D.8 Modeling *cro–cI* Genetic Network

Zarif Ahmed

February 2025

Abstract

The decision between lysis and lysogeny in bacteriophage lambda is dictated by the mutually repressive interaction between the transcription factors cI and cro. In this study, we model this interaction using both a deterministic model and stochastic model to investigate how initial concentration of cro and cI RNA affect the systems trajectory. Through the deterministic model, made using the Forward Euler method, we show that there are tipping points of initial concentrations that favor lysis or lysogeny. Lysis is generally preferred unless there is some amount of starting cI protein present when lambda injects its host. The predictable nature of the deterministic model also makes it easier to detect thresholds and ranges for our variables and parameters. The stochastic model implemented is useful for capturing the inherent randomness in chemical reactions. In addition, we introduce a stress factor to represent the stress-induced degradation of cl protein which results in the switch from a stable lysogenic pathway to lytic pathway. The deterministic model helps identify the critical threshold for the switch which is introduced to the stochastic model to add variability to the simulation. Our results highlight the importance of initial concentrations and stress factor in determining the reproduction cycle of

Introduction

When a bacteria is infected by the bacteriophage lambda, two fates are possible. First, the virus may replicate many progeny within the bacteria, ultimately resulting in the bursting of the bacteria to release new phage; this

is known as lysis. Second, the phage DNA remains in the bacteria without making new phage in process called Lysogeny. The phage DNA is replicated along with the bacterial genome during cell division, and thus all descendants of the infected cell also carry the phage DNA. The lysogenic state is generally stable, meaning that all progeny of a lysogenic bacteria remain in the lysogenic state indefinitely. However, exposure to stress (such as radiation) can convert a cell in the lysogenic state into the lytic phase. The bacteria enters a phase based on a pair of mutually repressive transcription factors, cI and cro. When levels of cI are high and levels of cro are low, an infected bacteria will be in the lysogenic phase. When levels of cI are low and levels of cro are high, the bacteria enters lytic phase. A simple model of this process can be given by the equations:

$$\frac{d[cI_P]}{dt} = \nu_1 - \nu_2, \quad \frac{d[cI_R]}{dt} = \nu_3 - \nu_4$$
$$\frac{d[cro_P]}{dt} = \nu_5 - \nu_6, \quad \frac{d[cro_R]}{dt} = \nu_7 - \nu_8$$

where:

$$\nu_1 = \omega_{cI}[cI_R], \quad \nu_2 = \chi_{cI,P}[cI_P],$$
$$\nu_3 = \mu_{cI}\left(1 - \frac{[cro_P]^2}{K_{cro,1/2}^2 + [cro_P]^2}\right), \quad \nu_4 = \chi_{cI,R}[cI_R]$$
$$\nu_5 = \omega_{cro}[cro_R], \quad \nu_6 = \chi_{cro,P}[cro_P],$$
$$\nu_7 = \mu_{cro}\left(1 - \frac{[cI_P]^2}{K_{cI,1/2}^2 + [cI_P]^2}\right), \quad \nu_8 = \chi_{cro,R}[cro_R]$$

where:

- $[cI_P]$: cI protein concentration.
- $[cI_R]$: cI RNA concentration.
- $[cro_P]$: cro Protein concentration.
- $[cro_R]$: cro RNA concentration.
- ω_{cI}: Rate of translation of cI_P

- ω_{cro}: Rate of translation of cro_P
- μ_{cI}: Maximum cI_{rna} transcription rate.
- μ_{cro}: Maximum cro_{rna} transcription rate.
- $\chi_{cI,P}$: cI protein degradation rate.
- $\chi_{cI,R}$: cI RNA degradation rate.
- $\chi_{cro,P}$: cro protein degradation rate.
- $\chi_{cro,R}$: cro RNA degradation rate.
- $K_{cI,1/2}$: cI protein concentration at half-maximal activation.
- $K_{cro,1/2}$: cro protein concentration at half-maximal activation.

