Introduction

What is bistability?

- We have seen several examples of ultrasensitivity so far: Hill curves, the MWC model, Goldbeter and Koshland's zero-order ultrasensitivity.
- All of these examples produce curves that change more sharply than Michaelis-Menten kinetics. (We used a specific measure called EC_{90}/EC_{10}: the concentration required for 90% activity divided by the concentration required for 10% activity.)
- The rapid change between two distinct regimes in these examples mimics the binary nature of states for switches/latches.
- However, the notion of memory is missing. We expect a switch/latch to stay in the state that we last put it in, even when we take our hand away.
- As an example, consider the Hill curve

$$\text{Steady-state fraction bound } = f = \frac{x^n}{K + x^n}, \qquad n > 1$$

  The steady-state fraction bound depends only on x; it does not depend at all on the most recent state of the system.

- By contrast, some systems show hysteresis: a dependence on the history of the input. Hysteresis can manifest as a tendency for the system to remain in its current state (drawing at board).
- Systems that have two stable states, requiring large perturbations to switch from one to the other, are called *bistable* systems.
- They are often used in biology to establish cell fate and commit to long-term decisions like sporulation, apoptosis, cell cycle progression, etc.
- How can we identify how many stable states a system has? Can we find a way to do this will scale easily with the number of parameters?

Systems as n-dimensional spaces

- The current state of an n-dimensional system can be thought of as a point in n-dimensional space (*phase space*). (Each variable represents an axis in this space.)
- In practice only a subset of this space may be accessible, since concentrations cannot be negative and moiety conservation places constraints on possible combinations of variable values.
- Consider a two-dimensional system because it is easiest to visualize on the board. If we plot all of the points a system occupies as time elapses, i.e. all points $(x(t), y(t))$ for $t > 0$, then we get a contour that represents the system's *trajectory*. (Board drawing.) If we don't have explicit equations for $x(t)$ and $y(t)$, we could determine the trajectory through integration of the time derivatives $\dot{x}$ and $\dot{y}$.
- There may be some points where all of the time derivatives are zero: these are called *fixed points* (or *nodes*, or steady states). If a system reaches a fixed point (e.g., if it is there initially), then it will never leave unless the system is perturbed.

- In these notes I will denote a fixed point as $\mathbf{x}^* = (x^*, y^*)$. The condition for a fixed point in a two-dimensional system is that:

$$\left.\frac{dx}{dt}\right|_{(x^*,y^*)} = \left.\frac{dy}{dt}\right|_{(x^*,y^*)} = 0$$

- It may be necessary to identify fixed points numerically, though in some cases the algebra is not too complex to solve for them.
- Notice that it is possible for a system to have more than one fixed point. (Up until today, most of our examples have had only one fixed point/steady state.)

Definition of stability

- Some fixed points have the special property that if a system gets within some radius r of the fixed point, it will be "drawn in," approaching the fixed point as $t \to \infty$. These fixed points are said to be *(asymptotically) stable*.
- The set of points whose trajectories lead to a certain stable fixed point is called its *basin of attraction*.
- Stable fixed points interest systems biologists because a system will tend to remain near a fixed point once it has approached, and any disturbance (e.g. noise) that shifts the system less than r will eventually dampen out: such constancy is the basis for homeostasis and memory.
- Example: $\dot{x} = x^2 - c$, $c > 0$. This system has two fixed points, $x^* = \pm\sqrt{c}$. Trajectories from all points in $(-\infty, \sqrt{c})$ lead to $x^* = -\sqrt{c}$ as $t \to \infty$: the fixed point is therefore stable ($r = 2\sqrt{c}$) and the interval $(-\infty, \sqrt{c})$ is its basin of attraction.
- A bistable system has two stable fixed points.
- Finding fixed points and determining their stability will be the focus of this lecture.

Other long-term behaviors

- Approaching a stable fixed point is not the only possible long-term behavior of a trajectory.
- The system could *diverge*: one or more variables could go off to infinity at long times. This is not a realistic property of a biological system (something always becomes limiting), though it sometimes appears in simplified models.
- Systems with three or more variables could show chaotic behaviors, as we saw in a clip of the Lorenz system. In practice chaos is rare in models of biological systems.
- The system could enter an infinite loop when a trajectory is closed: this is called a *limit cycle*. Limit cycles can also be stable/unstable/half-stable. Limit cycles are observed in oscillating systems, and we will discuss in more detail how to detect them in the section of this course on biological clocks.

