UNIVERSITY OF CALIFORNIA, SAN DIEGO

Geometric Control of Confined Active Flow

A Thesis submitted in partial satisfaction of the requirements for the degree

Master of Science

in

Engineering Sciences (Mechanical Engineering)

by

Piyush Prakash

Committee in charge:

Professor David Saintillan, Chair
Professor Juan Carlos del Alamo
Professor Frank Talke

2018

The Thesis of Piyush Prakash is approved and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

Chair

University of California, San Diego

2018

Dedication

I would like to dedicate this to my family, who have supported me from the very beginning.

Table of Contents

List of Figures

List of Tables

Acknowledgements

I would like to thank Professor Saintillan for guiding me through my time at UCSD as my faculty advisor. Without his patience and knowledge, I would not have succeeded as much in my time at UCSD.

I would also like to thank for Professor de Alamo and Professor Talke for agreeing to be on my thesis defense committee and teaching some of my most engaging classes at UCSD.

Additionally, I would like to thank Achal, Brato, and Roberto, my labmates, who supported me throughout the past two years.

Finally, I would like to thank Maxime for the discussions and ideas about the direction of my research.

ABSTRACT OF THE THESIS

Geometric Control of Confined Active Flow

by

by Piyush Prakash

Master of Science in Engineering Sciences (Mechanical Engineering)

University of California, San Diego, 2018

Professor David Saintillan, Chair

Active fluids, such as suspensions of swimming microorganisms, have been shown to spontaneously create flow. Controlling active flows could provide advantages for those designing microfluidic devices. Therefore, we looked into geometric control of active fluids confined in a ring. Primarily focusing on the effect on bacterial flow, we used a sawtooth perturbation to determine factors useful in guiding and controlling active flows. It was found that the amplitude, asymmetry, and size of the sawtooth affected the total bacterial flow

created by the geometry. Additionally, optimization methods were developed to determine a shape that creates maximum bacterial flux.

Introduction

Active fluids are an area of interest for fluid dynamists due to their potential for widespread use in microfluidic devices and other biomedical applications. In both theory and experiments, it has been shown that active fluids may spontaneously create large scale flow patterns with no external influence. The propensity for active fluids to spontaneously create organized flow patterns has been shown to exists in both bulk flow [3] and within confined containers [11]. Simulations and experiments have found a region of fluid properties where there is both laminar flow and net pumping of fluid [11,14].

This property has experimentally been shown to be a useful tool for microfluidic devices as a method to transfer particles or energy across a system [15,17]. However, while there has been research to show that active fluid flow can be used to do work, there has been less work in optimizing the flow through passive means.

Using the numerical model used in [11] optimization of container shape was performed in order to control the fluid motion. Primarily, we aimed to maximize the net pumping in an annulus, however slowing and changing the direction of the flow is also possible by manipulating the geometry of the annulus.

Chapter 1: Background

Bacterial Motion:

Active fluids are suspensions of microparticles that often begin to display emergent collective motion when the active particles are found in high enough concentration. For these fluids, the energy in the system (usually chemical energy) is enough to cause the particles to move, which causes hydrodynamic interactions between particles that eventually creates a self-sustaining flow. Suspensions of bacteria in particular have been studied in the past, but non-living matter such as colloidal rollers, kinesin motors, and other cellular extracts also display similar dynamics.

Microorganisms have two common methods of locomotion: beating flagella or undulation of a ciliated membrane. The two methods create movement in different ways. Undulating cilia essentially creates a slip velocity along the particle boundary and is described using a squirmer model [8]. Flagella-based movement instead can be described as a force dipole. The microorganisms can be further categorized as either a pusher or puller depending on the effect of the dipole on the external fluid [3].

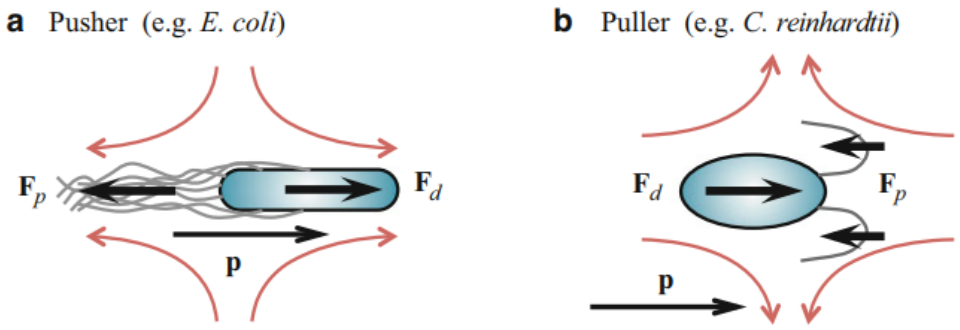

Figure 1: Diagram of swimming forces on pusher and puller microorganisms [3]

Pullers move by way of a stroke-like motion of their flagella, creating a force in the anterior (towards swimming direction) of the swimmer. In pullers, the propulsive force and the

drag force point towards each other, creating a contractile flow around the bacteria. Pushers move by beating their flagella, which creates a force in the posterior of the bacteria. As shown in the schematic diagram, in pushers the propulsive force and the drag force point away from each other creating an extensile fluid flow around the bacteria.

Flow caused by swimmers in an initially isotropic bacterial suspension relies on instabilities caused by the swimming of the bacteria. Pushers reliably create flows above a threshold concentration because the flow pattern of the swimming forces results in increasing perturbations that result in the large-scale flows seen in previous work [11].

In suspensions of pullers, flow is generally not found even in highly concentrated suspensions due to the stabilizing effect of the pullers' swimming forces. For example, it has been shown that suspensions of pullers increase the viscosity of the suspension [9]. The effect of the swimming forces can be seen in fig 1 [9], where the pullers' forces are working opposite the flow while the pushers are working in line with the flow. For this reason, only pushers are investigated in this research.

Analytical solutions

The swimming forces on a single bacterium result to leading order in a force dipole. Analysis of fluid flow around the particle shows that the fluid velocity around the microorganism scales as $\frac{1}{r^2}$ in terms of distance r from the particle center. A general form of the fluid velocity can be shown as follows:

$$\boldsymbol{u}^d(\boldsymbol{r}|\boldsymbol{p}) = \boldsymbol{S}(\boldsymbol{p}) : \nabla \boldsymbol{J}(\boldsymbol{r}) \quad (1)$$

J(r) is the fundamental solution of the Stokes equation, which is decays as $\frac{1}{r}$. The stresslet S(p) is a function of swimming orientation, p, and has magnitude σ_0 which characterizes the general flow of a bacterial suspension. A positive σ_0 describes flow around a puller while a negative σ_0 describes flow around a pusher [3].

While the previous equation details the effect of a singular bacterium, a suspension has many bacteria which must be taken into account to determine the overall movement of the suspension. The composition of the suspension can be described by a probability density function, Ψ, that describes the chance that a particle is at any given position and swimming direction for any time. In order to ensure conservation of particles within the system, Ψ must satisfy the Smoluchowski equation. Ψ can be approximated in terms of concentration, polarization, swimming orientation p, and nematic order, which are all defined in terms of position x, and time t [11].

$$\boldsymbol{\Psi}(\boldsymbol{x},\boldsymbol{p},t) \approx \frac{1}{2\pi}[c(\boldsymbol{x},t) + 2\boldsymbol{p}\cdot\boldsymbol{m}(\boldsymbol{x},t) + 4\boldsymbol{pp}:\boldsymbol{D}(\boldsymbol{x},t)] \textit{(2)}$$

The Smoluchowski equation is a special case of the Fokker-Planck equation for a probability density function of Brownian particles used to determine the positional and orientational distribution of the bacteria in a given system. The three contributions come from $\delta_t\Psi$, the time evolution of Ψ, and the divergence of Ψ in both the positions and orientational domains [4].

$$\delta_t\boldsymbol{\Psi} + \nabla_r\cdot(\dot{\boldsymbol{r}}\boldsymbol{\Psi}) + \nabla_p\cdot(\dot{\boldsymbol{p}}\boldsymbol{\Psi}) = 0 \textit{(3)}$$

The concentration, c, polarization, m, and nematic order, D, are found by taking the first three orientational moments of Ψ respectively. Polarization describes the average direction of the suspension, similar to the swimming direction. Nematic order describes the orderliness of particles within the suspension. For example, the concentration indicates the number of particles within any given cell, the polarization describes the direction the particles are pointing, and the nematic order describes the alignment of the particle in the suspension irrespective of their swimming direction.

