Implementation of Individual Based (IB) Models within LAMMPS to model biofilm dynamics in a waste water treatment plant

Prashant Gupta, Curtis Madsen, Jayathilake Pahala Gedara

July 22, 2015

Contents

List of Tables

List of Figures

1 Introduction

Water is one of the most essential commodities to sustain life. Even though 71 percent of the earth is water and 97 percent exists in form of oceans, only 0.3 percent of the water can be used by us. Water demands has been growing steadily with ever increasing world population. Furthermore, uneven human population distribution on earth adds to the water distribution problem. Usable and portable water scarcity is a major problem for many societies (citations). The problem is widely discussed and demands a judicial usage of available resources. However, several countries (mostly under-developed and developing) suffer from extreme shortage of the water resources. Waste water treatment plants have served the societies for several decades now. Activated sludge process treatment plants are the most common types as it provides an advantage of better efficiency and economics.

1.1 Activated Sludge process

Activated sludge is defined as a microbrial mass cultivated in order to break down the biomass into different components such as carbon dioxide, water and other nitrogen or phosphorous based compounds. Activated sludge process involves 3 different component mechanisms:

- A reactor in which the microbes are kept in suspension, aerated, and in contact with the waste they are treating.
- Settlement of bulkier solid (activated sludge).
- Recycling system for returning activated sludge back.

In present waste water industries, many variants of activated sludge processes, primarily variations on how the activated sludge is recycled. Present study attempts to model following activated sludge process as described in the figure 1.

1.2 Floc formation

A floc is described as an aggregate of microbes bonded together by adhesive material (EPS) secreted by them. These clusters formations, often called as activated sludge, play a key role in the functioning of any waste water treatment plant (WWTP). Efficiency or the energy requirement of either of the primary and secondary tanks in an activated sludge process, depends upon the floc size, settlement or motility. Furthermore, the floc formation is governed by the microbial components adherence and cluster breakages.

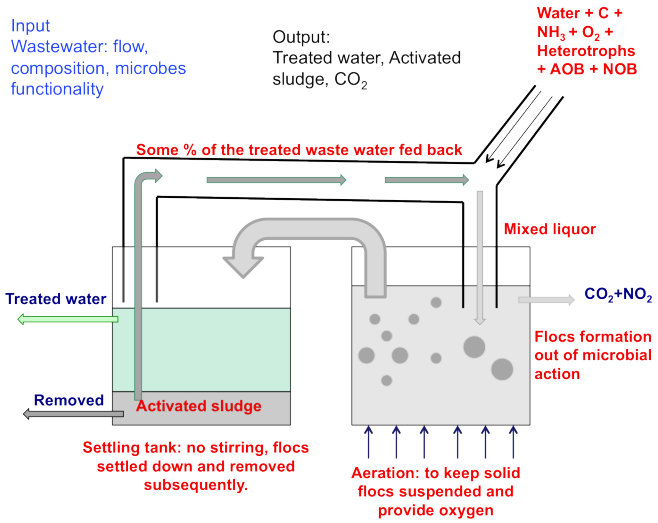

Figure 1: Activated sludge process description for a waste water treatment plant (WWTP)

1.2.1 Biological process

1.2.2 Physical process

2 Individual based modelling (IBm)

Pilot scale plants and laboratory scale experiments of WWTP are expensive, cumbersome, non-invasive and often can not provide information at the micro-scale, which is required for operational optimization of WWTP. Modelling of WWTP is challenging due to wide separation of temporal and spatial scales at which biological and physical processes key roles. Henceforth, a multi-scale modelling strategy is required to relate the microscopic microbial actions (often less than a micrometer size) to the macroscopic bulk WWTP operation (meter size scale). Identification of different length scales typical to a WWTP is presented can be shown in the figure 2.

