

Rapport de projet UP-09

Simulation d'interactions au sein d'un biofilm bactérien.

Raphaël RUBRICE

May 2024

1 Introduction

Depuis l'existence de la science moderne, la construction de modèles a été un objectif fondamental. Ces modèles servent d'outils essentiels pour comprendre des phénomènes complexes, nous permettant de disséquer des systèmes intriqués et parfois de prédire leur comportement. Dans le contexte des communautés microbiennes, les biofilms se distinguent comme des exemples captivants de vie microbienne collective. Ces biofilms, qui se composent de microorganismes densément agglomérés adhérant aux surfaces, présentent une résilience et une adaptabilité remarquables. Ils prospèrent dans des environnements variés, des écosystèmes naturels aux cadres industriels, et jouent des rôles pivots dans des processus tels que la biodégradation, le traitement des eaux usées et les infections médicales [1].

Complexité des Biofilms

Les biofilms ne sont pas de simples assemblages de microbes individuels ; ils représentent plutôt des écosystèmes complexes avec des propriétés émergentes. Au sein de ces communautés microbiennes, des interactions complexes se produisent — allant de la symbiose coopérative à l'antagonisme compétitif. La matrice du biofilm, composée de substances polymères extracellulaires (EPS), fournit une intégrité structurelle et facilite la communication entre les microorganismes résidents. Cette matrice agit comme un bouclier protecteur,

protégeant les habitants des menaces externes telles que les antibiotiques, les réponses immunitaires et les fluctuations environnementales. En conséquence, les biofilms présentent une résilience contre les conditions adverses, les rendant difficiles à éradiquer [1],[2].

Interactions Microbiennes

Comprendre la dynamique des interactions microbiennes au sein des biofilms est crucial. Ces interactions peuvent être coopératives, où les microorganismes collaborent pour exploiter les ressources ou améliorer la survie. Inversement, elles peuvent être compétitives, conduisant à la compétition pour les ressources, la prédation ou la production de composés inhibiteurs. L'équilibre entre les interactions coopératives et compétitives façonne le comportement des biofilms, influençant des aspects critiques tels que les taux de croissance, la distribution spatiale et la stabilité de la communauté. Décrypter ces relations complexes est essentiel pour développer des stratégies efficaces de gestion des biofilms.

Défis de la Recherche

Les méthodes traditionnelles d'étude des biofilms ont des limites. Les approches expérimentales, telles que la culture et la microscopie, fournissent des aperçus précieux mais manquent souvent de scalabilité et ne parviennent pas à saisir la pleine complexité de la dynamique des biofilms. De plus,

l'hétérogénéité des biofilms - à la fois spatialement et temporellement - pose des défis pour une caractérisation précise. C'est pour ces raisons que depuis plusieurs décennies, tout un pan de la recherche en biofilm se concentre sur la création de modèles permettant l'explication de la formation mais aussi de l'évolution spatio-temporelle des biofilms.

Modélisation de biofilms

La réalisation de modèles de biofilms est complexe puisque la nature même de ces communautés l'est et que nos connaissances à leurs sujets, bien que grandissantes, laissent encore à désirer. Lors de la construction de tels outils de modélisation, un grand nombre de phénomènes peuvent être pris en compte : les phénomènes chimiques, les phénomènes biologiques (croissance, compétition, coopération, mouvement cellulaire) et ceux physiques (diffusion, impact structural).

Ces divers phénomènes sont difficiles à modéliser dans le cas des biofilms car ils sont tous largement interdépendants. En effet, la formation d'un biofilm bactérien est hautement dépendante des conditions du milieu et de la, ou des souches bactériennes présentes. L'activité de chaque cellule impacte la forme, l'état et l'évolution du biofilm. Ainsi une des approches les plus répandue consiste en la création d'un automate cellulaire dont les règles dépendent de lois approchant les phénomènes que l'on souhaite incorporer dans le modèle. Cette approche est particulièrement adaptée car en définissant des règles à l'échelle unitaire, de nombreux phénomènes peuvent émerger. De plus, elle est relativement simple à implémenter ce qui en fait un outil très pertinent et largement modulable. C'est cette approche que j'ai choisi de poursuivre pour ce projet.

2 Méthodes et Modèle

Tout d'abord je tiens à rappeler ce qu'est un automate cellulaire. Il s'agit d'un algorithme dans lequel chaque case de l'espace ou de la grille utilisée respecte un ensemble de règles qui dictent son évolution. Ces règles sont calculées à partir du voisinage de chaque cellule à traiter. Le type de voisinage utilisé dans le modèle est défini au préalable et deux type de voisinages sont classiquement utilisés, à savoir le voisinage de VonNeuman ou celui de Moore dont une représentation est montrée en [Figure 1].

J'ai voulu réaliser ce projet sur ce sujet car j'ai pu travailler à divers reprises sur cette thématique avec messieurs Romain Briandet et Virgile Guéneau de l'équipe B3D de l'Institut Micalis. Plus particulièrement, au delà de la modélisation de la formation de biofilms, l'intérêt est ici de construire un modèle capable de rendre compte d'interactions entre bactéries de souches différentes. Pour se faire je me suis très largement appuyé sur le travail présenté dans [3].

Modèle utilisé

L'article précédemment cité présente une méthode d'automate cellulaire pour la simulation de formation de biofilm en incorporant la diffusion et consommation de nutriments, la division cellulaire et le partage de biomasse entre cellule voisine. Pour modéliser les phénomènes de diffusion, [3] propose la Loi de Fick, dont voici l'équation aux dérivées partielles suivi de son approximation numérique :

$$\frac{\partial P_X}{\partial t} = D_X \left(\frac{\partial^2 P_X}{\partial x^2} + \frac{\partial^2 P_X}{\partial y^2} + \frac{\partial^2 P_X}{\partial z^2} \right)$$

$$\frac{P_X(i, j)_{t+\Delta t} - P_X(i, j)_t}{\Delta t} = D_X \left(\sum_{m=1}^M P_X(m)_t - M P_X(i, j)_t \right)$$

- P_X : Concentration en nutriment X [$kg.m^{-3}$]
- D_X : Constante de diffusion du nutriment X [$m^2.s^{-1}$]

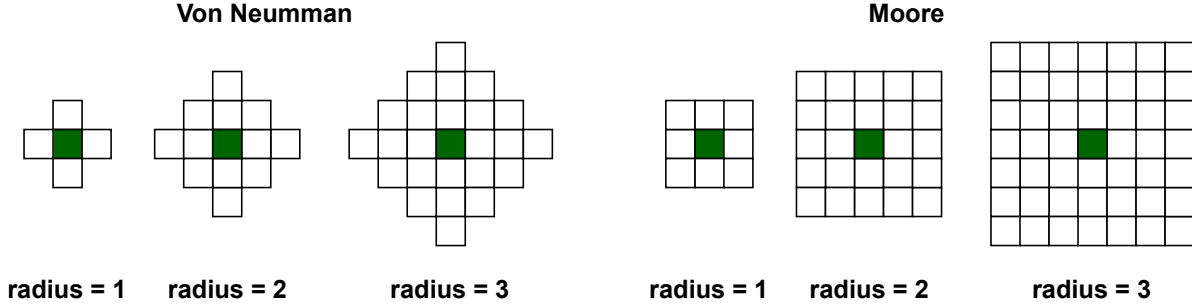


Figure 1: Types classiques de voisinage. Figure créée grâce au logiciel draw.io.