These three quantities are enough to determine the state of the suspension. These parameters are defined by taking integrals of Ψ over the unit circle. From the definitions it is

clear that the concentration is the zeroth moment, polarization is the first moment, and nematic order is the second moment of Ψ:

$$c(\boldsymbol{x},t) = \int_c \boldsymbol{\Psi}(\boldsymbol{x},\boldsymbol{p},t)d\boldsymbol{p} \quad (4)$$

$$\boldsymbol{m}(\boldsymbol{x},t) = \int_c \boldsymbol{p}\boldsymbol{\Psi}(\boldsymbol{x},\boldsymbol{p},t)d\boldsymbol{p} \quad (5)$$

$$\boldsymbol{D}(\boldsymbol{x},t) = \int_c (\boldsymbol{pp} - \frac{\boldsymbol{I}}{2})\boldsymbol{\Psi}(\boldsymbol{x},\boldsymbol{p},t)d\boldsymbol{p} \quad (6)$$

Using these definitions, we are able to take the first three orientational moments of Ψ to determine the following relations for the flux of the concentration, polarization, and nematic order. E and W are the rate of strain and vorticity tensors in the fluid.

$$\delta_t \boldsymbol{c} = -\nabla \cdot \boldsymbol{F}_c \quad (7)$$

$$\delta_t \boldsymbol{m} = -\nabla \cdot \boldsymbol{F}_m + \frac{1}{2}\zeta \boldsymbol{E} \cdot \boldsymbol{m} - \boldsymbol{W} \cdot \boldsymbol{m} - \boldsymbol{m} \quad (8)$$

$$\delta_t \boldsymbol{D} = -\nabla \cdot \boldsymbol{F}_D + \zeta c\boldsymbol{E} + \boldsymbol{D} \cdot \boldsymbol{W} - \boldsymbol{W} \cdot \boldsymbol{D} - 4\boldsymbol{D} \quad (9)$$

Dimensional analysis on the equations was performed using the following parameters: ρ, Newtonian fluid density; μ shear viscosity; V_s, swimming velocity; d_r, rotational diffusivity; d_t, translational diffusivity; and H, characteristic length (in this case channel width). From the dimensional analysis, two rdimensional parameters relevant to this investigation were found: the swimming Peclet, $Pe_s = \frac{V_s}{d_r H}$, and the swimming propulsion parameter, $\Lambda = \frac{d_t d_r}{V_s^2}$. Pe_s is the ratio of the persistence length of the swimmer trajectory to the size of the domain and Λ is a ratio of the strength of the the diffusive processes to the strength of self-propulsion and is unique to each type of swimmer [11].

After nondimensionalizing and isolating the flux terms, the fluxes are shown as functions of concentration, polarization, and nematic order as shown in equations 10-12 [11].

From these equations, it is clear there are three contributions to the fluxes. The first term describes the advective contribution, the second describes the contribution from the swimming motion, and the third describes the contribution from translational diffusion.

$$\boldsymbol{F}_c = \boldsymbol{u}c + Pe_s\boldsymbol{m} - \Lambda Pe_s^2 \nabla c \text{ (10)}$$

$$F_m = \boldsymbol{u}\boldsymbol{m} + Pe_s\left(\boldsymbol{D} + c\frac{\boldsymbol{I}}{2}\right) - \Lambda Pe_s^2 \nabla \boldsymbol{m} \text{(11)}$$

$$\boldsymbol{F}_D = \boldsymbol{u}\boldsymbol{D} + Pe_s\left(\boldsymbol{T} - \boldsymbol{m}\frac{\boldsymbol{I}}{2}\right) - \Lambda Pe_s^2 \nabla \boldsymbol{D} \text{(12)}$$

T is the third orientational moment and is approximated as: $T_{ijk} = \frac{1}{4}\left(m_i\delta_{jk} + m_j\delta_{ik} + m_k\delta_{ij}\right)$, which allows closure of the equations.

Chapter 2: Methods

Bacterial Flow Solver

A hybrid finite-difference finite volume framework is used to solve the governing equations [11]. For each time step, the diffusive terms are solved implicitly, the advective terms are solved using a semi-Lagrangian method, and all coupling terms are solved explicitly. The polarization and nematic order are solved at the nodes, but the concentration is solved at the cell center in order to better preserve mass throughout the simulation. No flux boundary conditions are applied for all three orientational moments [11].

The initial state is set to an isotropic state with a uniform concentration and no moments or flow. The initial numerical errors create instabilities that propagate and ultimately drive larger flow patterns.

Level Set Method

Description

The level set method is used to define the boundary of the domain. In the level set method, the boundary of the domain is defined using an isocontour of a continuous function, ϕ, where the function value is equal to 0. Both the inner and outer boundary is defined by different functions, with the fluid domain corresponding to the positive values between the inner and outer boundaries [10].

A quadtree discretization structure is used to create the mesh for the numerical analysis. In the quadtree method, each cell is split into 4 equally sized smaller cells per level of refinement. The refinement starts at level 0 with a cell spanning the entire 2-dimensional domain. The refinement is increased based on the following criteria.

First, the maximum refinement is used near the domain boundary, with multiple layers of fully refined cells being used around the interface to ensure the best approximation of the

level set curves. This is primarily enforced by manipulating the Lipschitz constant, which "splits any cell whose edge length exceeds its distance to the interface" [10] as shown in equation 13.

$$\min_{v \in vertices(C)} |\phi(v)| \leq Lip(\phi) \cdot diag\_size(C) \textit{(13)}$$

Secondly, additional refinement is used when the cell bacterial concentration is above a defined critical value. The reasoning is that areas with higher concentrations of bacteria are likely areas with higher changes in local flow parameters and increasing refinement should be used to capture these gradients.

Finally, additional refinement is used when the vorticity is above a defined critical value. The refinement in this case allows for complex flow patterns to be captured accurately.

Application

In the optimization runs, we analyze flows inside an annular geometry between two boundaries defined by two separate functions. The inner boundary was defined to be a circle of radius equaling 1. The outer boundary was defined as a 3rd order Taylor series as shown in equation 14 where r is the distance from the center. With n = 3, there are 6 parameters that can be changed in order to create complex shapes. When A_n and B_n are equal to 0 for all n, the outer boundary is a circle with radius $\frac{A_0}{2}$.

$$r = f(\theta) = \frac{A_0}{2} + \sum_1^n (A_n \cos(n\theta) + B_n \sin(n\theta)) \textit{(14)}$$

ϕ was taken to be the maximum of the inner and outer boundary functions as shown in equation 15. The first term defines the inner boundary, and the second defines the outer boundary. Radius and θ are the coordinates of each point in polar coordinates. $f(\theta)$, as described in equation 14, is the definition of the Fourier series perturbation. The area

adjustment parameter, γ, will be explained in the following section. The curves for which ϕ equals 0 determines the location of the boundaries.

$$\phi = Max\left(-R_{min} + radius, -(R_{min} + 1) + \gamma * radius * f(\theta)\right) \text{ (15)}$$

This thesis documents attempts to maximize bacterial flow in enclosed systems through manipulation of the container's external boundaries. We assume that the concentration per unit area in the initial time step is constant across the entire domain, which means that for larger domains, more bacteria are present in the run. Therefore, the area of each test geometry is maintained to be that of an unperturbed ring in order to standardize the number of bacteria in each run to remove that factor in the flux magnitude optimization.

Area normalization is done through an iterative method where the external boundary is scaled until the domain areas match within tolerance. For each iteration, an area ratio is calculated as the square root of ratio between the standard area and the test geometry domain. If the area residual (difference between test and standard areas) is above the tolerance, then a new area adjustment parameter is created by dividing the previous iteration's area adjustment parameter by the area ratio. Then a new geometry is created by scaling the previous external boundary by the area adjustment parameter. This process is repeated until the residual is below the accepted area tolerance.

