Modelling Simple Genetic Circuit Gates with Differential Equations
CPSC-791

Galen Michael Seilis
[redacted]

November 7, 2021

0.1 Introduction

This report practices the use of differential equations to model logic gates in genetic circuits. Genetic circuits are a type of biological circuit or network, and describes the way in which genes interact to each other and react to external stimuli [1]. Naturally-occurring regulatory networks have been of considerable interest to scientists such as molecular biologists and microbiologists, but also to industries such as pharmacology to provide improved treatements to ailments and disease [2, 3].

0.1.1 Why Not Use Fundamental Physics?

Biological systems such as bacterial cells contain a finite number of components that interact according to complex patterns. Reductionistically, the fundamental 'language' of biological systems is Physics. One might wish to use the standard model of particle physics to exactly describe a biological cell in complete detail, however in practice such an exercise is impossible. This impossibility can be attributed to at least two aspects that come about when working with Schrödinger's equation. Schrödinger's equation is a linear partial differential equation that describes the wave function for a given system, which is given below in its time-dependent and non-relativistic form as

$$i\hbar\frac{\partial}{\partial t}\left|\Psi(t)\right\rangle = \hat{H}\left|\Psi(t)\right\rangle \tag{1}$$

where i is the imaginary unit, $\hbar$ is the reduced Planck's constant[1], $|\Psi(t)\rangle$ is the vector of spatial components of a given time-dependent wavefunction $\Psi(t)$, and $\hat{H}$ is the Hamiltonian which is a differential operator that describes the total energy of a system.

The wave function is a complex-valued function over all the possible states of an isolated system of particles whose absolute square, according to Born's rule, gives the probability of a given configuration of the system [4]. Solving Schrödinger's equation in multibody problems with a realistic number of particles is computationally infeasible, even with numerical approximations such as Euler's method and the finite element method [5].[2][3] Insufficient knowledge of the initial conditions for any given cell is also realistic, but also leads to free parameters [5]. It is therefore necessary to use less fundamental models that still capture important aspects of the structure and dynamics of organisms.

[1] $\hbar = \frac{h}{2\pi}$ where h is Planck's constant and π is the familiar circle constant [4]. Planck's constant itself is a proportionality constant between a photon's energy and frequency.

[2] Euler's method are an approach to numerically approximating the solution to ordinary differential equations given some initial value [5].

[3] The finite element method is an approach to numerically solving partial differential equations in which solutions are calculating along a mesh (often a tesselation) of the space rather than for every point in a dense grid [5]. Generally more vertices are included where the derivatives are expected to be non-uniform [5].

0.1.2 Chemical Models

Instead of using a detailed description of individual particles, one can abstract to chemical concepts such as concentration of a chemical species [6]. This choice exchanges the detail of the descriptions and predictions we can make with Schrödinger's equation for computational feasibility and (often) measurability. By describing aggregate properties of populations of particles we both lose information about the system we're modelling, but gain tractability in solving problems.

In general an operon will have (1) a promoter region where transcription factors can bind to it, (2) a start codon in which RNA polymerase can bind to initiate transcription at the start codon, (3) at least one coding region (sometime more), and (4) a termination codon that indicate where to stop transcription [7]. Similar in concept to how Schrödinger's equation had a rate of change for how a quantum system evolved, we desire models for how concentrations evolve. In this report we will find rate equations that represent protein concentration $[P]$ as a function of transcription factor concentration $[T]$, and sometimes explicitly in terms of time. At the start of a given gene is a promoter region for which a transcription factor binds to enable RNA polymerase to initiate transcription [7]. Transcription leads to an mRNA product, which in turn leads translation into protein [7]. We will not model all of these steps in detail, however toy examples will be given that represent logic gates.

One important detail that will be included in the models for this report are Hill equations, which will be incorporated into the rate equations. The reason for this is ensure a sigmoidal (s-shaped) response in protein concentration to the concentration of transcription factor, which can serve as a continuous transition function from an 'off' state to an 'on' state (or *vice versa*). The Hill equation is given in Equation 2 where θ is the fraction of transcription factor bound to the promoter, K_A is a constant representing the transcription factor concentration when half of the transcription factors are bound, $[T]$ is the concentration of transcription factor that is unbound (free in solution), and h is the Hill coefficient that represents the equivalent 'sensitivity' of the binding interaction [7].

$$\theta = \frac{1}{1 + \left(\frac{K_A}{[T]}\right)^h} \tag{2}$$

When $h > 1$ it is interpreted that initially bound transcription factors would make it easier for additional transcription factors to bind, and when $h < 1$ it is interpreted that initially bound transcription factors decreases the affinity of additional transcription factors [7]. When $h = 1$ it is inferred that the binding of one transcription factor does not influence the binding affinity of additional transcription factors [7].

In the general case it is possible for copy number variation to occur, which is the duplication of a sequence of DNA within the genome of an organism [8]. While gene duplication can occur, and is a common step in the molecular evolution of genes, most repeats are non-coding regions [8]. There is often only one copy of a given gene in a genome, and each gene

generally only has one promoter [7]. Therefore we will assume that $h = 1$ for our purposes, since it is often the case that only one transcription factor can be bound at a time.

0.2 Modelling Gates

Assuming an idealized chemical environment in which all required reactants are available in solution, steady-state, and no interfering chemical species, we will model logic gates using differential equations in this section.

0.2.1 Unary Gates

In zeroeth order logic there are two unary operations: logical true, logical true, logical identity, and logical negation. Table 1 gives a summary of the outputs of these unary functions when given the input proposition p.

Table 1: Truth Table of Unary Logic Functions.

p	Logical True	Logical False	Logical Identity	Logical Negation
True	True	False	True	False
False	True	False	False	True

The logical true and logical false functions are constant functions in that their output does not ever change when the input changes. If True represents *up-regulated* and False represents *down-regulated* protein production, then they will both have the following rate equation.

$$\frac{d[P]}{d[T]} = 0$$

This differential equation posits that the rate of change in the protein concentration with respect to the free transcription factor concentration is zero, and therefore the aforementioned protein concentration is constant with respect to the transcription factor. The solution for the concentration of protein at any concentration of transcription factor can then be given as the following equation.

$$[P] = C$$

where C is a non-zero constant. Note that C does not need to be equal to be zero to represent a down-regulation of protein in general, but is an intuitive possibility. For a meaningful nothing of 'up'-regulation vs 'down'-regulation there must be (approximately) two modes that the system exists in. With the logical true and logical false there is only one state, and therefore the above rate equation does not represent a regulatory mechanism.

