A CASE STUDY OF TRANSCRIPTIONAL REGULATION IN BACTERIOPHAGE λ - INFECTED *ESCHERICHIA COLI* CELLS

GEERT WENES

Abstract. The goal of this Project is to introduce a framework to describe the stochastic and nonlinear processes of genetic pathway selection and to critically evaluate the validity of this framework by assessing its applicability to a model of the phage λ infection of *E.coli* cells. The framework is a Monte Carlo approach whereby the particles' trajectories through state space are simulated using appropriate probability distributions while the phage λ-infection model is a simplified three gene, three promotor regulatory network.

Introduction

Proteins are the workhorses of the cell; other than DNA or RNA, all the complex molecules in a cell are proteins. Highly specialized proteins fulfill their own tasks: from transporting oxygen, to facilitating specific biochemical reactions, to responding to extracellular signals, and many more. In particular, certain proteins bind directly or indirectly to DNA to perform transcriptional regulation, thus closing the circle of gene regulation whereby the information stored in DNA is transcribed to mRNA, followed by translation into proteins.

One of the key questions in gene regulation is: what genes are expressed in a given cell at a certain time under which conditions and how does this differ from cell to cell?

One of the many challenges in trying to answer this question is the construction of a modelling framework for transcriptional regulation that meets certain minimal requirements: it is generally and widely applicable, it can be validated experimentally, its system-level behavior can be understood in terms of (few) biochemical parameters, and it needs to have the built-in capability to describe a particularly rich set of interesting stochastic phenomena encountered in gene regulation.

This is a modest attempt to lay out how such a framework could look like.

1. Bacteriophage λ Infection of E. Coli Cells

1.1. **Introduction.** Phage λ is a virus, i.e. a capsule of genetic material that relies upon the transcription machinery (either used intact or modified to fit the virus' needs) of the host cell to express its genes. As such, the study of either initiation and subsequent growth or repression of bacteriophage development has provided a wealth of information [3] on the mechanisms of transcription and regulation, especially in prokaryotes. Phage λ in particular has had its complete genome (50

Received by the editors August 11, 2001.
Key words and phrases. Genetic Networks, Transcriptional Regulation, Stochastic Models .
This work was supported financially by the IBM Tuition Reimbursement Program.

kbp) sequenced some time ago. Its host, *E. coli*, has been studied extensively as well.

Phage λ looks a bit like a prop from a low-budget 1950*s* sci-fi movie: the virus has a "head" domain, which holds the packaged DNA, and a "tail" domain which helps bind the virus to the surface of the bacterial cell and serves as a conduit for the injection of the virus' DNA into the host. Upon injection, λ uses the host's enzymes, such as RNA polymerase (RNAp), for certain functions. In addition, λ does not encode its own ribosomes or DNA polymerase either. All in all, λ's DNA encodes for about 40 different genes, a substantial part of which (21)are structural and dedicated to the construction and maintenance of the tail and head of the virus.

λ is a temperate phage, i.e. it does not always kill its host. When a phage injects its DNA into the *E. coli* host one of two things can occur: Either the phage replicates (releasing its progeny) and in the process of doing so destroys its host. This is known as *lysis*, a cycle of growth. Or the phage integrates its DNA into the host's DNA. This is known as *lysogeny*, in which the viral chromosome circularizes and undergoes site-specific integration into the host-cell chromosome. Upon lysogeny the host acquires immunity from further infection for many generations. Under the proper conditions–for instance, treatment of the cell with *UV* light–lysogeny can be broken with subsequent induction of lytic growth.

1.2. **A Simplified Model for the Transcriptional Regulatory Network.** In a lysogenized cell, the only viral protein expressed in quantity over time is cI (a.k.a. λ repressor). cI represses transcription of lytic genes. However, the open complex (*i.e.*, in the absence of gene products) for *cI* has a very low basal activity while other proteins–with higher open-complex activation rates–such as Cro, are negative regulators of *cI* thus favoring lytic growth. In addition, their promoters, P_{RM} and P_R, are controlled by the concentration-dependent logic of a shared three-operator site. (This complex can be shared because *cI* and *Cro* genes are separated onto leftward, resp., rightward transcribed strands.) The whole set-up is usually referred to as the "λ switch" and makes for a precariously balanced system of early-stage gene production. [1] To complicate matters further, at least one other gene product, N, acts as an antiterminator protein allowing for regulated read-through of transcription terminators in certain genes, such as *cII* and *cIII* whose proteins are positive transcriptional activator and stabilizing proteins for *cI*. N itself, however, is repressed by cI and Cro through regulation of its own promoter, P_L. In this Project, we will restrict ourselves to these 3 genes and 3 promoters.

