Modeling Networks of Evolving Populations

Sean Elliott

under the direction of

Dominic Skinner
Department of Mathematics
Massachusetts Institute of Technology
and the Research Science Institute

Abstract

The goal of this research is to devise a method of differential equation based modeling of evolution that can scale up to capture complex dynamics by enabling the inclusion of many—potentially thousands—of biological characteristics. Towards that goal, a mathematical model for evolution based on the well-established Fisher-Eigen process is built with a unique and efficient structure.

The Fisher-Eigen partial differential equation (PDE) describes the evolution of a probability density function representing the distribution of a population over a phenotype space. This equation depends on the choice of a fitness function representing the likelihood of reproductive success at each point in the phenotype space. The Fisher-Eigen model has been studied analytically for simple fitness functions, but in general no analytic solution is known. Furthermore, with traditional numerical methods, the model becomes exponentially complex to simulate as the dimensionality of the problem expands to include more phenotypes.

For this research, a network model is synthesized and a set of ordinary differential equations (ODEs) is extracted based on the Fisher-Eigen PDE to describe the dynamic behavior of the system. It is demonstrated that, when juxtaposed with full numerical PDE simulations, this ODE model finds well-matched transient and precise equilibrium solutions. This prototype method makes modeling of high-dimensional data possible, allowing researchers to examine and even predict complex dynamic behavior based on a snapshot of a population.

Such models have many important possible humanitarian uses. For instance, they could help researchers understand the dynamics of evolution of bacteria. Using the models, researchers could estimate how fast bacteria mutate from a harmless to a harmful state and also find which states are passed through along the way. With these insights, they could attempt to block the harmful mutations. Such mutations could enable antibiotic resistance, for example, which poses a growing, deadly, global threat.

1 Introduction and Background

The theory of evolution by natural selection has transformed modern biology. Understanding evolution allows one to understand the origin of the genetic diversity present in the world today. With this in mind, accurate mathematical and computer models that capture the vast complexity and intricate dynamics of evolution may give researchers new insights into how evolution has occurred and may predict evolutionary events in the future.

Such model-based tools would have many vital humanitarian uses. For instance, they could help researchers understand the dynamics of evolution of bacteria. Using the models, researchers could estimate how fast bacteria mutate from a harmless to a harmful state and also find which states are passed through along the way. With these insights, researchers could attempt to block the harmful mutation. This is important because such mutations could enable antibiotic resistance, for example, which poses a growing, deadly, global threat.

Some models of evolution have been developed in the past, for example in [1, 2]. The model presented here builds on previous work but with the goal of enabling the inclusion of far more—potentially thousands more—biological characteristics. We model evolution using the Fisher-Eigen strategy, also known as a Darwinian strategy, based on the work of Fisher [3] and Eigen [2]. Consider an arbitrary list of genetic characteristics. While each characteristic takes on discrete values, we can approximate each one as a continuous variable as long as the total number of organisms is large, as explained in [4]. We define a probability density function that tells us the proportion of the population with a specific value of a genetic characteristic at a specific time. The Fisher-Eigen strategy models evolutionary adaptation to the environment through a differential equation that controls this probability density function.

We then consider a fitness function, which represents the likelihood of successful reproduction of an organism given a value of the genetic characteristic for that organism. We can think of this fitness function as the negative of a potential energy function; the population tends towards a local maximum, just as for a potential energy function an object tends towards a local minimum. As an example, suppose we had a population of finches whose beaks varied in length. We can imagine that certain lengths would be optimal for certain types of food, and that there would be local maxima centered around each such length.

With a fitness function, we can model the population given the initial condition of the population at time zero using the Fisher-Eigen equation as defined in [8, 9]. This equation includes a comparison of the local fitness to the average fitness at every point, making it a global selection process. It also includes a diffusion term that represents random changes in the population over time.

Dunkel et al. [8] contrasted the Fisher-Eigen strategy with the Smoluchowski process (a model for particles in Brownian motion) as a model for evolution. The Smoluchowski process, which comes from thermodynamics, is based on local interactions between particles and the potential, as opposed to the global selection used in the Fisher-Eigen strategy. They looked at the case where the fitness function is of the form $ax^2 - bx^4$ with $a, b > 0$. The potential associated with such a fitness function is called a symmetric double-well potential.

Instead of computing the solutions exactly, they estimated the transition rates between the two wells of the potential function using an eigenvalue analysis. They used a system of master equations, which models how a system of multiple states evolves. Each master equation describes the probability of being in a state through a differential equation involving transition rates with all of the states. In their study they had two states, each representing a well of the potential function. They had equal transition rates between the states due to the symmetry of the fitness function. By expanding the probability density function in terms of the eigenfunctions of the Hamiltonian operator $H = -D\nabla^2 + U$, they were able to estimate the transition rate in terms of the first two eigenvalues of H.

Dunkel et al. [7] analyzed these two strategies when the fitness function was quadratic and found an exact solution for each. For general fitness functions, however, the equation is unsolved analytically. Furthermore, simulating the equation quickly becomes computationally infeasible as the number of phenotypes grows. In practice, one would hope to model a population that varies in hundreds or even thousands of phenotypes, but this is impossible when considering a numerical simulation of the Fisher-Eigen equation.

We model a population that evolves according to the Fisher-Eigen process but using a system of ordinary differential equations (ODEs) over a network. The motivation for constructing such a model is to simplify the modeling of high-dimensional fitness landscapes and avoid needing to take data samples at different times. Pearce et al. [6] introduced such a technique in the case of a Smoluchowski process. Their work proceeds as follows. First, gather data on a population at a specific time. Then, project the data onto a lower dimensional space. In this new space, find clusters of points and associate them with nodes of a network. Consider these nodes as the minima of some function analogous to a potential energy. Then, analyze the dynamics of such a network. Finding the transition rates between states is enough to reconstruct the dynamics of the population in the original landscape. This allows one to calculate quantities such as the average time to move from one state to another.

We perform a similar analysis on networks which behave according to the Fisher-Eigen process. This allows one to model a population which evolves according to the Fisher-Eigen process but over a high-dimensional landscape. However, studying networks is more difficult for the Fisher-Eigen process than for the Smoluchowski process because of the global selection property of the Fisher-Eigen process. In the Smoluchowski process, each individual particle behaves as it would in equilibrium because the selection process is only based on local behavior. In contrast, for the Fisher-Eigen process an individual organism behaves differently depending on whether or not the entire population is in an equilibrium state. However, if we measure the population at equilibrium, we can determine the fitness landscape, which we can then use to study the dynamics of the population, provided we can analyze the dynamics of an arbitrary fitness landscape.

