Cellular Systems Biology and Biological Network Analysis

Joel S. Bader
Department of Biomedical Engineering
Johns Hopkins University

November 16, 2016

About the Author

Joel S. Bader, Ph.D., (joel.bader@jhu.edu, www.baderzone.org) is Professor of Biomedical Engineering at Johns Hopkins University, with a secondary appointment in Computer Science. He is Interim Director of the High Throughput Biology Center at the School of Medicine and a member of the Institute for Computational Medicine, the Institute for Genetic Medicine, and the Human Genetics Program. Prior to joining Johns Hopkins, Dr. Bader was employed by CuraGen Corporation (1995-2003) and is co-inventor of the Roche/454 Genome Sequencer. Dr. Bader has a Ph.D. in Theoretical Chemistry from U.C. Berkeley (1991), where he was an NSF Predoctoral Fellow, and performed post-doctoral research at Columbia University (1992-1995). Dr. Bader has a B.S. in Biochemistry from Lehigh University (1986, Phi Beta Kappa, Tau Beta Pi).

Research in the Bader lab focuses on the connection between genotype and phenotype, including human genetics, systems biology, and synthetic biology. The Bader lab has received funding from NIH, NSF CAREER, DOE, Microsoft, the Kleberg Foundation, and the Simons Foundation.

Preface

Cells are systems. Standard engineering and mathematics texts should provide an excellent introduction to understanding how cells behave, mapping inputs to outputs. Unfortunately, cells are not linear, time-independent systems. Saturation and cooperative response break linearity. Cellular states change with time. Cells are not even deterministic, violating the assumptions of non-linear systems analysis.

This book provides a self-contained introduction to cells as non-linear, time-dependent, stochastic, spatial systems. Each major section is motivated by a canonical biological pathway or phenomenon that requires the introduction of new concepts. All the required mathematical techniques are developed from the motivating examples.

The book is designed as a text for advanced undergraduate or graduate students. Prerequisites are univariate calculus, linear algebra, basic molecular biology, and rudimentary facility with a programming language for computational experiments. Linear systems and Laplace transforms are helpful, but are also reviewed in the initial chapters. Each chapter is designed to be covered in an hour lecture, and problems are provided in an Appendix.

This book is developed from course notes for "Systems Bioengineering III: Genes to Cells," taught by me since 2007 as a required course for our B.S. in Biomedical Engineering.

Joel S. Bader, Baltimore, MD

Contents

Part I

Cells as Linear Systems

Chapter 1

Cellular Signal Transduction

Chapter 2

Linear Systems Analysis

We left off last time with a model for a two-state biological signaling element,

$$(d/dt)x(t) = \beta(t) - \alpha x(t).$$

Here, $x(t)$ represents the concentration of the active form of a signaling molecule, usually meaning it is phosphorylated. The input is $\beta(t)$, and we consider it to be under our control. The rate that the activate form reverts to the inactive form is α.

Formally, we could write the solution as

$$[(d/dt) + \alpha]x(t) = \beta(t);$$

$$x(t) = [(d/dt) + \alpha]^{-1}\beta(t).$$

The problem is that we don't know what it means to take the inverse of an operator like the time derivative operator d/dt.

This is a lot like solving a matrix equation,

$$\mathbf{Ax} = \mathbf{b} - \alpha\mathbf{x}.$$

I use capital bold letters to indicate matrices and lower case bold to indicate column vectors. Elements of matrices and vectors are not bold, A_{ij} and x_i, We think about discretizing time so instead of $x(t)$ we have a vector $\mathbf{x}$ with elements $x_n = x(n\Delta t)$.

If we want this to be our actual problem, then $\mathbf{A}$ should be the time derivative operator in discrete form. Just to show you how we can do this, use the symmetric form

$$(d/dt)x_n = [x_{n+1} - x_{n-1}]/2\Delta t.$$

We also know that

$$(d/dt)x_n = \sum_{n'} A_{nn'}x_{n'} = A_{n,n+1}x_{n+1} - A_{n,n-1}x_{n-1}.$$

$$A_{n,n'} = (1/2\Delta t)(\delta_{n',n+1} - \delta_{n',n-1}).$$

The discrete or Kronecker δ-function is 1 if its arguments are the same and 0 otherwise. One way to picture $\mathbf{A}$ is a tridiagonal matrix with 1's in the diagonal above the main diagonal, 0's in the main diagonal, and -1's in the diagonal below the main diagonal.

Formally, we could solve the algebraic equation as

$$\mathbf{x} = [\mathbf{A} + \alpha\mathbf{I}]^{-1}\mathbf{x}.$$

The matrix $\mathbf{I}$ is the identity matrix, with $I_{nn'} = \delta_{nn'}$ using our friend the δ-function. We wouldn't want to solve this by hand though because taking an inverse of a large matrix is difficult.

Instead this is why we learned about eigenvectors and eigenvalues because they change the matrix inverse into a scalar inverse. We're going to do exactly the same thing here by thinking about eigenfunctions and eigenvalues.

An operator A operates on a function $f(t)$ to give a new function $Af(t) = g(t)$. We will limit ourselves to operators that we could express as matrices if we made time discrete. The main operator we will consider is the time derivative operator d/dt. We will simplify our problem is we can express everything in terms of eigenfunctions of d/dt, functions for which

$$(d/dt)f(t) \propto f(t).$$

The proportionality constant could be any scalar. Pure exponentials are eigenfunctions of d/dt,

$$(d/dt)e^{\lambda t} = \lambda e^{\lambda t}.$$

We use λ because everyone knows that λ is the name of a generic eigenvalue. Just the same way that a matrix can have many different eigenvectors, each with a different eigenvalue, an operator can have many eigenfunctions. Here we have an infinite number.

We could index each eigenfunction by its eigenvalue, $f_\lambda(t) = e^{\lambda t}$. If λ is pure real, then we have functions that grow or decay with time. We'll start instead with eigenvalues that are pure imaginary, $\lambda = i\omega$, because Fourier transforms seem more symmetric than Laplace transforms. Our convention is to think about basis functions $\phi_\omega(t) = e^{i\omega t}$.

Now really we could have any scalar in front of $\phi_\omega t$ and it would still have the same eigenvalue $i\omega$. This is the same as with eigenvectors where we fix the overall scale by insisting that eigenvectors are normalized to have a dot product of 1. Actually we want their dot products to be orthonormal. For functions, rather than the dot product, we use the inner product,

$$\langle f(t)|g(t)\rangle \equiv \int_{-\infty}^{\infty} dt\,[f(t)]^* g(t),$$

where $[f(t)]^*$ is the complex conjugate of $f(t)$. For eigenfunctions of d/dt we could abbreviate the inner product as $\langle\omega'|\omega\rangle$. If we are thinking about discrete time, then the ω values are also discrete, and we want $\langle\omega'|\omega\rangle = \delta_{\omega',\omega}$. We will do this as a homework problem to see that the correct scalar for $\phi_\omega(t)$ is $1/\sqrt{2\pi}$, so that

$$\phi_\omega(t) = (1/\sqrt{2\pi})e^{i\omega t}.$$

Notice that the inner product has two factors of $1/\sqrt{2\pi}$, and

$$\langle\omega'|\omega\rangle = (1/2\pi)\int_{-\infty}^{\infty} dt\, e^{-i\omega' t} e^{i\omega t}.$$

Math tends to split these factors symmetrically between $\langle\omega|$ and $|\omega\rangle$. Enygineering and physics usually puts the entire factor of $1/2\pi$ into $|\omega\rangle$ so that

$$x(t) = \int_{-\infty}^{\infty} d\omega \hat{x}(\omega)|\omega\rangle = \int_{-\infty}^{\infty} (d\omega/2\pi)\hat{x}(\omega)e^{i\omega t}$$

$$\hat{x}(\omega) = \langle\omega|x\rangle = \int_{-\infty}^{\infty} dt e^{-i\omega t} x(t).$$

While this would be the discrete Kronecker δ-function for a discrete time representation, in the limit that we have continuous time it becomes the Dirac δ-function, $\delta(\omega - \omega')$. For any finite value of $\Delta\omega = \omega - \omega'$, the integral goes to 0. Actually the convergence of the integral to 0 is tricky, but you can think about the indefinite integral being $e^{i\Delta\omega t}/i\Delta\omega$, which is evaluated at endpoints T and $-T$. These are so big that $e^{i\Delta\omega T}$ is oscillating so rapidly that it looks like 0.

When $\Delta\omega \to 0$, the function $\delta(\Delta\omega) \to \infty$, but in a very nice way: the area under the δ-function is 1. For any finite ε,

$$\int_{\omega-\varepsilon}^{\omega+\varepsilon} d\omega' \delta(\omega' - \omega) = 1.$$

This also makes integrals involving the δ-function easy,

$$\int_{-\infty}^{\infty} d\omega' f(\omega')\delta(\omega' - \omega) = f(\omega).$$

It just picks out the value of the rest of the integrand when its argument is 0.

If this doesn't make sense, don't worry. You'll prove all of this in homework.

As a note, we'll do one more quick thing with inner products. First notice that $\sum_{\omega'} |\omega'\rangle\langle\omega'|$ behaves like the identity matrix for functions. For example, if $f(t)$ can be expressed as $\sum_{\omega} \hat{f}(\omega)|\omega\rangle$, then

$$\sum_{\omega'} |\omega'\rangle\langle\omega'|f\rangle = \sum_{\omega'}\sum_{\omega} |\omega'\rangle\langle\omega'|\hat{f}(\omega)|\omega\rangle.$$

Remember that $\hat{f}(\omega)$ is just a scalar expansion coefficient that we can more around to get the inner product $\langle\omega'|\omega\rangle$,

$$\sum_{\omega'} |\omega'\rangle\langle\omega'|f\rangle = \sum_{\omega'}\sum_{\omega} \hat{f}(\omega)|\omega'\rangle\langle\omega'|\omega\rangle = \sum_{\omega'}\sum_{\omega} \hat{f}(\omega)|\omega'\rangle\delta_{\omega',\omega}.$$

The δ-function means that one of the sums goes away, finally giving

$$\sum_{\omega'} |\omega'\rangle\langle\omega'|f\rangle = \sum_{\omega} \hat{f}(\omega)|\omega\rangle = f(t).$$

Since this is true for any function $f(t)$ that can be expressed in the basis of $|\omega\rangle$, we conclude that $\sum_\omega |\omega\rangle\langle\omega|$ can be used as an identify operator for functions.

We can use this property to calculate the inner product $\langle f|g\rangle$ for two functions $f(t)$ and $g(t)$ as

$$\langle f|g\rangle = \langle f|\big[\sum_\omega |\omega\rangle\langle\omega|\big]|g\rangle = \sum_\omega \langle f|\omega\rangle\langle\omega|g\rangle.$$

The inner product $\langle\omega|g\rangle = \hat{g}(\omega)$. The inner product $\langle f|\omega\rangle$ is the complex conjugate of $\langle\omega|f\rangle = \hat{f}(\omega)$. Therefore, $\langle f|\omega\rangle = \hat{f}^*(\omega)$. This means that

$$\langle f|g\rangle = \sum_\omega \hat{f}^*(\omega)\hat{g}(\omega).$$

If $f(t)$ is pure real, then $\hat{f}^*(\omega) = \hat{f}(-\omega)$, and

$$\langle f|g\rangle = \sum_\omega \hat{f}(-\omega)\hat{g}(\omega).$$

Returning to our problem, our plan is to write each of our time domain functions as a sum of eigenfunctions.

$$x(t) = \sum_\omega \hat{x}(\omega)|\omega\rangle.$$

$$\beta(t) = \sum_\omega \hat{\beta}(\omega)|\omega\rangle.$$

The terms $\hat{x}$ and $\hat{\beta}$ are just the expansion coefficients. Putting this expansion into the starting equation,

$$(d/dt)\sum_\omega \hat{x}(\omega)|\omega\rangle = \sum_\omega \hat{\beta}|\omega\rangle > -\alpha\sum_\omega \hat{x}(\omega)|\omega\rangle.$$

Now we can eliminate the time derivative in favor of the eigenvalue,

$$\sum_\omega (i\omega+\alpha)\hat{x}(\omega)|\omega\rangle = \sum_\omega \hat{\beta}(\omega)|\omega\rangle.$$

Remember that what we know is $\beta(t)$, which means that we should be able to figure out the expansion coefficients $\hat{\beta}(\omega)$. We want to solve for the output expansion coefficients $\hat{x}(\omega)$. We can do this for a particular value ω' by taking the inner product,

$$\sum_\omega (i\omega+\alpha)\hat{x}(\omega)\langle\omega'|\omega\rangle = \sum_\omega \hat{\beta}(\omega)\langle\omega'|\omega\rangle.$$

$$(i\omega'+\alpha)\hat{x}(\omega') = \hat{\beta}(\omega').$$

$$\hat{x}(\omega) = (i\omega+\alpha)^{-1}\hat{\beta}(\omega).$$

We can write down the formal solution,

$$x(t) = \sum_\omega \hat{x}(\omega)|\omega\rangle.$$

For continuous time, the sum becomes an integral, with details in the homework,

$$x(t) = (1/2\pi)\int_{-\infty}^{\infty} d\omega (i\omega+\alpha)^{-1} e^{i\omega t}\hat{\beta}(\omega).$$

Substituting the inner product that gives us the expansion coefficient $\hat{\beta}(\omega)$,

$$x(t) = (1/2\pi)\int_{-\infty}^{\infty} d\omega (i\omega+\alpha)^{-1} e^{i\omega t}\int_{-\infty}^{\infty} dt' e^{-i\omega t'}\beta(t')$$

We will next change the order of the integrals. We can usually do this for physical systems. We will always be able to do it in this class.

$$x(t) = \int_{-\infty}^{\infty} dt' \frac{1}{2\pi}\int_{-\infty}^{\infty} d\omega \frac{\exp[i\omega(t-t')]}{i\omega+\alpha}\beta(t').$$

Let's think of this as a convolution or a filter,

$$x(t) = \int_{-\infty}^{\infty} dt' H(t-t')\beta(t'),$$

where the response function is

$$H(t-t') = \frac{1}{2\pi}\int_{-\infty}^{\infty} d\omega \frac{\exp[i\omega(t-t')]}{i\omega+\alpha}.$$

Take a step back and breathe after the math blizzard. We have an output $x(t)$ that comes from an ODE model for a system that is driven by input $\beta(t)$. In a causal universe, $x(t)$ should only depend on the input at times before t,

$$x(t) = \int_{-\infty}^{t} dt' H(t-t')\beta(t').$$

Plot twist! Our integral doesn't stop at t. The integral goes to infinity. What are the possibilities?

