Marine Algae Stalk Ca^{2+} Regeneration Using Mimetic Differences

Mohsin A. Mohammed * and Miguel A. Dumett †‡

March 14, 2024

Abstract

In this study, we delve into the complex interplay of chemical reactions and diffusion processes that underpin whorl formation in marine algae, focusing specifically on the influence of external calcium concentrations. At the heart of our investigation is the application of the mimetic operators MOLE library, a sophisticated numerical tool designed for solving partial differential equations with high accuracy and efficiency. Through this computational lens, we elucidate the dynamic changes in marine algae whorl formation, shedding light on the critical role of calcium in mediating these processes. Our findings not only advance the understanding of marine algae's physiological responses to environmental stimuli but also demonstrate the utility of the MOLE library in tackling complex biochemical systems. This research contributes valuable insights into the reaction-diffusion mechanisms governing biological pattern formation, with potential implications for broader ecological and environmental studies

1 Introduction

Marine algae, like Acetabularia acetabulum, are fascinating single-celled organisms that can grow quite large and have complex shapes. Imagine a tiny underwater tree: it has a base that looks like roots, a long middle stalk, and a top part that fans out like an umbrella. Interestingly, this algae can also grow leaves that are about the size of a small cookie!

What's really cool about these algae is their ability to regenerate, or grow back parts that are removed. If you take off the top "umbrella," the algae can completely regrow it thanks to a special part in its base called the nucleus, where all its genetic information is stored. Even if you cut a piece from its stalk, it can grow a new top again!

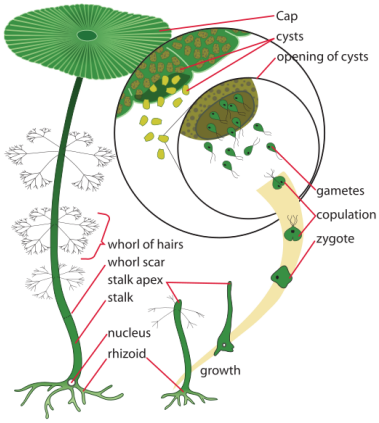


Figure 1: Algae Structure and it's body parts (taken from Wikipedia)

This study focuses on a particular aspect of the algae's growth: the formation of ring-like structures along its stalk, influenced by the amount of calcium in the water. These rings are important for understanding

*Computational Science Master's Program at San Diego State University (mmohammed5956@sdsu.edu).
†Editor: Jose E. Castillo
‡Computational Science Research Center at San Diego State University (mdumett@sdsu.edu).

how the algae grows into its unique shape, which is mainly due to chemical reactions inside the algae and its internal structure, rather than its genes. This simplified look into the algae's growth helps us learn more about how living things develop their forms in nature.

2 Related Work

Straightforward models with differential equations that excludes genetics and intracellular signaling are proposed. The pattern development is caused by the system's Turing instability (see [4]), applies Murray's model, which is a purely chemical model and reaction-diffusion equation involving two reactants. In order to demonstrate the kinematic nature of this chemical diffusion model, they made the following observations: temperature influences hair spacing, outside calcium concentration influences whorl hair spatial pattern, and whorl pattern is simultaneously initiated.

The effect of calcium concentration on the start of the whorl pattern is the main focus of this model. As mentioned in [3], whorl production requires a minimal amount of calcium concentration. In addition, whorls will stop developing once the external calcium concentrations reach certain upper bound. These facts are demonstrated in their model, and specifics of the shift from a stable stem to whorl production when calcium concentration crosses a key value are clearly estimated. More specifically, they obtain a specific interval of calcium concentration for the whorl pattern to occur, they have shown that all three kinds of transition type can occur though in the case of thin stem wall only continuous type and catastrophic type can happen [5]. They also show that the number of whorl hairs generated during transition can be quite irregular with underlying parameters. Different interesting dynamical behavior are described by the theory given in [5] and found numerically by [3]. Our purpose is to be able to replicate numerically all findings in [3] but utilizing mimetic differences and the Mimetic Operator Library Extended (MOLE) library developed at San Diego State University [1].

3 Model Equation

This is a model proposed by Murray [4], but also numerically investigated by [3], which is an adaptation of a simple two-species Turing mechanism. The equations are:

$$\begin{aligned} \frac{\partial A}{\partial t} &= D_A \Delta A + k_1 - k_2 A + k_3 A^2 B, \\ \frac{\partial B}{\partial t} &= D_B \Delta B + k_4 - k_3 A^2 B, \end{aligned}$$

where k_i, $i = 1, 2, 3, 4$, $D_A, D_B > 0$ and A and B are functions of r, θ, and t with the annulus domain defined by

$$R_i \leq r \leq R_), \qquad 0 \leq \theta \leq 2\pi, \qquad R_i = 1,\ R_0 = 1.5.$$

A and B define the density of two substances inside the annular growth region at the top mature Acetabularia.