Optimization Methods

We considered three different optimization methods for the optimization of the geometry, which we describe here:

Steepest Descent Method

The Steepest descent method is a gradient optimization method which involves determining the gradient of the function to be optimized with respect to the parameters before making an optimization step against the gradient [13].

The algorithm begins with the calculation of the gradient. Equation 16 shows an example gradient for a 2-dimensional problem. As shown, N+1 function solves must be done in order to estimate a linear gradient [5].

$$\nabla f \cong [\frac{f(x_i+\Delta x)-f(x_i)}{\Delta x}, \frac{f(y_i+\Delta y)-f(y_i)}{\Delta y}] \quad (16)$$

Once the gradient is determined, the next parameter guess is given by subtracting the gradient from the current parameters. Equation 17 shows this step for x_{i+1}, with a similar equation for y_{i+1}.

$$x_{i+1} = x_i - \gamma \nabla f \quad (17)$$

γ controls the step size between each parameter estimation at each step of the method. Oftentimes γ can be made a function of the size of the gradient, providing smaller steps in quickly changing areas and larger steps in flatter terrain. This operation ensures the solution gradually moves towards better solutions while attempting to minimize any overstepping [5].

Nelder-Mead Simplex Method

The second optimization method pursued was a simplex method first proposed by J.A. Nelder and R. Mead in 1965 [12]. As a simplex method, the primary strategy is to use an N+1 pointed simplex to determine the direction of the geometric evolution in each optimization step.

The Nelder-Mead method requires N+1 points in order to optimize for N number of parameters, which creates a simplex in N dimensional space. Central to the method is the calculation of the centroid of a modified version of the simplex in which the worst performing point is omitted. Then a new N+1 dimensional simplex is created by reflecting, expanding, contracting the worst point across the centroid. In this way the simplex is able to walk across the domain and eventually shrink towards a local minimum.

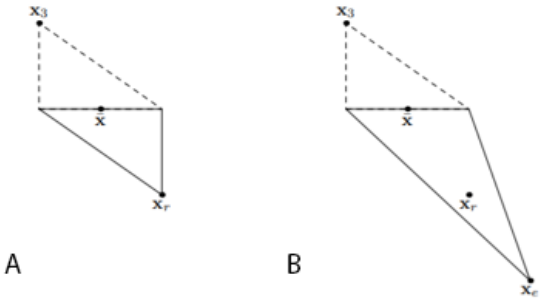

Figure 2: Reflection (A) and Expansion (B) of a simplex [18]

The initial step in every optimization step is reflection, where the worst performing node is reflected across the previously calculated centroid [12]. The new point is calculated by taking a weighted difference of the centroid and worst performing node. Essentially, this moves the node across the centroid in a straight line of some magnitude. The magnitude of the reflection is controlled by a coefficient of reflection, alpha, as shown in equation 18.

$$P^* = (1 + \alpha)\bar{P} - \alpha P_h \textit{(18)}$$

Expansion occurs if the reflected point performs better than any of the other points on the simplex. In this case, the new point is found by taking a weighted sum of the reflected point and the centroid. The magnitude of the expansion is controlled by a coefficient of expansion, γ, as shown in equation 19.

$$P^{**} = \gamma P^* + (1 - \gamma)\,\bar{P} \textit{(19)}$$

Contraction can occur in multiple cases in the Nelder-Mead method depending on the value of the reflected point and the worst performing node. In each contraction, the new node is calculated by taking a weighted difference between two nodes. The net result is the simplex volume is reduced [12].

$$P^{**} = \beta P_h + (1 - \beta)\,\bar{P} \textit{(20)}$$

- Inside Contraction

For the inside contraction, the simplex is contracted between the centroid and the worst performing node. The new node is then located on the line segment between the centroid and the worst performing node. The simplex is therefore shrunk along that axis. This procedure is used if the reflected node performed worse than the previous step's worst performing node.

- Outside contraction

For the outside contraction, the simplex is contracted between the centroid and the reflected node. The new node is then located on the line segment between the centroid and the reflected node. The simplex is then rotated and shrunk along that axis. This method is used if the reflected node performs better than the worst node but is still worse than the rest of the nodes.

- Shrinking

In the case in which none of the test nodes improve the performance of the simplex, then the entire simplex is shrunk. When shrinking, the two nodes used are the best performing node and every one of the other nodes. This can drastically cut the volume of the simplex and increases the resolution of the optimization around the best performing node.

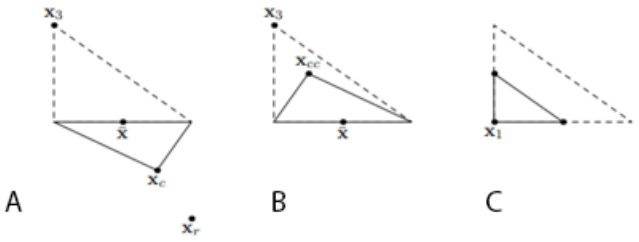

Figure 3: Simplex nodes after Outside Contraction (A), Inside Contraction (B), and Shrinking (C) [18].

Assessing the Minimum

There are two ways to determine that the method has converged. One is based on the simplex size, with the other based on the variance in function values across the simplex.

- Simplex based

Simplex based convergence criteria determine the minimum size of the simplex to determine convergence. At the minimum size, the differences between the function values at each point of the simplex is assumed to be a within an acceptable range such that the best performing node is the calculated minimum.

In a simplex, every node is connected to the other node which means that the size of the simplex can be quantified by looking at the length of each edge of the simplex.

- Function value Based

The function value-based convergence criteria determine the minimum variation between the function values at each node [12]. Once the variation reaches below the minimum threshold, the optimization terminates. Variation is defined by the standard deviation of the function values.

Shape Derivative Optimization

A level set based shape optimization method was also theoretically developed to optimize asymptotic bacterial flow [1]. We begin with the asymptotic flow moment equations,

$$0 = \nabla \cdot \boldsymbol{F}_c \text{(21)}$$

$$0 = \nabla \cdot \boldsymbol{F}_m + \boldsymbol{m} \text{(22)}$$

$$0 = \nabla \cdot \boldsymbol{F}_D + 4\boldsymbol{D} \text{(23)}$$

and their fluxes with a zero normal flux boundary condition.

$$\boldsymbol{F}_c = Pe_s\boldsymbol{m} - \Lambda Pe_s^2\nabla c \text{ (24)}$$

$$F_m = Pe_s\left(\boldsymbol{D} + c\frac{\boldsymbol{I}}{2}\right) - \Lambda Pe_s^2\nabla\boldsymbol{m}\text{(25)}$$

$$\boldsymbol{F}_D = Pe_s\left(\boldsymbol{T} - \boldsymbol{m}\frac{\boldsymbol{I}}{2}\right) - \Lambda Pe_s^2\nabla\boldsymbol{D}\text{(26)}$$

The equations can be made into matrix form by defining vector $\boldsymbol{M} = (c, \boldsymbol{m}, \boldsymbol{D})$ and $F(\boldsymbol{M}) = (\boldsymbol{F_c}, \boldsymbol{F}_m, \boldsymbol{F}_D)$.

$$\nabla \cdot \boldsymbol{F} + AM = 0\text{(27)}$$

A is defined as operator: $AM = (0, \boldsymbol{m}, 4\boldsymbol{D})$

The Flux Operator and boundary condition is therefore:

$$\boldsymbol{F}(M) = (\boldsymbol{F}_c, \boldsymbol{F}_m, \boldsymbol{F}_D) = \boldsymbol{BM} - \nabla\boldsymbol{M} \text{ (28)}$$

$$F(M)\cdot n = 0 \quad on\ \delta\Omega\text{(29)}$$

B is a matrix defined by the flux equations.