In the model presented here, the domain is split into 3 regions. The integrals over each region are modeled using a system of ODEs. To derive these ODEs the flux between regions is computed and is used to calculate the transition rates between nodes. The purpose of this method of finding transition rates is to demonstrate that there exist transition rates that make the ODE model accurate. To implement this model in practice, however, some other technique of calculating the transition rates that does not rely on knowing the data at every time would be used. For example, researchers could mark population members and measure the average transition time between regions.

This paper demonstrates by simulation that data from the equilibrium state can be used to model the selection process as a system of ODEs over a network. The organization of the paper is as follows. Section 2 gives a mathematical description of the problem. Section 3 explains how a 2-dimensional fitness landscape was simulated and the ODE model was constructed. The results are described in Section 4. Section 5 provides areas of future research. Finally, conclusions are given in Section 6.

2 Preliminaries

We model a phenotype as taking values $\mathbf{x}$ in $\mathbb{R}^n$. Let $p(\mathbf{x}, t)$ be the probability density function of the population at a time $t \in \mathbb{R}_{\geq 0}$. For a given time t, $p(\mathbf{x}, t)$ is a probability density function describing the distribution of the population over the phenotype space.

We also have a fitness function $F(\mathbf{x})$ defined over the phenotype space. Roughly, this tells us the likelihood of successful reproduction for organisms of phenotype vector $\mathbf{x}$. Let $U = -F$ be the potential associated with the fitness function. Then the Fisher-Eigen equation is

$$\frac{\partial p(\mathbf{x}, t)}{\partial t} = (\overline{U}(t) - U(\mathbf{x}))p(\mathbf{x}, t) + D\nabla^2 p(\mathbf{x}, t),$$

where $\overline{U}(t) := \displaystyle\int_{\mathbb{R}^n} U(\mathbf{x})p(\mathbf{x}, t)\, dx$ is the average potential of the population at time t, and $D > 0$ is the diffusion constant.The diffusion constant in this case is a property of the phenotype space that expresses the flux of the population through an area over time, based on a population gradient. The distance and area units of D are in quantified phenotypes, which may be aggregated phenotypes as a result of dimension reduction. The time units of D will be appropriate to the system under study, such as days, years, generations or lifetimes.

3 Methods

We simulated the Fisher-Eigen process for a specific 2-dimensional fitness landscape using MATLAB. The particular landscape was chosen so that it could be modeled as a network of three nodes while still possessing interesting dynamics; in one dimension, the three node situation is less interesting because organisms on each of the outer nodes have to pass through the middle node to get to the other side. In two dimensions, however, organisms can move around the middle node.

We selected the fitness function to be

$$F(x_1, x_2) = 3e^{-20(x_1^2+(x_2-\frac{\sqrt{3}}{2})^2)} + e^{-10((x_1+0.5)^2+x_2^2)} + e^{-10((x_1-0.5)^2+x_2^2)} + \\ + \frac{1}{2}e^{-15(x_1-0.25)^2-5(x_2-\frac{\sqrt{3}}{4})^2} + \frac{1}{2}e^{-5x_1^2-15x_2^2} - \frac{1}{10}(x_1^2 + x_2^2).$$

We chose the fitness function with one global maximum and two other local maxima. This gave the highest peak at $P_1 = \left(0, \dfrac{\sqrt{3}}{2}\right)$, the middle peak at $P_2 = (0.5, 0)$, and the lowest peak at $P_3 = (-0.5, 0)$. The smaller Gaussians created a less steep valley between

P_3 and P_2 and between P_2 and P_1. The last term was added so that the boundary of the space has a low fitness. The population starts out at the lowest peak and moves toward the highest one. Some organisms move towards the highest peak directly but are slowed down because of the steepness of the highest peak. However, by first moving to the middle peak, the population can move to the highest one more easily. Which of these paths is optimal depends on the parameters in the fitness function and is reflected in the transition rates between peaks. The fitness function is shown graphically in Figure 1.

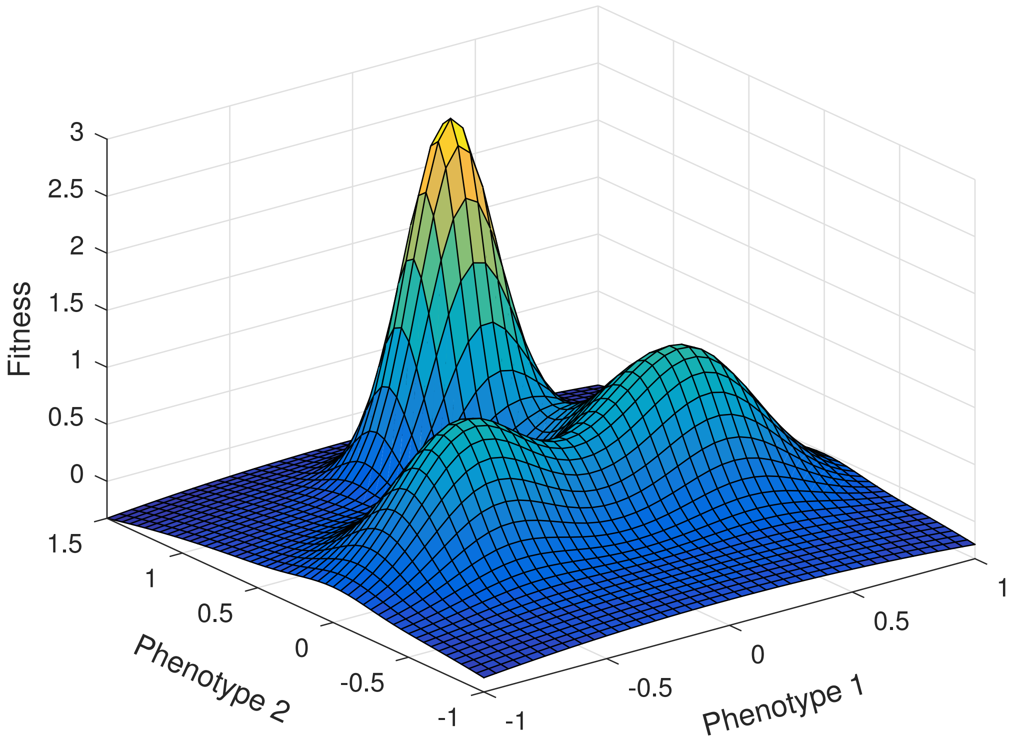


Figure 1: Fitness function used in the model

We modeled this process with a partial differential equation simulation of a discretized representation of the phenotype space around the peaks of the fitness function. We discretized the space and simulation time steps sufficiently to accurately capture the dynamic and steady state behavior of the model. We also tested the sensitivity of the responses to discretization granularity. A typical choice for the setup was a simulation area of $[-1.5, 1.5] \times [-1.5, 1.5]$ and the time interval $[0, 25]$ using $dx = dy = 0.04$ and $dt = 0.001$. We modeled p by