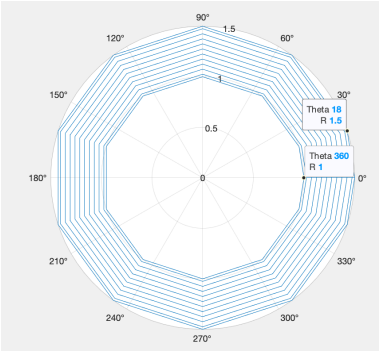


Figure 2: This is our model domain defined by R and θ in the blue color region

Here the assumption is that a reaction $2\,\mathrm{A} + \mathrm{B} \xrightarrow{\mathrm{k_3}} 3\,\mathrm{A}$ took place.

Basically, B is calcium, and A could be a molecule that is increased in the calcium presence but are constantly transformed to other substances at a rate of k_2. And they are generated by constant rate k_1, k_4, respectively, for A and B.

The model is non-dimensionalized via

$$u = A\left(\frac{k_3}{k_2}\right)^{1/2},\ v = B\left(\frac{k_3}{k_2}\right)^{1/2},\ t^* = \frac{D_A t}{R_i^2},\ x^* = \frac{x}{R_i},\ d = \frac{D_B}{D_A},\ a = \frac{k_1}{k_2}\left(\frac{k_3}{k_2}\right)^{1/2},\ \lambda = \frac{k_4}{k_2}\left(\frac{k_3}{k_2}\right)^{1/2},\ R^2 = \frac{R_i^2 k_2}{D_A}.$$

and becomes (where all stars are omitted to facilitate the formulas manipulation),

$$\begin{aligned}
\frac{\partial u}{\partial t} &= D\,u\,\Delta A + R^2(a - u - u^2 v) && \text{in } \Omega,\\
\frac{\partial v}{\partial t} &= d\,D\,v\,\Delta B + R^2(\lambda - u^2 v) && \text{in } \Omega,\\
u_r &= v_r = 0, && \text{on } \partial\Omega,
\end{aligned}$$

with

$$\Delta u = \frac{\partial^2 u}{\partial r^2} + r^{-1}\frac{\partial u}{\partial r} + r^{-2}\frac{\partial^2 u}{\partial \theta^2}$$

After substituting the Laplacian in polar coordinates, the partial differential equation system is given by,

$$\begin{aligned}
u_t &= R^2(a - u + v u^2) + \frac{\partial^2 u}{\partial r^2} + r^{-1}\frac{\partial u}{\partial r} + r^{-2}\frac{\partial^2 u}{\partial \theta^2}\\
v_t &= R^2(b - v u^2) + d\frac{\partial^2 v}{\partial r^2} + r^{-1}\frac{\partial v}{\partial r} + r^{-2}\frac{\partial^2 v}{\partial \theta^2}
\end{aligned}$$

4 Solving PDE

The MOLE library directly approximates the Cartesian Laplacian operator with single command but in our case, we have to slice the Laplacian in both r and θ because the second-order derivatives with respect to r and θ appear separately in polar coordinates and also because of the presence of r^{-2}, r^{-1} terms being multiplied. Since the matrix representation of the mimetic Laplacian operator is defined as the product of the matrix representations of both the divergence and the gradient operators, we need to exhibit the latter.

4.1 Divergence

The mimetic divergence is given by

$$\begin{aligned}
D &= \nabla \cdot \mathbf{F}\\
D &= \nabla \cdot \mathbf{F} = \frac{\partial F_r}{\partial r} + \frac{\partial F_\theta}{\partial \theta}\\
D &= \begin{bmatrix} | & | \\ D_r & D_\theta \\ | & | \end{bmatrix}
\end{aligned}$$

4.2 Gradient

The mimetic gradient is given by

$$\begin{aligned}
G &= \nabla F(r,\theta) = \frac{\partial F}{\partial r}\hat{i} + \frac{\partial F}{\partial \theta}\hat{j}\\
G &= \begin{bmatrix} — & G_r & — \\ — & G_\theta & — \end{bmatrix}
\end{aligned}$$

4.3 Laplacian

$$\begin{aligned}
L &= DG\\
L &= \nabla^2 = \vec{\nabla} \cdot \vec{\nabla}\\
L &= \nabla^2 = \frac{\partial^2}{\partial r^2} + \frac{\partial^2}{\partial \theta^2}
\end{aligned}$$