While the above model of phage λ-infection is anticipated to exhibit a rich set of gene production and pathway selection phenomena, it is still an oversimplification. More elaborate models [2] typically include many more of λ's genes–including some of *E.coli*'s genes–and an increased number of termination and antitermination sites as well as substantial non-genetic reaction subsystems.

1.3. **Formal Equations.** We start from the following reaction kinetics:

$$\dot{m}_{i,b} = K_m K_i A_i(p_j) - \frac{\lambda_i}{L_i} m_{i,b} \qquad i,j = Cro, N, cI \tag{1.1}$$

which describes the binding of RNA polymerase to the DNA and the initiation and elongation stages of the transcription process. In (1.1) K_m is the concentration of

[1] A typographic convention: *genes* will be in italic, the corresponding gene product in roman.

RNA polymerase (RNAp), assumed to be a constant $30nM$, A_i the (dimensionless) activation probability of the complex (a potentially fairly involved and non-linear function of the gene products p_j), K_i is the open complex reaction rate, while λ_i is the reaction rate for the elongation process, nucleotide by nucleotide, and L_i the transcription length. (We assume that once transcription is initiated, it runs to completion.) Taking into account the degradation of the free mRNA on its way to the ribosome due to ribonuclease (RNase) binding:

$$\dot{m}_i = \frac{\lambda_i}{L_i} m_{i,b} - (\tilde{\lambda}_i[ribo] + \hat{\lambda}_i[RNase])m_i \tag{1.2}$$

where [ ] are concentrations, we proceed to describe the translation process as

$$\dot{\tilde{m}}_i = \tilde{\lambda}_i[ribo]m_i - \frac{\beta_i}{\Gamma_i}\tilde{m} \tag{1.3}$$

where Γ_i is the translation length, β_i the translation rate per nucleotide. In turn, this leads to the gene product and protein degeneration or degradation process description:

$$\dot{p}_i = \frac{\beta_i}{\Gamma_i}\tilde{m}_i - k_i p_i \tag{1.4}$$

Here k_i is an overall reaction rate for the protein degeneration process(es) . No dimer production and/or higher order protein reactions (for instance, those involving cII and cIII) are considered.

The above set of equations are fully *deterministic* (in that they neglect the stochastic character of the interactions between the molecular components) and *continuous* (in that they neglect the discrete nature of the molecular components [2].)Stochastic effects are believed to be important in genetic networks [10] and are typically included as described [1]. We will discuss later on how to incorporate such effects in the framework.

This system of ODEs (1.1), (1.2), (1.3), (1.4) was solved using a 4^{th} order Runge-Kutta method with the parameters summarized in the Table below.

i	$K_i(s^{-1})$	$\lambda_i(s^{-1}n^{-1})$	$L_i(n)$	$\tilde{\lambda}_i[](s^{-1})$	$\hat{\lambda}_i[](s^{-1})$	$\beta(s^{-1}n^{-1})$	$\Gamma_i(n)$	$k_i(s^{-1})$
Cro	0.014	30	550	0.3	0.03	100	320	0.0025
N	0.0011	30	550	0.3	0.03	100	320	0.0023
cI	0.002 (est)	30	550	0.3	0.03	100	320	0.0007

Zero mean random noise was added to all state variables but not more than a 0.1 change in absolute magnitude in any variables was allowed.

We now turn to the activation function $A_i(p_j)$. Obviously, in the limit of no gene products ($p_i \to 0$), the activation function needs to match the open complex rates. Similarly, for fully repressed or activated promoters ($p_i \to \infty$) the activation probability needs to converge to either 0 or 1.

Perhaps the most fundamental way to calculate the activation level as a function of the concentration levels of several reactants and products occupying multiple operator sites comes from (equilibrium) thermodynamics in combination with the use of partition functions [7], [8]. However, such calculations have not always been

[2]For instance, per *E.coli* cell, we are dealing with one DNA, a few RNA molecules, and tens to hundreds of proteins.

carried out to the necessary extent since the number of equations can grow exponentially with the number of interdependent state variables. The latter complication also bedevils the next approach: the parametric approach.