$$\frac{p(\mathbf{x}, t+\Delta t) - p(\mathbf{x}, t)}{\Delta t} = (\overline{U}(t) - U(\mathbf{x}))p(\mathbf{x}, t) + D\nabla_h^2 p(\mathbf{x}, t),$$

where

$$\overline{U}(t) = \sum U(\mathbf{x})p(\mathbf{x}, t)\Delta x_1 \Delta x_2$$

is the average energy and

$$\begin{aligned}\nabla_h^2 p((x_1, x_2), t) = & \frac{p((x_1 + \Delta x_1, x_2), t) + p((x_1 - \Delta x_1, x_2), t) - 2p((x_1, x_2), t)}{(\Delta x_1)^2} \\ & + \frac{p((x_1, x_2 + \Delta x_2), t) + p((x_1, x_2 - \Delta x_2), t) - 2p((x_1, x_2), t)}{(\Delta x_2)^2}\end{aligned}$$

is a finite difference operator representing the diffusion term. We added a Dirichlet boundary condition $p(\mathbf{x}, t) = 0$ whenever $\mathbf{x}$ lies on the boundary of the simulation area. We used the RK4 method to time step based on this equation.

We chose time steps sufficiently small such that

$$p(\mathbf{x}, t) \geq 0, \forall \mathbf{x} \Rightarrow p(\mathbf{x}, t + \Delta t) \geq 0, \forall \mathbf{x}. \tag{1}$$

so that p would remain nonnegative in all cases. Assuming p is nonnegative everywhere at time t, then $D\nabla_h^2 p(\mathbf{x}, t) \geq \frac{-4Dp(\mathbf{x}, t)}{(\Delta x_1)^2}$, since the tiles are square so $\Delta x_1 = \Delta x_2$. Furthermore, $\overline{U}(t) - U(\mathbf{x})$ is at least $\min_{\mathbf{y} \in \Omega} U(\mathbf{y}) - \max_{\mathbf{y} \in \Omega} U(\mathbf{y})$. The maximum value is approximately 0.5, occurring at each of the corners of the boundary. The minimum is approximately -3 and occurs near $\left(0, \frac{\sqrt{3}}{2}\right)$, the peak of the tallest Gaussian. Combining these estimates, we see that $(\overline{U}(t) - U(\mathbf{x}))p(\mathbf{x}, t) \geq -3.5p(\mathbf{x}, t)$. Substituting $\frac{p(\mathbf{x}, t+\Delta t) - p(\mathbf{x}, t)}{\Delta t} \geq \left(-3.5 - \frac{4D}{(\Delta x_1)^2}\right) p(\mathbf{x}, t)$ into Equation 1 gives the Δt constraint for the full PDE:

$$\Delta t \leq \left(3.5 + \frac{4D}{(\Delta x_1)^2}\right)^{-1}.$$

In particular, choosing $\Delta t = 0.001$ and $\Delta x_1 = \Delta x_2 = 0.04$ will work as long as $D \leq 0.39$. After each time step, we divided p by its integral $\sum p(\mathbf{x}, t)\Delta x_1 \, \Delta x_2$ so that p would remain a probability distribution, to account for any loss of population, for example due to numerical precision.

To analyze the dynamics of this system, we split the space into 3 regions that will correspond to the nodes of the network function that will model the system as an ODE. The behavior of each region in the PDE will be lumped into an ODE equation for the network model:

$$\Omega_1 = \left\{ (x_1, x_2) \in [-1.5, 1.5] \times [-1.5, 1.5] \, : \, y > |\frac{\sqrt{3}}{3}x| + \frac{\sqrt{3}}{6} \right\}$$

$$\Omega_2 = (0, 1.5] \times [-1.5, 1.5] \setminus \Omega_1$$

$$\Omega_3 = [-1.5, 0] \times [-1.5, 1.5] \setminus \Omega_1.$$

These regions are depicted in Figure 2.

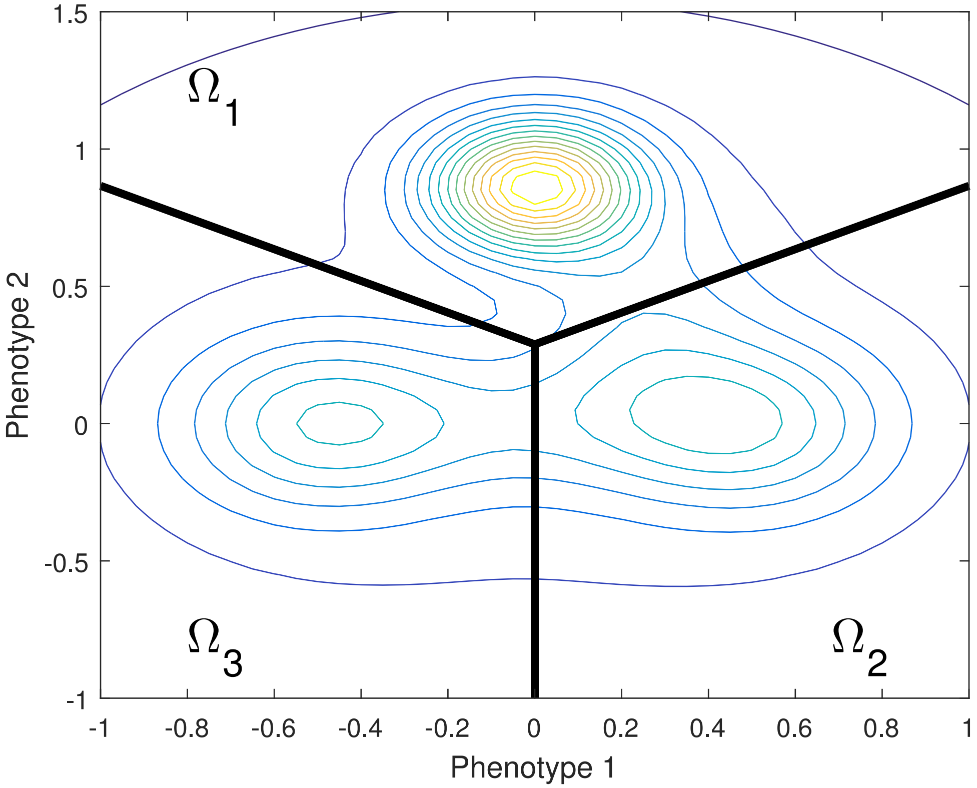


Figure 2: Partitioning the phenotype space into regions corresponding to network nodes. Contours show the fitness function peaks.