- M : Nombre total de voisins, m est le m -ème voisin.
- K_X : Constante de demi saturation du nutriment X. $[kg.m^{-3}]$

Pour ce modèle, les auteurs font l'hypothèse que la partie de l'environnement dont on simule la diffusion est assez éloignée des bords pour simplifier l'estimation de la diffusion dans l'espace-temps. Cependant il est important de noter que les simulations seront réalisées ici avec une condition de Neumann sur l'ensemble des bordures du domaine de simulation (flux nul sur toutes bordures ce qui correspond à de vraies parois). Cette contradiction n'a toutefois que peu d'impact sur les résultats ce qui justifie pourquoi nous conservons ces conditions aux limites. En ce qui concerne les phénomènes de croissance et de consommation cellulaire, ceux-ci sont modélisés par l'équation de Monod qui rend compte de la croissance bactérienne dans un milieu où le substrat est limitant. Dans l'article original, une extension de ce modèle est choisie pour prendre en compte l'impact de plusieurs substrats lors du calcul du taux de croissance spécifique :

$$\gamma_o[i, j] = \frac{P_o[i, j]}{(K_o + P_o[i, j])}$$

$$\mu_{o,n,c}[i, j] = \mu_{\max, \text{souche}} \cdot \gamma_o[i, j] \cdot \gamma_n[i, j] \cdot \gamma_c[i, j]$$

- μ_{\max} : Taux de croissance spécifique maximal $[h^{-1}]$

Grâce à ce modèle, [3] apporte un modèle capable de rendre compte du lien entre la structure du biofilm et les concentrations en nutriments du milieu ce qui en fait une fondation solide pour la construction d'un nouveau modèle. Les travaux de cet article ne prenaient toutefois pas en compte la mort cellulaire, une composante pourtant essentielle des biofilms puisque la présence de cellules mortes peut servir à la fois de support mais aussi de barrière protectrice [4],[5]. De plus, l'objectif étant ici d'ajouter des interactions au sein du modèle, l'ajout de conditions de mort permettrait de rendre compte d'un certain type d'interactions, à savoir, l'antagonisme bactérien.

En effet, les biofilms sécrètent un ensemble de molécules dont par exemple, des toxines, moyen de lutte des macro-colonies dans des situations de compétition nutritionnelle ou spatiale. Les nutriments étant limitants, l'espace étant fini, toutes les conditions sont réunies pour que les deux types de compétition aient lieu. Ainsi la sécrétion de toxines par les différentes souches permet de modéliser cet aspect des interactions bactériennes. Ainsi le modèle que je présente ici incorpore une production de toxine dont voici l'équation :

$$[Toxin]_{\text{cellule}_i} = \frac{n_{\text{other}}^2 \cdot \alpha_{\text{cellule}_i} \cdot 1}{\exp(n_{\text{same}} + 1)}$$

- $[Toxin]_{\text{cell}_i}$: Production de toxine de la cellule $[kg.m^{-3}]$
- n_{other} : Nombre de cellules vivantes de souches étrangères dans le voisinage de la cellule
- n_{same} : Nombre de cellules vivantes dans le voisinage étant de la même souche que la cellule
- α_{cell_i} : Taux de production de Toxine $[kg.m^{-3}.h^{-1}]$

(La multiplication par 1 est explicitement montrée ici pour spécifier que l'on multiplie la production horaire par une durée d'une heure, durée correspondant à l'unité de temps de ce projet.) L'équation que je propose a pour but de rendre compte de la production de toxine (paramètre α_{cell_i}) mais également de conditionner cette production. En effet, la bactérie produira des toxines uniquement si il y a une souche étrangère dans son voisinage ($n_{\text{other}} > 0$), ce coefficient est mis au carré pour accentuer l'importance de la présence de ces bactéries étrangères et pour introduire de la non linéarité dans la réponse de la bactérie à cette présence étrangère. Ceci est pondéré par l'exponentielle du nombre de cellule voisine de la même souche que la cellule concernée plus 1 pour compter la cellule en elle même. L'effet de ce dénominateur est de limiter la production individuelle de toxine lorsque la bactérie est entourée de cellules de sa population.

En effet ceci a été choisi car au sein des biofilms, grâce au quorum sensing entre autre, les cellules sont 'au courant' de leur environnement, la production de toxine étant un effort métabolique non négligeable, diminuer la production individuelle lorsqu'elle est en groupe permet d'économiser des ressources individuellement tout en assurant une forte production par l'ensemble du cluster. De plus, pour rendre compte du fait que cette production de toxine correspond à un effort métabolique non

négligeable, je propose de lier le taux de croissance à la production de toxine. Voici la nouvelle écriture :

$$\mu[i, j] = \mu_{o,n,c}[i, j] \exp(-\omega_{\text{strain}}[Toxin]_{\text{cell}_i})$$

- ω_{strain} : Coefficient de pénalité de production de Toxine. Défini à 1 par défaut.

Avec ce modèle, plus la cellule produit de toxine, plus son taux de croissance diminue. Les quantités produites individuellement étant faibles de manière absolue, $\exp(-\omega_{\text{strain}}[Toxin]_{\text{cell}_i})$ est proche de 1. Le paramètre ω_{strain} , dépendant de la souche permet de traduire à quel point la production de toxine impacte la croissance de la souche, plus ce dernier est élevé, plus une forte production diminuera le taux de croissance. Comme vous le constatez, ces ajouts permettent de grandement enrichir le modèle en prenant en compte des phénomènes cruciaux de la vie des biofilms. Le pouvoir de ce modèle est également de pouvoir simuler des relations complexes entre souches bactériennes aux caractéristiques variées. Plus précisément voici la liste exhaustive actuelle des paramètres permettant la définition d'une souche :

- μ_X : consommation spécifique maximale de l'élément nutritif X $[h^{-1}]$
- μ_{max} : taux de croissance spécifique maximal $[h^{-1}]$
- mobility : proportion de bactéries mobiles
- O_status : affinité pour l'oxygène (0 pour anaérobie et 1 pour aérobie)
- $X_{\text{threshold}}$: Concentration seuil pour l'état de mort pour le nutriment ou la toxine X $[kg.m^{-3}]$
- alpha_TOX : taux de production de toxines $[kg.m^{-3}.h^{-1}]$
- tox_prod_penalty : ω_{strain}
- tox_+_id : Type de toxine produite par la souche

- `tox_id` : Type de toxine qui inhibe cette souche
- C_{div} : biomasse minimale nécessaire pour qu'une cellule se divise $[kg]$
- `Shared_portion` : Proportion de biomasse à partager si la division et le mouvement sont impossibles

Maintenant que nous avons défini la croissance cellulaire, la production de toxines et les paramètres d'une souche, définissons la consommation de nutriments. [3] propose une façon de calculer la consommation bactérienne en utilisant l'équation suivante pour un nutriment X :

$$J_X[i, j] = \mu_X[i, j] \cdot \frac{P_X[i, j]}{(K_X + P_X[i, j])}$$

- $J_X[i, j]$: Taux de consommation (h^{-1}) du nutriment X.