4.4 Initial Conditions

The initial conditions are given in terms of Bessel functions:

$$\begin{aligned} u_0 &= a+b+\epsilon(J_1(r)Y_1^{'}(1)-J_1^{'}(\delta)Y_1(\delta))\cos\theta, \\ v_0 &= \frac{b}{(a+b)^2}+\epsilon(J_1(r)Y_1^{'}(1)-J_1^{'}(\delta)Y_1(\delta))\cos\theta, \\ J_\alpha(x) &= \sum_{m=0}^{\infty}\frac{(-1)^m}{m!\,\Gamma(m+\alpha+1)}\left(\frac{x}{2}\right)^{2m+\alpha}, \\ Y_\alpha(x) &= \frac{J_\alpha(x)\cos(\alpha\pi)-J_{-\alpha}(x)}{\sin(\alpha\pi)}. \end{aligned}$$

4.5 Mimetic discrete boundary and boundary conditions

The mimetic discrete analog of the spatial derivatives of

$$\begin{aligned} u_t &= R^2(a-u+vu^2)+\frac{\partial^2 u}{\partial r^2}+r^{-1}\frac{\partial u}{\partial r}+r^{-2}\frac{\partial^2 u}{\partial\theta^2} \\ v_t &= R^2(b-vu^2)+d\,\frac{\partial^2 v}{\partial r^2}+r^{-1}\frac{\partial v}{\partial r}+r^{-2}\frac{\partial^2 v}{\partial\theta^2} \end{aligned}$$

utilizing the mentioned discrete analogs as well as convenient interpolation operators together with the time-derivative discretization for u_t produce

$$\begin{aligned} u_t &= R^2(diag(a)-u+vu^2)+D_rG_ru+r^{-1}I^GG_ru+r^{-2}D_\theta G_\theta u \\ \frac{u_{new}-u_{old}}{dt} &= R^2(diag(a)-u_{new}+diag(v_{old})diag(u_{old})u_{new}) \\ &+ D_rG_ru_{new}+diag(r^{-1})I^GG_ru_{new}+diag(r^{-2})D_\theta G_\theta u_{new} \\ u_{new}-u_{old} &= dtR^2diag(a)+dt(-R^2I+R^2diag(v_{old})diag(u_{old}) \\ &+ D_rG_r+diag(r^{-1})I^GG_r+diag(r^{-2})D_\theta G_\theta)u_{new} \end{aligned}$$

Let

$$X=-R^2I+R^2diag(v_{old})diag(u_{old})+D_rG_r+diag(r^{-1})I^GG_r+diag(r^{-2})D_\theta G_\theta$$

$$\begin{aligned} u_{new}-u_{old} &= dtR^2diag(a)+dtXu_{new} \\ (I-dtX)u_{new} &= dtR^2diag(a)+u_{old} \\ (I-dtX+BC)u_{new} &= dtR^2diag(a)+u_{old} \end{aligned}$$

Similarly for v_t,

$$\begin{aligned} v_t &= R^2(b-vu^2)+dD_rG_rv+r^{-1}I^GG_rv+r^{-2}D_\theta G_\theta v \\ \frac{v_{new}-v_{old}}{dt} &= R^2(diag(b)-diag(u_{new}^2))v_{new}+d\,D_rG_rv_{new}+diag(r^{-1})I^GG_rv_{new}+diag(r^{-2})D_\theta G_\theta v_{new} \\ v_{new}-v_{old} &= dt\,R^2diag(b)+dt(-R^2diag(u_{new}^2)+D_rG_r+diag(r^{-1})I^GG_r+diag(r^{-2})D_\theta G_\theta)v_{new} \end{aligned}$$

Let

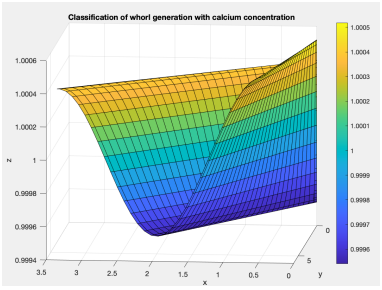
$$Y=-R^2diag(u_{new}^2)+D_rG_r+diag(r^{-1})I^GG_r+diag(r^{-2})D_\theta G_\theta$$

$$\begin{aligned} v_{new}-v_{old} &= dt\,R^2diag(b)+dt\,Yv_{new} \\ (I-dtY)v_{new} &= dt\,R^2diag(b)+v_{old} \\ (I-dtY+BC)v_{new} &= dt\,R^2diag(b)+v_{old} \end{aligned}$$

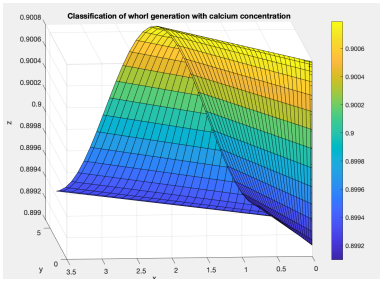
We are using the Neumann boundary conditions.