Let $I_i = \displaystyle\int_{\Omega_i} p(\mathbf{x}, t)\, d\mathbf{x}$, which represents the total population in region Ω_i. From the equation

$$\frac{\partial p(\mathbf{x}, t)}{\partial t} = (\overline{U}(t) - U(\mathbf{x}))p(\mathbf{x}, t) + D\nabla^2 p(\mathbf{x}, t),$$

we integrate both sides over the region Ω_i to obtain

$$\int_{\Omega_i} \frac{\partial p(\mathbf{x}, t)}{\partial t}\, d\mathbf{x} = \overline{U}(t) \int_{\Omega_i} p(\mathbf{x}, t)\, d\mathbf{x} - \int_{\Omega_i} U(\mathbf{x}) p(\mathbf{x}, t)\, d\mathbf{x} + D \int_{\partial\Omega_i} (\nabla p(\mathbf{x}, t)) \cdot \vec{n}\, ds, \qquad (2)$$

where $\partial\Omega_i$ is the boundary of Ω_i and $\vec{n}$ is the outward-pointing unit normal to $\partial\Omega_i$. The last term follows by the Divergence Theorem. Differentiating under the integral sign, we see that the left hand side is $\frac{dI_i}{dt}$. The last term on the right hand side is the flux through Ω_i, which because of the boundary condition is approximately the flux through the two line segments surrounding Ω_i. We chose basic linear region partitions based on the centers of the three fitness peaks in this model. Future work could study whether other partitioning schemes add more accuracy such as using saddle lines, equipotential-like isolines, or other model-based partition lines. If we let J_{ij} be the flux from Ω_i to Ω_j through the segment bordering Ω_i and Ω_j, we have $J_{ij} = -J_{ji}$ and $J_{ij} + J_{ik} = D \int_{\partial\Omega_i} (\nabla p(\mathbf{x}, t)) \cdot \vec{n}\, ds$ when i, j, k are all distinct. Then Equation 2 becomes

$$\frac{dI_i}{dt} = \overline{U}(t) \int_{\Omega_i} p(\mathbf{x}, t)\, d\mathbf{x} - \int_{\Omega_i} U(\mathbf{x}) p(\mathbf{x}, t)\, d\mathbf{x} + \sum_{j \neq i} J_{ij}. \tag{3}$$

We computed the flux across each boundary segment by discretizing the segment into N equally spaced points $\{x_{l_1}, x_{l_2}, \cdots, x_{l_N}\}$ and taking the summation $\sum_{i=1}^{N} \nabla p(x_{l_i}, t) \cdot \vec{n}\, ds$, where $ds = \frac{L}{N}$, L is the length of that segment, and $\vec{n}$ is the unit normal vector to the segment bordering Ω_i and Ω_j going from Ω_i to Ω_j. We found that $N = 15$ was sufficient to obtain an accurate approximation by testing the sensitivity of the result to line segment length, as we did with area and time discretization.

Let $U_i(t) = \frac{\int_{\Omega_i} U(\mathbf{x}) p(\mathbf{x}, t)\, d\mathbf{x}}{\int_{\Omega_i} p(\mathbf{x}, t)\, d\mathbf{x}}$. Because in each region we expect the population to be relatively stable around the peak, we assume that $U_i(t)$ is approximately constant. Thus, we can approximate $U_i(t)$ by its value at equilibrium, which is denoted U_i^{eq}. At equilibrium, we have $\frac{dI_1}{dt} = \frac{dI_2}{dt} = \frac{dI_3}{dt} = 0$. Note that Equation 3 is unchanged if we add a constant to $U(\mathbf{x})$. Hence, we may assume that $\overline{U}(t) = 0$ at equilibrium. Then Equation 3 becomes, at equilibrium,

$$0 = -U_i^{\text{eq}} I_i^{\text{eq}} + J_{ij}^{\text{eq}} + J_{ik}^{\text{eq}}. \tag{4}$$

We can rearrange this equation and use the fact that $J_{ij} = -J_{ji}$ to solve for U_i^{eq} in terms of three of the fluxes and the I_is.

$$\begin{aligned}U_1^{\text{eq}} &= \frac{-J_{21}^{\text{eq}} - J_{31}^{\text{eq}}}{I_1^{\text{eq}}}\\ U_2^{\text{eq}} &= \frac{J_{21}^{\text{eq}} - J_{32}^{\text{eq}}}{I_2^{\text{eq}}}\\ U_3^{\text{eq}} &= \frac{J_{31}^{\text{eq}} + J_{32}^{\text{eq}}}{I_3^{\text{eq}}}.\end{aligned}$$

So to run the ODE model based on the equilibrium data, we only need a method of estimating the flux in terms of the I_i values over time. We approximate the flux J_{ij} as being linear in I_i and I_j. In the case of a Smoluchowski process, such an assumption is reasonable because particles move essentially independently of the global movement of all the particles, so the rate of movement between peaks depends only on the number of organisms at each peak. In contrast, for the Fisher-Eigen process the chances of an organism surviving a crossing depends on the organism's fitness relative to the entire population, so it is not immediately clear that the flux can be linear in each population. We set up a linear flux model for Fisher-Eigen and show the process with details explained in the following paragraphs.

We wish to model J_{ij} as $k_{ji}I_j - k_{ij}I_i$, where k_{ij} is the *transition rate* from i to j. We assume that the transition rates are independent of time. Assuming such an approximation holds, from the equilibrium data we obtain 3 equations for the 6 unknown transition rates. In theory any set of transition rates that satisfies these equations will give an equilibrium solution for the model that matches that of the PDE simulation. However, to ensure the model follows the dynamic behavior of the PDE simulation, the transition rates should be chosen so that the flux approximations are accurate over time. Therefore, we used the least squares method to find the transition rates by comparing the actual flux over time extracted from the PDE simulation with the linear model of flux over time. We ran a transient simulation with a starting population far from equilibrium, and ran the simulation until the population and flux settled into equilibrium values. We split the time interval into equally spaced points $t_1, t_2, \cdots, t_M$, typically using 300 to 600 sample points.

Consider the following matrices:

$$X_1 = \begin{pmatrix} I_2(t_1) & I_3(t_1) \\ I_2(t_2) & I_3(t_2) \\ \vdots & \vdots \\ I_2(t_M) & I_3(t_M) \end{pmatrix}, \quad Y_1 = \begin{pmatrix} J_{23}(t_1) \\ J_{23}(t_2) \\ \vdots \\ J_{23}(t_M) \end{pmatrix}.$$

We found a vector $\beta_1 = (k_{23}, -k_{32})^T$ that minimizes $|Y_1 - X_1\beta_1|^2$. Similarly, we found vectors $\beta_2 = (k_{13}, -k_{31})^T$ and $\beta_3 = (k_{12}, -k_{21})^T$ for the corresponding matrices for J_{13} and J_{12}.