Cette équation permet d'établir un lien entre la consommation des cellules et les niveaux actuels de nutriments environnants, ce qui est effectivement important puisque lorsque les concentrations diminuent substantiellement, les cellules bactériennes adaptent leur consommation. Ceci leur permet de survivre un certain temps dans des conditions parfois peu favorables jusqu'à ce que les conditions du milieu s'améliore ou au contraire jusqu'à leur mort. Il est important de noter que dans l'article original, les auteurs ont choisi de multiplier le taux de consommation d'azote et de carbone par le coefficient de Monod pour l'oxygène. Ils justifient ce choix en avançant l'argument que presque tous les processus biologique impliquent de l'oxygène sous une forme ou une autre. Bien que cette hypothèse soit intéressante, j'ai décidé de ne pas appliquer le coefficient d'oxygène pour l'azote et le carbone.

J'ai estimé qu'il valait mieux utiliser l'équation par défaut de Monod pour des raisons numériques. En effet, la multiplication par le coefficient d'oxygène rend la consommation et, dans une

certaine mesure, la croissance, beaucoup plus faibles, mais, comme nous le verrons plus loin, la croissance de novo des cellules est déjà difficile à réaliser, ainsi ne pas faire cette multiplication a peu d'impact en terme de modélisation mais aussi en pratique mais rendra tout de même la croissance de novo plus fréquente.

Le dernier point à aborder concernant le modèle construit correspond à la définition concrète de la croissance bactérienne. Le calcul du taux de croissance est une chose, mais son utilisation pour faire évoluer la biomasse en est une autre. Une fois de plus ma réflexion est partie du travail de [3], dans ce dernier, la biomasse est mise à jour de la manière suivante :

$$Biomasse[i, j]_t = Biomasse[i, j]_{t-1} \cdot (\mu[i, j]_t + 1) \cdot 1$$

Ce modèle doit être interprété comme une accumulation de biomasse sur une durée d'une heure qui correspond à $Biomasse[i, j]_{t-1} \cdot \mu[i, j]_t$. On parlera de concentration en biomasse ($kg.m^{-3}$). Cependant ceci ne sera pas le modèle retenu ici. En effet le défaut constaté par ce modèle est le suivant : lors de l'utilisation de valeurs réalistes pour la simulation, notamment pour l'initialisation de la biomasse (une bactérie possède une masse moyenne de $1 * 10^{-12}kg$) mais aussi concernant les concentrations en nutriments en $kg.m^{-3}$, le taux de croissance μ est évidemment très faible. La multiplication de la valeur en biomasse déjà très petite par ce taux de croissance en résulte une accumulation véritablement infime de biomasse, si proche de 0 que la nouvelle valeur de biomasse est essentiellement inchangée de la première.

Au delà de cela il faut se rappeler qu'à chaque division, la biomasse est divisée par deux : l'effet d'un terme d'accumulation de biomasse infime couplée à une conservation de la biomasse lors de la division entraîne une impossibilité d'observer de la croissance de novo (c'est à dire une croissance due à l'accumulation et non pas par la simple valeur de biomasse initiale) lorsque l'on utilise des valeurs réalistes. Pour palier à ce problème j'ai donc réfléchi à un autre moyen de voir la croissance

bactérienne. Tout processus du monde est soumis à l'équation de conservation, les processus permettant la croissance d'un être vivant également. Tout être vivant utilise de l'énergie et de la matière pour croître, ainsi la consommation de nutriments par une bactérie peut être vue comme la source de matière que celle-ci peut utiliser pour sa croissance. Il y a tout de même un lien entre la taille actuelle d'un individu et sa croissance, lien exploité dans le modèle cité plus haut. En prenant tout cela en considération, voici finalement le modèle de croissance:

$$\beta_O[i, j]_t = J_O[i, j]_t \cdot P_{O, diff}[i, j]_t \cdot 1$$

$$\kappa[i, j]_t = \beta_O[i, j]_t + \beta_N[i, j]_t + \beta_C[i, j]_t$$

$$B[i, j]_t = B[i, j]_{t-1} + \mu[i, j]_t \cdot (\kappa[i, j]_t + B[i, j]_{t-1}) \cdot 1$$

Où :

- $\beta_X[i, j]_t$ est la consommation du nutriment X par la cellule (i, j) au temps t exprimée en $[kg.m^{-3}]$
- $B[i, j]_t$ est la concentration en biomasse correspondant à la cellule (i, j) au temps t exprimée en $[kg.m^{-3}]$

Enfin, sur la base des équations fournies dans [3] et de celles qui ont été ajoutées, nous pouvons maintenant définir notre système de mise à jour comme suit :

$$\begin{cases} P_O[i, j]_t = P_{O, diff}[i, j]_t - J_O[i, j]_t P_{O, diff}[i, j]_t \cdot 1 \\ P_C[i, j]_t = P_{C, diff}[i, j]_t - J_C[i, j]_t P_{C, diff}[i, j]_t \cdot 1 \\ P_N[i, j]_t = P_{N, diff}[i, j]_t - J_N[i, j]_t P_{N, diff}[i, j]_t \cdot 1 \\ Toxin[i, j]_t = Toxin[i, j]_{t-1} + Toxin_{prod}[i, j]_t \\ B[i, j]_t = B[i, j]_{t-1} + \mu[i, j]_t (\kappa[i, j]_t + B[i, j]_{t-1}) \cdot 1 \end{cases} \quad (1)$$

Autres fonctionnalités

Avec cet ensemble d'équations régissant l'environnement et l'évolution de la biomasse, j'ai décidé d'ajouter plus de complexité au modèle en ce qui concerne les processus biologiques. La

division et le partage des ressources étaient déjà présents dans [3] et j'ai déjà mentionné l'ajout de conditions de mort dans la simulation. Cependant, il y a un dernier phénomène que je n'ai pas encore abordé : le mouvement cellulaire. Dans les biofilms, il a été démontré que certaines bactéries sont encore mobiles au sein de la formation du biofilm [6], [7]. Cela dépend de la communauté bactérienne formant le biofilm et des conditions environnementales. Il a également été observé qu'au cours du cycle de vie d'un biofilm, les cellules bactériennes peuvent être éjectées du biofilm d'origine. J'ai donc ajouté la possibilité pour chaque cellule de se déplacer et la possibilité de créer des souches ayant une mobilité différente (le paramètre "mobility" introduit plus tôt).

Ainsi, le mouvement cellulaire sera possible dans le voisinage de la bactérie. En une itération, et donc au bout d'une heure, la bactérie pourra au maximum se déplacer de r microns, r étant le rayon du voisinage. Ces vitesses correspondent bien à l'ordre de grandeur ayant déjà été observé dans certains biofilms [6]. Il est toutefois nécessaire de prendre conscience que cette motilité est hautement souche spécifique et pourrait donc très largement varier en fonction de la bactérie considérée.

La reproduction étant l'objectif principal de tout être vivant, j'ai décidé que la reproduction serait le premier événement pris en compte lors du traitement des cellules. Ainsi, ce n'est que si une cellule ne peut pas se diviser qu'elle tentera de se déplacer dans son voisinage.

J'ai également ajouté la possibilité pour la matière morte de se décomposer (si le paramètre correspondant est fixé à True, la matière morte finira par disparaître), la possibilité d'initialiser l'environnement avec des concentrations non-homogènes de nutriments et différents états d'initialisation à savoir 'floor', 'surface', 'on_cluster', 'separate' et 'random'.