To derive the optimization, we follow the method used by Allaire [1]. We begin with an arbitrary cost function.

$$\boldsymbol{T}(\Omega) = \int_\Omega j(\boldsymbol{M}) \text{ (30)}$$

We define a Lagrangian operator L:

$$L(\Omega, \boldsymbol{N}, P) = \int_\Omega j(\boldsymbol{N}) - \int_\Omega(\nabla\cdot\boldsymbol{F}(\boldsymbol{N}) + AN)\cdot P + \int_{\delta\Omega}\boldsymbol{F}(\boldsymbol{N})\boldsymbol{n}\cdot P \text{ (31)}$$

For any direction ϕ at stationary point (M,Q), we find:

$$\left\langle\frac{\delta L}{\delta Q}(\Omega, M, Q)\middle|\phi\right\rangle = 0 = -\int_\Omega(\nabla\cdot\boldsymbol{F}(\boldsymbol{M}) + AM)\phi + \int_{\delta\Omega}\boldsymbol{F}(\boldsymbol{N})\boldsymbol{n}\cdot\phi \text{ (32)}$$

$$\left\langle \frac{\delta L}{\delta N}(\Omega, M, Q) \middle| \phi \right\rangle = 0 = \int_\Omega j'(M) \cdot \phi - \int_\Omega (\nabla \cdot \boldsymbol{F}(\phi) + A\phi) \cdot Q + \int_{\delta\Omega} \boldsymbol{F}(\phi)\boldsymbol{n} \cdot Q \text{ (33)}$$

Due to the diagonality of matrix A, equation 33 simplifies down to:

$$\left\langle \frac{\delta L}{\delta N}(\Omega, M, Q) \middle| \phi \right\rangle = 0 = \int_\Omega j'(M) \cdot \phi - \int_\Omega (\nabla \cdot \boldsymbol{F}(\phi)) \cdot Q) + AQ \cdot \phi + \int_{\delta\Omega} \boldsymbol{F}(\phi)\boldsymbol{n} \cdot Q \text{ (34)}$$

The boundary integral disappears after solving by parts, to leave:

$$\left\langle \frac{\delta L}{\delta N}(\Omega, M, Q) \middle| \phi \right\rangle = 0 = \int_\Omega j'(M) \cdot \phi - \int_\Omega (\nabla \cdot \boldsymbol{F}(\phi)) \cdot Q) + AQ \cdot \phi \text{(35)}$$

We may integrate once more by parts and substitute equation 28 for F to obtain:

$$\left\langle \frac{\delta L}{\delta N}(\Omega, M, Q) \middle| \phi \right\rangle = 0 = \int_\Omega j'(M) \cdot \phi - \int_\Omega (B^T \nabla Q + \nabla \cdot \nabla Q + AQ) \cdot \phi + \int_{\delta\Omega} \nabla \boldsymbol{Q}\boldsymbol{n} \cdot \phi \text{ (36)}$$

Taking ϕ with compact support provides an adjoint equation and a Neumann boundary condition:

$$\boldsymbol{B}^T \nabla \boldsymbol{Q} + \nabla \cdot \nabla \boldsymbol{Q} + \boldsymbol{AQ} = j'(\boldsymbol{M}) \text{(37)}$$

$$\nabla \boldsymbol{Q} \cdot \boldsymbol{n} = 0 \ \forall \, x \in \delta\Omega \text{(38)}$$

j'(M) must now be found. We begin by defining the cost function as the bacterial flux around the ring. The flux is dependent on the swimming velocity of the active particles. The swimming bacterial velocity is defined as the swimming Peclet number multiplied by the local polarization divided by the local concentration. The swimming velocity includes both the movement of the bacteria through self-propulsion.

$$v_{swimming} = v_{fluid} + \frac{Pe_s \cdot m}{c} \text{ (39)}$$

The net flux depends on the azimuthal component of the bacterial velocity since it is the component that allows the bacteria to flow around the ring. However, simply summing the

azimuthal velocities across the domain would be insufficient as the regions further from the center of the ring would contribute more due to the higher r value in those regions. Therefore, the flux is approximated as the sum of the azimuthal bacterial velocity normalized by the radius over the entire domain.

$$Flux = \int_{\Omega} \frac{v_\theta}{r} d\Omega \text{(40)}$$

In this case, the flux is the cost function of the optimization. Therefore, by comparing the cost function to the flux, we are able to see that in this case $j(M) = \frac{v_\theta}{r} = \frac{1}{r^2}(\frac{Pe_s \cdot m_x \cdot y}{c} + \frac{Pe_s \cdot m_y \cdot x}{c})$.

j'(M) can then be calculated by taking the $\frac{dj}{dM}$, giving:

$$j'(M) = \langle \frac{-Pe_s}{r^2 c^2}\left(m_x \cdot y + m_y \cdot x\right), \frac{Pe_s \cdot y}{r^2 c}, \frac{Pe_s \cdot x}{r^2 c}, 0,0,0,0 \rangle \text{.(41)}$$

Which may be used to find the adjoint.

The Shape Derivative can then be found from the relation: $\mathrm{T}(\Omega) = L(\Omega, M(\Omega), P(\Omega))$

After taking the derivative of T:

$$T'(\Omega) = \frac{\delta L}{\delta \Omega} + \frac{\delta L}{\delta N}\frac{\delta N}{\delta \Omega} + \frac{\delta L}{\delta P}\frac{\delta P}{\delta \Omega} \text{(42)}$$

from which $\frac{\delta L}{\delta n}$ and $\frac{\delta L}{\delta P}$ are 0 at the stationary point, giving:

$$T'(\Omega) = \frac{\delta L}{\delta \Omega}(\Omega, M, Q) \text{(43)}$$

From Allaire, we can combine with the previous equation to find the shape derivative direction equation:

$$\langle T'(\Omega)|\phi\rangle = \int_{\delta\Omega} \theta \cdot n(j(M) - (\nabla \cdot F(M) + AM) \cdot Q) + \int_{\delta\Omega} \theta \cdot n(\nabla(F(M)n \cdot Q)n + \kappa F(M)n \cdot Q)$$

(44)

Where the second term is equal to zero due to the boundary condition, giving a final equation. The method for the level set shape optimization can then be summarized as:

1. Compute M from solver

$$\boldsymbol{M} = (c, \boldsymbol{m}, \boldsymbol{D})$$

2. Compute j'(M)

$$j'(M) = \langle \frac{-Pe_s}{r^2 c^2} \left(m_x \cdot y + m_y \cdot x\right), \frac{Pe_s \cdot y}{r^2 c}, \frac{Pe_s \cdot x}{r^2 c}, 0,0,0,0 \rangle$$

3. Compute Adjoint

$$\mathbf{B}^{\mathrm{T}} \nabla \boldsymbol{Q} + \nabla \cdot \nabla \boldsymbol{Q} + \boldsymbol{A}\boldsymbol{Q} = j'(\boldsymbol{M})$$

$$\nabla \boldsymbol{Q} \cdot \boldsymbol{n} = 0 \ \ \forall \, x \ \in \delta\Omega$$

4. Compute Optimization direction

$$\langle \mathrm{T}'(\Omega) | \theta \rangle = \int_{\delta\Omega} \theta \cdot \boldsymbol{n} (j(M) + \ \nabla(\boldsymbol{F}(\boldsymbol{M})\boldsymbol{n} \cdot Q)\boldsymbol{n})$$

Chapter 3: Results

Optimization Validation

In order to test my optimization codes, I ran linear and nonlinear test cases. A sample set of data in shown in Table 1 using $f(x,y) = x^2 + y^2$ and $f(x,y) = \sin(x) + \sin(y)\cos(y)$ as the two test cases.

Table 1: Number of Optimization Steps until Convergence

	$x^2 + y^2$	$\sin(x) + \sin(y)\cos(y)$
Steepest Descent	30	68
Nelder-Mead	43	38

As the Table shows, the Nelder-Mead method converged faster than the steepest descent method in the tested nonlinear case. The faster convergence and lesser number of function solves per optimization step led to my decision to use the Nelder-Mead method in the Flux optimization tests.

Flow Control

Optimizing flow in a confined fluid requires the ability to control the flow. Without a definitive method to ensure the flow is in the required direction, active fluids would not be an effective tool in medical devices. Therefore, we begin by investigating the use of periodic perturbations in directing the bacterial flux.

For the first geometry series, we use a sawtooth perturbation along the exterior of the ring. Yariv and Schnitzer provided theoretical proof that sawtooth are able to direct flow along a linear unit cell [16]. In his theoretical work, increased skewness and amplitude of the

sawtooth correlated with higher bacterial flow in linear channels. We would expect similar results in a sawtooth perturbation on a ring.