We chose initial conditions that put 98% of the population in region 3, in a narrow Gaussian centered on the fitness peak in that region. As the simulation progressed to equilibrium, population flowed to regions 1 and 2 with the final value in region 3 being the smallest. The final values in all three regions depend on the fitness function and the diffusion constant D. At the beginning of the simulation, there is a short lag time while the initially applied population starts to flow resulting in rapid flux changes briefly. Thus, using $t_1 = 0$ causes error in computing the k_{ij}'s. We typically started flux samples for the least squares fit calculations at approximately time $t_1 = 1$ out of a total simulation time of 25.

With the transition rates, we obtain a system of ODEs that models I_1, I_2, and I_3.

$$\begin{aligned} \frac{dI_1}{dt} &= \left(\sum_{i=1}^{3} U_i^{\text{eq}} I_i - U_1^{\text{eq}}\right) I_1 - k_{12}I_1 + k_{21}I_2 - k_{13}I_1 + k_{31}I_3 \\ \frac{dI_2}{dt} &= \left(\sum_{i=1}^{3} U_i^{\text{eq}} I_i - U_2^{\text{eq}}\right) I_2 - k_{23}I_2 + k_{32}I_3 - k_{21}I_2 + k_{12}I_1 \\ \frac{dI_3}{dt} &= \left(\sum_{i=1}^{3} U_i^{\text{eq}} I_i - U_3^{\text{eq}}\right) I_3 - k_{31}I_3 + k_{13}I_1 - k_{32}I_3 + k_{23}I_2. \end{aligned}$$

We then simulated this system of ODEs using MATLAB's built in ode45 function. We used the same time interval as in the PDE simulation, typically $[0, 25]$. The initial conditions on I_i were the same as the initial conditions of the PDE simulation, with 98% of the population starting in region 3.

4 Results

We ran the simulation outlined in Section 3 with D varying between 0.01 and 0.2. Figure 3 shows the initial population distribution with the sharp peak in region 3. Figure 4 shows the equilibrium population distribution computed by the PDE simulation for various values of D. To check the accuracy of the linear model of flux, we plotted it against the flux that was computed as described in Section 3 during the transient response of the PDE simulation. The plots of J_{32}, J_{31}, and J_{21} for both $D = 0.01$ and $D = 0.09$ are shown in Figure 5.

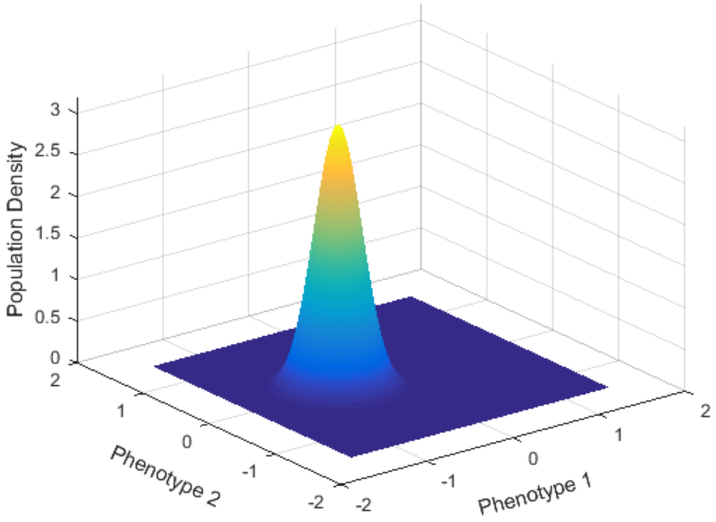
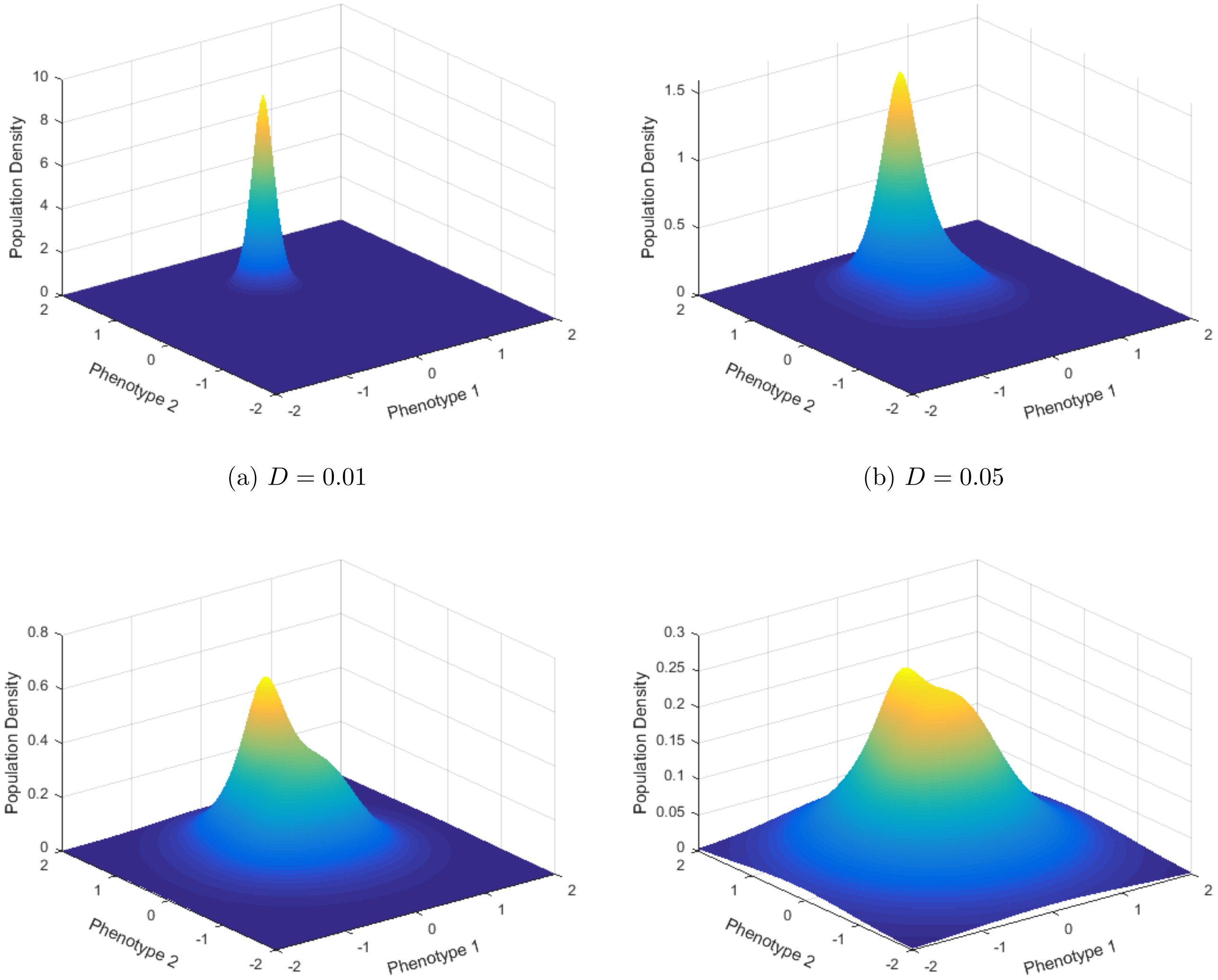


Figure 3: Starting population for transient simulations has 98% of the population in region 3



(a) $D = 0.01$

(b) $D = 0.05$

(c) $D = 0.09$

(d) $D = 0.20$

Figure 4: Equilibrium population distribution computed by PDE simulations for various D values

Figure 5 provides numerical evidence that for both large and small D, the linear models are an accurate approximation of the flux terms. The linear models all approximate the long-term behavior of the actual fluxes, and in general the flux and linear model are similar qualitatively. The largest error in these models appears for small t, which indicates that there still may be a loss of accuracy due to a lag time at the start of the simulation.