3 Algorithme

Pour mettre en œuvre ce simulateur de biofilm, j’ai écrit un script en Python qui définit une classe appelée ‘GRID’ qui contient toutes les méthodes nécessaires depuis les méthodes d’initialisation jusqu’aux règles de mise à jour, aux tracés et à la fonction d’exécution. Les principales difficultés rencontrées lors de la réalisation de ce projet Python ont été l’optimisation et la visualisation de la simulation en raison d’échelles de concentration très étalées et des valeurs de biomasse proches de 0. De petites astuces ont permis de résoudre la plupart des problèmes de visualisation. En ce qui concerne l’optimisation, j’ai essayé d’utiliser les opérations vectorielles avec la bibliothèque numpy aussi souvent que possible, ce qui a considérablement amélioré le temps de calcul. Cependant, le temps d’exécution reste lent en fonction des paramètres de simulation choisis. Les facteurs qui affectent grandement les performances sont :

- Taille de la grille (en particulier pour les simulations 3D)
- Seuil de division et valeurs initiales de biomasse : plus le seuil de division est petit par rapport à la valeur initiale de la biomasse, plus les cellules pourront se diviser longtemps. Si de nombreuses cellules survivent à chaque itération, la croissance augmente le nombre de cellules de manière exponentielle, ce qui ralentit le temps de calcul.
- Rayon de voisinage : les utilisateurs ont la possibilité de choisir le paramètre du rayon de voisinage, plus ce rayon est grand, plus chaque itération est coûteuse en temps de calcul en raison du fait que le nombre de cases associées à chaque cellule est une fonction non-linéaire du rayon ($r^2 + (r + 1)^2$ pour le voisinage de Von Neumann et $2(r + 2)^2$ pour celui de Moore).

Toutefois, ceci est principalement dû à la nature du projet, puisqu’il s’agit d’un automate cellulaire avec de nombreuses règles basées sur des

équations, le nombre d’opérations ne s’adapte pas bien aux augmentations d’échelles en raison des considérations de voisinage et des différents calculs effectués à l’intérieur de tous les processus simulés. Une représentation de l’algorithme est présentée dans [Figure 2].

Cet algorithme commence par initialiser toutes les grilles utiles à la simulation en cours, fixe les concentrations initiales puis inocule les premières cellules bactériennes. Après ces étapes d’initialisation, pour chaque itération, la diffusion des molécules de l’environnement est calculée puis, pour chaque cellule à traiter, nous calculons la production de toxines, la consommation de nutriments et mettons à jour le taux de croissance sur la base des calculs effectués, nous mettons enfin à jour l’environnement abiotique (concentrations de nutriments et de toxines).

Après cela, nous mettons à jour les décès si possible, puis nous examinons les conditions de reproduction, de mouvement et de partage des ressources. Les décès sont mis à jour avant la reproduction pour rendre compte du fait que dans n’importe quelle population d’être vivants, seuls les individus vivant dans de suffisamment bonnes conditions et dont la fitness est suffisamment forte se reproduisent. Ainsi en calculant les morts avant les reproductions une forme de sélection se fait, bien qu’elle soit très environnement dépendante et peu fitness dépendante.

Enfin, si la décomposition de la matière morte est activée, nous mettons à jour les états de la matière morte. Ce processus est répété le nombre d’itérations souhaité, sauf s’il n’y a plus de cellules vivantes à traiter ou si la matrice des souches ne change plus (équilibre). Notez que cet état d’équilibre avec des cellules vivantes toujours présentes ne peut être atteint que si le mouvement est désactivé et dans certaines conditions. Le modèle désormais connu, l’algorithme décrit, il est maintenant temps d’observer les résultats de simulations.

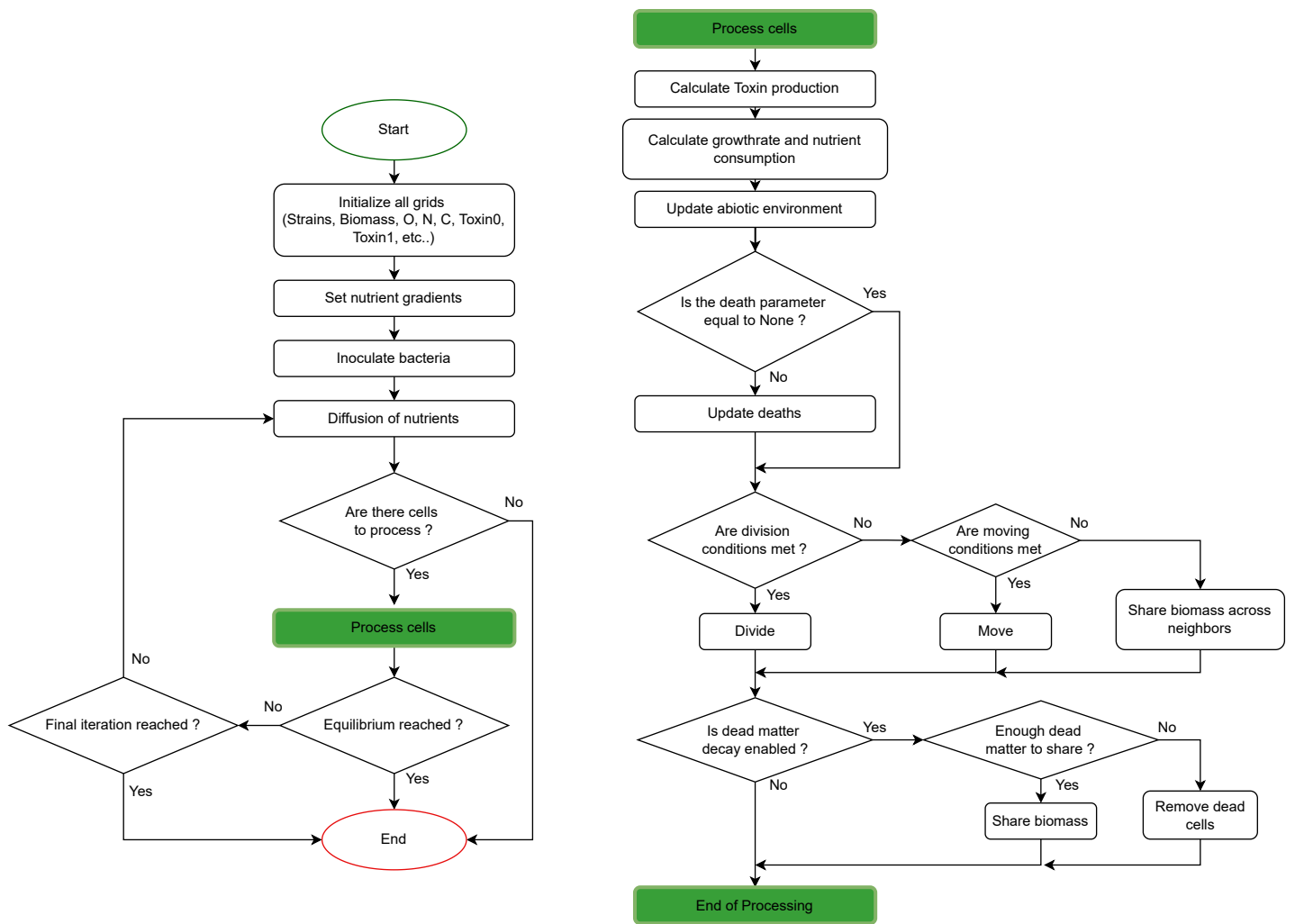


Figure 2: Algorithme. Figure créée grâce au logiciel draw.io.