2.1 Introduction

Discrete unit models such as the cellular automaton (Picioreanu et al., 1998a) or the individual-based models (Kreft et al., 1998), have been first developed and are now becoming widely applied to study effects of spatially multidimensional gradients in biofilms. Models which use individuals as a basic unit have occasionally been used in ecology since 1970s, but only since the visionary review of Huston et al. (1988) has individual-base modelling been an explicitly delineated approach of ecological modelling. Individual-based

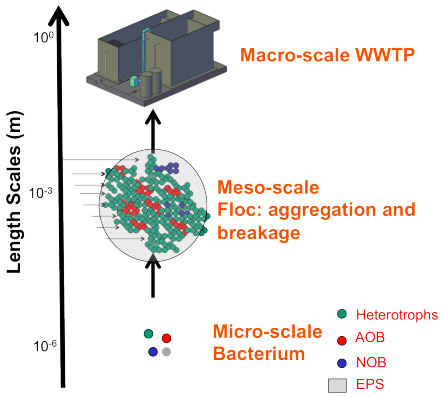

Figure 2: Schematic of typical length scales for multi-scale modelling of a activated sludge process based Waste water treatment plant (WWTP)

modelling refers in the following to simulation models that treat individuals as unique and discrete entities which have at least one property in addition to age that changes during the life cycle, e.g. weight, rank in a social hierarchy, etc. For biofilm modelling, Kreft et al. (1998, 2001) initially proposed the Individual-based Modelling (IbM) as a proper tool compared with existing Cellular Automaton (CA) method used for biofilm modelling. In CA models, the biofilm is represented by collection of bricks, in which each brick is represented by a Cartesian grid (Picioreanu et al., 1998a, 1998b,2000a; Knutson et al., 2005; Tang and Valocchi, 2013, Radu et al. 2010). In IbM, bacterial cells are represented as hard spheres, with each cell having, besides a variable volume and mass, a set of variable growth parameters, position, velocity, genotype etc. The IbM model consists of two parts: one deals with the growth and behaviour of individual bacteria as autonomous agents (i.e., biological processes); the other deals with the substrate and product diffusion and reaction and fluid flow (i.e., physical processes). Each cell grows by consuming the substrate and divides when a certain volume or mass is reached. The pressure build-up due to the growth of biomass is released by maintenance of a minimum distance between the neighbouring cells. In the traditional IbM approach, for each cell, the vector sum of all positive overlap radii with the neighbouring cells, which is called the shoving vector, is calculated and then the position of the cell is shifted in the direction opposite to this vector 3 sum. The substrate concentration is governed by a convection-diffusion-reaction equation and this transport equation is solved in a fixed Cartesian grid. Figure 1 shows the typical computation domain associated with IbM of biofilms. Basically it has three sub-domains each for biofilm, mass transfer boundary layer, and bulk fluid. Numerical simulations in 2D showed that the IbM produced a more confluent and rounded biofilm struc-

tures than the CA based models, due to its deterministic and directionally unconstrained spreading of the biomass (Wang and Zhang, 2010).

2.2 Model description

2.3 Purpose of the model

The model treats each microbes as individual entities interacting with each other and the surrounding environmental conditions. Various sub-models are employed for describing biology and physics at different length and temporal scales. The multi-physics/biological model developed can be used to investigate following key questions at different scales:

- Linking the microscopic (bacterium) to the mesoscopic (floc) to the macroscopic bulk operational parameters.
- Floc formation mechanism: How the microscopic phenomenon such as microbial growth kinetics, division, death, nutrient availability lead to microbial floc formation or breakage?
- Effects of floc morphology and motility on the overall plant efficiency?
- How the hydrodynamics of the system can influence energy requirements of the WWTP?
- Effect of the nutrient availability spatially and temporally?

2.4 Model variables and principles

The solver treats the microbes as spherical particles with EPS shells bounded on them. Fluid is treated as a continuum. Nutrients are treated as concentrations in dissolved and convected through the flow fields. Given by their representations each components are solved accordingly.