We use Yariv's formulation to create the sawtooth exterior boundary as shown in equation 45 [16]. L and M determine the location and amplitude of the perturbation respectively; R is the radius of the ring at any given angle θ; and θ_0 defines the sawtooth period across the ring.

$$R = \begin{cases} \theta < L & 1 - M + 2 \cdot \frac{M\theta}{L} \\ \theta > L & 2 + M - 2 \cdot M \cdot \frac{\theta - L}{\theta_0 - L} \end{cases} \quad (45)$$

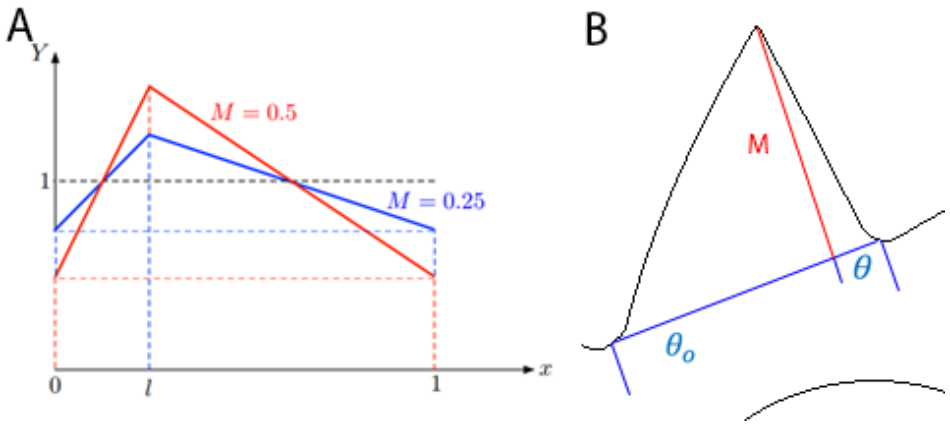

Figure 4: Yariv Defintion of sawtooth (A) and implementation as annulus outer boundary as defined in equation 45 (B)

Sawtooth Results - L Dependency

We began by testing the sensitivity of the active flow to changes in L. Changing L changes the position of max amplitude of the sawtooth shape, effectively changing its symmetry.

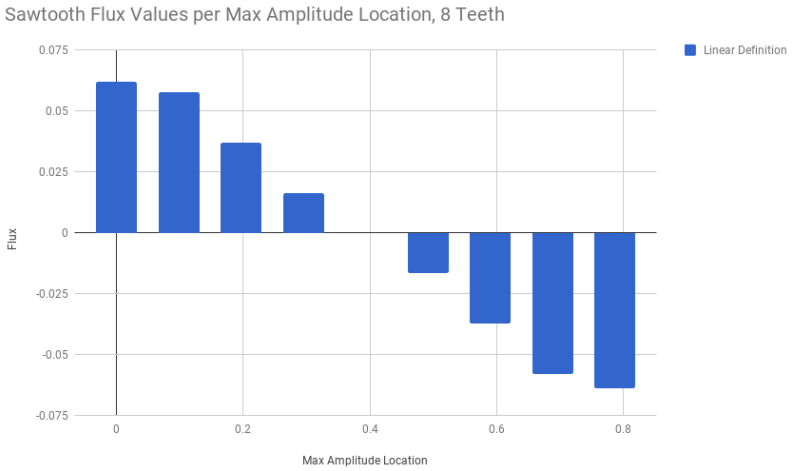

Figure 5: Flux results for 8 tooth sawtooth at M = 0.5 and varying L.

The direction of the flux changed depending on whether L was higher or lower than half the period of the sawtooth, $\frac{period}{2}$. Additionally, changing L increased the magnitude of the flux as the teeth became more asymmetric ($L - \frac{period}{2} \gg 0$). At $L = \frac{period}{2}$, the teeth were symmetric and the net flux was 0. This is consistent with expectations as without a break in symmetry, there is no geometric reason for the net flux to favor any direction. These trends were consistent for all tested numbers of teeth.

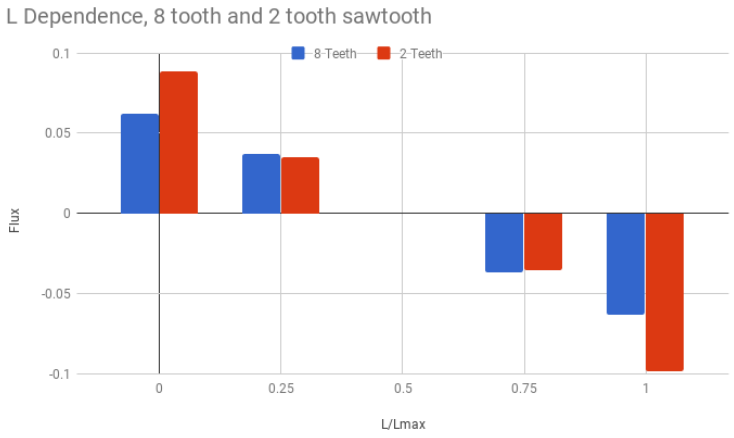

Figure 6: Flux results for varying L and number of teeth, M = 0.5.

Sawtooth Results - M Dependency

We then tested sensitivity of the flux to changes in M. Changing M changes the amplitude of each sawtooth. Essentially, this parameter controls the size of the sawtooth perturbation.

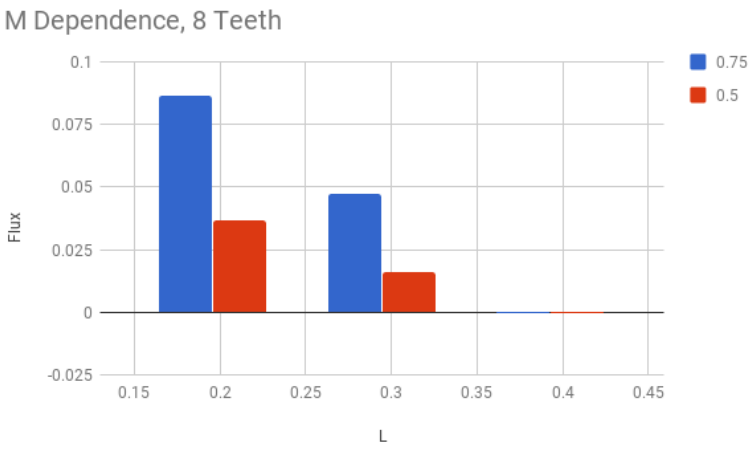

Figure 7: Flux Results for 8 tooth sawtooth at varying L and M.

M was found to be directly correlated with increased flux for every L position, except for the symmetric case where the flux was still 0.

Sawtooth Results - Period Dependency

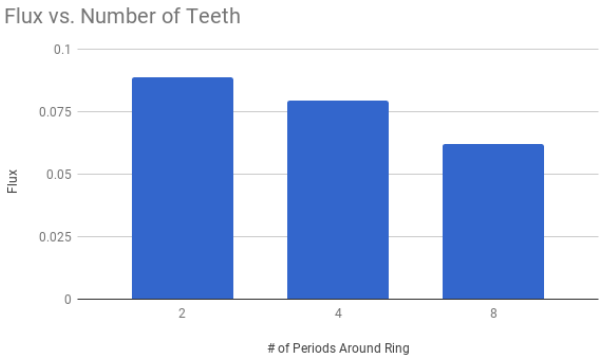

Figure 8: Flux Results for varying period size.

We finally tested the sensitivity of the period size on the net flux.

We found that for highly skewed teeth, longer periods improved the net flux, but the difference in magnitude of the net flux decreased as the teeth were made more symmetric.

The parameter sensitivities, when taken together, provide a general trend pointed to larger, highly skewed teeth providing higher net flux.