4 Expérience numérique

Comme je l’ai déjà annoncé, le principal intérêt de ce projet est de pouvoir simuler des interactions bactériennes. La possibilité de définition des divers caractéristiques de chaque souche permet de représenter des souches bactériennes différentes sur de nombreux points. Ainsi, j’ai voulu reprendre une des conditions expérimentées par [8] : la co-inoculation d’*E.coli* et de bactéries du genre *Bacillus*. Le but de cette expérience est de quantifier l’effet antagoniste ou non de la présence des bactéries *Bacillus* sur le développement du biofilm pathogène d’*E.coli*. À l’aide de différentes sources citées ci-après, j’ai pu me renseigner en particulier sur la bactérie *B. velezensis*, ainsi imaginons deux souches : une d’*E.coli*, ne produisant pas de toxine et une autre de *B.velezensis* produisant une ”toxine” à l’encontre d’*E.coli*. Voici quelques caractéristiques des bactéries que nous mettrons dans la simulation :

- *E. coli* : Aérobie, Taux de croissance spécifique maximum connu le plus élevé ($3 h^{-1}$), Très Mobile. [9], [10], [7].
- *B. velezensis* : Aérobie, Taux de croissance élevé, Production d’acide bacillunoïque (considéré ici comme ”toxine” sécrétée en cas de compétition). [11], [12], [13], [14].

La paramétrisation des deux bactéries est présentée en [Table 1]. Comme annoncé plus tôt, je suppose ici que l’éventuelle toxine sécrétée par *E.coli* n’affecte pas *B.velezensis* et pour cela j’ai fixé une production de toxine nulle pour *E.coli* vis à vis de *B.velezensis*. Cette hypothèse simplificatrice est utile pour d’autant plus distinguer ces deux bactéries. En effet telles qu’elles sont représentées, *E.coli* présentera une croissance extrêmement rapide, un considérable avantage dans cet espace fini aux ressources limitées, cependant *B.velezensis* pourra quant à elle utiliser la production de ”toxine” pour tenter de gagner cette compétition. En tant que référence pour cette interaction j’ai pris les résultats présentés par [8],

bien que nos souches d’*E.coli* et de *B.velezensis* soient hypothétiques et ne correspondent donc absolument pas à celles utilisées dans ces expériences, ces résultats me permettent d’avoir une comparaison avec un exemple de biofilm mixte réel de ces deux bactéries. Ces résultats montrent au bout de 24h un antagonisme très faible entre la bactérie *B. velezensis* et *E.coli* dont le score d’interaction est de 0.39 (score entre 0 et 1, en dessous de 0.5 on peut considérer qu’il n’y a pas ou très peu d’antagonisme). L’unité d’itération étant l’heure, on simulera pour 24 itérations afin de correspondre à 24h. Le résultat obtenu est présenté en [Figure 3].

Tout d’abord, les paramètres de simulation ayant permis d’obtenir ses résultats sont :

- Taille grille: 160x160 (une case correspondant à une bactérie, cela correspond donc à une grille de 160 μm de côté)
- Nombre initial de cellule : 32 pour chaque souche (20% de la longueur)
- Biomasse initiale : $1 * 10^{-12} [kg.m^{-3}]$
- Inoculation : ’floor’ avec ’thickness’ = 1 =, Un tapis de cellule est inoculé sur le fond.
- Voisinage de Moore avec rayon de 1
- Constante de diffusion des Toxines : [15] a estimé la diffusion dans le périplasma entre 3 et 6 $\mu m^2.s^{-1}$. Le périplasma étant moins concentré en molécules que le cytoplasme, je fais ici l’hypothèse que l’on peut comparé la diffusion des protéines et métabolites au sein du périplasma à celle observé dans le milieu aquatique entourant le biofilm. Ainsi $D_{tox} = 4.5 * 10^{-6} \mu m^2.s^{-1}$
- Les valeurs initiales de nutriment (en $[kg.m^{-3}]$) sont $7 * 10^{-3}$ pour l’Oxygène, $0.7 * 10^{-3}$ pour l’azote, $25 * 10^{-3}$. Valeurs utilisées par [3]. Une variation linéaire est appliquée de haut en bas pour l’Oxygène (concentration la plus élevée

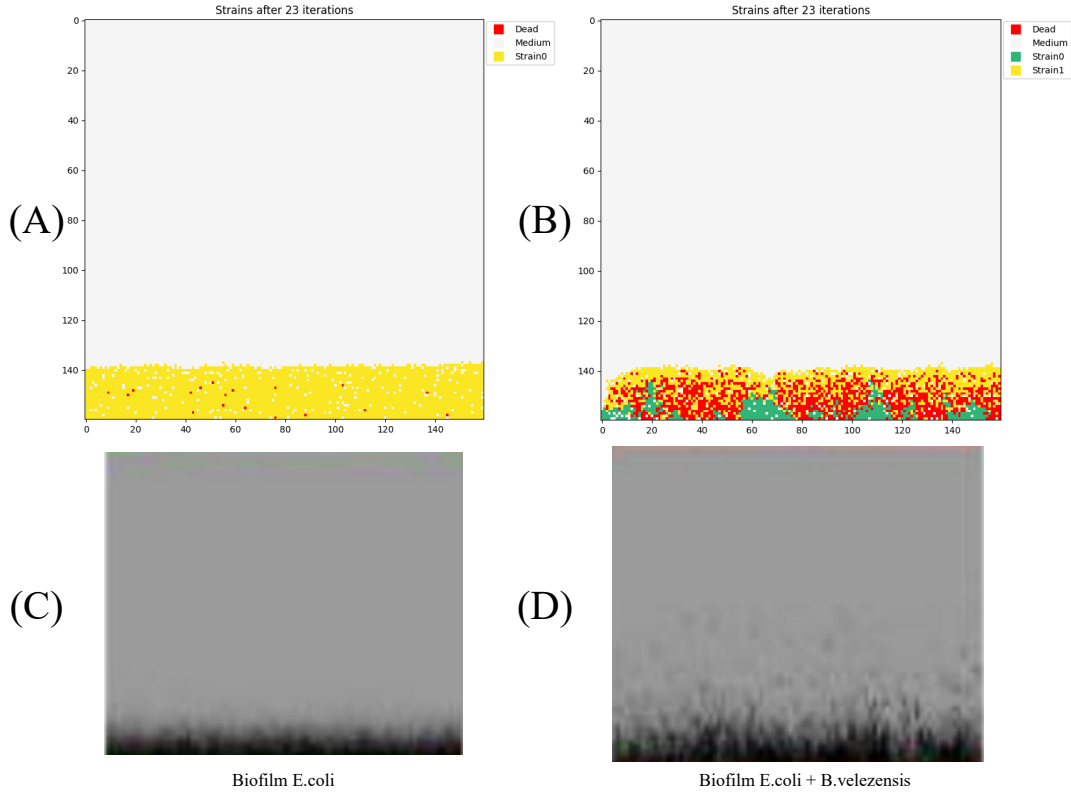


Figure 3: (A) Tranche du biofilm simulé d'*E.coli* (Strain0 en jaune) après 24h. (B) Tranche du biofilm mixte simulé d'*E.coli* (Strain1, en jaune) et de *B.velezensis* (Strain0, en vert) après 24h. (C) Tranche du biofilm d'*E.coli* après 24h issue de [8]. (D) Tranche du biofilm mixte d'*E.coli* et de *B.velezensis* après 24h issue de [8]. (C) et (D) ont été obtenus par [8] en utilisant la microscopie confocale et le logiciel BiofilmQ permettant la reconstruction des images obtenues. Ces images correspondent à des carrés de 160 microns de côté, la même taille a été utilisée pour la simulation.