In this study we will simulate this system using both a deterministic model using Forward Euler approach and a stochastic model using the Gillespie Algorithm. We analyze how different starting concentrations affect the trajectories of out models and add a stress factor to simulate the transition from Lysogeny to Lysis caused from stress. Unless otherwise specified, all results were obtained from simulations run with the parameters values below.

$$\chi_{cI,R} = \chi_{cI,P} = 1.2\, s^{-1} \quad \chi_{cro,R} = \chi_{cro,P} = 0.8\, s^{-1}$$

$$\omega_{cI} = \mu_{cI} = \omega_{cro} = \mu_{cro} = 50\, s^{-1} \quad K_{cI,1/2} = K_{cro,1/2} = 10\,\text{molecules/cell} \cdot s^{-1}$$

The initial codes for both the deterministic and stochastic models are given in Figures 1, 2, and 3.

```
cl_omega = 50; %cl translation rate
cl_Xp = 1.2; %cl protein degredation rate
cl_mu = 50; %cl transcription rate
cl_Xr = 1.2; %cl rna degredatation rate
cl_khalf = 10; % cl k_half

cro_Khalf = 10; %cro k_half
cro_omega = 50; %cro translation rate
cro_Xp = 0.8; % cro protein degredation rate
cro_mu = 50; % cro transcription rate
cro_Xr = 0.8; %cro rna degredation rate
time = 20; %total run time of the simulation
time_step = .01;
num_steps = time/time_step; %num of steps taken calculation from total time divided by the time step

%intializing arrays to hold all concentrations and the time
cl_pro = zeros(1, num_steps);
cl_rna = zeros(1, num_steps);
cro_pro = zeros(1, num_steps);
cro_rna = zeros(1, num_steps);
t_values = zeros(1, num_steps);

%starting concentrations
cl_rna(1) = 0;
cl_pro(1) = 0;
cro_rna(1) = 0;
cro_pro(1) = 0;

for i = 1:num_steps-1
    t_values(i+1) = t_values(i) + time_step;     %calculating t of next step

    %calculating all rate of change of concentrations
    v1 = cl_omega * cl_rna(i);
    v2 = cl_Xp * cl_pro(i);
    v3 = cl_mu * (1- cro_pro(i)^2/(cro_Khalf^2 + cro_pro(i)^2));
    v4 = cl_rna(i) * cl_Xr;
    v5 = cro_omega * cro_rna(i);
    v6 = cro_Xp * cro_pro(i);
    v7 = cro_mu * (1- cl_pro(i)^2 /(cl_khalf^2 + cl_pro(i)^2));
    v8 = cro_Xr * cro_rna(i);

    %Forward Eular implementation for calculating concentration at next step
    cl_pro(i+1) = cl_pro(i) + time_step * (v1 - (v2));
    cl_rna(i+1) = cl_rna(i) + time_step * (v3 - v4);

    cro_pro(i+1) = cro_pro(i) + time_step * (v5 - v6);
    cro_rna(i+1) = cro_rna(i) + time_step * (v7 - v8);
end
figure;
hold on;
plot(t_values, cl_pro, 'r', 'DisplayName', "cl_{pro}, cl\_pro(1) = 0");
plot(t_values, cl_rna, 'g', 'DisplayName', "cl_{rna}, cl\_rna(1) = 0");
plot(t_values, cro_pro, 'b', 'DisplayName', "cro_{pro}, cro\_pro(1) = 0");
plot(t_values, cro_rna, 'k', 'DisplayName', "cro_{rna}, cro\_rna(1) = 0");
title('Concentration over Time');
xlabel('Time')
ylabel('Concentration')
```

