1 Introduction

Enzymes are the basic machinery that make chemical reactions occur in living cells. They are proteins - macromolecules which consist of chains of amino acids ranging in length from under one hundred to several thousand. These chains fold upon themselves and interact with other proteins to form a wide variety of structures. The structure and any subsequent (post-translational) modification of these amino acid chains ultimately govern the enzyme’s function.

Figure 1: Orotidine-monophosphate decarboxylase, an enzyme required for DNA synthesis, from left to right represented by a cartoon model, a surface model, and a stick model. Figures generated with PyMOL (pymol.org). PDB code 1EIX (from www.rcsb.org).

Enzymes are required in all living organisms because chemical reactions usually require activation energy, which is the energy necessary to form the transition state between reactants and products (see Figure 2). While it may be energetically favorable to go from reactant to product, this only means that the reaction will proceed - not that it will go quickly. It is actually the activation energy which determines the rate at which the reaction proceeds\(^1\). Enzymes stabilize transition states for reactions, and thus lower the activation energy required. This has the overall effect of speeding up a reaction. A common measure for how much a reaction is sped up is called the rate enhancement, equal to the ratio of the catalyzed rate to the uncatalyzed rate. This ratio varies widely, ranging from one (which is technically no longer an enzyme - merely a protein) to \(1.4 \times 10^{17}\) for orotidine-monophosphate decarboxylase (an enzyme involved in DNA synthesis) [11].

\(^1\)The relationship between activation energy and kinetic constants is governed by the Arrhenius equation, 
\[ k = A \exp \left[ \frac{-E_a}{RT} \right] \]
where \(A\) is an empirical reaction-dependent constant, \(E_a\) is the activation energy for the reaction, \(R\) is the gas constant, and \(T\) is the temperature in degrees Kelvin. While it is possible to speed up equations by heating them, this is typically not feasible in biological systems for a variety of reasons. The much simpler solution is to lower the activation energy.
This review will cover the mathematics governing basic enzyme interactions beginning with the Michaelis-Menten equation, and will then diverge into systems which require more complex formulae to describe them. In particular, I will describe the following:

- Michaelis-Menten Model
- Inhibitors
- Comparing Michaelis-Menten vs. Exact (Numerical) Solutions
- Other Models of Enzyme Activity

## 2 Michaelis-Menten Kinetics

### 2.1 Background and Basic Chemical Kinetics

Reaction rates are typically described as being proportional to some multiple of powers of concentrations of reactants [9]. For example, for the reaction\(^2\) \(aA + bB \xrightarrow{k} cC\), where \(a\), \(b\), and \(c\) are the stoichiometric amounts of each species in the reaction, one might describe the kinetics by:

\[
\frac{1}{c} \frac{d[C]}{dt} = -\frac{1}{a} \frac{d[A]}{dt} = -\frac{1}{b} \frac{d[B]}{dt} = k[A]^m[B]^n
\]  

(1)

In this equation, \(k\), \(m\) and \(n\) are experimentally determined parameters, and \([A]\), \([B]\), and \([C]\) are measures of the concentration of each species (usually molarity, moles/liter). It is worth noting that the orders of the rate equation in each parameter \((m\ and\ n)\) are not necessarily related to the stoichiometric amount of each reactant used, and are typically experimentally determined values which depend on the reaction mechanism [9].

\(^2\)For the non-chemist audience, molecular reactions can be thought of as state diagrams, in which the flux in and out of each state can be described by kinetic equations.
2.2 Michaelis-Menten Equation

In 1913, Leonor Michaelis and Maud Menten\(^3\) published a set of equations believed to govern enzyme kinetics based on the concept of an enzyme forming a non-covalent complex with its substrate before catalyzing the reaction, and then dissociating from the product [8]. This chemical scheme is shown below.

\[
E + S \xrightleftharpoons[k_{-1}]{k_1} ES \xrightleftharpoons[k_{-2}]{k_2} E + P
\]  

(2)