Parameter	<i>B.velezensis</i>	<i>E.coli</i>
mu_O [h^{-1}]	0.3	0.3
mu_N [h^{-1}]	0.15	0.15
mu_C [h^{-1}]	1.5	1.5
mu_max [h^{-1}]	0.95	3
mobility	0.095	1
O_status	1	1
O_threshold [$kg.m^{-3}$]	2e-10	2e-10
N_threshold [$kg.m^{-3}$]	5e-7	5e-7
C_threshold [$kg.m^{-3}$]	0.2e-7	0.2e-7
alpha_TOX [$kg.m^{-3}.h^{-1}$]	0.8e-15	0
TOX_threshold [$kg.m^{-3}$]	7.1e-6	0.1e-13
tox_prod_penalty	1	1
tox_+_id	0	1
tox_-_id	1	0
Cdiv [$kg.m^{-3}$]	8.1e-13	8.1e-12
Shared_portion	0.5	0.5

Table 1: Paramètres caractérisant *B.velezensis* et *E.coli*.

en haut de la grille) et de bas en haut pour le Carbone et l'Azote.

- Toutes les conditions de mort sont activées, pas de renouvellement de matière morte.

En comparant le biofilm simulé d'*E.coli* (**Figure3A**) avec l'image du biofilm réel d'*E.coli* (**Figure3C**), on constate que ce dernier est assez plat avec peu d'épaisseur et que ces caractéristiques sont également retrouvées avec le biofilm simulé. Ensuite en observant les biofilms simulés d'*E.coli* et celui des deux espèces mélangées après 24h (**Figure3B**), on observe que la hauteur des deux est similaire et qu'*E.coli* s'est bien développée dans les deux cas. Cependant, on voit bien l'effet des toxines de *B.velezensis*, de nombreuses bactéries *E.coli* sont mortes. Ceci est attendu compte tenu des paramètres utilisés. On constate tout de même que cela n'a pas été suffisant pour avoir un effet antagoniste sur le développement de *E.coli* puisque celle-ci parvient encore à prendre le dessus et à coloniser largement le fond de la grille. Ainsi je juge ces résultats satisfaisants en termes de modélisation de l'interaction d'intérêt puisque la comparaison avec les images réelles peut se faire

assez justement et que les conclusions qui en sont tirées sont les mêmes.

Une des perspectives de [8] est l'utilisation d'un consortium de souches positives afin d'endiguer le développement de biofilms pathogènes. La première souche de *B.velezensis* précédemment utilisée dans la simulation n'a pas permis d'atteindre cet objectif, mais qu'en est-il d'une combinaison de deux souches *B.velezensis* ? Ainsi ajoutons une souche *B.velezensis* qui diffère légèrement de la précédente par le fait qu'elle possède un taux de croissance légèrement plus faible (fixons le à $0.85 h^{-1}$), une production de toxine 2 fois plus faible que la première, mais une plus grande mobilité (0.3). Cette souche consommera légèrement plus de carbone en raison de sa mobilité accrue ($\mu_C = 2 h^{-1}$) et légèrement moins d'azote pour une raison arbitraire ($\mu_N = 0.1 h^{-1}$). Tous les autres paramètres sont identiques. Ainsi, on aura dans l'environnement, deux souches similaires, mais aux caractéristiques légèrement différentes qui produiront une toxine impactant *E.coli*.

Mais avant de voir le résultat en inoculant ces 3 souches, regardons si la seconde souche de *B.velezensis* amène aux mêmes conclusions ou non que la première. Les résultats de cette nouvelle simulation sont présentés en [Figure 4].

Sur cette figure, on peut constater que si on observe uniquement le biofilm à 24h (Figure4A), on ne sait pas s'il y a un effet antagoniste ou pas, les deux souches semblent assez bien occuper l'espace. Cette observation est supportée par l'analyse de la structure de la population au cours du temps jusqu'à 24h (Figure4C) où l'on peut en déduire que les 2 souches sont aux coudes à coude avec des passages en supériorité pour l'une et pour l'autre, cependant, à 24h, la dynamique ne semble clairement pas avoir pris sa tendance finale. En simulant l'évolution du biofilm sur une plus longue durée, on peut enfin conclure sur l'effet antagoniste ou non de cette seconde souche : que cela soit sur l'état du biofilm à 48h (Figure4B) ou sur l'évolution de la structure de la population au cours du temps, on constate que c'est bien le pathogène *E.coli* qui sort vainqueur de cette compétition, ainsi la seconde souche n'a pas non plus réussi à endiguer le développement du biofilm pathogène. Cependant on notera qu'à 24h, la présence de cette souche ralentit davantage la croissance du biofilm pathogène que la première souche de *B.velezensis* testée.

Seule les souches *B.velezensis* ne parviennent pas à empêcher la croissance de *E.coli*, regardons désormais l'effet d'une co-inoculation des 3 souches, résultats présentés en [Figure 5].

On observe alors une nette différence vis à vis du développement d'*E.coli* avec une bien plus grande proportion de cellule *E.coli* morte visible dans le biofilm ainsi qu'une épaisseur de biofilm légèrement plus faible comparée aux simulations précédentes. Ainsi on observe donc un effet synergique de l'association des deux souches qui résulte en un véritable antagonisme contre *E.coli*.

Par cette expérience, on peut donc voir que le modèle est utile pour tenter de comprendre diverses

interactions au sein de biofilms microbiens et est capable de rendre compte de phénomènes observés dans l'étude de biofilms. Cependant, ce simulateur présente un certains nombre de limite qu'il est important de souligner.

5 Discussion

En effet, on peut commencer cette critique du simulateur par le fait qu'aucun processus implémenté ne rends réellement compte de la production d'exopolysaccharides (EPS) pourtant nécessaire à la formation de n'importe quel biofilm et étant la source de nombreuses des propriétés de résistance mécanique, chimique et biologique des biofilms. Notre modèle actuel peut être vu comme ignorant totalement les EPS ou alors en ne considérant que ces derniers formes une couche infiniment petite autour des bactéries. Dans les deux cas, le problème reste le même : les propriétés de cette couche d'EPS sont négligées, empêchant à notre modèle de rendre compte de certains processus clé comme la résistance aux antibiotiques des cellules au centre du biofilm, car la diffusion à travers la couche d'EPS est largement ralenti, la nature de certaines structures comme les biofilms caliciformes ou autres qui sont en partie possible grâce au support mécanique que constitue la couche d'EPS.

