An Application of M-matrices to Preserve Bounded Positive Solutions to the Evolution Equations of Biofilm Models

Richard S. Landry Jr.

*University of New Orleans, rslandr1@uno.edu*

Follow this and additional works at: https://scholarworks.uno.edu/td

Part of the Biological and Chemical Physics Commons, Biological Engineering Commons, Catalysis and Reaction Engineering Commons, Computational Engineering Commons, Engineering Physics Commons, Numerical Analysis and Computation Commons, Partial Differential Equations Commons, and the Water Resource Management Commons

**Recommended Citation**


https://scholarworks.uno.edu/td/2418

This Dissertation-Restricted is protected by copyright and/or related rights. It has been brought to you by ScholarWorks@UNO with permission from the rights-holder(s). You are free to use this Dissertation-Restricted in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you need to obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/or on the work itself.

This Dissertation-Restricted has been accepted for inclusion in University of New Orleans Theses and Dissertations by an authorized administrator of ScholarWorks@UNO. For more information, please contact scholarworks@uno.edu.
An Application of M-matrices to Preserve Bounded Positive Solutions to the Evolution Equations of Biofilm Models

A Dissertation

Submitted to the Graduate Faculty of the University of New Orleans in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Engineering and Applied Science

by

Richard Spencer Landry, Jr.

B.S.E. Tulane School of Engineering, 2002
M.S University of New Orleans, 2005

December, 2017
# Contents

List of Figures ................................................................. iii
List of Symbols ................................................................. v

1 Abstract ................................................................. vii

2 Introduction .................................................. 1

3 Background .................................................. 9
   1 Model Development .................................................. 9
   2 Analytical Results .................................................. 13

4 Computational Models Investigated ........................................... 15
   1 Basic Definitions .................................................. 15
   2 Finite Difference Operators ........................................ 16
   3 Construction of Finite Difference Equations ...................... 19
   4 Assembly of M-matrices ............................................. 26
   5 Criteria to Ensure M-matrices ..................................... 38

5 Numerical Implementation .................................................. 63

6 Illustrative Results .................................................. 67
   1 Results for $\mathcal{E}_1$ ............................................. 67
   2 Results for $\mathcal{E}_2$ ............................................. 75
   3 Results for $\mathcal{E}_3$ ............................................. 83

7 Concluding Remarks .................................................. 99

8 Appendix .............................................................. 108
   1 Matlab Code ......................................................... 108

9 Vita ................................................................. 137
List of Figures

2.1 (a) Biofilm growth on rocks in a stream (USGS) and within a kitchen pipe (MSU Center for Biofilm Engineering). (b) Biofilm phenomena can be caused by bacteria such as *Thiobacillus ferrooxidans* (Iron Bacteria). Bacterial biofilm create oil-like films when they attach themselves to the water surface. Sunlight bounces off the films, giving them an oily appearance. (c) An image from an electron scanning microscope of a *Staphylococcus aureus* biofilm on a vascular prosthesis[1]. .............................................................. 3

5.1 Plot of residuals vs. iterations of 11 iterative schemes for one time step of $E_3$ (a) without and (b) with preconditioning LU matrix. ................................................................. 65

6.1 Time course of $E_1$ for Example 1.1 when $t = \{0, 1, 2\}$ with (a) homogeneous Dirichlet and (b) homogeneous Neumann conditions. ................................. 69
6.2 Continued time course of $E_1$ for Example 1.1 when $t = \{3, 4, 5\}$. ......................... 70
6.3 Continued time course of $E_1$ for Example 1.1 when $t = \{6, 7, 8\}$. ......................... 71
6.4 Time course of $E_1$ for Example 1.2 when $t = \{0, 2, 4\}$ with (a) homogeneous Dirichlet and (b) homogeneous Neumann conditions. ......................... 72
6.5 Continued time course of $E_1$ for Example 1.2 when $t = \{6, 8, 10\}$. ......................... 73
6.6 Continued time course of $E_1$ for Example 1.2 when $t = \{12, 14\}$. ......................... 74
6.7 Time course of $E_1$ for Example 1.3 when $t = \{15, 20, 30\}$. ................................. 76
6.8 Time course of $E_2$ for Example 2.1 when $t = \{0, 0.25, 0.5\}$. ................................. 78
6.9 Continued course of $E_2$ for Example 2.1 when $t = \{0.75, 1, 1.5\}$. ......................... 79
6.10 Continued course of $E_2$ for Example 2.1 when $t = \{2, 5, 10\}$. ............................ 80
6.11 Time course of $E_2$ for Example 2.2 when $t = \{0, 0.5, 1\}$. ................................. 84
6.12 Continued course of $E_2$ for Example 2.2 when $t = \{2, 5, 10\}$. ............................ 85
6.13 Continued course of $E_2$ for Example 2.2 when $t = \{14, 18, 22\}$. ......................... 86
6.14 Continued course of $E_2$ for Example 2.2 when $t = \{26, 30, 40\}$. ......................... 87
6.15 Continued course of $E_2$ for Example 2.2 when $t = \{60, 80, 100\}$. ....................... 88
6.16 Time course of $E_3$ for Example 3.1 when $t = \{0, 0.1, 0.2, 0.3, 0.5, 0.7\}$. ............. 90
6.17 Continued course of $E_3$ for Example 3.1 when $t = \{0.9, 1, 1.5, 2, 2.5, 3\}$. ............ 91
6.18 Continued course of $E_3$ for Example 3.1 when $t = \{4, 5, 6, 7, 8, 9\}$. .................... 92
6.19 Continued course of $E_3$ for Example 3.1 when $t = \{10, 15, 20, 25, 30, 35\}$. .......... 93
6.20 Time course of $E_3$ for Example 3.2 when $t = \{0, 3, 6, 9, 12, 15\}$. ..................... 95
6.21 Continued course of $E_3$ for Example 3.2 when $t = \{18, 21, 24, 27, 30, 33\}$. .......... 96
6.22 Continued course of $E_3$ for Example 3.2 when $t = \{36, 39, 42, 45, 48, 51\}$. ........ 97
6.23 Continued course of $\mathcal{C}_3$ for Example 3.2 when $t = \{54, 57, 60\}$.
List of Symbols

$\Omega$ the spatial boundary domain

$\overline{\Omega}$ the closure of the boundary domain $\Omega$

$\mathbb{N}$ the set of natural numbers

$\mathbb{Z}$ the set of integers

$\mathbb{R}$ the set of Euclidean real numbers

$\mathbb{R}^+$ the positive reals

$\mathbb{R}^p$ p-dimensional Euclidean numbers

$x \in \Omega$ $x$ belongs to the set $\Omega$

$D : [0,1) \rightarrow \mathbb{R}$ the function $D$ mapping the interval $[0,1)$ to the reals

$\times$ the outer product operator

$\cup$ the set union operator

$\nabla$ the gradient operator

$\nabla^2$ the Laplacian operator

$\frac{\partial}{\partial t}$ the partial differentiation operator with respect to $t$

$E_j$ biofilm model experiment $j \in \{1, 2, 3\}$

$L_{E_j}$ length of vectors in experiment $E_j$. $L_{E_1} = 2(M + 1)(N + 1)$, $L_{E_2} = 4(M + 1)$, $L_{E_3} = 4(M + 1)(N + 1)$

$a_{E_j}$ the vector $a$ in boldface of length $L_{E_j}$

$a_i$ the vector element of $a$ where $i \in \{1, 2, \ldots, L_{E_j}\}$

$e_{E_j}$ the vector of length $L_{E_j}$ corresponding to $E_j$ where each entry $e_i = 1$
\[ a < e \]
the operators and relations \{+, -, <, \leq, =, \geq, >\} on vectors are componentwise so that \( a_i < e_i = 1 \) for all \( i \in \{1, 2, \ldots, L_{\mathcal{E}_j}\} \)

\[ 1_A \]
the identity (characteristic) function on the set \( A \)

\[ Y = Y(x, t) \]
the biofilm function of any state \( \{s, u, S, X, I, E\} \) referring to substrate and biomass for \( \mathcal{E}_1 \), or substrate, active biomass, inert biomass, and extracellular polymeric substance for \( \mathcal{E}_2 \) and \( \mathcal{E}_3 \), respectively

\[ Y^0, Y^0_{m,n}, Y^k, Y^k_{m,n} \]
the discrete form of continuous state variables \( Y(x, t) \) at node corresponding to \( Y(t = t_k, x = x_m, y = y_n) \)

\[ \Delta z \]
the step size between nodes of \( z \in \{x, y\} \)

\[ \psi_{k, \pm m,n,z} \]
an arbitrary constant at \( t = t_k, x = x_m, y = y_n \) where \( \pm \) indicates a shift by either +1 or −1 in the \( z \in \{x, y\} \) dimension (defined in the dissertation during construction of the functions as warranted)

\[ \mathcal{N} \]
the set of function solutions that obey Neumann boundary conditions

\[ \mathcal{I} \]
the identity block matrix

\[ \mathcal{I}_N \]
the modified identity block matrix defined as needed in the dissertation

\[ \mathcal{F}_{k, \pm} \]
caligraphic capital letters correspond to block matrices of sizes defined in the dissertation. Superscript \( k \) represents time step, \( \pm \) is either of + (−) case where the block matrix is shifted right (left) of the diagonal block in the main matrix, and \( m \) is the block row number of the spatial component it represents

\[ Y^Y_{\mathcal{E}_j} \]
any one of the diagonal block matrices \( Y^k \in \{A^k_{\mathcal{E}_1}, B^k_{\mathcal{E}_1}, B^k_{\mathcal{E}_2}, B^k_{\mathcal{E}_3}\} \), where \( Y \) is defined as corresponding to any one of the biofilm state variables as before

\[ M^k_{\mathcal{E}_j} \]
the M-matrix at time \( t = t_k \) for experiment \( \mathcal{E}_j \)

\[ \mathcal{W} = (W_{i,j}) \]
an arbitrary matrix where indices are indicated as a subscript corresponding to the \( i \)th row and \( j \)th column

\[ \rho(A^k) \]
the spectral radius of \( A^k \), defined as the absolute maximum eigenvalue

\[ \text{sparse} \]
programs and scripts such as the command \texttt{sparse} use typewriter font to indicate code
1 Abstract

In this work, we design a linear, two step implicit finite difference method to approximate the solutions of a biological system that describes the interaction between a microbial colony and a surrounding substrate. Three separate models are analyzed, all of which can be described as systems of partial differential equations (PDE)s with nonlinear diffusion and reaction, where the biological colony grows and decays based on the substrate bioavailability. The systems under investigation are all complex models describing the dynamics of biological films. In view of the difficulties to calculate analytical solutions of the models, we design here a numerical technique to consistently approximate the system evolution dynamics, guaranteeing that nonnegative initial conditions will evolve uniquely into new, nonnegative approximations. This property of our technique is established using the theory of M-matrices, which are nonsingular matrices where all the entries of their inverses are positive numbers. We provide numerical simulations to evince the preservation of the nonnegative character of solutions under homogeneous Dirichlet and Neumann boundary conditions. The computational results suggest that the method proposed in this work is stable, and that it also preserves the bounded character of the discrete solutions.

Keywords: biofilm, M-matrix, nonlinear diffusion reaction, finite difference model


2 Introduction

A biofilm can be described abstractly as a system of prokaryotic and eukaryotic cells attached to a surface and embedded in an organic biological matrix. Some of these biofilms have positive effects and have widespread use in industry, environmental preservation, and biomedical applications. Other biofilms have serious detrimental consequences to the environment, the medical industry, and even efficiency of thermal reactors. In many cases the study of biofilms derives from learning the structure of the independent types of biofilms so as to promote their growth, or in the bad case, limiting or preventing them from forming on the host environment. To this end, both experimental studies and mathematical modeling based on the field results, or using models built through first principles are paramount in their investigation. In this study, we search for suitable mathematical models for simple biofilm structures and then harness the power of the M-matrix to ensure that the model obeys certain presupposed conditions.

The problem of faithfully reproducing the growth dynamics of biological or chemical constituents and their corresponding reactions with each other and the surrounding environment is an important task in view of the many realistic problems where these phenomena appear. Negative effects of biofilms span all types of systems, ranging from mechanical devices to biological settings. Even in some heat exchange equipment, fouling biofilm formation causes a significant energy loss by increasing heat transfer resistance [2]. It has been suggested that up to 60% of hospital-acquired infections are due to biofilms [3]. Bacterial
biofilms present an emerging link to the disease pathogenesis of many chronic human infections [4]. A higher incidence of biofilms in chronic rhinosinusitis patients suggests a role in its pathogenesis for example [3]. Bacteria are responsible for a range of human diseases that are difficult to clear for a variety of reasons. Research has shown that bacteria can seek protected environments passively by avoiding deadly environments or actively by manipulating their phenotypic expression and gathering in structured biofilm communities [5].

Additionally, many nosocomial infections are assumed to be the result of the presence of pathogenic films in a wide range of medical devices, like catheters and probes used in different hospital services. Biofilms have a major role in implants or devices placed inside the human body, and pose a serious risk for organ transplants. Future researchers have to aim at identifying effective mechanisms for controlling biofilm formation and to develop antimicrobial agents against bacteria in biofilms [6]. As an example, Fungal keratitis is commonly caused by *Fusarium* species and less commonly by *Candida* species, the more recent outbreaks of which were associated with contact lens wear and with certain brands of contact lens care solutions where there were formations of biofilms [7]. Biofilms are found within the lungs of patients with chronic pulmonary infarctions and in particular within patients with cystic fibrosis and are the major cause of morbidity and mortality for these patients. Antimicrobial treatment demonstrates a biphasic killing rate, which indicates the presence of a persister population, and a ready supply of nutrient throughout its depth has fewer persister bacteria and hence may be easier to treat than one with less nutrient [8].

Unwanted biofilms can create enormous increases in fluid frictional resistances, unacceptable reductions in heat transfer efficiency, product contamination, enhanced material deterioration, and accelerated corrosion. The motivation for study is not all due to the negative effects, however. Compared to suspended culture systems, intentionally engineered biofilms are heterogeneous reaction systems that can increase reactor productivity, system stability, and provide inherent cell product separation. There is an active amount of study
Figure 2.1: (a) Biofilm growth on rocks in a stream (USGS) and within a kitchen pipe (MSU Center for Biofilm Engineering). (b) Biofilm phenomena can be caused by bacteria such as *Thiobacillus ferrooxidans* (Iron Bacteria). Bacterial biofilm create oil-like films when they attach themselves to the water surface. Sunlight bounces off the films, giving them an oily appearance. (c) An image from an electron scanning microscope of a *Staphylococcus aureus* biofilm on a vascular prosthesis[1].
devoted to the growth and behavior of the more beneficial biofilms [9]. Phototropic biofilms occur on surfaces exposed to light in a range of terrestrial and aquatic environments, and have widespread applications in wastewater treatment, bioremediation, fish feed production, biohydrogen production, and soil improvement [10].

There have been some promising measures to date that show that our knowledge base is increasing. Bacteria survive in nature by forming biofilms on surfaces and most bacteria and fungi are capable of forming them. Biofilms can be prevented in some cases by antibiotic prophylaxis or early aggressive antibiotic therapy and treated by chronic suppressive antibiotic therapy. Promising strategies include the use of compounds which can easily dissolve the biofilm matrix and quorum sensing inhibitors, which increases biofilm susceptibility to antibiotics and phagocytosis [11]. Phenol is a byproduct of the industrial process of cork manufacturing, and a phenol degrading bacteria immobilized onto residual cork particles is proposed for the remediation of this industrial effluent, known as self-remediation [12]. The first study to observe the reversible redox conversion of cytochrome c552 in viable Geobacter sulfurreducens biofilms demonstrates that spectral changes were fully reversible in both positive and negative directions of the scanning potential [13].

Strategies for improving bioremediation efficiency involving genetic engineering to improve strains and chemotactic ability, the use of mixed population biofilms and optimization of physiochemical conditions have been studied [14]. Bacterial signal responsive regulatory circuits have been employed as a platform to design and construct whole cell bacterial biosensors for reporting toxicity, bioremediation, and killing targeted cells [15]. Finally, a down-well aquifer microbial sampling system has been developed using glass wool or Biosep beads as a solid phase support matrix to monitor microbial community dynamics during field bioremediation experiments at Oak Ridge National Laboratory [16].

Many other realistic problems also involve the investigation of biological films, like the biological treatment of polluted fluids. Specialized bacteria were domesticated and cultivated
with polluted stream water and this biological contact oxidation ditch system augmented with the bacteria could be a viable alternative for treating polluted stream water to achieve improved nitrogen removal [17]. Polluted surface water was remediated in a bioreactor using biofilms on filamentous bamboo in batch and continuous flow models [18]. The responses of cultured phototropic biofilms to diverse phosphorus regimes were assessed using a semi continuous flow incubator, and consequently it is proposed that for efficient nutrient removal from wastewaters, biofilms should be regularly removed to continually maintain growth at the initial stages [19].

Biological films have been previously investigated mathematically in the specialized literature. For instance, there are various mathematical models featuring systems of nonlinear PDEs [20, 21, 22, 23, 24] or cellular automata [25, 26, 27, 28] that describe the dynamics of interaction of microbial conglomerates with respect to a surrounding substrate of nutrients. The most realistic models available in the literature take into account many of the most important characteristics of biological films observed in the laboratory. Most importantly (and a major part of the models we will introduce), we will require the presence of a sharp front of biomass at the fluid/solid transition, the existence of a threshold of biomass density, the fact that the biomass spreading is significant only when the biomass is close to the threshold, and the application of reaction kinetics mechanisms in the production of biomass. Other models further require the compatibility of the biomass spreading mechanism with hydrodynamics and with nutrient transfer/consumption models, among other relevant features. For most of these mathematical models, the literature also reports on theorems of existence and uniqueness of suitable solutions which are biologically relevant. On the other hand, however, the analytical apparatus to solve such systems of PDEs is overshadowed by the complexity of the models. The analytical systems typically involve some type of nonlinear diffusion which is similar in structure to the porous medium equations, and very few analytical solutions to that category of PDE exist, much less with the additional terms involved in the reaction
kinetics. Simpler mathematical forms of the model under investigation possess analytical results which guarantee the existence and uniqueness of positive solutions, and we reference and utilize such results within the current study. However, the exact resolution of such models for experimentally relevant initial conditions is a task which is practically impossible to accomplish.

To resolve this problem, the difficulties of deriving exact analytical solutions for realistic biological film models are avoided through using computational methods that strive to faithfully approximate continuous systems. Some of the techniques used in the literature are capable of preserving important features of the continuous systems, like the positivity and the boundedness of solutions. It is important to emphasize that the conditions of positivity and boundedness are physically important characteristics in view that the populations of colonies are measured in densities (which means that negative values are meaningless) and that the media where the microbial colonies grow are spatially limited environments, respectively. The models we consider will include the presence of a substrate of nutrients, and a total biological mass. In two of these cases we consider three interacting components in the biomass. Namely, an active portion of biological mass, an inert component which largely has no function other than being a byproduct of biomass substrate consumption, and the extracellular polymeric substance, which is a protective matrix surrounding bacterial colonies that makes them resistant to both antibiotic treatment and host defense systems. The model in these two cases is a system of four nonlinear PDEs proposed first by [20], in which each of the components interacts non-trivially with the others. In the present work, we propose a finite difference discretization of the two dimensional form of this system as our final model under Neumann and Dirichlet boundary conditions, as well as two precursory models. The method proposed is a two-step, linear technique which is capable of preserving the property of positivity of the approximate solutions for relatively small temporal step-sizes. The preservation of this characteristic follows from the fact that, under such conditions, the method
is represented by an M-matrix, whose properties will be described in the following chapters. As a consequence of its properties, the matrix is nonsingular and the entries of its inverse are all positive numbers, whence the preservation of the positivity of the approximations readily follows.

In this dissertation our focus will be on the solution of the equations governing the spatiotemporal dynamics of three experimental biofilm representations. The choice of equations will be built using first principles and some phenomenological models commonly found in the literature. Specifically, a system of continuous nonlinear diffusion reaction equations will be determined in the background chapter, as well as descriptions of the state variables, meaningful parameters, and simplifications assumed. In the next chapter, we describe the three biofilm systems analytically and move on to creating the finite difference schemes relevant to the equations. Once we dissect the three experimental models into their finite difference schemes, we move on to make sure certain criteria are met to establish solutions using M-matrices. The following chapters are devoted to providing the linear finite difference discretization of our mathematical model, along with an equivalent algebraic form of it. Several remarks are stated in order to show that the vector form of our technique is represented by a square matrix which, under suitable conditions on the temporal step-size, is a nonsingular matrix for which all the entries of its inverse are positive numbers. Clearly, these conditions guarantee the preservation of the positivity of the approximations, and make our technique a useful tool in the computational investigation of complex biological films. Motivation for the general diffusion reaction equations and using M-matrices in other fields is also a significant factor in the development of the solution methods. As positivity preserving, bounded solutions are needed by other subjects, the M-matrix solution method is amenable to certain types of these mathematical models as well.

An crucial part of the third chapter is to describe the importance of boundary conditions that faithfully obey the requirements of the M-matrix. Homogeneous Dirichlet condi-
tions possess the necessary property of maintaining strict diagonal dominance by construction, but Neumann conditions require a much more careful treatment. In fact, given certain criteria, we would have to satisfy conditions of strict diagonal dominance which are no longer feasible on the corresponding boundary rows. Upon increasing the spatial dimension from one to two, we increase the number of boundary nodes where strict diagonal dominance cannot hold. The third chapter provides a method to resolve this problem in establishing a weakly chained diagonal dominance for the overall matrix, thereby ensuring that even under Neumann conditions we can still use this M-matrix method.

In the chapter on numerical implementation we describe some novel and efficient methods of solving the system of linear equations that result from the finite difference method through MATLAB®. This is followed by a chapter on illustrative results for simulations on the growth dynamics of biological films under our models. We close with a discussion and concluding remarks. Code for the simulations is included in the appendix.
3  Background

1  Model Development

Starting in the 1970s, several mathematical models were developed to link substrate flux into the biofilm to the fundamental mechanisms of substrate utilization and mass transport. In the 1980s, models still maintained a one dimensional geometry but spatial patterns for several substrates and different types of biomass were added. Today new mathematical models are being developed to provide mechanistic representations for the factors controlling the formation of complex two and three dimensional biofilm morphologies. Features included in these mathematical models usually are motivated by observations made with powerful new tools for observing biofilms in experimental systems.

The first step in creating or choosing a biofilm model is to identify the essential features of the biofilm system. These features are organized into a logical hierarchy. Compartments define the different functional components of the biofilm system for more complicated varieties. The biofilm itself must be distinguished from the overlying water and substratum to which it is attached. A mass transport boundary layer often separates the biofilm from the water. Within the compartment are the constituents, including different components of the biomass, active, structural, and inert microbial species, as well as the substrate. The components themselves can undergo transformation, transport, and transfer processes. Finally, all processes affecting each component in each compartment are mathematically linked
together into a mass balance equation that contains terms and parameters for each process. Because most biofilms are complex systems, a biofilm model that attempts to capture all the complexity would need to include mass balance equations for all processes occurring in all compartments, continuity and momentum equations for the fluid in all compartments, and defined conditions for all variables at all system boundaries [29]. Below is a general diagram for the system dynamics which we will be using for our analysis:

\[
\begin{pmatrix}
\text{Net rate of accumulation of mass of component in the system} \\
\text{Mass flow of the component into the system} \\
\text{Mass flow of the component out of the system} \\
\text{Rate of production of the component by transformations} \\
\text{Rate of consumption of the component by transformations}
\end{pmatrix} =
\begin{pmatrix}
\text{Mass flow of the component into the system} \\
\text{Mass flow of the component out of the system} \\
\text{Rate of production of the component by transformations} \\
\text{Rate of consumption of the component by transformations}
\end{pmatrix} -
\begin{pmatrix}
\text{Net rate of accumulation of mass of component in the system} \\
\text{Mass flow of the component into the system} \\
\text{Mass flow of the component out of the system} \\
\text{Rate of production of the component by transformations} \\
\text{Rate of consumption of the component by transformations}
\end{pmatrix}
\]

Throughout this discussion, we employ the notation \( \mathbb{R}^+ \) to represent the closure of \( \mathbb{R}^+ \) in the set of the real numbers with the standard topology. We begin with a simple model that describes the interaction between an active biomass and its substrate. This model corresponds to Experiment 1 (or \( \mathcal{E}_1 \) throughout the discussion), and the case of homogeneous Dirichlet conditions was published from our research in [30]. Let \( d^s \) and \( d^u \) be positive, real numbers, and let \( K_1, K_2, K_3 \) and \( K_4 \) be nonnegative numbers. Suppose that \( \alpha \) and \( \beta \) are real numbers such that \( \alpha, \beta \geq 1 \), and let \( p \) be a positive integer. Let \( \Omega \) be a subset of \( \mathbb{R}^p \) which is open, bounded and connected, and let \( s \) and \( u \) be real functions defined in \( \overline{\Omega} \times \mathbb{R}^+ \) which are twice differentiable in the interior of their domains, and that satisfy the following system of PDEs, for every \((x, t) \in \Omega \times \mathbb{R}^+\):

\[
\begin{align*}
\frac{\partial s}{\partial t}(x, t) &= d^s \nabla^2 s(x, t) - K_1 \frac{s(x, t)u(x, t)}{K_4 + s(x, t)} \\
\frac{\partial u}{\partial t}(x, t) &= d^u \nabla \cdot (D(u(x, t))\nabla u(x, t)) - K_2 u + K_3 \frac{s(x, t)u(x, t)}{K_4 + s(x, t)}.
\end{align*}
\] (3.1.1)
Here, the spatial operators $\nabla$ and $\nabla^2$ denote, respectively, the gradient/divergence and the Laplacian operators; meanwhile, the function $D : [0, 1) \rightarrow \mathbb{R}$ is given by:

$$D(f) = \frac{f^\beta}{(1 - f)^\alpha}$$

(3.1.2)

for every $f \in [0, 1)$. Appropriate initial/boundary conditions are required and for Dirichlet conditions we additionally impose:

$$s(x, t) = 1, \quad u(x, t) = 0, \quad \forall x \in \partial\Omega, \forall t \geq 0,$$

$$s(x, 0) = s^0(x), \quad u(x, 0) = u^0(x), \quad \forall x \in \Omega,$$

(3.1.3)

for suitable functions $s^0, u^0 : \overline{\Omega} \rightarrow \mathbb{R}$. In the case of Neumann conditions we have:

$$\hat{n} \cdot \nabla s(x, t) = 0, \quad \hat{n} \cdot \nabla u(x, t) = 0, \quad \forall x \in \partial\Omega, \forall t \geq 0,$$

$$s(x, 0) = s^0(x), \quad u(x, 0) = u^0(x), \quad \forall x \in \Omega,$$

(3.1.4)

for suitable functions $s^0, u^0 : \overline{\Omega} \rightarrow \mathbb{R}$.