Figure 1: Deterministic model, implemented using Forward Euler

```
cl_omega = 50;
cl_Xp = 1.2;
cl_mu = 50;
cl_Xr = 1.2;
cl_khalf = 10;

cro_Khalf = 10;
cro_omega = 50;
cro_Xp = 0.8;
cro_mu = 50;
cro_Xr = 0.8;

figure;
hold on;

nums_steps = 50000;
num_reactions = 8;
%for loop repeating the simulation 20 times
for k = 1:20
    %initializing concentrations
    cl_pro = zeros(1, nums_steps);
    cl_rna = zeros(1, nums_steps);
    cro_pro = zeros(1, nums_steps);
    cro_rna = zeros(1, nums_steps);
    t_values = zeros(1, nums_steps);
    cl_rna(1) = 50;
    cl_pro(1) = 0;
    cro_rna(1) = 0;
    cro_pro(1) = 0;
    for i= 1:nums_steps-1
        %calculating each rection rate
        rxn_rate = zeros(1, num_reactions);
        rxn_rate(1) = cl_omega * cl_rna(i);
        rxn_rate(2) = cl_Xp * cl_pro(i);

        rxn_rate(3) = cl_mu * (1- cro_pro(i)^2/(cro_Khalf^2 + cro_pro(i)^2));
        rxn_rate(4) = cl_rna(i) * cl_Xr;

        rxn_rate(5) = cro_omega * cro_rna(i);
        rxn_rate(6) = cro_Xp * cro_pro(i);

        rxn_rate(7) = cro_mu * (1- cl_pro(i)^2 /(cl_khalf^2 + cl_pro(i)^2));
        rxn_rate(8) = cro_Xr * cro_rna(i);
        %calculate next time step
        alpha0 = sum(rxn_rate);
        y = rand();
        tau = -log(y)/alpha0;
        t_values(i+1) = t_values(i) + tau;
        %calculate next reaction
        y = rand();
        y_mod = alpha0*y;
        next_rxn = 0;
        for j = 1:num_reactions
            if y_mod < sum(rxn_rate(1:j))
                next_rxn = j;
                break;
            end
```

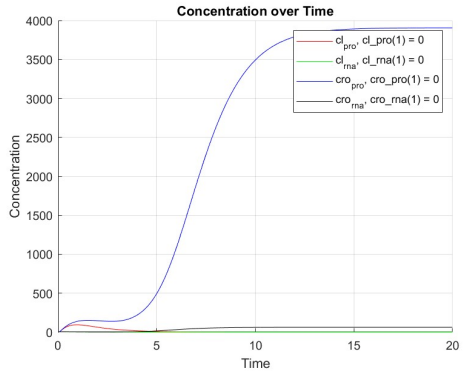
Figure 2: Stochastic model, partial code used to initialize parameters, and calculating time and reactions

```
        %select next reaction
        if (next_rxn == 1)
            cl_pro(i+1) = cl_pro(i) + 1;
            cl_rna(i+1) = cl_rna(i);
            cro_pro(i+1) = cro_pro(i);
            cro_rna(i+1) = cro_rna(i);
        elseif (next_rxn == 2)
            cl_pro(i+1) = cl_pro(i) - 1;
            cl_rna(i+1) = cl_rna(i);
            cro_pro(i+1) = cro_pro(i);
            cro_rna(i+1) = cro_rna(i);
        elseif (next_rxn == 3)
            cl_pro(i+1) = cl_pro(i);
            cl_rna(i+1) = cl_rna(i) + 1;
            cro_pro(i+1) = cro_pro(i);
            cro_rna(i+1) = cro_rna(i);
        elseif (next_rxn == 4)
            cl_pro(i+1) = cl_pro(i);
            cl_rna(i+1) = cl_rna(i) - 1;
            cro_pro(i+1) = cro_pro(i);
            cro_rna(i+1) = cro_rna(i);
        elseif (next_rxn == 5)
            cl_pro(i+1) = cl_pro(i);
            cl_rna(i+1) = cl_rna(i);
            cro_pro(i+1) = cro_pro(i) + 1;
            cro_rna(i+1) = cro_rna(i);
        elseif (next_rxn == 6)
            cl_pro(i+1) = cl_pro(i);
            cl_rna(i+1) = cl_rna(i);
            cro_pro(i+1) = cro_pro(i) - 1;
            cro_rna(i+1) = cro_rna(i);
        elseif (next_rxn == 7)
            cl_pro(i+1) = cl_pro(i);
            cl_rna(i+1) = cl_rna(i);
            cro_pro(i+1) = cro_pro(i);
            cro_rna(i+1) = cro_rna(i) + 1;
        elseif (next_rxn == 8)
            cl_pro(i+1) = cl_pro(i);
            cl_rna(i+1) = cl_rna(i);
            cro_pro(i+1) = cro_pro(i);
            cro_rna(i+1) = cro_rna(i) - 1;
        end
    end
    subplot(2,1,1); % Concentration vs Time
    hold on;
    stairs(t_values, cl_pro, 'r');
    stairs(t_values, cl_rna, 'g');
    stairs(t_values, cro_pro, 'b');
    stairs(t_values, cro_rna, 'k');
    legend('cl_{pro}', 'cl_{rna}', 'cro_{pro}', 'cro_{rna}')
    xlabel('Time');
    ylabel('Concentration');
    grid on;
    title('All Concentrations');

    subplot(2,1,2); % Cro_pro vs Cl_pro
    stairs(cl_pro, cro_pro);
    hold on;
    xlabel('Cl_{Pro}');
    ylabel('Cro_{Pro}');
    grid on;
    title('cl_{pro} vs cro_{pro}');
    axis([0 600 0 1000])
end
%legend('cl_{pro}', 'cl_{rna}', 'cro_{pro}', 'cro_{rna}')
%xlabel('Time');
%ylabel('Concentration');
%grid on;
hold off;
```