Un autre phénomène très étudié chez les biofilms bactériens correspond à la possibilité d'expulser des bactéries du biofilm une fois ce dernier arrivé à un stade de maturité avancé afin de coloniser une autre partie de l'environnement. Ce phénomène est crucial pour de nombreux types de biofilm et n'est pas reproductible avec le modèle construit.

En ce qui concerne du modèle en lui-même, on peut aussi critiquer mes choix d'équation pour la production de toxine, le lien entre production de toxine et taux de croissance et pour finir le calcul de la biomasse à chaque itération. En effet ces équations se basent sur ma vision des phénomènes importants et ma façon de les traduire en équation,

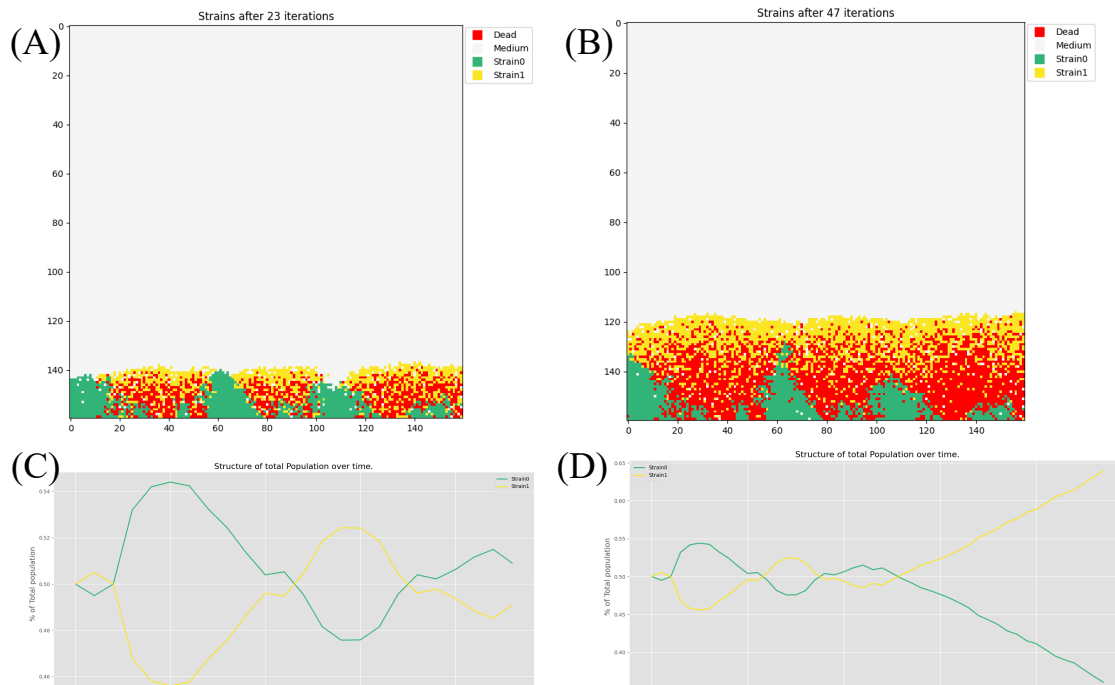


Figure 4: (A) Tranche du biofilm mixte simulé d'*E.coli* (Strain1, en jaune) et de la seconde souche de *B.velezensis* (Strain0, en vert) après 24h. (B) Etat du biofilm après 48h. (C) et (D) Structure de la population total au cours du temps jusqu'à respectivement 24 et 48h.

cependant ils ne sont absolument pas supportés par des données, à la fois car l'on manque de données pour ce type de phénomènes en particulier sous la forme de biofilm mais aussi par ce que ma démarche n'a pas été d'adapter ces équations à une quelconque série de données, rendant ces choix très fortement criticables. Néanmoins ces équations permettent tout de même de rendre compte des phénomènes souhaités dans le cadre de ce simulateur ce qui est satisfaisant pour l'aspect modélisation à but pédagogique et non à but prédictif.

6 Conclusion

Pour conclure sur ce projet, au cours de ces 2 semaines, j'ai pu construire un simulateur de biofilm permettant la simulation de divers phénomènes comme la compétition spatiale, nutritionnelle, la

formation de structures de biofilms ressemblant à certaines observées en culture, mais aussi la production de toxine. Bien qu'il comporte un certain nombre de défauts, ce modèle peut s'avérer intéressant pour tenter de comprendre les interactions bactériennes au sein des biofilms et les dynamiques populationnelles qui en résulte.

Toutefois trouver un ensemble de paramètres permettant des simulations cohérentes peut demander un certain effort. C'est pourquoi il est donc conseillé de bien réfléchir à l'ensemble des paramètres, notamment d'être conscient que le modèle a été pensé pour utiliser des valeurs numériques réalistes et donc très proches de 0. La simulation 3D est extrêmement coûteuse en termes d'opérations et cela résulte en un temps d'exécution excessivement long. On pourrait envisager l'emploi du multiprocessing pour tenter de mieux gérer le passage à l'échelle 3D, et même pour mieux gérer les plus grandes dimensions de grilles.