In the context of the investigation of microbial biology, (3.1.1) describes the dynamics of interaction between a colony of bacteria whose biomass density at the point $x \in \Omega$ and time $t$ is given by $u(x, t)$, and the corresponding substrate concentration containing the nutrients that are beneficial to the colony which is given by $s(x, t)$. In this case, the parameters $d^s$, $d^u$, $K_1$, $K_2$, $K_3$ and $K_4$ represent, respectively, the substrate diffusion coefficient, the biomass diffusion coefficient, the maximum specific consumption rate, the biomass decay rate, the maximum specific growth rate, and the Monod half saturation constant. The reaction kinetic component of (3.1.1) follows Monod kinetics, a phenomenological model of growth for bacterial colonies very similar to the Michaelis Menten equation which relates the reaction rate of substrate consumption to concentration and availability.
Experiment 2 (E_2) and Experiment 3 (E_3) will focus on the following equations, where we now subdivide the biomass into three separate components, namely the active biomass \(X\), inert biomass \(I\), and extracellular polymeric matrix \(E\), where the mass sum total of all biofilm components is \(U = X + I + E\):

\[
\begin{align*}
\frac{\partial S}{\partial t}(x, t) &= d^S \nabla \cdot (D(U(x, t)) \nabla S(x, t)) - \frac{\mu}{Y_H} \frac{XS}{\kappa_S + S} + \frac{\kappa_L \kappa_S X}{\kappa_S + S} + \frac{\kappa_E \kappa_S E}{\kappa_S + S} \\
\frac{\partial X}{\partial t}(x, t) &= d^U \nabla \cdot (D(U(x, t)) \nabla X(x, t)) + \frac{\mu XS}{\kappa_S + S} - \frac{\kappa_L \kappa_S X}{\kappa_S + S} - \frac{\kappa_I \kappa_S X}{\kappa_S + S} \\
\frac{\partial I}{\partial t}(x, t) &= d^I \nabla \cdot (D(U(x, t)) \nabla I(x, t)) + \frac{\kappa_I \kappa_S X}{\kappa_S + S} \\
\frac{\partial E}{\partial t}(x, t) &= d^E \nabla \cdot (D(U(x, t)) \nabla E(x, t)) + \frac{Y_E \mu XS}{\kappa_S + S} - \frac{\kappa_E \kappa_S E}{\kappa_S + S}
\end{align*}
\]

Here \(D\) is defined as in (3.1.2). With all parameters being defined as nonnegative constants, we take \(Y\) to refer to any one of our four state variables \(\{S, X, I, E\}\), so that \(d^Y\) refers to the diffusion coefficient for \(Y\) (in our description we only use \(d^S\) and \(d^U\) as will explained later), \(\kappa_S\) is the Monod half saturation constant for the substrate, \(\mu\) is the maximum specific growth rate of active biomass, \(Y_H\) and \(Y_E\) the yield ratios of biomass grown to substrate consumed and EPS formed to substrate consumed, respectively, \(\kappa_L\) is the biomass decay rate, and \(\kappa_I\) and \(\kappa_E\) are the growth coefficients for inert biomass and EPS. We will once again impose Dirichlet boundary conditions such that:

\[
\begin{align*}
Y(x, t) &= 0, \quad \forall x \in \partial \Omega, \forall t \geq 0 \\
Y(x, 0) &= Y^0(x), \quad \forall x \in \Omega
\end{align*}
\]

(3.1.6)

and Neumann conditions where:

\[
\begin{align*}
\hat{n} \cdot \nabla Y(x, t) &= 0, \quad \forall x \in \partial \Omega, \forall t \geq 0, \\
Y(x, 0) &= Y^0(x), \quad \forall x \in \Omega
\end{align*}
\]

(3.1.7)

12
This system of PDEs can be adapted to multiple spatial dimensions, but we will restrict the present study to the spatially one ($\mathbb{R}^2$) and two ($\mathbb{R}^3$) dimensional cases.

2 Analytical Results

Let $F$ be the real function defined on $[0,1)$ through the expression

$$F(u) = \int_0^u \frac{v^\beta}{(1-v)^\alpha} dv.$$  \hspace{1cm} (3.2.8)

Our work is greatly motivated by the next result that establishes conditions under which nonnegative and bounded solutions of (3.1.1) with boundary conditions from (3.1.3) or (3.1.4) exist and are unique; its proof is a direct consequence of Theorems 2.1 and 2.2 of [21] (repeated below):

**Proposition 2.1.** Let $s^0$ and $u^0$ satisfy the following conditions:

(A) $s^0 \in L^\infty(\Omega) \cap H^1(\Omega)$ and $0 \leq s^0(x) \leq 1$ for every $x \in \Omega$,

(B) $u^0 \in L^\infty(\Omega)$ and $F \circ u^0 \in H^1_0(\Omega)$

(C) $u^0(x) \geq 0$ for every $x \in \Omega$, and $\|u^0\|_{L^\infty(\Omega)} < 1$.

Then, there exists a unique solution of the problem (3.1.1) satisfying the following properties:

1. $s, u \in L^\infty(\Omega \times \mathbb{R}^+) \cap C(L^2(\Omega), [0, \infty))$,
2. $s, F \circ u \in L^\infty(H^1(\Omega), \mathbb{R}^+) \cap C(L^2(\Omega), [0, \infty))$,
3. $0 \leq s(x,t), u(x,t) \leq 1$ for every $(x,t) \in \Omega \times \mathbb{R}^+$, and $\|u\|_{L^\infty(\Omega \times \mathbb{R}^+)} < 1$.

It is important to mention that the system of equations (3.1.1) was derived in [20] using several mathematical assumptions which are motivated by experimental results. Among other physical hypotheses, the derivation of the model of interest accounted for:
(A) the presence of a sharp front of biomass at the fluid/solid transition,
(B) the existence of a threshold of biomass density,
(C) the fact that the biomass spreading is significant only when the biomass concentration is close to the threshold,
(D) the application of reaction kinetics mechanisms in the production of biomass,
(E) the compatibility of the biomass spreading mechanism with hydrodynamics and with nutrient transfer/consumption models.

One readily notices that the expression of the diffusion factor $D$, as given by (3.1.2), satisfies conditions (B) and (C) above. On the other hand, the mathematical assumptions of Proposition 2.1 yield sufficient conditions for the existence and uniqueness of solutions of (3.1.1) which satisfy the physical constraints.
4 Computational Models Investigated

1 Basic Definitions

We begin with the following definitions in order to discretize the three experiments investigated, and will cover the numerical methods and procedures involved in all three experiments at the same time to avoid restatement. We establish the quantities \( \{K, M, N\} \in \mathbb{Z}^+ \), our spatial boundary \( \Omega^p \in \mathbb{R}^p \) for \( p \in \{1, 2\} \) where \( \Omega^1 = [a, b] \) and \( \Omega^2 = [a, b] \times [c, d], \{a, b, c, d\} \in \mathbb{R}, a < b, \) and \( c < d \). We fix uniform partitions of the intervals \([a, b], [c, d]\) of the form:

\[
a = x_0 < x_1 < \cdots < x_m < \cdots < x_M = b
\] (4.1.1)

and

\[
c = y_0 < y_1 < \cdots < y_n < \cdots < y_N = d
\] (4.1.2)

for every \( m \in \{0, 1, \ldots, M\} \) and \( n \in \{0, 1, \ldots, N\} \). Let \( \Delta x \) and \( \Delta y \) represent the spatial step sizes in the \( x \) and \( y \) directions, respectively, where \( \Delta x = (b-a)/M \) and \( \Delta y = (d-c)/N \). Fix the temporal period of length equal to \( T \in \mathbb{R}^+ \), and take a uniform partition of the interval \([0, T]\) of the form

\[15\]
\begin{equation}
0 = t_0 < t_1 < \cdots < t_k < \cdots < t_K = T \tag{4.1.3}
\end{equation}

for every \( k \in \{0, 1, \ldots, K\} \), the norm being \( \Delta t = T/K \). The definitions for all three experiments will use these quantities with the restriction of \( \mathcal{E}_2 \) being in only the \( x \) spatial dimension.

We adopt the typical convention for the biomass state variables for all three experiments: \( \mathcal{E}_1 \) modeling the interaction of substrate \( s \) and active biomass \( u \) in \( \Omega^2 \times \mathbb{R}^+ \) as \( s_{m,n}^k, u_{m,n}^k, \mathcal{E}_2 \) in \( \Omega^1 \times \mathbb{R}^+ \) but with the addition of inert biomass \( I \) and EPS \( E \) (through the components \( S_m^k, X_m^k, I_m, E_m \)) and \( \mathcal{E}_3 \) in \( \Omega^2 \times \mathbb{R}^+ \) with \( S_{m,n}^k, X_{m,n}^k, I_{m,n}^k, E_{m,n}^k \) to represent approximations to exact values of \( S, X, I, \) and \( E \) respectively at the point \((x_m, y_n, t_k)\) for each \( m \in \{0, 1, \ldots, M\}, \ n \in \{0, 1, \ldots, N\}, \) and \( k \in \{0, 1, \ldots, K\} \). \( U_m^k \) for \( \mathcal{E}_2 \) is defined as the sum of the constituent biomass state variables without substrate:

\begin{equation}
U_m^k = X_m^k + I_m + E_m \tag{4.1.4}
\end{equation}

whereas in \( \mathcal{E}_3 \) we define the sum \( U_{m,n}^k \) as:

\begin{equation}
U_{m,n}^k = X_{m,n}^k + I_{m,n} + E_{m,n}^k \tag{4.1.5}
\end{equation}

2 Finite Difference Operators

We make the following definitions to simplify the numerical finite difference method, where \( Y \) is any one of our state variables for \( S, X, I, E \) in \( \mathcal{E}_2 \), remembering \( m \in \{1, \ldots, M - 1\} \) and \( k \in \mathbb{Z}^+ \cup \{0\} \):
\[ \delta^+_t Y^k_m = \frac{Y^{k+1}_m - Y^k_m}{\Delta t_k} \] (4.2.6)

The above equation (4.2.6) is a standard forward difference in time operator.

\[ \delta^+_x Y^k_m = \frac{Y^{k+1}_{m+1} - Y^k_m}{\Delta x} \] (4.2.7)

Similarly, (4.2.7) corresponds to the standard forward (backward) difference in space operators.

\[ \mu^+_x Y^k_m = \frac{Y^{k+1}_{m+1} + Y^k_m}{2} \] (4.2.8)

(4.2.8) Acts as an averaging operator on the state variables.

\[ \epsilon^+_x Y^k_m = D(\mu^+_x U^k_m) \delta^+_x Y^{k+1}_m \] (4.2.9)

\[ \epsilon_x Y^k_m = \frac{\epsilon^+_x Y^k_m + \epsilon^-_x Y^k_m}{\Delta x} \] (4.2.10)

Definitions eqs. (4.2.8)–(4.2.10) are operators somewhat more specific to the problem that greatly simplify the finite difference equations.

The definitions for the finite difference operators change only slightly for \( \mathcal{E}_1 \) and \( \mathcal{E}_3 \), and are included below, with \( Y^k_{m,n} \) representing any one of \{u, s, S, X, I, E\} for \( m \in \{1, \ldots, M - 1\} \), \( n \in \{1, \ldots, N - 1\} \), \( k \in \mathbb{Z}^+ \cup \{0\} \), and \( z \in \{x, y\} \):
\[
\delta_{t}^{+} Y_{m,n}^{k} = \frac{Y_{m,n}^{k+1} - Y_{m,n}^{k}}{\Delta t_{k}} \quad (4.2.11)
\]

\[
\delta_{x}^{\pm} Y_{m,n}^{k} = \frac{Y_{m\pm1,n}^{k} - Y_{m,n}^{k}}{\Delta x} \quad (4.2.12)
\]

\[
\delta_{y}^{\pm} Y_{m,n}^{k} = \frac{Y_{m,n\pm1}^{k} - Y_{m,n}^{k}}{\Delta y} \quad (4.2.13)
\]

\[
\mu_{x}^{\pm} Y_{m,n}^{k} = \frac{Y_{m\pm1,n}^{k} + Y_{m,n}^{k}}{2}, \quad \mu_{y}^{\pm} Y_{m,n}^{k} = \frac{Y_{m,n\pm1}^{k} + Y_{m,n}^{k}}{2} \quad (4.2.14)
\]

\[
\epsilon_{z}^{\pm} Y_{m,n}^{k} = D(\mu_{z}^{\pm} Y_{m,n}^{k}) \delta_{z}^{\pm} Y_{m,n}^{k+1} \quad (4.2.15)
\]

\[
\epsilon_{z} Y_{m,n}^{k} = \frac{\epsilon_{z}^{+} Y_{m,n}^{k} + \epsilon_{z}^{-} Y_{m,n}^{k}}{\Delta z} \quad (4.2.16)
\]

It should be understood that operator eqs. (4.2.11)–(4.2.16) are just eqs. (4.2.6)–(4.2.10) in two spatial dimensions operating on those dimensions separately. Finally we define the second order operators (4.2.17) and (4.2.18):

\[
\delta_{x}^{2} Y_{m,n}^{k} = \frac{Y_{m+1,n}^{k} - 2Y_{m,n}^{k} + Y_{m-1,n}^{k}}{(\Delta x)^2} \quad (4.2.17)
\]

18
\[ \delta^2 Y_{m,n}^k = \frac{Y_{m,n+1}^k - 2Y_{m,n}^k + Y_{m,n-1}^k}{(\Delta y)^2} \]  

(4.2.18)

3 Construction of Finite Difference Equations

The fully discretized system of eq. (3.1.1) in \( E_1 \) now reads as:

\[
\begin{align*}
\delta_t^+ s_{m,n}^k &= d^s(\delta_x^2 + \delta_y^2)s_{m,n}^{k+1} - K_1 \frac{u_{m,n}^k s_{m,n}^{k+1}}{K_4 + s_{m,n}^k} \\
\delta_t^+ u_{m,n}^k &= d^u(\epsilon_x + \epsilon_y)u_{m,n}^k - K_2 X_{m,n}^k + K_3 \frac{s_{m,n}^k u_{m,n}^{k+1}}{K_4 + s_{m,n}^k}
\end{align*}
\]  

(4.3.19)

Experiments \( E_2 \) and \( E_3 \) corresponding to eq. (3.1.5), which are similar in structure, are both shown below. For \( E_2 \) we have:

\[
\begin{align*}
\delta_t^+ s_{m,n}^k &= d^s \epsilon_x s_{m,n}^k - a_1 \frac{X_m^k S_{m,n}^{k+1}}{\kappa_S + S_{m,n}^k} + a_2 \frac{X_m^k}{\kappa_S + S_{m,n}^k} + a_3 \frac{E_m^k}{\kappa_S + S_{m,n}^k} \\
\delta_t^+ X_{m,n}^k &= d^u \epsilon_x X_{m,n}^k + \mu \frac{X_m^k S_{m,n}^{k+1}}{\kappa_S + S_{m,n}^k} - a_4 \frac{X_{m,n}^k}{\kappa_S + S_{m,n}^k} \\
\delta_t^+ I_{m,n}^k &= d^u \epsilon_x I_{m,n}^k + a_4 \frac{X_{m,n}^k}{\kappa_S + S_{m,n}^k} \\
\delta_t^+ E_{m,n}^k &= d^u \epsilon_x E_{m,n}^k + a_5 \frac{X_m^k S_{m,n}^{k+1}}{\kappa_S + S_{m,n}^k} - a_3 \frac{E_{m,n}^k}{\kappa_S + S_{m,n}^k}
\end{align*}
\]  

(4.3.20)

and for \( E_3 \):
\[
\begin{align*}
\delta^+_t S^k_{m,n} &= d^S(\epsilon_x + \epsilon_y) S^k_{m,n} - a_1 \frac{X^k_{m,n} S^k_{m,n+1}}{\kappa_S + S^k_{m,n}} + a_2 \frac{X^k_{m,n}}{\kappa_S + S^k_{m,n}} + a_3 \frac{E^k_{m,n}}{\kappa_S + S^k_{m,n}}, \\
\delta^+_t X^k_{m,n} &= d^U(\epsilon_x + \epsilon_y) X^k_{m,n} + \mu \frac{X^k_{m,n} S^k_{m,n+1}}{\kappa_S + S^k_{m,n}} - a_4 \frac{X^{k+1}_{m,n}}{\kappa_S + S^k_{m,n}}, \\
\delta^+_t I^k_{m,n} &= d^U(\epsilon_x + \epsilon_y) I^k_{m,n} + a_4 \frac{X^{k+1}_{m,n}}{\kappa_S + S^k_{m,n}}, \\
\delta^+_t E^k_{m,n} &= d^U(\epsilon_x + \epsilon_y) E^k_{m,n} + a_5 \frac{X^k_{m,n} S^k_{m,n+1}}{\kappa_S + S^k_{m,n}} - a_3 \frac{E^{k+1}_{m,n}}{\kappa_S + S^k_{m,n}},
\end{align*}
\]
Neumann boundary conditions will require that the normal derivative to the boundary is 0, so we need to make a suitable approximation. \( E_2 \) is one dimensional so the approximate derivative to eq. (3.1.7) in \( x \) gives us the conditions for all \( k \in \{0, 1, \ldots, K\} \):

\[
\frac{\partial Y}{\partial x} \bigg|_{x \in \partial \Omega} = 0 \quad \Rightarrow \quad \frac{\partial Y}{\partial x} \approx \frac{Y^k - Y^0}{\Delta x} = 0 \quad \Rightarrow \quad Y^k_1 - Y^0_0 = 0, \\
\frac{\partial Y}{\partial x} \bigg|_{x \in \partial \Omega} = 0 \quad \Rightarrow \quad \frac{\partial Y}{\partial x} \approx \frac{Y^k_M - Y^k_{M-1}}{\Delta x} = 0 \quad \Rightarrow \quad Y^k_M - Y^k_{M-1} = 0 
\]

(4.3.25)

For the two dimensional Neumann boundaries in \( E_1 \) and \( E_3 \) for eq. (3.1.7) we have similar conditions, but we need to take linear combinations of endpoints at the corners:

\[
Y^k_{m,0} - Y^k_{m,1} = 0, \quad Y^k_{m,N} - Y^k_{m,N-1} = 0 \quad \forall m \in \{1, 2, \ldots, M - 1\}, \\
Y^k_{1,n} - Y^k_{0,n} = 0, \quad Y^k_{M,n} - Y^k_{M-1,n} = 0 \quad \forall n \in \{1, 2, \ldots, N - 1\}, \quad \text{and} \\
2Y^k_{0,0} - Y^k_{0,1} - Y^k_{1,0} = 0, \\
2Y^k_{0,N} - Y^k_{0,N-1} - Y^k_{1,N} = 0, \\
2Y^k_{M,0} - Y^k_{M-1,0} - Y^k_{M,1} = 0, \\
2Y^k_{M,N} - Y^k_{M-1,N} - Y^k_{M,N-1} = 0
\]

(4.3.26)

We also require that for all \( m \in \{0, 1, \ldots, M\} \) and \( n \in \{0, 1, \ldots, N\} \), we have the following for \( E_1, E_2, \) and \( E_3 \):

\[
0 \leq s^0_{m,n} \leq 1, \quad 0 \leq u^0_{m,n} < 1 
\]

(4.3.27)

\[
0 \leq S^0_m \leq 1, \quad 0 \leq X^0_m < 1, \quad 0 \leq I^0_m < 1, \quad 0 \leq E^0_m < 1, \\
U^0_m = X^0_m + I^0_m + E^0_m < 1
\]

(4.3.28)
0 ≤ S^0_{m,n} ≤ 1, 0 ≤ X^0_{m,n} < 1, 0 ≤ I^0_{m,n} < 1, 0 ≤ E^0_{m,n} < 1
U^0_{m,n} = X^0_{m,n} + I^0_{m,n} + E^0_{m,n} < 1  \quad (4.3.29)

This ensures that the nonlinear diffusion mechanism does not attain its infinite discontinuity.

Our overall system will be composed of the iterative solution to a $2(M+1)(N+1) \times 2(M+1)(N+1)$ M-matrix in $\mathcal{E}_1$, a $4(M+1) \times 4(M+1)$ M-matrix in $\mathcal{E}_2$, and a $4(M+1)(N+1) \times 4(M+1)(N+1)$ M-matrix in $\mathcal{E}_3$. After defining the term for each $k \in \{0,1,\ldots,K-1\}$

$$R^k_z Y = d^Y \frac{\Delta t^k}{(\Delta z)^2}, \quad z \in \{x,y\}, \quad Y \in \{s,u,S,U\} \quad (4.3.30)$$

we can begin to examine the constituent equations row-wise for each state variable in all experiments. We first define the discrete form of the state variables for $s(x,y,t) = s^k_{m,n}$ and $u(x,y,t) = u^k_{m,n}$ of $\mathcal{E}_1$. After some algebraic manipulations, we have:

$$-R^k_x s^k_{m-1,n} - R^k_y s^k_{m,n-1} + \phi^k_{m,n} s^{k+1}_{m,n} - R^k_y s^{k+1}_{m,n+1} - R^k_x s^{k+1}_{m+1,n} = s^k_{m,n} \quad (4.3.31)$$

$$\psi^k_{m,n,x} u^{k+1}_{m-1,n} + \psi^k_{m,n,y} u^{k+1}_{m,n-1} + \chi^k_{m,n} u^{k+1}_{m,n} + \psi^k_{m,n,y} u^{k+1}_{m,n+1} + \psi^k_{m,n,x} u^{k+1}_{m+1,n} = u^k_{m,n} \quad (4.3.32)$$

Here the sets of equations run through all $m = \{1,2,\ldots,M-1\}$, $n = \{1,2,\ldots,N-1\}$, and $k = \{0,1,\ldots,K-1\}$ where $\phi^k_{m,n}$, $\psi^k_{m,n,z}$, and $\chi^k_{m,n}$ are defined below.
\[
\phi_{m,n}^k = 1 + 2R_{x,s}^k + 2R_{y,s}^k + K_1 \Delta t_k \frac{\eta_{m,n}^k}{K_4 + s_{m,n}^k} \tag{4.3.33}
\]
\[
\psi_{m,n,z}^{k,\pm} = -R_{z,u}^k D(\mu_{z,u}^k, m_{m,n}), \tag{4.3.34}
\]
\[
\chi_{m,n}^k = 1 - \psi_{m,n,x}^{k,-} - \psi_{m,n,y}^{k,-} - \psi_{m,n,x}^{k,+} - \psi_{m,n,y}^{k,+} + K_2 \Delta t_k - K_3 \Delta t_k \frac{s_{m,n}^k}{K_4 + s_{m,n}^k} \tag{4.3.35}
\]

**Remark 3.1.** By inspection, it is easy to see that:

1. Every \( \phi_{m,n}^k \) is positive.

2. Every \(-R_{z,s}^k\) and \(\psi_{m,n,z}^{k,\pm}\) are negative.

3. If we ensure that \(K_3 \Delta t_k < 1 + K_2 \Delta t_k\), then we have:

\[
\chi_{m,n}^k \geq 1 + K_2 \Delta t_k - K_3 \Delta t_k \frac{s_{m,n}^k}{K_4 + s_{m,n}^k} \geq 1 + K_2 \Delta t_k - K_3 \Delta t_k > 0, \tag{4.3.36}
\]

Therefore every \( \chi_{m,n}^k \) is positive.