Parametric solutions are then usually proposed where the parameters are the binding constants or probabilities of a gene product j to a specific operator site i (but correlated with the presence of other gene products k bound to other operator sites l) in the complex when the operator is active, resp., inactive. Needless to say, the number of parameters is $2(N+1)^M$ where N is the number of gene products involved (the extra 1 comes from the event that no gene product occupies an operator site) and M is the number of operator sites. For our simplified model, this would amount to 2×4^3, clearly an unrealistically high number to fit. Nevertheless, parametric methods enjoy a certain justifiable popularity with the practitioners of trainable (*e.g.*, learning the weights of a neural network) gene regulation networks. In particular, the Hierarchical Cooperative Activation (E. Mjolsness in [5]) (HCA) method which describes the activation of "promotor modules" in terms of transcription factor concentrations which then in their turn are described in terms of the corresponding promotor activations (i.e. $A_i(p_j) \equiv A_i(f_j(A_l))$ has an undeniable appeal to it.

1.4. **Discussion.** Despite the observation that the set of equations above is fully deterministic–which may not be appropriate for genetic networks –it is worthwhile to explore it a little further. The motivation for this is threefold: *(i)* it is possible to add a noise term to this set and continue to use the machinery of ODE solving techniques (see, for instance, the contribution of M. A. Gibson and E. Mjolsness in [5]) all the while keeping in mind that one now solves for expected values of the state variables and their variance [11], *(ii)* the ODE formulation has an intuitive appeal to it and typically "inspires" state space formulations such as those we will propose below, and *(iii)* a classical study of the system when–or if–protein production reaches the equilibrium state is particularly instructive in anticipation of further stochastic modelling. Indeed, in the equilibrium case, solving the system of ODEs simplifies to solving the set of algebraic equations:

$$k_i p_i = K_m K_i A_i(p_j) \tag{1.5}$$

or, by rescaling the protein concentrations:

$$\tilde{p}_i = (K_i/k_i)\tilde{A}_i(\tilde{p}_j) \tag{1.6}$$

where $\tilde{p} = p/K_m$. These equations can have either zero, or a finite number, or an infinite number of solutions. Stability for the equilibrium solution requires that in the vincinity of that solution:

$$\sum_j (\partial_j \tilde{A}_i - \frac{k_i}{K_i}\delta_{i,j})\dot{\tilde{p}}_i \geq 0 \tag{1.7}$$

where ∂_j is a partial derivative w.r.t. $\tilde{p}_j$.

In particular, this system was solved for the oscillatory network of transcriptional regulators [9] with cyclic repression, i.e. $\tilde{A}_i(\tilde{p}_j) = 1/(1+\tilde{p}_j^n)$, resulting in a unique steady state solution. This steady state can be either stable or unstable. In the unstable case, cyclic transcriptional feedback loops or oscillators were obtained. Such behavior was then further observed [9] in the "repressilator" constructed from the

transcriptional regulators LacI from *E.coli*, cI from λ, and tetR from the transposon *Tn10* [3].

Another case of some educational interest is a two gene model where the first gene is repressed by the gene product of the second but activated by its own gene product while the second gene is repressed by both genes' proteins. This is a simplification of the *cI, Cro* network where cI production is repressed by Cro but enhanced by itself (at least, at not too large a concentration) and Cro production is repressed by itself and cI. If it so happens that the activation level for *cI*'s and *Cro*'s promoter are well approximated by some simple polynomial functions, then the system of algebraic equations (1.6) becomes:

$$\tilde{p}_{cI} = \alpha_{cI}(\epsilon + \frac{\tilde{p}_{cI}^2}{1+\tilde{p}_{cI}^2} \times \frac{1}{1+\tilde{p}_{Cro}^2}) \tag{1.8}$$

$$\tilde{p}_{Cro} = \alpha_{Cro}(\frac{1}{1+\tilde{p}_{cI}^2} \times \frac{1}{1+\tilde{p}_{Cro}^2}) \tag{1.9}$$

(where $\alpha_i = K_i/k_i$ and ϵ is used to ensure a low basal activation level of the open complex for *cI*) has a solution for any pair of cI, Cro concentration levels that satisfies $\tilde{p}_{cI} \times \tilde{p}_{Cro} = \alpha_{Cro}/\alpha_{cI}$. Hence, random fluctuations in either protein's concentration around equilibrium can be accommodated by a commensurate change in the other protein's concentration.