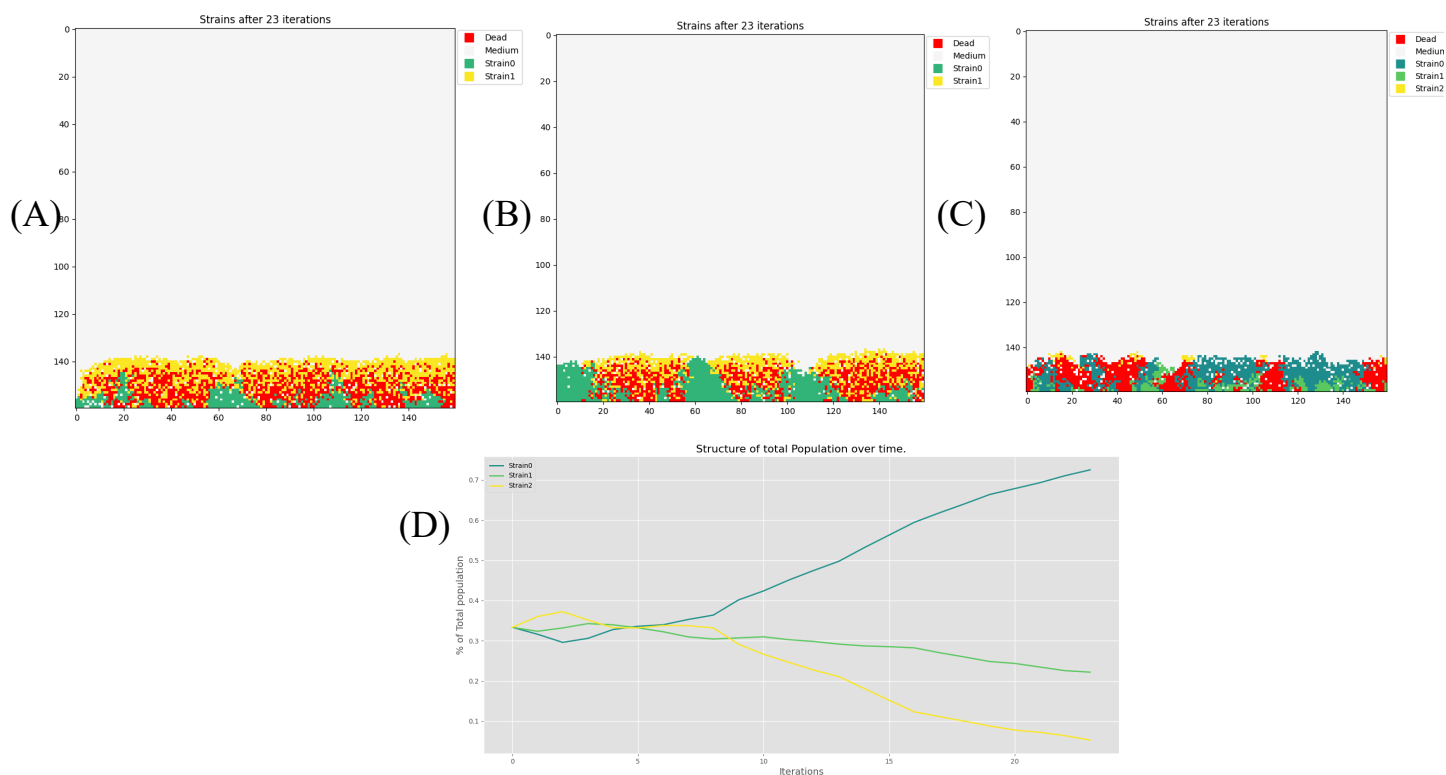


Figure 5: (A) Tranche du biofilm mixte simulé d'*E.coli* (Strain1, en jaune) et de la première souche *B.velezensis* (Strain0, en vert) après 24h. (B) Tranche du biofilm mixte simulé d'*E.coli* (Strain1, en jaune) et de la seconde souche de *B.velezensis* (Strain0, en vert) après 24h. (C) Tranche du biofilm mixte simulé d'*E.coli* (Strain2, en jaune), de la 1ère souche de *B.velezensis* (Strain1, en vert) et de la seconde (Strain0, en bleu) après 24h. (D) Structure de la population total au cours du temps associée à la simulation ayant produit (C).

Il est évident que ce simulateur pourrait être amélioré en prenant en compte la production d'EPS, en incorporant des équations rendant compte des autres types de structures de biofilms possibles ainsi que les phénomènes de colonisation par éjection. De plus, l'utilisation de modèles data-driven pour la production de toxine et la croissance sous forme de biofilm permettra d'améliorer sa fiabilité et peut être même de faire passer ce simulateur du côté prédictif. Cependant, ceci demanderait beaucoup plus de temps ainsi que des efforts expérimentaux considérables afin de produire les données nécessaires pour la validation des modèles. Le code source de ce projet est disponible à l'adresse suivante [<https://github.com/Aelrach/Biofilm-Simulator>] et qui que vous soyez, si cela vous intéresse n'hésitez pas à l'améliorer :D!

7 Remerciements

Merci à l'équipe pédagogique de l'UP-09 pour le contenu très enrichissant de leurs cours et merci tout particulièrement à Mr. Christophe Dourzat, pour ses cours, son encadrement et l'ambiance agréable qu'il a su mettre en place durant l'UP-09.

8 Bug

Le seul bug qui demeure et dont j'ai connaissance est un problème lié à la sauvegarde des graphiques des toxines. Je n'ai pas réussi à reproduire ce bug dans d'autres conditions et il ne m'est jamais arrivé à d'autres moments. L'erreur en question était : "ValueError: vmin must be less or equal to vmax" soulevée par le module matplotlib lors de la sauvegarde. Je n'ai pas réussi à trouver l'origine de ce bug, d'autant plus que lors de la simulation qui l'a produit, les itérations précédentes ne présentaient aucun problème. Il semble que cela soit dû à une définition automatique erronée de vmax et vmin au sein du module matplotlib lors de la sauvegarde. Cependant, dans tous les autres cas que j'ai pu simuler avec la version actuelle, le code se comporte

correctement lors de la sauvegarde des Toxines. Par manque de temps, je n'ai pas pu expliquer cet unique bug qu'il est donc possible de rencontrer. Si jamais vous le rencontrez également, désactivez la sauvegarde des Toxines (paramètres 'save_TOX'), cela vous permettra d'observer toutes les autres composantes de la simulation sans problèmes.

References

- [1] Virgile Guéneau et al. "Positive biofilms to guide surface microbial ecology in livestock buildings". In: *Biofilm* 4 (2022), p. 100075.
- [2] Yasmine Dergham et al. "Bacillus subtilis NDmed, a model strain for biofilm genetic studies". In: *Biofilm* (2023), p. 100152.
- [3] Samvel Sarukhanian, Anna Maslovskaya, and Christina Kuttler. "Three-dimensional cellular automaton for modeling of self-similar evolution in biofilm-forming bacterial populations". In: *Mathematics* 11.15 (2023), p. 3346.
- [4] Odile Tresse, S Lescob, and D Rho. "Dynamics of living and dead bacterial cells within a mixed-species biofilm during toluene degradation in a biotrickling filter". In: *Journal of applied microbiology* 94.5 (2003), pp. 849–854.
- [5] Gabi Steinbach et al. "Accumulation of dead cells from contact killing facilitates coexistence in bacterial biofilms". In: *Journal of the Royal Society Interface* 17.173 (2020), p. 20200486.
- [6] AR Rice, Martin A Hamilton, and Anne K Camper. "Movement, replication, and emigration rates of individual bacteria in a biofilm". In: *Microbial Ecology* 45 (2003), pp. 163–172.
- [7] Thomas K Wood et al. "Motility influences biofilm architecture in Escherichia coli". In: *Applied microbiology and biotechnology* 72 (2006), pp. 361–367.

- [8] Virgile Guéneau. “Vers l’utilisation rationnelle de biofilms positifs pour aider à maîtriser les microorganismes indésirables dans les bâtiments d’élevage”. PhD thesis. Université Paris-Saclay, 2023.
- [9] Jonathan D Partridge et al. “Transition of *Escherichia coli* from aerobic to micro-aerobic conditions involves fast and slow reacting regulatory components”. In: *Journal of Biological Chemistry* 282.15 (2007), pp. 11230–11237.
- [10] Maria Schei Haugan et al. “Growth rate of *Escherichia coli* during human urinary tract infection: implications for antibiotic effect”. In: *Antibiotics* 8.3 (2019), p. 92.
- [11] Muhammad Fazle Rabbee et al. “*Bacillus velezensis*: a valuable member of bioactive molecules within plant microbiomes”. In: *Molecules* 24.6 (2019), p. 1046.
- [12] Elliot Nicholas Grady et al. “Characterization and complete genome analysis of the surfactin-producing, plant-protecting bacterium *Bacillus velezensis* 9D-6”. In: *BMC microbiology* 19 (2019), pp. 1–14.
- [13] Lin Shao et al. “*ccdC* Regulates Biofilm Dispersal in *Bacillus velezensis* FZB42”. In: *International Journal of Molecular Sciences* 25.10 (2024), p. 5201.
- [14] Rong Huang et al. “A toxin-mediated policing system in *Bacillus* optimizes division of labor via penalizing cheater-like nonproducers”. In: *Elife* 12 (2023), e84743.
- [15] Christopher H Bohrer and Jie Xiao. “Complex diffusion in bacteria”. In: *Physical Microbiology* (2020), pp. 15–43.