In turn we examine the constituent equations row-wise for the each state variable in \( \mathcal{E}_2 \). After some algebraic manipulations, we have:

\[
\alpha_m^k - \alpha_{m-1}^k + \beta_m^k S_{m}^{k+1} + \alpha_m^k \xi_{m+1}^{k+1} + \zeta_m^k X_m^{k+1} + \eta_m^k E_m^{k+1} = S_m^k \tag{4.3.37}
\]
\[
\alpha_m^k X_m^{k+1} + \beta_m^k X_{m-1}^{k+1} + \alpha_m^k X_{m+1}^{k+1} + \gamma_m^k S_m^{k+1} = X_m^k \tag{4.3.38}
\]
\[
\alpha_m^k I_m^{k+1} + \beta_m^k I_{m-1}^{k+1} + \alpha_m^k I_{m+1}^{k+1} + \zeta_m^k X_m^{k+1} = I_m^k \tag{4.3.39}
\]
\[
\alpha_m^k E_m^{k+1} + \beta_m^k E_{m-1}^{k+1} + \alpha_m^k E_{m+1}^{k+1} + \gamma_m^k X_m^{k+1} = E_m^k \tag{4.3.40}
\]

The constants, which only depend on the current values of the state variables, are defined below:
\[ \alpha_{m}^{k,\pm} = -R_{x}^{k,Y}(\mu_{x}^{\pm}U_{m}^{k}) \] (4.3.41)

\[ \beta_{m}^{k,S} = 1 - \alpha_{m}^{k,-} - \alpha_{m}^{k,+} + a_{1} \frac{\Delta t_{k}X_{m}}{\kappa_{S} + S_{m}^{k}} \] (4.3.42)

\[ \beta_{m}^{k,X} = 1 - \alpha_{m}^{k,-} - \alpha_{m}^{k,+} + a_{4} \frac{\Delta t_{k}}{\kappa_{S} + S_{m}^{k}} \] (4.3.43)

\[ \beta_{m}^{k,E} = 1 - \alpha_{m}^{k,-} - \alpha_{m}^{k,+} + a_{3} \frac{\Delta t_{k}}{\kappa_{S} + S_{m}^{k}} \] (4.3.45)

\[ \beta_{m}^{k,Y} = 1 - \alpha_{m}^{k,-} - \alpha_{m}^{k,+} + a_{4} \frac{\Delta t_{k}}{\kappa_{S} + S_{m}^{k}} \] (4.3.44)

Remark 3.2. By inspection, it is easy to see that:

1. Every \( \alpha_{m}^{k,\pm} \) is negative.

2. Every \( \zeta_{m,j}^{k,1}, \gamma_{m,j}^{k,1}, \) and \( \eta_{m}^{k} \) are negative.

3. Every \( \beta_{m}^{k,Y} \) is positive.

Finally, for \( \mathcal{E}_{3} \) and just an expansion in spatial dimensions of \( \mathcal{E}_{2} \), we have:

\[ \alpha_{m,n,x}^{k,-}S_{m-1,n}^{k+1} + \alpha_{m,n,y}^{k,-}S_{m-1,n}^{k+1} + \beta_{m,n,x}^{k,S}S_{m,n}^{k+1} + \alpha_{m,n,y}^{k,+}S_{m,n+1}^{k+1} \]
\[ + \alpha_{m,n,x}^{k,+}S_{m+1,n}^{k+1} + \epsilon_{m,n}^{k,1}X_{m,n}^{k+1} + \eta_{m,n}^{k}E_{m,n}^{k+1} = S_{m,n}^{k} \] (4.3.51)

\[ \alpha_{m,n,x}^{k,-}X_{m-1,n}^{k+1} + \alpha_{m,n,y}^{k,-}X_{m-1,n}^{k+1} + \beta_{m,n,x}^{k,X}X_{m,n}^{k+1} + \alpha_{m,n,y}^{k,+}X_{m,n+1}^{k+1} \]
\[ + \alpha_{m,n,x}^{k,+}X_{m+1,n}^{k+1} + \gamma_{m,n}^{k,1}S_{m,n}^{k+1} = X_{m,n}^{k} \] (4.3.52)

24
\[ \alpha_{m,n,x}^k I_{m-1,n}^{k+1} + \alpha_{m,n,y}^k I_{m,n-1}^{k+1} + \beta_{m,n}^k I_{m,n}^{k+1} + \alpha_{m,n,y}^k I_{m,n+1}^{k+1} + \alpha_{m,n,x}^k I_{m+1,n}^{k+1} + \zeta_{m,n}^2 x_{m,n}^{k+1} = I_{m,n}^k \] (4.3.53)

\[ \alpha_{m,n,x}^k E_{m-1,n}^{k+1} + \alpha_{m,n,y}^k E_{m,n-1}^{k+1} + \beta_{m,n}^k E_{m,n}^{k+1} + \alpha_{m,n,y}^k E_{m,n+1}^{k+1} + \alpha_{m,n,x}^k E_{m+1,n}^{k+1} + \gamma_{m,n}^2 X_{m,n}^{k+1} = E_{m,n}^k \] (4.3.54)

Constants are defined below.

\[ \alpha_{m,n,z}^{k, \pm} = -R_z^k Y_m(\mu^z U_m^k) \] (4.3.55)

\[ \beta_{m,n}^{k,Y} = 1 - \alpha_{m,n,x}^k - \alpha_{m,n,x}^k - \alpha_{m,n,y}^k - \alpha_{m,n,y}^k + a_1 \frac{\Delta t_k X_{m,n}^k}{\kappa_S + S_{m,n}^k} \] (4.3.56)

\[ \beta_{m,n}^{k,X} = 1 - \alpha_{m,n,x}^k - \alpha_{m,n,x}^k - \alpha_{m,n,y}^k - \alpha_{m,n,y}^k + a_4 \frac{\Delta t_k}{\kappa_S + S_{m,n}^k} \] (4.3.57)

\[ \beta_{m,n}^{k,I} = 1 - \alpha_{m,n,x}^k - \alpha_{m,n,x}^k - \alpha_{m,n,y}^k - \alpha_{m,n,y}^k \] (4.3.58)

\[ \beta_{m,n}^{k,E} = 1 - \alpha_{m,n,x}^k - \alpha_{m,n,x}^k - \alpha_{m,n,y}^k - \alpha_{m,n,y}^k + a_3 \frac{\Delta t_k}{\kappa_S + S_{m,n}^k} \] (4.3.59)

\[ \zeta_{m,n}^{k,1} = -a_2 \frac{\Delta t_k}{\kappa_S + S_{m,n}^k} \] (4.3.60)

\[ \eta_{m,n}^k = -a_3 \frac{\Delta t_k}{\kappa_S + S_{m,n}^k} \] (4.3.61)

\[ \gamma_{m,n}^{k,1} = -\mu \frac{\Delta t_k X_{m,n}^k}{\kappa_S + S_{m,n}^k} \] (4.3.62)

\[ \zeta_{m,n}^{k,2} = -a_4 \frac{\Delta t_k}{\kappa_S + S_{m,n}^k} \] (4.3.63)

\[ \gamma_{m,n}^{k,2} = -a_5 \frac{\Delta t_k X_{m,n}^k}{\kappa_S + S_{m,n}^k} \] (4.3.64)

**Remark 3.3.** By inspection, it is easy to see that:

1. Every \( \alpha_{m,n,z}^{k, \pm} \) is negative.
2. Every \( \zeta_{m,n}^{k,1} \), \( \gamma_{m,n}^{k,1} \), and \( \eta_{m,n}^k \) are negative.
3. Every \( \beta_{m,n}^{k,Y} \) is positive.
4 Assembly of M-matrices

Now that we have established the equations corresponding to each of our experiments, we are ready to assemble the matrices which will in turn be used to progress the evolution of the biofilm complexes forward in time. Up to this point we have assembled the main coefficients out of the finite difference equations, but have not indicated the functional dependence of them on the previous state of the model. The definitions below serve to describe the ordering of the nodal points as vectors, and in our matrix definitions and vector approximations it should be implicitly understood that there is a functional dependence on the preceding state vector. This is only for notational convenience. As before we start with $E_1$. Let $k$ be an element of $\{0,1,\ldots,K\}$, and let $s^k_{E_1}$ and $u^k_{E_1}$ be the ordered vector approximations of the state variables at time $t_k$, defined as:

$$s^k_{E_1} = (s^k_{0,0}, s^k_{0,1}, \ldots, s^k_{0,N}, s^k_{1,0}, s^k_{1,1}, \ldots, s^k_{1,N}, \ldots, s^k_{M,0}, s^k_{M,1}, \ldots, s^k_{M,N})$$

(4.4.65)

$$u^k_{E_1} = (u^k_{0,0}, u^k_{0,1}, \ldots, u^k_{0,N}, u^k_{1,0}, u^k_{1,1}, \ldots, u^k_{1,N}, \ldots, u^k_{M,0}, u^k_{M,1}, \ldots, u^k_{M,N})$$

(4.4.66)

Let $v^k_{E_1} = (s^k_{E_1} | u^k_{E_1})^t$, and further let $s^0_{E_1}$ and $u^0_{E_1}$ be defined as:

$$s^0_{E_1} = (0, 0, \ldots, 0, s^0_{1,1}, \ldots, s^0_{1,N-1}, 0, \ldots, \ldots, 0, s^0_{M-1,1}, \ldots, s^0_{M-1,N-1}, 0, 0, \ldots, 0)$$

(4.4.67)
Finally we let \( \mathbf{v}_{\delta_1}^0 = (\mathbf{s}_{\delta_1}^0 | \mathbf{u}_{\delta_1}^0)^t \). By construction it should be noted that both \( \mathbf{v}_{\delta_1}^k \) and \( \mathbf{v}_{\delta_1}^0 \) are length \( 2(M+1)(N+1) \) vectors. Each matrix and submatrix will be assigned its own unique capital letter and given a reference number for the proofs to follow. As we will be defining block matrices based on not only eqs. (4.3.31) and (4.3.32), we must include the boundary conditions on the values corresponding to indices not contained in the interior of the domain as indicated in (4.3.26). We will use the indicator function defined as follows to denote the difference in the matrix if it is used for the Dirichlet or the Neumann conditions, where \( \mathcal{N} \) refers to the set of functions whose closed boundary obeys Neumann conditions, and if not they obey Dirichlet conditions:

\[
\mathbb{1}_{\mathcal{N}}(\partial \Omega) := \begin{cases} 
1 & \text{if } \partial \Omega \in \mathcal{N}, \\
0 & \text{if } \partial \Omega \notin \mathcal{N}.
\end{cases}
\]  

Additionally, we let \( \mathcal{I} \) represent the identity matrix of size \( (N+1) \times (N+1) \) and \( \mathcal{I}_{\mathcal{N}} \) represent the modified identity matrix of size \( (N+1) \times (N+1) \) defined below:

\[
\mathcal{I}_{\mathcal{N}} = \begin{pmatrix}
1 + \mathbb{1}_{\mathcal{N}}(\partial \Omega) & -\mathbb{1}_{\mathcal{N}}(\partial \Omega) & 0 & \cdots & 0 & 0 & 0 \\
0 & 1 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 1 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 1 & 0 & 0 \\
0 & 0 & 0 & \cdots & 0 & 1 & 0 \\
0 & 0 & 0 & \cdots & 0 & -\mathbb{1}_{\mathcal{N}}(\partial \Omega) & 1 + \mathbb{1}_{\mathcal{N}}(\partial \Omega)
\end{pmatrix}
\]  

(4.4.70)
For every $m \in \{1, 2, \ldots, M - 1\}$ and every $k \in \{0, 1, \ldots, K\}$, let $C^k$ and $D^k_m$ be the diagonal and tridiagonal matrices of size $(N + 1) \times (N + 1)$ given by

$$C^k = \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & -R_{k,s}^k & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & -R_{k,s}^k & 0 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & -R_{k,s}^k & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & -R_{k,s}^k & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0
\end{pmatrix}$$

(4.4.71)

and

$$D^k_m = \begin{pmatrix}
1 & -\mathbb{1}_N(\partial \Omega) & 0 & 0 & \cdots & 0 & 0 & 0 \\
-R_{y}^k & \phi_{m,1}^k & -R_{y}^k & 0 & \cdots & 0 & 0 & 0 \\
0 & -R_{y}^k & \phi_{m,2}^k & -R_{y}^k & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & \phi_{m,N-2}^k & -R_{y}^k & 0 \\
0 & 0 & 0 & 0 & \cdots & -R_{y}^k & \phi_{m,N-1}^k & -R_{y}^k \\
0 & 0 & 0 & 0 & \cdots & 0 & -\mathbb{1}_N(\partial \Omega) & 1
\end{pmatrix}$$

(4.4.72)

Now we define the square matrix $A^k_{\delta_1}$ as the block matrix of size $(M + 1)(N + 1) \times (M + 1)(N + 1)$ given by the following, keeping mind that the zeros correspond in this matrix to size $(N + 1) \times (N + 1)$ matrices:
\[
A_k^{\ell_1} = \begin{pmatrix}
I_N - \mathbb{1}_{\mathcal{A}}(\partial \Omega) I & 0 & 0 & \cdots & 0 & 0 & 0 \\
C_k & D_1^k & C_k & 0 & \cdots & 0 & 0 \\
0 & C_k & D_2^k & C_k & \cdots & 0 & 0 \\
0 & 0 & C_k & D_3^k & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & D_{M-2}^k & C_k & 0 \\
0 & 0 & 0 & \cdots & C_k & D_{M-1}^k & C_k \\
0 & 0 & 0 & \cdots & 0 & -\mathbb{1}_{\mathcal{A}}(\partial \Omega) I & I_N \\
\end{pmatrix}
\] (4.4.73)

In that same fashion, we create the submatrices \(F_m^{k,\pm}\) and \(G_m^k\) of size \((N + 1) \times (N + 1)\)

\[
F_m^{k,\pm} = \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & \psi_{m,1,x}^{k,\pm} & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & \psi_{m,2,x}^{k,\pm} & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & \psi_{m,3,x}^{k,\pm} & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & \psi_{m,N-2,x}^{k,\pm} & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & \psi_{m,N-1,x}^{k,\pm} & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
\end{pmatrix}
\] (4.4.74)

and

\[
G_m^k = \begin{pmatrix}
1 & -\mathbb{1}_{\mathcal{A}}(\partial \Omega) & 0 & \cdots & 0 & 0 & 0 \\
\psi_{m,1,y}^{-k,-} & \chi_{m,1} & \psi_{m,1,y}^{k,+} & \cdots & 0 & 0 & 0 \\
0 & \psi_{m,2,y}^{-k,-} & \chi_{m,2} & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \chi_{m,N-2} & \psi_{m,N-2,y}^{k,+} & 0 \\
0 & 0 & 0 & \cdots & \psi_{m,N-1,y}^{-k,-} & \chi_{m,N-1} & \psi_{m,N-1,y}^{k,+} \\
0 & 0 & 0 & \cdots & 0 & -\mathbb{1}_{\mathcal{A}}(\partial \Omega) & 1 \\
\end{pmatrix}
\] (4.4.75)

Once again, we define the square matrix \(B_{\ell_1}^k\) as the block matrix of size \((M + 1)(N + 1) \times (M + 1)(N + 1)\) given by the following (and following the ideas of \(A_k^{\ell_1}\)):
Now, for every \( k \in \{0,1,\ldots,K-1 \} \), we define the block matrix \( \mathcal{M}_{\delta_1}^k \) of size \( 2(M+1)(N+1) \times 2(M+1)(N+1) \) where zero entries are matrices of size \( (M+1)(N+1) \times (M+1)(N+1) \):

\[
\mathcal{M}_{\delta_1}^k = \begin{pmatrix}
\mathcal{I}_N & -1_{\mathcal{V}}(\partial \Omega) I \\
\mathcal{F}_1^{-}\mathcal{G}_1^{k} & \mathcal{F}_1^{k, \dagger} \\
0 & \mathcal{F}_2^{-}\mathcal{G}_2^{k} & \mathcal{F}_2^{k, \dagger} \\
0 & 0 & \mathcal{F}_3^{-}\mathcal{G}_3^{k} & \mathcal{F}_3^{k, \dagger} \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \ldots & \mathcal{G}_{M-2}^{k} & \mathcal{F}_{M-2}^{k, \dagger} & 0 \\
0 & 0 & 0 & 0 & \ldots & \mathcal{F}_{M-1}^{-}\mathcal{G}_{M-1}^{k} & \mathcal{F}_{M-1}^{k, \dagger} & 0 \\
0 & 0 & 0 & 0 & \ldots & 0 & -1_{\mathcal{V}}(\partial \Omega) I & \mathcal{I}_N
\end{pmatrix}
\]

Now, for every \( k \in \{0,1,\ldots,K-1 \} \), we define the block matrix \( \mathcal{M}_{\delta_1}^k \) of size \( 2(M+1)(N+1) \times 2(M+1)(N+1) \) where zero entries are matrices of size \( (M+1)(N+1) \times (M+1)(N+1) \):

\[
\mathcal{M}_{\delta_1}^k = \begin{pmatrix}
\mathcal{A}_{\delta_1}^{k} & 0 \\
0 & \mathcal{B}_{\delta_1}^{k}
\end{pmatrix}
\]

(4.4.77)

With this nomenclature for \( \delta_1 \) we can write our continuous eq. (3.1.1) as a finite difference approximation in eq. (4.3.19) using a matrix equation with \( \mathcal{M}_{\delta_1}^k \) and \( \mathbf{v}_{\delta_1}^k \), keeping in mind our first step would use \( \mathbf{v}_{\delta_1}^0 \) on the right hand side (and making sure this initial condition vector’s elements are all indeed positive and bounded by one):

\[
\mathcal{M}_{\delta_1}^k \mathbf{v}_{\delta_1}^{k+1} = \mathbf{v}_{\delta_1}^k
\]

(4.4.78)

To reiterate what was said on the matrix equation, we assume that we have a linear mapping \((\mathcal{M}_{\delta_1}^k)^{-1} : \mathbf{v}_{\delta_1}^k \rightarrow \mathbf{v}_{\delta_1}^{k+1}\). As the inverse M-matrix only depends on the values of \( \mathbf{v}_{\delta_1}^k \) and we prove that we do indeed have an M-matrix throughout the dissertation, we are ensured that the solution exists whence we have a valid \( \mathbf{v}_{\delta_1}^k \).

We perform this same construction for our next two experiments, acknowledging the notational convenience of not supplying the dependence of the M-matrix (and therefore
its corresponding inverse on the previous state vector supplied. For \( \sigma_2 \), let \( Y^k_{\sigma_2} \) be any of \( \{ S^k_{\sigma_2}, X^k_{\sigma_2}, I^k_{\sigma_2}, E^k_{\sigma_2} \} \) the ordered vector approximations of the state variables at time \( t_k \), defined as:

\[
S^k_{\sigma_2} = (S^k_0, S^k_1, \ldots, S^k_M) \quad (4.4.79)
\]

\[
X^k_{\sigma_2} = (X^k_0, X^k_1, \ldots, X^k_M) \quad (4.4.80)
\]

\[
I^k_{\sigma_2} = (I^k_0, I^k_1, \ldots, I^k_M) \quad (4.4.81)
\]

\[
E^k_{\sigma_2} = (E^k_0, E^k_1, \ldots, E^k_M) \quad (4.4.82)
\]

Let \( v^k_{\sigma_2} = (S^k_{\sigma_2} \ | \ X^k_{\sigma_2} \ | \ I^k_{\sigma_2} \ | \ E^k_{\sigma_2})^t \), and further let \( S^0_{\sigma_2}, X^0_{\sigma_2}, I^0_{\sigma_2} \) and \( E^0_{\sigma_2} \) be defined as:

\[
S^0_{\sigma_2} = (0, S^0_1, \ldots, S^0_{M-1}, 0) \quad (4.4.83)
\]

\[
X^0_{\sigma_2} = (0, X^0_1, \ldots, X^0_{M-1}, 0) \quad (4.4.84)
\]

\[
I^0_{\sigma_2} = (0, I^0_1, \ldots, I^0_{M-1}, 0) \quad (4.4.85)
\]

\[
E^0_{\sigma_2} = (0, E^0_1, \ldots, E^0_{M-1}, 0) \quad (4.4.86)
\]

Finally we let \( v^0_{\sigma_2} = (S^0_{\sigma_2} \ | \ X^0_{\sigma_2} \ | \ I^0_{\sigma_2} \ | \ E^0_{\sigma_2})^t \). By construction it should be noted that both \( v^k_{\sigma_2} \) and \( v^0_{\sigma_2} \) are length \( 4(M + 1) \) vectors.
The case of assembling the matrices for $\mathcal{E}_2$ is slightly easier. Defining a collection of matrices of size $(M + 1) \times (M + 1)$ for each $k \in \{0, 1, \ldots, K - 1\}$, and letting $Y$ represent any of the functions $S, X, I, E$, we define the tridiagonal matrix

$$
\mathcal{B}^{k,Y}_{\mathcal{E}_2} = \begin{bmatrix}
1 & -\mathbb{1}_M(\partial \Omega) & 0 & 0 & \cdots & 0 & 0 & 0 \\
\alpha_1^{-k} & \beta_1^{k,Y} & \alpha_1^{k,+} & 0 & \cdots & 0 & 0 & 0 \\
0 & \alpha_2^{-k} & \beta_2^{k,Y} & \alpha_2^{k,+} & \cdots & 0 & 0 & 0 \\
0 & 0 & \alpha_3^{-k} & \beta_3^{k,Y} & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & \alpha_{M-1}^{-k} & \beta_{M-1}^{k,Y} & \alpha_{M-1}^{k,+} \\
0 & 0 & 0 & 0 & \cdots & 0 & -\mathbb{1}_M(\partial \Omega) & 1
\end{bmatrix}
$$

(4.4.87)

We define also the diagonal matrices with the same size as $\mathcal{B}^{k,Y}_{\mathcal{E}_2}$ for $j \in \{1, 2\}$,

$$
\mathcal{H}^{k}_{\mathcal{E}_2} = \begin{bmatrix}
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \eta_1^{k} & 0 & \cdots & 0 & 0 \\
0 & 0 & \eta_2^{k} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \eta_{M-1}^{k} & 0 \\
0 & 0 & 0 & \cdots & 0 & 0
\end{bmatrix}
$$

(4.4.88)

and

$$
\mathcal{Z}^{k,j}_{\mathcal{E}_2} = \begin{bmatrix}
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \zeta_1^{k,j} & 0 & \cdots & 0 & 0 \\
0 & 0 & \zeta_2^{k,j} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \zeta_{M-1}^{k,j} & 0 \\
0 & 0 & 0 & \cdots & 0 & 0
\end{bmatrix}, \quad \Gamma^{k,j}_{\mathcal{E}_2} = \begin{bmatrix}
0 & 0 & 0 & \cdots & 0 & 0 \\
0 & \gamma_1^{k,j} & 0 & \cdots & 0 & 0 \\
0 & 0 & \gamma_2^{k,j} & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & \gamma_{M-1}^{k,j} & 0 \\
0 & 0 & 0 & \cdots & 0 & 0
\end{bmatrix}
$$

(4.4.89)
Now, for every \( k \in \{0, 1, \ldots, K - 1\} \), we define the block matrix \( M^k_{\mathcal{E}_2} \) of size \( 4(M + 1) \times 4(M + 1) \) where zero entries are matrices of size \((M + 1) \times (M + 1):\)

\[
M^k_{\mathcal{E}_2} = \begin{pmatrix}
B^k_{\mathcal{E}_2} & Z^k_{\mathcal{E}_2} \ 0 & H^k_{\mathcal{E}_2} \\
\Gamma^k_{\mathcal{E}_2} & B^k_{\mathcal{E}_2} \ 0 & 0 \\
0 & Z^k_{\mathcal{E}_2} & B^k_{\mathcal{E}_2} & 0 \\
0 & 0 & 0 & B^k_{\mathcal{E}_2}
\end{pmatrix}
\] (4.4.90)

Once again we can represent \( \mathcal{E}_2 \) from the continuous to discrete eqs. (3.1.5) and (4.3.20) as a matrix evolution equation identical in form to eq. (4.4.78) of \( \mathcal{E}_1 \) but with matrix and vectors defined to reflect eqs. (3.1.5) and (4.3.20):

\[
M^k_{\mathcal{E}_2} v^{k+1}_{\mathcal{E}_2} = v^k_{\mathcal{E}_2}
\] (4.4.91)

\( \mathcal{E}_3 \) uses the same ideas from both of these methods in the sense that it derives the construction from \( \mathcal{E}_1 \) but the number of state variables from \( \mathcal{E}_2 \). Let \( k \) be an element of \( \{0, 1, \ldots, K\} \), and let \( S^k_{\mathcal{E}_3}, X^k_{\mathcal{E}_3}, I^k_{\mathcal{E}_3}, \) and \( E^k_{\mathcal{E}_3} \) be the ordered vector approximations of the state variables at time \( t_k \), defined as:

\[
S^k_{\mathcal{E}_3} = (S^k_{0,0}, S^k_{0,1}, \ldots, S^k_{0,N}, S^k_{1,0}, S^k_{1,1}, \ldots, S^k_{1,N}, \ldots, S^k_{M,0}, S^k_{M,1}, \ldots, S^k_{M,N})
\] (4.4.92)

\[
X^k_{\mathcal{E}_3} = (X^k_{0,0}, X^k_{0,1}, \ldots, X^k_{0,N}, X^k_{1,0}, X^k_{1,1}, \ldots, X^k_{1,N}, \ldots, X^k_{M,0}, X^k_{M,1}, \ldots, X^k_{M,N})
\] (4.4.93)

\[
I^k_{\mathcal{E}_3} = (I^k_{0,0}, I^k_{0,1}, \ldots, I^k_{0,N}, I^k_{1,0}, I^k_{1,1}, \ldots, I^k_{1,N}, \ldots, I^k_{M,0}, I^k_{M,1}, \ldots, I^k_{M,N})
\] (4.4.94)

\[
E^k_{\mathcal{E}_3} = (E^k_{0,0}, E^k_{0,1}, \ldots, E^k_{0,N}, E^k_{1,0}, E^k_{1,1}, \ldots, E^k_{1,N}, \ldots, E^k_{M,0}, E^k_{M,1}, \ldots, E^k_{M,N})
\] (4.4.95)

Let \( v^k_{\mathcal{E}_3} = (S^k_{\mathcal{E}_3}, X^k_{\mathcal{E}_3}, I^k_{\mathcal{E}_3}, E^k_{\mathcal{E}_3})^t \), and further let \( S^0_{\mathcal{E}_3}, X^0_{\mathcal{E}_3}, I^0_{\mathcal{E}_3}, \) and \( E^0_{\mathcal{E}_3} \) be defined as:
\[ S_{\delta_3}^0 = (0, 0, \ldots, 0, 0, S_{1,1}^0, \ldots, S_{1,N-1}^0, 0, \ldots, \underbrace{0, \ldots, 0}_{N+1 \text{ entries}}) \] (4.4.96)

\[ X_{\delta_3}^0 = (0, 0, \ldots, 0, 0, X_{1,1}^0, \ldots, X_{1,N-1}^0, 0, \ldots, \underbrace{0, \ldots, 0}_{N+1 \text{ entries}}) \] (4.4.97)

\[ I_{\delta_3}^0 = (0, 0, \ldots, 0, 0, I_{1,1}^0, \ldots, I_{1,N-1}^0, 0, \ldots, \underbrace{0, \ldots, 0}_{N+1 \text{ entries}}) \] (4.4.98)

\[ E_{\delta_3}^0 = (0, 0, \ldots, 0, 0, E_{1,1}^0, \ldots, E_{1,N-1}^0, 0, \ldots, \underbrace{0, \ldots, 0}_{N+1 \text{ entries}}) \] (4.4.99)

Finally we let \( v_{\delta_3}^0 = (S_{\delta_3}^0 | X_{\delta_3}^0 | I_{\delta_3}^0 | E_{\delta_3}^0)^t \). By construction it should be noted that both \( v_{\delta_3}^k \) and \( v_{\delta_3}^0 \) are length \( 4(M+1)(N+1) \) vectors.
To create the M-matrix corresponding to eq. (4.3.21) for $\mathcal{E}_3$ we start by observing that a grouping of $(N + 1) \times (N + 1)$ submatrices will correspond to the discretized $y$ index $n$. For every $m \in \{1, \ldots, M - 1\}$, every $k \in \{0, 1, \ldots, K\}$, $z = \{x, y\}$, and $Y$ representing any of the functions $S, X, I, E$, we define $F_m^{k,\pm}$ and $G_m^{k,Y}$ as:

$$
F_m^{k,\pm} = \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
\end{pmatrix}, \quad (4.4.100)
$$

and

$$
G_m^{k,Y} = \begin{pmatrix}
1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
\end{pmatrix}, \quad (4.4.101)
$$

Next we construct the block matrices $B_m^{k,Y}$ of size $(M + 1)(N + 1) \times (M + 1)(N + 1)$, where the identity matrix $I$, the modified $I_N$, and the zero entries below are all $(N + 1) \times (N + 1)$ submatrices as before.
\[
B^{k,Y}_{\delta_3} = \begin{pmatrix}
I_N & -I_N(\partial\Omega)I & 0 & 0 & \cdots & 0 & 0 & 0 \\
F^{k,-}_1 & \mathcal{G}^{k,Y}_1 & F^{k,+}_1 & 0 & \cdots & 0 & 0 & 0 \\
0 & F^{k,-}_2 & \mathcal{G}^{k,Y}_2 & F^{k,+}_2 & \cdots & 0 & 0 & 0 \\
0 & 0 & F^{k,-}_3 & \mathcal{G}^{k,Y}_3 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & \mathcal{G}^{k,Y}_{M-2} & F^{k,+}_{M-2} & 0 \\
0 & 0 & 0 & 0 & \cdots & F^{k,-}_{M-1} & \mathcal{G}^{k,Y}_{M-1} & F^{k,+}_{M-1} \\
0 & 0 & 0 & 0 & \cdots & 0 & -I_N(\partial\Omega)I & I_N \\
\end{pmatrix}
\] (4.4.102)