Yes Gate

The logical identity function can be thought of as a 'yes' gate. If the input is increased, the output is increased. If the input is decreased, then the output is decreased. For simple gates we should aim to have functions with the same monotonicity in the solution to the rate equations as found in the logic function itself. For the identity gate the rate equation can be posited as

$$\frac{d[P]}{d[T]} = \frac{\alpha}{1 + \left(\frac{K_A}{[T]}\right)} - \beta[P] \tag{3}$$

where $\frac{1}{1+\left(\frac{K_A}{[T]}\right)}$ represents the proportion of bound transcription factor, and α represents a rate constant indicating the maximum possible synthesis rate. The term $-\beta[P]$ represents the background degradation rate of the protein product, with β being a specific rate constant associated with this degradation. Equation 3 can be arranged into the following integral solution.

$$[P] = \frac{\alpha}{\beta}\frac{[T]}{K_A + [T]} \tag{4}$$

Equation 4 indicates that the proposed yes gate has solutions. The expression $\frac{[T]}{K_A+[T]}$ is equivalent to $\frac{1}{1+\left(\frac{K_A}{[T]}\right)}$, which represents the ratio of free transcription factor. The ratio alone would have an image on $(0, 1)$, but it is additionally scaled by a factor of $\frac{\alpha}{\beta}$. This ratio of α-to-β represents the competing processes of synthesis and degradation. Thus the ratio $\frac{\alpha}{\beta}$ decides the equilibrium concentration of the protein product. We will now qualitatively examine this differential equation.

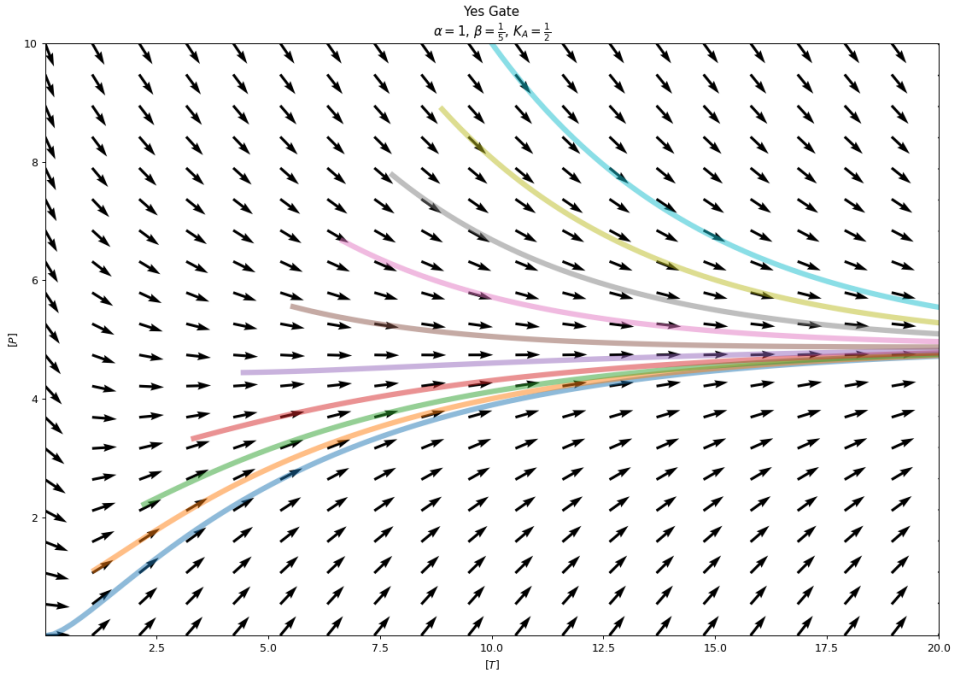

Figure 1: Phase portrait of the yes gate model hypothesized in Equation 3. The vertical axis $[P]$ is the concentration of protein product. The horizontal axis $[T]$ is the concentration of free transcription factor. The arrows arranged in a lattice embedded in the plane represent the slope $\frac{d[P]}{d[T]}$ for each coordinate $([T], [P])$. The coloured lines represent trajectories from initial conditions with different starting concentrations of transcription factor in free solution and protein concentration. The rate constants were chosen to be $\alpha = 1$ and $\beta = \frac{1}{5}$. The binding half-occupation constant was chosen to $K_A = \frac{1}{2}$.

Figure 1 illustrates a phase portrait of the dynamics between the protein product concentration and the free transcription factor concentration for a choice of rate constants $\alpha = 1$ and $\beta = \frac{1}{5}$. Implicit in this visualization is the effect of time, which parametrizes the changes in the free transcription factor concentration. Here the concentration of transcription factor is assumed to be constantly increasing. The main feature of this visualization is the convergent behaviour of the protein product concentration as the transcription factor concentration increases. Where this converged value sits depends upon the chosen rate constants, so the particular value that it takes. Without the presence of transcription factor, the protein product concentrations plummets due to its exponential-like decay, thus matching the notion that the output is zero when the input is zero. With transcription factor present, the protein concentration may initially increase or decrease. However, it only decreases when the initial protein concentration was already greater than the steady-state concentration. If the initial protein product concentration is initially lower than the equilibrium concentration, then it will increase.

Not Gate

The logical negation function is also simply referred to as the *not gate*. The pattern of the equations we are looking for should have an opposite pattern to the yes gate. When the input is high, the output should be low. When the input is low, then the output should be high. Instead of considering a system that must be induced to be activated, we can consider an operon that is always active unless repressed. This negative repressible system could hypothetically follow Equation 5

$$\frac{d[P]}{d[R]} = \frac{\alpha}{1 + \left(\frac{K_A}{[T]}\right)} - \frac{\gamma}{1 + \left(\frac{K_R}{[R]}\right)} - \beta[P] \tag{5}$$

where the $[P]$ is the protein product concentration, $[R]$ is the concentration of the repressor that prevents transcription of the gene that codes for P, α represents the maximum rate of synthesis of P that depends on the binding of the transcription factor concentration $[T]$, K_A is the half-occupation constant between the transcription factor and the gene's promoter region, γ is the maximum rate of inhibition of gene transcription due to the binding of the repressor protein R, $[R]$ is the concentration of free repressor protein, K_R is the half-occupation constant between the repressor protein and its recognition site, and β is a rate constant for the background rate of degradation of the protein product P in the chemical environment. As a simplifying assumption, let $[T]$ be a constant background set by exogenous variables. The promoter site and the repressor site are physically separated, and which is represented here by the binding interactions of the transcription factor and repressor being additively separated The solution to Equation 5 is given in Equation 6.

$$[P] = \left(C_1 - \frac{\int \frac{K_A[R]\gamma e^{[R]\beta}}{K_R+[R]}\, d[R] + \int \left(-\frac{K_R[T]\alpha e^{[R]\beta}}{K_R+[R]}\right) d[R] + \int \left(-\frac{[R][T]\alpha e^{[R]\beta}}{K_R+[R]}\right) d[R] + \int \frac{[R][T]\gamma e^{[R]\beta}}{K_R+[R]}\, d[R]}{K_A + [T]} \right) e^{-[R]\beta} \tag{6}$$

Equation 6 is rather complicated, especially compared to Equation 4 which was the solution to the Equation 3 that represented the yes gate. Thus it is even more beneficial to consider a qualitative approach to checking the behaviour of this system.