Bifurcations

- One challenge in identifying and characterizing systems is that the number of fixed points/limit cycles and their stability can change as we vary parameters. (cf. the common misconception that parameters only change a model quantitatively.)
- Example: $\dot{x} = x^2 - c$, with $c \in \mathbb{R}$. We saw above that this system has two fixed points when c is positive, one of which is stable. When c is negative, there are no fixed points at all. At the boundary, when $c = 0$, there is one fixed point which is *half-stable* (trajectories approach it, but only from negative values of x).
- Such qualitative changes to the number and types of fixed points/limit cycles are called bifurcations.

Finding fixed points and determining their stability through plotting

Plotting

- In some cases, the easiest way to identify whether a fixed point is stable is to plot sample trajectories and/or the system's *direction field*.
- A direction plot shows which way the system would be going if it were currently at certain points (usually chosen along a grid). The direction of the arrows is determined from $\dot{x}$ and $\dot{y}$.
- In principle we could sketch in many trajectories just by smoothly connecting one arrow to the next. (We have already covered how to use numerical integration to get $x(t)$ and $y(t)$ from $\dot{x}$ and $\dot{y}$ in a previous lecture and on problem set one, so this is also a convenient method.)
- This approach works well for systems with one or two variables, but plots in higher dimensions quickly become unwieldy to the human eye.

2-D example: mutual repression

- Suppose an organism has two transcriptional repressors, X and Y, that "antagonize" one another: X binds to the promoter of the Y gene to prevent its expression, and vice versa.
- Like most proteins, X and Y are naturally degraded/diluted at a rate proportional to their current concentration.
- Intuitively, we imagine that this system will wind up in one of two states: either [X] will be high and [Y] low, or vice versa. But are these states stable (giving us a bistable system)? How easy would it be to jump from one state to another? Are there other fixed points?
- To answer these questions, we must produce a model. Last week we learned that many transcription factors bind cooperatively to their recognition sites: we will assume this is true for our two repressors so that:

$$\text{Probability X is bound to its site } = \frac{[X]^n}{K + [X]^n}$$

where K is an (apparent) dissociation constant and n is a Hill coefficient.

- When X is bound to its site, Y is not expressed.Therefore the rate at which Y is produced ought to be:

$$\text{Rate of Y expression } \propto 1 - \frac{[X]^n}{K+[X]^n} = \frac{K}{K+[X]^n}$$

- Assuming the same is true for Y, and letting x=[X] and y=[Y], we arrive at the model:

$$\begin{aligned}\frac{dx}{dt} &= \frac{\alpha K}{K_y + y^n} - \beta x \\ \frac{dy}{dt} &= \frac{\alpha K}{K_x + x^n} - \beta y\end{aligned}$$

- One disadvantage of the numerical integration/direction field method is that we need to plug in values for parameters in order to produce plots. This means that if the behavior depends qualitatively on a parameter (e.g. n), we stand to miss that trend unless we draw many plots with different parameter values.

- Suppose we choose $K = 0.5$ and $n = 3$, and $\alpha = \beta = 1$ for simplicity. Our simplified model is:

$$\begin{aligned}\frac{dx}{dt} &= \frac{0.5}{0.5 + y^3} - x \\ \frac{dy}{dt} &= \frac{0.5}{0.5 + x^3} - y\end{aligned}$$

- Finding the fixed points algebraically is a rather unpleasant experience. Noting that since $\dot{y} = 0$ at a fixed point, $y^* = 0.5/(0.5 + x^{*3})$, and plugging this expression into $\dot{x} = 0$, we get that:

$$\frac{1}{0.5 + \left(\frac{0.5}{0.5+x^{*3}}\right)^3} = x^*$$

- Rearranging gives a high-order polynomial in x^*. We would have an unpleasant time even finding the fixed points analytically, but MATLAB is happy to oblige:

```
vpasolve(2*x == 1/(0.5 + (0.5/(0.5+x^3))^3),x)
```

- The three positive solutions are: $x^* = 0.46, 0.64, 0.84$.