1. We made a math mistake somewhere.
2. The universe (or our model for it) is not causal.
3. There is something special about the response function $H(t)$ for causal systems.

Spoiler alert: it's the last one. Response functions for classical causal systems are only non-zero for responses to inputs in the past. In other words, if the response function $H(t-t')$ is the response of the system at time t to an input at time t', then $H(t-t')$ must be 0 for $t < t'$. Next class we'll show this by doing the integral for our system's response function.

Chapter 3

The Laplace Transform and Complex Variables

We left ourselves with the puzzle of the response function,

$$H(t) = (1/2\pi)\int_{-\infty}^{\infty} d\omega\, \frac{e^{i\omega t}}{i\omega + \alpha}.$$

We'll factor the i from the denominator,

$$H(t) = \frac{1}{2\pi i}\int_{-\infty}^{\infty} d\omega\, \frac{e^{i\omega t}}{\omega - i\alpha}.$$

Much of math depends on multiplying by 1 in an interesting way (as we did previously using $1 = \sum_\omega |\omega\rangle\langle\omega|$) or by adding 0 in an interesting way. Here well add 0 to the integral in a way that changes the integration from a line integral to an integral over a closed contour.

We start by thinking about ω in the complex plane. We can write $\omega = u + iv$, where u and v are pure real, $u = \Re(\omega)$ is the real part of ω, and $v = \Im(\omega)$ is the imaginary part of ω. The exponential factor in the integrand is $e^{i\omega t} = e^{i(u+iv)t} = e^{iut}e^{-vt}$. The line integral to evaluate is

$$H(t) = \lim_{U\to\infty}(2\pi i)^{-1}\int_{-U}^{U} du \frac{e^{iut}e^{-vt}}{u + i(v-\alpha)}.$$

At the end of the line at U, for $t > 0$, we'll take a left turn. Call this integral $A/2\pi i$,

$$A = \lim_{V\to\infty}\int_0^V dv \frac{e^{iUt}e^{-vt}}{U + i(v-\alpha)}.$$

We care about the magnitude of A,

$$|A| = |\lim_{V\to\infty}\int_0^V dv \frac{e^{iUt}e^{-vt}}{U + i(v-\alpha)}| \leq \int_0^V dv \frac{|e^{iUt}e^{-vt}|}{|U + i(v-\alpha)|},$$

since a very reasonable theorem tells us that the absolute value of an integral is no larger than the integral of the absolute value of the integrand. Next, since $|U+i(v-\alpha)| \leq |U|$, and $|e^{iUt}| = 1$,

$$|A| \leq \lim_{V\to\infty} \int_0^V dv \frac{e^{-vt}}{|U+i(v-\alpha)|} \leq (1/U) \int_0^V dv e^{-vt}$$

Finally we have an integral we can do!

$$|A| \leq 1/Ut.$$

Remember that we are taking the limit $U \to \infty$. For any finite t, $1/Ut \to 0$, which means that $A = 0$.

Now we turn left again and call this line integral $B/2\pi i$, with magnitude

$$|B| = |\lim_{U,V\to\infty} \int_U^{-U} du \frac{e^{iut}e^{-Vt}}{u+i(V-\alpha)}|.$$

Here we add the absolute value inside the integral and use $|u+i(V-\alpha)| \geq |V-\alpha|$. Then $|V-\alpha| = V|1-(\alpha/V)|$, and in the limit that $V \to \infty$, $\alpha/V \to 0$. Therefore the magnitude of B is

$$|B| \leq |\lim_{U,V\to\infty} \int_U^{-U} du \frac{|e^{iut}e^{-Vt}|}{|u+i(V-\alpha)|}| \leq \frac{e^{-Vt}}{V}|\int_U^{-U} du 1|.$$

This is another integral that is easy,

$$|B| \leq \lim_{U,V\to\infty} e^{-Vt}(2U/V).$$

If U and V approach ∞ together, then $2U/V \to 2$, and $|B| \leq 2e^{-Vt}$. For finite t, $\lim_{V\to\infty} 2e^{-Vt} = 0$, and $|B| = 0$.

We take another left turn to close the circuit, adding on $C/2\pi i$, with

$$C = \int_V^0 dv \frac{e^{-iUt}e^{-vt}}{-U+i(v-\alpha)}.$$

Notice that $C = A^*$, so $C = 0$ as well. This means that

$$H(t) = H(t) + (A+B+C)/(2\pi i) = (1/2\pi i) \oint e^{i\omega t}/(\omega - i\alpha),$$

where the $\oint$ means that the integral is over a closed contour. The contour we are considering is the large loop across the real axis, then counterclockwise into the upper imaginary plane and back down and around.

We will stop for another puzzler. Suppose that we have a function $F(\omega)$ with derivative $(d/d\omega)F(\omega) = f(\omega)$. We do an integral over a closed loop, starting at some value ω_0 and ending at the same point. Over that loop, we want to evaluate the integral $\oint d\omega f(\omega)$. In general, it should be true that

$$\int_{\omega_0}^{\omega_1} d\omega f(\omega) = F(\omega)|_{\omega_0}^{\omega_1} = F(\omega_1) - F(\omega_0).$$

For the closed loop, then, should we get $F(\omega_0) - F(\omega_0) = 0$?

The error we've made is that the endpoint isn't ω_0. Instead, if we write ω_0 in terms of a magnitude $|\omega_0|$ and a phase ϕ, $\omega_0 = |\omega_0|e^{i\phi}$, our ending point has accumulated a phase of 2π, $\omega_1 = |\omega_0|e^{i(\phi+2\pi)}$. For some functions, $F(\omega_0) = F(\omega_1)$. For these functions, the contour integral is 0. For many functions, though, $F(\omega_0) \neq F(\omega_1)$, and the contour integral has a non-zero value. For example, think about $F(\omega) = \omega^{1/2}$, and for simplicity choose $\omega_0 = 1$. In this case, $F(\omega_1) = (e^{2\pi i})^{1/2} = e^{\pi i} = -1$, $F(\omega_0) = 1$, and the contour integral gives -2.

What type of function $F(\omega)$ contributes nothing to the contour integral? Suppose that $F(\omega) = \omega^n$ where n is any integer. Then $F(\omega_1) = |\omega_0|^n e^{n(\phi i+2\pi i} = |\omega_0|^n e^{n\phi i} e^{2n\pi i} = F(\omega_0)$. A function that can be expressed as a sum of positive or negative integer powers never contributes to a contour integral. Fractional powers can contribute, though, because when n is not an integer, $e^{2n\pi i} \neq 1$.

A very special type of function $F(\omega)$ that can contribute is $F(\omega) = \ln(\omega)$ because $\Im \ln(\omega)$ is equal to the phase. For this function around a contour starting at $\omega_0 = |\omega_0|e^{i\phi}$ and ending at $\omega_1 = \omega_0 e^{2\pi i}$,

$$F(\omega_1) - F(\omega_0) = \ln(|\omega_0|) + 2\pi i + \phi i - \ln(|\omega_0|) - \phi i = 2\pi i.$$

Remember that $F(\omega)$ is integral. The integrand in the contour integral is $f(\omega) = (d/d\omega)F(\omega)$. For $F(\omega) = \ln(\omega)$, $f(\omega) = 1/\omega$. And the contour integral for $H(t)$ has something like $1/\omega$ in the denominator.

Returning to the contour integral for $H(t)$,

$$H(t) = (1/2\pi i) \oint d\omega e^{i\omega t}/(\omega - i\alpha).$$

To make things simpler, change variables to $z = \omega - i\alpha$, with

$$H(t) = \frac{e^{-\alpha t}}{2\pi i} \oint dz e^{zt}/z.$$

Then we do a power series expansion about this point. If we think about the contour for ω starting at 0 then making a big counterclockwise loop, then the contour for z starts at $z_0 = -i\alpha$ and ends at $z_1 = z_0 e^{2\pi i}$.

We can do a series expansion of $e^{zt} = \sum_{n=0}^{\infty} (zt)^n/n!$ and integrate term by term,

$$H(t) = \frac{e^{-\alpha t}}{2\pi i} \oint dz(1/z) \sum_{n=0}^{\infty} z^n t^n/n! = \frac{e^{-\alpha t}}{2\pi i} \sum_{n=0}^{\infty} (t^n/n!) \oint dz z^{n-1}.$$

From our work before, we know that all the integer terms give 0 except for the term with $n = 0$, integrating $1/z$, which gives a factor of $2\pi i$. The factor $t^0/0! = 1$. Therefore the response function for $t > 0$ is

$$H(t) = e^{-\alpha t}.$$

What about for $t < 0$? In this case, we follow the same logic of adding 0 to the integral, but instead of closing in the upper half plane we have to close in the lower half plane to make $e^{i\omega t}$

small. We end up with a clockwise rather than counterclockwise integral,

$$H(t) = (1/2\pi i) \oint d\omega e^{-i\omega|t|}/(\omega - i\alpha) = (1/2\pi i) \oint d\omega.$$

We can think about a power series expansion again. For any value of ω in the lower half plane, write $\omega_0 = \omega - i\alpha$, and consider nearby points $\omega + z$. For these points,

$$1/(\omega + z - i\alpha) = 1/(\omega_0 + z) = 1 - (z/\omega_0) + (z/\omega_0)^2 - (z/\omega_0)^3 + \ldots,$$

which is a convergent series when $|\omega_0| > 0$. The smallest magnitude of ω_0 is for $\omega = 0$, $|\omega_0| = \alpha$. Provided that $\alpha > 0$, we have a convergent series everywhere in the lower half plane, and all the powers of z are positive integers. There is no contribution to the contour integral, and $H(t) = 0$ for $t < 0$.

For Laplace transforms, instead of $\lambda = i\omega$, we use $\lambda = s$ for the eigenvalue. In other words, $s = i\omega$, or $\omega = -is$. The forward transforms are

$$\mathscr{F}[f(t)] = \hat{f}(\omega) = \int_{-\infty}^{\infty} dt e^{-i\omega t} f(t)$$

$$\mathscr{L}[f(t)] = \tilde{f}(s) = \int_{-\infty}^{\infty} dt e^{-st} f(t).$$

The inverse transforms are

$$f(t) = (1/2\pi) \int_{-\infty}^{\infty} d\omega\, e^{i\omega t} \hat{f}(\omega)$$

$$f(t) = (1/2\pi i) \int_{-i\infty}^{i\infty} ds\, e^{st} \tilde{f}(s) = (1/2\pi i) \oint ds\, e^{st} \tilde{f}(s).$$

For the inverse Laplace transform for positive time, we close the contour in the left half-plane. For positive time, we close the contour in the right half-plane.

Nothing in the definition of the Laplace transform requires that we start the time integral at 0. For an initial value problem, we essentially are saying that $f(t) = 0$ for $t < 0$ and then start the integral at 0.

If we think of a Laplace space eigenfunction of d/dt as a normalized version of e^{st}, then the eigenvalue is s. This means that there is a correspondence between d/dt in the time domain and s in the Laplace domain. We will look at two examples.

First, consider time displacement, $f(t+a)$. A Taylor series for $f(t+a)$ around $f(t)$ is

$$f(t+a) = f(t) + a(d/dt)f(t) + (a^2/2)(d/dt)^2 f(t) + (a^3/3!)(d/dt)^3 f(t) + \ldots = \sum_{n=0}^{\infty} (a^n/n!)(d/dt)^n f(t).$$

If d/dt were a scalar, we could write the sum as an exponential,

$$\sum_{n=0}^{\infty} (a^n/n!)(d/dt)^n = \exp[a(d/dt)].$$

We can do the same for operators if we just say to ourselves that the series expansion defines the meaning of the exponential. Therefore we find that

$$f(t+a) = e^{a(d/dt)} f(t).$$

For the Laplace transform,

$$\mathscr{L}[f(t+a)] = \int_{-\infty}^{\infty} dt\, e^{-st} f(t+a).$$

Changing variables to $u = t+a$, $st = su - sa$,

$$\mathscr{L}[f(t+a)] = \int_{-\infty}^{\infty} dt\, e^{-su+sa} f(u) = e^{as}\tilde{f}(s).$$

To summarize, $f(t+a) = e^{a(d/dt)} f(t)$ and $\mathscr{L}[f(t+a)] = e^{as}\mathscr{L}[f(t)]$.

We similarly look at $\mathscr{L}[(d/dt)f(t)]$. Here we consider an initial value problem where $f(t) = 0$ for $t < 0$, and then we change $f(t)$ to $f(0)$ at $t = 0$. This is done by integrating by parts,

$$\mathscr{L}[(d/dt)f(t)] = \int_0^{\infty} dt e^{-st}(d/dt)f(t) = e^{-st}f(t)|_0^{\infty} + s\int_0^{\infty} dt\, e^{-st} f(t) = -f(0) + s\tilde{f}(s).$$

Again, the (d/dt) in the time domain becomes a factor of s in the Laplace domain.

An important property of the Laplace transform is the convolution theorem. The convolution $f \star g(t)$ is defined as

$$f \star g(t) = \int^{t} dt' f(t-t')g(t').$$

Usually we are interested in initial value problems where $f(t) = g(t) = 0$ for $t = 0$ and the system turns on at $t = 0$, in which case the starting point of the integral is $t = 0$. Note that for a linear system with response function $H(t)$, the response $x(t)$ to an input $\beta(t)$ is $x(t) = H \star \beta(t)$.

The Laplace transform of a convolution is

$$\mathscr{L}[f \star g(t)] = \int_0^{\infty} dt\, e^{-st} \int_0^{t} dt'\, f(t-t')g(t').$$

Changing variables from t and t' to $t - t'$ and t' and multiplying by $1 = e^{st'}e^{-st'}$,

$$\mathscr{L}[f \star g(t)] = \int_0^{\infty} dt' e^{-st'} g(t') \int_0^{\infty} d(t-t')\, e^{-s(t-t')} f(t-t') = \tilde{f}(s)\tilde{g}(s).$$

For a linear system, the response in Laplace space is $\tilde{x}(s) = \tilde{H}(s)\tilde{\beta}(s)$.

Now a few notes on the inverse Laplace transform. Suppose we are working on an initial value problem with step input, $\beta(t) = \beta_0$ for $t > 0$ and $\beta(t) = 0$ for $t < 0$. The Laplace transform is

$$\tilde{\beta}(s) = \int_0^{\infty} e^{-st}\beta_0 = \beta_0/s.$$

When we go to do the inverse transform, though, we notice that the pole at $s = 0$ lies on the integration contour. What do we do? The answer depends on the physical interpretation of the problem. Here, our convention is that everything dies. We don't allow an input that stays on forever. Instead, we take an input of the form $\beta(t) = \beta_0 e^{-\varepsilon t}$ and take the limit $\varepsilon \to 0$. For this input,

$$\tilde{\beta}(s) = \beta_0/(s+\varepsilon),$$

and the pole is inside the integration contour. Therefore for $t > 0$ when we close the contour on the left, we get the full value of the pole, $\beta(t) = \beta_0$. For $t < 0$, we close the contour on the right, there are no poles, and $\beta(t) = 0$. Some texts will tell you to "shift the contour to the right of the imaginary axis" or "shift the contour to the right of any poles", but really you have to know how the equations correspond to the physical system to be sure about what to do. And you also have to know that for negative t you close on the right.