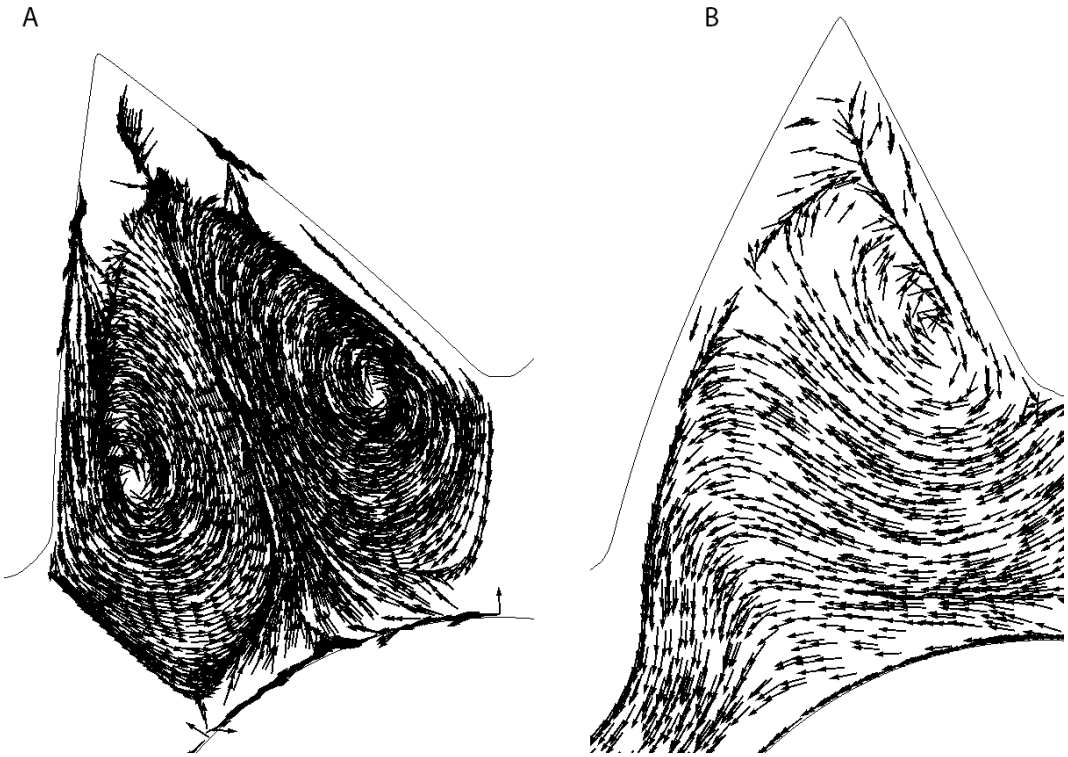

Figure 9: Example Sawtooth flow patterns: Symmetric (A) and Asymmetric (B)

The flow patterns suggest that the reason behind this trend is the creation of vortices by the sawtooth as shown in figure 9. Symmetric sawteeth have two counter rotating vortices of equal size and strength that nullify the overall net flow within the ring. However, as the teeth become more skewed, the size difference between the vortices changes, causing the net flux in the ring.

Fourier Definition & Results

In order to create a robust definition for the external boundary, I decided to generalize the level set definition of the external boundary as a Fourier series. With such a formulation, the optimization will go towards any arbitrary shape while also possibly closely approximating the verified trends determined by the sawtooth parameter sensitivity tests previously run.

$$f(\theta) = \frac{A_0}{2} + \sum_1^n (A_n \cos(n\theta) + B_n \sin(n\theta)) \quad (46)$$

Fourier Sawtooth Formulation

We tested the Fourier formulation by approximating the sawtooth formulation as a Fourier series. We solved the following equations to determine the correct Fourier coefficients:

$$A_0 = \frac{1}{\pi} \int_0^{2\pi} f(x)dx = \frac{1}{\pi} \left[\int_0^l \left(\frac{2Mx}{l} + 4 - M\right) dx + \int_l^{2\pi} \left(4 + M - 2M\frac{x-1}{2\pi - l}\right) dx \right. \quad (47)$$

$$A_n = \frac{1}{\pi} \int_0^{2\pi} f(x)\cos(nx)\,dx$$

$$= \frac{1}{\pi} \left[\int_0^l \left(\frac{2Mx}{l} + 4 - M\right) cos(nx)\,dx + \int_l^{2\pi} \left(4 + M - 2M\frac{x-1}{2\pi - l}\right) sin(nx)\,dx\right]$$

$$= \frac{1}{\pi} \left[\int_0^l \left(\frac{2Mx}{l} + 4 - M\right) cos(nx)\ dx + \int_l^{2\pi} \left(4 + M - 2M\frac{x-1}{2\pi - l}\right) cos(nx)\,dx \right] \quad (48)$$

$$B_n = \frac{1}{\pi} \int_0^{2\pi} f(x)\sin(nx)\,dx$$

$$= \frac{1}{\pi} \left[\int_0^l \left(\frac{2Mx}{l} + 4 - M\right) sin(nx)\,dx + \int_l^{2\pi} \left(4 + M - \left(2M\frac{x-1}{2\pi - l}\right)\right) sin(nx)\,dx\right] \quad (49)$$

The new formulation was checked for fit at different number for modes. We decided to check between the three mode and seven mode Fourier series to determine which number of modes we would use for a generalized optimization study.

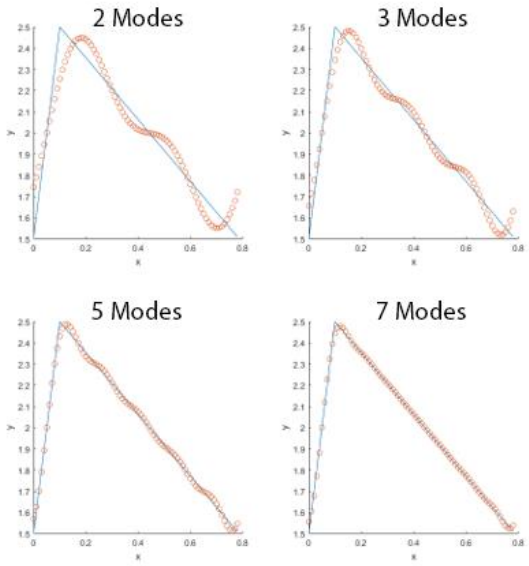

Figure 10: Fourier Series fit to the linear sawtooth formulation.

We swept through L values for an 8 teeth sawtooth boundary to confirm the Fourier defined sawtooth boundary resulted in similar flow patterns and flux values.

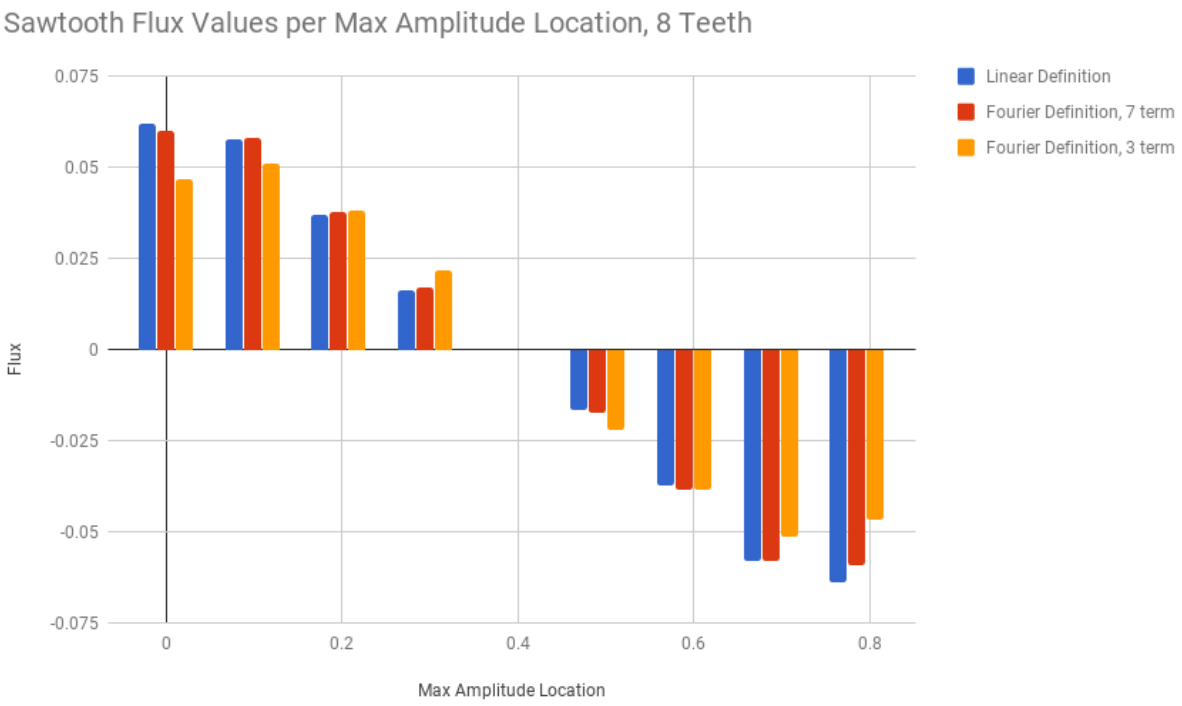

Figure 11: Flux comparison for 8 tooth sawtooth at M = 0.5 and varying L between linear and Fourier boundary formulations.