- Microbes: solved in a Lagragian framework. Each microbes is represented as a finite size sphere and has certain functionality. They are tracked throughout the simulations. The interactions between the microbes and with their ambient environment are complex and are described through different biological and physical sub-models described later. Following functional groups are considered: Heterotrophs (HET), Ammonia oxidizing bacteria (AOB), Nitrogen oxidizing bacteria (NOB), EPS particles and dead inert particles.
- Fluid: Fluid or the water phase is treated as a continuum media with microbes mostly suspended (neutrally buoyant). The fluid motion is solved in an Eulerian grid and takes into the account of the microbial suspension. Fluid and particle motion are tightly coupled by the hydrodynamics solver.

- Nutrients: The nutrients are treated as concentrations of the food supply in the water phase. The fluid solver supplies these nutrients to the microbes and hence contributes to their metabolism. Following nutrients relevant to the waste water treatment plant are taken into the account: Carbon substrates (S_s), Oxygen supply (O_2), Nitrates (NO_2), Nitrites (NO_3) and Ammonia (NH_4).

2.5 IBm submodels

Various cell level processes constitutes microbial working at cell level. Each of these processes can be termed as components of IBm model presented here. Modelling and fundamentals of each of these processes are presented here:

2.5.1 Cell Growth

Commonly found microbes in a WWTP can be grouped into functional group: Heterotrophs (HET), Ammonia oxidizing bacteria (AOB) and Nitrogen oxidizing bacteria (BOB). Each of the functional groups consume different nutrients and compete with each other for survival. Present tool allows a functional group and its function to be defined accordingly. Following equations give the reactions and their description is presented in a matrix form (3):

$$\frac{dm_i}{dt} = Gm_i \tag{1}$$

Each species has its own growth rate (G) and can be described according to the equation 2. $\gamma_{HET} = 1$ and rest are zero, when solving for the HET equations. Same can be extended to AOB, NOB and EPS particles. R_j (j=1-9) are given in the table 3.

$$G = \gamma_{HET}\,(R_1 + R_4 + R_5) + \gamma_{AOB}\,(R_2) + \gamma_{NOB}\,(R_3) - \gamma_{EPS}\,(R_9) \tag{2}$$

HET bound EPS growth rate is given by:

$$G_{HET-EPS} = \frac{Y_{EPS}}{Y_{HET}} Gm_i \tag{3}$$

Reaction equations are primarily mass balance solved on the mesh discretization defined earlier.

Process	Rate (d^{-1})	
Aerobic growth of HET	$\mu_{m,HET} \frac{S_S}{K_{S,HET}+S_S} \frac{S_{O2}}{K_{O2,HET}+S_{O2}}.$	R_1
Aerobic growth of AOB	$\mu_{m,AOB} \frac{S_{NH4}}{K_{NH4,AOB}+S_{NH4}} \frac{S_{O2}}{K_{O2,AOB}+S_{O2}}.$	R_2
Aerobic growth of NOB	$\mu_{m,NOB} \frac{S_{NO2}}{K_{NO2,NOB}+S_{NO2}} \frac{S_{O2}}{K_{O2,NOB}+S_{O2}}.$	R_3
Anoxic growth of HET on NO_3	$\eta_H \mu_{m,HET} \frac{S_S}{K_{S,HET}+S_S} \frac{S_{NO3}}{K_{NO3,HET}+S_{NO3}} . \frac{K_{O2,HET}}{K_{O2,HET}+S_{O2}}$	R_4
Anoxic growth of HET on NO_2	$\eta_H \mu_{m,HET} \frac{S_S}{K_{S,HET}+S_S} \frac{S_{NO2}}{K_{NO2,HET}+S_{NO2}} . \frac{K_{O2,HET}}{K_{O2,HET}+S_{O2}}$	R_5
Decay of HET	$b_{HET}.$	R_6
Decay of AOB	$b_{AOB}.$	R_7
Decay of NOB	$b_{NOB}.$	R_8
Decay of EPS	b_{EPS}	R_9

Figure 3: Growth and decay rates for AOB, NOB, HET and EPS particles.

