physics. 1995; 117(1):1–19. https://doi.org/10.1006/jcph.1995.1039.

13. Lardon L, Merkey BV, Martins S, Dotsch A, Picioreanu C, Kreft JU, et al. iDynoMiCS: next-generation individual-based modelling of biofilms. Environmental Microbiology. 2011; 13:2416–34. https://doi.org/10.1111/j.1462-2920.2011.02414.x PMID: 21410622

14. Naylor J, Fellermann H, Ding Y, Mohammed W, Jakubovics N, Mukherjee J, et al. Simbiotics: a multi-scale integrative platform for 3D modeling of bacterial populations. ACS Synthetic Biology. 2017; 6. https://doi.org/10.1021/acssynbio.6b00315 PMID: 28475309

15. Gonzalez-de-Aledo P, Vladimirov A, Manca M, Baugh J, Asai R, Kaiser M, et al. An optimization approach for agent-based computational models of biological development. Advances in Engineering Software. 2018; 121:262–275. https://doi.org/10.1016/j.advengsoft.2018.03.010.

16. Rudge T, Steiner PJ, Phillips A, Haseloff J. Computational modeling of synthetic microbial biofilms. ACS Synthetic Biology. 2012; 1:345–52. https://doi.org/10.1021/sb300031n PMID: 23651288

17. Jayathilake PG, Gupta P, Li B, Madsen C, Oyebamiji OK, González-Cabaleiro R, et al. A mechanistic individual-based model of microbial communities. PLOS ONE. 2017; 12(8):1–26. https://doi.org/10.1371/journal.pone.0181965

18. Grimm V, Berger U, Bastiansen F, Eliassen S, Ginot V, Giske J, et al. A standard protocol for describing individual-based and agent-based models. Ecological Modelling. 2006; 198(1):115–126. https://doi.org/10.1016/j.ecolmodel.2006.04.023.

19. Ofiţeru I, Bellucci M, Picioreanu C, Lavric V, Curtis T. Multi-scale modelling of bioreactor-separator system for wastewater treatment with two-dimensional activated sludge floc dynamics. Water Research. 2013; 50. https://doi.org/10.1016/j.watres.2013.10.053 PMID: 24246170

20. Picioreanu C, Kreft JU, van Loosdrecht M. Particle-based multidimensional multispecies biofilm model. Applied and Environmental Microbiology. 2004; 70. https://doi.org/10.1128/AEM.70.5.3024-3040.2004 PMID: 15128564

21. Monod J. The growth of bacterial cultures. Annual Review of Microbiology. 1949; 3(1):371–394. https://doi.org/10.1146/annurev.mi.03.100149.002103

22. González-Cabaleiro R, Ofiţeru I, Lema J, Rodríguez R J. Microbial catabolic activities are naturally selected by metabolic energy harvest rate. The ISME Journal. 2015; 9. https://doi.org/10.1038/ismej.2015.69 PMID: 26161636

23. Kleerebezem R, van Loosdrecht M. A generalized method for thermodynamic state analysis of environmental systems. Critical Reviews in Environmental Science and Technology. 2010; 40. https://doi.org/10.1080/10643380802000974

24. Xavier J, Picioreanu C, van Loosdrecht M. A framework for multidimensional modelling of activity and structure of multispecies biofilms. Environmental Microbiology. 2005; 7:1085–103. https://doi.org/10.1111/j.1462-2920.2005.00787.x PMID: 16011747

25. Chambless JD, Stewart PS. A three-dimensional computer model analysis of three hypothetical biofilm detachment mechanisms. Biotechnology and Bioengineering. 2007; 97(6):1573–1584. https://doi.org/10.1002/bit.21363 PMID: 17274065

26. Brilliantov N, Spahn F, Hertzsch JM, Pöschel T. Model for collisions in granular gases. Physical Review E. 2002; 53.

27. Sun R, Xiao H, Sun H. Investigating the settling dynamics of cohesive silt particles with particle-resolving simulations. Advances in Water Resources. 2017.

28. Syamlal M, Rogers W, O'brien TJ, Syamlal M. MFIX documentation theory guide. USDOE Morgantown Energy Technology Center; 1993.

29. LAMMPS' user manual;. https://lammps.sandia.gov/doc/Manual.html.

30. Heinz H, Paul W, Binder K. Calculation of local pressure tensors in systems with many-body interactions. Phys Rev e. 2005; 72. https://doi.org/10.1103/PhysRevE.72.066704

31. Wanner O, Eberl H, Morgenroth E, Noguera D, Picioreanu C, Rittmann B, et al. Mathematical modeling of biofilms. vol. 18. IWA Scientific and Technical Report Series; 2006.

32. Gupta P. Verification and validation of a DEM-CFD model and multiscale modelling of cohesive fluidization regimes. University of Edinburgh; 2015.

33. Gupta P, Sun J, Ooi J. DEM-CFD simulation of a dense fluidized bed: wall boundary and particle size effects. Powder Technology. 2015; 293.