If this is in fact the case, then the equation describing the rate at which product forms is

\[
\frac{d[P]}{dt} = k_2[ES]^m - k_{-2}[E]^n[P]^p
\]  

(3)

Because elementary reactions (those involving a single reaction step and a single transition state) have a rate order of one in each reactant and enzyme-substrate complex formation can be roughly thought of as an elementary reaction\(^4\), we can simplify (3) to

\[
\frac{d[P]}{dt} = k_2[ES] - k_{-2}[E][P]
\]  

(4)

Oftentimes, (4) is simplified further by assuming what is called 'steady-state kinetics,' in which \([ES]\) is presumed to be constant \(\left(\frac{d[ES]}{dt} = 0\right)\). As long as this assumption holds, (4) is equal to

\[
\frac{d[P]}{dt} = a - b[P]
\]  

(5)

where \(a = k_2[ES]\) and \(b = k_{-2}([E_{total}] - [ES])\). With a little bit of rearranging, this equation can be integrated by separating variables to give

\[
-\ln\left(\frac{a - b[P_1]}{b}\right) + \ln\left(\frac{a - b[P_0]}{a - b[P_1]}\right) = (t_1 - t_0)
\]

\[
\ln\left(\frac{a - b[P_0]}{a - b[P_1]}\right) = b(t_1 - t_0)
\]

\[
\frac{a - b[P_0]}{a - b[P_1]} = \exp[b(t_1 - t_0)]
\]

\[
[P_1] = \frac{a}{b} - \frac{a - b[P_0]}{b\exp[b(t_1 - t_0)]}
\]  

(6)

As the reaction progresses, this formula would predict that concentration of product \(([P_1])\) approaches \(\frac{a}{b}\), or \(\frac{k_2[ES]}{k_{-2}([E_{total}] - [ES])}\).

One more common simplification of the above model is to assume that enzymes have very low affinity for their products, and thus that \(k_{-2}\) is negligible. Equation (4) then simplifies to

\[
\frac{d[P]}{dt} = k_2[ES]
\]  

(7)

---

\(^3\)While Michaelis and Menten typically receive all the credit for this work, Victor Henri had observed a decade earlier that catalysis occurred at a rate that varied non-linearly both with substrate concentration and time. Although his name is often omitted, many have made the argument for referring to the following as Henri-Michaelis-Menten kinetics [13]. However, due to its common use, I will use the name Michaelis-Menten in this paper.

\(^4\)For more on elementary reactions and how they can be predicted by collision theory, see [7].
Integrating this equation yields a linear equation in $[P]$, in which product formation is constant as long as the steady-state approximation holds. However, it is not particularly useful since we do not know $[E \cdot S]$, which is a function of $[E_{\text{total}}]$ and $[S_{\text{total}}]$. Given $\frac{d[E \cdot S]}{dt} = 0$ and $k_{-2} = 0$, we can solve for $[E \cdot S]$ as follows:

$$ (k_{-1} + k_2) [E \cdot S] = k_1 [E] [S] $$

$$ K_m \overset{\text{def}}{=} \frac{k_{-1} + k_2}{k_1} = \frac{[E][S]}{[E \cdot S]} \quad (8) $$

Solving for $[E \cdot S]$, and remembering that $[E_{\text{total}}] = [E] + [E \cdot S]$, we obtain

$$ K_m = \frac{[E_{\text{total}}] - [E \cdot S]}{[E \cdot S]} $$

$$ K_m [E \cdot S] + [S][E \cdot S] = [S][E_{\text{total}}] $$

$$ [E \cdot S] = \frac{[E_{\text{total}}][S]}{K_m + [S]} $$

Substituting into (7) gives the following:

$$ \frac{d[P]}{dt} = \frac{k_2[E_{\text{total}}][S]}{K_m + [S]} \quad (9) $$

This is the classical form of the Michaelis-Menten\(^5\) equation. Oftentimes, $k_2[E_{\text{total}}]$ is written as $V_{\text{max}}$, as it represents the rate of product formation provided that all enzyme were bound to substrate.