In this same fashion, for every \( m \in \{1, \ldots, M - 1\} \), and every \( k \in \{0, 1, \ldots, K\} \), we construct the \((N + 1) \times (N + 1)\) diagonal matrices \( P_{m}^{k,j} \), \( Q_{m}^{k,j} \), and \( T_{m}^{k} \) for \( j \in \{1, 2\} \):

\[
P_{m}^{k,j} = \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & \zeta_{m,1} & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & \zeta_{m,2} & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & \zeta_{m,3} & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & \zeta_{m,N-2} & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & \zeta_{m,N-1} & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
\end{pmatrix},
\] (4.4.103)

\[
Q_{m}^{k,j} = \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
\gamma_{m,1} & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & \gamma_{m,2} & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & \gamma_{m,3} & 0 & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & \gamma_{m,N-2} & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & \gamma_{m,N-1} & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
\end{pmatrix},
\] (4.4.104)

and

36
$\mathcal{T}_m^k = \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & \eta_{m,1}^k & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & \eta_{m,2}^k & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & \eta_{m,3}^k & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & \eta_{m,N-2}^k & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & \eta_{m,N-1}^k & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 
\end{pmatrix}$. \hfill (4.4.105)

Similarly we create the $(M+1)(N+1) \times (M+1)(N+1)$ block matrices $Z_{\delta_3}^{k,j}$, $\Gamma_{\delta_3}^{k,j}$, and $\mathcal{H}_{\delta_3}^k$ where the zero entries below are all $(N+1) \times (N+1)$ submatrices:

$Z_{\delta_3}^{k,j} = \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & P_{1,j}^k & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & P_{2,j}^k & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & P_{3,j}^k & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & P_{M-2,j}^k & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & P_{M-1,j}^k & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 
\end{pmatrix}$, \hfill (4.4.106)

$\Gamma_{\delta_3}^{k,j} = \begin{pmatrix}
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & Q_{1,j}^k & 0 & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & Q_{2,j}^k & 0 & \cdots & 0 & 0 & 0 \\
0 & 0 & 0 & Q_{3,j}^k & \cdots & 0 & 0 & 0 \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & 0 & 0 & \cdots & Q_{M-2,j}^k & 0 & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & Q_{M-1,j}^k & 0 \\
0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 
\end{pmatrix}$, \hfill (4.4.107)

and
Finally we construct the entire $4(M + 1)(N + 1) \times 4(M + 1)(N + 1)$ block matrix $\mathcal{M}^k_{\mathcal{E}_3}$ where zero entries are matrices of size $(M + 1)(N + 1) \times (M + 1)(N + 1)$:

$$\mathcal{M}^k_{\mathcal{E}_3} = \begin{pmatrix}
B_{\mathcal{E}_3}^{k,S} & \mathcal{Z}_{\mathcal{E}_3}^{k,1} & 0 & \mathcal{H}_{\mathcal{E}_3}^k \\
\Gamma_{\mathcal{E}_3}^{k,1} & B_{\mathcal{E}_3}^{k,X} & 0 & 0 \\
0 & \mathcal{Z}_{\mathcal{E}_3}^{k,2} & B_{\mathcal{E}_3}^{k,Y} & 0 \\
\Gamma_{\mathcal{E}_3}^{k,2} & 0 & 0 & B_{\mathcal{E}_3}^{k,Z}
\end{pmatrix}$$

(4.4.109)

We can now write our finite difference method as before in eqs. (4.4.78) and (4.4.91):

$$\mathcal{M}^k_{\mathcal{E}_3} \mathbf{v}^k_{\mathcal{E}_3} = \mathbf{v}^k_{\mathcal{E}_3}$$

(4.4.110)

## 5 Criteria to Ensure M-matrices

Up until this point, despite a few remarks, we have not spent any time establishing the requirements for there to be viable solutions to the finite difference systems previously mentioned for experiments $\mathcal{E}_1$ through $\mathcal{E}_3$. This section will be devoted to the task of ensuring that our solutions will be stable, bounded, and positive. First, we make use of the following Lemma from [31].
Lemma 5.1. [Fujimoto and Ranade (2004)] Every $M$-matrix is invertible, and the entries of its inverse are all positive real numbers. The necessary requirements for an $M$-matrix are:

(A) All entries on the diagonal must be positive.

(B) All off-diagonal entries must be nonpositive.

(C) The matrix must be strictly diagonally dominant.

In light of Lemma 5.1, we need to show that each matrix $M^k_{E_1}$, $M^k_{E_2}$, $M^k_{E_3}$ described in eqs. (4.4.77), (4.4.90) and (4.4.109) satisfies the requirements (A), (B), and (C) above. The conditions (A) and (B) are easy to show and will be described below in a series of remarks combined with ones made previously. Condition (C) however, the requirement of strict diagonal dominance, cannot be shown for any matrix with Neumann conditions. However, there is a weaker requirement for a square matrix to be an $M$-matrix which we will prove in the following section. The requirement is due to properties shared with the so called weakly chained diagonally dominant (WCDD) matrices for which strictly diagonally dominant matrices are a subclass [32, 33]. Essentially (and to be described completely further down), an L-matrix which possesses the property of WCDD is an $M$-matrix. The requirement for an L-matrix is that (A) and (B) are satisfied.

We begin with establishing (A) and (B) for all experiments. We see that for the first and last $(N + 1)$ rows corresponding to the block matrices $A^k_{\delta_1}$, $B^k_{\delta_1}$, and $B^k_{\delta_3}$, we have one or two on the diagonal which is obviously positive. Every other off diagonal entry on the top and bottom $(N + 1)$ rows for each of $A^k_{\delta_1}$, $B^k_{\delta_1}$, and $B^k_{\delta_3}$ are -1 (if there are Neumann conditions) or 0, corresponding to Dirichlet conditions.

Remark 5.2. Recall that the common convention for labeling indexes of matrices uses the natural numbers. By inspection, it is easy to see that for all boundary conditions we have the following cases:
1. For every $i = j$ or $i = M(N + 1) + j$ when $j \in \{2, \ldots, N\}$, $A_{i,j}^k = 1$ and $B_{i,j}^k = 1$ in $E_1$, and $B_{i,j}^{k,Y} = 1$ in $E_3$, using either Dirichlet or Neumann conditions. If $i = j$ for exactly $j \in \{1, 2(N + 1), M(N + 1) + 1, (M + 1)(N + 1)\}$, $A_{i,j}^k = 2$ and $B_{i,j}^k = 2$ in $E_1$, and $B_{i,j}^{k,Y} = 2$ in $E_3$ for Neumann conditions, whereas $A_{i,j}^k = 1$ and $B_{i,j}^k = 1$ in $E_1$, and $B_{i,j}^{k,Y} = 1$ in $E_3$ for Dirichlet. Every $B_{i,j}^{k,Y} = 1$ and $B_{M+1,M+1}^{k,Y} = 1$ in $E_2$ for Neumann or Dirichlet. In all cases listed above these values are on the diagonal and positive.

2. If for any $q \in \{1, 2, \ldots, N + 1\}$ and letting the pair of row index $i = q$, column index $j = N + 1 + q$, or the pair of row index $i = (M + 1)(N + 1) + 1 - q$, column index $j = M(N + 1) + 1 - q$, then $A_{i,j}^k = -1$ and $B_{i,j}^k = -1$ in $E_1$, and $B_{i,j}^{k,Y} = -1$ in $E_3$ for Neumann conditions, and $A_{i,j}^k = 0$ and $B_{i,j}^k = 0$ in $E_1$, and $B_{i,j}^{k,Y} = 0$ in $E_3$ for Dirichlet. In all cases these values are off diagonal and nonpositive. Every $B_{1,1}^{k,Y} = 1$ and $B_{M+1,M+1}^{k,Y} = 1$ in $E_2$ for Neumann or Dirichlet. In all cases listed above these values are off diagonal and nonpositive.

3. If for any $q \in \{1, 2, \ldots, M + 1\}$ and letting the pair of row index $i = (q - 1)(N + 1) + 1$, column index $j = (q - 1)(N + 1) + 2)$, or the pair of row index $i = q(N + 1)$, column index $j = (q - 1)(N + 1) + N$, then $A_{i,j}^k = -1$ and $B_{i,j}^k = -1$ in $E_1$, and $B_{i,j}^{k,Y} = -1$ in $E_3$ for Neumann conditions, and $A_{i,j}^k = 0$ and $B_{i,j}^k = 0$ in $E_1$, and $B_{i,j}^{k,Y} = 0$ in $E_3$ for Dirichlet. In all cases these values are off diagonal and nonpositive. Every $B_{1,2}^{k,Y} = -1$ and $B_{M+1,M}^{k,Y} = -1$ in $E_2$ for Neumann conditions, or $B_{1,2}^{k,Y} = 0$ and $B_{M+1,M}^{k,Y} = 0$ in $E_2$ for Dirichlet. In all cases listed above these values are off diagonal and nonpositive.

Keeping in mind the observations of 5.2, we are assured that all boundary conditions of Neumann or Dirichlet type for $E_1$, $E_2$, and $E_3$ obey the criteria for $M_{E_1}^k$, $M_{E_2}^k$, and $M_{E_3}^k$ to be L-matrices. Given our conditions from 3.1, 3.2, and 3.3 (and ensuring the condition that $K_3 \Delta t_k < 1 + K_2 \Delta t_k$ in $E_1$), we can say that all diagonal entries are indeed positive and all off diagonal entries are negative or zero. Therefore we can conclude that $M_{E_1}^k$, $M_{E_2}^k$, and $M_{E_3}^k$ are indeed L-matrices.
To show that $M^k_{E_1}, M^k_{E_2}, M^k_{E_3}$ are M-matrices by Lemma 5.1 and [32], we would be required to honor condition (C), that is, strict diagonal dominance for every row in each $M^k$. We say row $i$ of a complex matrix $W = (W_{i,j})$ is strictly diagonally dominant (SDD) if $|W_{i,i}| > \sum_{j \neq i} |W_{i,j}|$. We say $W$ is SDD if all of its rows are SDD. Weakly diagonally dominant (WDD) is defined with $\geq$ instead.

**Remark 5.3.** Consider the 2 separate $(M + 1)(N + 1) \times (M + 1)(N + 1)$ block rows ($A^k_{E_1}$ and $B^k_{E_1}$) of $M^k_{E_1}$ corresponding to the conditions on the 2 state variables in $E_1$, or the 4 separate $(M + 1)(N + 1) \times (M + 1)(N + 1)$ block rows $B^{k,Y}_{E_3}$ of $M^k_{E_3}$ corresponding to the conditions on the 4 state variables in $E_3$ under Neumann conditions. Note that the rows corresponding to the 4 outer corners of each of the block rows $A^k_{E_1}, B^k_{E_1},$ and $B^{k,Y}_{E_3}$ reveal the relationship corresponding to WDD rows, namely for any of $Y^k \in \{A^k_{E_1}, B^k_{E_1}, B^{k,Y}_{E_3}\}$, letting $i = j$ for exactly $j = \{1, 2(N + 1), M(N + 1) + 1, (M + 1)(N + 1)\}$, we have the nonzero row sum relationships:

\[ |Y^k_{1,1}| = |Y^k_{1,2}| + |Y^k_{1,N+2}|, \tag{4.5.111} \]
\[ |Y^k_{2(N+1),2(N+1)}| = |Y^k_{2(N+1),2N+1}| + |Y^k_{2(N+1),(N+1)}|, \tag{4.5.112} \]
\[ |Y^k_{M(N+1)+1,M(N+1)+1}| = |Y^k_{M(N+1)+1,M(N+1)}| + |Y^k_{M(N+1)+1,(M-1)(N+1)}|, \tag{4.5.113} \]
\[ |Y^k_{(M+1)(N+1),(M+1)(N+1)}| = |Y^k_{(M+1)(N+1),(M+1)(N+1)-1}| + |Y^k_{(M+1)(N+1),M(N+1)}|. \tag{4.5.114} \]

Therefore the sum of the absolute values of the off-diagonal entries are equal to the diagonal and weak diagonal dominance holds for these rows.

**Remark 5.4.** Consider the 2 separate $(M + 1)(N + 1) \times (M + 1)(N + 1)$ block rows ($A^k_{E_1}$ and $B^k_{E_1}$) of $M^k_{E_1}$ corresponding to the conditions on the 2 state variables in $E_1$, or the 4 separate $(M + 1)(N + 1) \times (M + 1)(N + 1)$ block rows $B^{k,Y}_{E_3}$ of $M^k_{E_3}$ corresponding to the conditions on the 4 state variables in $E_3$ under Neumann conditions. Note that the rows corresponding
to the edges without outer corners of each of the block rows $A_k^{E_1}$, $B_k^{E_1}$, and $B_k^{E_2,Y}$ reveal the relationship corresponding to WDD rows, namely for any of $Y^k \in \{A_k^{E_1}, B_k^{E_1}, B_k^{E_2,Y}\}$, where $q \in \{2, \ldots, M\}$ and $j \in \{2, \ldots, N\}$

$$|Y^k_{(q-1)(N+1)+1,(q-1)(N+1)+1}| = |Y^k_{(q-1)(N+1)+1,(q-1)(N+1)+2}|,$$  
(4.5.115)

$$|Y^k_{q(N+1),q(N+1)}| = |Y^k_{q(N+1),q(N+1)-1}|,$$  
(4.5.116)

$$|Y^k_{j,j}| = |Y^k_{j,j+(N+1)}|,$$  
(4.5.117)

$$|Y^k_{j+M(N+1),j+M(N+1)}| = |Y^k_{j+M(N+1),j+(M-1)(N+1)}|.$$  
(4.5.118)

Therefore the sum of the absolute values of the off-diagonal entries are equal to the diagonal and weak diagonal dominance holds for these rows. For completeness, each of the $4 (M + 1) \times (M + 1)$ block matrices corresponding to $B_k^{E_2,Y}$ show weak diagonal dominance under Neumann conditions according to the two relationships:

$$|Y^k_{1,1}| = |Y^k_{1,2}|,$$  
(4.5.119)

$$|Y^k_{(M+1),(M+1)}| = |Y^k_{(M+1),M}|$$  
(4.5.120)

for each $Y^k \in B_k^{E_2,Y}$.

Remark 5.5. Consider now all of the separate block rows of any of $Y^k \in \{A_k^{E_1}, B_k^{E_1}, B_k^{E_2,Y}, B_k^{E_3,Y}\}$ under Dirichlet conditions. Note that the first and last rows of each of the block rows are equal to 1, whereas the off-diagonal entries are equal to 0. Therefore the sum of the absolute values of the off-diagonal entries are less than the diagonal and strict diagonal dominance holds for all boundary conditions.

Remark 5.6. Consider the remaining block rows of $Y^k \in \{A_k^{E_1}, B_k^{E_1}, B_k^{E_2,Y}, B_k^{E_3,Y}\}$ describing the interior conditions on the state variables not corresponding to Remarks 5.3–5.5.
We intend to ensure that all of these equations will display strict diagonal dominance for each experiment in turn, arriving at relationships on the parameters.

1. **BLOCK ROW 1 of $\mathbf{A}^{k}_{\varepsilon_{1}}$**: Using Remarks from 3.1, all off-diagonal entries in each row are nonpositive being equal to $-R_{x}^{k,s}$, $-R_{y}^{k,s}$, or zero, for every $m \in \{1, 2, \ldots, M - 1\}$, $n \in \{1, 2, \ldots, N - 1\}$ and $k \in \{0, 1, \ldots, K - 1\}$. The corresponding diagonal entry $\phi_{m,n}^{k}$ is positive. The inequality $2|R_{x}^{k,s}| + 2|R_{y}^{k,s}| < |\phi_{m,n}^{k}|$ holds always.

2. **BLOCK ROW 2 of $\mathbf{B}^{k}_{\varepsilon_{1}}$**: Per 3.1, all off-diagonal entries in each row are nonpositive being equal to $\psi_{m,n,x}^{k}$, $\psi_{m,n,y}^{k}$, or $\psi_{m,n,x}^{k}$, or $\psi_{m,n,y}^{k}$, or $\psi_{m,n,x}^{k}$, or zero, for every $m \in \{1, 2, \ldots, M - 1\}$, and $n \in \{1, 2, \ldots, N - 1\}$. The corresponding diagonal entry $\chi_{m,n}^{k}$ is positive. The inequality $|\psi_{m,n,x}^{k}| + |\psi_{m,n,y}^{k}| + |\psi_{m,n,x}^{k}| + |\psi_{m,n,y}^{k}| < |\chi_{m,n}^{k}|$ holds if and only if

$$1 + K_{2}\Delta t_{k} - K_{3}\Delta t_{k} \frac{s_{m,n}^{k}}{K_{4} + s_{m,n}^{k}} < \chi_{m,n}^{k} \qquad (4.5.121)$$

which is satisfied if $K_{3}\Delta t_{k} < 1 + K_{2}\Delta t_{k}$.

3. **BLOCK ROW 1 of $\mathbf{B}^{k,S}_{\varepsilon_{2}}$**: All off-diagonal entries in each row are nonpositive being equal to $\alpha_{m}^{k,-}$, $\alpha_{m}^{k,+}$, $\zeta_{m}^{k,1}$, $\eta_{m}^{k}$, or zero, for every $m \in \{1, 2, \ldots, M - 1\}$. The corresponding diagonal entry $\beta_{m}^{k,S}$ is positive. The inequality $|\alpha_{m}^{k,-}| + |\alpha_{m}^{k,+}| + |\zeta_{m}^{k,1}| + |\eta_{m}^{k}| < |\beta_{m}^{k,S}|$ holds if and only if

$$\frac{(a_{2} + a_{3})\Delta t_{k}}{\kappa_{S} + S_{m}^{k}} < 1 + \frac{a_{1}X_{m}^{k}\Delta t_{k}}{\kappa_{S} + S_{m}^{k}} \qquad (4.5.122)$$

which is satisfied if $(a_{2} + a_{3})\Delta t_{k} < \kappa_{S}$.

4. **BLOCK ROW 2 of $\mathbf{B}^{k,X}_{\varepsilon_{2}}$**: All off-diagonal entries in each row are nonpositive being equal to $\alpha_{m}^{k,-}$, $\alpha_{m}^{k,+}$, $\gamma_{m}^{k,1}$, or zero, for every $m \in \{1, 2, \ldots, M - 1\}$. The corresponding
diagonal entry $\beta^{k,X}_m$ is positive. The inequality $|\alpha^{k,-}_m| + |\alpha^{k,+}_m| + |\gamma^{k,1}_m| < |\beta^{k,X}_m|$ holds if and only if

$$\frac{\mu X^k_m T^k_k}{\kappa_S} < 1 + \frac{a_4 \Delta t_k}{\kappa_S} + \frac{a_4 \Delta t_k}{\kappa_S} \kappa_S + S^k_m$$

(4.5.123)

which is satisfied if $\mu \Delta t_k < \kappa_S$.

5. BLOCK ROW 3 of $B^{k,I}_{\sigma_2}$: All off-diagonal entries in each row are nonpositive being equal to $\alpha^{k,-}_m$, $\alpha^{k,+}_m$, $\zeta^{k,2}_m$, or zero, for every $m \in \{1,2,\ldots,M-1\}$. The corresponding diagonal entry $\beta^{k,I}_m$ is positive. The inequality $|\alpha^{k,-}_m| + |\alpha^{k,+}_m| + |\zeta^{k,2}_m| < |\beta^{k,I}_m|$ holds if and only if

$$\frac{a_4 \Delta t_k}{\kappa_S} < 1$$

(4.5.124)

which is satisfied if $a_4 \Delta t_k < \kappa_S$.

6. BLOCK ROW 4 of $B^{k,E}_{\sigma_2}$: All off-diagonal entries in each row are nonpositive being equal to $\alpha^{k,-}_m$, $\alpha^{k,+}_m$, $\zeta^{k,2}_m$, or zero, for every $m \in \{1,2,\ldots,M-1\}$. The corresponding diagonal entry $\beta^{k,E}_m$ is positive. The inequality $|\alpha^{k,-}_m| + |\alpha^{k,+}_m| + |\zeta^{k,2}_m| < |\beta^{k,E}_m|$ holds if and only if

$$\frac{a_5 X^k_m \Delta t_k}{\kappa_S} < 1 + \frac{a_3 \Delta t_k}{\kappa_S} + \frac{a_3 \Delta t_k}{\kappa_S} \kappa_S + S^k_m$$

(4.5.125)

which is satisfied if $a_5 \Delta t_k < \kappa_S$.

7. BLOCK ROW 1 of $B^{k,S}_{\sigma_3}$: All off-diagonal entries in each row are nonpositive being equal to $\alpha^{k,-}_{m,n,x}$, $\alpha^{k,-}_{m,n,y}$, $\alpha^{k,+}_{m,n,y}$, $\alpha^{k,+}_{m,n,x}$, $\zeta^{k,1}_{m,n}$, $\eta^{k,1}_{m,n}$, or zero, for every $m \in \{1,2,\ldots,M-1\}$, and $n \in \{1,2,\ldots,N-1\}$. The corresponding diagonal entry $\beta^{k,S}_{m,n}$ is positive. The
inequality \(|\alpha_{m,n,x}^k| + |\alpha_{m,n,y}^k| + |\alpha_{m,n,x}^{k,+}| + |\alpha_{m,n,y}^{k,+}| + |\gamma_{m,n}^k| + |\eta_{m,n}^k| < |\beta_{m,n}^{k,S}|\) holds if and only if

\[
\frac{(a_2 + a_3)\Delta t_k}{\kappa_S + S_{m,n}^k} < 1 + \frac{a_1X_{m,n}^k\Delta t_k}{\kappa_S + S_{m,n}^k}
\] (4.5.126)

which is satisfied if \((a_2 + a_3)\Delta t_k < \kappa_S\).

8. **BLOCK ROW 2** of \(B_{E}^{k,X}\): All off-diagonal entries in each row are nonpositive being equal to \(\alpha_{m,n,x}^k, \alpha_{m,n,y}^k, \alpha_{m,n,x}^{k,+}, \alpha_{m,n,y}^{k,+}, \gamma_{m,n}^{k,1}, \) or zero, for every \(m \in \{1, 2, \ldots, M - 1\}\), and \(n \in \{1, 2, \ldots, N - 1\}\). The corresponding diagonal entry \(\beta_{m,n}^{k,X}\) is positive. The inequality \(|\alpha_{m,n,x}^k| + |\alpha_{m,n,y}^k| + |\alpha_{m,n,x}^{k,+}| + |\alpha_{m,n,y}^{k,+}| + |\gamma_{m,n}^{k,1}| < |\beta_{m,n}^{k,X}|\) holds if and only if

\[
\frac{\mu X_{m,n}^k\Delta t_k}{\kappa_S + S_{m,n}^k} < 1 + \frac{a_4 \Delta t_k}{\kappa_S + S_{m,n}^k}
\] (4.5.127)

which is satisfied if \(\mu \Delta t_k < \kappa_S\).

9. **BLOCK ROW 3** of \(B_{E_3}^{k,I}\): All off-diagonal entries in each row are nonpositive being equal to \(\alpha_{m,n,x}^k, \alpha_{m,n,y}^k, \alpha_{m,n,x}^{k,+}, \alpha_{m,n,y}^{k,+}, \gamma_{m,n}^{k,2}, \) or zero, for every \(m \in \{1, 2, \ldots, M - 1\}\), and \(n \in \{1, 2, \ldots, N - 1\}\). The corresponding diagonal entry \(\beta_{m,n}^{k,I}\) is positive. The inequality \(|\alpha_{m,n,x}^k| + |\alpha_{m,n,y}^k| + |\alpha_{m,n,x}^{k,+}| + |\alpha_{m,n,y}^{k,+}| + |\gamma_{m,n}^{k,2}| < |\beta_{m,n}^{k,I}|\) holds if and only if

\[
\frac{a_4 \Delta t_k}{\kappa_S + S_{m,n}^k} < 1
\] (4.5.128)

which is satisfied if \(a_4 \Delta t_k < \kappa_S\).

10. **BLOCK ROW 4** of \(B_{E_3}^{k,E}\): All off-diagonal entries in each row are nonpositive being equal to \(\alpha_{m,n,x}^k, \alpha_{m,n,y}^k, \alpha_{m,n,x}^{k,+}, \alpha_{m,n,y}^{k,+}, \gamma_{m,n}^{k,2}, \) or zero, for every \(m \in \{1, 2, \ldots, M - 1\}\), and \(n \in \{1, 2, \ldots, N - 1\}\). The corresponding diagonal entry \(\beta_{m,n}^{k,E}\) is positive. The inequality \(|\alpha_{m,n,x}^k| + |\alpha_{m,n,y}^k| + |\alpha_{m,n,x}^{k,+}| + |\alpha_{m,n,y}^{k,+}| + |\gamma_{m,n}^{k,2}| < |\beta_{m,n}^{k,E}|\) holds if and only if
\[
\frac{a_5 X_m^k \Delta t_k}{\kappa_S + S_{m,n}^k} < 1 + \frac{a_3 \Delta t_k}{\kappa_S + S_{m,n}^k} \quad (4.5.129)
\]

which is satisfied if \(a_5 \Delta t_k < \kappa_S\).

Having set up all conditions in Remark 5.6 for the guaranteed SDD rows for our experiments, we only need to show that each of our matrices is WCDD. To reiterate, we say row \(i\) of a matrix \(W = (W_{ij})\) is strictly diagonally dominant (SDD) if \(|W_{ii}| > \sum_{j \neq i} |W_{ij}|\). We say \(W\) is SDD if all of its rows are SDD. Weakly diagonally dominant (WDD) is defined with \(\geq\) instead. The directed graph associated with an \(m \times m\) matrix \(W = (W_{ij})\) is given by the vertices \(\{1, \ldots, m\}\) and edges defined as follows: there exists an edge from \(i \to j\) if and only if \(W_{ij} \neq 0\). The square matrix \(W\) is said to be weakly chained diagonally dominant (WCDD) if \(W\) is WDD and for each row \(i\) of \(W\), there exists a path in the directed graph of \(W\) from \(i\) to an SDD row. Note that if \(i\) is itself an SDD row, the trivial path \(i \to i\) satisfies the second requirement in the above definition.

**Remark 5.7.** In light of Remark 5.6, we need only find a path from the WDD rows found in our Neumann boundary conditions for each experiment.