We apply the MATLAB inverse matrix command in the above functions. We are using interpolation because directly applying gradient will make the calculations at faces instead of centers but we need it to centers so one can apply the divergence later.

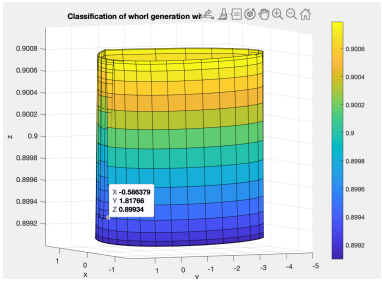
After factorising and applying boundary conditions, the code is ran a time loop from 0 to 2 with a time step of 1e-3.



(a) Cartesian plot of U



(b) Cartesian plot of V



(c) Polar plot using surf

5 Results

The first two panels of the figure on top, exhibit a snapshot of the evolution of concentrations of chemical species A and B (non-dimensionalized) in Cartesian coordinates. The last panel displays a plot in polar coordinates of them.

On the other hand, Figures 4

In addition, Figure 5 depicts, utilizing MATLAB command surf, the growing of the whorl hair.

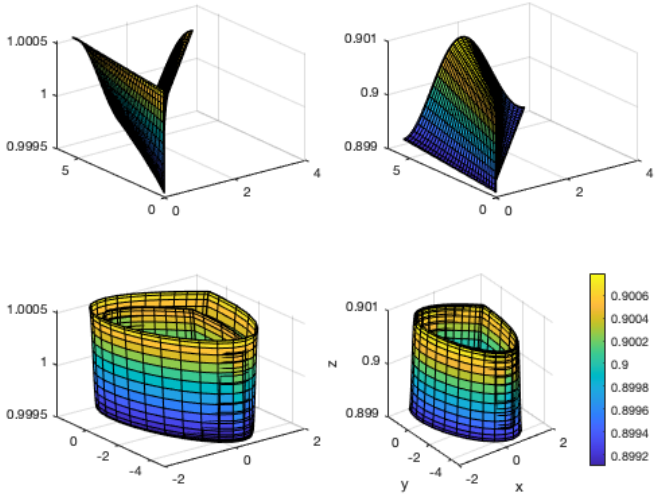


Figure 4: The different variables at time 2.

6 Conclusion

In this project a system of chemical reaction-diffusion equations is solved using the MOLE library. Analyzed the two reactants dynamic transitions with respect to the whorl formation of marine algae when it is caused by outside calcium concentration on its thin annulus.

We share our MATLAB code to facilitate the readers to reproduce our findings.