There are a few interesting practical consequences of the Michaelis-Menten equation. The first is that at very large substrate concentrations, $\frac{d[P]}{dt} \approx k_2[E_{\text{total}}]$. Thus by setting up an experiment so that substrate is in great excess, it one can approximate $k_2$ as simply the slope of $P$ formation over time, divided by $[E_{\text{total}}]$.

Another interesting consequence occurs at the other end of the spectrum, in which $[S]$ is very low, and thus the $[S]$ in the denominator can be ignored. Then the initial slope of product formation is $\frac{k_2[E_{\text{total}}]}{K_m}$. However, this slope changes very quickly (because $[S]$ is very small and is being rapidly consumed). If we set $\frac{-d[S]}{dt} = \frac{k_2[E_{\text{total}}][S]}{K_m}$, this is easily integrated by separating variables to give

$$ [S] = [S_0] \exp \left[ -\frac{k_2[E_{\text{total}}]}{K_m} \right] \quad (10) $$

[13]. By taking the natural log of both sides, $\frac{k_2}{K_m}$ is easily determined by simple linear regression, and is often reported in papers on enzyme parameters. Alternatively, these days it is possible to estimate parameters directly with nonlinear regression via numerical methods.

Note also that in (9), $[S]$ in the concentration of free substrate. In many assays this is approximated as $[S_{\text{free}}] = [S_{\text{total}}]$ by using a substantial excess of inhibitor relative to enzyme ([S_{\text{total}}] $\gg$ [E_{\text{total}}]), such that only a very small portion of the substrate can be in the bound form at any given time. However, this can be problematic, because if $[S_{\text{total}}] \gg K_m$, it is not possible to estimate $K_m$, because the rate of product formation is essentially $V_{\text{max}}$. Thus, it is usually necessary to use substrate concentrations around $K_m$, and the enzyme concentration must be substantially lower than $K_m$. Practically speaking, this can be difficult as the purpose of the assay to begin with is to measure $k_2$ and $K_m$ - they are not known \textit{a priori}. However, it is easy to detect if $[S]$ is substantially above $K_m$, because the product formation curve will be essentially linear.

\(^5\)Again, occasionally and perhaps more properly referred to as the Henri-Michaelis-Menten equation [13].
Only when $[S]$ becomes close to $K_m$ will the rate of product formation begin to slow.

Before looking at the integrated form of this equation over time, it is worth reviewing the simplifying assumptions that went into deriving (9):

- $k_{-2} = 0$
- $\frac{d[E \cdot S]}{dt} = 0$

And equally importantly, what we did not assume:

- that $k_2$ is the rate limiting step
- that $E + S \rightleftharpoons E \cdot S$ was at equilibrium.

Technically speaking, we never made the assumption that $[S_{\text{free}}] = [S_{\text{total}}]$, although this assumption is commonly made when actually attempting measure the kinetic constants in a laboratory.

### 2.3 Integrated Michaelis-Menten Equation

Recognizing that $[S]$ is a function of time, we can integrate (9) by setting $d[P]/dt$ equal to $-d[S]/dt$ (note however that this only holds while $[E \cdot S]$ is constant). Our equation becomes

$$\frac{-d[S]}{dt} = \frac{k_2[E_{\text{total}}][S]}{K_m + [S]}$$

$$- \int_{[S_0]}^{[S_1]} \frac{K_m + [S]}{[S]} d[S] = \int_{t_0}^{t_1} k_2[E_{\text{total}}]dt$$

$$- (K_m \ln [S] + [S])_{[S_0]}^{[S_1]} = k_2[E_{\text{total}}]t_1$$

$$- \Delta[S] + K_m \ln \frac{[S_0]}{[S_1]} = k_2[E_{\text{total}}] \Delta t$$

This is the integrated form of the Michaelis-Menten equation[12], although it has been written in a variety of forms. Unfortunately, it is in the form of an implicit equation, and thus is not especially useful for prediction of changes in substrate levels. However, it can be rearranged as:

$$[S_0] + K_m (\ln [S_0] - \ln [S_1]) - k_2[E_{\text{total}}] \Delta t = [S_1]$$

While it is often difficult to measure the level of every species in a reaction over time, it is sometimes possible to measure the amount of one species (often by fluorescence or absorbance). Equation (12) gives a way to determine $k_2$ and $K_m$ by ordinary least squares (OLS) regression, given only change in substrate over time [12].