Anywhere that a function is well behaved, you can move an integration contour without affective the result. This means that for a function with multiple poles, you can evaluate their contributions separately,

$$(1/2\pi i) \oint e^{st}/(s+\alpha)(s+\beta) = e^{-\alpha t}/(-\alpha+\beta) + e^{-\beta t}/(-\beta+\alpha).$$

For a second-order pole $1/(s+\alpha)^2$, you can take the limit as $\beta \to \alpha$. More generally, for

$$(1/2\pi i) \oint f(s)/(s+\alpha)^n$$

where $f(s)$ is well behaved for s close to $s = -\alpha$, the solution is to do a series expansion of $f(s)$ around this point. The only term that contributes is $[1/(n-1)!](d/ds)^{n-1} f(s)|_{-\alpha}$.

And that is it for the theory of Laplace transforms.

Chapter 4

Signal Transduction Cascades and MAPK Signaling

We have made a hypothesis that cells exposed to a weak stimulus will exhibit linear response. Next we made a model for the response function based on the idea that the most important step in the cell's response is activation and deactivation of a signaling molecule. The model has only two parameters. One is the deactivation rate α to return the signaling molecule to the inactive state. The other parameter was the activation rate constant, which is subsumed into the activation rate β. Here will will use the resulting system response function $H(t) = \exp(-\alpha t)$ to predict the response of the cell to different inputs.

Usually we are concerned about a few different types of input:

- δ-function input, $\beta(t) = \beta_0 \delta(t)$;
- exponential input, $\beta(t) = \beta_0 k e^{-kt}$;
- step-function input, $\beta(t) = \beta_0 \Theta(t \geq 0)$;
- oscillating input, $\beta(t) = \beta_0 \cos(\omega t) = \beta_0 \Re e^{i\omega t}$.

The step-function input introduces the logic function $\Theta(x)$ which is 1 if the argument x is true and 0 if false. The oscillating input could more generally be $\cos(\omega t + \theta)$ where θ is a phase shift, for example $\theta = -\pi/2$ giving $\sin(\omega t)$ rather than $\cos(\omega t)$. Usually an oscillating input is applied for a long time, making it more natural to take the phase shift as 0. Often it simplifies calculations to write $\cos(\omega t) = \Re e^{i\omega t}$.

The δ-function input and the exponential input are normalized to have the same integrated area β_0. The step-function input and oscillating input are normalized to have the same amplitude β_0.

From before, the system dynamics are

$$\dot{x}(t) = \beta(t) - \alpha x(t).$$

We take boundary conditions that the system is prepared at time 0 in state $x(0)$. Applying the Laplace transform,

$$s\tilde{x}(s) - x(0) = \tilde{\beta}(s) - \alpha\tilde{x}(s);$$

$$\tilde{x}(s) = x(0)/(s+\alpha) + \tilde{\beta}(s)/(s+\alpha);$$

$$x(t) = x(0)e^{-\alpha t} + \int_0^t dt'\, e^{-\alpha(t-t')}\beta(t').$$

Notice that the effect of the preparation is entirely through the transient term $x(0)e^{-\alpha t}$. This is a feature of linear response: the system response is the simple sum of the decay of the time 0 value and the response to the applied input. If we prepare the system at time 0 and then apply no input ($\beta(t) = 0$), we can measure the decay of the transient and use this to determine the value of the single parameter. Our model also says that the decay rate is the same regardless of the value of $x(0)$. From here on, unless explicitly mentioned, we will assume for simplicity that $x(0) = 0$.

For δ-function input, $\beta(t) = \beta_0\delta(t)$, we use the convention that $\tilde{\beta}(s)$ captures all of the input, with the δ-function shifted infinitesimally to the right of the origin $t = 0$. Alternatively, $\delta(t)$ can be represented as fast exponential input, $\delta(t) = \lim_{k\to\infty} ke^{-kt}$. In either case, $\tilde{\delta}(s) = 1$, $\tilde{x}(s) = \beta_0\tilde{H}(s)$, and $x(t) = \beta_0 H(t)$, a general result for any linear system with response function $H(t)$. In our case, $H(t) = e^{-\alpha t}$, and we can use the response to a δ-function input to measure the response function.

For step-function input, $\beta(t) = \beta_0\Theta(t > 0)$, and $\tilde{\beta}(s) = \beta_0/s$. In Laplace space,

$$\tilde{x}(s) = \beta_0/s(s+\alpha).$$

Taking the inverse Laplace transform with our new-found skills,

$$x(t) = \int_{-i\infty}^{i\infty} (ds/2\pi i)\beta_0 e^{st}/s(s+\alpha).$$

We remember that the term $s = 0$ in the denominator should really be the factor $s + \varepsilon$ for a step function that decays infinitesimally slowly ($\varepsilon \to 0^+$), pushing the pole at 0 just inside our contour. We then do the integral,

$$x(t) = \beta_0[(1/\alpha) + e^{-\alpha t}/(-\alpha)] = (\beta_0/\alpha)(1 - e^{-\alpha t}).$$

As $t \to \infty$, the system goes to steady-state value β_0/α.

When the input approaches a constant value at long time, $\lim_{t\to\infty}\beta(t) = \beta_0$, a system with dissipation will have $\lim_{t\to\infty}\dot{x} = 0$. A non-dissipative system will not necessarily have this property. For example, a perfect spring will continue to bounce. My car in graduate school had worn out shock absorbers and it would bounce forever when I went over a bump. A postdoc who got a real job at a scientific contracting company sold it to me for $100 in 1989 or so, and I decided that any repair that cost more than $100 wasn't worth it. So, instead of a gas pedal it just had the metal bar to push, it had lost its paint at some point and was just primer brown, and instead of a dashboard it had a layer of astroturf. Also the door locks were broken and any key would open them. For a

while I think only two of the four spark plugs were working, which made me late and greasy for a blind date when I had to put in new spark plugs on the side of the road. The actual key was required to start the car, and to save space on my keychain I just left it tied next to driver's seat. When I got my PhD, I tried to sell the car for $100 but no one would buy it. Then I tried to give it away but no one would take it. Finally I parked it in front of a friend's apartment and left the key on his desk at work. He drove it for a while but then it broke down on the freeway. He got out and just left it there.

Meanwhile back at the ranch, $\lim_{t\to\infty}\dot{x}(t) = 0 = \beta_0 - \alpha x(t)$, and $\lim_{x\to\infty} x(t) = \beta_0/\alpha$. This is a useful check on our math for the actual system, where we obtained $x(t) = (\beta_0/\alpha)(1-e^{-\alpha t})$. It is a good sign that the full solution agrees with the expected long-time behavior.

Finally, the oscillating input, $\beta(t) = \beta_0\cos(\omega t) = \beta_0\Re e^{i\omega t}$. The input in Laplace space is

$$\tilde{\beta}(s) = \beta_0\int_0^\infty dt\, e^{-st}\Re e^{i\omega t}.$$

Since the rest of the integrand is real, we can move the $\Re$ operator outside the integral (the sum of the real part is the real part of the sum),

$$\tilde{\beta}(s) = \beta_0\Re\int_0^\infty dt\, e^{-(s-i\omega)t} = \beta_0\Re 1/(s-i\omega) = \beta_0 s/(s^2+\omega^2).$$

The output is then

$$\tilde{x}(s) = \beta_0 s/(s+\alpha)(s-i\omega)(s+i\omega).$$

We have simple poles at $-\alpha$ and $\pm i\omega$. As usual, the poles on the imaginary axis are shifted just inside the contour, equivalent to a physical system that switches of as $t\to\infty$ with input $e^{-\varepsilon t}\cos(\omega t)$.

The time-domain output is the sum of the contribution from each pole,

$$x(t) = \int_{-i\infty}^{i\infty}(ds/2\pi i)\beta_0 s e^{st}/(s+\alpha)(s-i\omega)(s+i\omega).$$

We'll write $x(t) = x_\alpha(t) + x_\omega(t)$ where x_α is the contribution from the real pole at $-\alpha$ and x_i is the contribution from the imaginary poles at $\pm i\omega$.

The pole at $s = -\alpha$ gives a decaying contribution,

$$x_\alpha(t) = \beta_0(-\alpha)e^{-\alpha t}/(-\alpha-i\omega)(-\alpha+i\omega) = \beta_0\alpha e^{-\alpha t}/(\alpha^2+\omega^2).$$

In the limit that the input frequency ω is slow compared to the system response time α, $1/(\alpha^2+\omega^2)\to 1/\alpha^2$, and $x_\alpha(t)\to -(\beta_0/\alpha)e^{-\alpha t}$.

The pure imaginary poles give an oscillating contribution,

$$x_\omega(t) = \beta_0(i\omega)e^{i\omega t}/(i\omega+\alpha)(2i\omega) + \beta_0(-i\omega)e^{-i\omega t}/(-i\omega+\alpha)(-2i\omega).$$

$$x_\omega(t) = (\beta_0/2)[e^{i\omega t}/(\alpha+i\omega) + e^{-i\omega t}/(\alpha-i\omega)].$$

It is excellent to see that the response is the sum of an imaginary number and its complex conjugate, which gives a real response. The response of a physical system should always be real.

The way to make progress is to write $\alpha \pm i\omega$ as a complex number with magnitude $\sqrt{\alpha^2+\omega^2}$ and phase $\pm i\phi$. Now think about this; it is a very standard transformation between Cartesian and polar coordinates. If $a+ib$ and $re^{i\phi}$ are the same number in Cartesian and polar coordinates, then $\tan(\phi) = b/a$. Here we have $a+ib = \alpha + i\omega$ and we want to find ϕ. The inverse is $\phi = \tan^{-1}(b/a) = \tan^{-1}(\omega/\alpha)$. For very slow input compared to the system response, $\omega/\alpha \to 0$, and $\phi \to 0$. For very fast input compared to the system response, $\omega/\alpha \to \infty$, and $\phi \to \pi/2$.

Continuing with our solution,

$$x_\omega(t) = (\beta_0/2\sqrt{\alpha^2+\omega^2})(e^{i\omega t - i\phi} + e^{-i\omega t + i\phi});$$

$$x_\omega(t) = (\beta_0/\sqrt{\alpha^2+\omega^2})\cos(\omega t - \phi).$$

The oscillating part of the output has the same frequency as the input. This is a general property for linear systems. Since the output is

$$\tilde{x}(s) = \tilde{H}(s)\tilde{\beta}(s),$$

if $\tilde{\beta}(s) = 0$ for some frequency $\omega = \Im s$, then $\tilde{x}(s)$ must also also be missing that frequency. If $\beta(t)$ has non-zero weight at a frequency, then $x(t)$ does also as long as the response function $H(t)$ and respond at that frequency. Sometimes the response function is absent at a frequency; this tends to happen for inputs that oscillate so fast compared to the system response time that the system sees a time-averaged input of 0.

Let's imagine that we've been running the oscillating input for long enough that the transients have all relaxed, leaving just the oscillating part of the output,

$$x(t) = x_\omega(t) = (\beta_0/\sqrt{\alpha^2+\omega^2})\cos[\omega t + \tan^{-1}(\omega/\alpha)].$$

For slow input, we get the full response amplitude β_0/α, and the output follows the input exactly because the phase shift is 0,

$$x(t) \to (\beta_0/\alpha)\cos(\omega t).$$

For fast input, $1/\sqrt{\alpha^2+\omega^2} \to 1/\omega$, and the phase shift $\to \pi/2$, giving

$$x(t) \to (\beta_0/\omega)\sin(\omega t).$$

For high frequency, we expect that the response amplitude decreases as $1/\text{frequency}$. The system is $\pi/2$ behind the input, changing the cosine input to sine output. This is what you do naturally when you push someone (or yourself) on a swing, where for maximum energy transfer you push just before the change of direction.

How do cells respond to signals in real life? Does a one parameter model really work? Amazingly, yes. For some reason biologists prefer square waves to sine waves as input. The math is

more difficult for square waves because they are non-differentiable at the corners. They are easier to program, though, which reminds me of programming temperature settings for a PCR thermocycler. Many of you have used PCR already. I am older than my sister, and she is so old that when she was a graduate student PCR had just been invented. They didn't have thermocyclers back then. Instead, you had water baths set at different temperatures and you just held your test tube in one bath and walked over to the other bath.

Yeast cells response to changes in osmotic pressure by signaling through a mitogen-activated protein kinase (MAPK) cascade. These cascades have 3 levels, each level corresponding to a kinase that requires phosphorylation for activity. Sometimes biologists go wild with names, and usually the it Drosophila community has the best names; sometimes they don't. Here the proteins in the cascade are called generically MAPKKK, MAPKK, and MAPK, for kinase kinase kinase, kinase kinase, and kinase. A MAPK cascade very important in cancer is the RAS-RAF-ERK cascade. In addition to osmotic response, yeast use MAPK cascades for responding to sex pheromones, changing from proliferative growth through mitosis to mating and sporulation through meiosis. Sometimes the MAPK components are reused in pathways, with a scaffold protein holding them in place to prevent cross-talk.

The response of a linear cascade is very easy to calculate. The ODE model is

$$\dot{x}_1(t) = k_1\beta(t) - \alpha_1 x_1(t);$$

$$\dot{x}_2(t) = k_2 x_1(t) - \alpha_2 x_2(t);$$

$$\dots$$

$$\dot{x}_n(t) = k_n x_{n-1}(t) - \alpha_n x_n(t).$$

The output of each level in the cascade is the input to the next level. We will assume that the system is off at time 0, with $x_i(t) = 0$ for all $i = 1, 2, 3, \dots, n$. The solution in the Laplace domain is

$$\tilde{x}_1(s) = [k_1/s + \alpha_1]\tilde{\beta}(s);$$

$$\tilde{x}_1(s) = [k_2/s + \alpha_2]\tilde{x}_1(s) = [k_2 k_1/(s+\alpha_1)(s+\alpha_2)]\tilde{\beta}(s);$$

$$\dots$$

$$\tilde{x}_n(s) = \tilde{\beta}(s)\prod_{i=1}^{n} k_i/(s+\alpha_i).$$

The system response function at level n is

$$\tilde{H}_n(s) = \prod_{i=1}^{n} k_i/(s+\alpha_i).$$

This is messy to convert to the time domain response function $H(t)$ because of all the different decay constants $\{\alpha_i\}$. It is very tractable, however, if we assume that the constants are all the same. This is a reasonable approximation because the dephosphorylation steps are often catalyzed by the

same phosphatase. Different activation rate constants $\{k_i\}$ don't matter as much because they just give an overall prefactor rather than any difference in the shape of the response function.