The Fourier series formulations were able to approximate the original formulation fairly well. The 7 mode Fourier Series matched the original formulation within 7%. The 3 mode Fourier series matched the general trends as the original but had higher deviation. We chose to use the 3 mode Fourier series in our optimization because it required less parameters to optimize, allowing us to optimize using the 6 coefficients instead of the 14 coefficients necessary to optimize the 7 mode Fourier series.

Numerical Optimization

Our initial optimization test ran the optimization with no constraints on the coefficients. However, the shapes created from the optimization quickly became unrealistic with the outer boundary intersecting the inner boundary and breaking the channel. The resulting optimized

shape before the broken geometries is shown in figure 12, which is reminiscent of a 2 teeth sawtooth.

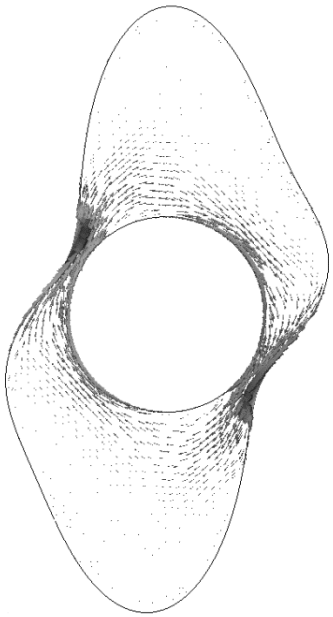

Figure 12: Vector plot of intermediate shape found during unbounded optimization.

In order to fix this issue, the Fourier Coefficients were bounded in order to prevent unfeasible shapes with sections that were too thin for realistic use. Bounding the coefficients made sure that the external boundary would never obstruct the channel by intersecting the inner boundary.

Bounded Optimization Case 1:

For this optimization case, we began the optimization from a ring defined using the first three frequencies in the Fourier series. In order to optimize only the coefficients that changed the shape of the perturbation, A_0 was set to be constant to maintain the outer radius value of 2 used in the base ring. Therefore, there were 6 parameters that were optimized from the perturbation function (equation 50) for this run.

$$f(\theta) = 2 + \sum_1^3 (A_n \cos(n\theta) + B_n \sin(n\theta)) \quad (50)$$

Using the simplex method, a converged shape was found within 15 maximum flow shapes as shown in figure 13. The net maximum flux was calculated to be 0.117, approximately 35% higher than the highest flux found in previous tests.

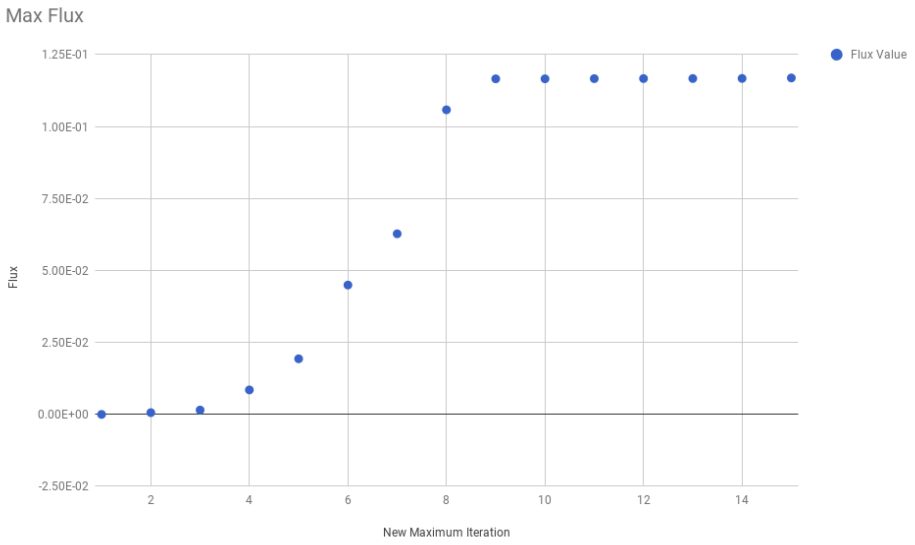

Figure 13: Convergence Plot for optimization case 1.

The ring shape evolved towards the shape shown in figure 14. The overall shape has two lobes of two different sizes. Each lobe contains two vortices that redirects the flow around the ring. The flow pattern fits the general trend of larger, longer “teeth” correlating with higher flux values.

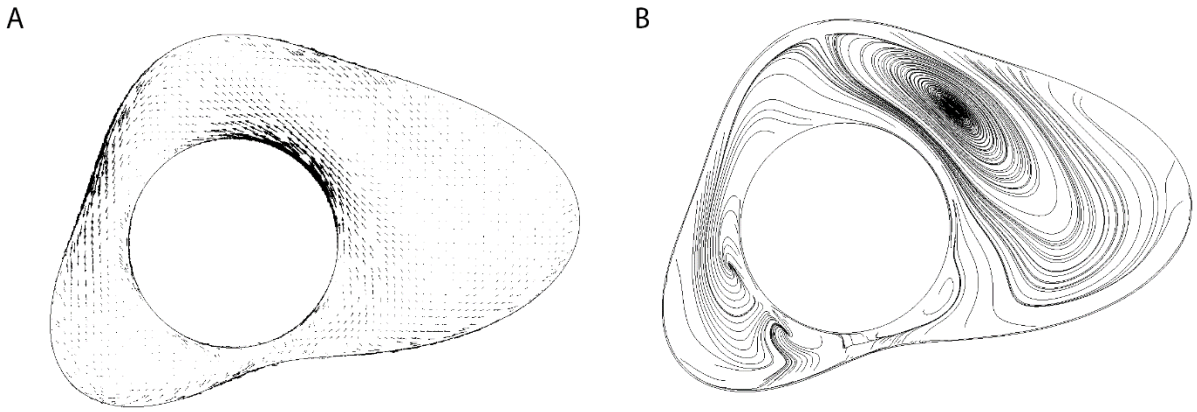

Figure 14: A) Vector Plot of converged solution of optimization case 1. B) Streamline Plot of converged solution of optimization case 1.

Bounded Optimization Case 2:

In order to see how a different set of frequencies changes what the optimal shape is, we then defined the external boundary with a Fourier series using the second through fourth frequencies. Once again, in order to optimize only the coefficients that changed the shape of the perturbation, A_0 was set to be constant to maintain the outer radius value of 2 used in the base ring. Therefore, there were 6 parameters that were optimized from the perturbation function (equation 51) for this run.

$$f(\theta) = 2 + \sum_2^4 (A_n \cos(n\theta) + B_n \sin(n\theta)) \quad (51)$$

The simplex method converged within 15 maximum flow shapes as shown in the convergence plot (figure 15). The net maximum flux was calculated to be 0.068, which did not improve upon the optimal shape from case 1, nor did it improve from the maximum found with the sawtooth boundary. This trend is consistent with the results from the sawtooth parameter sweeps where the longer period sawtooth shapes (equivalent to lower frequency) provided more bacterial flux than the shorter period sawtooth shapes.