7 Code

```
clc
close all
clear

```

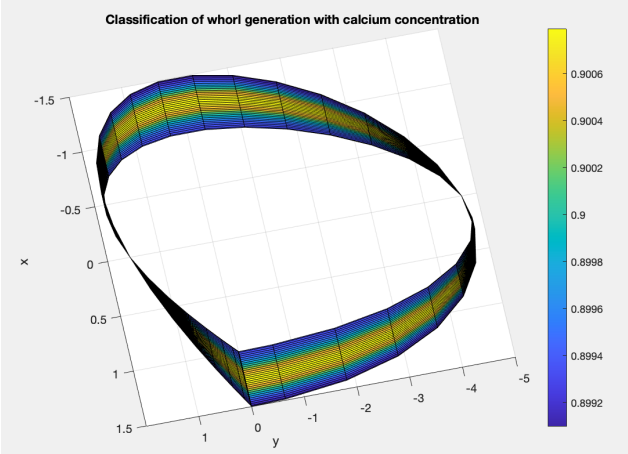


Figure 5: Polar plot using surf, yellow color regions depicts that whorl hair growth when reacted with outside calcium concentration

```
addpath('../mole_MATLAB')


%% Spatial discretization
k = 2;          % Order of accuracy (spatial)
m = 30;         % Number of cells in r direction
n = 30;         % Number of cells in theta direction
r1 = 1;
delta = 1.5;
r2 = delta;
theta1 = 0;
theta2 = 2*pi;
dr = (r2-r1)/m;  % Step length in r direction
dtheta = (theta2-theta1)/n; % Step length in theta direction


%% 2D Staggered grid
r_grid = [r1 r1+dr/2 : dr : r2-dr/2 r2];
theta_grid = [theta1 theta1+dtheta/2 : dtheta : theta2-dtheta/2 theta2];

[r_pts, theta_pts] = meshgrid(r_grid, theta_grid);
x = r_pts.*cos(theta_pts);
y = theta_pts.*sin(theta_pts);

[x1, y1] = pol2cart(r_pts, theta_pts);


%% Parameters
a = 0.1;
b = 0.9;
d = 9;
R = 3.45;
epsilon = 1e-2;

%% Simulation time
t_end = 0.5;
```

```
41 dt = 0.01;
42
43 %% Bessel functions calculation
44 Jn = besselj(1, r_grid);
45 Yn = besselj(2, r_grid);
46 J1 = besselj(1, r_pts);
47 Y1 = Yn(end);
48 DJn = diff(Jn) / dr;
49 DYn = diff(Yn) / dr;
50 DJ_delta = DJn(end);
51 DY1 = DYn(1);
52
53 %% IC
54 U0 = a + b + epsilon*((J1*DY1)-(DJ_delta*DY1)).*cos(theta_grid);
55 V0 = (b/(a+b)^2) + epsilon*((J1*DY1)-(DJ_delta*DY1)).*cos(theta_grid);
56
57 U_old = U0(:);
58 V_old = V0(:);
59
60 %% Equation discretization
61 D = div2D(k, m, dr, n, dtheta);
62 G = grad2D(k, m, dr, n, dtheta);
63
64 Dr = D(:,1:n*(m+1));
65 D_theta = D(:,n*(m+1)+1:end);
66
67 Gr = G(1:n*(m+1), :);
68 G_theta = G(n*(m+1)+1: end, :);
69
70 r_pts = r_pts(:);
71 theta_pts = theta_pts(:);
72
73 IG = interpGMat2D(k, m, n);
74 I_Gr = IG(:,1:n*(m+1));
75
76 %% BC
77 BC_term_eq = robinBC2D(k, m, dr, n, dtheta, 0, 1);
78
79 %% Time loop
80 for t = 0 : dt : t_end
81     X = R^2 * diag(V_old) * diag(U_old)                ...
82        + (Dr * Gr) + (diag(1./r_pts) * I_Gr * Gr)  ...
83        + (diag(1./r_pts.^2) * D_theta * G_theta)   ...
84        - R^2*diag(length(U_old));
85     X = eye(length(U_old), length(U_old)) - dt*X + BC_term_eq;
86     U_new = inv(X)*(dt*R^2*a*ones(length(U_old),1))+U_old;
87
88
89     Y = (d*(Dr*Gr) + diag(1./r_pts) * I_Gr * Gr)  ...
90         + (diag(1./r_pts.^2) * D_theta * G_theta)  ...
91         - R^2* diag(U_new) * diag(U_old);
92     Y = eye(length(V_old), length(V_old)) - dt * Y + BC_term_eq;
93     V_new = inv(Y)*(dt*R^2*b*ones(length(V_old),1)+V_old);
94
95     U_old = U_new;
```

```
96      V_old = V_new;
97
98      subplot(2,2,1);
99      surf(x1,y1,reshape(U_new, m+2, n+2));
100 %%
101     subplot(2,2,2);
102     surf(x1,y1,reshape(V_new, m+2, n+2));
103
104     subplot(2,2,3);
105     surf(x,y,reshape(U_new, m+2, n+2));
106     subplot(2,2,4);
107     surf(x,y,reshape(V_new, m+2, n+2));
108
109 %     title('Classification of whorl generation with calcium concentration');
110     xlabel('x')
111     ylabel('y')
112     zlabel('z')
113
114     colorbar
115     drawnow
116
117
118 end
```

References

[1] Corbino, J., and Castillo, J.E., MOLE: Mimetic Operators Library Enhanced: The Open-Source Library for Solving Partial Differential Equations using Mimetic Methods, 2017.

[2] Corbino, J., and Castillo, J.E. High-order mimetic finite-difference operators satisfying the extended Gauss divergence theorem. In: Journal of Computational and Applied Mathematics 364 (2020), p. 112326. issn: 0377-0427. doi: https://doi.org/10.1016/j.cam.2019.06.042. url:

https://www.sciencedirect.com/science/article/pii/S0377042719303231.8\item command.

[3] Mao, Y., Yan, D., and Lu, C.H., Dynamic transitions and stability for the acetabularia whorl formation. arXiv:1810.10120 [math.AP]

[4] Murray, J.D., Mathematical biology II: Spatial models and biomedical applications, 3rd edition, Springer-Verlag, New York, 2001.

[5] Ma, T., and Wang, S., Phase transition dynamics, Springer, 2014.