With the approximation that each level has the same parameters,

$$\tilde{H}_n(s) = k^n/(s+\alpha)^n.$$

The time-domain response function is

$$H(t) = k^n \int_{-i\infty}^{i\infty} (ds/2\pi i) e^{st}/(s+\alpha)^n.$$

With a pole of order n, the approach is to write the numerator as an expansion in terms of $(s+\alpha)$ and to take the $n-1$ term to get the residue. Carrying out this plan,

$$e^{st} = e^{-\alpha t} e^{t(s+\alpha)} = e^{-\alpha t} \sum_{n'=0}^{\infty} (t^{n'}/n'!)(s+\alpha)^{n'}.$$

Only the $n' = n-1$ term contributes, giving

$$H(t) = e^{-\alpha t} k^n t^{n-1}/(n-1)!.$$

We are happy to see that for a single level cascade, we get our old friend $H_1(t) = e^{-\alpha t}$.

If you think about this function, we have two terms fighting it out. The term t^{n-1} increases with t, but the term $e^{-\alpha t}$ decreases with t. Exponentials beat algebraic terms, so $\lim_{t\to\infty} H(t) = 0$. The maximum value occurs when $(d/dt)H(t) = 0$. Usually it's easier to do these calculations on a logarithmic scale: if $(d/dt)H(t) = 0$ and $H(t) \neq 0$, then $[1/H(t)](d/dt)H(t) = 0 = (d/dt)\ln H(t)$. In this case,

$$\begin{aligned}(d/dt)[-\alpha t + (n-1)\ln t] &= 0;\\ \alpha &= (n-1)/t;\\ t &= (n-1)/\alpha.\end{aligned}$$

For an n-step cascade, the maximum response to an impulse at time 0 is at $t = (n-1)/\alpha$. Each level of the cascade adds a delay of $1/\alpha$, which is the timescale to return to the unactivated state.

If we wanted, we could substitute the time $t = (n-1)/\alpha$ back into the expression for $x_n(t)$ to find the maximum response. Instead, we will think about the response to a constant input. For a constant input, $\beta(t) = \beta_0\Theta(t>0)$, the response at long time is

$$\lim_{t\to\infty} x(t) = \int_0^{\infty} dt' H(t-t')\beta(t') = \beta_0 \int_0^{\infty} d\tau\, H(\tau) = \beta_0 \tilde{H}(0).$$

Since $\tilde{H}_n(s) = k^n/(s+\alpha)^n$, the long-time response of x_n is $\beta_0(k/\alpha)^n$. If $k > \alpha$, the activation is greater than the deactivation at each level and the response increases along the cascade. If $k < \alpha$, the response decreases.

Notice that we calculated a time-domain property directly from the Laplace-domain response function. There are several other time-domain properties that are easy to generate from the Laplace-domain response functions. These are moments of the response, and $\tilde{H}(s)$ is the moment generating function. We will see how easy this at our next lecture.

Chapter 5

Generating Functions, Pharmacokinetics and Pharmacodynamics

Calculating time domain properties for simple cascades can become somewhat messy. From last lecture, we saw that the response function for an n-level cascade,

$$\tilde{H}_n(s) = k^n/(s+\alpha)^n,$$

has the time-domain solution

$$H_n(t) = k^n e^{-\alpha t} t^{n-1}/(n-1)!.$$

We also found that the time of maximum response is

$$t_n = (n-1)/\alpha.$$

If we want to know the amplitude A at maximum response, we can substitute this time back into the expression for $H_n(t)$,

$$A = H_(t_n) = k^n e^{-\alpha(n-1)/\alpha}(n-1)^{n-1}/\alpha^{n-1}(n-1)!$$

$$A = (k^n/\alpha^{n-1})(n-1)^{n-1}/e^{n-1}(n-1)!.$$

If we use Stirling's approximation, $n! \approx (n/e)^n$, the result is

$$A \approx k^n/\alpha^{n-1}.$$

There are other more useful measures of gain, though. We are often interested in the integrated response, also known as the area under the curve,

$$\text{AUC} = \int_0^\infty dt\, x(t).$$

For this chapter, we will define the gain as the ratio of the area under the curve for the response $x(t)$ normalized by the area under the curve for the input. Note that

$$\int_0^\infty dt\, f(t) = \lim_{s\to 0}\int_0^\infty dt\, e^{-st} f(t) = \lim_{s\to 0}\tilde{f}(s).$$

We have to take the limit because for functions that go to a long-time non-zero value, $|\tilde{f}(s)| \to \infty$. We've seen this for the unit step function. The gain G is defined as

$$G \equiv \lim_{s\to 0}\tilde{x}(s)/\tilde{\beta}(s).$$

Even when $\tilde{\beta}(0)$ is ill-defined, for example the unit step or a never ending cosine wave, the ratio should be well defined by l'Hôpital's rule. In fact, though, we don't have to go to the hospital. Instead, note that

$$\tilde{x}(s) = \tilde{H}(s)\tilde{\beta}(s).$$

Therefore the gain is

$$G = \lim_{s\to 0}\tilde{H}(s) = \tilde{H}(0).$$

The limit goes away because in any universe lacking perpetual motion, $\tilde{H}(0)$ is finite. An infinite value would mean that a finite kick would create infinite response.

Back to our example of an n-level cascade, the AUC gain G_n is

$$G_n = (k/\alpha)^n.$$

We can generalize this result very easily for the case where each level has its own k and α,

$$G_n = \prod_{j=1}^{n}(k_i/\alpha_i).$$

We will pause to think about physical meaning. The k terms are the activation rate for each level, roughly proportional to the abundance of the activating enzyme. The α terms are the deactivation rate. If we perturb a cell to increase the number of activating enzymes, the gain increases. If we reduce the number of activating enzymes, the gain decreases. Similarly, we can make cells with extra or reduced deactivating enzymes, phosphatases for MAPK signaling. This can be done by transforming or transfecting cells with plasmids that over-express a protein of interest, or by making knockdowns with shRNA or RNAi or knockouts with mutagenesis or genome editing, these days using CRISPR/Cas9 systems.

We are also interested in the mean time of response $\bar{t}$. For any non-negative time-domain function $f(t)$, we define the mean time of response as

$$\bar{t} = \int_0^\infty dt\, t f(t) / \int_0^\infty dt\, f(t).$$

This is related to the time of maximum response the same way that the mean of a probability distribution is related to the mode, $\bar{t}$: $\arg\max_t f(t)$ as mean:mode. In fact you can think of $f(t)$ as a weighting function describing how much of the response comes at time t.

If you know the Laplace transform and its derivatives at $t \to 0^+$, you can calculate $\bar{t}$ very easily. The Laplace transform $\tilde{f}(s)$, or rather its logarithm, $\ln \tilde{f}(s)$, is a moment-generating function because its derivatives give the moments of the corresponding time-domain function $f(t)$. This proof is important and is usually the subject of quiz or exam questions. Here goes.

$$(-d/ds)\ln \tilde{f}(s) = \tilde{f}(s)^{-1}(-d/ds)\int_0^\infty dt\, e^{-st} f(t) = \tilde{f}(s)^{-1}\int_0^\infty dt\, t\, e^{-st} f(t).$$

$$\lim_{s\to 0}(-d/ds)\ln \tilde{f}(s) = \lim_{s\to 0}\int_0^\infty dt\, t f(t) e^{-st} / \int_0^\infty dt\, f(t) e^{-st}.$$

$$\lim_{s\to 0}(-d/ds)\ln \tilde{f}(s) = \int_0^\infty dt\, t f(t) / \int_0^\infty dt\, f(t) = \bar{t}.$$

For a system response $\bar{t}_H$, the mean time of response is the difference between the mean time of the output, $\bar{t}_x$ and the input, $\bar{t}_\beta$,

$$\bar{t} = \bar{t}_x - \bar{t}_\beta = \lim_{s\to 0}(-d/ds)\ln \tilde{x}(s) - (-d/ds)\ln \tilde{\beta}(s).$$

For a linear system, $\tilde{x}(s) = \tilde{H}(s)\tilde{\beta}(s)$, and

$$\bar{t} = \lim_{s\to 0}(-d/ds)\ln[\tilde{H}(s)\tilde{\beta}(s)] - (-d/ds)\ln \tilde{\beta}(s) = (-d/ds)\ln \tilde{H}(s)|_{s=0}.$$

Even if the integrated response is infinite, the integrated response function and its derivatives should be finite.

For our n-level cascade, we have the simple result

$$\bar{t}_n = (-d/ds)\ln k^n/(s+\alpha)^n|_{s=0} = n/(s+\alpha)|_{s=0} = n/\alpha.$$

Recall that the time of maximum response was $(n-1)/\alpha$. Both the time of maximum response and the mean time of response increase by $1/\alpha$ at each step.

We can easily generalize $\bar{t}$ to cascades with unequal parameters,

$$\bar{t} = (-d/ds)\ln \prod_{j=1}^{n} k_j/(s+\alpha)^j|_{s=0} = (-d/ds)\sum_{j=1}^{n} -\ln(s+\alpha_j)|_{s=0} = \sum_{j=1}^{n} 1/\alpha_j.$$

Each step in the cascade has its own relaxation time $1/\alpha_j$, and the sum of the relaxation times is the mean time of response.

This theory also tells us that for a linear cascade, the response time depends only on the de-activation rates. If we increase the activation rates, we increase the gain, but we do nothing to the response time. You might think that increasing the activating rate increases the speed of response, but you know nothing John Snow.

The second moment tells us about the square duration of response, Δt^2, similar to a variance for a probability distribution,

$$\Delta t^2 \equiv \bar{t^2} - \bar{t}^2 \equiv \int_0^\infty dt\,(t-\bar{t})^2 f(t) / \int_0^\infty dt\, f(t),$$

where $\bar{t}$ is the previously defined mean time of response. As a homework, you can prove that

$$\Delta t^2 = \lim_{s\to 0}(-d/ds)^2 \ln \tilde{f}(s).$$

As before, for linear response $\tilde{x}(s) = \tilde{H}(s)\tilde{\beta}(s)$, and we calculate the duration of response as the duration of the output minus the duration of the input. We have

$$\Delta t^2 = \lim_{s\to 0}(-d/ds)^2[\ln \tilde{H}(s)\tilde{\beta}(s) - \ln \tilde{\beta}(s)] = (-d/ds)^2 \ln \tilde{H}(s)|_{s=0}.$$

To do these problems, always take the logarithm before taking the derivative, and set $s = 0$ as the very last step. If you take the derivative before the logarithm, you'll still get the correct answer, but the calculations will be more involved. If you set $s = 0$ before the end, there's no s left to take the derivative.

For our general n-level cascade, the square duration of response is

$$\Delta t^2 = (-d/ds)\sum_{j=1}^{n}(s+\alpha_j)^{-1}|_{s=0} = \sum_{j=1}^{n} 1/(s+\alpha_j)^n|_{s=0} = \sum_{j=1}^{n} 1/\alpha_j^2.$$

Again, the duration depends only on the deactivation rates, not the activation rates.

A final property that is useful to calculate is the mean amplitude, $\bar{A}$. If we approximate the response as a square wave with duration Δt and amplitude $\bar{A}A$, the AUC gain G is the product $\bar{A}\Delta t$. Therefore we can obtain the mean amplitude as

$$\bar{A} = G/\Delta t = \tilde{H}(0)/\sqrt{(-d/ds)^2 \ln \tilde{H}(s)|_{s=0}}.$$

If the parameters are identical for each level in the hierarchy, then

$$G = (k/\alpha)^n;$$

$$\bar{t} = n/\alpha;$$

$$\Delta t = \sqrt{\Delta t^2} = \sqrt{n}/\alpha;$$

$$\bar{A} = k^n/\alpha^{n-1}\sqrt{n}.$$

For a single level, $\bar{A} = k$. The interpretation is that the activation rate constant k determines the response amplitude, and the deactivation rate constant determines the response duration. The product of the amplitude and the duration then gives the response area.

In summary, for a serial pathway, the gains multiply and the mean times add. What about a pathway with convergent branches? We will think about a signaling protein x, that is activated by an upstream signal, with dynamics

$$\dot{x}(t) = k_{x,\beta}\beta(t) - \alpha_x x(t).$$

In mathematics and physics, we usually read subscripts from right to left. The parameter $k_{x,\beta}$ is the natural way to write the activation of x due to input β. We'll assume that the system is off at time 0, giving the Laplace-space solution

$$\tilde{x}(s) = [k_{x,\beta}/(s+\alpha_x)]\tilde{\beta}(s).$$

We'll define the feedback-free response function as $H_0(t)$ or $\tilde{H}_0(s)$ in Laplace space,

$$\tilde{H}_0(s) = k_{x,\beta}/(s+\alpha_x).$$

You can imagine that we could put together a much more complicated model for $H_0(t)$. For example we could have a cascade leading to a product $\prod_j k_j/(s+\alpha_j)$, or we could have convergent signaling from multiple upstream branches,

$$\dot{a}(t) = k_{a,\beta}\beta(t) - \alpha_a a(t)$$

$$\dot{b}(t) = k_{b,\beta}\beta(t) - \alpha_b b(t)$$

$$\dot{x}(t) = k_{x,a}a(t) + k_{x,b}b(t) - \alpha_x x(t)$$

with Laplace-space solution

$$\tilde{a}(s) = [k_{a,\beta}/(s+\alpha_a)]\tilde{\beta}(s)$$

$$\tilde{b}(s) = [k_{b,\beta}/(s+\alpha_b)]\tilde{\beta}(s)$$

$$\tilde{x}(t) = [k_{x,a}k_{a,\beta}/(s+\alpha_x)(s+\alpha_a)]\tilde{\beta}(s) + [k_{x,b}k_{b,\beta}/(s+\alpha_x)(s+\alpha_b)]\tilde{\beta}(s) = [\tilde{H}_a(s) + \tilde{H}_b(s)]\tilde{\beta}(s).$$

For convenience, define the response function for the converging branches as

$$\tilde{H}_x(s)\tilde{\beta}(s) \equiv \tilde{H}_a(s) + \tilde{H}_b(s).$$

Defining gains as G_a, G_b, G_x and mean times as $\bar{t}_a$, $\bar{t}_b$, $\bar{t}_x$, the converging branches give

$$G_x = G_a + G_b.$$

The mean time of response is

$$\bar{t}_x = \lim_{s\to 0}(-d/ds)\ln[\tilde{H}_a(s) + \tilde{H}_b(s)]$$

$$\bar{t}_x = [\tilde{H}_a(s) + \tilde{H}_b(s)]^{-1}[(-d/ds)\tilde{H}_a(s) + (-d/ds)\tilde{H}_b(s)]|_{s=0}$$

$$\bar{t}_x = [G_a + G_b]^{-1}[\tilde{H}_a(s)(-d/ds)\ln\tilde{H}_a(s) + \tilde{H}_b(s)(-d/ds)\ln\tilde{H}_b(s)]|_{s=0}$$

$$\bar{t}_x = [G_a + G_b]^{-1}[G_a\bar{t}_a + G_b\bar{t}_b],$$

the gain-weighted mean. The mean square variation in response time is left as a homework exercise.