### 3 Inhibitors

Inhibitors of enzymes make up the bulk of the drugs on the market today. There are a variety of ways to inhibit enzymes including competitive reversible inhibition, competitive irreversible inhibition, and allosteric
inhibition. This section will focus on competitive reversible inhibitors, although for competitive irreversible inhibitors the system is essentially the same.

### Table 1: Drugs and the enzymes they inhibit

<table>
<thead>
<tr>
<th>Drug</th>
<th>Enzyme Inhibited</th>
<th>What it does</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fluoxetine (Prozac)</td>
<td>Pre-synaptic serotonin receptors</td>
<td>Antidepressant</td>
</tr>
<tr>
<td>Atorvastatin (Lipitor)</td>
<td>HMG-CoA reductase</td>
<td>Reduces cholesterol levels.</td>
</tr>
<tr>
<td>Naproxen (Aleve)</td>
<td>Cyclooxygenase (COX) 1 and 2</td>
<td>Used to relieve pain, fever, and inflammation</td>
</tr>
<tr>
<td>Methotrexate (Trexall)</td>
<td>Dihydrofolate Reductase</td>
<td>Used to treat cancer, autoimmune and inflammatory diseases, including psoriasis and rhematoid arthritis</td>
</tr>
<tr>
<td>Loratadine (Claritin)</td>
<td>H-1 Histamine Receptor</td>
<td>Antihistamine/anti-allergen</td>
</tr>
<tr>
<td>Sildenafil (Viagra)</td>
<td>Type 5 phosphodiesterases</td>
<td>Used to treat erectile dysfunction, also used for pulmonary artery hypertension</td>
</tr>
<tr>
<td>Amoxicillin</td>
<td>Bacterial transpeptidases</td>
<td>Antibiotic</td>
</tr>
<tr>
<td>Oseltamivir (Tamiflu)</td>
<td>Viral neuraminidase</td>
<td>Used to treat influenza, including avian flu</td>
</tr>
<tr>
<td>Lidocaine</td>
<td>Voltage-gated Na⁺ channels</td>
<td>Pain relief / anaesthetic</td>
</tr>
<tr>
<td>Clozapine</td>
<td>Serotonin and dopamine receptors</td>
<td>Antipsychotic / anti-schizophrenic</td>
</tr>
</tbody>
</table>

Table data compiled from [4]. HMG-CoA is 3-Hydroxy-3-methylglutaryl-(coenzyme-A).

Competitive inhibitors work by binding to the active site\(^7\) of an enzyme, which prevents the enzyme from binding substrate. If the \(E \cdot S\) complex cannot form, then product cannot form either. Chemically, it can be written as:

\[
E + S \overset{k_{on}}{\rightleftharpoons} E \cdot S \overset{k_{1}}{\rightleftharpoons} E \cdot I \overset{k_{2}}{\rightleftharpoons} E + P + I \tag{13}
\]

This is very similar to the Michaelis-Menten scheme, except for the addition of an extra possible state, in which the inhibitor is bound to the enzyme. Formation of this enzyme-inhibitor complex occurs according to the following equation:

\[
\frac{d[E \cdot I]}{dt} = k_{on}[E][I] - k_{off}[E \cdot I] \tag{14}
\]

Intuitively, it seems logical that the greater \(k_{on}\) is relative to \(k_{off}\), the better an inhibitor will be. A typical measure of inhibitor binding is \(\frac{k_{on}}{k_{off}}\), termed \(K_i\). Assuming that the inhibitor and enzyme are at equilibrium, then \(K_i = \frac{[E][I]}{[E \cdot I]}\).