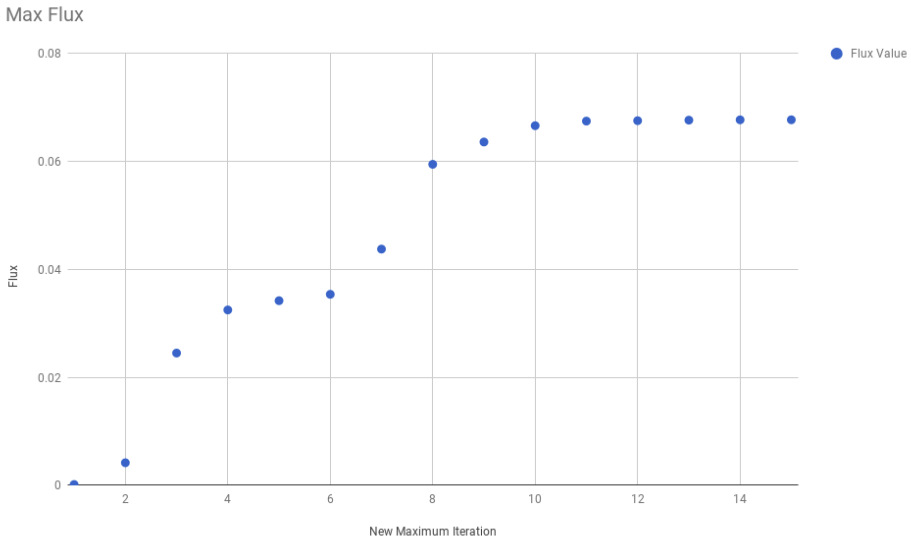

Figure 15: Convergence Plot for optimization case 2.

The ring shape evolved to the shape shown in figure 16. The overall shape has three lobes, with each lobe having at least one vortex creating a recirculating zone, similar to what was found in the sawtooth tests.

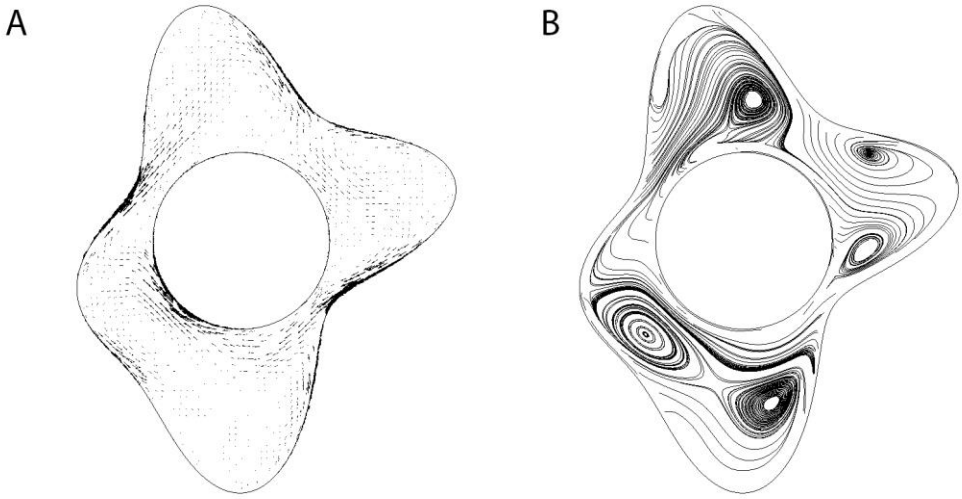

Figure 16: A) Vector Plot of converged solution of optimization case 2. B) Streamline Plot of converged solution of optimization case 2.

Chapter 4: Conclusion

Spontaneous flows have been found to occur in suspensions of active fluids. This phenomenon has the potential for use in medical devices, in particular microfluidic devices, where flow is required. A method to determine geometric methods to control and amplify active bacterial flow was created for this reason.

A 2-dimensional active fluid solver was used to determine the flow of various geometries containing active fluids. The level set method was used to define the fluid domain.

Multiple Optimization schemes were investigated. The steepest descent method relied upon computing gradients to find an optimal value at a zero-gradient point. The Nelder-Mead simplex method is a heuristic that searches for an optimal point through the geometric manipulation of a simplex defined by N test coordinates in a N-dimensional space. The final method investigated was a level set optimization method that relied on the determination of a shape derivative to evolve towards an optimal shape.

For the external boundary, the fluid response to a linear sawtooth geometry was investigated. It was found that the bacterial flux increased when increasing the period, the asymmetry, and the amplitude of the sawtooth shape. Additionally, a Fourier series definition of the external boundary was created. The Fourier series definition was shown to provide similar results as the linearly defined sawtooth.

The ring was optimized using the simplex method in order to determine an optimal shape to create optimal bacterial flux. A more effective shape was found by optimizing the coefficients of the first three modes of the Fourier series definition of the external boundary.

Throughout all the tested geometries, it was shown that larger, more skewed sawtooth-type geometries improved the bacterial flux of the fluid. The most effective shapes had similar flow patterns. Generally, the more effective shapes had vortices emerge from the flow that increased the movement of the bacteria around the ring.

References

[1] Allaire, Grégoire, François Jouve, and Anca-Maria Toader. "Structural optimization using sensitivity analysis and a level-set method." *Journal of Computational Physics* 194.1 (2004): 363-393.

[2] Saintillan, David, and Michael J. Shelley. "Active suspensions and their nonlinear models." *Comptes Rendus Physique* 14.6 (2013): 497-517.

[3] Saintillan, David, and Michael J. Shelley. "Theory of active suspensions." *Complex Fluids in Biological Systems*. Springer, New York, NY, 2015. 319-355.

[4] Saintillan, David, and Michael J. Shelley. "Instabilities and pattern formation in active particle suspensions: kinetic theory and continuum simulations." *Physical Review Letters* 100.17 (2008): 178103.

[5] Fletcher, Roger, and Michael JD Powell. "A rapidly convergent descent method for minimization." *The Computer Journal* 6.2 (1963): 163-168.

[6] Hohenegger, Christel, and Michael J. Shelley. "Stability of active suspensions." *Physical Review E* 81.4 (2010): 046311.

[7] Lambert, Guillaume, David Liao, and Robert H. Austin. "Collective escape of chemotactic swimmers through microscopic ratchets." *Physical Review Letters* 104.16 (2010): 168102.

[8] Lintuvuori, Juho S., Aloïs Würger, and K. Stratford. "Hydrodynamics defines the stable swimming direction of spherical squirmers in a nematic liquid crystal." *Physical Review Letters* 119.6 (2017): 068001.

[9] Marchetti, M. Cristina, Joanny, Jean-Francois, Ramaswamy, Sriram, Liverpool, Tanniemola, Prost, Jacques, Rao, Madan, Simha, R. Aditi. "Hydrodynamics of soft active matter." *Reviews of Modern Physics* 85.3 (2013): 1143.

[10] Min, Chohong, and Frédéric Gibou. "A second order accurate level set method on non-graded adaptive cartesian grids." *Journal of Computational Physics* 225.1 (2007): 300-321.

[11] Theillard, Maxime, Roberto Alonso-Matilla, and David Saintillan. "Geometric control of active collective motion." *Soft Matter* 13.2 (2017): 363-375.

[12] Nelder, John A., and Roger Mead. "A simplex method for function minimization." *The computer journal* 7.4 (1965): 308-313.

[13] Tong, Jiajun, and Michael J. Shelley. "Directed migration of microscale swimmers by an array of shaped obstacles: modeling and shape optimization." *arXiv preprint arXiv:1709.04449* (2017).

[14] Wu, Kun-Ta, Hishamunda, Jean Bernard, Chen, Daniel T.N., DeCamp, Stephen J., Chang, Ya-Wen, Fernandez-Nieves, Alberto, Fraden, Seth, Dogic, Zvonimir. "Transition from turbulent to coherent flows in confined three-dimensional active fluids." *Science* 355.6331 (2017).

[15] Yang, W., Misko, V.R., Nelissen, K., Kong, M., Peeters, F.M. "Using self-driven microswimmers for particle separation." *Soft Matter* 8.19 (2012): 5175-5179.

[16] Yariv, Ehud, and Ory Schnitzer. "Ratcheting of Brownian swimmers in periodically corrugated channels: A reduced Fokker-Planck approach." *Physical Review E* 90.3 (2014): 032115.

[17] Thampi, Sumesh P., Doostmohammadi, Amin, Shendruk, Tyler N., Golestanian, Ramin, Yeomans, Julia M. "Active micromachines: Microfluidics powered by mesoscale turbulence." *Science advances* 2.7 (2016): e1501854.

[18] Lagarias, Jeffrey C., Reeds, James A., Wright, Magaret H., Wright, Paul E. "Convergence properties of the Nelder--Mead simplex method in low dimensions." *SIAM Journal on optimization* 9.1 (1998): 112-147.