This type of analysis is important for drug action. For example, suppose a drug is given in an unavailable form, either a pro-drug or a pill form, that has to be converted to an available or active form, which is then degraded. The active form couples to a biological pathway to have an affect. We have control over $U(t)$, the unavailable form. The active drug is denoted $D(t)$. A reasonable minimal model for the action of a dose U_0 given at time 0 is

$$\dot{U}(t) = -cU(t),$$

$$\dot{D}(t) = cU(t) - dD(t),$$

$$\dot{x}(t) = kD(t) - \alpha x(t).$$

The parameter c is the rate of conversion from inactive to active form. For a time release medication, c would be small, roughly 1/(6 hours) to 1/(1 day). For direct delivery into the blood, c could be faster, 1/(1 min). The parameter d is the rate of degradation by metabolism, excretion, or other mechanisms. The pathway activity is represented by x. These dynamics have the Laplace-space solution

$$\tilde{U}(s) = U_0/(s+c);$$

$$\tilde{D}(s) = \tilde{U}(s)c/(s+d) = cU_0/[(s+c)(s+d)];$$

$$\tilde{x}(s) = \tilde{D}(s)k/(s+\alpha) = ckU_0/[(s+c)(s+d)(s+\alpha)].$$

Now suppose we calculate the AUC gain G for $x(t)$ per initial dose U_0. The result is

$$G = \tilde{x}(0)/U_0 = ck/cd\alpha = k/d\alpha.$$

The AUC gain is independent of how fast the drug is converted from inactive to active form. Instead, it depends on the rate at which it is degraded and on the standard pathway parameter combination k/α, activation vs. deactivation rate for the signaling pathway. The mean time of activity $\bar{t}$ is

$$\bar{t} = 1/c + 1/d + 1/\alpha.$$

Over the years, we have learned quite a bit about the factors that affect these rates because they are critical to accurate dosing. Many of the most important enzymes for drug metabolism are Cytochrome P450's (CYPs), which use heme as a cofactor for redox reactions. Human genetic variation in CYPs leads to differences in drug activity. If the drug is given as a prodrug, the CYPs are often responsible for conversion to an active form, represented by the model parameter c. If the drug is metabolized rather than excreted, the CYPs are often responsible for degradation, represented by the model parameter d. CYP inhibitors can either stretch out the affect of a

drug (decreasing c and increasing $\bar{t}$) to the extent that the effective concentration is too low. CYP inhibitors can also reduce the degradation, decreasing d and resulting in a much higher gain.

Foods can also affect CYP activity. A well-known example is grapefruit, which contains furanocoumarins and flavonoids that inhibit CYPs, in particular CYP3A4, with a half-life of 1-2 days. Grapefruit has known interactions with almost 100 drugs, including benzodiazepines (Valium, Xanax), ADHD therapeutics (Adderall), and sertraline (Zoloft).

Chapter 6

Positive and Negative Feedback and Caffeine Response

We have seen that a cascade is one way to increase the gain and duration of a response. Cells have evolved more efficient ways to accomplish these goals, and most often they involve positive feedback. Here we explore a positive feedback in the linear regime. We will then turn to an actual example, positive feedback in the dopamine / adenosine receptors in neurons that is also responsible for the long-duration effect of caffeine on the human brain.

We will think about the part of a signaling pathway that involves a feedback loop. The top of the pathway is x, and the bottom is y. Our standard example is

$$\dot{x}(t) = k_{x\beta}\beta(t) - \alpha_x x(t)$$

$$\dot{y}(t) = k_{yx}x(t) - \alpha_y y(t)$$

This has solution

$$\tilde{x}(s) = \frac{k_{x\beta}}{s+\alpha_x}\tilde{\beta}(s)$$

$$\tilde{y}(s) = \frac{k_{yx}k_{x\beta}}{(s+\alpha_y)(s+\alpha_x)}\tilde{\beta}(s).$$

We could write this as

$$\tilde{y}(s) = \tilde{H}_0(s)\tilde{\beta}(s)$$

where $\tilde{H}_0(s)$ is the feedback-free response,

$$\tilde{H}_0(s) = \frac{k_{yx}k_{x\beta}}{(s+\alpha_y)(s+\alpha_x)}.$$

The feedback-free gain is

$$G_0 = \tilde{H}_0(s)|_{s=0} = k_{yx}k_{x\beta}/\alpha_y\alpha_x.$$

The feedback-free mean time of response is

$$\bar{t}_0 = \alpha_y^{-1} + \alpha_x^{-1}$$

.

Now we will add feedback through a signaling protein z that is activated by y and in turn can activate x,

$$\dot{x}(t) = k_{x\beta}\beta(t) + k_{xz}z(t) - \alpha_x x(t)$$

$$\dot{y}(t) = k_{yx}x(t) - \alpha_y y(t)$$

$$\dot{z}(t) = k_{zy}y(t) - \alpha_z z(t)$$

This is a standard algebra problem where we solve for y then substitute back in to solve for x. For convenience, we will abbreviate the notation because we will stay in Laplace space, with x representing $\tilde{x}(s)$, H_0 representing $\tilde{H}_0(s)$, and so on.

$$x = \frac{k_{x\beta}}{s+\alpha_x}\beta + \frac{k_{xz}}{s+\alpha_x}z$$

$$y = \frac{k_{yx}}{s+\alpha_y}x$$

$$z = \frac{k_{zy}}{s+\alpha_z}y = \frac{k_{zy}k_{yx}}{(s+\alpha_z)(s+\alpha_y)}x.$$

Substituting back into x,

$$x = \frac{k_{x\beta}}{s+\alpha_x}\beta + \frac{k_{xz}k_{zy}k_{yx}}{(s+\alpha_z)(s+\alpha_y)(s+\alpha_x)}x.$$

For more convenience, write the response function for loop through the system as

$$H_L = \frac{k_{xz}k_{zy}k_{yx}}{(s+\alpha_z)(s+\alpha_y)(s+\alpha_x)}.$$

Then

$$x = \frac{k_{x\beta}}{s+\alpha_x}\beta + H_L x = \frac{1}{1-H_L}\frac{k_{x\beta}}{s+\alpha_x}\beta$$

$$y = \frac{1}{1-H_L}\frac{k_{yx}k_{x\beta}}{(s+\alpha_y)(s+\alpha_x)}\beta = \frac{H_0}{1-H_L}\beta.$$

The full response with feedback is $H = H_0/(1-H_L)$.

The gain with feedback is

$$G = G_0/(1-G_L)$$

where G_L is the gain for one round trip through the feedback loop,

$$G_L = \tilde{H}_L(s)|_{s=0} = k_{xz}k_{zy}k_{yx}/\alpha_z\alpha_y\alpha_x.$$

As feedback increases, measured as the magnitude of the feedback loop gain G_L, the gain increases, provided $G_L < 1$. When $G_L \to 1$, $G \to \infty$. For larger values of the feedback loop gain, G is negative. In the physical system, the gain neither diverges nor goes negative. Instead, when the feedback is strong, we are no longer in the regime of linear models. When the gain through the loop is greater than 1, the system becomes locked in the fully active state with all of the signaling protein activated. It is better described as a toggle switch with on and off states rather than a continuous distribution. We will work on this type of non-linearity in the next major section of the course. You could really save on coffee this way; you'd have one cup and you'd be activated for life. The drawback is that you wouldn't deactivate.

We could also write the response function as

$$H = H_0(1 + H_L + H_L^2 + \ldots),$$

with one factor of H_L for each time through the loop.

The mean time of response including feedback is

$$\bar{t} = (-d/ds)\ln H_0/(1-H_L)|_{s=0} = \bar{t}_0 + (d/ds)\ln(1-H_L)|_{s=0}$$

$$\bar{t} = \bar{t}_0 + (1-H_L)^{-1}(-d/ds)H_L|_{s=0}$$

$$\bar{t} = \bar{t}_0 + (1-H_L)^{-1}H_L(-d/ds)\ln H_L|_{s=0}$$

$$\bar{t} = \bar{t}_0 + (1-G_L)^{-1}G_L\bar{t}_L$$

where $\bar{t}_L$ is the mean time for one full trip through the feedback loop,

$$\bar{t}_L = 1/\alpha_y + 1/\alpha_z.$$

An important outcome is that the production rates k_{zy}, k_{yz}, and k_{xz} all contribute to the mean time of response. This is the first time that we've seen a production rate affect a time scale. Up until now, production rates have only contributed to gains. Calculations of mean square duration are left as a homework exercise.

We can also investigate negative feedback in the linear regime. Suppose that protein z in its active form binds tightly to the signaling molecule β, inhibiting its activity. In this case the activation of x is proportional to $\beta(t) - z(t)$. The corresponding dynamics are

$$\dot{x}(t) = k_{x\beta}[\beta(t) - z(t)] - \alpha_x x(t)$$

$$\dot{y}(t) = k_{yx}y(t) - \alpha_y y(t)$$

$$\dot{z}(t) = k_{zy}y(t) - \alpha_z z(t).$$

The Laplace-space solutions are

$$x = \frac{k_{x\beta}}{s+\alpha_x}\beta - \frac{k_{x\beta}}{s+\alpha_x}z$$

$$y = \frac{k_{yx}}{s+\alpha_y}x$$

$$z = \frac{k_{zy}}{s+\alpha_z}y = \frac{k_{zy}k_{yx}}{(s+\alpha_z)(s+\alpha_y)}x.$$

Substituting back in for x,

$$x = \frac{k_{x\beta}}{s+\alpha_x}\beta - \frac{k_{x\beta}k_{zy}k_{yx}}{(s+\alpha_z)(s+\alpha_y)(s+\alpha_x)}x$$

The loop feedback response H_L in this case is

$$H_L = \frac{k_{x\beta}k_{zy}k_{yx}}{(s+\alpha_z)(s+\alpha_y)(s+\alpha_x)}.$$

It differs from the positive feedback case because the inhibition of x by z shares the same parameter $k_{x\beta}$ as the activation of x by the input. The gain through the feedback loop is

$$G_L = \frac{k_{x\beta}k_{zy}k_{yx}}{\alpha_z\alpha_y\alpha_x}.$$

The solution for y in terms of H_L is

$$y = \frac{1}{1+H_L}\frac{k_{yx}k_{x\beta}}{(s+\alpha_y)(s+\alpha_x)}\beta = \frac{H_0}{1+H_L}\beta.$$

The response function for negative feedback is

$$H = \frac{H_0}{1+H_L}.$$

We no longer have the instability for large gain. The feedback loop gain can increase without bound and the system is still stable, although the output is reduced. The gain with feedback is

$$G = G_0/(1+G_L),$$

which goes to 0 as the feedback loop gain increases. The mean response time is

$$\bar{t} = \lim_{s\to 0}(-d/ds)\ln[H_0/(1+H_L)]$$

$$\bar{t} = \bar{t}_0 + \lim_{s\to 0}(d/ds)\ln(1+H_L)$$

$$\bar{t} = \bar{t}_0 + \lim_{s\to 0}(1+H_L)^{-1}(d/ds)H_L$$

$$\bar{t} = \bar{t}_0 + \lim_{s\to 0}(1+H_L)^{-1}H_L(d/ds)\ln H_L$$

$$\bar{t} = \bar{t}_0 - \frac{G_L}{1+G_L}\bar{t}_L$$

where, as before,

$$\bar{t}_L = \alpha_x^{-1} + \alpha_y^{-1} + \alpha_z^{-1}.$$

Here, increasing the feedback gain decreases the response time, provided that the linear response approximation holds. Notice that $\bar{t}_L > \bar{t}_0$, which means that for sufficiently large gain the linear model gives a negative response time. This of course doesn't happen in the real world. Instead, the formal solution probably gives $z(t) > x(t)$ and the signal $x(t)$ may become negative.

Part II

Cells as Non-linear Systems

Chapter 7

Information Content and Gene Regulation

This section of the course will describe non-linear aspects of cellular behavior, using transcription and translation as the basic model. We will investigate two types of non-linearity: saturation and cooperative response. Saturation arises from the finite limits of cells. The main limiting resource we will consider is the copy number of genes. Many model organisms are haploids, containing a single copy of most genes. Humans and other mammals are diploid, containing two copies of each gene. When the copies are fully transcriptional active, the transcriptional response can't increase any further.

Actually, this is mostly true but not all the way true. Proteins that are in high demand, for example histones that serve as spools for DNA to wrap around, exist as multiple copy genes. Genes that encode tRNA's are also present at multiple copies. But most genes are one copy per haploid genome.

Cooperativity is a completely different type of non-linearity. In a linear system, any input creates a proportional output. A signal's a signal, no matter how small. A cooperative system has a threshold. An ideal cooperative system works as an analog-to-binary converter. Inputs below a threshold result in no output; inputs above the threshold result in full, saturating output.

These systems are still deterministic, and we will work towards deterministic models of time-dependent outputs given time-dependent inputs. First we will start with a biophysics and information theory analysis of gene regulation by transcription factors.

Our signal transduction model ends with a cliff-hanger. A transcription factor has just translocated to the nucleus. How does it recognize its cognate regulatory element? Without knowing anything about the physical mechanism, which we'll look at later, we can make predictions about how it happens. We will think about a transcription factor (TF) that activates a single gene. As a quick backstory, a TF can activate transcription by binding to a regulatory element called a promoter located $5'$ to the transcription start. TF's include transcriptional activation domains that recruit the rest of the transcriptional apparatus. If a TF binds promiscuously, then transcription will

occur too many places. In fact there seems to be quite a bit of noisy transcription, and it's active research to determine whether certain transcripts have biological function or whether they're just the result of transcriptional noise. Other TFs act as repressors. Other TFs bind to regulatory elements called enhancers that are far away from the genes they regulate in DNA primary sequence, but which can be close in three dimensional space due to DNA looping.

In any event, suppose that a TF is supposed to bind at a single location in a genome. A reasonable mechanism is that the TF recognizes a specific DNA sequence that occurs upstream of the gene, called a motif. How long must this motif be to ensure that it only occurs upstream of our desired gene, and nowhere else?