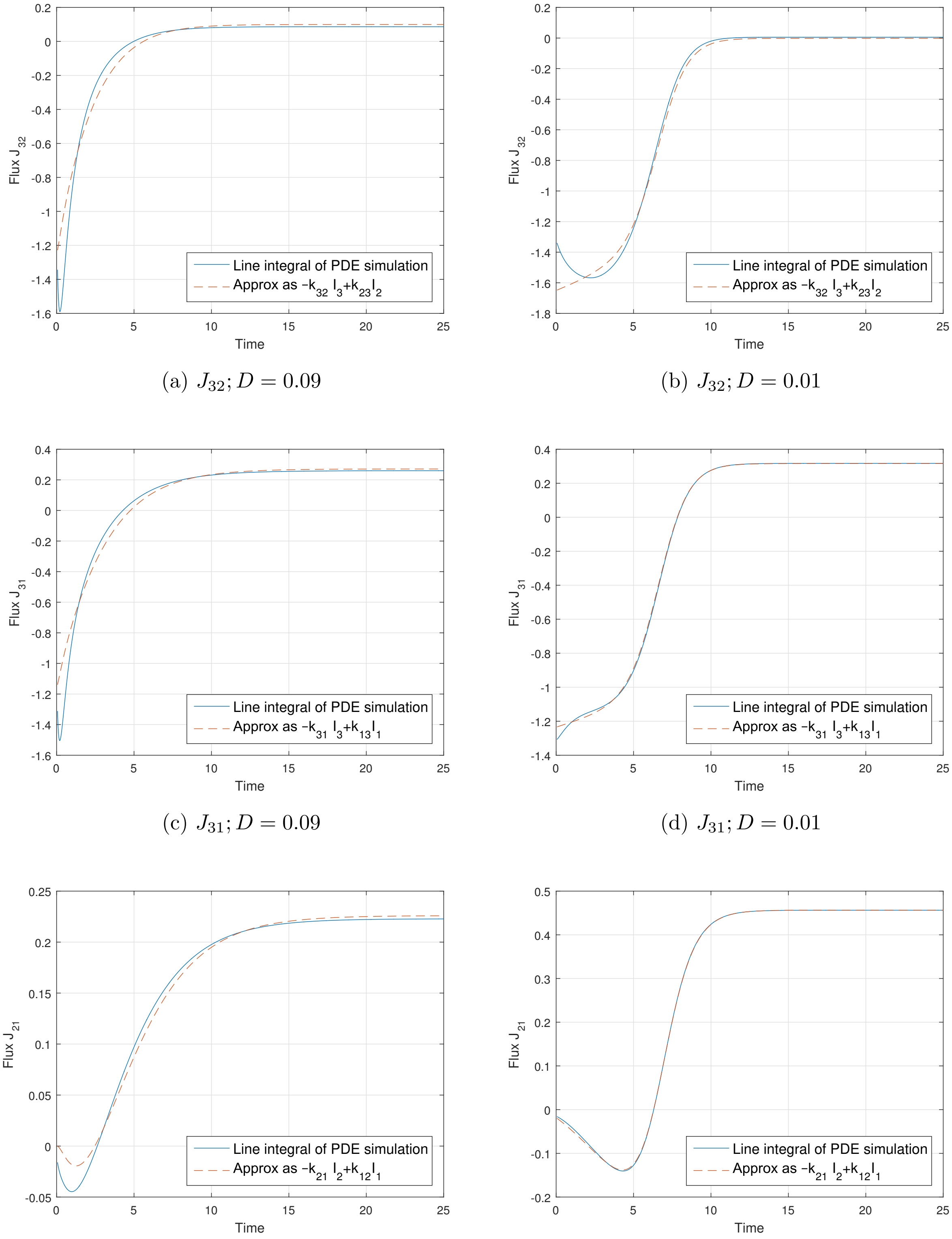
Once we obtained the transition rates from these models, we simulated I_1, I_2, and I_3

by a system of ODEs, as described in Section 3. Figure 6 shows the values of I_i over time as described by the PDE simulation and by the ODE model for $D = 0.09$, $D = 0.05$ and $D = 0.01$, respectively.

The ODE model has the desired property of reaching the same equilibrium state as the actual PDE simulation. Qualitatively the graphs show similar dynamics. However, the time the ODE model takes to reach equilibrium is sometimes slightly less than the time the PDE model takes, particularly for small D.

One possible cause of the increased rate for the ODE model is that there is no lag to move around a peak. In the PDE model, once an organism has entered one of the three regions it takes time for that organism to move within that region to reach a point where it can move to another region. In contrast, in the ODE model, once an organism moves to a region it can instantly move to a different region at the next time step. The lag is more pronounced with slower diffusion (lower D values). Also, for these trial simulations to validate the mathematical models, the population starts at an artificially narrow peak in the interior of one region, so minimal flux is present until some population reaches an edge, during which the PDE simulations show no population changes initially. For a more natural initial population smoothly distributed through the phenotype space, some flux will be present at the region boundaries which will minimize the PDE response lag times and allow the PDE and ODE responses to match more closely.

Based on our trials, the ODE model will reach an equilibrium state determined only by the transition rates k_{ij} and average fitness U_i. This means one can change the initial distribution of the population without changing the equilibrium solution. For example, instead of having all the population start at the lowest fitness peak, one could have them all start at the middle peak or all start at the highest peak. Figure 7 shows the results from running the $D = 0.09$ case with those three initial conditions, using parameters k_{ij} and U_i determined by the $D = 0.09$ case analyzed for Figures 5 and 6.