Due to its similarity to the classical Michaelis-Menten equation, we can derive an equation for the rate of

\(^6\)The only difference being that in the irreversible case, \(k_{off}\) is equal to zero, whereas it is allowed to vary in the reversible case.

\(^7\)The active site of an enzyme is the part of the enzyme which actually comes into contact with the substrate and performs the catalysis. One might ask - why do we have the rest of the enzyme then? The rest functions basically to stabilize the active site.
product formation in essentially the same manner as we did previously.

\[
\frac{d[P]}{dt} = k_2[E \cdot S]
\]

\[
= \frac{k_2[E \cdot S][E_{total}]}{[E] + [E \cdot S] + [E \cdot I]}
\]

\[
= \frac{k_2[S][E_{total}]}{[E] + \frac{[E][S]}{K_m} + \frac{[E][I]}{K_i}}
\]

\[
= \frac{k_2[S][E_{total}]}{K_m + [S] + \frac{[I][K_m]}{K_i}}
\]

\[
= \frac{k_2[S][E_{total}]}{K_m \left(1 + \frac{[I]}{K_i}\right) + [S]}
\] (15)

Thus it can be seen that introducing an inhibitor has essentially the same effect as increasing the \(K_m\) by \(100 \times \frac{[I]}{K_i}\)\%. Since the \(K_m\) can be interpreted either as the concentration of substrate at which velocity is half-maximal or simply as a measure of binding (lower values indicate tighter binding), increasing this value decreases the rate at which substrate binds enzyme, and thus the rate at which product is formed.

We now turn to discuss a simple example. Acetylcholinesterase is an enzyme in the nervous system which ensures that when motor neurons fire, the stimulus they generate only lasts for a fraction of a second. The enzyme does this by degrading acetylcholine, a neurotransmitter specifically involved in motor neuron activity. If acetylcholine in the synapse is allowed to persist, the stimulus continues longer than it ought to. Thus, inhibiting acetylcholinesterase increases synapse activity. While these inhibitors have been used as weapons (nerve gases), at low doses, they can be effective therapeutics for Alzheimer’s and myasthenia gravis. Tacrine (Cognex) is a known inhibitor of acetylcholinesterase, with a \(K_i\) of 40nM [1].

Suppose we wanted to know what concentration inhibitor would be necessary to decrease the velocity of acetylcholinesterase activity to an arbitrary percentage \(p\) of \(V_{max}\), given that when the synapse fires, the concentration of substrate raises to .5mM within several milliseconds [3], and that the \(K_m\) of acetylcholinesterase is 8µM [14].

\[
pV_{max} = \frac{V_{max} [S]}{K_m \left(1 + \frac{[I]}{K_i}\right) + [S]}
\]

With a bit of rearranging, this becomes

\[
[I] = K_i \left(\frac{[S]}{pK_m} - \frac{[S]}{K_m} - 1\right)
\]

\[
= K_i \left(\frac{[S]}{K_m} \left(\frac{1}{p} - 1\right) - 1\right)
\] (16)

If we wished to make the velocity of acetylcholinesterase catalysis equal to 1% of \(V_{max}\), then

\[
[I] = 4 \times 10^{-8} \left(\frac{.0005}{.01 \times (8 \times 10^{-6})} - \frac{.0005}{8 \times 10^{-6}} - 1\right)
\]

\[
= 0.00024746 = 247 \mu M
\]

However, it is important to remember that \([I] = [I_{free}] \neq [I_{total}]\), so the utility of the this formula is limited. Better estimates can be obtained by numerical solutions.
4 Comparing Michaelis-Menten vs. Exact (Numerical) Solutions

I have utilized the 4th-order Runge-Kutta method to estimate time-dependent solutions to the above systems in R [10]. Code for this method is attached. The Runge-Kutta method is essentially an improved version of the improved Euler’s method, in which the derivatives around the current point are sampled and averaged to provide a tangent line upon which the numerical solution proceeds forward some small increment in time [2]. This process is iterated and produces numerical estimates of the progression of a system of differential equations over time. Graphs of exact (numerical Runge-Kutta) solutions versus Michaelis-Menten approximations are shown below in Figures 3 and 4.