For this we need a model of DNA sequences. We will take a very simple model of independent uniform probability of each of the 4 nucleotides at each position in the genome. Now suppose that the TF binds to a motif with width W and the total genome has G nucleotides. What is the probability that the motif occurs in just the desired location? Or described differently, what is the probability that a specific motif of width W occurs k times in the genome?

We begin with the probability that a particular location matches the motif by chance. Since each position in the genome is random, the probability of a match is $1/4$ for each position. The joint probability is $(1/4)^W$, which decreases exponentially with width W. There are G possible starting locations for the motif. If you're a stickler you might argue that there are at most $G-W$ starting positions in a linear position, we have two strands to worry about, what about low-complexity regions like centromeres, telomeres, and repeats, and even what about heterochromatin vs. euchromatin. We will see that these details won't be overly important.

Our next important assumption is that this probability is that each of the G possible start locations has the same probability $(1/4)^W$ of a match. Here you could argue that knowing the sequence at one location sets constraints on the possible sequences at an overlapping window. But it turns out we can really take each position to have an independent identical probability of a match. The probability of k matches is therefore the binomial probability

$$P(k) = C(G,k)p^k(1-p)^{G-k}$$

where p is the per-site success probability,

$$p = (1/4)^W.$$

Since p is small, we can use the Taylor series approximation $e^x \approx 1+x$ to replace $(1-p)^{G-k}$ with $e^{p(G-k)}$. Also, since G is very big and k is probably much smaller, we will ignore terms of order k/G and write this as e^{pG}. Also note that the product pG is the success probability per site times the number of sites, which equals the expected number of occurrences. We'll call this λ because everybody else does.

The combinatorial prefactor is

$$C(G,k) = \frac{G \times (G-1) \times (G-2) \ldots (G-k+1)}{1 \times 2 \times 3 \ldots \times k}.$$

We can write the numerator as $G^k \prod_{j=0}^{k-1}(1-j/G)$. Since we are ignoring terms of order k/G and $j \leq k$, we might as well ignore j/G also. This means that we can make the approximation

$$C(G,k) \approx G^k/k!.$$

Putting it together,

$$P(k) \approx (G^k p^k/k!)e^{-\lambda} = (\lambda^k/k!)e^{-\lambda},$$

our old friend the Poisson distribution. Most people do not think that the Poisson distribution is named after the probability of drawing a winning card in the game "Go Fish".

A reasonable constraint is the expected number of random occurrences of the motif is less than 1, $\lambda < 1$, or $G/4^W < 1$. This means that

$$4^W > G,$$

$$W > \log_4 G.$$

We are only allowed to calculate logarithms using 3 bases, in descending order base 10, base e, and base 2. Therefore we instead write

$$2^{2W} > G$$

$$W > (1/2)\log_2 G.$$

What does this mean for a typical genome? Typical genome sizes are viruses $\sim 10^4$ nucleotides, bacteria and yeast $\sim 10^7$ nucleotides, and metazoans (multi-celled animals including nematodes, fruit flies, and animals) $\sim 10^9$ nucleotides. Recalling that $\log_2 10 \approx 3.3 \approx 10/3$, we predict that regulatory motifs are about 6-7 nt in viruses, about 70/6 = 12 nt in bacteria and yeast, and about 90/6 = 16 nt in human and other metazoans.

Now, how does recognition happen? We will look at structures of TF-DNA complexes and see that most families of TFs recognize DNA sequences by having an α-helix that sits in the DNA major groove. Side-chains of the TF extend into the major groove and make hydrogen bonds with specific DNA basepairs. For some TF families, people have attempted to develop recognition codes that define which side-chains to use for which DNA basepairs. You will recall that one turn of DNA corresponds to about 10 bp. Looking at structures, though, it's not really possible for the α-helix to make contact with the entire major groove. Instead, at most about 6 bp are recognized. Therefore we predict that a viral TF will have 1 helix, a bacterial or yeast TF will have 2-3 helices, and a human TF will have 3-4 helices.

As it turns out, these predictions motivated purely by information theory and DNA-protein structure are correct. The only major difference is that the multiple recognition helices are not always provided by a single protein. Instead, multiple TFs provide individual helices that together provide the full number of basepairs of recognition for specific transcription. A notable exception are the zinc finger transcription factors, in which multiple α-helices are presented on a single chain. This also happens to be the largest family of mammalian TFs. In plants, the largest families involve binding by β-sheets rather than α-helices, but the theory is the same.

When multiple α-helices must bind to a DNA sequence, we have the possibility of cooperative binding, where binding of one helix makes binding of the next helix more likely.

When multiple TFs must come together for transcriptional activation, we have possibilities of homodimers involving multiple copies of the same TF and heterodimers where different TFs must be present simultaneously. We also have the possibility of TFs that recruit transcriptional repression complexes rather than transcriptional activation complexes. These scenarios give the possibility of combinatorial regulation through mix-and-match pairing of TFs. Homeodomain TFs and leucine zipper TFs bind in mix-and-match combinations. Multi-TF complexes also permit logic-like functions based on different subsets of TFs that lead to activation vs. repression. We will of course cover these topics in future chapters.

Chapter 8

Saturation and Cooperative Response

Chapter 9

Joint Models of Transcription and Translation

Chapter 10

Positive and Negative Auto-Regulation

In many textbooks and research papers we see diagrams of biological pathways with full page, 8x10 color glossy pictures with circles and arrows and a paragraph underneath each one explaining what each one is. Today we will work through the meaning of positive and negative arrows in gene regulatory networks.

The best example of pictures like these are from work on early embryonic development by Eric Davidson, a professor at CalTech (1937-2015).

We will start with models for positive auto-regulation. For these models, we will call our favorite transcription factor X. It is under control of a signaling pathway whose output is S because signal start with the letter 's'. Don't get confused between S for input signal and s for Laplace transform variable. The dynamics without feedback are given by

$$\dot{X}(t) = \beta\Theta[S(t) > K] - \alpha X(t).$$

We will typically solve for dynamics in regions where the logic function Θ has constant value. Here we use boundary conditions that $S(t) = 0$ for $t < 0$ and $S(t) > K$ for $t > 0$. The exact value of $S(t)$ doesn't matter, as long as it is above threshold. For this system,

$$\dot{X}(t) = \beta - \alpha X(t)$$

$$\tilde{X}(s) = \beta/s(s+\alpha)$$

$$X(t) = (\beta/\alpha)(1 - e^{-\alpha t}).$$

We can summarize the response with the gain G and the mean time of response τ, which we will call the response time for short,

$$G = \lim_{s\to 0} \tilde{X}(s)/\tilde{\Theta}(s) = \lim_{s\to 0} [\beta/s(s+\alpha)]/(1/s) = \beta/\alpha.$$

$$\tau = \lim_{s\to 0} (-d/ds) \ln[\tilde{X}(s)/\tilde{\Theta}(s)] = (-d/ds) \ln[\beta/(s+\alpha)]|_{s=0} = 1/\alpha.$$

Now we add positive feedback. Since X is a transcription factor, if a regulatory element for X exists in front of its own promoter, it can bind to its promoter and activate transcription. We will think about an additive model in which the signal and the transcription factor contribute additively to activation,

$$\dot{X}(t) = \beta_S \theta[S(t) > K_S] - \alpha X(t) + \beta_X (X/K_X)^n / [1 + (X/K_X)^n].$$

We've added subscripts to the production rates and the equilibrium constants to keep everything organized. We'll work separately with two limits, a weak limit and a strong limit.

In the weak limit, X only adds a little bit to its transcription. It has a weak binding constant for its own promoter, which means that K_X is large and in particular is much larger than the abundance of X ever reaches. In this case, $1 + (X/K_X)^n \approx 1$, and

$$\dot{X}(t) \approx \beta_S \theta[S(t) > K_S] - \alpha X(t) + \beta_X (X/K_X)^n.$$

So that we can solve this with Laplace transforms we'll make a further assumption that X binds weakly as a monomer, so that $n \approx 1$. We end with a model for weak self-activation,

$$\dot{X}(t) \approx \beta_S \theta[S(t) > K_S] - \alpha X(t) + \beta_X X/K_X.$$

Rearranging terms,

$$\dot{X}(t) \approx \beta_S \theta[S(t) > K_S] - [\alpha - (\beta_X/K_X)]X(t).$$

We see that this looks just like our original feedback-free model with a reduced degradation rate α',

$$\alpha' \equiv \alpha - (\beta_X/K_X).$$

We can immediately see that the gain G' with feedback is larger than the gain without feedback,

$$G'/G = \alpha/\alpha' = 1/[1 - (\beta_X/\alpha K_X)],$$

and the new response time τ' is longer as well, $\tau'/\tau = \alpha/\alpha'$.

We can also think about where this approximation breaks down. We run into trouble when $\alpha' \leq 0$. This condition is that

$$0 \geq \alpha - \beta_X/K_X.$$

Rearranging terms,

$$\beta_X/\alpha \geq K_X.$$

Interpret the left-hand side as the steady-state abundance of X when it drives its own transcription with production rate β_X. If the steady-state concentration is above K_X, then our approximation to the Hill function breaks down because $1 + (X/K_X)^n \approx (X/K_X)^n$, rather than our weak activation assumption that $1 + (X/K_X)^n \approx 1$.

We can also think about the violation of the assumption in terms of the decay rate α'. When α' is positive, it represents a decay rate. When it becomes negative, it represents a growth rate. Our

result for $\alpha' < 0$ is that the transcription factor concentration grows exponentially. This also means that we violate the assumption that $X < K_X$.

In many real systems, we do have $\beta_X/\alpha \geq K_X$. This is the condition for strong activation. With strong activation, we use a Hill function or a logic function to approximate the auto-activation of X,

$$\dot{X}(t) \approx \beta_S \theta[S(t) > K_S] - \alpha X(t) + \beta_X \Theta[X(t) > K_X].$$

Our standard approach, again, is to solve for dynamics in regimes where the logic functions are constant, and to match at the boundaries. We start with the system off for $t < 0$ and with $S(t)$ high for $t > 0$. In regime I, $X(t) < K_X$, and

$$X(t) = (\beta_S/\alpha)[1 - e^{-\alpha t}].$$

This regime continues until $X(t) = K_X$ and the second logic function is triggered. We will call this time t_X,

$$K_X = (\beta_S/\alpha)[1 - e^{-\alpha t_X}]$$

$$K_X \alpha/\beta_S = [1 - e^{-\alpha t_X}]$$

$$t_X = -(1/\alpha)\ln[1 - K_X/(\beta_S/\alpha)].$$

After t_X, the total production rate is $\beta' = \beta_S + \beta_X$. Thinking through the time-domain solution, this is like an initial value problem where the initial value of X is K_X rather than 0. The solution is

$$X(t) = (\beta'/\alpha)[1 - e^{-\alpha(t-t_X)}] + e^{-\alpha(t-t_X)} K_X.$$

We could also think about resetting the origin of time to t_X. The Laplace space solution is

$$\tilde{X}(s) = \beta'/s(s+\alpha) + K_X/(s+\alpha).$$

Now we will think about the gain and the response time. It looks like the gain is $G' = \beta'/\alpha = G + \beta_X/\alpha$ and the response time is $\tau' = 1/\alpha = \tau$. This calculation is based on a signal $S(t)$ that remains high for all time. Consider, however, a square wave input,

$$S(t) = K_S \Theta(0 \leq t \leq t_X).$$

Here we've been a little sloppy in calling S the result of the logic function rather than the input to the logic function. This input has Laplace transform

$$\tilde{S}(s) = \int_0^{t_X} dt\, K_S e^{-st} = (K_S/s)(1 - e^{-st_X}).$$

The output $X(t)$ subsequence to time t_X is due to production rate β_X alone,

$$X(t) = (\beta_X/\alpha)[1 - e^{-\alpha(t-t_X)}] + e^{-\alpha(t-t_X)} K_X.$$

The gain G' is

$$G' = \lim_{s\to 0} \tilde{X}(s)/\tilde{S}(s).$$

We know that $\tilde{X}(s) = (\beta_X/\alpha)/s \to \infty$ in this limit. For the input signal, though,

$$\lim_{s\to 0} \tilde{S}(s) = (K_s/s)(1 - 1 + st_X) = K_s t_X,$$

which is finite. We conclude that $G' \to \infty$ for strong feedback.

Similarly, left for an exercise, show that $\tau' \to \infty$ for strong feedback.

For negative feedback, we think of a model where the transcription factor binds downstream of the signaling pathway output. We will model the binding state of the signaling pathway and X as independent. The probability that the promoter is bound by the activating signaling protein is $\Theta[S(t) > K_S]$, a logic function. The probability that the promoter is not bound by the transcription factor X is $1/[1 + (X/K_X)^n]$. Remember that before we were using the bound probability, which is $(X/K_X)^n/[1 + (X/K_X)^n]$. You can see that the bound probability plus the unbound probability equals 1. Transcription in this model requires that the activator is bound and X is not bound,

$$\dot{X}(t) = \beta\Theta[S(t) > K_X]\frac{1}{1 + (X/K_X)^n} - \alpha X(t).$$

For simplicity, we will approximate the repressor Hill function as a logic function,

$$\frac{1}{1 + (X/K_X)^n} \approx \Theta(X < K_X).$$

The behavior exactly at $X = K_X$ isn't well defined. It could be 0, $1/2$, or 1. This just shows that the logic function isn't as realistic a model at the Hill function, which has smooth behavior through the threshold value K_X. In turn, the Hill function is also an approximation because it treats the discrete number of transcription factors and promoters as continuous variables. Our next major section will look at the effects of discrete particle numbers.

Returning to the model for negative auto-regulation, the dynamics with the logic function are

$$\dot{X}(t) = \beta\Theta[S(t) > K_X]\Theta(X < K_X) - \alpha X(t).$$

We follow the standard approach for nonlinear dynamics with logic functions. We analyze each linear regime where logic functions have a constant value, and then switch to a new linear regime when a logic function switches states.

We will think about a system prepared with $X(t) = S(t) = 0$ for $t < 0$ and then $S(t) > K_X$ for $t > 0$. In Regime I, both logic functions are on, and the dynamics are

$$\dot{X}(t) = \beta - \alpha X(t).$$

We know the solution in this regime is

$$X(t) = (\beta/\alpha)[1 - e^{-\alpha t}].$$

These dynamics continue until Regime II when $X > K_X$ and X represses itself. Regime II begins at the time t_X when $X > K_X$. We previously solved for this time because we had the same threshold problem for strong positive auto-regulation. The solution is

$$t_X = -(1/\alpha)\ln[1 - K_X/(\beta/\alpha)].$$

Just after time t_X, the dynamics switch to

$$\dot{X}(t) = -\alpha X(t),$$

and X begins to decay. As soon as the decay begins, however, X falls back below K_X, and the dynamics return to regime I,

$$\dot{X}(t) = \beta - \alpha X(t).$$

Then X grows again and switches itself off.