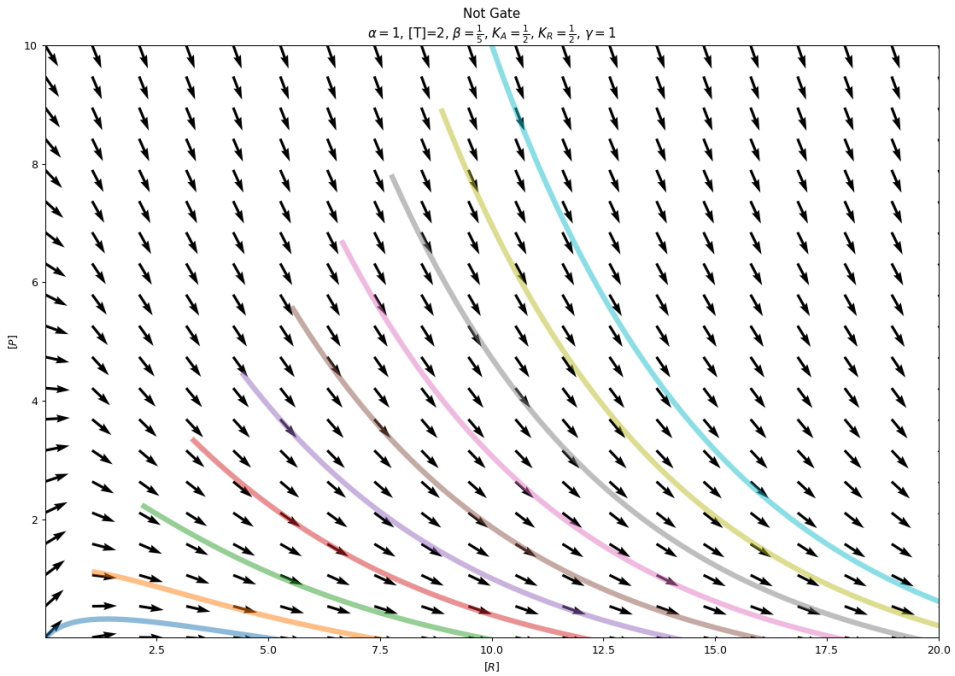

Figure 2: Phase portrait of the not gate model hypothesized in Equation 3. The vertical axis $[P]$ is the concentration of protein product. The horizontal axis $[R]$ is the concentration of repressor protein. The arrows arranged in a lattice embedded in the plane represent the slope $\frac{d[P]}{d[R]}$ for each coordinate $([T], [R])$. The coloured lines represent trajectories from initial conditions with different starting concentrations of repressor protein and protein product coon. The rate constants were chosen to be $\alpha = 1$, $\beta = \frac{1}{5}$, and $\gamma = 1$. The binding half-occupation constants was chosen to $K_A = \frac{1}{2}$ and $K_R = \frac{1}{2}$ respectively. The background transcription factor concentration was set to $[T] = 2$.

As desired, Figure 2 illustrates that the rate equation given in Equation 5 yields a system where the repressor concentration shuts down production of the protein product even in the presence of background levels of transcription factor. It is also apparent from the slope field that when the repressor concentration is near zero that a similar pattern to the yes gate is observed. That is to say, when the repressor concentration is zero and the protein product concentration is initially low, the production of the protein product actually increases.

0.2.2 Binary Gates

While the unary gates consisted of a collection of only 4 functions, there are 16 possible logic functions for binary gates. We do not need to model all of them for a couple of different reasons.

Similar to the unary case, there are two gates that simply represent returning true regardless of input and returning false regardless of input. Such constant functions are again

modelled with rate equations equal to zero, and do not show interesting dynamics. Two additional functions are identity functions for one of the inputs, and another two functions are negation functions of one of the inputs. These six gates do not generalize what has already been covered in the last section in any interesting way.

Given a fixed number of inputs, there is only a finite number of possible functions; 16 in this case. Thus function compositions of any of these functions will result in one of these 16 functions. It has been of interest to computer scientists, logicians, and mathematicians to find a minimal set of functions whose compositions can produce all of the other functions. The property of such a collection is called *functional completeness*. Two such collections of binary functions are {NAND} and {NOR} [9]. It is therefore sufficient to show that one of these two gates can be modelling in principle in order to demonstrate that all the others can be modelled in principle via compositions of functions and the functional completeness property.

NAND Gate

A NAND gate is a binary logic gate that represents the negation of a conjunction of two inputs. If both inputs are 'on', then the output is off. Otherwise the output is on.

Table 2: Truth table for the NAND function with inputs A, B and output C.

A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

From Table 2 we can infer that the inputs could be modelled as repressors because their activation should deactivate the output. Table 7 also shows that NAND is a non-additive function, and that as a weaker inference we should hypothesize a rate equation which is not an additive function of the effect of the repressor inputs. In a physical system, the binding of both repressors (possibly one on top of another in a specific order) would be required for the gene be deactivated.

We acheive this in Equation 7 in which the complement of the product of the effects of the repressors modelling in terms of Hill functions. Additionally, a protein product degradation term is included.

$$\frac{d[P](t,[R_1],[R_2])}{dt} = \left(1 - \left[\frac{\alpha}{1+\frac{K_{A_1}}{[R_1](t)}}\right]\left[\frac{\gamma}{1+\frac{K_{A_2}}{[R_2](t)}}\right]\right) - \beta[P](t,[R_1],[R_2]) \tag{7}$$