(a) $J_{32}; D = 0.09$

(b) $J_{32}; D = 0.01$

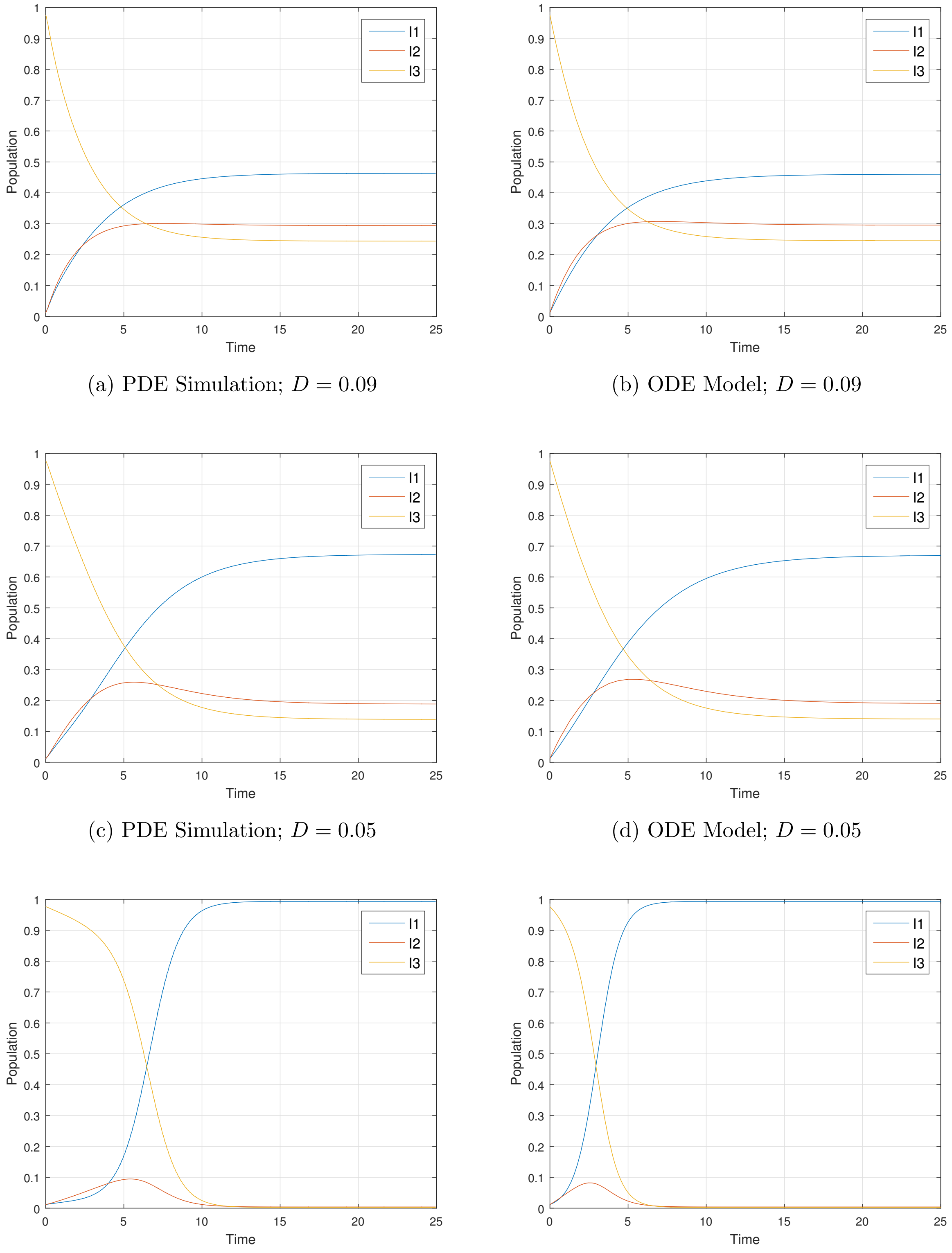
(c) $J_{31}; D = 0.09$

(d) $J_{31}; D = 0.01$

(e) $J_{21}; D = 0.09$

(f) $J_{21}; D = 0.01$

Figure 5: Comparison of the linear model of flux with the actual flux



(a) PDE Simulation; $D = 0.09$

(b) ODE Model; $D = 0.09$

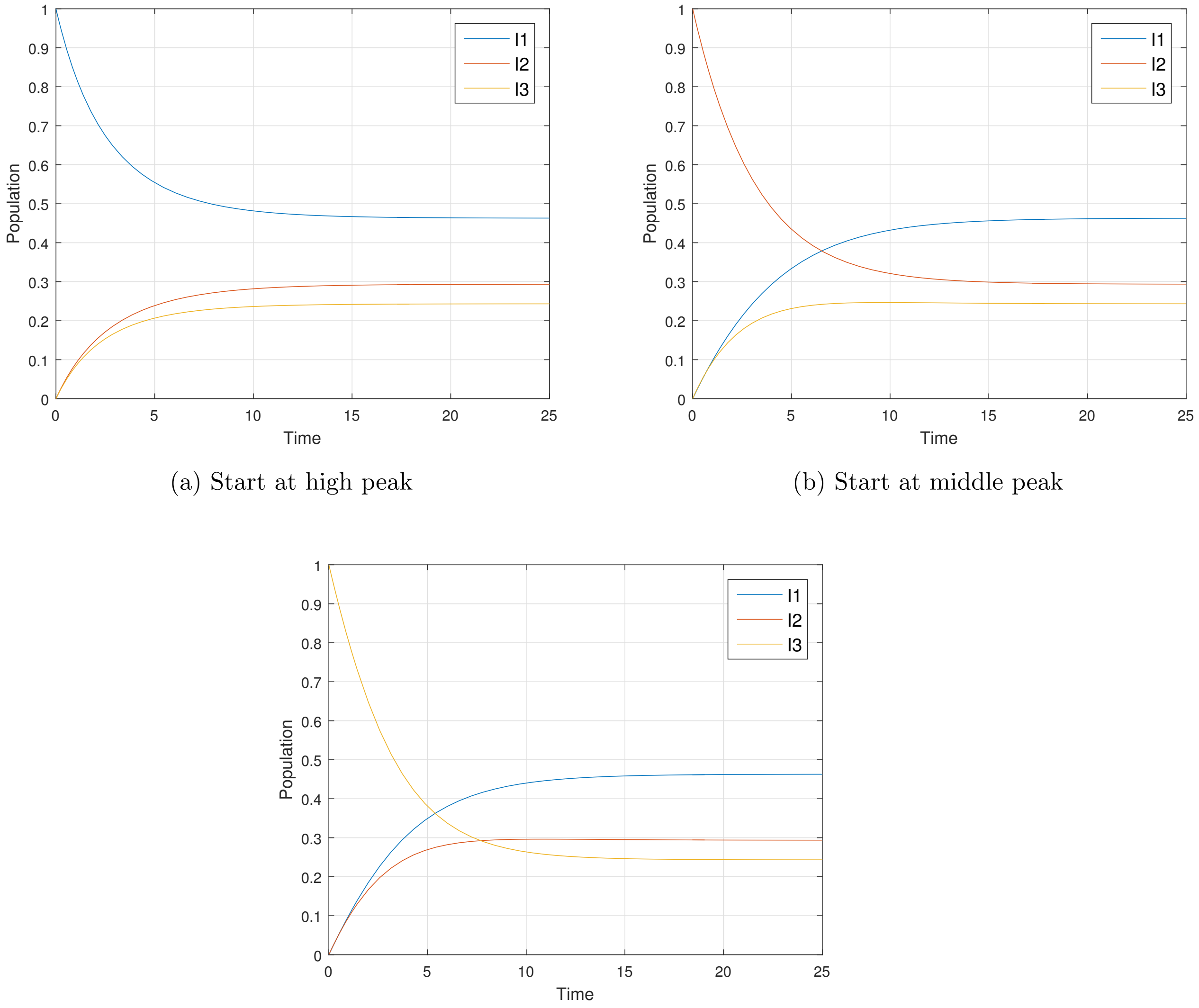
(c) PDE Simulation; $D = 0.05$

(d) ODE Model; $D = 0.05$

(e) PDE Simulation; $D = 0.01$

(f) ODE Model; $D = 0.01$

Figure 6: Comparison of the PDE simulation with the ODE model for various D values. Initial condition: 98% of population starts in a narrow peak in region 3.



(a) Start at high peak

(b) Start at middle peak

(c) Start at low peak

Figure 7: Results of the ODE model under the same parameters and three different initial conditions; $D = 0.09$

5 Future Work

To implement this model in practice, one would need to find the transition rates k_{ij} without relying on the flux data taken over time. One such method would be to derive analytic expressions for these transition rates. The method for doing so would be similar to that of [8]. First, use spectral analysis to determine an equation for the probability density

$p(\mathbf{x}, t)$ in terms of eigenfunctions. Then, truncate the eigenfunction expansion to obtain an approximate solution for $p(\mathbf{x}, t)$. After solving the master equation in terms of the transition rates, compare the expression it gives for $I_i(t)$ with the one given by integrating the truncated eigenfunction expansion over Ω_i. From this equation derive approximate expressions for the transition rates in terms of the first few eigenvalues.

Another direction of future work could be to prove some of these results analytically. One could attempt to prove that the initial conditions do not affect the equilibrium state of the ODE model, and further that the ODE model is asymptotically stable, which it appears to be in practice. A more difficult task would be to determine for which cases the flux is approximated well by linear models. Based on our trials, we would expect the approximation to be better for smaller D.

We would also like to study the dynamics of the ODE model where it differs at a fine grain level from those of the PDE model. The network model is a lumped model of the system, which assumes state variables of a node such as population density apply everywhere in the node. Slow diffusion with low D values results in larger differences in population density across a node, making the accuracy of a single lumped model more problematic compared to the PDE model. With this in mind, one possible modification to the model would be to generate additional subnodes to model diffusion within a top-level model based on D and the size of the region. Also, adding transition nodes between each region may capture additional diffusion effects. Another aspect of improving the ODE model would be studying the initial conditions such that the ODE model could match the PDE more closely at time $t_1 = 0$.

Another area of focus for future work would be other partitioning schemes and criteria for mapping a phenotype space onto a network of nodes based on the biological data. For larger and more complex systems, these methods would require algorithms and automation. We would also like to extend this work to fitness functions other than the one chosen. It would be interesting to consider fitness functions which produce much more complicated networks, in which organisms have many viable paths to reach peaks. Also, we would like to consider fitness functions in higher dimensions. A 2-dimensional fitness function is a reasonable starting place, but it does not capture the complexity that would be involved in modeling in higher dimensions.