Each of the components of uptake or consumption is calculated separately and added together following superposition principle. The rate is defined according to Monod-kinetics where the kinetics factors are product together. Also, the rate of the growth/decay is proportional to the instantaneous mass of the substrate. The local nutrient availability for the growth of each cell is based upon the ambient conditions solved through the passive scalar transport of the nutrients. Fluid-biofilm coupling solver ensures that each cell is aware of its ambient nutrient conditions.Each of the cell has nutrient availability concentration attributes as: Soluble substrate (Ss), Oxygen (O_2), NO2, NO3, NH4.

2.5.2 Cell Division

Microbes can not grow forever (even with infinite food supply), division occurs due to sustenance of their sizes. Present model employs cell division rather empirically, by fixing up a maximum cell diameter, after which it divides. The model is implemented in following way: If the mass of a bacterial cell becomes greater than twice the mass of an inoculated individual bacterium, it divides into two daughter cells each. During the division process, the cell mass is split in a ratio randomly selected between 0.4-0.6. This generated two daughter cells from a parent cell. These daughter cells are oriented randomly around the centre of the parent cell. Figure 4 shows the division of daughter cells from the mother cell, randomly oriented around the centre of mother cell. This generation of newer particles occur at start of every time step and the forces equilibrated according to the overlaps, ensuring mechanical stability. The diameters of daughter cells are calculated

using the cell mass and biomass density. Biomass density is constant for each species.

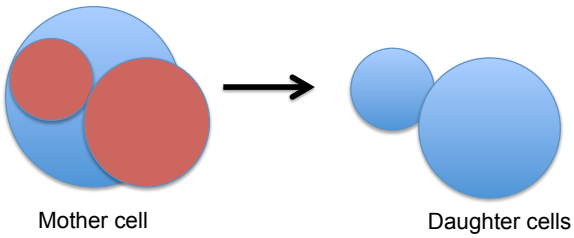

Figure 4: Division of daughter cells from the mother cell, randomly oriented around the centre of mother cell

2.5.3 Cell Death

Microbial death is modelled as a random stochastic process. Every so often, a few microbes becomes biologically inactive i.e. they would not consume nutrient, grow or divide. These cells are treated as inactive particles by the solver and only participate in any physical/mechanical interactions with their neighbours. Depending on the total decay mass of HET, AOB, and NOB, the appropriate numbers of HET, AOB, and NOB cells are calculated and died. The mass of the dying cell shares between dead cell and carbon source. The mass of decaying EPS cells totally converts to carbon. Completely dissolved EPS particles are removed from the calculation. In the model this implemented by the total virtual dead mass of each species (HET or AOB or NOB) and is calculated from equations as below:

$$D = \gamma_{HET} R_6 + \gamma_{AOB} R_7 + \gamma_{NOB} R_8 \tag{4}$$

$$M_{virtual} = \sum Dm \tag{5}$$

where, D is the decay rate. At each time step (dt), number of cells equivalent to $M_{virtual}$ is set as dead (i.e. ID is changed to 5). These particles are randomly selected. When a cell is died, Y fraction of its mass remains in the dead cell and the other fraction (1-Y) is spontaneously converted to carbon substrate (S_s) and distributed to the Cartesian grid cell centre in which the particle resides.

2.5.4 EPS production and excretion

Microbes secrete extracellular polymeric substances (EPS) every so often as a waste product of their metabolic activities. EPS is secreted into their neighbouring environment and have known to lend structural integrity to

the biofilms. Previous solvers have represented EPS in continuum and discrete manner. The solver works on the common knowledge that HETs excrete EPS, while AOB and NOB microbes do not. The present solver follows the iDynnomics approach of EPS treatment. Initially, EPS is accumulated as a extra shell beyond the HET particle (figure 5). It should be noted that the EPS density is much lower than the HET density. When the relative thickness of the EPS shell bound to HET particle exceeds a certain threshold value, almost half (random ratio between 0.4-0.6) of the EPS mass excretes as a separate EPS particle and positions next to the HET cell.