34. Anderson T, Jackson R. A fluid mechanical description of fluidized beds. Industrial & Engineering Chemistry Fundamentals. 1967; 6.

35. Sun R, Xiao H. Diffusion-based coarse graining in hybrid continuum–discrete solvers: applications in CFD–DEM. International Journal of Multiphase Flow. 2014; 72.

36. Sun R, Xiao H. Diffusion-based coarse graining in hybrid continuum–discrete solvers: theoretical formulation and a priori tests. International Journal of Multiphase Flow. 2014; 77.

37. Gogulancea V, González-Cabaleiro R, Li B, Taniguchi D, Jayathilake PG, Chen J, et al. Individual based model links thermodynamics, chemical speciation and environmental conditions to microbial

growth. Frontiers in Microbiology. 2019; 10:1871. https://doi.org/10.3389/fmicb.2019.01871 PMID: 31456784

38. Verlet L. Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. Physical Review. 1967; 159:98–103. https://doi.org/10.1103/PhysRev.159.98

39. Sun R, Xiao H. SediFoam: a general-purpose, open-source CFD-DEM solver for particle-laden flow with emphasis on sediment transport. Computers & Geosciences. 2016; 89.

40. Weller HG, Tabor G, Jasak H, Fureby C. A tensorial approach to computational continuum mechanics using object-oriented techniques. Computers in Physics. 1998; 12(6):620–631. https://doi.org/10.1063/1.168744

41. Issa R, Befrui B, Beshay K, Gosman AD. Solution of the implicitly discretized reacting flow equations by operator-splitting. Journal of Computational Physics. 1991; 93:388–410. https://doi.org/10.1016/0021-9991(91)90191-M

42. Oyebamiji OK, Wilkinson DJ, Jayathilake PG, Curtis T, Rushton SP, Li B, et al. Gaussian process emulation of an individual-based model simulation of microbial communities. Journal of Computational Science. 2017; 22. https://doi.org/10.1016/j.jocs.2017.08.006

43. Eberl H, van Loosdrecht M, Morgenroth E, Noguera D, Perez J, Picioreanu C, et al. Modelling a spatially heterogeneous biofilm and the bulk fluid: selected results from Benchmark Problem 2 (BM2). Water Science and Technology. 2004; 49:155–62. https://doi.org/10.2166/wst.2004.0829 PMID: 15303736

44. Noguera D, Picioreanu C. Results from the multi-species Benchmark Problem 3 (BM3) using two-dimensional models. Water Science and Technology. 2004; 49:169–76. https://doi.org/10.2166/wst.2004.0833 PMID: 15303738

45. Alpkvist E, Klapper I. Description of mechanical response including detachment using a novel particle model of biofilm/flow interaction. Water Science and Technology. 2007; 55 8-9:265–73. https://doi.org/10.2166/wst.2007.267 PMID: 17546995

46. Vo G, Brindle E, Heys J. An experimentally validated immersed boundary model of fluid-biofilm interaction. Water Science and Technology. 2010; 61:3033–40. https://doi.org/10.2166/wst.2010.213 PMID: 20555199

47. Stoodley P, Wilson S, Hall-Stoodley L, Boyle JD, Lappin-Scott HM, Costerton JW. Growth and detachment of cell clusters from mature mixed-species biofilms. Applied and Environmental Microbiology. 2001; 67(12):5608–5613. https://doi.org/10.1128/AEM.67.12.5608-5613.2001 PMID: 11722913

48. Larsen P, Nielsen JL, Svendsen TC, Nielsen PH. Adhesion characteristics of nitrifying bacteria in activated sludge. Water Research. 2008; 42(10):2814–2826. https://doi.org/10.1016/j.watres.2008.02.015. PMID: 18359056

49. Nogueira R, Elenter D, Brito A, Melo L, Wagner M, Morgenroth E. Evaluating heterotrophic growth in a nitrifying biofilm reactor using fluorescence in situ hybridization and mathematical modeling. Water Science and Technology. 2005; 52. https://doi.org/10.2166/wst.2005.0192

50. Klapper I, Dockery J. Finger formation in biofilm layers. SIAM Journal of Applied Mathematics. 2002; 62:853–869. https://doi.org/10.1137/S0036139900371709

51. Jayathilake PG, Li B, Zuliani P, Curtis T, Chen J. Modelling bacterial twitching in fluid flows: a CFD-DEM approach. Scientific Reports. 2019; 9. https://doi.org/10.1038/s41598-019-51101-3 PMID: 31601892

52. Oyebamiji OK, Wilkinson DJ, Jayathilake PG, Rushton S, Bridgens B, Li B, et al. A bayesian approach to modelling the impact of hydrodynamic shear stress on biofilm deformation. PLOS ONE. 2018; 13 (4):1–21. https://doi.org/10.1371/journal.pone.0195484

53. Oyebamiji OK, Wilkinson DJ, Li B, Jayathilake PG, Zuliani P, Curtis T. Bayesian emulation and calibration of an individual-based model of microbial communities. Journal of Computational Science. 2018; 30.