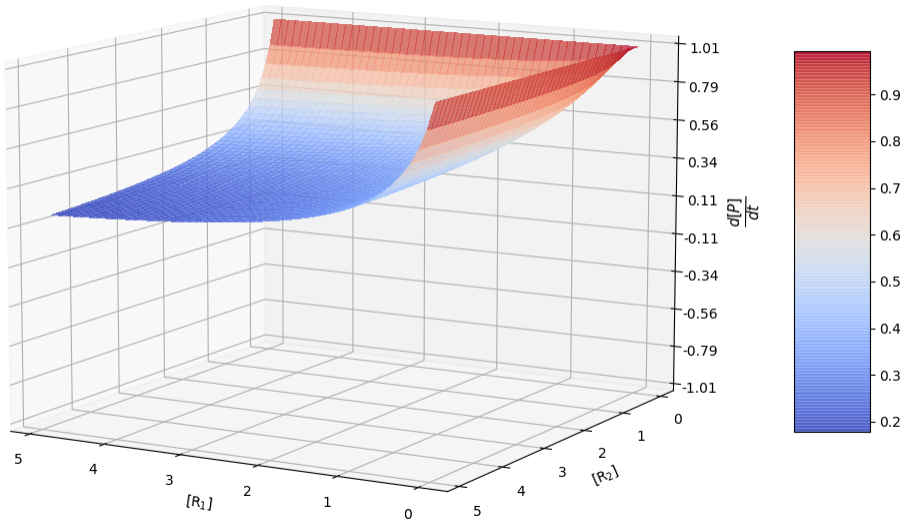

Figure 3: The surface $\frac{d[P]}{dt}$ resulting from different inputs of $[R_1]$ and $[R_2]$ in Equation 7. The rate constants were assigned to be $\alpha = 1$, $\beta = 1$, and $\gamma = 1$. The protein concentration product was held constant at $[P] = 0$. The half-occupation constants were assigned as $K_{A_1} = \frac{1}{2}$ and $K_{A_2} = \frac{1}{2}$. Take care to note the right-to-left orientation of the $[R_1]$ axis in comparison to the left-to-right orientation of the $[R_2]$ axis that results from the chosen projection of the surface model.

Note that in previous gates we have considered the rate equation $\frac{d[P]}{d[T]}$ or $\frac{d[P]}{d[R]}$ because we simplified our problem by assuming that only one independent variable was changing. With two inputs, this assumption is less reasonable as the inputs may vary simultaneously. Thus it is appropriate to parametrize by time by assuming that $[P]$ is function of time and the current repressor concentrations, but that the current repressor concentrations are also possibly functions of time.

In order to get a better sense of the rate equation given in Equation 7, we turn our attention to Figure 3 which illustrates the rate equation as a surface over the pairs of repressor concentrations. We can see that near $([R_1], [R_2]) = (4, 4)$ that the rate $\frac{d[P]}{dt}$ is nearly zero, which is compatible with the intuition drawn from Table 2 that relatively large inputs should shut down the production of protein. In similar fashion, we see that any of $(0, 4)$, $(4, 0)$ or $(0, 0)$ produce relatively larger amounts of protein per unit time, which also agrees with being a continous approximation of Table 2. Furthermore, the surface over the bounded region within the aforementioned points appears continuous and concave up, giving some indication that this function is 'nicely behaved' for the purposes of modelling.

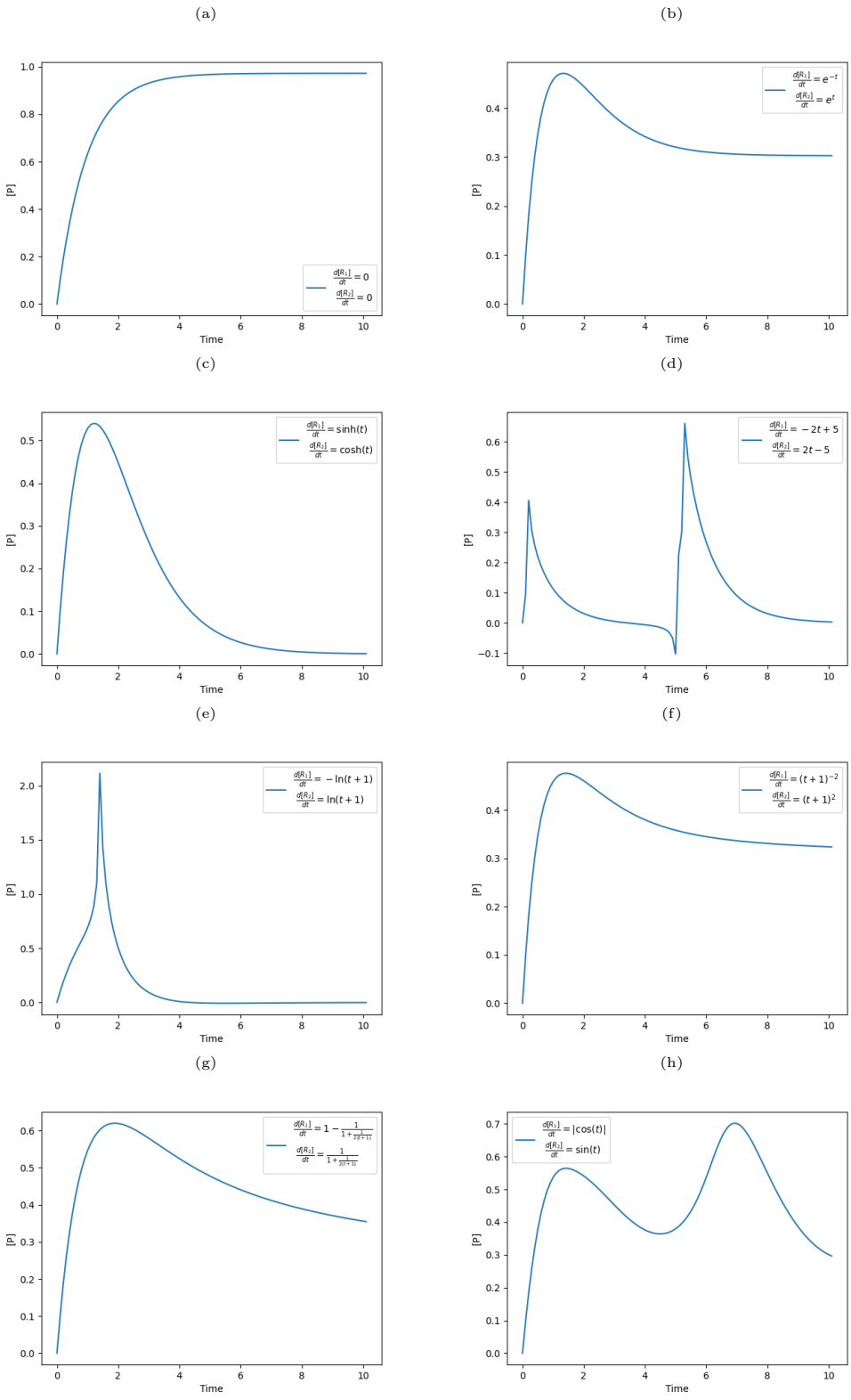

Figure 4: Example solutions to Equation 7 specifying the dependence of $[R_1]$ and $[R_2]$ on time. Different families of functions were selected for illustration: (a) constant, (b) exponential, (c) hyperbolic, (d) linear, (e) logarithmic, (f) rational (non-linear), (g) sigmoidal, and (h) sinusoidal. For all panels, the initial conditions were $[R_1]_0 = [R_2]_0 = 0.1$, the half-occupation constants were $K_{A_1} = K_{A_2} = \frac{1}{2}$, the rate constants were $\alpha = \beta = \gamma = 1$, and the initial protein product concentration was $[P]_0 = 0$. See Appendix for enlarged images.

