Michaelis-Menten Kinetics

A reaction is said to undergo first order kinetics if the reaction rate depends linearly on the concentration of only one reactant. Radiactive decay is a classic example of a first order reaction. Another example is the spontaneous change of a protein from one conformation to another. Such reactions can be modeled using the equation

$$\frac{dP}{dt} = k_0 S \tag{1}$$

where S is the concentration of reactant (or substrate) and P is concentration of the product. Most first order reactions of biological interest have rate constants k_0 that depend on the physical properties of the intramolecular bonds and are therefore difficult to modify quickly in response to signaling events. The way that cells avoid this problem is to use reactions that are slow to occur spontaneously ($k_0 \approx 0$) but that can be facilitated by catalyzing agents called enzymes. A typical enzyme-mediated reaction proceeds as illustrated in Fig. 1.



Figure 1: Caption

We would like to write down a simple reaction equation analogous to Eq. (1) that describes this enzyme-mediated reaction. Such an equation ought to have certain features. For example, as the concentration of the enzyme goes to zero, the production rate of P should go to zero. Furthermore, in the limit of high concentrations of substrate, the production rate is limited by enzyme availability. Although we could simply invent a functional form that has these features and use it when building more complicated models that involve many such component reactions, it would be preferable to derive such a functional form.

Consider that a single substrate molecule (concentration denoted by S) can bind to a single enzyme (E) to form a single complex (C) at rate k_1 . Complex can revert to E and S at rate k_{-1} or can facilitate change in S producing product P at rate k_2 . Once formed, product and enzyme decouple and enzyme is free to pick up more substrate S. Binding P is considered to be essentially impossible so the last step in the reaction diagram in Fig. 1 is considered to be irreversible. The equations for all concentrations in question are

$$\frac{dS}{dt} = -k_1 SE + k_{-1}C$$

$$\frac{dE}{dt} = -k_1 SE + k_{-1}C + k_2C$$

$$\frac{dC}{dt} = k_1 SE - k_{-1}C - k_2C$$

$$\frac{dP}{dt} = k_2C.$$
(2)

Note that because the reaction has one substrate molecule reacting with one enzyme to form one complex, the units on S, E and C can be treated as the same. With a population model, we

would usually think of the units as including the population type. For example, in the predator prey model, the variable representing predator has units of number or density of predators and the coefficients on the predation term convert from prey to predator or vice versa. In keeping with this, it might be more appropriate to put a factor in front of both terms in the S and P equations with units of concentration of substrate/concentration of enzyme but this is usually omitted (as is the case with SIR models, as well).

Because we assumed the $C \to E + P$ reaction is irreversible, the P equation is entirely "downstream" of the other equations meaning that once we find solutions S(t), E(t) and C(t), we can simply integrate the P equation. Or, if our goal is to find an equation analogous to Eq. (1) that is appropriate to the enzyme-mediated reaction, we simply need to figure out how to write C in terms of S.

Notice that $S + C + P = S_{tot}$ (an exercise left to the reader) so we don't even have to integrate to find P if S(t) and C(t) are known. As a further simplification, note that $E + C = E_{tot}$. This reduces the system from 4 to 2 equations. We choose to study it using S and C:

$$\frac{dS}{dt} = -k_1 S(E_{tot} - C) + k_{-1}C$$
$$\frac{dC}{dt} = k_1 S(E_{tot} - C) - k_{-1}C - k_2C.$$

We could have chosen the S and E equations instead without much change in the analysis but not the E and C equations (why not?).

Our first step is to figure out the time scales in the problem. The parameters in the system are k_1 (concentration⁻¹ time⁻¹), k_{-1} (time⁻¹), k_2 (time⁻¹), S_{tot} (concentration), and E_{tot} (concentration). We can also consider initial condition values as parameters but we assume here that S and E are initially both at their "tot" values and C(0) = 0 so this provides no new parameters. Parameter combinations with units of time are $1/k_{-1}$, $1/k_2$, $1/(k_1S_{tot})$ and $1/(k_1E_{tot})$.