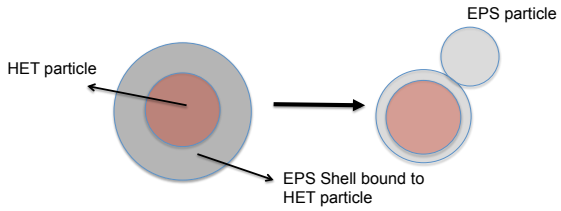

Figure 5: EPS bound to the the HET particle, extracted to an EPS particle according to the ratio between the EPS shell and the HET particle radius.

2.5.5 EPS-Adhesion forces

The excreted EPS mass from the HET particles can be employed as a parameter of adhesion force models between the particles. The EPS link between the particles are treated as much more stiffer springs, but only employing the attractive forces. Total effective EPS mass is calculated between the microbes ($Meps_{eff}$) and a spring stiffness is defined per unit mass (k_{eps}). Effective overlap between the EPS links are calculated and the forces calculated according to the effective spring stiffness ($Meps_{eff}$*k_{eps}) multiplied by the overlap. This approach was first applied by Head et al. to reproduce a mechanically stable biofilm.

2.6 Stochastic processes: Scope of improvement

Different processes are modelled by the sub-models as stochastic processes. This adds variability and randomness to the biological process, typically evident in the nature. Following processes are random:

- Initialization of the system by placing the microbes with random physical and biological attributes are placed at random locations within the domain. This introduces the randomness of the initial configuration.
- Cell division criterion is based on a critical random maximum ratio, which can be selected by the user. Solver further divides the daughter

cell masses randomly in a ratio between 0.4-0.6 placing daughter cells in a random orientation.

- During the death sub-model, microbes equivalent to the dead-mass are randomly chosen by the solver.
- EPS excretion occurs after the shell bound EPS exceeds a randomly selected value by the user. This excreted EPS is oriented and placed randomly around the mother cell.

As recommended by the iDynomics solver, these processes can be tackled from a microscopic point of view with fundamental chemical and biological processes driving these mechanisms.

3 LAMMPS

3.1 Introduction to LAMMPS

LAMMPS is a classical molecular dynamics code developed at Sandia labs and primarily built to solve the particle physics including wide range of inter-particle interactions and potentials. The code treats each particle as an individual discrete unit, much similar to the popular IB approach. Sandia Labs distributes LAMMPS under the terms of the GNU Public License (http://lammps.sandia.gov/). The current version of the code is written in C++ with an open architecture and provides an opportunity to couple with other open-source codes. LAMMPS can run efficiently in both serial and parallel versions depending upon the computational facilities available to the users. The LAMMPS code is designed to modify and extend it with newer capabilities as desired by the user. While only 25 percent of the 140K line code in LAMMPS forms the core of the solver, rest of the code is contributed by a large user database across the globe in order to extend its capabilities. An overview can of current LAMMPS capabilities can be found at LAMMPS-feature.

3.2 LAMMPS working methodology

LAMMPS solves the motion of every single particle by simply integrating Newton's equations of motion in response to sum of the forces (short or long range based on their interaction with neighbours). At a particular time instance, motion of each particle is collectively solved when subjected to initial or boundary conditions. In order to maintain computational tractability while calculating the interaction forces, LAMMPS maintains a neighbourhood list for each particle which gets updated every so often. These lists are optimized so that local densities and particle overlaps never becomes

non-physical. For parallel simulations, LAMMPS spatially partition the domain into smaller sub-domains assigned to each processors. Interprocessor communications are maintained by storing ghost atom interactions with the sub-domain boundaries. LAMMPS development can be helped by two user manuals: User manual and developer manual. The following links will be helpful for the users to get started on LAMMPS:

1. User manual: `http://lammps.sandia.gov/doc/Manual.pdf`
2. Developers guide: `http://lammps.sandia.gov/doc/Developer.pdf`
3. Tutorials: `http://lammps.sandia.gov/tutorials.html`
4. Commands: `http://lammps.sandia.gov/doc/Section_commands.html`
5. Features: `http://lammps.sandia.gov/features.html`

In the present study, LAMMPS-Feb14 version is developed and newer IB features and capabilities added, this version will be now on referred as LAMMPS-NCL.