If we sent these dynamics to an ODE solver, the discontinuity in the production rate of X at the threshold value $X = K_X$ would cause problems. The smoother Hill function isn't discontinuous and would lead to an easier time for the solver. In the end, what we have for $t > t_X$ is that X stays at its threshold value, $X(t) = K_X$ for $t > t_K$. This is an example of feedback control. Negative auto-regulation is controlling the concentration of the transcription factor X.

Now we will think about gain and response time. We no longer have a linear system because the output is no longer linearly dependent on the input. A more natural characterization of the gain of the system is the plateau value, K_X rather than β/α. The plateau value depends on a thermodynamics binding constant rather than production and degradation rates.

For non-linear systems, the definitions of gain and response time aren't as defined as for linear systems. For a linear system, the response function is independent of the input. For a non-linear system, though, the response function depends on the input. We saw this for the strong auto-activation, where an input below a threshold duration does not switch the system into a stable active state, giving a finite gain, whereas an input longer than threshold yields an infinite gain. There is no standard formal definition such as $\lim_{s\to 0}\tilde{H}(s)$ for gain or $\lim_{s\to 0}(-d/ds)\ln\tilde{H}(s)$ for response time. Instead it can be more meaningful to define a practical measurement. For gain, the steady-state amplitude or plateau value (if one exists) is a natural choice. Because saturation of the promoter can make the plateau value independent of the input signal, it may not make sense to normalize by the input strength. Instead we often talk about strong vs. weak promoters. For the same level of input transcription factor, a strong promoter gives a higher transcription rate represented by the parameter β in our model.

For response time, a reasonable characterization is just the time that it takes the system to reach some fraction of the plateau value. Often the time to reach half the maximum is chosen and called $t_{1/2}$. This leads to annoying factors of $\ln 2$ in the response time. To avoid the factor of $\ln 2$, we could instead describe the response time as the time for the slowest transient to have decayed to $1/e$. This choice gives the same response time of $1/\alpha$ for linear response functions like $H(t) = e^{-\alpha t}$.

With negative auto-regulation, for simplicity we will define the response time τ as the time to reach the plateau value, we have

$$\tau = -(1/\alpha)\ln[1 - K_X/(\beta/\alpha)].$$

The response time depends on the threshold and the production rate, in addition to the usual dependence on the degradation rate. If the threshold is low compared to the feedback-free plateau, though, then we can make an additional approximation,

$$\ln[1 - K_X/(\beta/\alpha)] \approx -K_X/(\beta/\alpha)].$$

Then multiplying we find that

$$\tau \approx K_X/\beta.$$

In the low threshold regime, the response time doesn't depend on the degradation rate at all. Instead, it depends on the threshold and the production rate. Increasing the production rate increases the response.

As a quick check on why this makes sense, think about the threshold being so low that degradation doesn't happen. The early dynamics are just

$$\dot{X}(t) = \beta,$$

giving $X(t) = \beta t$. Solving for $X(t) = K_X$ we have $t = K_X/\beta$.

Chapter 11

Non-Linear Cascades and Logic Gates

Now that we have worked on the dynamics of a single transcription factor and on positive and negative feedback, we will work through some simple calculations for cascades. First, we consider our standard linear cascade with no loops:

$$\dot{X}_1(t) = \beta_1 \theta[S(t) > K_1] - \alpha_1 X_1(t)$$

$$\dot{X}_2(t) = \beta_2 \theta[X_1 > K_2] - \alpha_2 X_2(t)$$

$$\cdots$$

$$\dot{X}_n(t) = \beta_n \theta[X_{n-1} > K_n] - \alpha_n X_n(t).$$

We typically consider three types of input: the 'on' step, where all variables are 0 for $t < 0$ and then $S(t)$ is in a high state for $t > 0$; the 'off' step, where all variables are in their high states at $t = 0$ and then $S(t) = 0$ for $t > 0$; and a pulse input where $S(t)$ is a square wave of finite duration.

For the on-step, we simple have each transcription factor activated in turn. Transcription factor n is inactive for times $t < t_n$. After t_n, it follows dynamics

$$X_n(t) = (\beta/\alpha)[1 - e^{-\alpha(t-t_n)}].$$

The delay t_n is 0 for the X_0. For subsequent transcription factors,

$$t_n = \Delta t_n + t_{n-1},$$

where the incremental delay is

$$\Delta t_n = -(1/\alpha_{n-1}) \ln[1 - K_{n-1}/(\beta_{n-1}/\alpha_{n-1}].$$

The steady-state value of each transcription factor is β_n/α_n.

Compared to our results for a linear systems cascade, the system is similar in that the activation time is linear through the network. It is different in that there is no multiplicative amplification or gain. Instead, saturation at each step insulates the steady-state protein levels from each other.

For the off-step, the system begins with each transcription factor at level $X_n = \beta_n/\alpha_n$. When the signal turns off, the message propagates through the signaling network. The factor X_n begins to turn off at time t_n with dynamics

$$X_n(t) = (\beta_n/\alpha_n)e^{-\alpha(t-t_n)}.$$

The delay time $t_n = 0$ for $n = 1$. For subsequent factors, $t_n = \Delta t_n + t_{n-1}$, where the incremental delay is the solution to

$$(\beta/\alpha)e^{-\alpha\Delta t} = K,$$

$$\Delta t = (1/\alpha)\ln[(\beta/\alpha)/K].$$

Again we see the leading behavior of the response time depending on the degradation rate. Also notice that the delay times for the off-step are different from the on-step, another example of the difference between a non-linear system and a linear system.

- Linear cascade. Additive response time. Saturation insulates gain.
- Three-gene feed-forward networks with all positive interactions. And and Or logic. Different delays for on and off steps. Evolutionary selection as a low-pass filter.
- Three-gene feed-forward networks with positive and negative interactions. First-derivative generator or edge detector in time and space.

Chapter 12

Delta-Notch Signaling

Today we will look at a very special two-gene network. In this network, the two proteins X and Y are mutual repressors of each other's transcription. The dynamics are

$$\dot{X}(t) = \beta_X/[1+(Y/K_Y)^m] - \alpha X(t)$$

$$\dot{Y}(t) = \beta_Y/[1+(X/K_X)^n] - \alpha Y(t).$$

We make the reasonable assumption that the proteins have the same decay rate α. This isn't necessary, but it makes mathematical analysis easier. We can work with reduced variables $x \equiv X/K_X$ and $y \equiv Y/K_Y$, We can also define $t' = \alpha t$ so that $d/dt = (1/\alpha)(d/dt')$. We then have

$$\dot{x}(t') = (\beta_X/K_X\alpha)/(1+y^m) - x(t')$$

$$\dot{y}(t') = (\beta_Y/K_Y\alpha)/(1+x^m) - y(t').$$

Finally, we define the new production rates as $\beta_x = \beta_X/K_X\alpha)$ and $\beta_y = \beta_Y/K_Y\alpha$. For the rest of the discussion, we will assume that β_x and β_y are greater than 1 unless explicitly stated otherwise. Physically, these are the ratio of the plateau value of each protein in a feedback-free system to the abundance required to repress the other factor. We also call time t again instead of t'. Our system with reduced coordinates is

$$\dot{x}(t) = \beta_x/(1+y^m) - x(t)$$

$$\dot{y}(t) = \beta_y/(1+x^m) - y(t).$$

When m and n are large, we as usual approximate the Hill functions by logic functions,

$$\dot{x}(t) = \beta_x\Theta(y<1) - x(t)$$

$$\dot{y}(t) = \beta_y\Theta(x<1) - y(t).$$

What are the dynamics? Since this is a non-linear system, the response depends on the input, which we will represent as the initial preparation $x(0)$ and $y(0)$. Suppose we start the system with

only x present, and at time 0 permit y to be expressed. For example, y requires an additional activator, for example a doxycycline-responsive system. In this case, the initial state is $x(0) = \beta_x$ and $y(0) = 0$. As long as the plateau value $\beta_x > 1$, which means that x reaches its threshold to repress y, the initial dynamics are

$$\dot{x}(t) = \beta_x - x(t)$$

$$\dot{y}(t) = -y(t).$$

The initial state is the steady-state solution. Similarly, if we prepare the system with $x(0) = 0$ and $y(0) = \beta_y$, or more generally $x(0) < 1$ and $y(0) > 1$, and $\beta_y > 0$, we have dynamics

$$\dot{x}(t) = -x(t)$$

$$\dot{y}(t) = \beta_y - y(t)$$

with time-domain solution

$$x(t) = 0$$

$$y(t) = e^{-t}y(0) + (1 - e^{-t})\beta_y.$$

The system is attracted to the steady-state solution with $x = 0$, $y = \beta_y$. In general for the logic-function model we can show that there is a line that separates evolution to the state $x = \beta_x$, $y = 0$, which we call the x state, and the state $x = 0$, $y = \beta_y$, which we call the y state. If $\beta_x = \beta_y$, it is easy to show that the line $x = y$ separates the initial states that evolve to the x state, $x > y$ at time 0, from the initial states that evolve to the y state, $x < y$ at time 0. These two points are called fixed points because if the system is prepared at a fixed point (or if it reaches it), it stays there forever. They are stable fixed points if any point that starts near the fixed point evolves back to the fixed point. We'll be more formal about this later.

The line $x = y$ is called a separatrix because it separates the initial states that evolve towards the x vs. y fixed points. For general β_x and β_y, the calculation of the separatrix is left as an exercise.

What happens if the initial state is $x(0) = y(0) = 0$? The initial dynamics are

$$\dot{x} = \beta_x - x$$

$$\dot{y} = \beta_y - y$$

with solution

$$x(t) = \beta_x(1 - e^{-t})$$

$$y(t) = \beta_y(1 - e^{-t}).$$

If $\beta_x > \beta_y$, then x reaches its threshold first and we evolve to state x, and similarly for y. If $\beta_x = \beta_y$, though, then both transcription factors reach their thresholds simultaneously, $x(t) = y(t) = 1$. This looks like negative auto-regulation, with x and y switching in tandem between dynamics

$$\dot{x} = \beta - x, \dot{y} = \beta - y$$

and

$$\dot{x} = -x, \dot{y} = -y$$

with the result that $x(t) = y(t) = 1$ for all subsequent times. This is an unstable fixed point, though, because any small change to x or y results in one of the two proteins reaching its threshold slightly ahead, leading to evolution away from the unstable fixed point and towards the stable fixed point. In a real biological system, remember that the continuous variables x and y really refer to discrete numbers of proteins. It is easy to picture the discrete numbers being different, and in fact it is more difficult to picture a biological system that is so tightly controlled that x and y have exactly the same number of proteins even if they are essentially identical proteins expressed from identical promoters or plasmids. Therefore, early naturally occurring fluctuations in particle number lead to very different long-term states.

Would cells behave is such a horrifyingly random way? We wouldn't be studying this system otherwise. This is a standard mechanism for tissues to generate patterns of cell-type diversity with different cell compositions but no long-range gradient or pattern. For example, your skin has a mix of hair cells, sweat glands, and more typical epidermal cells. The fraction of each cell type is relatively constant in different patches of skin, but there is no long-range order. This is different from a crystal lattice where there is long-range order, or a perfect chess or checker board with alternating squares.

On average, Delta-Notch patterning ensures that hair cells won't be next to other hair cells. Variation in the genes responsible for patterning can shift the balance of the patterning, giving slightly more or fewer hair cells proportionally to other cells from animal to animal. This type of variation was observed in bristle counts of fruit flies. A phenotype with a numeric value is termed a quantitative trait, as opposed to a categorical phenotypes based on categories, such as case-control phenotypes for human disease. Back to bristle count, the phenotypic variation was used to identify genes whose alleles contributed to the differences in bristle count between flies. These genes were among the first members of the Delta-Notch signaling pathway to be discovered.

Chapter 13

Stability Analysis

Returning to our Delta-Notch model, our goal is to determine what happens close to the fixed point. The problem we have is that non-linear dynamics is difficult and in general has no closed-form solution. Dynamics can also be chaotic, with trajectories depending sensitively on initial conditions. Any finite-precision calculation will necessarily lead to errors in calculating a chaotic trajectory at sufficiently long time. Chaotic trajectories are a plot point in "The Three-Body Problem" by Cixin Liu (translated from the original Chinese by Ken Liu), a fantastic recent science fiction book. The translation of the final installment of the trilogy has just been published, and if you like science fiction you should read it.

In contrast to our inability to solve non-linear dynamics, we know how to solve for the dynamics of a linear system. I will try to be consistent in using regular math font to represent scalars, lowercase bold font to represent vectors, and uppercase bold font to represent matrices. The most general linear system defines the coordinates as a vector $\mathbf{r}(t)$ with time derivative $(d/dt)\mathbf{r}(t) \equiv \dot{\mathbf{r}}(t)$. In a linear system, the time derivate depends linearly on the coordinates,

$$\dot{r}_i(t) = \sum_j k_{ij} r_j(t),$$

where $r_i(t)$ is one of the components of the coordinate vector. In matrix form,

$$\dot{\mathbf{r}}(t) = \mathbf{K}\mathbf{r}(t).$$

If this were a scalar equation,

$$\dot{r}(t) = kr(t),$$

we would write down the solution by inspection,

$$r(t) = e^{kt} r(0).$$

For a matrix equation, we can use the same solution,

$$\dot{\mathbf{r}}(t) = e^{\mathbf{K}\mathbf{t}} \mathbf{r}(0).$$

You can validate this solution by taking the time derivative of the right-hand side.

We've left out an explicit constant term in the dynamics, for example

$$\dot{\mathbf{r}}(t) = \mathbf{a} + \mathbf{K}\mathbf{r}(t).$$

We don't really need one because we could introduce a constant component, $r_0(t) = 1$ with $\dot{r}_0(t) = 0$, and then a_i become k_{i0} and $k_{00} = 0$.

We will go one step further by considering what happens to an initial point $\mathbf{r}(0)$ by representing this point as an expansion over eigenvalues of the matrix $\mathbf{K}$. Formally,

$$\mathbf{r}(0) = \sum_{\lambda} c_\lambda \mathbf{u}_\lambda,$$

where $\mathbf{u}_\lambda$ is an eigenvalue of $\mathbf{K}$ with eigenvalue λ,

$$\mathbf{K}\mathbf{u}_\lambda = \lambda \mathbf{u}_\lambda.$$

You should remember from linear algebra that if $\mathbf{K}$ has full rank, then any initial coordinate can be written as a linear combination of eigenvectors. We'll accept this rather than proving it.