1. Each of the WDD rows of (4.5.115) and (4.5.116) possess the following paths to a corresponding SDD row for each \(q \in \{2, \ldots, M\}\):

\[
(q - 1)(N + 1) + 1 \to (q - 1)(N + 1) + 2 \quad (4.5.130)
\]
\[
q(N + 1) \to q(N + 1) - 1 \quad (4.5.131)
\]
2. Each of the WDD rows \( j \) for (4.5.117) and (4.5.118) possess the following paths to corresponding SDD rows for each \( j \in \{2, \ldots, N\} \):

\[
\begin{align*}
  j & \rightarrow j + (N + 1) \quad (4.5.132) \\
  j + M(N + 1) & \rightarrow j + (M - 1)(N + 1) \quad (4.5.133)
\end{align*}
\]

3. For the corner boundaries, for each one of eqs. (4.5.111)–(4.5.114), we define the paths:

\[
\begin{align*}
  1 & \rightarrow 2 \rightarrow N + 3 \quad (4.5.134) \\
  N + 1 & \rightarrow N \rightarrow 2N + 1 \quad (4.5.135) \\
  M(N + 1) + 1 & \rightarrow M(N + 1) + 2 \rightarrow (M - 1)(N + 1) + 2 \quad (4.5.136) \\
  (M + 1)(N + 1) & \rightarrow M(N + 1) + N \rightarrow (M - 1)(N + 1) + N \quad (4.5.137)
\end{align*}
\]

4. Paths for eqs. (4.5.119) and (4.5.120) are simply:

\[
\begin{align*}
  1 & \rightarrow 2 \quad (4.5.138) \\
  M + 1 & \rightarrow M \quad (4.5.139)
\end{align*}
\]

This concludes all cases and gives examples of paths in the directed graphs to indicate that all matrices are indeed WCDD by design.

**Lemma 5.8.** We suppose that under either Neumann or Dirichlet conditions, \( 0 \leq \psi_k^m < 1 \) for every \( m \in \{1, 2, \ldots, M - 1\} \), and \( n \in \{1, 2, \ldots, N - 1\} \). Then \( M_{\phi_0}^k \) defined in eq. (4.4.78) is an M-matrix if

\[
K_3 \Delta t_k < 1 + K_2 \Delta t_k \quad (4.5.140)
\]
We suppose also that $0 \leq v_{E_2}^k < 1$ for every $m \in \{1, 2, \ldots, M - 1\}$, and $0 \leq v_{E_3}^k < 1$ for every $m \in \{1, 2, \ldots, M - 1\}$, and $n \in \{1, 2, \ldots, N - 1\}$. Then $M_{E_2}^k$ and $M_{E_3}^k$ defined in eqs. (4.4.91) and (4.4.110) are M-matrices if

$$\Delta t_k \max\{a_2 + a_3, \mu, a_4, a_5\} < \kappa_S$$

(4.5.141)

**Proof.** Under assumption eqs. (4.5.140) and (4.5.141), conditions made in Remark 5.7 as an adaptation to (C) of Lemma 5.1 are satisfied. As (A) and (B) of Lemma 5.1 are also satisfied, $M_{E_1}^k$, $M_{E_2}^k$, and $M_{E_3}^k$ of equations eqs. (4.4.78), (4.4.91) and (4.4.110) are M-matrices. □

**Proposition 5.9.** (Non-negativity) Once again assuming that $0 \leq v_{E_j}^k < 1$ for $j \in \{1, 2, 3\}$ and every $m \in \{1, 2, \ldots, M - 1\}$, and $n \in \{1, 2, \ldots, N - 1\}$, and our assumption of eqs. (4.5.140) and (4.5.141), then there exists a unique solution $v_{E_j}^{k+1}$ of each of eqs. (4.4.78), (4.4.91) and (4.4.110), whose components are all nonnegative numbers.

**Proof.** We have already established that through our assumptions 4.4.109 is an M-matrix by Lemma 5.1 and Remark 5.7, and therefore each $M_{E_j}^k$ is nonsingular and all entries are positive real numbers. As $v_{E_j}^k$ contains only nonnegative components, the new approximation $v_{E_j}^{k+1}$ given by $(M_{E_j}^k)^{-1}v_{E_j}^k$ is a vector whose components are all nonnegative. □

To elaborate further on the inverse positivity:

**Remark 5.10.** Let $M^k$ be a real matrix with entries $M_{i,j}^k$ be an M-matrix. We have:

1. $M_{i,j}^k \leq 0$ when $i \neq j$

2. the eigenvalues of $M^k$ have positive real part

To clarify, let $\sigma^k$ be equal to the greatest diagonal entry of $M^k$ (which must be positive by construction. Otherwise, $M^k$ would have a negative eigenvalue). We can rewrite $M^k$ as $M^k = \sigma^k I - A^k$, where $A^k$ is a nonnegative matrix.
By the Perron-Frobenius Theorem [34], $A^k$ must have a positive eigenvalue equal to $\rho(A^k)$. Furthermore, for any eigenvalue $\lambda^k$ of $A^k$, $\sigma^k - \lambda^k$ is an eigenvalue of $M^k$. Thus, $\sigma^k - \rho(A^k)$ is an eigenvalue of $A^k$. Since the eigenvalues of $M^k$ all have positive real part, we note that $\text{Re}(\sigma^k - \rho(A^k)) > 0 \implies \rho(A^k) < \sigma^k$. So, we have $\rho(A^k) < \sigma^k$. Denote $A^k = \frac{1}{\sigma^k} A^k$. We note that $\rho(A^k) = \rho(A^k)/|\sigma^k| < 1$.

Let $M^k = \frac{1}{\sigma^k} M^k = I - A^k$. Since $\rho(A^k) < 1$, we can show that the infinite series $\sum_n (A^k)^n$ converges. More importantly, we note that (defining the zeroth power to be the identity),

$$M^k \sum_{n=0}^{\infty} (A^k)^n = (I - A^k) \sum_{n=0}^{\infty} (A^k)^n = \left( \sum_{n=0}^{\infty} (A^k)^n \right) - \left( \sum_{n=1}^{\infty} (A^k)^n \right) = (A^k)^0 = I$$

Thus, we have:

$$(M^k)^{-1} = (\sigma^k M^k)^{-1} = \frac{1}{\sigma^k} (M^k)^{-1} = \frac{1}{\sigma^k} \sum_{n=0}^{\infty} (A^k)^n.$$

Since $(M^k)^{-1}$ is a sum of the multiple of nonnegative matrices, it must be nonnegative. \qed

Finally, it is common convention in literature to treat as below that inequality used between a vector and a vector or a scalar with a vector denotes the element-wise inequality of each component with the vector or scalar, respectively.

**Proposition 5.11.** (Boundedness of solutions to $\delta_1$) Let $0 \leq v^k_{\delta_1} < e_{\delta_1}$ for some $k \in \{0, 1, \ldots, K-1\}$. If $(1 + K_2 \Delta t_k - K_3 \Delta t_k) e_{\delta_1} - v^k_{\delta_1} > 0_{\delta_1}$ is satisfied, then $0_{\delta_1} \leq v^{k+1}_{\delta_1} < e_{\delta_1}$.

**Proof.** We define the following:

$$x^{k+1}_{\delta_1} = e_{\delta_1} - v^{k+1}_{\delta_1} \quad (4.5.142)$$

where $e_{\delta_1}$ is the vector of the same dimension as $v^{k+1}_{\delta_1}$ whose components are all equal to 1. We can rewrite our implicit problem as:
\[ M^{k}_{\delta_1} x^{k+1}_{\delta_1} = b^k_{\delta_1}, \]  
(4.5.143)

where

\[ b^k_{\delta_1} = M^k_{\delta_1} e_{\delta_1} - v^k_{\delta_1}. \]  
(4.5.144)

We claim that \( b^k_{\delta_1} \geq 0_{\delta_1} \). Let \( i \in \{1, 2, \ldots, 2(M + 1)(N + 1)\} \).

(A) If \( i = j, i = M(N + 1) + j, i = (M + 1)(N + 1) + j, i = (2M + 1)(N + 1) + j \) for some \( j \in \{1, 2, \ldots, N + 1\} \), or \( i = j(N + 1) + 1, i = (j + 1)(N + 1), i = (M + j + 1)(N + 1) + 1 \), or \( i = (M + j + 2)(N + 1) \) for \( j \in \{1, \ldots, M - 1\} \), then the \( i \)th component of \( b^k_{\delta_1} \) is equal to 1 for Dirichlet or 0 for Neumann conditions.

(B) If \( i \leq (M + 1)(N + 1) \) and does not satisfy condition (A), then the \( i \)th component takes the form:

\[ b^k_i = \phi^k_{m,n} - 2R_x^s - 2R_y^s - s^k_{m,n}, \]  
(4.5.145)

for suitable \( m \in \{1, \ldots, M - 1\} \) and \( n \in \{1, \ldots, N - 1\} \). Therefore observe that:

\[ b^k_i = 1 + K_1 \Delta t \frac{u^k_{m,n}}{K_4 + s^k_{m,n}} - s^k_{m,n} \geq 1 - s^k_{m,n} \geq 0. \]  
(4.5.146)

(C) If \( i \) does not satisfy conditions (A) or (B), then the \( i \)th component takes the form:

\[ b^k_i = \psi^k_{m,n,x} + \psi^k_{m,n,y} + \chi^k_{m,n} + \psi^k_{m,n,y} + \psi^k_{m,n,x} - u^k_{m,n}, \]  
(4.5.147)

for suitable \( m \in \{1, \ldots, M - 1\} \) and \( n \in \{1, \ldots, N - 1\} \). Therefore observe that:

\[ b^k_i = 1 + K_2 \Delta t_k - K_3 \Delta t_k \frac{s^k_{m,n}}{K_4 + s^k_{m,n}} - u^k_{m,n} > 0. \]  
(4.5.148)
Therefore by our initial assumption, the conditions are satisfied and \( \mathbf{0}_{\delta_1} \leq \mathbf{v}_{\delta_1}^{k+1} < \mathbf{0}_{\delta_1} \).

\[ \square \]

**Proposition 5.12.** (Boundedness of solutions to \( \mathbf{e}_2 \)) Let \( \mathbf{0}_{\delta_2} \leq \mathbf{v}_{\delta_2}^k < \mathbf{e}_{\delta_1} \) for some \( k \in \{0, 1, \ldots, K - 1\} \). If \( (1 - \Delta t_k \max\{\frac{a_5}{K^2}, \kappa_L + \max\{\kappa_E, \kappa_I\}\})\mathbf{e}_{\delta_2} - \mathbf{v}_{\delta_2}^k > \mathbf{0}_{\delta_2} \) is satisfied, then \( 0 \leq \mathbf{v}_{\delta_2}^{k+1} < 1 \).

**Proof.** We define the following:

\[ \mathbf{x}_{\delta_2}^{k+1} = \mathbf{e}_{\delta_2} - \mathbf{v}_{\delta_2}^{k+1} \quad (4.5.149) \]

where \( \mathbf{e}_{\delta_2} \) is the vector of the same dimension as \( \mathbf{v}_{\delta_2}^{k+1} \) whose components are all equal to 1. We can rewrite our implicit problem as:

\[ \mathbf{M}_{\delta_2}^k \mathbf{x}_{\delta_2}^{k+1} = \mathbf{b}_{\delta_2}^k, \quad (4.5.150) \]

where

\[ \mathbf{b}_{\delta_2}^k = \mathbf{M}_{\delta_2}^k \mathbf{e}_{\delta_2} - \mathbf{v}_{\delta_2}^k. \quad (4.5.151) \]

We claim that \( \mathbf{b}_{\delta_2}^k \geq 0 \). Let \( i \in \{1, 2, \ldots, 4(M + 1)\} \).

(A) If \( i = q(M + 1) + 1, i = q(M + 1) + 2, i = (q + 1)(M + 1) - 1, \) or \( i = (q + 1)(M + 1) \)

for some \( q \in \{0, 1, 2, 3\} \), then the \( i \)th component of \( \mathbf{b}_{\delta_2}^k \) is equal to 1 for Dirichlet or 0 for Neumann conditions.

(B) If \( i \leq (M + 1) \) and does not satisfy condition (A), then the \( i \)th component takes the form:

\[ b_i^k = \alpha_{m}^{k-} + \beta_{m}^{k,S} + \alpha_{m}^{k+} + \epsilon_{m}^{k,1} + \eta_{m}^{k} - S_{m}^{k}, \quad (4.5.152) \]

51
for suitable $m \in \{1, \ldots, M - 1\}$. Therefore observe that:

\[ b_i^k = 1 + a_1 \frac{\Delta t_k X_m^k}{\kappa S_m + S_m^k_m} - a_2 \frac{\Delta t_k}{\kappa S_m + S_m^k_m} - a_3 \frac{\Delta t_k}{\kappa S_m + S_m^k_m} - S_m^k \]  \hspace{1cm} (4.5.153)

\[ b_i^k > 1 - a_2 \frac{\Delta t_k}{\kappa S_m + S_m^k_m} - a_3 \frac{\Delta t_k}{\kappa S_m + S_m^k_m} - S_m^k \]  \hspace{1cm} (4.5.154)

\[ b_i^k > 1 - (\kappa_L + \kappa_E) \Delta t_k - S_m^k \]  \hspace{1cm} (4.5.155)

(C) If $(M + 1) < i \leq 2(M + 1)$ and does not satisfy condition (A), then the $i$th component takes the form:

\[ b_i^k = \alpha_m^{k-} + \beta_{m}^{k,X} + \alpha_m^{k+} + \gamma_m^{k,1} - X_m^k \]  \hspace{1cm} (4.5.156)

for suitable $m \in \{1, \ldots, M - 1\}$. Therefore observe that:

\[ b_i^k = 1 - a_4 \frac{\Delta t_k}{\kappa S_m + S_m^k_m} + \mu \frac{\Delta t_k X_m^k}{\kappa S_m + S_m^k_m} - X_m^k \]  \hspace{1cm} (4.5.157)

\[ b_i > 1 - a_4 \frac{\Delta t_k}{\kappa S_m + S_m^k_m} - X_m^k \]  \hspace{1cm} (4.5.158)

\[ b_i > 1 - (\kappa_L + \kappa_1) \Delta t_k - X_m^k \]  \hspace{1cm} (4.5.159)

(D) If $2(M + 1) < i \leq 3(M + 1)$ and does not satisfy condition (A), then the $i$th component takes the form:
\[ b^k_i = \alpha^k_m - \beta^k_m + \alpha^k_{m^+} + \gamma^k_{2m} - I_m^k \]  
\hspace{1cm} (4.5.160)

for suitable \( m \in \{1, \ldots, M - 1\} \). Therefore observe that:

\[ b^k_i = 1 + a_4 \frac{\Delta t^k}{\kappa_S + S^k_m} - I_m^k \]  
\hspace{1cm} (4.5.161)

\[ b^k_i > 1 - I_m^k \]  
\hspace{1cm} (4.5.162)

(E) If \( i > 3(M + 1)(N + 1) \) and does not satisfy condition (A), then the \( i \)th component takes the form:

\[ b^k_i = \alpha^k_m - \beta^k_{m^E} + \alpha^k_{m^+} + \gamma^k_{2m^E} - E_m^k \]  
\hspace{1cm} (4.5.163)

for suitable \( m \in \{1, \ldots, M - 1\} \). Therefore observe that:

\[ b^k_i = 1 + a_3 \frac{\Delta t^k}{\kappa_S + S^k_m} - a_5 \frac{\Delta t^k X_m^k}{\kappa_S + S^k_m} - E_m^k \]  
\hspace{1cm} (4.5.164)

\[ b^k_i > 1 - a_5 \frac{\Delta t^k}{\kappa_S} - E_m^k \]  
\hspace{1cm} (4.5.165)

Therefore by our initial assumption, the conditions are satisfied and \( 0_{\delta_2} \leq v_{\delta_2}^{k+1} < e_{\delta_1} \).

\[ \Box \]

**Proposition 5.13.** (Boundedness of solutions to \( \delta_3 \)) Let \( 0_{\delta_3} \leq v_{\delta_3}^k < e_{\delta_3} \) for some \( k \in \{0, 1, \ldots, K - 1\} \). If \( (1 - \Delta t_k \max\{a_5^{\kappa_S}, \kappa_L + \max\{\kappa_E, \kappa_I\}\})e_{\delta_3} - v_{\delta_3}^k > 0_{\delta_3} \) is satisfied, then \( 0_{\delta_3} \leq v_{\delta_3}^{k+1} < e_{\delta_3} \).
Proof. We define the following:

\[ x^{k+1}_E = e_E - v^{k+1}_E \]  \hspace{1cm} (4.5.166)

where \( e_E \) is the vector of the same dimension as \( v^{k+1}_E \) whose components are all equal to 1.

We can rewrite our implicit problem as:

\[ M^{k}_E x^{k+1}_E = b^{k}_E, \]  \hspace{1cm} (4.5.167)

where

\[ b^{k}_E = M^{k}_E e_E - v^{k}_E. \]  \hspace{1cm} (4.5.168)

We claim that \( b^{k}_E \geq 0 \). Let \( i \in \{1, 2, \ldots, 4(M+1)(N+1)\} \).

(A) If \( i = j, \) \( i = M(N + 1) + j, \) \( i = (M + 1)(N + 1) + j, \) \( i = (2M + 1)(N + 1) + j, \)
\( i = 2(M + 1)(N + 1) + j, \) \( i = (3M + 2)(N + 1) + j, \) \( i = 3(M + 1)(N + 1) + j, \)
and \( i = (4M + 3)(N + 1) + j \) for some \( j \in \{1, 2, \ldots, N + 1\} \), or \( i = j(N + 1) + 1, \)
\( i = (j + 1)(N + 1), \) \( i = (M + j + 1)(N + 1) + 1, \) \( i = (M + j + 2)(N + 1), \) \( i = (2M+1)+(N+1) + 1, \)
i = (2(M + 1) + j)(N + 1) + 1, \( i = (2(M + 1) + j + 1)(N + 1), \) \( i = (3(M + 1) + j)(N + 1) + 1, \)
and \( i = (3(M + 1) + j + 1)(N + 1) \) for \( j \in \{1, \ldots, M - 1\} \), then the \( i \)th component of \( b^{k}_E \) is equal to 1 for Dirichlet or 0 for Neumann conditions.

(B) If \( i \leq (M + 1)(N + 1) \) and does not satisfy condition (A), then the \( i \)th component takes the form:

\[ b^{k}_i = \alpha_{m,n,x}^{k} + \alpha_{m,n,y}^{k} + \beta_{m,n}^{k}S_{m,n} + \alpha_{m,n,y}^{k+} + \alpha_{m,n,x}^{k+} + \alpha_{m,n}^{k+1} + \eta_{m,n}^{k} - S_{m,n}, \]  \hspace{1cm} (4.5.169)

for suitable \( m \in \{1, \ldots, M - 1\} \) and \( n \in \{1, \ldots, N - 1\} \). Therefore observe that:
\[ b_k^i = 1 + a_1 \frac{\Delta t_k X_{m,n}^k}{\kappa S + S_{m,n}^k} - a_2 \frac{\Delta t_k}{\kappa S + S_{m,n}^k} - a_3 \frac{\Delta t_k}{S_{m,n}^k} - S_{m,n}^k \] (4.5.170)

\[ b_k^i > 1 - a_2 \frac{\Delta t_k}{\kappa S + S_{m,n}^k} - a_3 \frac{\Delta t_k}{S_{m,n}^k} - S_{m,n}^k \] (4.5.171)

\[ b_k^i > 1 - (\kappa_L + \kappa_E) \Delta t_k - S_{m,n}^k \] (4.5.172)

(C) If \((M + 1)(N + 1) < i \leq 2(M + 1)(N + 1)\) and does not satisfy condition (A), then the \(i^{th}\) component takes the form:

\[ b_k^i = \alpha_{m,n,x}^{k,-} + \alpha_{m,n,y}^{k,-} + \beta_{m,n}^{k,x} + \alpha_{m,n,y}^{k,+} + \alpha_{m,n,x}^{k,+} + \gamma_{m,n}^{k,1} - X_{m,n}^k \] (4.5.173)

for suitable \(m \in \{1, \ldots, M - 1\}\) and \(n \in \{1, \ldots, N - 1\}\). Therefore observe that:

\[ b_k^i = 1 - a_4 \frac{\Delta t_k}{\kappa S + S_{m,n}^k} + \mu \frac{\Delta t_k X_{m,n}^k}{S_{m,n}^k} - X_{m,n}^k \] (4.5.174)

\[ b_k^i > 1 - a_4 \frac{\Delta t_k}{\kappa S + S_{m,n}^k} - X_{m,n}^k \] (4.5.175)

\[ b_k^i > 1 - (\kappa_L + \kappa_I) \Delta t_k - X_{m,n}^k \] (4.5.176)

(D) If \(2(M + 1)(N + 1) < i \leq 3(M + 1)(N + 1)\) and does not satisfy condition (A), then the \(i^{th}\) component takes the form:

\[ b_k^i = \alpha_{m,n,x}^{k,-} + \alpha_{m,n,y}^{k,-} + \beta_{m,n}^{k,i} + \alpha_{m,n,y}^{k,+} + \alpha_{m,n,x}^{k,+} + \gamma_{m,n}^{k,2} - I_{m,n}^k \] (4.5.177)
for suitable $m \in \{1, \ldots, M - 1\}$ and $n \in \{1, \ldots, N - 1\}$. Therefore observe that:

$$b^k_i = 1 + a_4 \frac{\Delta t_k}{\kappa_S + S^k_{m,n}} - I^k_{m,n}$$  \hfill (4.5.178)$$

$$b^k_i > 1 - I^k_{m,n}$$  \hfill (4.5.179)$$

(E) If $i > 3(M + 1)(N + 1)$ and does not satisfy condition (A), then the $i$th component takes the form:

$$b^k_i = a_{i,m,n,x} + a_{i,m,n,y} + b_{i,m,n}^k + a_{i,m,n,y} + a_{i,m,n,x} + \gamma_{i,m,n}^k - E^k_{m,n}$$  \hfill (4.5.180)$$

for suitable $m \in \{1, \ldots, M - 1\}$ and $n \in \{1, \ldots, N - 1\}$. Therefore observe that:

$$b^k_i = 1 + a_3 \frac{\Delta t_k}{\kappa_S + S^k_{m,n}} - a_5 \frac{\Delta t_k X^k_{m,n}}{\kappa_S + S^k_{m,n}} - E^k_{m,n}$$  \hfill (4.5.181)$$

$$b^k_i > 1 - a_5 \frac{\Delta t_k}{\kappa_S} - E^k_{m,n}$$  \hfill (4.5.182)$$

Therefore by our initial assumption, the conditions are satisfied and $0_{\varepsilon_3} \leq \mathbf{v}_{\varepsilon_3}^{k+1} < \varepsilon_{\varepsilon_3}$.

\[ \square \]

Up to this point, we have not assured total mass boundedness for $\varepsilon_2$ or $\varepsilon_3$, so that we can guarantee our sums $U^{k+1}_m = X^{k+1}_m + I^{k+1}_m + E^{k+1}_m < 1$ and $U^{k+1}_{m,n} = X^{k+1}_{m,n} + I^{k+1}_{m,n} + E^{k+1}_{m,n} < 1$ pointwise for the discretized system. We start by observing that summing the rows of the final three equations of eqs. (4.3.20) and (4.3.21) yields:
\[\delta^+_t S_m = d^S \epsilon_x S_m - a_1 \frac{X_m S_{m+1}}{\kappa_S + S_m} + a_2 \frac{X_m}{\kappa_S + S_m} + a_3 \frac{E_m}{\kappa_S + S_m}, \tag{4.5.183}\]
\[\delta^+_t U_m = d^U \epsilon_x U_m + \left(\mu + a_5\right) \frac{X_m S_{m+1}}{\kappa_S + S_m} - a_3 \frac{E_m}{\kappa_S + S_m} \]

for \( \mathcal{E}_2 \), and for \( \mathcal{E}_3 \):
\[\delta^+_t S_{m,n} = d^S (\epsilon_x + \epsilon_y) S_{m,n} - a_1 \frac{X_{m,n} S_{m,n+1}}{\kappa_S + S_{m,n}} + a_2 \frac{X_{m,n}}{\kappa_S + S_{m,n}} + a_3 \frac{E_{m,n}}{\kappa_S + S_{m,n}}, \tag{4.5.184}\]
\[\delta^+_t U_{m,n} = d^U (\epsilon_x + \epsilon_y) U_{m,n} + \left(\mu + a_5\right) \frac{X_{m,n} S_{m,n+1}}{\kappa_S + S_{m,n}} - a_3 \frac{E_{m,n}}{\kappa_S + S_{m,n}}. \]

It is worth noticing that we can rewrite implicit forms of the total biomass \( U_m \) for the \( \mathcal{E}_2 \) equations in eq. (4.5.183) as:
\[\alpha_m^{k,-} S_{m-1} + \beta_m^{k,S} S_{m+1} + \alpha_m^{k,+} S_{m+1} + \zeta_m^{k,1} X_m^{k+1} + \eta_m^{k} E_m^{k+1} = S_m \tag{4.5.185}\]
\[\alpha_m^{k,-} U_{m-1} + \beta_m^{k,M} U_{m+1} + \alpha_m^{k,+} U_{m+1} = U_m - \delta_m^k. \tag{4.5.186}\]

where we define \( \alpha_m^{k,\pm} \), \( \beta_m^{k,S} \), \( \zeta_m^{k,1} \), and \( \eta_m^{k} \) as before and:
\[\beta_m^{k,U} = 1 - \alpha_m^{k,\pm} - \alpha_m^{k,\pm} - \Delta t_k (\mu + a_5) X_m^k \frac{X_m}{\kappa_S + S_m}, \tag{4.5.187}\]
\[\delta_m^k = \frac{\Delta t_k (\mu + a_5) X_m^k (X_m^{k+1} + I_m^{k+1}) + a_3 \Delta t_k E_m^{k+1}}{\kappa_S + S_m}. \tag{4.5.188}\]
The definitions are similar for $U_{m,n}^k$ for the $\varepsilon_3$ equations in eq. (4.5.184) but with the added spatial dimension:

$$
\alpha_{m,n,x}^{k,-}S_{m-1,n}^{k+1} + \alpha_{m,n,y}^{k,-}S_{m,n-1}^{k+1} + \beta_{m,n}^{k,S}S_{m,n}^{k+1} + \alpha_{m,n,y}^{k,+}S_{m,n+1}^{k+1} \\
+ \alpha_{m,n,x}^{k,+}S_{m+1,n}^{k+1} + \zeta_{m,n}^{k,1}X_{m,n}^{k+1} + \eta_{m,n}^{k}E_{m,n}^{k+1} = S_{m,n}^{k} \tag{4.5.189}
$$

where we define $\alpha_{m,n,z}^{k,\pm}$, $\beta_{m,n}^{k,S}$, $\zeta_{m,n}^{k,1}$, and $\eta_{m,n}^{k}$ as before and:

$$
\beta_{m,n}^{k,U} = 1 - \alpha_{m,n,x}^{k,-} - \alpha_{m,n,y}^{k,-} - \alpha_{m,n,y}^{k,+} - \alpha_{m,n,x}^{k,+} - \frac{\Delta t_k (\mu + a_5) X_{m,n}^{k}}{\kappa_S + S_{m,n}^{k}}, \tag{4.5.191}
$$

$$
\delta_{m,n}^{k} = \frac{\Delta t_k (\mu + a_5) X_{m,n}^{k} (X_{m,n}^{k+1} + I_{m,n}^{k+1}) + a_3 \Delta t_k E_{m,n}^{k+1}}{\kappa_S + S_{m,n}^{k}}. \tag{4.5.192}
$$