3.3 Operating systems

In general, LAMMPS can be run on Windows, Linux, Mac OS using prebuilt executables. LAMMPS-NCL can be compiled with almost any linux or Mac OS (instructions in the user manual). It is emphasized that present NCL version has been rigorously tested on Ubuntun-14.10 and Fedora-22. In near future, prebuilt executables,binaries or RPMS will be provided to be used on any OS.

3.4 Pre-compilation instructions

Before compiling LAMMPS, please make sure you are installed with these packages depending upon the operating system used:

- fftw (http://www.fftw.org/doc/Installation-on-Unix.html)
- openmpi (http://www.open-mpi.org/)
- libjpegm (http://libjpeg-turbo.virtualgl.org/)
- gcc/g++ (https://help.ubuntu.com/community/InstallingCompilers)

4 LAMMPS-NCL and getting started

For the pre-existing LAMMPS commands, features and documentation, please refer to the LAMMPS user manual, listed above. The manual covers extensive instructions on compiling LAMMPS and how to get started. LAMMPS-NCL is compiled the same way as you would compile LAMMPS and there is no change in those instructions. The newer capabilities and commands will be highlighted and emphasized in the next sections and the sample input script.

4.1 Downloading LAMMPS-NCL

In general, different versions of LAMMPS can be downloaded from the download section of the LAMMPS website. For the present study, LAMMPS version Feb 2014 was developed and enhanced with biological IB model capabilities. It is advised to use the version developed at NCL. It can be downloaded using following set of instructions from the NUFEB Github repository.

4.2 LAMMPS extension

In the present study, capabilities of LAMMPS have been extended to include cell level biological processes such as cell growth, division, maintenance, death etc. These extended capabilities involve addition of newer particles at each time-steps e.g. during a cell division: a single parent cell divide into two daughter cells. A summary of list of newer capabilities added to the LAMMPS and its implementation is presented here:

4.3 Atom type

Classical LAMMPS provide different atom types that could be used by user. These are specified in the input script by the command: *atom – style*. *atom – style* command must be used before a simulation is setup via a read-data, read-restart or create-box command. A newer *atom – style* is added (named "bio") to increase the number of attributes such as nutrients available to each microbe type. The new *atom – style* is inherited from already existing *atom – style* sphere. Though LAMMPS provides another route for adding newer attributes through fix property/atom. However, since the newer attributes would be used to couple with the fluid solver, they should be more tightly linked to the atom style.

The new atom type "Bio" have following attributes in conjunction to the sphere type: ID, type, radius, outerradius, mass, outermass,x,y,z, S_s, O_2, NO_3, NO_2, NH_4. As a general rules following types are assigned to the functional groups.

- 1-HET

- 2-AOB
- 3-NOB
- 4-EPS
- 5-Inert

Each microbe has unique moniker ID but share its type according to the functional group. Physical parameters such as initial radius, outer radius, density and outer density can be mentioned. Radius and density are the radius and density of the microbe. Outer radius and densities are the initial EPS-shell bound radius and densities. Density is constant throughout the simulation and time invariant. However radius and mass (calculated from density and radius) can change. EPS is only bound to the HET particles (type 1), hence outer radius should be specified as radius for other particle types initially. LAMMPS will give an error message if this is not followed. For LAMMPS book-keeping, inner radius is always same as outer-radius for other types than 1 throughout the simulation. Initial configuration can be either generated randomly through LAMMPS or by the user. Caution must be taken to avoid overlaps when defined by the user. A typical user input script is included in the folder (IC5nut.in).