6 Conclusions

This paper detailed a simulation of the Fisher-Eigen process for a 2-dimensional fitness landscape, and it explained the use of this simulation in constructing a model of the Fisher-Eigen process as a system of ODEs over a network. The results indicated that the flux between two nodes in the network could be approximated as a linear combination of the integrals of the probability density over each region. The results also showed that this ODE model reached well-matched transient and precise equilibrium solutions. The motivation for constructing such a model was to simplify the modeling of high-dimensional fitness landscapes, well beyond 2 dimensions, and model dynamic behavior with data taken from equilibrium conditions. This prototype can be replicated by others, and the savings in computational complexity could enable the inclusion of far more phenotypes and more complicated fitness functions than can be used in current Fisher-Eigen models. These new models could illuminate behavior for populations previously too complex to predict. This would allow researchers to combat the growing global public health menace of antimicrobial resistant pathogens, for example, as well as many other applications.

7 Acknowledgments

I would like to thank my mentor, Dominic Skinner, for his guidance and support throughout this project. Many thanks also go to Professor Thomas Hagstrom of SMU for his feedback on this paper, and to Dr. John Rickert and Dr. Tanya Khovanova for their advice on writing and presenting this work. In addition, I would like to thank MIT, and specifically the MIT Math Department, for hosting this project, and the RSI staff for helping me throughout the process. I am very grateful also to CEE and my sponsors, Mr. and Mrs. Kenneth Hennes and Dr. Albert C. Lin, for making this project possible through their support.

References

[1] R. Feistel and W. Ebeling. *Evolution of complex systems: selforganisation, entropy and development*, volume 30. Springer, 1989.

[2] M. Eigen. Selforganization of matter and the evolution of biological macromolecules. *Naturwissenschaften*, 58(10):465–523, 1971.

[3] R. A. Fisher. *The genetical theory of natural selection.* Oxford University Press, 1930.

[4] W. Ebeling and R. Feistel. Studies on manfred eigen's model for the self-organization of information processing. *European Biophysics Journal*, 47(4):395–401, 2018.

[5] A. Forrow, F. G. Woodhouse, and J. Dunkel. Functional control of network dynamics using designed laplacian spectra. *arXiv preprint arXiv:1801.01573*, 2018.

[6] P. Pearce, F. G. Woodhouse, A. Forrow, A. Kelly, H. Kusumaatmaja, and J. Dunkel. Inference of complex state transition networks via reconstructed high-dimensional energy landscapes. 2018. Unpublished preprint.

[7] J. Dunkel, L. Schimansky-Geier, and W. Ebeling. Exact solutions for evolutionary strategies on harmonic landscapes. *Evolutionary computation*, 12(1):1–17, 2004.

[8] J. Dunkel, W. Ebeling, L. Schimansky-Geier, and P. Hänggi. Kramers problem in evolutionary strategies. *Physical Review E*, 67(6):061118, 2003.

[9] T. Asselmeyer, W. Ebeling, and H. Rosé. Evolutionary strategies of optimization. *Physical Review E*, 56(1):1171, 1997.

[10] H. A. Kramers. Brownian motion in a field of force and the diffusion model of chemical reactions. *Physica*, 7(4):284–304, 1940.