We can solve for the time evolution of a single eigenvector. If the initial coordinate is a pure eigenvector, $\mathbf{r}(0) = \mathbf{u}(0)$, then

$$\mathbf{r}(t) = e^{\mathbf{K}t}\mathbf{u}_\lambda.$$

In linear algebra we learn how to add and multiple matrices, but often we don't learn what the exponential of a matrix means. As usual, when we see a function of a matrix, we interpret it using a series expansion,

$$e^{\mathbf{K}t} = \sum_{n=0}^{\infty} \mathbf{K}^n t^n / n!.$$

If this operator is acting on an eigenvalue, the matrix product is simple to calculate:

$$\mathbf{K}u_\lambda = \lambda \mathbf{u}_\lambda$$

$$\mathbf{K}^2 u_\lambda = \lambda \mathbf{K}\mathbf{u}_\lambda = \lambda^2 \mathbf{u}_\lambda$$

$$\dots$$

$$\mathbf{K}^n u_\lambda = \lambda^n \mathbf{u}_\lambda.$$

The result is that the matrix products all become scalars. We can therefore calculate the operator acting on the eigenvector as

$$e^{\mathbf{K}t}\mathbf{u}_\lambda = \sum_{n=0}^{\infty} (\mathbf{K}^n t^n / n!)\mathbf{u}_\lambda = \sum_{n=0}^{\infty} (\lambda^n t^n / n!)\mathbf{u}_\lambda = e^{\lambda t}\mathbf{u}.$$

In the last step, we roll up the series back into the scalar exponential $e^{\lambda t}$.

Note that this works in general when a function of a matrix can be expressed as a power series of that matrix, $F(\mathbf{K})\mathbf{u}_\lambda = F(\lambda)\mathbf{u}_\lambda$.

Returning to our problem, we now have the exact solution

$$\mathbf{r}(t) = \sum_\lambda c_\lambda e^{\lambda t} u_\lambda .$$

We can characterize our linear system by considering what happens to trajectories as $t \to 0$.

stable If all eigenvalues have negative real part, then each trajectory decays to the stable fixed point at the origin and the system is stable.

unstable If at least one eigenvalue has a positive real part, then trajectories are unstable along this direction and the system is unstable.

periodic If some eigenvalues have negative real part and others have zero real part, then the system is periodic with period given by the imaginary part of the eigenvalues with zero real part.

Stability analysis for the Delta-Notch system

- Calculation of Jacobian
- Calculation of eigenvalues
- Conditions on Hill coefficients

We can think about this system as similar to a chemical kinetics problem with two stable states and a transition state. A reasonable reaction coordinate z, particularly for a symmetric Delta-Notch system, is $z = x - y$, with the symmetric fixed point at $z = 0$. We can think about the probability $P(z)$ that the system has a particular value of z and then define a surface like an energy surface as $E(z) = -\ln P(z)$. When the Hill coefficients are smaller than the critical value, the surface $E(z)$ has a single minimum at $z = 0$ and climbs on either side. By symmetry, $dE(z)/dz = 0$ at $z = 0$, making the $z = 0$ a minimum (stable fixed point) or maximum (unstable fixed point) of the energy function. We then have to look at $(d/dz)^2 E(z)$ at $z = 0$ to determine whether the symmetric fixed point is a maximum or a minimum.

When the Hill coefficients are below the critical value, $z = 0$ is a stable fixed point corresponding to an energy minimum, $(d/dz)^2 E(z)|_{z=0} > 0$. When the Hill coefficients are exactly at the critical value, $(d/dz)^2 E(z)|_{z=0} = 0$ and the energy function is very flat at $z = 0$. Above the critical value, $(d/dz)^2 E(z)|_{z=0} < 0$ and a barrier begins to grow from the center, creating the two stable wells on either side. The ability to pattern requires the barrier to be sufficiently high that the rate of barrier crossing is small. Even if the system is formally stable according to stability analysis, we might still have to rapid a rate of transitions between the stable states due to natural variation in the number of molecules of each type within a cell.

From chemical kinetics, we can intuit that the rate of barrier crossing k depends on the height of the barrier, $k \sim \nu \exp(-E^\star/k_B T)$ where $E^\star$ is the activation energy and $k_B T$ is Boltzmann's

constant times temperature. The factor $E^\star/k_BT$ compares the barrier height to the size of energy fluctuations at thermal equilibrium.

For our system, a similar expression for the barrier crossing rate is

$$k \sim \nu \exp[-(z^\star - \mu)\sigma_z]$$

where $1/\nu$ is a typical timescale for cellular processes, $z^\star$ and μ are the values of $x-y$ at the barrier and at the stable state, and σ_z is the size of typical fluctuations in z. We already know that $z^\star = 0$ by symmetry. The stable states have $\mu = x - y \approx \beta/\alpha$ since $x = \beta/\alpha$ and $y = 0$ at the stable state. We've learned that the main timescale for dynamics is $1/\alpha$. This means that the barrier crossing rate is $k \sim \alpha \exp[-(\beta/\alpha)/\sigma_z]$ where σ_z is the size of typical fluctuations in protein copy number. If we want stable patterning, we want the rate to be small and therefore $\beta/\alpha > \sigma_z$. Unfortunately, we don't know what this is because we haven't created a stochastic model. Even for a simple model of a single protein, $\dot{x} = \beta - \alpha x$, we don't know how σ for depends on α and β for an equivalent stochastic model. Our next step will be to derive a model for fluctuations at equilibrium to answer this practical question about patterning.

Part III

Cells as Stochastic Systems

Chapter 14

Noise in Gene and Protein Expression

We have been discussing stochastic cell fate selection by Delta-Notch signaling. We've decided that if the cooperativity is large enough, the symmetric fixed point is unstable and instead we have two stable states. We still don't know how to model the system, though, in a practical sense. For example, imaging that the X and Y dynamics are symmetric,

$$\dot{X} = \beta/[1+(Y/K)^n] - \alpha X$$

$$\dot{Y} = \beta/[1+(X/K)^n] - \alpha Y.$$

If we start the system at time 0 with $X = Y = 0$ and use a standard ODE solver, we get to the unstable fixed point at remain there. How do we model the system in a realistic way?

Remember that in the real system, proteins aren't produced in fractional amounts. We have almost step-wise transitions between n and $n+1$ copies of a protein. We also won't have exact synchronization in production of X and Y proteins. Therefore, just by chance, one of the two species will go above threshold first, and the cell will harness these fluctuations to achieve patterning. Our ODE model doesn't represent this at all because it changes smoothly from n to $n+1$.

To build a better model, we return to the simple model for a single protein,

$$\dot{X} = \beta - \alpha X.$$

We can think about this as describing a random process. If we have n copies of the protein in a cell, the rate that the system changes to $n+1$ proteins is β, and the rate that it changes to $n-1$ proteins in αn. The probability that the system has n proteins is defined as P_n. Given that the system is in state n, the rate of transitions to state n' is defined as $k_{n',n}$. Note that the direction is $n' \leftarrow n$ for the indices. At equilibrium, the quantities P_n do not change. A stricter requirement is that for any two states n and n', the probability of a transition $n' \leftarrow n$ exactly equals the probability of the reverse transition $n \leftarrow n'$. With our notation,

$$k_{n'n}P_n = k_{nn'}P_{n'}.$$

This condition is called detailed balance, and we will apply it to determine the equilibrium distribution P_n.

First we look at transitions between state 0 and state 1. The rate of transition from 0 to 1 is β and the rate from $1 to 0$ is alpha, yielding

$$\beta P_0 = \alpha P_1$$

$$P_1 = (\beta/\alpha)P_0.$$

Next we consider states 1 and 2. The upward rate is always β, but the downward rate is 2α because we have two proteins that can decay. The detailed balance condition is

$$\beta P_1 = 2\alpha P_2.$$

We then solve for P_2 to find

$$P_2 = (\beta/2\alpha)P_1 = (1/2)(\beta/\alpha)^2 P_0.$$

Next for states 2 and 3 we have detailed balance condition

$$\beta_2 = 3\alpha P_3$$

and solve for P_3 as

$$P_3 = (\beta/3\alpha)P_2 = (1/3!)(\beta/\alpha)^3 P_0.$$

In general we find that

$$P_n = (\beta/n\alpha)P_{n-1} = (1/n!)(\beta/\alpha)^n P_0.$$

We therefore know each probability in terms of P_0. To find this final factor, we recognize that the probabilities are normalized,

$$\sum_{n=0}^{\infty} P_n = 1.$$

This sum is

$$\sum_{n=0}^{\infty} P_n = \sum_{n=0}^{\infty} P_0(\beta/\alpha)^n/n! = P_0 \exp(\beta/\alpha)$$

because we know that the Taylor series for e^{λ} is $\sum_{n=0}^{\infty} \lambda^n/n!$. The probability of state 0 is $P_0 = e^{-\beta/\alpha}$, and in general

$$P_n = [(\beta/\alpha)^n/n!]e^{-\beta/\alpha}.$$

If you look back to your notes from probability and statistics, you will see that this is a Poisson distribution with Poisson parameter β/α. To denote that the number of protein copies n is distributed as a Poisson distribution with Poisson parameter β/α, we write

$$n \sim \text{Pois}(\beta/\alpha).$$

This is shorthand for our previous statement that $P_n = [(\beta/\alpha)^n/n!]e^{-\beta/\alpha}$.

We are about to see what this result says about the equilibrium fluctuations in n. We already can make the important observation, though, that the fluctuations can only depend on the ratio β/α, which in turn equals the mean copy number in the ODE model. We conclude that fluctuations should only depend on mean copy number, and that proteins with the same mean copy number should have the same magnitude fluctuations. This should be a general property, similar to the general property that the parameter α governs timescales.

Returning to the Poisson distribution, we now calculate the mean $\langle n \rangle$ and the variance $\mathrm{Var}(n)$,

$$\mathrm{Var}(n) \equiv \langle n^2 \rangle - \langle n \rangle^2.$$

We use our moment-generating results by defining a characteristic function $\tilde{P}(s)$ as

$$\tilde{P}(s) = \sum_{n=0}^{\infty} e^{-sn} P_n.$$

This is the form of the Laplace transform for a function P_n defined on a discrete domain of integers. We could convert P_n to a function $P(t)$ on a continuous domain as

$$P(t) \equiv \sum_n P_n \delta(t-n)$$

to obtain an identical expression for $\tilde{P}(s)$. In discrete signal processing, this is termed a Z-transform.

Since P_n is normalized, $\tilde{P}(s)|_{s=0} = 1$, or $\ln \tilde{P}(0) = 0$. The mean and variance are obtained as higher derivatives. We express these moment-generating rules (really cumulant-generating rules) as

$$\ln \tilde{P}(s) = 0$$
$$(-d/ds) \ln \tilde{P}(s) = \langle n \rangle$$
$$(-d/ds)^2 \ln \tilde{P}(s) = \mathrm{Var}(n).$$

The transform for a Poisson probability distribution with parameter λ is

$$\tilde{P}(s) = \sum_{n=0}^{\infty} e^{-sn} (\lambda^n/n!) e^{-\lambda}$$

$$\tilde{P}(s) = e^{-\lambda} \sum_{n=0}^{\infty} (\lambda e^{-s})^n/n!$$

$$\tilde{P}(s) = e^{-\lambda} \exp(\lambda e^{-s}),$$

where we again recognize the Taylor series for an exponential. The exponential of an exponential looks complicated, but remember that we really work with the cumulant-generating function $\ln \tilde{P}(s)$ rather than the moment-generating function $\tilde{P}(s)$,

$$\ln \tilde{P}(s) = -\lambda + \lambda e^{-s}.$$

This already looks simpler, which is usually indicates that we are going in the right direction.

For the overall normalization,

$$\ln \tilde{P}(s) = 0,$$

which is a good check that we haven't made an error. Always check this when you are transforming a probability distribution.

Next for the mean and variance,

$$(-d/ds) \ln \tilde{P}(s) = \lambda e^{-s},$$
$$(-d/ds)^2 \ln \tilde{P}(s) = \lambda e^{-s}.$$

Evaluating these results at $s = 0$ gives

$$\langle n \rangle = \lambda.$$
$$\mathrm{Var}(n) = \lambda.$$

For our probability distribution, the mean and variance of protein copy number should be identical, both equal to β/α.

Over the past 10 years or so, advances in genomics and biotechnologies have permitted measurements of protein and mRNA copy number in individual cells. For the most part, variation in copy number follows the predictions of this simple model. There are some deviations, however. For mRNA, transcription can come in bursts, with several transcripts made at once. This leads to noise somewhat higher than the prediction. For proteins, fluctuations in copies of high-copy-number proteins have an additional source of noise ascribed to the overall number of ribosomes in the cell.

We can use these predictions for noise to return to the barrier height requirements for Delta-Notch patterning. Suppose that the stable states have one protein at copy number $\bar{n}$ and the second protein at copy number close to 0. We've suggested that the transition rate k between stable states is approximately

$$k \sim \alpha e^{-\bar{n}/\sigma_n}.$$

Now we know that $\mathrm{Var}(n) = \bar{n}$, and $\sigma_n = \sqrt{\bar{n}}$. The transition rate is approximately

$$k \sim \alpha e^{-\sqrt{\bar{n}}}.$$

Now suppose that we want cells to go about 100 generations, or $100/\alpha$ time units, without a reverse transition. This means that $e^{-\sqrt{\bar{n}}} < 1/100$, or

$$\sqrt{\bar{n}} > 2 \ln 10 \approx 5.$$

To avoid reverse transitions, we require mean protein copy number $\bar{n} > 25$.

This prediction in general agrees with copy numbers of transcription factors and other proteins that determine cell state. We also can see that processes that increase noise in gene and protein expression can increase fluctuations in copy number, which can exponentially increase the rates of transitions to unwanted cellular states. This is thought to occur in cancer, where deregulation of chromatin structure leads to aberrant gene activation and transitions to proliferative or metastatic cell states.

Chapter 15

Time correlation functions and the fluctuation-dissipation theorem

Chapter 16

Stochastic Dynamics

Chapter 17

Stochastic Simulations and the Gillespie Algorithm

Part IV

Cells as Spatial Systems

Chapter 18

Morphogen Gradient Patterning

Chapter 19

Diffusion

Chapter 20

Solving the Diffusion Equation

Chapter 21

Patterning and Noise

Part V

Cellular Networks

Chapter 22

Diffusion on a Network

Chapter 23

Network Topology, Motifs, and Clustering

Chapter 24

The Giant Component

Chapter 25

Network Partitioning and Spectral Clustering

Chapter 26

Metabolic Networks and Flux Balance Analysis

Appendix A

Problems