Equation 7 does not tell us how $[R_1]$ and $[R_2]$ are dependent on time, so these relationships must be specified by the real-world system. Since the system under consideration is hypothetical, it is informative to consider how a selection of functions behave. Figure 4 demonstrates how different families of functions representing the time-dependence relations of $[R_1]$ and $[R_2]$ yield different time-dependence curves for the protein product concentration $[P]$.

The first property that Figure 4 shows is that the solutions are not necessarily monotonic. In fact, all of the panels except panel (a) are non-monotonic. Panel (a) illustrates the time dependence of $[P]$ given that the repressor concentrations did not change from their initial value of $[R_1]_0 = [R_2]_0 = 0$.

The second property of note is non-smoothness in some of the solutions. Specifically panels (d) and (e) had two or one sharp peaks respectively, which imposes both mathematical limitations and questions of physical possibility. While chemical reactions at the scale of one reaction are discrete events that *might* match a discontinuous change in a solution to a differential equation, the feasibility of such discontinuities should be questioned when compared to real data and a knowledge base.

A third property should be the satisficing of the boundary conditions, which is generally considered to be a *necessary* condition for a candidate solution to be a solution. In Figure 4 all panels except for panel (d) meet this criteria for being a solution. Around Time = 5 the value of $[P]$ drops below zero, which is an impossibility by virtue of the way that chemical concentrations are defined as the ratio of the moles of compound (a counting measure) diveded by the volume (a continuous measure).[4] If the relations in panel (d) really were the best-fitting relations notwithstanding the boundary conditions, then further changes in the numerical methods can be used to enforce that the boundary conditions are satisfied while using these chosen relations.

0.3 Limitations

There are limitations to the modelling that we performed in this report.

The largest limitation of these models is that they are hypothetical, rather than empirically verified. While the chosen differential equations had the desired behaviour of the gates they were intended to represent, a given collection of real molecules may not match these models. The particular chemical and physical processes can be different than those specified in the model. For instance, not all binding interactions follow the Hill equation. Similarly, our models did not explain where the original transcription factors and repressors came from, and were treated as entirely exogenous independent variables for most of our hypotheticals.

[4]Measure theory is beyond the scope of this report. It pertains to certain functions on sets that quantify their 'size' in some sense. See *A Concise Introduction to Measure Theory* by Satish Shirali for more information, but one of the notable properties of a measure μ on a set S is that $\mu(S) \geq 0$.

Another example involves a cell's production of proteases (enzymes that break down proteins) that lead to the degradation rate of the protein product being faster than the direct proportionality hypothesis used above. Indeed, every parameter in the models of this report could themselves be replaced with functions.

For a real-world system the initial conditions, rate constants and half-occupation constants are experimentally known or estimated through parametrization. The rate constants cannot be arbitrary tweaked, but must rather be chosen implicitly by the choice of macromolecules that undergo chemical reactions and physical interactions.

Effective modelling of biological systems must yield models that are predictive of the data, and explainable in terms of theory. Since this project used no real-world data, they can be considered incomplete at best.

0.4 Conclusion

In summary, differential equations are useful in modelling systems in which the rates of change are understood either with or without the direct functional relations between variables. Sometimes analytical solutions are possible, but often they are not for realistic systems.

We began with discussing how fundamental physical principles are unsuitable in practice, and how the use of chemical principles could motivate rate equations instead. To introduce some differential equations in detail, some unary gates were discussed and modelled. Lastly the NAND gate was used as a more complicated example in which the solution of the main equation of interest depending on how multiple independent variables were changing.

0.5 Learning Outcomes

In this project I got hands-on experience with using differential equations to model systems. I avoided using state-of-the-art solutions partly due to their complexity, and also to put the challenge on myself to figure out how to match rate equations to gates using my own intellect. This project also gave me extensive experience in learning and practicing tools for visualizing and solving differential equations either symbolically (SymPy) or numerically (py-pde). While I am still a novice at modelling with differential equations, I am finishing this project more concepts and tools to quickly make progress on future modelling projects.

Bibliography

[1] Jennifer A N Brophy and Christopher A Voigt. Principles of genetic circuit design. *Nature Methods*, 11(5):508–520, April 2014.

[2] T. I. Lee. Transcriptional regulatory networks in saccharomyces cerevisiae. *Science*, 298(5594):799–804, October 2002.

[3] Yixuan Xie, Yanfang Yang, Yu He, Xixi Wang, Peng Zhang, Haocheng Li, and Shufang Liang. Synthetic biology speeds up drug target discovery. *Frontiers in Pharmacology*, 11, February 2020.

[4] David J. Griffiths and Darrell F. Schroeter. *Introduction to quantum mechanics*. Cambridge University Press, Cambridge ; New York, NY, third edition edition, 2018.

[5] K. W. Morton and D. F. Mayers. *Numerical solution of partial differential equations*. Cambridge University Press, Cambridge ; New York, 2nd ed edition, 2005.

[6] Daniel D. Seaton. ODE-based modeling of complex regulatory circuits. In *Methods in Molecular Biology*, pages 317–330. Springer New York, 2017.

[7] David L. Nelson, Michael M. Cox, and Albert L. Lehninger. *Lehninger principles of biochemistry*. W.H. Freeman and Company ; Macmillan Higher Education, New York, NY : Houndmills, Basingstoke, seventh edition edition, 2017. OCLC: ocn986827885.

[8] Lindell Bromham. *An introduction to molecular evolution and phylogenetics*. Oxford University Press, Oxford, United Kingdom ; New York, NY, United States of America, second edition edition, 2016. OCLC: ocn920739196.

[9] Thomas W. Scharle. Axiomatization of propositional calculus with sheffer functors. *Notre Dame Journal of Formal Logic*, 6(3), January 1965.