- Plotting in MATLAB quickly confirms the presence of two stable fixed points. (We can "see" that these two points are stable by looking at the direction field around them: all arrows point inwards.)

```
function [] = mutualrepression()
    % Pick some parameter values for plotting
    global k n
    k = 0.5; n=3;
```

```

    [x, y] = meshgrid(0:0.05:1.5, 0:0.05:1.5);
    dx = k ./(k+y.^n) - x;
    dy = k ./(k+x.^n) - y;
    r = (dx.^2 + dy.^2).^0.5;
    dx = dx ./ r;
    dy = dy ./ r;

    quiver(x,y,dx,dy); hold on;
    xlabel('[X]')
    ylabel('[Y]')
    axis([0,1.5,0,1.5])
    [t, c] = ode45(@updater, [0 50], [0.7, 0.8]);
    plot(c(:,1),c(:,2),'-r', 'LineWidth', 3)
    plot(0.7, 0.8, 'or');
    [t, c] = ode45(@updater, [0 50], [0.6,0.5]);
    plot(c(:,1),c(:,2),'-g', 'LineWidth', 3);
    plot(0.6, 0.5, 'og');
    [t, c] = ode45(@updater, [0 50], [1.4,0.2]);
    plot(c(:,1),c(:,2),'-b', 'LineWidth', 3);
    plot(1.4, 0.2, 'ob');
    [t, c] = ode45(@updater, [0 50], [0.3,1.2]);
    plot(c(:,1),c(:,2),'-k', 'LineWidth', 3);
    plot(0.3, 1.2, 'ok');
    [t, c] = ode45(@updater, [0 50], [0.3,0.3]);
    plot(c(:,1),c(:,2),'-m', 'LineWidth', 3);
    plot(0.3, 0.3, 'om');

end

function dc = updater(t, c)
    x = c(1);
    y = c(2);
    global k n
    dx = k/(k+y^n) - x;
    dy = k/(k+x^n) - y;
    dc = [dx; dy];
end
```

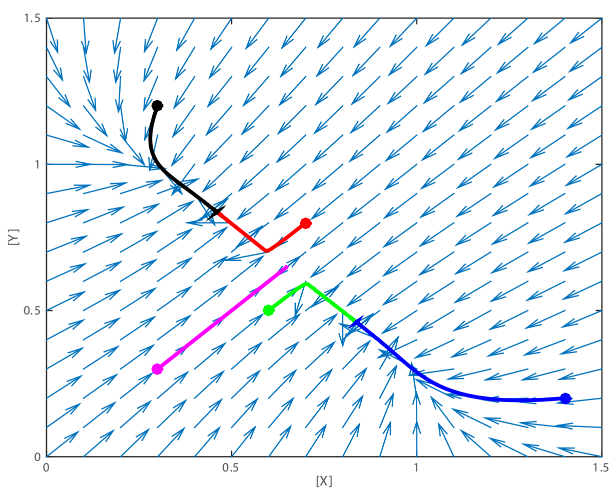

Figure 1: Direction field and five sample trajectories for the mutual repression model.

- The third fixed point attracts trajectories that lie along a single line. This type of fixed point is called a *saddle*. (Drawing at board; explain analogy.) It is hard to get a good direction field near that fixed point.

- A sketch of the 2-D system's *phase plane* (i.e., the value of one variable on each axis) showing major features like fixed points and limit cycles, with a few sample trajectories, is sometimes called a *phase portrait*.

Stability analysis for linear systems

- As mentioned, the plotting method will not scale when we have three or more dimensions in our system. We will introduce a more general method based on analysis of the stability of fixed points of linear systems.
- In two dimensions, a linear system has the form:

$$\begin{aligned} \frac{dx}{dt} &= ax + by \\ \frac{dy}{dt} &= cx + dy \end{aligned}$$

- Notice that system has a fixed point at $x = y = 0$ (it may have others, depending on the values of a through d).
- We can rewrite this system in matrix form as:

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{A}\begin{pmatrix} x \\ y \end{pmatrix}$$

- Rates of change in biological systems often contain non-linear terms like Hill equations, hyperbolic terms from Michaelis-Menten kinetics, etc. However, all of these systems can be *linearized* near a fixed point in order to study the fixed point's stability.
- Will now show how this is done; afterwards, we'll return to learn how to determine the fixed points of linear systems.

Linear approximations

- Suppose we are given the value of a function f and its time derivative at some point x^*, and we would like to estimate the value of f at a nearby point x.
- As long as x and x^* are sufficiently close, we can estimate that:

$$f(x) \approx f(x^*) + (x - x^*) \left.\frac{df}{dx}\right|_{x^*}$$

- The last symbol denotes the time derivative of f evaluated at a specific point x^*: it will just be some constant number.
- Notice that this by rearrangement we can show that this is just the equation for a line (slope in red, y-intercept in blue):

$$f(x) = \color{red}{\left.\frac{df}{dt}\right|_{x^*}} x + \color{blue}{f(x^*) - x^* \left.\frac{df}{dt}\right|_{x^*}}$$

  That is why call this approximation linearization.