In general, (1.6) is visualized by plotting its left hand side against the right hand side and looking for intersections. Exploring alternative paths to these intersection points as well as inspecting the curves for areas where they are particularly sensitive to small changes in the constituent protein concentrations provides for valuable insight in genetic networks.

In all our calculations, we used:

$$A_i = \frac{J_i u_i}{1 + J_i u_i} \tag{1.10}$$

where

$$u_i = \prod_j \left(\frac{1 + \tilde{K}\tilde{p}_j^n}{1 + K\tilde{p}_j^n} \right) \tag{1.11}$$

and $K, \tilde{K}$ are binding rates to operator sites.

2. A Framework for Recursive Nonlinear Estimation of Genetic and Biochemical Networks

2.1. **Introduction.** From the discussion above, it is now clear that any general-purpose approach for computational modelling of genetic networks needs to be able to:

- handle stochasticity on the level of individual cells;
- accurately model highly non linear systems;
- include discrete as well as continuous state variables;

[3] As an aside: oscillatory behavior is not expected to occur in λ-infected cells; once the lysis/lysogeny switch is thrown, spontaneous reversal of the decision is almost non existent. However, as mentioned before, lysis can be induced in the lysogenized cell. We can speculate if the repressilator described above is capable to induce lysis as well by periodically repressing *cI* transcription in such a cell.

- be *trainable i.e.,* be able to include the possibility of adjusting state variables or weights based upon measurements;
- include hidden state variables *i.e.,* not directly observable ones.

We believe that the framework discussed below meets all these criteria.

2.2. **The Unscented Particle Filter.** We will now summarize in a very short space developments in recursive estimation theory that have played out over decades and have been the object of intensive analysis. Of necessity, this summary will be broad rather than deep.

The Kalman filter is an exact filter, originally proposed for on-line optimal estimation of linear systems, which can be viewed as a highly efficient method for analytically propagating a Gaussian Random Variable (GRV) ("the state") through linear system dynamics[4]. The Kalman Filter has been extended to the case of recursive non-linear estimation via first-order linearizing procedures. Not surprisingly, this is known as the Extended Kalman Filter (EKF). The *Unscented* Kalman Filter (UKF), in contrast, is a recently proposed [6]–and rather ingenuous–derivative-free alternative to the extended Kalman filter which provides superior performance at an equivalent algorithmic cost. In essence, the method samples carefully chosen additional points from the original GRV ("sigma points") and propagates those exactly (i.e. no Jacobians or Hessians required)through the state equations. It has been applied very successfully in a number of application areas (*state estimation*, *parameter estimation*, and *dual estimation*)[5]. Finally, after having consecutively relaxed various conditions and limitations on the filter, the recently proposed [4] Unscented Particle Filter (UPF) now also gets rid of the restriction of the state variable distribution to a GRV by using sequential Monte Carlo (MC) methods. These methods are also known as "particle filters" because of their historic applications in nuclear physics. They allow for a complete representation of the posterior state distribution and can therefore deal with any non-linearities or distributions. Particle filters, however, rely on importance sampling and thus require the design of proposal distributions that can approximate the posterior distribution reasonably well. The UPF relies on the UKF to construct such a proposal distribution at each time step.

2.3. **Discussion.** We are now ready to give the main result of this section, stripped of almost all equations, but applicable to a wide variety of situations encountered in modelling of genetic and biochemical networks. In this we follow the general outline and notation of [4].

The basic framework for the UKF involves estimation of the state of a discrete-time nonlinear dynamic system,

$$x_i(t+1) = F_i(x_j(t), u_j(t), v_j(t)) \tag{2.1}$$

$$y_l(t+1) = H_l(x_i(t+1), n_i(t+1)) \tag{2.2}$$

where $\mathbf{x}$ represents the (unobserved or hidden) state of the system as a function of time t, $\mathbf{u}$ a known exogenous input, and $\mathbf{y}$ is the observed measurement signal. The *process* noise $\mathbf{v}$ drives the dynamic system and the *observation* noise is $\mathbf{n}$. Noises

[4]There are other finite dimensional filters. For instance, the Hidden Markov Model (HMM) filter is a very appropriate one–but rather specialized to the case of **discrete** state spaces.