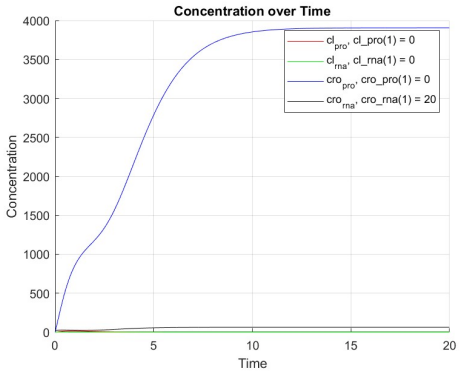
Figure 3: Stochastic model, remaining code for selecting next reaction and plotting the results

Results and Discussion

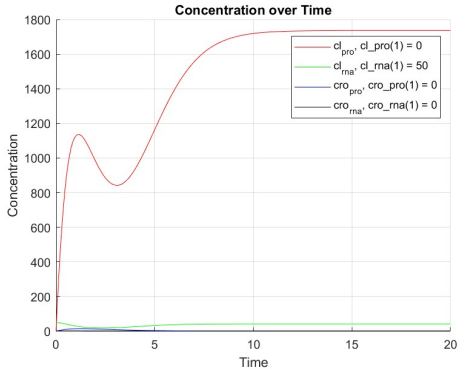
To start off, we will explore how different initial concentrations impact the trajectory of our model. We run 3 simulations for 20 seconds each with different initial conditions: all initial concentrations set to 0; 20 molecules of cro RNA per cell only; 50 molecules of cI RNA per cell only. We plot the data in a concentration vs time graph shown in Figure 4.



(a) All initial concentrations set to 0



(b) 20 molecules of cro RNA per cell only



(c) 50 molecules of cI RNA per cell only

Figure 4: Simulating Deterministic model with varying starting concentrations

The results of Figure 4a show that setting all initial concentrations to 0 leads to $[cro_{pro}]$ always having a higher concentration than cI_{pro} through the simulation. Our simulation cannot reach a stable Lysogenic state without some amount of initial $[cI_{rna}]$. This is further supported by Figure 4c where

a starting $[cI_{rna}]$ of 50 molecules leads to a stable Lysogenic state whereas, in Figure 4b, a starting concentration of 20 $[cro_{rna}]$ molecules significantly inhibits $[cI_{rna}]$ production which in turn inhibits the translation of $[cI_{pro}]$.