Figure 3: Simulated values via fourth-order Runge-Kutta (with timestep=.005s) are the thin lines, dotted lines represent Michaelis-Menten Approximation, assuming that $\frac{ds}{dt} = -\frac{d[p]}{dt}$ and that $[I] = [I_{total}]$. It can be seen the rate of product formation is drastically overestimated by the Michaelis-Menten scheme if the approximation $[S] = [S_{total}]$ does not hold very well. Black, red, green and blue correspond to $[I_{total}] = 0, 1, 2$ and $3\mu M$, respectively.

In trying to use the Michaelis-Menten equation to predict substrate levels over time, we must know the amount of free substrate, $[S]$. Since this is rarely the case in practice, $[S]$ is typically replaced with $[S_{total}]$. At high substrate concentrations, this is roughly the case, but it is worth noting that since $[S_{total}] > [S]$, this will always produce overestimates of the reaction rate. This can be seen clearly in both the upper left and lower
right panels of Figure 3.

However, at higher substrate concentrations, this approximation works far better - for example, if we take the conditions of Figure 3, and change the concentration of substrate from 1µM to 10µM, we find the product formation plot approximated by Michaelis-Menten almost perfectly matches the exact solution (see Figure 4).

Similarly, for inhibitors, assuming \([I] = [I_{total}]\) tends to overestimate the effect of an inhibitor, since \([I]\) is always less than \([I_{total}]\).

![Plots](image)

Figure 4: Again, simulated values via fourth-order Runge-Kutta are the thin lines, dotted lines represent Michaelis-Menten approximations, assuming that \(\frac{d[S]}{dt} = \frac{d[P]}{dt}\) and that \([I] = [I_{total}]\). Black and red correspond to \([I_{total}] = 0 \text{ and } 1\mu M\), respectively. As compared to Figure 3, the Michaelis-Menten approximation now much more accurately predicts product formation, although it does not account for initially very high levels of substrate consumption (in order to reach steady-state, where \(d[E \cdot S]/dt = 0\)).

5 Alternative Enzyme Models

There are a variety of kinetic models which have been omitted from this review, mainly due to time and space constraints. However, I would like to briefly mention two other interesting areas of enzyme kinetics. For more on any of these topics, Segel [13] provides excellent coverage.

- Allosteric regulation and substrate inhibition
- Interfacial enzymes
5.1 Allosteric regulation and substrate inhibition

Allosteric regulation refers to the situation in which an enzyme has multiple binding sites, and binding at one site affects the affinity of another site for a particular ligand. Mathematically, this could be thought of as though \( k_1, k_{-1}, k_2, \) and \( k_{-2} \) all being functions of the concentration of a regulating molecule. In some cases, it is the product itself which downregulates activity of the enzyme as a negative feedback process (called negative cooperativity). In other cases, binding in one active site increases the affinity of the other active sites for ligands. This is called positive cooperative binding. An example of this is hemoglobin, which although it is not an enzyme (it does not catalyze a reaction), binds oxygen very differently depending on how many oxygen molecules are already bound to it [8].

5.2 Interfacial enzymes

All models discussed so far have assumed that these reactions are occurring in solution, where concentrations are not dependent on location. However, a large number of proteins are found on or in protein membranes, where it is not appropriate to assume that the concentration of substrate is localized in certain places, typically on membranes. The scheme typically used to describe the activity of an interfacial enzyme is shown in Figure 5.

Figure 5: Scheme used to describe interfacial catalysis, from [6]. Enzyme must first bind to the interface, and only then can it bind substrate or inhibitor.