Appendix: Enlarged Figures

Figure 4(a) constant

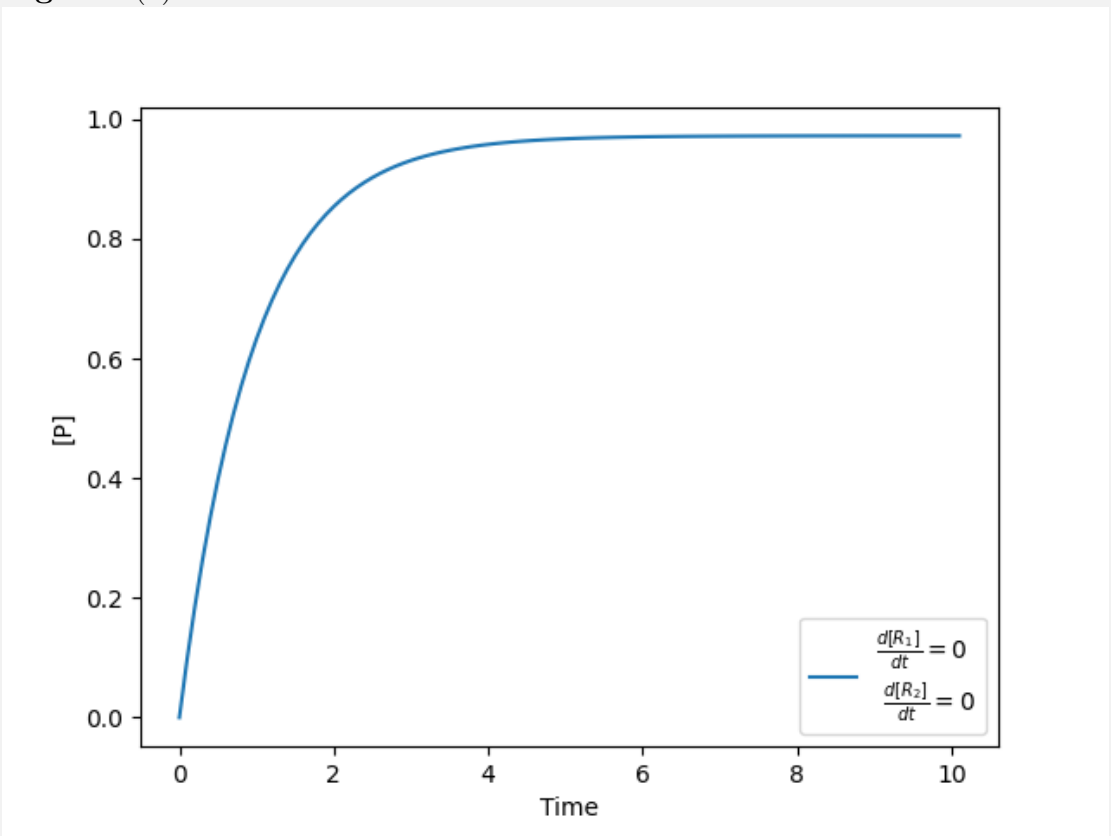

Figure 4(b) exponential

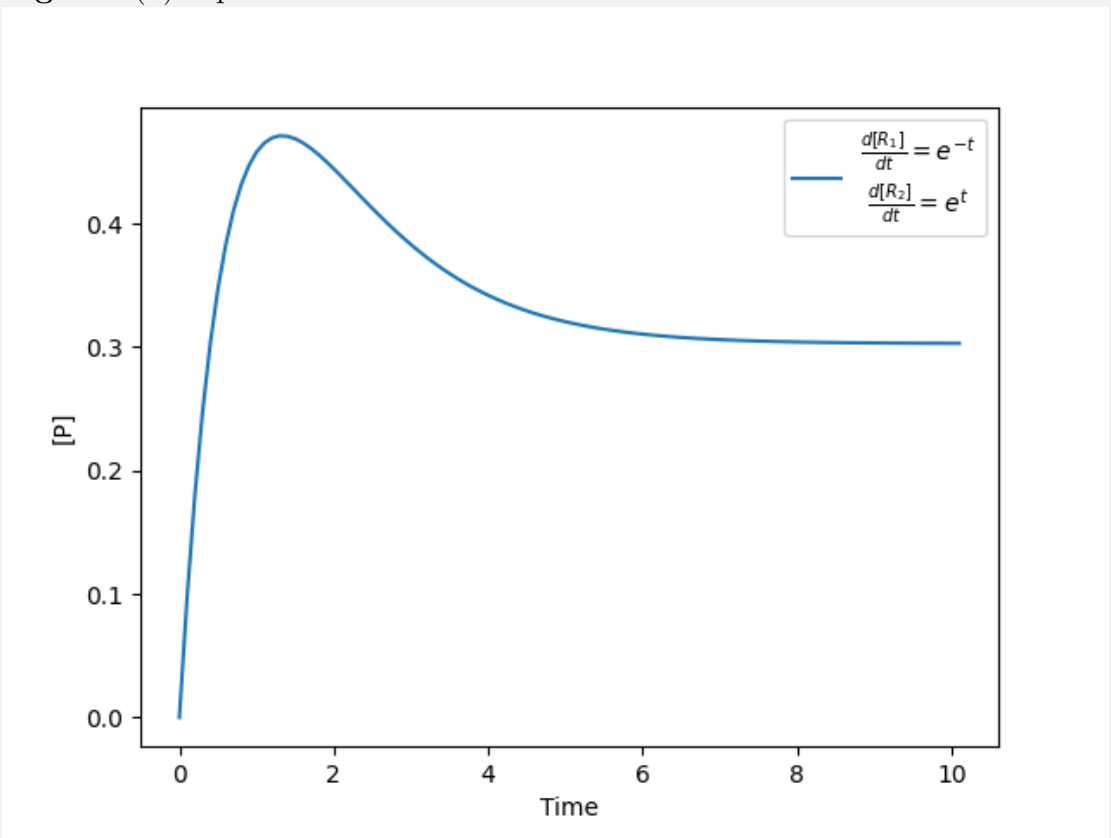

Figure 4(c) hyperbolic