Different sub-models are coded within LAMMPS in a modular way. Each of these models, implementations and command lines are explained in the following sub-sections.

5 Input script generated for the NUFEB-Bio simulations

In order to execute LAMMPS commands, an input script is usually prepared with certain sub-commands and parameters list. Figure 6 gives the input script for the NUFEB project. This script can be generated by GUI and will be explained later. New additions to the LAMMPS commands are pointed in the boxes as shown in the figure.

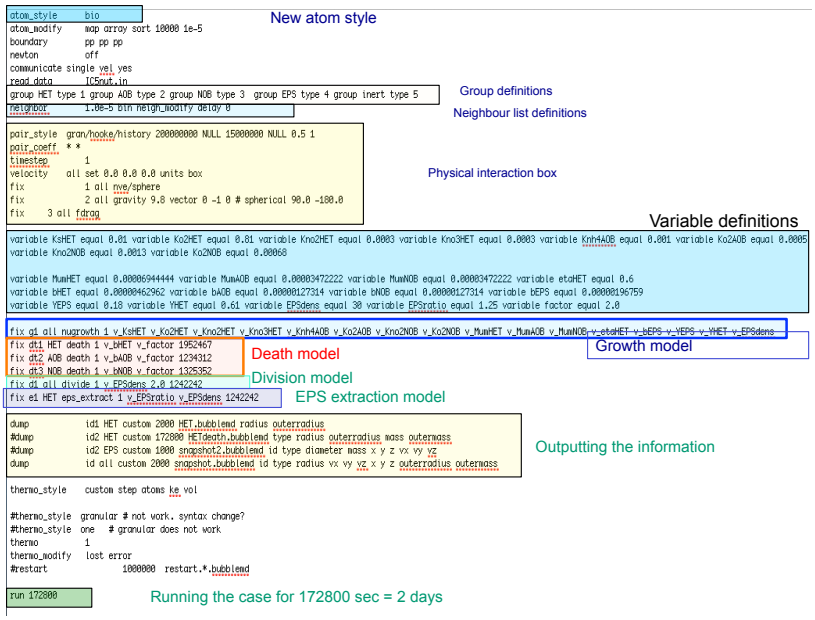

Figure 6: Input script for NUFEB simulation of biofilm dynamics.

5.1 Cell growth

Cell growth model described in the section 2.5.1 is implemented as a fix type "growth" in the LAMMPS, the source code can be found in the src folder of the LAMMPS directory. In the present implementation, GUI will help to generate LAMMPS command line for the fix growth and user just have to make sure to put in correct parameters. GUI is described later. Figure 7 gives the LAMMPS command line for the growth model.

```
fix g1 all nugrowth 1 v_KsHET v_Ko2HET v_Kno2HET v_Kno3HET v_KnH4AOB v_Ko2AOB v_Kno2NOB v_Ko2NOB v_MumHET v_MumAOB v_MumNOB v_etaHET v_bEPS v_YEPS v_YHET v_EPSdens
```

Figure 7: Command line for the growth fix in LAMMPS

In total, 20 parameters are given for the growth fix command. Specifically, "fix g1 all nugrowth 1" means invoke a fix nugrowth instance named g1 and apply it at everytimestep (1 = frequency of call) to "all" the particle types in the assembly. The second parameter "all" indicates that all the microbes can grow. Any other microbial type could also be mentioned, which would indicate that growth only occurs only for that specific type. All of the parameters has to be in SI units. A table of the typical parameters can be seen at the end of the documentation. The parameter description:

- fixID: unique ID of the fix (not used in the script before)

- group: Functional groups on which fix will be invoked, it can take: HET, AOB, NOB, all.
- fix-name: nugrowth (fix name). It can not take another value.
- freq: frequency at which the fix is invoked (typically at every timestep). Values has to be integers.
- KsHET Ko2HET Kno2HET Kno3HET Knh4AOB Ko2AOB Kno2NOB Ko2NOB: Affinity of all the functional groups to the substrate.
- MumHET MumAOB MumNOB: Monod kinetic parameters for the HET, AOB and NOB.
- etaHET: Reduction factor in anoxic conditions for HET.
- bEPS: Decay rate of EPS.
- YEPS, YHET: Yield coefficient for EPS and HET particles.
- EPS-density: Density of the EPS, required for the divisions of HET particles. Values can be floating type number.