It is apparent from this structure that we can compose a matrix similar to the $(M+1) \times (M+1)$ size $B_{\varepsilon_2}^{k,U}$ and $(M+1)(N+1) \times (M+1)(N+1)$ size $B_{\varepsilon_3}^{k,U}$ block matrices for $\varepsilon_2$ and $\varepsilon_3$, respectively, for $U_m^k$ and $U_{m,n}^k$, defined as the $B_{\varepsilon_2}^{k,U}$ and $B_{\varepsilon_3}^{k,U}$ block matrices. By the linearity of the discrete operators in the model and assuming our diffusion constant $d^U$ (and therefore $R_z^U$) is the same for all biomass constituents, $B_{\varepsilon_2}^{k,U}$ and $B_{\varepsilon_3}^{k,U}$ will be identically defined to those assembled in [35]. With some algebraic effort, we can rewrite a system of linear equations for just the evolution of length $(M+1)$ vectors $U_{\varepsilon_2}^k$ and $\delta_{\varepsilon_2}^k$ defined as:

58
with both initial vectors $\mathbf{U}_\mathcal{E}_2^0$ and $\delta_\mathcal{E}_2^0$, and vectors prior to iteration given the fixed Neumann or Dirichlet boundary conditions after a previous iteration:

\[
\mathbf{U}_\mathcal{E}_2^k = (0, U_0^k, \ldots, U_{M-1}^k, 0) \quad \text{(4.5.195)}
\]
\[
\delta_\mathcal{E}_2^k = (0, \delta_1^k, \ldots, \delta_{M-1}^k, 0) \quad \text{(4.5.196)}
\]

The evolution of this system is (keeping in mind once again that the initial condition would have $\mathbf{U}_\mathcal{E}_2^0$ and $\delta_\mathcal{E}_2^0$ on the right hand side and $\mathcal{B}_{\mathcal{E}_2}^{0,U} \mathbf{U}_{\mathcal{E}_2}^1$ on the left):

\[
\mathcal{B}_{\mathcal{E}_2}^{k,U} \mathbf{U}_{\mathcal{E}_2}^{k+1} = \mathbf{U}_{\mathcal{E}_2}^k - \delta_{\mathcal{E}_2}^k \quad \text{(4.5.197)}
\]

A system of linear equations for just the evolution of length $(M + 1)(N + 1)$ vectors $\mathbf{U}_{\mathcal{E}_3}^k$ and $\delta_{\mathcal{E}_3}^k$ defined as:

\[
\mathbf{U}_{\mathcal{E}_3}^k = (U_{0,0}^k, U_{0,1}^k, \ldots, U_{0,N}^k, U_{1,0}^k, U_{1,1}^k, \ldots, U_{1,N}^k, \ldots, U_{M,0}^k, U_{M,1}^k, \ldots, U_{M,N}^k) \quad \text{(4.5.198)}
\]

59
\[ \delta^k_{\mathcal{E}_3} = (\delta^k_{0,0} \cdot \delta^k_{0,1}, \ldots, \delta^k_{0,N}, \delta^k_{1,0}, \delta^k_{1,1}, \ldots, \delta^k_{1,N}, \ldots, \delta^k_{M,0}, \delta^k_{M,1}, \ldots, \delta^k_{M,N}) \]  

(4.5.199)

with initial vectors \( U_0^{\mathcal{E}_3} \) and \( \delta^0_{\mathcal{E}_3} \), or previous iteration vectors \( U^k_{\mathcal{E}_3} \) and \( \delta^k_{\mathcal{E}_3} \) given their fixed Neumann or Dirichlet boundary conditions:

\[ U^k_{\mathcal{E}_3} = (0, 0, \ldots, 0, U^k_{1,1}, \ldots, U^k_{1,N-1}, 0, \ldots, \ldots, 0, U^k_{M-1,1}, \ldots, U^k_{M-1,N}, 0, 0, \ldots, 0) \]  

(4.5.200)

\[ \delta^k_{\mathcal{E}_3} = (0, 0, \ldots, 0, \delta^k_{1,1}, \ldots, \delta^k_{1,N-1}, 0, \ldots, \ldots, 0, \delta^k_{M-1,1}, \ldots, \delta^k_{M-1,N}, 0, 0, \ldots, 0) \]  

(4.5.201)

will be:

\[ B^k_{\mathcal{E}_3} \cdot U^{k+1}_{\mathcal{E}_3} = U^k_{\mathcal{E}_3} - \delta^k_{\mathcal{E}_3} \]  

(4.5.202)

**Proposition 5.14.** (Mass boundedness for \( \mathcal{E}_2, \mathcal{E}_3 \)). We let \( k \in \{0, 1, \ldots, K-1\} \) and require for \( j \in \{2, 3\} \):

\[ U^k_{\mathcal{E}_j} < (1 - \frac{\Delta t_k (\mu + a_5)}{\kappa_S}) e_{\mathcal{E}_j}. \]  

(4.5.203)
We further require the assumptions made by Propositions (5.9), (5.12), and (5.13) hold. Let $0_{\mathcal{E}_j} \leq v^k_{\mathcal{E}_j} < e_{\mathcal{E}_j}$ for some $k \in \{0, 1, \ldots, K-1\}$. If $(1 - \Delta t \max\{\frac{a_5}{K^5}, \kappa_L + \max\{\kappa_E, \kappa_I\}\}) e_{\mathcal{E}_j} - v^k_{\mathcal{E}_j} > 0_{\mathcal{E}_j}$ is satisfied for $j \in \{2, 3\}$, then $0_{\mathcal{E}_j} \leq v^{k+1}_{\mathcal{E}_j} < e_{\mathcal{E}_j}$ and therefore $U^{k+1}_{\mathcal{E}_j} < e_{\mathcal{E}_j}$.

Proof. The proof is an immediate consequence of our assumptions and is identical to the one made in [35] so will not be included here.

We have thus concluded that for all experiments, solutions are bounded by 0 and 1 elementwise. Analytically speaking there is difficulty in showing that all experiments possess bounded unique solutions, and in fact the argument made for $\mathcal{E}_1$ in [21] concludes that a purely homogeneous Neumann boundary value problem can become unbounded in finite time (due to the physical possibility of a source term supplying an infinite amount of substrate).

We can however provide a proposition for the existence and uniqueness of our numerical schemes, making the assumptions on the parameters as before.

**Proposition 5.15.** (Existence and uniqueness for $\mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3$) Suppose that $0_{\mathcal{E}_j} \leq v^k_{\mathcal{E}_j} < e_{\mathcal{E}_j}$ for $j = \{1, 2, 3\}$ and $(\Delta t)_k$ are available for $k \in \mathbb{Z}^+ \cup \{0\}$. Let $n_k$ be the smallest nonnegative integer such that $(1 + 2^{-n_k}(K_2 - K_3)(\Delta t)_k)e_{\mathcal{E}_j} - v^k_{\mathcal{E}_j} > 0_{\mathcal{E}_j}$ in $\mathcal{E}_1$, or the smallest nonnegative integer such that $(1 - 2^{-n_k} \max\{\frac{a_5}{K^5}, \kappa_L + \max\{\kappa_E, \kappa_I\}\})(\Delta t)_k e_{\mathcal{E}_j} - v^k_{\mathcal{E}_j} > 0_{\mathcal{E}_j}$ in $\mathcal{E}_j = \{\mathcal{E}_2, \mathcal{E}_3\}$, for every $m \in \{1, \ldots, M-1\}$ and $n \in \{1, \ldots, N-1\}$. Let $(\Delta t)_{k+1} = (\Delta t)_k / 2^{n_k}$. If $0 \leq v^0_{\mathcal{E}_j} < 1$ is satisfied and $(\Delta t)_0 > 0$, then there exists a unique sequence $(v^0_{\mathcal{E}_j})_{k=0}^{\infty}$ satisfying the above recursive algorithm, and $0_{\mathcal{E}_j} \leq v^k_{\mathcal{E}_j} < e_{\mathcal{E}_j}$ for every $k \in \mathbb{Z}^+$.

Proof. The proof is an immediate consequence of our assumptions in that $(\Delta t)_{k+1} > 0$ always exists (we simply can refine the resolution of $\Delta t$ at each step $k$ of the algorithm).

**Proposition 5.16.** (Stability of the M-matrix method to solutions of $\mathcal{E}_j$) Suppose that we have constructed an M-matrix per the prescription of our numerical method above and suppose also that for $j = \{1, 2, 3\}$, $(\Delta t)_k$ are available for $k \in \mathbb{Z}^+ \cup \{0\}$. The solution to $v^{k+1}_{\mathcal{E}_j} = (M^k_{\mathcal{E}_j})^{-1} v^k_{\mathcal{E}_j}$ is stable for each successive step in $k$. 

61
Proof. The proof as follows is a consequence of the properties of the inverse of the M-matrix \cite{36,37}, \cite{38} and the outline for stability cited in \cite{39}. After each step $k$ of the algorithm where we have guaranteed a new M-matrix for the corresponding $\Delta t_k (M^k_{\delta_j})$, we have that after performing successive steps:

$$
0_{\delta_j} \leq v^{K+1}_{\delta_j} = \prod_{k=0}^{K} (M^k_{\delta_j})^{-1}v^k_{\delta_j} < e_{\delta_j}
$$

(4.5.204)

In view of the spectral radius of the inverse of the M-matrix $\rho((M^k_{\delta_j})^{-1}) < 1$ we have a product of values always less than 1 and stability is assured so long as the $\Delta t_k$ chosen provides us with an M-matrix at each step.

\[\square\]

In the algorithm used to determine the next solution, a test to verify that all eigenvalues of the M-matrix are greater than 1 is used to ensure that we can provide this stability. If there were such an eigenvalue (no test cases have shown this), we would simply decrease the value of $\Delta t$. 

62
5 Numerical Implementation

The finite difference methods for $E_1$-$E_3$ were implemented in MATLAB® using multiple versions (R2009, R2015b, R2016a) and using different computer systems including an ASUS Q500A laptop running the 64 bit Windows 8.1 operating system, an HP running Windows 10 64 bit, and an HP running 64 bit Linux Fedora 25. All results took various times across the different machines and versions of MATLAB®, but the same code was used for the purpose of showing reproducibility with different machines. The results were identical up to machine epsilon verifying the reproducible nature of the experimental simulations, and the code is included in the appendix.

Due to the considerable memory overhead of the large M-matrix at step-sizes $\Delta x = \Delta y = 0.01$ (which consists of $4(M+1)(N+1) \times 4(M+1)(N+1) \approx 1.655 \times 10^9$ size double entries for $E_3$), continual use of the internal subroutine sparse efficiently compacted the submatrices that contained zeros off of the banded diagonal region. A call to sparse takes the coordinate values of the nonzero entries of the matrix and assigns them to the coordinates of an array. It should be noted that most of the traditional operations of matrix manipulation are also possible using the sparse version of the original matrix. It is worth mentioning that this method is conventional for determining solutions to implicit systems of PDEs.

The internal subroutines tridiag and blocktridiag were used to minimize computational complications of assembly. These two subroutines vectorize what would normally be a collection of for loops that place coordinate values in the matrices in the overall pro-
gram, thus increasing the speed of the computations. Previous sub-matrices are cleared from system memory through each iteration. Preliminary inspection of nonzero entries used a variation on the subroutine **spy** which colored all nonzero entries for a quick visualization of correct block matrix setup.

Upon assembly of the M-matrix, we need to solve the linear system of equations $M^k e_i v^{k+1} = v^k e_i$ for $v^{k+1}$. As with all large and sparse banded matrices, inversion methods for the system that are faster than Gaussian elimination are of paramount importance. There numerous methods that can be implemented to solve the system, but a good experimental approach is to test certain algorithms for fast convergence and stability. To this end, the subroutines offered in MATLAB® offer significant increases in speed and stability in the solution of these sparse systems. In our case we provide the example of 11 different iterative methods built into MATLAB®. We use initial values for $e_3$ listed in The 11 iterative methods used for solution to the sparse matrices correspond to **bicg** (biconjugate gradients method), **bicgstab** (biconjugate gradients stabilized method), **bicgstabl** (biconjugate gradients stabilized (l) method), **cgs** (conjugate gradients squared method), **gmres** (generalized minimum residual method (with restarts)), **lsqr** (LSQR method), **minres** (minimum residual method), **pcg** (preconditioned conjugate gradients method), **qmr** (quasi-minimal residual method), **symmlq** (symmetric LQ method), and **tfqmr** (transpose-free quasi-minimal residual method), and Neumann boundary conditions to give an example of the convergence of the different methods at the final iteration when $T = 1$. Figure 5.1a represents each of the methods using a tolerance of $1 \times 10^{-8}$ and a maximum number of 40 iterations of the solver subroutine. As can be clearly seen, some methods fail to converge entirely. In Figure 5.1b, we provide a preconditioner to the iterative method by way of LU decomposition. The results show how important a preconditioner is to solution methods. It also gives some insight to which methods are better for this particular problem.
Figure 5.1: Plot of residuals vs. iterations of 11 iterative schemes for one time step of $\delta_3$ (a) without and (b) with preconditioning LU matrix.
Notice in Figure 5.1a that only four iterative methods even manage to converge to a relative residual of $1 \times 10^{-8}$ before the allotted 40 iterations. These iterative methods would be deemed unstable and useless, so careful consideration of the proper technique is paramount in investigation of properly convergent and stable methods to find a potential solution. In our case, due to its stable and convergence properties, we will use the internal subroutine \texttt{bicgstab} (a MATLAB® implementation of the stabilized bi-conjugate gradient method) using a tolerance of $1 \times 10^{-8}$, and a maximum number of iterations equal to 40 with the same same preconditioner in all our experiments. Our results follow below, where specific parameter settings, initial profiles, and boundary conditions for the substrate and biomass system have been indicated for every case.
6 Illustrative Results

1 Results for $\mathcal{E}_1$

Beginning with $\mathcal{E}_1$ we will now provide some illustrative examples to show the behavior of each problem in turn. Throughout $\mathcal{E}_1$ we take $\Omega = [0,1] \times [0,1]$ (without loss of generality a normalized square grid with equal spacing allows the study to focus on the problem dynamics, so it will be used throughout) and give the initial value of $\Delta t = 0.01$ (although remember this value could be made smaller to support our assumptions in each case).

Example 1.1. Let us consider the dynamics described by $\mathcal{E}_1$ for both Dirichlet and Neumann conditions on $\Omega = [0,1] \times [0,1]$. Fix the model parameter values $d^s = 0.0001$, $K_1 = 0$, $K_2 = 0$, $K_3 = 0.4$, $K_4 = 0$, $\alpha = 4$, and $\beta = 4$. Fix $\Delta x = 0.01$, $\Delta y = 0.01$ and $\Delta t = 0.01$, and consider initial conditions of the form:

$$s^0(x) = 1, \forall x \in \Omega, \quad (6.1.1)$$
$$u^0(x) = \sum_{l=1}^{L} C_le^{-r_l||x-x_l||^2}, \forall x \in \Omega \quad (6.1.2)$$

where $L = 5$, $C_1 = 0.025$, $C_2 = 0.03$, $C_3 = 0.035$, $C_4 = 0.02$, $C_5 = 0.025$, $r_1 = 25$, $r_2 = 50$, $r_3 = 125$, $r_4 = 100$, $r_5 = 50$, and $x_1 = (0.25,0.3)$, $x_2 = (0.5,0.25)$, $x_3 = (0.7,0.65)$, $x_4 = (0.4,0.8)$, $x_5 = (0.5,0.55)$. Figures 6.1, 6.2, and 6.3 present the results of our simulations.
at 9 different times, namely \( t = \{0, 1, 2, 3, 4, 5, 6, 7, 8\} \). Results are nearly equivalent to our work cited in [30] with slight modifications to ensure the WCDD properties using Neumann conditions.

Our results, although somewhat unrealistic in Example 1.1 would describe a system with an unending, nondiffusive nutrient source and a nondecaying active biomass. As would be expected, the biomass grows to the maximum permitted value and slowly fills the entire grid. At a certain point this model becomes physically unrealistic not only for the nonphysical parameter settings but the constant supply of nutrients being fed into a closed system with no mass flow out of the region. In this particular situation, we only are evincing the bounded nature of the solutions, whereby \( \Delta t \) will continue decreasing as all of the space is filled. It is worthy of note that Dirichlet and Neumann conditions in this case would have no solution analytically, so this is one way of testing how some parameters can be physically unreasonable through numerical trial and error.

Next we consider a less trivial case where the substrate concentration is not a constant function. In this example we set the maximum specific consumption rate of the system to a relatively high value, giving us the physical representation of a rapid decrease in substrate concentration.

**Example 1.2.** Let us consider the dynamics described by \( \mathcal{E}_1 \) for both Dirichlet and Neumann conditions on \( \Omega = [0, 1] \times [0, 1] \). Fix the model parameter values \( d^s = 0.002, d^u = 0.0001, K_1 = 0.85, K_2 = 0.0012, K_3 = 0.4, K_4 = 0.3, \alpha = 4, \) and \( \beta = 4 \). Fix \( \Delta x = 0.01, \Delta y = 0.01 \) and \( \Delta t = 0.01 \), and consider initial conditions identical to Example 1.1. Figures 6.4, 6.5, and 6.6 represent the numerical results for \( t = \{0, 2, 4, 6, 8, 10, 12, 14\} \). Due to the large maximum consumption rate, by \( t = 14 \) the substrate function is falling off markedly.

Figures 6.4, 6.5, and 6.6 of Example 1.2 show that by making the biomass growth rate large relative to its decay rate produces a similar result as in Example 1.1 when the
Figure 6.1: Time course of $\mathcal{E}_1$ for Example 1.1 when $t = \{0, 1, 2\}$ with (a) homogeneous Dirichlet and (b) homogeneous Neumann conditions.
Figure 6.2: Continued time course of $\mathcal{E}_1$ for Example 1.1 when $t = \{3, 4, 5\}$. 

70
Figure 6.3: Continued time course of $E_1$ for Example 1.1 when $t = \{6, 7, 8\}$.
Figure 6.4: Time course of $e_1$ for Example 1.2 when $t = \{0, 2, 4\}$ with (a) homogeneous Dirichlet and (b) homogeneous Neumann conditions.
Figure 6.5: Continued time course of $E_1$ for Example 1.2 when $t = \{6, 8, 10\}$.
Figure 6.6: Continued time course of $\mathcal{E}_1$ for Example 1.2 when $t = \{12, 14\}$. 
substrate is finite but still set at a slower depletion rate than would be necessary to curtail the increase in concentration of the active biomass. The end result is we once again observe the bounded nature of the biomass, but this time the slow loss of substrate will eventually disappear completely.

Our final case for $\mathcal{E}_1$ considers a problem where the biomass decay rate is relatively high, resulting in a decrease in the microbial colony over time. As is expected the purpose is to observe that the positive character of the approximate biomass function is conserved by our technique at all times.

**Example 1.3.** In this example, we multiply the approximate final biomass function obtained in Example 1.2 times 1.4 and use this as our initial profile for both Dirichlet and Neumann conditions on $\Omega = [0, 1] \times [0, 1]$. Fix the model parameter values $d^s = 0.0015$, $d^u = 0.0001$, $K_1 = 0.65$, $K_2 = 0.36$, $K_3 = 0.2$, $K_4 = 0.3$, $\alpha = 4$, and $\beta = 4$. Fix $\Delta x = 0.01$, $\Delta y = 0.01$ and $\Delta t = 0.01$. Figure 6.7 represents numerical results for $t = \{15, 20, 30\}$. The fact that the biomass decay rate takes on a relatively high value results in a drastic decrease of the biomass function toward zero. The simulations show that the solution is decreasing, and the nonnegative and bounded character of the approximate solutions remains intact.

The side by side time course of Neumann and Dirichlet conditions finally begin to become apparent in this final Example 1.3, indicating the importance of preestablished and physically appropriate boundary conditions that unconditionally satisfy the M-matrix requirements for positive bounded solutions.

### 2 Results for $\mathcal{E}_2$

This section of examples corresponds to $\mathcal{E}_2$ where we will be only observing one spatial dimension, but now we have a more fully descriptive model of the biomass complex as three separate but interacting components. Throughout $\mathcal{E}_2$ we take $\Omega = [0, 1]$ and give the initial
Figure 6.7: Time course of $E_1$ for Example 1.3 when $t = \{15, 20, 30\}$. 

76
value of $\Delta t = 0.01$ (although keeping in mind the assumptions for bounded behavior, strict
diagonal dominance, and mass boundedness, this value can be made smaller to support our
assumptions).

**Example 2.1.** Let us consider the dynamics described by $\mathcal{E}_2$ for both Dirichlet and Neumann
conditions on $\Omega = [0,1]$. Fix the model parameter values $d^S = 0.002$, $d^U = 0.0001$, $\mu = 10$,
$Y_H = 0.03$, $Y_E = 0.03$, $\kappa_S = 0.2$, $\kappa_I = 0.04$, $\kappa_E = 0.03$, $\kappa_L = 0.01$, $\alpha = 2$, and $\beta = 2$. Fix
$\Delta x = 0.01$ and $\Delta t = 0.01$, and consider initial conditions for any $Y \in \{S,X,I,E\}$ of the
form:

$$
Y^0(x) = \sum_{l=1}^{L^Y} C_Y^l e^{-r_Y^l \|x-x_Y^l\|^2}, \forall x \in \Omega
$$

(6.2.3)

In our simulations unless otherwise noted we consider initial configurations of the following
forms:

1. For the initial profile for the substrate, $L^S = 1$, $C_1^S = 0.8$, $r_1^S = 10$, and $x_1^S = 0$.

2. For the initial profile for active biomass, $L^X = 2$, $C_1^X = 0.15$, $C_2^X = 0.2$, $r_1^X = 200$,
$\kappa_2^X = 150$, and $x_1^X = 0.3$, $x_2^X = 0.65$.

3. For the initial profile for inert biomass, $L^I = 2$, $C_1^I = 0.02$, $C_2^I = 0.01$, $r_1^I = 80$,
$\kappa_2^I = 60$, and $x_1^I = 0$, $x_2^I = 1$.

4. For the initial profile for EPS we assume a constant function $C^E = 0.005$.

The configuration stated above for Example 2.1 establishes an initial concentration
of substrate $S$ at the left wall of the boundary. The legend corresponds to each of the
substrate, active biomass $X$, EPS $E$, and inert biomass $I$ over the time course ranging from
$t = \{0, 0.25, 0.5, 0.75, 1, 1.5, 2.5, 10\}$ for both Neumann and Dirichlet conditions. In $\mathcal{E}_2$ it is
Figure 6.8: Time course of $E_2$ for Example 2.1 when $t = \{0, 0.25, 0.5\}$. 
Figure 6.9: Continued course of $\phi_2$ for Example 2.1 when $t = \{0.75, 1, 1.5\}$. 

79
Figure 6.10: Continued course of $\mathcal{E}_2$ for Example 2.1 when $t = \{2, 5, 10\}$. 

80
easier to see the fixed nature of the Dirichlet condition to the wall on the substrate over
the course of time, and this produces a very different dynamic time course than that of
the moving fixed flow boundary in the Neumann case. The two Gaussians representing the
concentrations of active biomass initially form a more rounded structure with distinctive
barrier where there is less substrate and hence the tendency to diffuse outward, but over the
time course we see slow movement toward the wall until a slow equilibrium is reached. The
EPS remains relatively constant and the inert component increases as the byproduct of the
active biomass, as expected. As the substrate vanishes due to consumption by active biomass
(instigated by a very large value for $\mu$ indicative of a high rate of growth to low yield ratio),
diffusive properties begin to take over the complex. The final time shows that Dirichlet and
Neumann conditions are indeed very different over the passage of time due to fixed versus
no flow conditions. The active biomass under Neumann conditions has completely used all
available substrate and after dying off only the inert component remains, whereas in Dirichlet
conditions the active source is still stable and not yet finished consuming all of the substrate
on the wall.

As we continue with a final, more dynamic example of $\mathcal{E}_2$, it is important to stress
the value of models using only one spatial dimension. Having simple plots of behavior for
different points in time give a great deal of insight to the inner dynamics of the phenomena,
allow parameters to be modified easily with simple results, and can be extended to multiple
dimensions with only slight changes to the constants. The single spatial dimension also allows
us to probe parameter space with higher accuracy and fewer overall computations as well
as lightening the overall consumption of computer power and memory for the simulations,
allowing for quick feedback on experimental ideas.

Example 2.2. Let us consider the dynamics described by $\mathcal{E}_2$ for both Dirichlet and Neumann
conditions on $\Omega = [0, 1]$ once again, but fix the model parameter values $d^S = 1$, $d^U = 0.001$,
$\mu = 1$, $Y_H = 1$, $Y_E = 0.03$, $\kappa_S = 0.8$, $\kappa_I = 0.04$, $\kappa_E = 0.03$, $\kappa_L = 0.03$, $\alpha = 1$, and $\beta = 6$. 

81
Fix $\Delta x = 0.01$ and $\Delta t = 0.01$, and consider initial conditions for any $Y \in \{S, X, I, E\}$ of the form:

$$Y^0(x) = \sum_{l=1}^{L^Y} C^Y_l e^{-r_l^Y ||x-x_l^Y||^2}, \forall x \in \Omega \quad (6.2.4)$$

1. For the initial profile of the substrate, $L^S = 1$, $C_1^S = 0.8$, $r_1^S = 10$, and $x_1^S = 0$.

2. For the initial profile of active biomass, $L^X = 2$, $C_1^X = 0.15$, $C_2^X = 0.2$, $r_1^X = 200$, $r_2^X = 150$, and $x_1^X = 0.3$, $x_2^X = 0.65$.