In general, it is not possible to reduce this system to a single equation for P that depends only on S. However, if we assume that enzyme is present in much lower concentrations than the substrate, we can use the techniques associated with multiple time scales to do so. In terms of biology, this assumption is often a reasonable one.

Under the assumption that $E_{tot} \ll S_{tot}$, we have (at least) two very different time scales, $1/(k_1S_{tot})$ and $1/(k_1E_{tot})$. Interestingly, both of these time scales relate to the same expression (k_1SE) , corresponding to substrate-enzyme binding. The reason these rates are different is due the difference in concentrations and not due to any physical feature of the molecular interactions (and hence rate constants) involved in the reactions. The subtlety here is that the rate of substrate depletion is exactly the same as the rate of enzyme depletion due to this reaction but, when translated into nondimensional rates, the enzyme gets turned into complex much quicker than the substrate. This may seem odd at first but it demonstrates the power of nondimensionalization which, in this case, turned the concentrations into fractions of the totals available. The *fraction* of the total substrate available does not change as quickly as the *fraction* of total enzyme available even though the absolute rates are identical simply because there is a lot more substrate than enzyme.

Nondimensionalizing the concentrations, the equations become

$$\frac{ds}{dt} = -k_1 E_{tot} s(1-c) + k_{-1} \frac{E_{tot}}{S_{tot}} c$$
$$\frac{dc}{dt} = k_1 S_{tot} s(1-c) - k_{-1} c - k_2 c$$

where $s = S/S_{tot}$ and $c = C/E_{tot}$. To nondimensionalize time, we choose as a scale either $1/(k_1S_{tot})$ to get the fast time scale or $1/(k_1E_{tot})$ to get the slow time scale. On the fast time scale, the equations are

$$\begin{aligned} \frac{ds}{d\tau_f} &= -\frac{E_{tot}}{S_{tot}}s(1-c) + \frac{k_{-1}E_{tot}}{k_1S_{tot}^2}c\\ \frac{dc}{d\tau_f} &= s(1-c) - \frac{k_{-1}}{k_1S_{tot}}c - \frac{k_2}{k_1S_{tot}}c \end{aligned}$$

where $\tau_f = k_1 S_{tot} t$. Notice the small nondimensional parameter $\epsilon = E_{tot}/S_{tot}$ is a factor on the RHS of the first equation. Thus, the nondimensional fast system is

$$\frac{ds}{d\tau_f} = \epsilon(-s(1-c) + \alpha c)$$
$$\frac{dc}{d\tau_f} = s(1-c) - \alpha c - \beta c$$

where $\alpha = k_{-1}/(k_1 S_{tot})$ and $\beta = k_2/(k_1 S_{tot})$. To understand the fast dynamics (approximately), we set $\epsilon = 0$ and discover that the substrate as a fraction of the total amount present (s) is constant on this scale (or, more accurately, it depletes extremely slowly). By setting $\epsilon = 0$, we are actually assuming that because ϵ is small, it is ok to assume that the entire RHS of the equation is small. Because s and c are fractions, they are necessarily less than 1 but α might be large. Here, we make a second assumption on the parameters, one that is often glossed over in most treatments of Michaelis-Menten kinetics and that is that $\epsilon \alpha c \ll 1$. Because c depends on time and we cannot be certain of its size, other than having an upper bound of 1, we play it safe and require that $\epsilon \alpha \ll 1$ or equivalently that $k_{-1}/(k_1 S_{tot}) \ll 1/\epsilon$. Translating this into biochemical terms, this means that the backward rate of substrate falling out of the enzyme must not be too large compared to the (upper bound on the) forward rate of substrate binding to the enzyme. How large is too large? That depends on the ratio $\epsilon = E_{tot}/S_{tot}$ – the more extreme this ratio, the less we have to worry about the size of the backward rate k_{-1} relative to the (upper bound on the) forward rate k_1S_{tot} .