- For a two-dimensional system, the approximation would be slightly more complicated but of a similar spirit:

$$f(x,y) \approx f(x^*,y^*) + (x-x^*)\left.\frac{df}{dx}\right|_{(x^*,y*)} + (y-y^*)\left.\frac{df}{dy}\right|_{(x^*,y*)}$$

Linearization of non-linear systems near fixed points

- Suppose we define a two-variable system in terms of non-linear functions f and g:

$$\begin{aligned}\frac{dx}{dt} &= f(x,y)\\ \frac{dy}{dt} &= g(x,y)\end{aligned}$$

- If (x^*, y^*) is a fixed point of the system, i.e.

$$f(x^*,y^*) = g(x^*,y^*) = 0,$$

  then applying the linearization approximation above, at some point (x, y) near this fixed point:

$$\begin{aligned}f(x,y) &\approx (x-x^*)\left.\frac{\partial f}{\partial x}\right|_{(x^*,y^*)} + (y-y^*)\left.\frac{\partial f}{\partial y}\right|_{(x^*,y^*)}\\ g(x,y) &\approx (x-x^*)\left.\frac{\partial g}{\partial x}\right|_{(x^*,y^*)} + (y-y^*)\left.\frac{\partial g}{\partial y}\right|_{(x^*,y^*)}\end{aligned}$$

- This is still not a linear system because some terms (e.g. $-x^* df/dx|_{(x^*,y^*)}$) are constants, which never appear in linear systems.
- To get rid of the constant terms, we introduce two new variables $a = x - x^*$ and $b = y - y^*$ and note that:

$$\begin{aligned}\frac{dx}{dt} = \frac{da}{dt} &\approx a\left.\frac{\partial f}{\partial x}\right|_{(x^*,y^*)} + b\left.\frac{\partial f}{\partial y}\right|_{(x^*,y^*)}\\ \frac{dy}{dt} = \frac{db}{dt} &\approx a\left.\frac{\partial g}{\partial x}\right|_{(x^*,y^*)} + b\left.\frac{\partial g}{\partial y}\right|_{(x^*,y^*)}\end{aligned}$$

- Equivalently, in matrix form,

$$\frac{d}{dt}\begin{pmatrix}a\\b\end{pmatrix} = \begin{pmatrix}\frac{\partial f}{\partial x} & \frac{\partial f}{\partial y}\\ \frac{\partial g}{\partial x} & \frac{\partial g}{\partial y}\end{pmatrix}_{(x^*,y^*)}\begin{pmatrix}a\\b\end{pmatrix} = \mathbf{J}\begin{pmatrix}a\\b\end{pmatrix}$$

- The matrix $\mathbf{J}$ is the *Jacobian* of the non-linear system. Note that after we have evaluated the Jacobian at a point, all of its values are just scalars. We have converted to a linear system!
- Note that the fixed point $(x, y) = (x^*, y^*)$ is equivalent to $(a, b) = (0, 0)$. Therefore, to assess the stability of the fixed point (x^*, y^*) in the non-linear system, we examine the stability of the fixed point $(0, 0)$ in the linearized system.

Determining stability of fixed points in linear systems

- Recall our general form for a linear system:

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{A}\begin{pmatrix} x \\ y \end{pmatrix}$$

- The matrix $\mathbf{A}$ contains everything that makes this system's behavior unique from other linear systems, including whether or not $(0, 0)$ is a stable fixed point.
- One way of characterizing $\mathbf{A}$ is by determining its *eigenvalues* and *eigenvectors*.
- Eigenvectors are column vectors which I'll denote by $\mathbf{v} = (v_x, v_y)^T$. Eigenvectors have the special property that when multiplied by $\mathbf{A}$, the result is a complex multiple of the original eigenvector:

$$\begin{aligned} \begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} v_x \\ v_y \end{pmatrix} &= \lambda \begin{pmatrix} v_x \\ v_y \end{pmatrix} \\ \mathbf{Av} &= \lambda \mathbf{v} \end{aligned}$$

- Notice that any multiple of $\mathbf{v}$ is also an eigenvector.
- The multiplier λ is the corresponding eigenvalue.
- We can find distinct eigenvectors – that is, ones which are not just multiples of each other – and their corresponding eigenvalues by hand, but in practice it is much handier to make MATLAB do it. The command that we need is called `eig()`:

```
a = [1, 2; 3, 4];
[v, l] = eig(A);
 disp(sprintf('The first eigenvector is (%0.4f,%0.4f) with eigenvalue %0.4f', v(1,1), ...
     v(2,1), l(1,1)));
disp(sprintf('The second eigenvector is (%0.4f,%0.4f) with eigenvalue %0.4f', v(1,2), ...
     v(2,2), l(2,2)));
```