3. For the initial profile of inert biomass, $L^I = 2$, $C_1^I = 0.02$, $C_2^I = 0.01$, $r_1^I = 80$, $r_2^I = 60$, and $x_1^I = 0$, $x_2^I = 1$.

4. For the initial profile of EPS we assume the form of a small scaffolding structure of windowed Gaussians: $L^E = 6$, $C_1^E = 0.05$, $C_2^E = 0.15$, $C_3^E = 0.175$, $C_4^E = 0.19$, $C_5^E = 0.2$, $C_6^E = 0.125$, $r_1^E = 100$, $r_2^E = 200$, $r_3^E = 300$, $r_4^E = 500$, $r_5^E = 600$, $r_6^E = 400$, $x_1^E = 0.0$, $x_2^E = 0.2$, $x_3^E = 0.3$, $x_4^E = 0.45$, $x_5^E = 0.7$, and $x_6^E = 0.8$.

The configuration stated above for Example 2.2 establishes an initial concentration of substrate $S$ at the left wall of the boundary. The legend corresponds to each of the substrate, active biomass $X$, EPS $E$, and inert biomass $I$ over the time course ranging from $t = \{0, 0.5, 1, 2, 5, 10, 14, 18, 22, 26, 30, 40, 60, 80, 100\}$ for both Neumann and Dirichlet conditions. In this example, we created a larger number of windowed Gaussians to represent the EPS, set a small diffusion constant $d^U$ for the biomass complex with respect to the diffusion constant $d^S$ of the substrate, and a one to one ratio of $\mu$ to $Y_H$. We keep the yield of EPS low as would be expected if it forms a structure and should remain intact but unchanging, and make the inert biomass growth rate small. We also change the power of
the nonlinear diffusion factor \( \alpha \) to increase when the concentration is very high, thereby promoting growth in the direction of the concentration of unused substrate. The results show that active biomass travels toward the substrate location and creates inert material in its wake. The ultimate active biomass drops to zero when the substrate is consumed, and the inert component remains in its place, tending to block the active biomass from moving in those locations.

3 Results for \( \mathcal{E}_3 \)

This section of examples correspond to \( \mathcal{E}_3 \). Throughout \( \mathcal{E}_3 \) we take \( \Omega = [0, 1] \times [0, 1] \) and give the initial value of \( \Delta t \) (although keeping in mind the assumptions for bounded behavior as in \( \mathcal{E}_2 \), where for diagonal dominance and mass boundedness this value must be made smaller to support our assumptions).

Example 3.1. Let us consider the dynamics described by \( \mathcal{E}_3 \) for Neumann conditions on \( \Omega = [0, 1] \times [0, 1] \). We have omitted Dirichlet conditions in the following examples because fixing the sides as in the previous studies tends to display unphysical or unrealistic results. Although mathematically sound and numerically stable, the notion of fixing the substrate or the biomass to the boundaries would only be feasible if there were adherence properties which are somewhat more constrictive than a natural zero flow approach. With this in mind, fix the model parameter values \( d^S = 0.0002, d^U = 0.0001, \mu = 1, Y_H = 0.35, Y_E = 0.03, \kappa_S = 0.8, \kappa_I = 0.4, \kappa_E = 0.03, \kappa_L = 0.01, \alpha = 6, \) and \( \beta = 1 \). Fix \( \Delta x = 0.01, \Delta y = 0.01, \) and \( \Delta t = 0.01 \), and consider initial conditions for any \( Y \in \{S,X,I,E\} \) of the form:
Figure 6.11: Time course of $E_2$ for Example 2.2 when $t = \{0, 0.5, 1\}$. 
Figure 6.12: Continued course of $\phi_2$ for Example 2.2 when $t = \{2, 5, 10\}$.
Figure 6.13: Continued course of $E_2$ for Example 2.2 when $t = \{14, 18, 22\}$. 
Figure 6.14: Continued course of $\varepsilon_2$ for Example 2.2 when $t = \{26, 30, 40\}$.
Figure 6.15: Continued course of $E_2$ for Example 2.2 when $t = \{60, 80, 100\}$.
In our simulations unless otherwise noted we consider initial configurations of the following forms:

1. The initial profile for the substrate is a constant, which physically represents a homogeneous medium.

2. For the initial profile of active biomass, $L^X = 6$, $C^X_1 = 0.25$, $C^X_2 = 0.325$, $C^X_3 = 0.275$, $C^X_4 = 0.3$, $C^X_5 = 0.2$, $C^X_6 = 0.225$, $r^X_1 = 100$, $r^X_2 = 50$, $r^X_3 = 30$, $r^X_4 = 80$, $r^X_5 = 90$, $r^X_6 = 100$, $x^X_1 = (0.25, 0.3)$, $x^X_2 = (0.5, 0.25)$, $x^X_3 = (0.7, 0.65)$, $x^X_4 = (0.4, 0.8)$, $x^X_5 = (0.5, 0.55)$, $x^X_6 = (0.8, 0.3)$.

3. For the initial profile of inert biomass, $L^I = 5$, $C^I_1 = 0.025$, $C^I_2 = 0.03$, $C^I_3 = 0.035$, $C^I_4 = 0.2$, $C^I_5 = 0.025$, $r^I_1 = 25$, $r^I_2 = 50$, $r^I_3 = 125$, $r^I_4 = 100$, $r^I_5 = 50$, $x^I_1 = (0.3, 0.3)$, $x^I_2 = (0.55, 0.25)$, $x^I_3 = (0.76, 0.65)$, $x^I_4 = (0.45, 0.8)$, $x^I_5 = (0.55, 0.55)$.

4. For the initial profile of EPS we have $L^E = 4$, $C^E_1 = 0.3$, $C^E_2 = 0.35$, $C^E_3 = 0.4$, $C^E_4 = 0.25$, $r^E_1 = 200$, $r^E_2 = 300$, $r^E_3 = 600$, $r^E_4 = 400$, $x^E_1 = (0.2, 0.2)$, $x^E_2 = (0.8, 0.8)$, $x^E_3 = (0.3, 0.7)$, $x^E_4 = (0.7, 0.2)$.

In these examples we begin by creating a similar collection of windowed Gaussians but set the surrounding regions around each Gaussian to zero past a certain radius. This gives a more realistic representation of individual particles of the biomass. The results show that with a high concentration of substrate and relatively small yield, the biomass grows and diffuses outward slowly. Here the time course for our initial conditions stated in Example 3.1 range over the following points: $t = \{0, 0.1, 0.2, 0.3, 0.5, 0.7, 0.9, 1, 1.5, 2, 2.5, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 35\}$.
Figure 6.16: Time course of $E_3$ for Example 3.1 when $t = \{0, 0.1, 0.2, 0.3, 0.5, 0.7\}$. 
Figure 6.17: Continued course of $\mathcal{E}_3$ for Example 3.1 when $t = \{0.9, 1, 1.5, 2, 2.5, 3\}$.
Figure 6.18: Continued course of $\mathcal{E}_3$ for Example 3.1 when $t = \{4, 5, 6, 7, 8, 9\}$. 
Figure 6.19: Continued course of $e_3$ for Example 3.1 when $t = \{10, 15, 20, 25, 30, 35\}$. 
What we see in Example 3.1 is that moderate conditions with our parameters but a slightly low yield ratio result in rapid loss of substrate before the biomass has reached the point to diffuse outward and absorb adjacent regions of substrate. The result is that substrate is completely depleted locally and inert biomass grows until the biomass eventually decays to zero.

Our final experiment is an attempt to provide dynamics by small changes in the parameters. The results are intended to show that by increasing the yield and growth factor of the active biomass while holding the substrate relatively fixed in locations in proximity of the biomass, overall spreading occurs directionally toward the food source. As the diffusion takes over the particles begin to merge and consume all of the substrate within their collective area of location. The active portion slowly decays to zero after consuming all of the substrate in the region, and inert biomass concentration increases in the places where consumption was greatest.

Example 3.2. Let us consider the dynamics described by $\mathcal{E}_3$ for Neumann conditions on $\Omega = [0,1] \times [0,1]$. Fix the model parameter values $d^S = 1$, $d^U = 0.0001$, $\mu = 1$, $Y_H = 1$, $Y_E = 0.03$, $\kappa_S = 0.8$, $\kappa_I = 0.04$, $\kappa_E = 0.03$, $\kappa_L = 0.03$, $\alpha = 6$, and $\beta = 1$. Fix $\Delta x = 0.01$, $\Delta y = 0.01$, and $\Delta t = 0.01$, and consider initial conditions for any $Y \in \{S,X,I,E\}$ as in Example 3.1 but with the substrate located at two corners of the boundaries, so that for the initial state of $S$ we have $L^S = 2$, $C^S_1 = 0.7$, $C^S_2 = 0.9$, $r^S_1 = 15$, $r^S_2 = 5$, $x^S_1 = (1,0)$, $x^S_2 = (1,1)$. Leave the same initial conditions for the active biomass and EPS as in Example 3.1 but set the inert biomass initially to zero everywhere. We show the results for $t = \{0,3,6,9,12,15,18,21,24,27,30,33,36,39,42,45,48,51,54,57,60\}$.
Figure 6.20: Time course of $E_3$ for Example 3.2 when $t = \{0, 3, 6, 9, 12, 15\}$.
Figure 6.21: Continued course of $E_3$ for Example 3.2 when $t = \{18, 21, 24, 27, 30, 33\}$.
Figure 6.22: Continued course of $E_3$ for Example 3.2 when $t = \{36, 39, 42, 45, 48, 51\}$.
Figure 6.23: Continued course of $\mathcal{E}_3$ for Example 3.2 when $t = \{54, 57, 60\}$. 
7 Concluding Remarks

Beginning from three systems of diffusive PDEs in the investigation of the growth dynamics of some types of biological films, we provided a finite difference methodology to consistently approximate their solutions. The mathematical diffusion reaction models that motivate this investigation are nonlinear systems for which the analytic calculation of exact solutions is a difficult task. Although certain analytical methods have been determined in very specific cases for models related to the porous medium equations (a similar parabolic PDE with nonlinear diffusion based on power series or fractional powers of the concentration), the addition of diffusion reaction makes exact solutions intractable. The need of designing reliable, stable numerical techniques to approximate the solutions is therefore an important task of practical and numerical interest. It is important to point out that the systems of nonlinear equations that motivate this work generalizes some particular models investigated previously in the literature, like a nonlinear PDE in the investigation of the growth of certain colonies of bacteria [40], and a system of diffusive equations which describes the interaction between a substrate of nutrients and a microbial system [21]. The generalization investigated in this dissertation assumes in the second two models that the biological mass can be decomposed as the sum of three components, namely, the active biological mass, the inert mass, and the extracellular polymeric substance, which is are important ingredients in the development of biological films. The first test environment $E_1$ had two spatial dimensions so the matrix assembly was similar to $E_3$, with the restriction being to a non-compartmentalized
active biomass and substrate only. A second test $E_2$ was performed in one spatial dimension, followed by the slightly more complicated two spatial dimension case $E_3$, where in both of these problems the relative dynamics of a compartmentalized biosystem demonstrated a more realistic internal structure.

The numerical method proposed in this dissertation followed a non-local finite difference perspective, resembling some similar and non-standard approaches reported in literature for models requiring nonnegative results. Our numerical technique is a linear discretization of the system of PDEs under consideration, and it may be conveniently represented through a matrix that, under suitable conditions, turns out to be an M-matrix. To be more descriptive, this M-matrix is a weakly chained diagonally dominant matrix with nonpositive off-diagonal entries and positive diagonal components. Such matrices are nonsingular provided they are square and obey the properties mentioned throughout, and the entries of their inverses are all positive numbers. This characteristic of M-matrices is employed in order to guarantee the existence and uniqueness of nonnegative solutions of our numerical method, given non-negative initial profiles for both homogeneous discrete Dirichlet and Neumann conditions at the boundary. We provided some simulations on the growth and decay of biological films, taking nonnegative initial conditions bounded from above by 1. The reason for the bound by one is not only to prevent blowup mathematically of the nonlinear portion of the model, but also a one of normalization where we are permitted to divide the maximum obtained concentration by the overall amount. This is a standard in the literature when observing concentrations of biomass.

The MATLAB® environment has been chosen in this study to avoid the need for explicitly coded matrix methods that tend to be slower than MATLAB®’s internal use of the Intel® Math Kernel Library and vectorization techniques. We demonstrated that convergence of even some inappropriately chosen linear solvers can be achieved by choice of an appropriate preconditioning matrix, but in the effort to speed up calculations the
stabilized biconjugate gradient method was chosen without significant loss in accuracy over time. The results of our experiments indicate that our finite difference method preserves the nonnegative character of the initial profile (as evinced by our analytical and numerical results), and the bounded character of the solutions. Moreover, the method is guaranteed to be stable and convergent provided conditions on the parameters and size of the time step are met.

Our results for $E_1$ permitted us to look at a two state simple model in two spatial dimensions to provide useful information on modeling parameters. Example 1.1 was used to give an example of how the model would even function at the point where physically unreasonable situations were indeed permitted. Although nonphysical, it allowed for insight into the robustness of the model design. Example 1.2 then assumed the traditional behavior of a substrate diminishing with a depletion factor which ultimately increased the biomass and then diffused outward to consume the remaining biomass. Example 1.3 used the end result of Example 1.2 as an initial condition, but increasing the decay factor of the biomass, and finally caused it to slowly fall off to zero as the substrate was absorbed. The methods in this experiment provided insight into retaining memory for larger systems of state variables, a sound organizing of the system in matrices to allow for suppositions to be met for mixed boundary conditions while still adhering to the M-matrix requirements, and an understanding of the dynamics of a system that could display more interactive behavior.

Results for $E_2$ were made for a complete and descriptive picture of the simple dynamics of the four state model, where three separate components of the biomass were at play. As computational experimental time was vastly shortened, this model allowed for an exploration of parameter state space to make model parameters more realistic and pedantic. Although both Neumann and Dirichlet boundaries were used, the notion of Dirichlet conditions posited the previous hypothesis that they would be physically unrealizable, and the figures certainly display this discontinuity in comparison to the flow constriction of Neumann boundaries.
Example 2.2 gave wealthy insight into changing the dynamics of the model by increasing the overall coverage of EPS while still maintaining the same source of nutrients, but and increase in uptake and decrease in immediate diffusion provided a more realistic model of how biofilms could grow massively in localized regions but fall off significantly while slowly drifting to more substrate saturated localities.

Results for $\mathcal{E}_3$ required a small change to the region of influence of the active biomass, and therefore an attempt was made to sequester the Gaussian regimes to circular regions of concentration apart from other adjacent spots of biomass. This was partly to maintain the local properties of an initial deposit of biomass, but also to test how the particles would flow and interact. Our first trial with Example 3.1 was created to show how independent spots of active biomass evolved slowly to take over the substrate located everywhere homogeneously, but only slowly diffused together based on the low diffusion factor and small yield. Our second experiment in Example 3.2 gives a portrayal of a condition where diffusion takes over because the growth to yield ratio is balanced, and the active biomass slowly diffuses to regions where substrate is large and in specific regions. The inert biomass grows in the wake much as it did in $\mathcal{E}_2$, while the EPS mains intact.

The experiments in this dissertation give an affirmative outlook on the veracity of finite difference models that are required to maintain positive bounded solutions in finite difference equations regarding the hydrostatic interactions of biofilms. We have provided here a structure that not only supports the in vivo behavior of the growth dynamics of biofilms, but an insight into the method of forming M-matrices for the numerical solutions. The method initially required that strict diagonal dominance be held, but this condition can be in fact weakened by the weak chain diagonal dominance properties described. It is also the purpose of this dissertation to affirm that Neumann conditions can in fact be held on the boundaries, and the physical interpretation of such a condition is much more reasonable than one of Dirichlet. The analytic determinations of closed form solutions to the equations
studied are open problems that deserve further investigation, and our work has shown a powerful method for comparing results numerically. Our results indicate many possible future applications of our methodology to practical, scientific problems on the growth of biological films, and other systems requiring positive and bounded solutions.
Bibliography


8 Appendix

1 Matlab Code

```

% dissertation1.m
% Biofilm Experiment 1

warning off; %#ok<*WNOFF>
clc;
clear all;

% begin recording time of total program
tic;

% save and close images or animations

% end time
T = 0.02;

% evaluate using predefined parameters and initial conditions
p = 2;

% display total amount of memory used
checkmemory = 0;

% flag to create and save animation and not save images
animate = 0;

% flag to create a window showing estimated time left
waitbarflag = 1;

% loop through two conditions only changing one setting
for loop = 1:2

  % establish initial spatial and temporal resolution
  dx = 0.05;
dy = 0.06;
dt = 0.01;
```

108
% this performs loop over Neumann and Dirichlet conditions
if(loop == 1)
    Neu = 0;
else
    Neu = 1;
end

% mesh grid settings, number of node points
Rx = dt/dx/dx;
Ry = dt/dy/dy;
x = 0:dx:1;
y = 0:dy:1;
M = length(x)-1;
N = length(y)-1;
[XX, YY] = meshgrid(x, y);

% predefined settings for parameters
switch p
    case 1
        % set 1
        K1 = 0;
        K2 = 0;
        K3 = 0.4;
        K4 = 0;
        alpha = 4;
        beta = 4;
        d1 = 0;
        d2 = 0.0001;
        timevec = [0, 2.5, 5, 7.5, 10];

    case 2
        % set 2:
        K1 = 0.85;
        K2 = 0.0012;
        K3 = 0.4;
        K4 = 0.3;
        alpha = 4;
        beta = 4;
        d1 = 0.002;
        d2 = 0.0001;
        timevec = [0, 5, 10, 20, 30, 50];

    case 3

end
% set 3:
K1 = 0.65;
K2 = 0.36;
K3 = 0.2;
K4 = 0.3;
alpha = 4;
beta = 4;
d1 = 0.0015;
d2 = 0.0001;
timevec = [0, 1, 2.5, 5, 10, 30];

% perform test to ensure strict diagonal dominance inside boundary
while(dt*(K3-K2) ≥ 1)
    dt = dt/2;
    Rx = dt/dx/dx;
    Ry = dt/dy/dy;
end

% active biomass initial conditions
Li = 5;
C = [0.025, 0.03, 0.035, 0.02, 0.025];
r = [25, 50, 125, 100, 50];
xy = [0.25, 0.3; 0.5, 0.25; 0.7, 0.65; 0.4, 0.8; 0.5, 0.55];
INITX = initprofile(XX, YY, Li, C, r, xy)';

% default initialization starting at t = 0
INITS = ones(size(XX))';
S = INITS';
X = INITX';

St = S';
Xt = X';

% establish initial boundary conditions
Bcond = ones(M+1, N+1);
Bcond(:, 1) = 0;
Bcond(:, N+1) = 0;
Bcond(1, :) = 0;
Bcond(M+1, :) = 0;

if(Neu == 0)
    % for Dirichlet we fix substrate to walls at 1
    S(:, 1) = 1;
    S(:, M+1) = 1;
    S(1, :) = 1;
end
S(N+1, 1) = 1;

% for Dirichlet we fix biomass to ground at 0
X = X.*Bcond';

else

% for Neumann we fix flow at boundaries to 0
S = S.*Bcond';
X = X.*Bcond';

end

% create a vector of all state variables
vold = [reshape(S, (M+1)*(N+1), 1); reshape(X, (M+1)*(N+1), 1)];
lv = length(vold);

% preallocate a size(N+1, N+1) zero matrix
Balloc = zeros(N+1);
BS = sparse(blkdiag(eye(N+1), zeros((M-1)*(N+1)), eye(N+1)));

% associate Neumann or Dirichlet boundary conditions outside of loop
for k = 1:N+1
    BS(k, N+1+k) = -Neu;
    BS((M+1)*(N+1)-k+1, (N+1)*(M+1)-k-N) = -Neu;
end

BS(1, 1) = 1+Neu;
BS(1, 2) = -Neu;
BS(N+1, N) = -Neu;
BS(N+1, N+1) = 1+Neu;
BS(M*(N+1) + 1, M*(N+1) + 1) = 1+Neu;
BS(M*(N+1) + 1, M*(N+1) + 2) = -Neu;
BS((M+1)*(N+1), (M+1)*(N+1)) = 1+Neu;

BX = BS;

% commands for animation
if animate == 1
    if(loop == 1)
        % set up the movie
        writerObj = VideoWriter('D.avi', 'Uncompressed AVI');
        writerObj.FrameRate = 1;
        myVideo.Quality = 100;
        open(writerObj);
    end

end
else

    writerObj = VideoWriter('N.avi', 'Uncompressed AVI');
    writerObj.FrameRate = 1;
    myVideo.Quality = 100;
    open(writerObj);

end

end

if(waitbarflag == 1)

    h = waitbar(0,'Please wait...');
    steps = length(0:dt:T);
    step = 1;

end

t = 0;

% main loop for iterations over time
while(t <= T)

    % enforce solution to be bounded by 1
    voldu = vold(lv/2+1:lv);

    while(any(1 + dt*(K2 - K3) - voldu <= 0) == 1)

        dt = dt/2;
        Rx = dt/dx/dx;
        Ry = dt/dy/dy;

    end

    S = St;
    X = Xt;

    % basic definitions for matrix assembly
    Phipx = -Rx*d1*X(2:M, 2:N);
    Phimx = -Rx*d1*X(2:M, 2:N);
    Phipy = -Ry*d1*X(2:M, 2:N);
    Phimy = -Ry*d1*X(2:M, 2:N);

    Psipx = -Rx*Dfunc((X(3:M+1, 2:N) + X(2:M, 2:N)/2), ...
                       alpha, beta, d2);
    Psimx = -Rx*Dfunc((X(1:M-1, 2:N) + X(2:M, 2:N)/2), ...
                       alpha, beta, d2);
    Psipy = -Ry*Dfunc((X(2:M, 3:N+1) + X(2:M, 2:N)/2), ...
                       alpha, beta, d2);
    Psimy = -Ry*Dfunc((X(2:M, 1:N-1) + X(2:M, 2:N)/2), ...
                       alpha, beta, d2);
alpha, beta, d2);

sPhi = 1 - Phimx - Phipx - Phimy - Phipy;
sPsi = 1 - Psimx - Psipx - Psimy - Psipy;
monod = dt./(K4 + S(2:M, 2:N));

PhiS = sPhi + K1*X(2:M, 2:N).*monod;
ChiX = sPsi + dt*K2 - K3*S(2:M, 2:N).*monod;

for m = 1:M-1

% create block matrices
BS((N+1)*m+1:(m+1)*(N+1), m*(N+1)-N:(m+2)*(N+1)) = ... 
[diag([0, Phimx(m, :) , 0]) ... 
eye(N+1)*tridiag([Phimy(m, :) , 0], ... 
[1, PhiS(m, :) , 1], [0, Phipy(m, :)]) ... 
diag([0, Phipx(m, :) , 0])];
BX((N+1)*m+1:(m+1)*(N+1), m*(N+1)-N:(m+2)*(N+1)) = ... 
[diag([0, Psimx(m, :) , 0]) ... 
eye(N+1)*tridiag([Psimy(m, :) , 0], ... 
[1, ChiX(m, :) , 1], [0, Psipy(m, :)]) ... 
diag([0, Psipx(m, :) , 0])];

% input boundary conditions
BS(m*(N+1) + 1, m*(N+1) + 2) = -Neu;
BX(m*(N+1) + 1, m*(N+1) + 2) = -Neu;
BS((m+1)*(N+1), m*(N+1) + N) = -Neu;
BX((m+1)*(N+1), m*(N+1) + N) = -Neu;
end

% create M-matrix
MM = sparse([BS zeros((M+1)*(N+1)); zeros((M+1)*(N+1)) BX]);

% call solver to find next state vnew
bcgptest1;

% revert back to matrices from vector form
S = reshape(vnew(1:lv/2), N+1, M+1)';
X = reshape(vnew(lv/2+1:lv+1:lv), N+1, M+1)';

% save results
St = S;
Xt = X;

% plot commands if animation is not used
if(animate ~ 1)
    if(Neu == 0 && any(abs(t - timevec) < 1e-8))
        ts = datetime('now','Format','yyyy-MM-dd''T''HHmmss');
    end
end
date = char(ts);
dird = 'C:\Users\Richard\Desktop\Dissertation\DE1';
file = [dird, date, '\', num2str(t), '\', num2str(p)];
save([file, '.mat']);
surfinset1;
export_fig(file,'-png','-jpg','-tiff');

elseif(Neu == 1 && any(abs(t - timevec) < 1e-8))
   ts = datetime('now','Format','yyyy-MM-dd''T''HHmmss');
dirn = 'C:\Users\Richard\Desktop\Dissertation\NE1';
file = [dirn, date, '\', num2str(t), '\', num2str(p)];
save([file, '.mat']);
surfinset1;
export_fig(file,'-png','-jpg','-tiff');
end

else
   close(writerObj);
end

% reset boundary conditions
if(Neu == 0)
   % for Dirichlet we fix substrate to walls
   S(:, 1) = 1;
   S(:, N+1) = 1;
   S(1, :) = 1;
   S(M+1, 1) = 1;
   % for Dirichlet we fix biomass to ground
   X = X.*Bcond;
else
   % for Neumann we fix zero flow at boundaries
   S = S.*Bcond;
   X = X.*Bcond;
end

vold = [reshape(S', (M+1)*(N+1), 1); reshape(X', (M+1)*(N+1), 1)];
BS = sparse(blkdiag(eye(N+1), zeros((M-1)*(N+1)), eye(N+1)));
% reassociate Neumann or Dirichlet boundary conditions
for k = 1:N+1
end

BS((M+1)*(N+1)-k+1, (N+1)*(M+1)-k-N) = -Neu;

end

BS(1, 1) = 1+Neu;
BS(1, 2) = -Neu;
BS(N+1, N) = -Neu;
BS(N+1, N+1) = 1+Neu;
BS(M*(N+1) + 1, M*(N+1) + 1) = 1+Neu;
BS(M*(N+1) + 1, M*(N+1) + 2) = -Neu;
BS((M+1)*(N+1), M*(N+1) + N) = -Neu;
BS((M+1)*(N+1), (M+1)*(N+1)) = 1+Neu;

BX = BS;

if(animate == 1)
    surfinsert;
    frame = getframe(gcf);
    writeVideo(writerObj, frame);
 % pause(2);
    closereq;
end