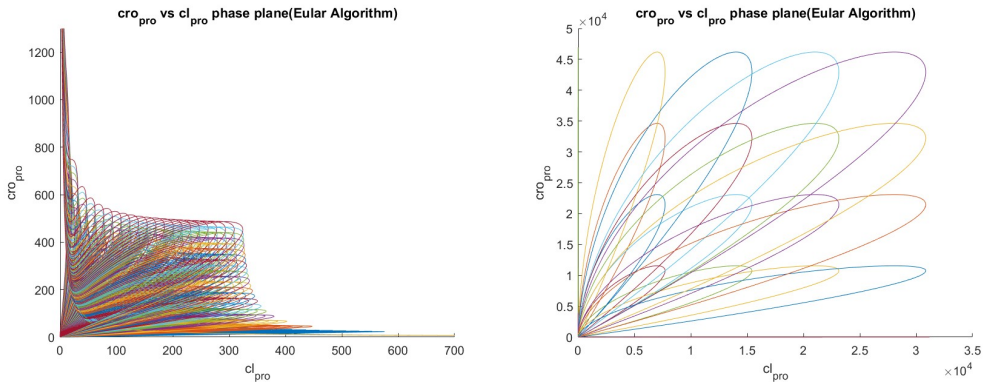
To better understand the dynamic between *cro* and *cI* in a mutually repressive system, we plot a $[cro_{pro}]$ vs $[cI_{pro}]$ graph. We run multiple simulations with varying starting concentrations of *cro* and *cI* RNA. We can do so by modifying the deterministic model code in Figure 1. We next the existing for loop in a double for loop to repeat the simulation with varying starting concentrations. The modified code is shown in Figure 5:

```
for j = 0:500:2000
    for k = 0:500:2000
        cl_pro = zeros(1, num_steps);
        cl_rna = zeros(1, num_steps);
        cro_pro = zeros(1, num_steps);
        cro_rna = zeros(1, num_steps);
        t_values = zeros(1, num_steps);

        cro_rna(1) = k;
        cl_rna(1) = j;
        cro_pro(1) = 0;
        cl_pro(1) = 0;

        for i = 1:num_steps-1
            t_values(i+1) = t_values(i) + time_step;
            v1 = cl_omega * cl_rna(i);
            v2 = cl_Xp * cl_pro(i);

            v3 = cl_mu * (1 - cro_pro(i)^2 / (cro_Khalf^2 + cro_pro(i)^2));
            v4 = cl_rna(i) * cl_Xr;

            v5 = cro_omega * cro_rna(i);
            v6 = cro_Xp * cro_pro(i);

            v7 = cro_mu * (1 - cl_pro(i)^2 / (cl_khalf^2 + cl_pro(i)^2));
            v8 = cro_Xr * cro_rna(i);

            cl_pro(i+1) = cl_pro(i) + time_step * (v1 - v2);
            cl_rna(i+1) = cl_rna(i) + time_step * (v3 - v4);

            cro_pro(i+1) = cro_pro(i) + time_step * (v5 - v6);
            cro_rna(i+1) = cro_rna(i) + time_step * (v7 - v8);
        end

        % Plot the trajectory for the current initial conditions
        plot(cl_pro, cro_pro);
    end
end
```

Figure 5: Part of the original code modified to run multiple simulations at once

We run simulations of 2 sets of concentrations: (1) all combinations from

0 to 20 molecules in intervals of 1; (2) all combinations of 0 to 2000 molecules in intervals of 500. The initial concentration of both cro and cI proteins are zero in all cases. The results are shown in Figure 7.



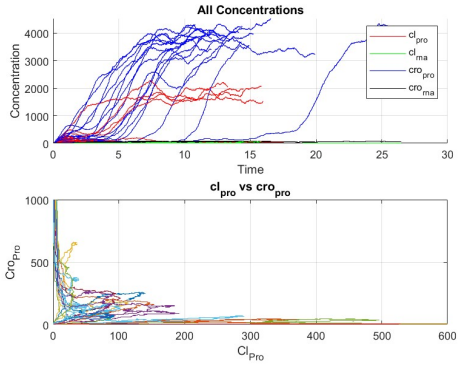
(a) 0 to 20 molecules, intervals of 1 (b) 0 to 2000 molecules, intervals of 500

Figure 6: Simulation of varying initial concentrations of $[cro_{rna}]$ and $[cI_{rna}]$

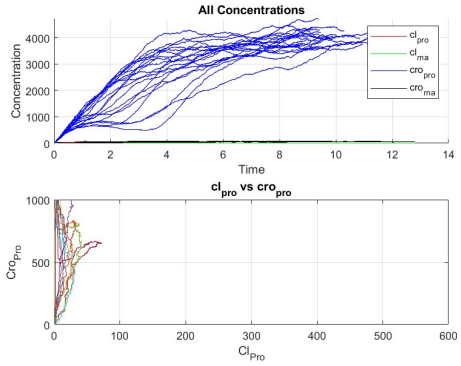
Our simulations demonstrate that the initial concentrations of $[cro_{rna}]$ and $[cI_{rna}]$ are critical in determining whether the system enters the lytic or lysogenic pathway. This means that when our phage, lambda, attaches itself to a host and injects its DNA, it also needs some amount of initial concentration of $[cI_{rna}]$ so that it enters a stable Lysogenic state. This could either happen from the phage DNA containing some initial promoter of $[cI_{rna}]$ production, or the Bacteria having some $[cI_{rna}]$ before the phage having injected its DNA. The minimum concentration required to enter lysis, as shown by Figure 4a, is when all concentrations are at 0. Figure 6a shows that at least 6 molecules of initial $[cI_{rna}]$ is required to enter stable Lysogeny when $[cro_{rna}]$ starts at 0 molecules.