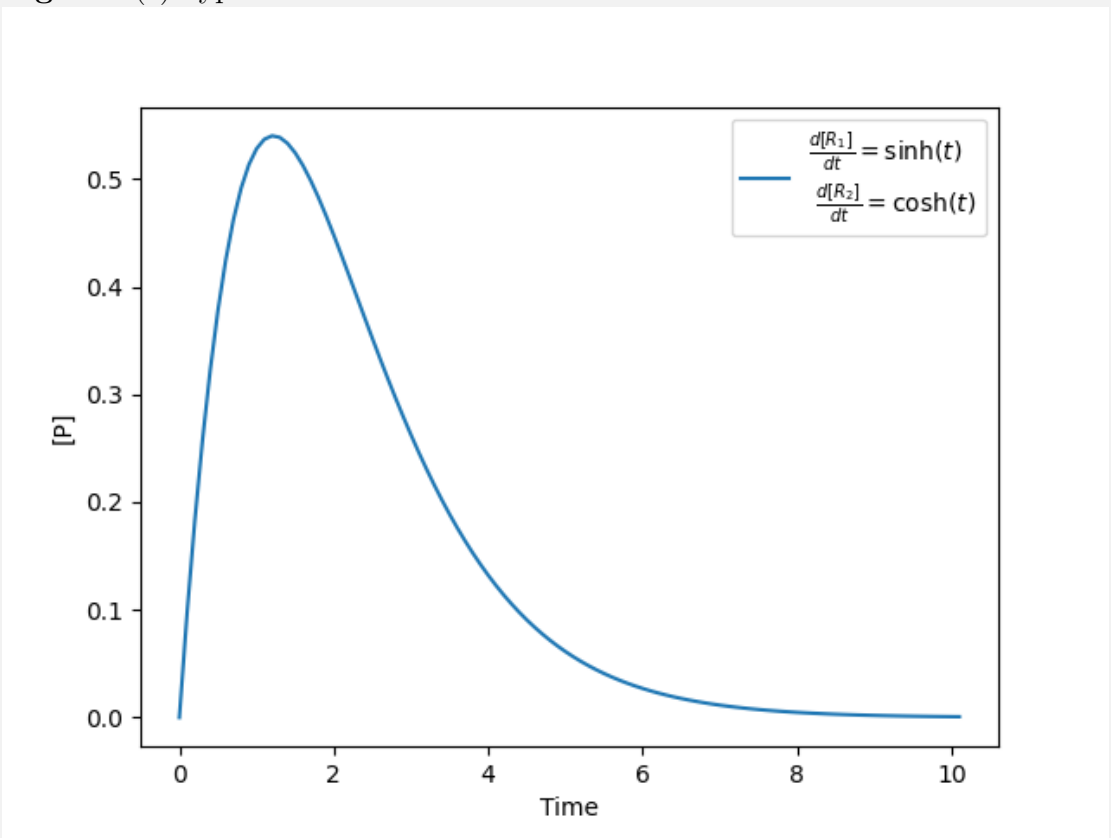

Figure 4(d) linear

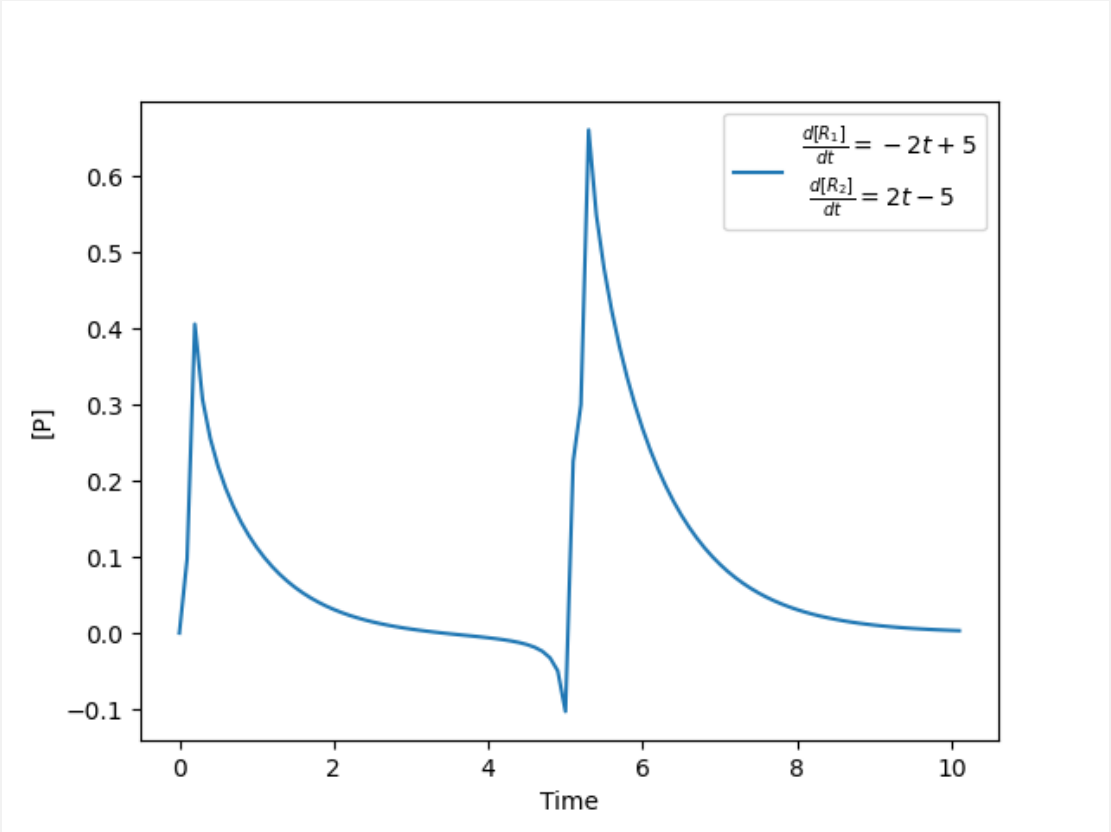

Figure 4(e) logarithmic

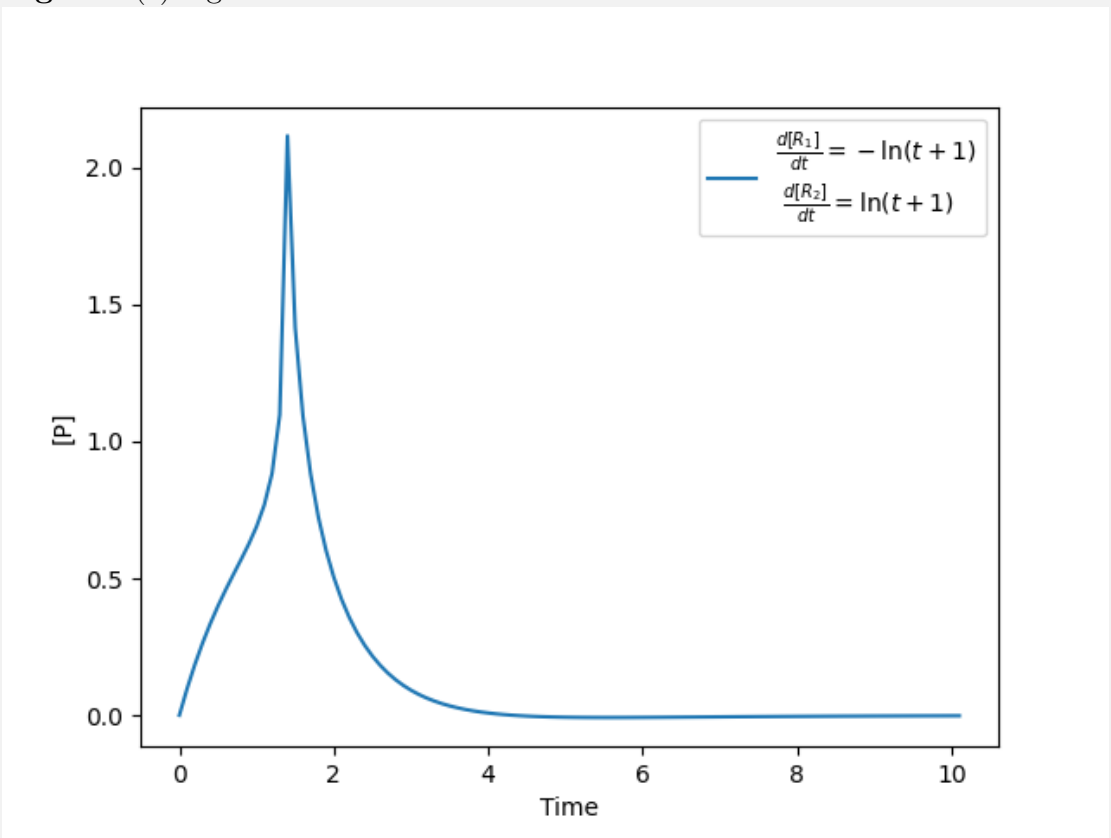

Figure 4(f) rational (non-linear)