To emphasize that s does not change on the fast time scale, we denote it as s_0 . When working with the slow system, we will explicitly include the dependence on the slow time variable (i.e. $s_0(\tau_s)$) but omit it for now. Thus, we can plug $s(\tau_f) = s_0$ into the c equation which is now a simple linear equation,

$$\frac{dc}{d\tau_f} = s_0 - (s_0 + \alpha + \beta)c,$$

so the solution is exponential. The quasi-steady state value of c is $c_{qss}(s_0) = s_0/(s_0 + \alpha + \beta)$ and, because there is a "-" in front of the c term, we know this steady state is stable. We can now use this information to solve the dynamics on the slow time scale.

The nondimensional slow system is

$$\frac{ds_0}{d\tau_s} = -s_0(1-c) + \alpha c$$

$$\epsilon \frac{dc}{d\tau_s} = s_0(1-c) - \alpha c - \beta c$$

where $\tau_s = \epsilon \tau_f$ and I have written s_0 to make it explicit that the constant s from the fast scale is actually a slow changing value that is no longer (approximately) constant on the slow scale. Provided $dc/d\tau_s$ is not large (which is only true after the initial fast dynamics approach QSS) we can approximate the RHS of the c equation to zero (which formally looks like we've taken $\epsilon = 0$). The system becomes

$$\frac{ds_0}{d\tau_s} = -s_0(1-c) + \alpha c = -s_0 + (s_0 + \alpha)c$$

$$0 = s_0(1-c) - \alpha c - \beta c.$$

Notice that the c equation is now the same steady state equation that we just solved on the fast scale. Using the expression for $c_{qss}(s_0)$ and recognizing that s_0 is actually changing on the slow time scale (so denoted $s_0(\tau_s)$), we reduce the slow system to a single equation

$$\frac{ds_0}{d\tau_s} = -s_0 + (s_0 + \alpha)c_{qss}(s_0) = -\beta \frac{s_0}{s_0 + \alpha + \beta}.$$
(3)

This equation can be solved (almost) by separating variables to get

$$\tau_s = \frac{s_0 + K \ln s_0}{\beta} + C.$$

Although this cannot be solved for s_0 , we can plot the solution as s_0 versus τ_s as shown in Fig. ?? (figure not yet available).

A few interesting facts can be extracted from this analysis. First, notice that c converges to the QSS with a time constant $1/(s_0 + \alpha + \beta)$. As the solution approaches s = c = 0 (the unique steady state), s_0 becomes negligible so the time constant is $1/(\alpha + \beta)$. This time constant is relative to the fast time scale in the sense that it represents exponential decay of the form $\exp(-(\alpha + \beta)t_f)$. In terms of dimensional time, this tells us that one of the eigenvalues for the steady state at (0,0) is $\lambda_1 = k_{-1} + k_2$. On the slow time scale, we can see from equation (3) that s_0 approaches steady state with time constant $(\alpha + \beta)/\beta$. This tells us that the other eigenvalue of the full system is approximately $\lambda_2 = k_1 k_2 E_{tot}/(k_{-1} + k_2)$. This can be verified by calculating the eigenvalues in the usual way – by linearizing about the steady state.

Recall that our goal was not so much to solve the system nor to find eigenvalues but instead to find an analogous formula to Eq. (1) for the production rate of P. The QSS expression for cprovides exactly what we need. Redimensionalizing c and plugging the resulting QSS formula into Eq. (2), we get

$$\frac{dP}{dt} = k_2 E_{tot} \frac{S}{S + K_m}$$

where $K_m = (k_{-1} + k_2)/k_1$.

Practice problem

1. The equation

$$\frac{dN}{dt} = rN\left(1 - \frac{N}{K}\right) - \alpha \frac{N}{A+N}$$

has been used to model the population dynamics in the fisheries where the first term accounts for inherent population dynamics and the second term describes the harvesting of fish. Write down a more general model, analogous to the initial S, C system in the enzyme kinetics context, and use a similar assumption ($E_{tot} \ll S_{tot}$) to reduce your model to the equation above using QSS analysis.

2. Find the steady state of the original full system and linearize about it to determine its eigenvalues and stability. Compare these to the approximation given in the text above. You might want to use the approximation $\sqrt{1+x} \approx 1 + x/2$ when x is small.