A deterministic model has many limitations. No consideration of spatial variation of concentrations in different parts of the cell. No consideration of the indivisible nature of molecules infinitesimal changes in concentration are not possible in reality. No consideration of the random collisions that underly chemical reactions. A Stochastic model better captures these various factors and the randomness that exists in chemical reactions at this micro scale. Figure 2 and 3 is our implementation of a stochastic model in MATLAB using the Gillespie algorithm. We run three sets of 20 simulations of 50,000 steps each, and plot the results for each set as both concentrations versus

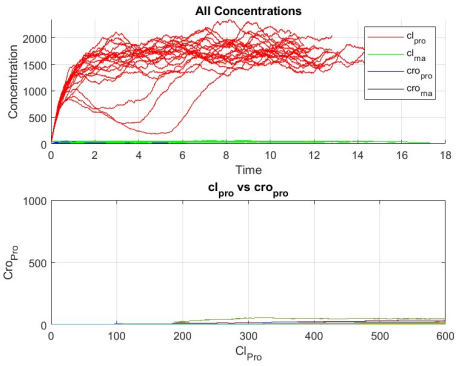
time and on the $[cro_{rna}]$ vs $[cI_{pro}]$ phase plane with each plot containing all trajectories for a single set of initial conditions. Each of the three sets use different initial conditions: (1) all initial concentrations set to 0; (2) 20 molecules of cro RNA per cell only; (3) 50 molecules of cI RNA per cell only. The results of the simulations are shown in Figure 7.



(a) All initial concentrations set to 0



(b) 20 molecules of cro RNA per cell only



(c) 50 molecules of cI RNA per cell only

Figure 7: Simulating stochastic model with varying starting concentrations

The stochastic model introduces some randomness to the trajectories even when the initial concentration in our system remains the same. This helps account for the randomness that exists in chemical reactions. Even when all initial concentrations are set to 0, in Figure 7a, there are still some trajectories that manage to reach a stable lysogenic phase out of 20 simulations. This is a much better representation of what happens in the real world. Once we are in stable lysogeny we need a way to trigger the switch to lysis.

The switch from lysogeny to lysis is mediated by stress-induced degradation of the cI protein. We add a second degradation constant of the cl protein in our models which represents stress-induced degradation. We can make it the same as our previous degradation factor so that the stress-induced degradation rate is dependent on the concentration of cI_{pro}. The modified deterministic model with the new stress-induced degradation factor is shown in Figure 8.

```
cl_omega = 50; %cl translation rate
cl_Xp = 1.2; %cl protein degredation rate
cl_mu = 50; %cl transcription rate
cl_Xr = 1.2; %cl rna degredatation rate
cl_khalf = 10; % cl k_half
stress_factor = 3.453;%////////////////////////////////////// stress factor//////////////////////////////////////////////////////////

cro_Khalf = 10; %cro k_half
cro_omega = 50; %cro translation rate
cro_Xp = 0.8; % cro protein degredation rate
cro_mu = 50; % cro transcription rate
cro_Xr = 0.8; %cro rna degredation rate
time = 40; %total run time of the simulation
time_step = .01;
num_steps = time/time_step; %num of steps taken calculation from total time divided by the time step

%intializing arrays to hold all concentrations and the time
cl_pro = zeros(1, num_steps);
cl_rna = zeros(1, num_steps);
cro_pro = zeros(1, num_steps);
cro_rna = zeros(1, num_steps);
t_values = zeros(1, num_steps);

%starting concentrations
cl_rna(1) = 50;
cl_pro(1) = 0;
cro_rna(1) = 0;
cro_pro(1) = 0;

for i = 1:num_steps-1
    t_values(i+1) = t_values(i) + time_step;     %calculating t of next step

    %calculating all rate of change of concentrations
    v1 = cl_omega * cl_rna(i);
    v2 = cl_Xp * cl_pro(i);
    v2_half = stress_factor * cl_pro(i);%////////////////////////////////////// degradation rate from stress//////////////////////////////////////////////////////////
    v3 = cl_mu * (1- cro_pro(i)^2/(cro_Khalf^2 + cro_pro(i)^2));
    v4 = cl_rna(i) * cl_Xr;
    v5 = cro_omega * cro_rna(i);
    v6 = cro_Xp * cro_pro(i);
    v7 = cro_mu * (1- cl_pro(i)^2 /(cl_khalf^2 + cl_pro(i)^2));
    v8 = cro_Xr * cro_rna(i);

    %Forward Eular implementation for calculating concentration at next step
    cl_pro(i+1) = cl_pro(i) + time_step * (v1 - (v2 + v2_half)); %////////////////////////////////////// concentration change from stress degradation ////////////////////
    cl_rna(i+1) = cl_rna(i) + time_step * (v3 - v4);

    cro_pro(i+1) = cro_pro(i) + time_step * (v5 - v6);
    cro_rna(i+1) = cro_rna(i) + time_step * (v7 - v8);
end
```