[5]The main emphasis here will be on *state estimation* which is meant to be either *on-line*, as the observations become available, or in *batch* mode, for a complete set of measurements.

are zero mean. We are not assuming additivity of the noises. **F** and **H** are known and define the dynamic system. Note too that all noises and inputs can be time-dependent. Because the equations are rather involved, we summarize in words the basic UKF procedure for state estimation:

- Time Update–Given the mean and covariance of the state at time t:
  Propagate the mean of the state through the dynamic system via (2.1).
  Calculate a new state covariance at time $t + 1$.
- Measurement Update–Given the updated state at $t + 1$:
  Predict the measurement at $t + 1$ via (2.2)
  Compare the predicted measurement with the actual one.
  Construct a "gain factor" based upon the covariance of the error.
- State Correction–Given the "gain factor":
  Correct the state mean and covariance at time $t + 1$.
- Substitute $t + 1 \leftarrow t$ and repeat the Time Update step.

As an example: For our 3 genes, 3 promotor model, the set of ODEs discussed in the previous Section forms a suitable starting point for defining the state transformation **F** from 2.1. Assuming the stochastic process is approx. a GRV, one obtains the time evolution of the mean and covariance of the GRV. The estimation is further refined by feeding the model noisy experimental data.

3. Conclusions

We sketched a framework for modelling of genetic networks which combines the power of sequential Monte Carlo based methods with accurate techniques for performing recursive nonlinear estimation for dynamic systems. We make no assumptions about the exact nature of the system dynamics at this point but this framework can clearly accomodate continuous reaction kinetics systems at the one end of the spectrum and fully stochastic ones (such as master equations and transition probability matrices) at the other end. A particularly efficient implementation can be obtained if one can assume that a stochastic process is approximately a Gaussian at any fixed point in time, with a mean and covariance that follow differential equations, but such an approximation need not be made.

Such a framework could be operated in different and flexible ways: *(i)* as a parameter estimation method (a.k.a. machine learning or system identification) with numerous applications in regression, classification, and dynamic modeling, *(ii)* in dual estimation mode when both the system state and the modelling parameters need to be simultaneously estimated from the noisy data, and *(iii)* in state estimation mode.

References

1. D. T. Gillespie, *Exact Stochastic Simulation of Coupled Chemical Reactions,* J. Phys. Chem. **81(25)** (1977), 403-434.
2. A. Arkin, J. Ross, and H. H. McAdams, *Stochastic Kinetic Analysis of Development Pathway Bifurcation in Phage λ-Infected Escherichia coli Cells,* Genetics **149**, (1998), 1633–1648.
3. M. Ptashne, *A Genetic Switch: Phage λ and Higher Organisms,* Cell Press and Blackwell Scientific Publications, Cambridge, MA, 1992.
4. E. A. Wan and R. van der Merwe, *The Unscented Kalman Filter for Nonlinear Estimation,* in "Proc. of Symposium 2000 on Adaptive Systems for Signal Processing, Communications, and Control (AS-SPCC)," (2000).

5. J. M. Bower and H. Bolouri, (eds.) *Computational Modeling of Genetic and Biochemical Networks,* The MIT Press, Cambirdge, MA, 2001.
6. S. J. Julier, J. K. Uhlmann, and H. Durrant-Whyte, *A New Approach for Filtering Nonlinear Systems,* in "Proc. of AeroSense: The 11^{th} Int. Symposium on Aerospace/Defense Sensing, Simulation, and Controls," (1997).
7. T. L. Hill, *Cooperativity Theory in Biochemistry: Steady-State and Equilibrium Systems,* Springer-Verlag, New York, 1985.
8. M. A. Shea and G. K. Ackers, *The OR Control System of Bacteriophage λ. A Physical-Chemical Model for Gene Regulation,* J. Mol. Biol. **181**, (1985), 211–230.
9. M. B. Elowitz and S. Leibler, *A Synthetic Oscillatory Network of Transcriptional Regulators,* Nature **403**, (2000), 335–338.
10. H. H. McAdams and A. Arkin, *It's a Noisy Business! Genetic Regulation at the Nanomolar Scale,* Trends Genet. **15**, (1999), 65–69.
11. N. G. van Kampen, *Stochastic Processes in Physics and Chemistry,* Elsevier, Amsterdam, (1992).

IBM Server Group, Santa Fe, New Mexico
E-mail address: gwenes@us.ibm.com