6 Summary

While it is far beyond the scope of this review, one of the most exciting applications of being able to set up and numerically estimate the results a system of equations in this manner concerns metabolic processes. Metabolic processes consist of large pathways in which enzymes interact both with each other and with substrates to maintain homeostasis within an organism. They must therefore be stable, and capable of adapting to stimuli
(for example, eating a meal increases blood glucose levels, which the body must be able to handle). There is tremendous potential to model these metabolic pathways to determine what kinetic parameter cause what kinds of behavior within an organism. However, in order to be able to accurately model metabolic behavior in any complex organism, it is essential to first be able to measure and understand kinetic constants of enzymes.

In this review, I have covered several simple models of enzyme kinetics, including Michaelis-Menten for single-substrate reactions with and without inhibitors. I have shown that these approximations can work well by comparing them to exact (numerical) solutions, but that if the assumptions behind various substitutions are invalid, the model substantially overpredicts reaction rates. I have also briefly touched on ways to model other enzyme systems, and note that these systems are also easily modeled by setting up a system of differential equations and numerically estimating them using the fourth-order Runge-Kutta method.

References


# Appendix: Runge-Kutta Source Code

```r
# Set up string concatenation operator #
#
":" <- function(...) UseMethod(":")
".default" <- .Primitive(":")
".character" <- function(...) paste(...,sep="")

#Multivariate Runge-Kutta (4th order)
# f = list of functions (length=n)
# all functions must take a vector of n+1 variables, even if
# they don't use them. the first one must be t, and the others
# in order.
# initials = initials for the variables
# tstart = starting time value
# tstop = ending time value
# step = timestep

mv.runge.kutta = function (f, initials, tstart, tstop, step) {
  # verify valid arguments
  if (length(initials) != length(f))
    stop("initials must be a of equal length to f")
  nf = length(f)
  for (i in nf){
    if (!is.function(f[[i]]))
      stop("f must be a list of functions")
    if (!is.numeric(initials[i]))
      stop("initials must be a vector of numerics!")
  }
  if (!is.numeric(tstart))
    stop("tstart must be numeric!")
  if (!is.numeric(tstop))
    stop("tstop must be numeric!")
  if (!is.numeric(step))
    stop("step must be numeric!")
  if (tstart >= tstop)
    stop("tstart must be less than tstop!")
  steps = ceiling((tstop- tstart)/step)+1
  y <- matrix(nrow=steps, ncol=nf+1)
  y[1,] = c(tstart, initials)
  for (i in 1:(steps-1)) {
    a = b = c = d = vector(length=nf)
    for (j in 1:nf)
      a[j] <- f[[j]](y[i,])
    for (j in 1:nf)
      b[j] <- f[[j]](y[i,]+c(step/2,step/2*a))
    for (j in 1:nf)
      c[j] <- f[[j]](y[i,]+c(step/2,step/2*b))
    for (j in 1:nf)
      d[j] <- f[[j]](y[i,]+c(step,step*c))
    y[i+1, 1] = y[i]+step
    y[i+1, 2:(nf+1)] = y[i, 2:(nf+1)] + step/6*(a+2*b+2*c+d)
  }
  y
}

# The actual simulator

simEnzyme = function (k.1,
```
k.neg1,
k.2,
k.neg2,
S_tot,
E_tot,
I_tot,
k.on,
k.off,
timestep,
endtime,
showMM)
{
inits = c(S_tot, 0, 0)
cat("\nInitial values: [ES] = 0, [EI] = 0, [P] = 0")

dS.dt = function(x) {
  -k.1*x[2]*(E_tot-x[3]-x[4])+k.neg1*x[4]
}
dEI.dt = function(x) {
  k.on*(E_tot-x[3]-x[4])*x[2]-k.off*x[3]
}
dES.dt = function(x) {
  k.1*x[2]*(E_tot-x[3]-x[4])+k.neg2*x[5]*(E_tot-x[4]) - (k.neg1+k.2)*x[4]
}
dP.dt = function(x) {
  k.2*x[4] - k.neg2*x[5]*(E_tot-x[3]-x[4])
}
nconcs = length(I_tot)
fxns = list(dS.dt, dEI.dt, dES.dt, dP.dt)