Figure 8: Deterministic model with stress factor

We set our initial condition as cro_{rna} at 50 molecules, while the rest are at 0. Through trial and error we find that a minimum of 3.453 stress factor is needed to switch from stable lysogeny to lysis pathway. The result is shown in Figure 9.

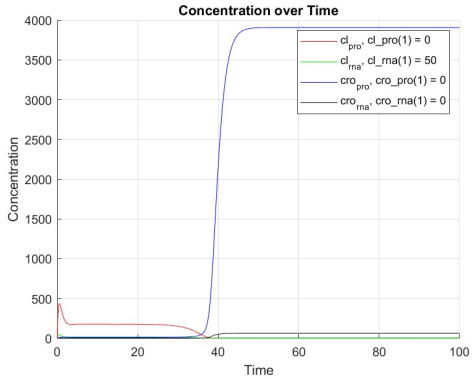


Figure 9: Stress factor = 3.453, initial cI RNA at 50 molecules

In the Stochastic model, we add the stress factor as the ninth reaction that can occur and adjust the code accordingly. We run the stochastic model with the same stress factor and initial condition we used for our deterministic model. The result is shown in Figure 10.

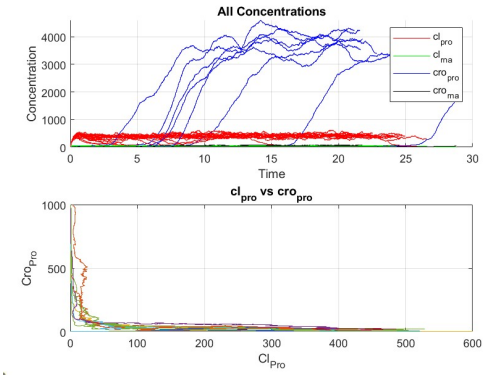


Figure 10: Stress factor = 3.453, initial cI RNA at 50 molecules

In the deterministic model, it takes almost 40 seconds before the switch from lysogeny to lysis. In the deterministic model we had 8 simulation out of 20 switch to lysis. If the simulation ran longer, we likely would eventually have all simulations reach lysis.

The Stochastic model accounts for random fluctuation in molecule concentrations and reaction rates. This leads to a range of possible outcomes from the same starting position which better reflects the inherent random-

ness that exists in chemical reactions unlike the deterministic model where we always get the same trajectory each time from the same starting position. However, the stochastic model still has some downsides. The range of outcomes makes it difficult to define clear patterns or thresholds. As seen in Figure 10, the range of time at which the simulations switch to lysis is too wide to detect any meaningful threshold or range of values. The predictable nature of the deterministic model provides allows us to easily find thresholds and ranges. This makes it easy to understand what conditions lead to lysis and what conditions lead to lysogeny.

Conclusion

The deterministic and stochastic models provide different advantages over each other. The deterministic model gives a predictable method of calculating thresholds for parameters. It gave clear thresholds for initial RNA concentrations to achieve lysis or lysogeny and also stress factor threshold. In contrast, the stochastic model representations the accounts for probabilistic interactions which provide a more realistic representation of the system. Due to these distinct advantages, both these models should be used together to make the most accurate model possible.