5.2 Cell Division

Cell division model described in the section 2.5.2 is implemented as a fix type "division" in the LAMMPS, the source code can be found in the src folder of the LAMMPS directory. The command line is given in the figure 8:

```
fix d1 all divide 1 v_EPSdens 2.0 1242242
```

Figure 8: Command line for the division fix in LAMMPS

The command line follows the template: "fix fixID group fix-name freq EPS-density Ratio seed". The parameter description:

- fixID: unique ID of the fix (not used in the script before)
- group: Functional groups on which fix will be invoked, it can take: HET, AOB, NOB, all.
- fix-name: divide (fix name). It can not take another value.
- freq: frequency at which the fix is invoked (typically at every timestep). Values has to be integers.

- EPS-density: Density of the EPS, required for the divisions of HET particles. Values can be floating type number.
- Ratio: User defined diameter ratio at which the division takes place. The ratio is calculated as the instantaneous diameter divided by the average diameter of the microbes. The average diameter is initially defined and hard coded. The values have been indicated in the earlier sections. Values typically floating type number greater than 1.0.
- Seed: Seed for the orientation and introduce the randomness, very large integer values. Same seed will result in the same orientation for every simulation run.

5.3 Cell Death

Cell death model described in the section 2.5.3 is implemented as a fix type "death" in the LAMMPS, the source code can be found in the src folder of the LAMMPS directory. Figure 9 gives the specific command lines.

```
fix dt1 HET death 1 v_bHET v_factor 1952467
fix dt2 AOB death 1 v_bAOB v_factor 1234312
fix dt3 NOB death 1 v_bNOB v_factor 1325352
```

Figure 9: Command line for the death fix in LAMMPS

The command line follows the template: "fix fix-ID group fix-name freq decay-rate ratio seed". The parameter description:

- fixID: unique ID of the fix (not used in the script before)
- group: Functional groups on which fix will be invoked, it can take: HET, AOB, NOB, all.
- fix-name: death (fix name). It can not take another value.
- freq: frequency at which the fix is invoked (typically at every timestep). Values has to be integers.
- decay-rate: Rate of decay, according to the type of microbe.
- Ratio: Death ratio, described earlier in the model, typically floating number greater than 1.0.
- Seed: Seed for randomly choosing particles to kill.

5.4 EPS extract

EPS extraction model described in the section 2.5.4 implemented as a fix type "EPS extract" in the LAMMPS, the source code can be found in the src folder of the LAMMPS directory. Figure 10 gives the specific command lines.

```
fix e1 HET eps_extract 1 v_EPSratio v_EPSdens 1242242
```

Figure 10: Command line for the EPS extract fix in LAMMPS

The command line follows the template: "fix fix-ID group epsextract freq EPSratio EPSdens seed". The parameter description:

- fixID: unique ID of the fix (not used in the script before)
- group: Functional groups on which fix will be invoked, it can take: HET, AOB, NOB, all.
- fix-name: death (fix name). It can not take another value.
- freq: frequency at which the fix is invoked (typically at every timestep). Values has to be integers.
- EPS ratio: Ratio between outer-radius and inner-radius of the microbe. Value is typically floating number greater than 1.0.
- EPS-density: Density of the EPS, required for the divisions of HET particles. Values can be floating type number.
- Seed: Seed for the orientation and introduce the randomness, very large integer values. Same seed will result in the same orientation for every simulation run.

6 How to run a case

Describe GUI.