#Dimensions of the following are [CONC, TIMES(ROWS), COLUMN(SPECIES)]
#second set of inhibitor concentrations is for MM approximation
all_i_plots = array(dim=c(nconcs*2, endtime/timestep+1, 5))

par(mfrow=c(2,2))
n=1

## NUMERICAL (EXACT) SOLUTION
## x[1]=t
## x[2]=[S]
## x[3]=[EI]
## x[4]=[ES]
## x[5]=[P]

for(i in I_tot){
dEI.dt = function(x) {
  k.on*(E_tot-x[3]-x[4])*x[2]-k.off*x[3]
}
fxns = list(dS.dt, dEI.dt, dES.dt, dP.dt)
all_i_plots[n,,1:3] = mv.runge.kutta(fxns, inits, 0,endtime,timestep)
n=n+1
}
if (showMM){
  ## MICHAELIS-MENTEN APPROXIMATION
  # PRESUMES [S] = [S_total] and [I] = [I_total]
  #z[2] = [S]
  #z[3] = [EI]
  #z[4] = [ES]
  #z[5] = [P]
  for (i in I_tot){
    K_m = (k.neg1+k.2)/k.1
    K_i = k.on/k.off
dP.dt = function(x) (k.2*E_tot*x[2])/(K_m*(1+i/K_i) + x[2])
dS.dt = function(x) -(k.2*E_tot*x[2])/(K_m*(1+i/K_i) + x[2])
    all_i_plots[n,,1:3] = mv.runge.kutta(list(dS.dt, dP.dt), c(S_tot, 0), 0,endtime,timestep)
  n=n+1
  }
}

species = 4
species.names = c("[S]", ",[E.I]", ",[E.S]", ",[P]"")
#k = species index (column 3 of all_i_plots)
#
j = concentration index (column 1 of all_i_plots)

for (k in 1:species) {
  #PLOT THE EXACT SOLUTIONS
  plot(all_i_plots[1,,1], all_i_plots[1,,(k+1)], col=1, main=species.names[k], pch=".", xlab="time (s)", ylab=species
  ylim=c(min(all_i_plots[1:nconcs,,(k+1)], na.rm=T), max(all_i_plots[,,(k+1)], na.rm=T))
  if (length(I_tot)> 1){
    for (j in 2:nconcs){
      points(all_i_plots[j,,1], all_i_plots[j,,(k+1)], col=j, pch=".")
    }
  }
  if (showMM){
    #PLOT THE MM SOLUTIONS
    totalsteps = endtime/timestep+1
    plotsteps = seq(from=1, to=totalsteps, by=ceiling(totalsteps/25))
    if (species.names[k] == "[S]"){
      for (j in 1:nconcs){
        lines(all_i_plots[nconcs+j,plotsteps,1], all_i_plots[nconcs+j,plotsteps,2], col=j, pch=20, type
      }
    }
    if (species.names[k] == "[P]"){
      for (j in 1:nconcs){
        lines(all_i_plots[nconcs+j,plotsteps,1], all_i_plots[nconcs+j,plotsteps,3], col=j, pch=20, type
      }
    }
  }
  par(mfrow=c(1,1))
  title(main=c("Plots for k1=":k.1:, k.neg1=":k.neg1:, k.2=":k.2:, k.neg2=":k.neg2,
  "S_tot=":S_tot:, E_tot=":E_tot:, I_tot=":paste(I_tot, collapse=","):\), cex.main=.7)
  all_i_plots
}

data = simEnzyme(k.1=1, k.neg1=.1, k.2=.4, k.neg2=.1, S_tot=10, E_tot=.1, I_tot=c(0,1),
  k.on=.6, k.off=.3, timestep=.005, endtime=10, showMM=T)