- Notice that the eigenvectors are columns of the output variable `v`, and the eigenvalues are along the diagonal of the output variable `l`.
- Executing this code tells us that for the matrix

$$\mathbf{A} = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}: \qquad \mathbf{v}_1 = \begin{pmatrix} -0.8246 \\ 0.5658 \end{pmatrix} \text{ and } \lambda_1 = -0.3723, \qquad \mathbf{v}_2 = \begin{pmatrix} -0.4160 \\ -0.9094 \end{pmatrix} \text{ and } \lambda_2 = 5.3723$$

- Why should we concern ourselves with finding eigenvectors and eigenvalues? Suppose that the system's initial state $(x_0, y_0)^T = c\mathbf{v}$, that is, the initial state is a (real) multiple of an eigenvector.
- At time zero, the rate of change for the system is:

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{A}\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \lambda\begin{pmatrix} x_0 \\ y_0 \end{pmatrix}$$

- So, after a short time $t = \Delta t$, the system's state will be approximately:

$$\begin{pmatrix} x \\ y \end{pmatrix} \approx \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \lambda\Delta t\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = (1 + \lambda\Delta t)\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = c\,(1 + \lambda\Delta t)\,\mathbf{v}$$

  In other words, the system is still a complex multiple of the eigenvector.
- If we imagine repeating this for many infinitessimal timesteps, we can see that at time $t = n\Delta t$, in the limit where n is large and Δt is small:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \lim_{\Delta t \to 0,} c\mathbf{v}\,(1 + \lambda\Delta t)^n = c\mathbf{v}e^{\lambda t}$$

- Most points do not lie along eigenvectors: how will systems that start at those points evolve with time?
- Suppose our matrix $\mathbf{A}$ has two distinct eigenvalues λ_1 and λ_2 and two distinct eigenvectors $\mathbf{v}_1$ and $\mathbf{v_2}$, i.e. the eigenvectors are not real multiples of one another. This means that we could describe any initial condition for our system as some combination of real multiples of those eigenvectors:

$$\begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = c_1\mathbf{v_1} + c_2\mathbf{v_2}$$

- We know how each of these two terms will evolve with time. This allows us to write the general solution for linear systems with two distinct eigenvalues:

$$\begin{pmatrix} x \\ y \end{pmatrix} = c_1\mathbf{v_1}e^{\lambda_1 t} + c_2\mathbf{v_2}e^{\lambda_2 t}$$

  where the constants c_i are chosen to match the initial conditions. For an n-dimensional system, the general solution is essentially the same but with n terms.
- Now that we have a general solution for trajectories originating at any point, we can determine whether the fixed point is stable. Recall that the fixed point is stable if all sufficiently nearby trajectories approach the fixed point as $t \to \infty$.
- The eigenvalues are complex numbers, so we can get a better picture if we break them up into real and imaginary parts and simplify:

$$
\begin{aligned}
\begin{pmatrix} x \\ y \end{pmatrix} &= c_1\mathbf{v_1}e^{(\alpha+i\beta)t} + c_2\mathbf{v_2}e^{(\gamma+i\delta)t} \\
&= c_1\mathbf{v_1}e^{\alpha t}e^{i\beta t} + c_2\mathbf{v_2}e^{\gamma t}e^{i\delta t} \\
&= c_1\mathbf{v_1}e^{\alpha t}\left[\cos\beta t + i\sin\beta t\right] + c_2\mathbf{v_2}e^{\gamma t}\left[\cos\delta t + i\sin\delta t\right]
\end{aligned}
$$

- Now it is clear that the imaginary parts of the eigenvalues determine whether the system rotates with time, and in what direction. If the real parts of the eigenvalues are zero but the imaginary parts are not: then, the system will rotate around the fixed point over time but never get any closer or further away. We call this type of fixed point a *center*.
- The real parts, which appear in the exponential terms, determine whether the variables' values will grow or shrink with time. If the real parts are all negative, then the system will decay to zero: this indicates that the fixed point is stable.
- If both of the real parts are positive, then the system will diverge towards the largest eigenvector. When the system behaves this way, we say the fixed point is unstable
- What if one of the real parts of the eigenvalues is positive, and the other is negative? In this case, the fixed point is called a *saddle*. Trajectories will approach a saddle node along one direction only to veer off at the last minute in a different direction. We saw this type of behavior already in our mutual repression system.

Summary

Although we have written out examples for a two-dimensional system, the same basic principles apply to an n-dimensional linear system:

- An n-variable system is stable if the real parts of all eigenvalues are negative.
- If the real part of even one eigenvalue is positive, then the system is unstable.

At the beginning of our next lecture, we will practice determining the stability of fixed points and linearizing non-linear systems. Then we will return to the example of mutual repression introduced here to begin our explicit discussion of bistability.