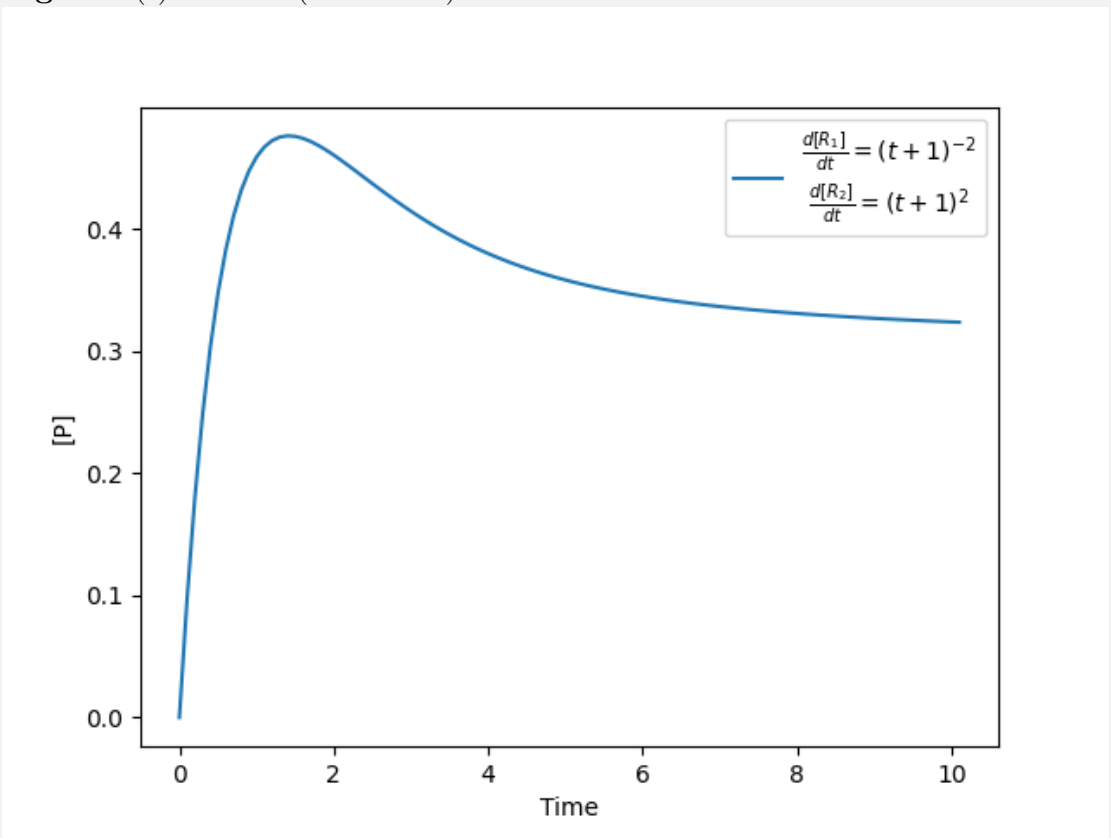

Figure 4(g) sigmoidal

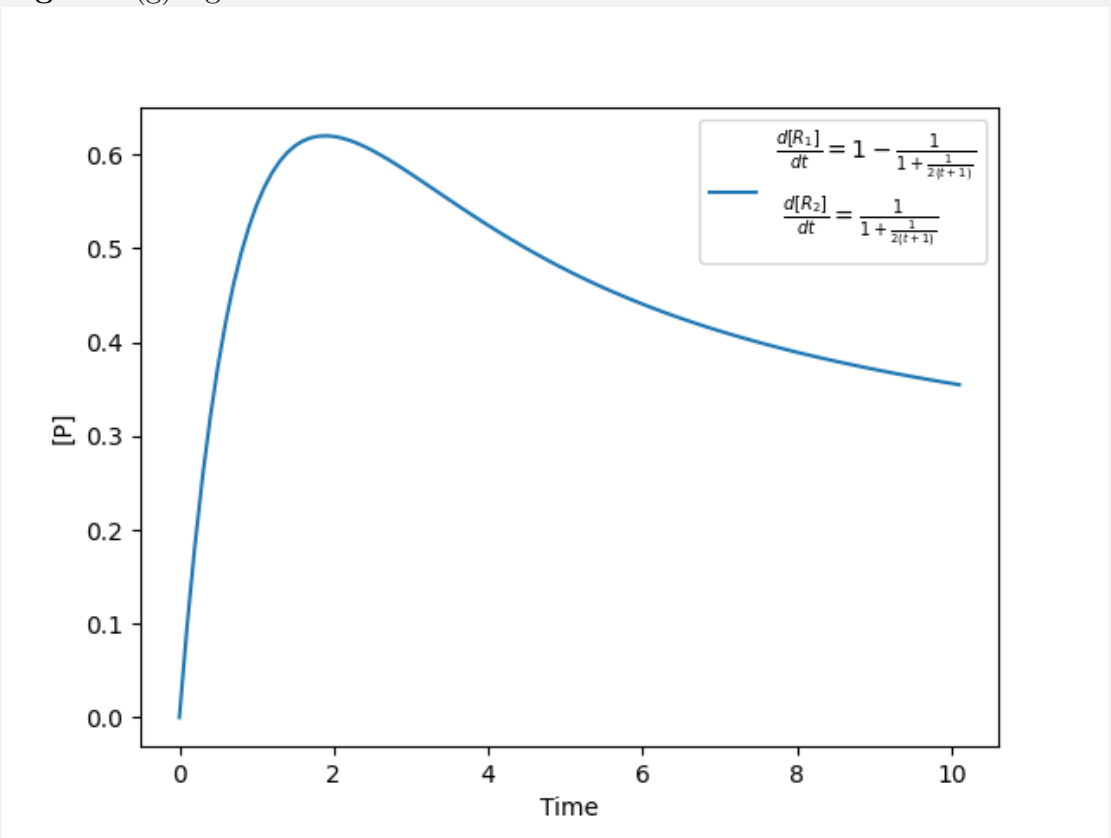

Figure 4(h) sinusoidal