% for current memory used flag this
if checkmemory == 1
    memory;
end

if(waitbarflag == 1)
    step = step + 1;
    waitbar(step / steps, h);
end

end

t = t + dt;
end

if(waitbarflag == 1)
    close(h);
end

% give total execution time for entire test run
toc;
if(loop == 1)
    US = St;
    UX = Xt;
elseif(loop == 2)
    VS = St;
    VX = Xt;
end
end
if(animate ≠ 1)
    % compare the looped conditions
    comparecontour1(XX, YY, US, VS, UX, VX, M, N);
end
% dissertation2.m
% Biofilm Experiment 2

warning off; %#ok<*WNOFF>
clc;
clear all;

% begin recording time of total program
tic;

% end time
T = 10;

% evaluate using predefined parameters and initial conditions
p = 1;

% display total amount of memory used
checkmemory = 0;

% flag to create and save animation and not save images
animate = 1;

% flag to create a window showing estimated time left
waitbarflag = 0;

% loop through two conditions only changing one setting
for loop = 1:2

% establish initial spatial and temporal resolution
dx = 0.01;
dt = 0.01;

% this performs loop over Neumann and Dirichlet conditions
if(loop == 1)
    Neu = 0;
else
    Neu = 1;
end

% mesh grid settings, number of node points
Rx = dt/dx/dx;
x = 0:dx:1;
t = 0:dt:T;
M = length(x)-1;
% predefined settings for parameters
switch p

case 1

% set 1:
mu = 1;
yh = 1;
k1 = 0.01;
ks = 0.008;
ke = 0.03;
ki = 0.4;
ye = 0.03;
alpha = 4;
beta = 4;
d1 = 1;
d2 = 1;
timevec = [0, 0.25, 0.5, 0.75, 1, 1.5, 2, 5, 10];

case 2

% set 2:
mu = 10;
yh = 0.03;
k1 = 0.01;
ks = 0.2;
ke = 0.03;
ki = 0.04;
ye = 0.03;
alpha = 2;
beta = 2;
d1 = 1;
d2 = 1;
timevec = [0, 0.25, 0.5, 0.75, 1, 1.5, 2, 5, 10];

case 3

% set 3:
mu = 2;
yh = 0.35;
ye = 0.03;
ks = 0.5;
ki = 0.4;
ke = 0.03;
k1 = 0.0001;
alpha = 4;
beta = 4;
d1 = 1;
d2 = 1;
timevec = [0, 0.25, 0.5, 0.75, 1, 1.5, 2, 5, 10];
% simplify parameters
a1 = mu/yh;
a2 = kl*ks;
a3 = ke*ks;
a4 = (kl+ki)*ks;
a5 = ye*mu;

% perform test to ensure strict diagonal dominance inside boundary
while((dt*max([a2+a3, mu, a4, a5]) ≥ ks) || (dt*(mu + a5) ≥ ks))
    dt = dt/2;
    Rx = dt/dx/dx;
end

% substrate
Li = 1;
C = [0.8];
r = [10];
x0 = [0.0];
INITS = initprofile2(x, Li, C, r, x0);

% active biomass
Li = 2;
C = [0.15, 0.2];
r = [200, 150];
x0 = [0.3, 0.65];
INITX = initprofile2(x, Li, C, r, x0);

% inert biomass
Li = 2;
C = [0.02, 0.01];
r = [80, 60];
x0 = [0.0, 1.0];
INITI = initprofile2(x, Li, C, r, x0);

% extracellular polymeric matrix
%Li = 4;
%C = [0.3, 0.35, 0.4, 0.25];
%r = [200, 300, 600, 400];
%x0 = [0.2, 0.8, 0.3, 0.7];
INITE = 0.005*ones(size(x)); %initprofile2(x, Li, C, r, x0);

% default initialization starting at t = 0
S = INITS';
X = INITX';
I = INITI';
E = INITE';
St = S;
Xt = X;
It = I;
Et = E;

% test if initial mass is bounded by 1
if (X + I + E ≥ 1)
    break
end

% establish initial boundary conditions
S(1) = 0;
S(M+1) = 0;
X(1) = 0;
X(M+1) = 0;
I(1) = 0;
I(M+1) = 0;
E(1) = 0;
E(M+1) = 0;

% create a vector of all state variables
vold = [S' X' I' E'];

% preallocate a size(M+1, M+1) zero matrix
Balloc = sparse(zeros(M+1));
Z1 = Balloc;
Z2 = Balloc;
H = Balloc;
G1 = Balloc;
G2 = Balloc;

% commands for animation
if (animate == 1)

    % set up the movie
    if (loop == 1)
        writerObj = VideoWriter('out1.avi', 'Uncompressed AVI');
        writerObj.FrameRate = 1; % How many frames per second.
        myVideo.Quality = 100;
        open(writerObj);
    else
        writerObj = VideoWriter('out2.avi', 'Uncompressed AVI');
        writerObj.FrameRate = 1; % How many frames per second.
        myVideo.Quality = 100;
        open(writerObj);
    end

120
if(waitbarflag == 1)
    h = waitbar(0,'Please wait...');
    steps = length(0:dt:T);
    step = 1;
end

for t = 0:dt:T
    S = St;
    X = Xt;
    I = It;
    E = Et;

    % enforce solution to be bounded by 1
    while(1 - dt*max(a5/ks, kl + max(ke, ki)) <= vold)
        dt = dt/2;
        Rx = dt/dx/dx;
    end

    % basic definitions for matrix assembly
    Alphapx = -Rx*Dfunc((X(3:M+1) + X(2:M) + ...
                         I(3:M+1) + I(2:M) + E(3:M+1) + ...
                         E(2:M))/2, alpha, beta, d2);
    Alphamx = -Rx*Dfunc((X(1:M-1) + X(2:M) + ...
                         I(1:M-1) + I(2:M) + E(1:M-1) + ...
                         E(2:M))/2, alpha, beta, d2);

    salpha = 1 - Alphamx - Alphapx;
    monod = dt./(ks + S(2:M));

    BetaS = salpha + a1*X(2:M).*monod;
    BetaX = salpha + a4.*monod;
    BetaI = salpha;
    BetaE = salpha + a3.*monod;

    Zeta1 = -a2.*monod;
    Zeta2 = -a4.*monod;

    Eta = -a3.*monod;
    Gamma1 = -mu*X(2:M).*monod;
    Gamma2 = -a5*X(2:M).*monod;

    % create block matrices and input boundary conditions
\text{BS} = \text{tridiag}(\text{[Alphamx; -Neu]}, [1; \text{BetaS}; 1], [-\text{Neu}; \text{Alphapx}]);
\text{BX} = \text{tridiag}(\text{[Alphamx; -Neu]}, [1; \text{BetaX}; 1], [-\text{Neu}; \text{Alphapx}]);
\text{BI} = \text{tridiag}(\text{[Alphamx; -Neu]}, [1; \text{BetaI}; 1], [-\text{Neu}; \text{Alphapx}]);
\text{BE} = \text{tridiag}(\text{[Alphamx; -Neu]}, [1; \text{BetaE}; 1], [-\text{Neu}; \text{Alphapx}]);
\text{Z1} = \text{sparse}(\text{diag}(\text{[0; \text{Zeta1}; 0]}));
\text{Z2} = \text{sparse}(\text{diag}(\text{[0; \text{Zeta2}; 0]}));
\text{G1} = \text{sparse}(\text{diag}(\text{[0; \text{Gamma1}; 0]}));
\text{G2} = \text{sparse}(\text{diag}(\text{[0; \text{Gamma2}; 0]}));
\text{H} = \text{sparse}(\text{diag}(\text{[0; \text{Eta}; 0]}));
\%	ext{create M-matrix}
\text{MM} = \text{sparse}(\text{[BS Z1 Balloc H; ... \text{G1 BX Balloc Balloc; ... \text{G2 Balloc Balloc BE}]});
\% \text{call solver to find next state vnew}
\text{b cg test2;}
\text{lv = length(vnew);}
\%	ext{revert back to individual state variables from vector form}
\text{S = vnew(1:lv/4);}
\text{X = vnew(lv/4+1:1lv/2);}
\text{I = vnew(lv/2+1:3+1lv/4);}
\text{E = vnew(3+1lv/4+1:lv);}
\%	ext{save results for LHS}
\text{St = S;}
\text{Xt = X;}
\text{It = I;}
\text{Et = E;}
\%	ext{reset boundary conditions for RHS}
\text{S(1) = 0;}
\text{S(M+1) = 0;}
\text{X(1) = 0;}
\text{X(M+1) = 0;}
\text{I(1) = 0;}
\text{I(M+1) = 0;}
\text{E(1) = 0;}
\text{E(M+1) = 0;}
\text{vold = [reshape(S', M+1, 1); reshape(X', M+1, 1); ...}
\text{ reshape(I', M+1, 1); reshape(E', M+1, 1)]';}
\text{Z1 = Balloc;}
\text{Z2 = Balloc;}
\text{H = Balloc;}
\text{G1 = Balloc;}
\text{G2 = Balloc;}
if(animate == 1)
    surfinset2;
    frame = getframe(gcf);
    writeVideo(writerObj, frame);
    % pause(2);
closereq;
end

% for current memory used flag this
if checkmemory == 1
    memory;
end
if(waitbarflag == 1)
    step = step + 1;
    waitbar(step / steps, h);
end

% plot commands if animation is not used
if(animate != 1)
    if(Neu == 0 && any(abs(t - timevec) < 1e-8))
        ts = datetime('now','Format','yyyy-MM-dd''T''HHmmss');
        date = char(ts);
        dirn = 'DE2';
        file = [dirn, date, '_', num2str(t), '_', num2str(p)];
        save([file, '.mat']);
        surfinset2;
        export_fig(file,'-png','-jpg','-tiff');
    elseif(Neu == 1 && any(abs(t - timevec) < 1e-8))
        ts = datetime('now','Format','yyyy-MM-dd''T''HHmmss');
        date = char(ts);
        dirn = 'NE2';
        file = [dirn, date, '_', num2str(t), '_', num2str(p)];
        save([file, '.mat']);
        surfinset2;
        export_fig(file,'-png','-jpg','-tiff');
    end
end
end
end

if (animate == 1)
    close (writerObj);
end

if (waitbarflag == 1)
    close (h);
end
toc;

if (loop == 1)
    UX = Xt;
    US = St;
    UI = It;
    UE = Et;
    elseif (loop == 2)
    VX = Xt;
    VS = St;
    VI = It;
    VE = Et;
    end
end

if (animate ≠ 1)
    figure;
    comparecontour2 (x, UX, VX, US, VS, UI, VI, UE, VE)
end
% dissertation3.m
% Biofilm Experiment 3

warning off; %#ok<*WNOFF>
clc;
clear all;

% begin recording time of total program
tic;

% end time
T = 1;

% evaluate using predefined paramaters and initial conditions
p = 3;

% display total amount of memory used
checkmemory = 0;

% flag to create and save animation and not save images
animate = 0;

% flag to create a window showing estimated time left
waitbarflag = 1;

% loop through two conditions only changing one setting
for loop = 1:2

% establish intial spatial and temporal resolution
dx = 0.06;
dy = 0.05;
dt = 0.01;

% this performs loop over Neumann and Dirichlet conditions
if(loop == 1)
    Neu = 0;
else
    Neu = 1;
end

% mesh grid settings, number of node points
Rx = dt/dx/dx;
Ry = dt/dy/dy;
x = 0:dx:1;
y = 0:dy:1;
t = 0:dt:T;
M = length(x)-1;
N = length(y)-1;
[XX, YY] = meshgrid(x, y);

% predefined settings for parameters
switch p

case 1

    % set 1:
    mu = 1;
yh = 0.35;
kl = 0.01;
ke = 0.03;
ki = 0.4;
ye = 0.03;
alpha = 1;
beta = 6;
d1 = 0.0002;
d2 = 0.0001;
timevec = [0, 0.1, 0.2, 0.3, 0.5, 0.7, 0.9, 1.0, 1.5, 2, ...
          2.5, 3, 4, 5];

case 2

    % set 2:
    mu = 0.02;
yh = 0.07;
kl = 0.1;
ks = 0.2;
ke = 0.01;
ki = 2;
ye = 0.8;
alpha = 2;
beta = 2;
d1 = 0.0002;
d2 = 0.0001;
timevec = [0, 0.1, 0.2, 0.3, 0.5, 0.7, 0.9, 1.0];

case 3

    % set 3:
    mu = 1;
yh = 0.05;
ye = 0;
ks = 0.2;
ki = 2;
ke = 0;
kl = 0.1;
alpha = 2;
beta = 2;
d1 = 0.0002;
d2 = 0.0001;
timevec = [0, 0.1, 0.2, 0.3, 0.5, 0.7, 0.9, 1.0];
case 4
% set 4:
mu = 1;
yh = 0.35;
kl = 0.01;
ks = 0.8;
ke = 0.03;
ki = 0.4;
ye = 0.03;
alpha = 2;
beta = 2;
d1 = 1;
d2 = 1;
timevec = [0, 0.5, 1, 1.5, 2, 2.5];
case 5
% set 5:
mu = 0.02;
yh = 0.07;
kl = 0.1;
ks = 0.2;
ke = 0.01;
ki = 2;
ye = 0.8;
alpha = 2;
beta = 2;
d1 = 1;
d2 = 1;
timevec = [0, 0.25, 0.5, 1, 2, 3];
case 6
% set 6:
mu = 1;
yh = 0.05;
ye = 0;
ks = 0.2;
ki = 2;
ke = 0;
kl = 0.1;
alpha = 2;
beta = 2;
d1 = 0.0001;
d2 = 0.0001;
timevec = [0, 3, 6, 9, 12, 15];

end

% simplify parameters
a1 = mu/yh;
a2 = kl*ks;
a3 = ke*ks;
a4 = (kl+ki)*ks;
a5 = ye*mu;

% perform test to ensure strict diagonal dominance inside boundary
while((dt*max([a2+a3, mu, a4, a5]) ≥ ks) && (dt*(mu + a5) ≥ ks))
    dt = dt/2;
    Rx = dt/dx/dx;
    Ry = dt/dy/dy;
end

% active biomass
Li = 6;
C = [0.25, 0.325, 0.275, 0.3, 0.2, 0.225];
r = [100, 50, 30, 80, 90, 100];
xy = [0.25, 0.3; 0.5, 0.25; 0.7, 0.65; 0.4, 0.8; 0.5, 0.55; ... 0.8, 0.3];
INITX = initprofile(XX, YY, Li, C, r, xy)';

% inert biomass
Li = 5;
C = [0.025, 0.03, 0.035, 0.2, 0.025];
r = [25, 50, 125, 100, 50];
xy = [0.3, 0.3; 0.55, 0.25; 0.76, 0.65; 0.45, 0.8; 0.5, 0.55];
INITI = initprofile(XX, YY, Li, C, r, xy)';

% extracellular polymeric matrix
Li = 4;
C = [0.3, 0.35, 0.4, 0.25];
r = [200, 300, 600, 400];
xy = [0.2, 0.2; 0.8, 0.8; 0.3, 0.7; 0.7, 0.2];
INITE = initprofile(XX, YY, Li, C, r, xy)';

% default initialization starting at t = 0
if(p == 1 || p == 4)
    INITS = ones(size(XX))';
elseif(p == 2 || p == 5)
    INITS = 0.2*ones(size(XX))';
end
INITE = 0.4*ones(size(XX))';

elseif(p == 3 || p == 6)
    INITS = exp(-5*(XX.^2+YY.^2))';
    INITX = 0.1*INITX;
    INITI = zeros(size(XX))';
    INITE = zeros(size(XX))';
end

S = INITS';
X = INITX';
I = INITI';
E = INITE';

St = S';
Xt = X';
It = I';
Et = E';

% test if initial mass is bounded by 1
if(X + I + E ≥ 1)
    break
end

% establish initial boundary conditions
Bcond = ones(M+1, N+1);
Bcond(:, 1) = 0;
Bcond(:, N+1) = 0;
Bcond(1, :) = 0;
Bcond(M+1, :) = 0;

if(Neu == 0)
    % for Dirichlet we fix substrate to walls at 1
    S(:, 1) = 1;
    S(:, M+1) = 1;
    S(1, :) = 1;
    S(N+1, 1) = 1;
    % for Dirichlet we fix biomass to ground at 0
    X = X.*Bcond';
    I = I.*Bcond';
    E = E.*Bcond';
else

end
% for Neumann we fix flow at boundaries to 0
S = S .* Bcond';
X = X .* Bcond';
I = I .* Bcond';
E = E .* Bcond';

end

% create a vector of all state variables
vold = [reshape(S, (M+1)*(N+1), 1); reshape(X, (M+1)*(N+1), 1); ...
       reshape(I, (M+1)*(N+1), 1); reshape(E, (M+1)*(N+1), 1)];

% preallocate a size(N+1, N+1) zero matrix
Balloc = zeros(N+1);
Z1 = Balloc;
Z2 = Balloc;
H = Balloc;
G1 = Balloc;
G2 = Balloc;
BS = sparse(blkdiag(eye(N+1), zeros((M-1)*(N+1)), eye(N+1)));

% associate Neumann or Dirichlet boundary conditions outside of loop
for k = 1:N+1
   BS(k, N+1+k) = -Neu;
   BS((M+1)*(N+1)-k+1, (N+1)*(M+1)-k-N) = -Neu;
end

BS(1, 1) = 1+Neu;
BS(1, 2) = -Neu;
BS(N+1, N) = -Neu;
BS(N+1, N+1) = 1+Neu;
BS(M*(N+1) + 1, M*(N+1) + 1) = 1+Neu;
BS(M*(N+1) + 1, M*(N+1) + 2) = -Neu;
BS((M+1)*(N+1), M*(N+1) + N) = -Neu;
BS((M+1)*(N+1), (M+1)*(N+1)) = 1+Neu;

BX = BS;
BI = BS;
BE = BS;

% commands for animation
if animate == 1
   if(loop == 1)
       % set up the movie
       writerObj = VideoWriter('D.avi', 'Uncompressed AVI');
       writerObj.FrameRate = 1;
       myVideo.Quality = 100;
   end
open(writerObj);

else

    writerObj = VideoWriter('N.avi', 'Uncompressed AVI');
    writerObj.FrameRate = 1;
    myVideo.Quality = 100;
    open(writerObj);

end

end

if(waitbarflag == 1)

    h = waitbar(0,'Please wait...');
    steps = length(0:dt:T);
    step = 1;

end

for t = 0:dt:T

    % enforce solution to be bounded by 1
    while(1 - dt*max(a5/ks, kl + max(ke, ki)) ≤ vold)
        dt = dt/2;
        Rx = dt/dx/dx;
        Ry = dt/dy/dy;
    end

    S = St;
    X = Xt;
    I = It;
    E = Et;

    % basic definitions for matrix assembly
    Alphapx = -Rx*Dfunc((X(3:M+1, 2:N) + X(2:M, 2:N) + ... I(3:M+1, 2:N) + I(2:M, 2:N) + E(3:M+1, 2:N) + ... E(2:M, 2:N))/2, alpha, beta, d2);
    Alphamx = -Rx*Dfunc((X(1:M-1, 2:N) + X(2:M, 2:N) + ... I(1:M-1, 2:N) + I(2:M, 2:N) + E(1:M-1, 2:N) + ... E(2:M, 2:N))/2, alpha, beta, d2);
    Alphapy = -Ry*Dfunc((X(2:M, 3:N+1) + X(2:M, 2:N) + ... I(2:M, 3:N+1) + I(2:M, 2:N) + E(2:M, 3:N+1) + ... E(2:M, 2:N))/2, alpha, beta, d2);
    Alphamy = -Ry*Dfunc((X(2:M, 1:N-1) + X(2:M, 2:N) + ... I(2:M, 1:N-1) + I(2:M, 2:N) + E(2:M, 1:N-1) + ... E(2:M, 2:N))/2, alpha, beta, d2);

end

131
\[
\begin{align*}
\text{salpha} &= 1 - \text{Alphamx} - \text{Alphapx} - \text{Alphamy} - \text{Alphapy} ; \\
\text{monod} &= \text{dt./(ks + S(2:M, 2:N))} ; \\
\text{BetaS} &= \text{salpha} + a1*\text{X(2:M, 2:N)}.*\text{monod} ; \\
\text{BetaX} &= \text{salpha} + a4.*\text{monod} ; \\
\text{BetaI} &= \text{salpha} ; \\
\text{BetaE} &= \text{salpha} + a3.*\text{monod} ; \\
\text{Zeta1} &= -a2.*\text{monod} ; \\
\text{Eta} &= -a3.*\text{monod} ; \\
\text{Gamma1} &= -\mu*\text{X(2:M, 2:N)}.*\text{monod} ; \\
\text{Zeta2} &= -a4.*\text{monod} ; \\
\text{Gamma2} &= -a5*\text{X(2:M, 2:N)}.*\text{monod} ; \\
\end{align*}
\]

% create block matrices
% input boundary conditions

\textbf{for m = 1:M-1}

\texttt{
% create block matrices
BS((N+1)*m+1:(m+1)*(N+1), m*(N+1)-N:(m+2)*(N+1)) = ... \\
[diag([0, Alphamx(m, :) , 0]) ... \\
\phantom{BS((N+1)*m+1:(m+1)*(N+1), m*(N+1)-N:(m+2)*(N+1)) = ...} \\
\phantom{\texttt{for m = 1:M-1}} \\
\phantom{\texttt{\% create block matrices}} \\
\phantom{\texttt{\% input boundary conditions}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \\
\phantom{\texttt{\%}} \<}
BE((m+1)*(N+1), m*(N+1) + N) = -Neu;

end

Z1 = sparse(blkdiag(Z1, zeros(N+1)));  
Z2 = sparse(blkdiag(Z2, zeros(N+1)));  
G1 = sparse(blkdiag(G1, zeros(N+1)));  
G2 = sparse(blkdiag(G2, zeros(N+1)));  
H = sparse(blkdiag(H, zeros(N+1)));

% create M-matrix
MM = sparse([BS Z1 zeros((M+1)*(N+1)) H;...  
  G1 BX zeros((M+1)*(N+1)) zeros((M+1)*(N+1));...  
  zeros((M+1)*(N+1)) Z2 BI zeros((M+1)*(N+1));...  
  G2 zeros((M+1)*(N+1)) zeros((M+1)*(N+1)) BE]);

% call solver to find next state vnew
bcgstest3;
 lv = length(vnew);

% revert back to matrices from vector form
S = reshape(vnew(1:lv/4), N+1, M+1)';
X = reshape(vnew(lv/4+1:lv/2), N+1, M+1)';
I = reshape(vnew(lv/2+1:3*lv/4), N+1, M+1)';
E = reshape(vnew(3*lv/4+1:lv), N+1, M+1)';

% save results
St = S;
Xt = X;
It = I;
Et = E;

% reset boundary conditions
if(Neu == 0)
  % for Dirichlet we fix substrate to walls
  S(:, 1) = 1;
  S(:, N+1) = 1;
  S(1, :) = 1;
  S(M+1, 1) = 1;
  % for Dirichlet we fix biomass to ground
  X = X.*Bcond;
  I = I.*Bcond;
  E = E.*Bcond;
else
  % for Neumann we fix zero flow at boundaries
  S = S.*Bcond;
  X = X.*Bcond;
end
I = I.*Bcond;
E = E.*Bcond;
end

vold = [reshape(S', (M+1)*(N+1), 1); ... 
    reshape(X', (M+1)*(N+1), 1); ... 
    reshape(I', (M+1)*(N+1), 1); reshape(E', (M+1)*(N+1), 1)];

Z1 = Baloc;
Z2 = Baloc;
H = Baloc;
G1 = Baloc;
G2 = Baloc;
BS = sparse(blkdiag(eye(N+1), zeros((M-1)*(N+1)), eye(N+1)));

% associate Neumann or Dirichlet boundary conditions
for k = 1:N+1
    BS(k, N+1+k) = -Neu;
    BS((M+1)*(N+1)-k+1, (N+1)*(M+1)-k-N) = -Neu;
end

BS(1, 1) = 1+Neu;
BS(1, 2) = -Neu;
BS(N+1, N) = -Neu;
BS(N+1, N+1) = 1+Neu;
BS(M*(N+1) + 1, M*(N+1) + 1) = 1+Neu;
BS(M*(N+1) + 1, M*(N+1) + 2) = -Neu;
BS((M+1)*(N+1), M*(N+1) + N) = -Neu;
BS((M+1)*(N+1), (M+1)*(N+1)) = 1+Neu;

BX = BS;
BI = BS;
BE = BS;

if(animate == 1)
    surfinset3;
    frame = getframe(gcf);
    writeVideo(writerObj, frame);
% pause(2);
closereq;
end

% for current memory used flag this
if checkmemory == 1
    memory;
if(waitbarflag == 1)
    step = step + 1;
    waitbar(step / steps, h);
end

if(animate ≠ 1)
    if(Neu == 0 && any(abs(t - timevec) < 1e-8))
        ts = datetime('now','Format','yyyy-MM-dd''T''HHmmss');
        date = char(ts);
        dird = 'C:\Users\Richard\Desktop\Dissertation\DE3';
        file = [dird, date, '_', num2str(t), '_', num2str(p)];
        save([file, '.mat']);
        surfset3;
        export_fig(file,'-png','-jpg','-tiff');
    elseif(Neu == 1 && any(abs(t - timevec) < 1e-8))
        ts = datetime('now','Format','yyyy-MM-dd''T''HHmmss');
        date = char(ts);
        dirn = 'C:\Users\Richard\Desktop\Dissertation\NE3';
        file = [dirn, date, '_', num2str(t), '_', num2str(p)];
        save([file, '.mat']);
        surfset3;
        export_fig(file,'-png','-jpg','-tiff');
    end
else
    close(writerObj);
end

if(waitbarflag == 1)
    close(h);
end

testresidual;

% give total execution time for entire test run
toc;

if(loop == 1)
    UX = Xt;
    US = St;
    UI = It;
    UE = Et;
elseif(loop == 2)
    VX = Xt;
    VS = St;
    VI = It;
    VE = Et;
end
end

if(animate \neq 1)
    comparecontour3(XX, YY, UX, VX, US, VS, UI, VI, UE, VE)
end
9 Vita

The author began his career as an undergraduate student at Tulane School of Engineering and obtained his Bachelor of Science in biomedical engineering, with a second major in mathematics and third major in physics. His research there included working in quantum optics and teaching linear algebra labs. He then received his Master of Science in applied physics at University of New Orleans where his research was in computational neurophysiology. He moved to Chicago, Illinois after Hurricane Katrina and worked as a research physicist for L’Oreal Institute for Ethnic Hair and Skin Research. He moved back to New Orleans, Louisiana and began pursuing his Doctor of Philosophy in engineering and applied physics, where his research is in mathematical modeling. His favorite pastimes include solving current problems in theoretical and computational physics, mathematics, and engineering, reading books on modern physics and stochastic differential equations, drawing, running, and sleeping.