20.320 — Problem Set # 3

October 1st, 2010

Due on October 8th, 2010 at 11:59am. No extensions, no electronic submissions.

General Instructions:

1. You are expected to state all your assumptions and provide step-by-step solutions to the numerical problems. Unless indicated otherwise, the computational problems may be solved using Python/MATLAB or hand-solved showing all calculations. Both the results of any calculations and the corresponding code must be printed and attached to the solutions. For ease of grading (and in order to receive partial credit), your code must be well organized and thoroughly commented, with meaningful variable names.

2. You will need to submit the solutions to each problem to a separate mail box, so please prepare your answers appropriately. Staples the pages for each question separately and make sure your name appears on each set of pages. (The problems will be sent to different graders, which should allow us to get the graded problem set back to you more quickly.)

3. Submit your completed problem set to the marked box mounted on the wall of the fourth floor hallway between buildings 8 and 16.

4. The problem sets are due at noon on Friday the week after they were issued. There will be no extensions of deadlines for any problem sets in 20.320. Late submissions will not be accepted.

5. Please review the information about acceptable forms of collaboration, which was provided on the first day of class and follow the guidelines carefully.

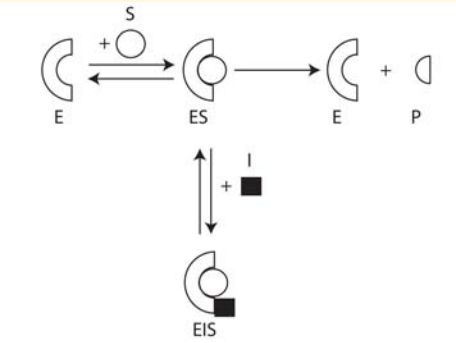
76 points for problem set 3.

1 Enzyme Inhibition

A scientist is interested in developing an inhibitor for a liver enzyme whose product can be harmful. To do so he considers three strategies: i) competitive inhibition, ii) non-competitive inhibition and iii) uncompetitive inhibition. In all cases assume the following kinetic constants: $k_1 = 10^7$ $\mathrm{M}^{-1}\mathrm{s}^{-1}$, $k_{-1} = 5 \cdot 10^{-4}\mathrm{s}^{-1}$, $k_2 = 0.2\mathrm{s}^{-1}$, $k_i = 10^5$ $\mathrm{M}^{-1}\mathrm{s}^{-1}$, $k_{-i} = 10^{-3}$ s^{-1}, $[E]_0 =$ 10 nM, $[S]_0 = 100$ nM.

a) In uncompetitive inhibition, the inhibitor binds only to the enzyme-substrate complex in a reversible manner and impedes it from catalyzing the reaction. Draw the system including all relevant species

Solution:



Total 1 point.

b) Give the governing differential equations that model this system

Solution:

$$
\begin{aligned}
[\dot{E}] &= -k_1 \cdot [E][S] + k_{-1} \cdot [ES] + k_2 \cdot [ES] \\
[\dot{S}] &= -k_1 \cdot [E][S] + k_{-1} \cdot [ES] \\
[\dot{ES}] &= +k_1 \cdot [E][S] - k_{-1} \cdot [ES] - k_2 \cdot [ES] - k_i \cdot [ES] \cdot [I] + k_{-i} \cdot [ESI] \\
[\dot{ESI}] &= +k_i \cdot [ES] \cdot [I] - k_{-i} \cdot [ESI] \\
[\dot{I}] &= -k_i \cdot [ES] \cdot [I] + k_{-i} \cdot [ESI] \\
[\dot{P}] &= k_2 \cdot [ES]
\end{aligned}
$$

Total 2.5 points: 0.5 points per equation.

c) Give the product turnover rate formula for each of these inhibition mechanisms using QSSA. (The derivation to obtain product turnover rate for uncompetitive binding has been done in recitation, you do not need to repeat it)

Solution:

Competitive inhibition

$$\nu = \frac{k_2 \cdot [E]_0 \cdot [S]_0}{\left(1 + \frac{[I]}{K_\mathrm{I}}\right) \cdot K_\mathrm{M} + [S]_0}$$

Non-competitive inhibition

$$\nu = \frac{\frac{k_2 \cdot [E]_0}{\left(1 + \frac{[I]}{K_\mathrm{I}}\right)} \cdot [S]_0}{K_\mathrm{M} + [S]_0}$$

Uncompetitive inhibition

$$\nu = \frac{\frac{k_2 \cdot [E]_0}{\left(1 + \frac{[I]}{K_\mathrm{I}}\right)} \cdot [S]_0}{\frac{K_\mathrm{M}}{\left(1 + \frac{[I]}{K_\mathrm{I}}\right)} + [S]_0}$$

where

$$K_\mathrm{M} = \frac{k_{-1} + k_2}{k_1} \text{ and } K_\mathrm{I} = \frac{k_{-i}}{k_i}$$

Total 1.5 points: 0.5 point per formula

d) Explain what effects each of these different categories of inhibitor have on the product rate formation. Give a qualitative explanation for these effects.

Solution:

- Competitive inhibition: The apparent K_{M} is increased as the concentration of inhibitor increases or its affinity to the enzyme increases. $\nu_{\max}$ is unaffected because for large enough substrate it can compete with the inhibitor. The inhibitor causes the substrate concentration to reach $\nu_{\max}$ to rise and thus larger K_{M} values.
- Non-competitive inhibition: $\nu_{\max}$ is reduced by the presence of inhibitor because its binding reduces the number of available free enzyme for catalysis (recall the definition of $\nu_{\max}$). Since this type of inhibition affects the total number of free enzyme, K_{M} is unaffected.
- Uncompetitive inhibition: $\nu_{\max}$ and K_{M} are reduced as the concentration of inhibitor increase or its affinity to the enzyme increases. $\nu_{\max}$ decreases because the inhibitor reduces the number of available free enzyme for catalysis. K_{M} is lower because the inhibitor acts only on the complexed form of the enzyme and therefore acts by reducing the apparent k_2, therefore, by definition K_{M} is reduced. Also note how uncompetitive inhibition effect is greater with increased substrate concentration.

Total 9 points: 0.5 point for qualitative effect on ν_{max} and K_M for each inhibition type (increase, decrease). 1 point for qualitative explanation for each observed effect.

e) Using plot the product concentration as a function of time without inhibitors. On the same plot do the same for all three inhibition mechanisms with $[I] = 500\mathrm{nM}$

Solution:

The differential equation system can be solved using `ode15s` in

Total 6 points: 1.5 points for each curve

f) For each type of inhibitor, give the concentration at which there will be 50% less product after 100, 1,000, 10,000 and 100,000 seconds. Describe qualitatively the trends you observe.

Solution:

Using the ode solver in and plotting the normalized product concentration in the inhibited case versus the non-inhibited case we obtain:

Time s	**Competitive**	**Non-competitive**	**Uncompetitve**
100	1.5μM	300nM	400nM
1,000	3μM	600nM	800nM
10,00	15μM	3μM	4μM
10,000	150μM	40μM	40μM

We thus observe that competitive inhibition requires the most inhibitor concentration to hinder product formation. Non-competitive inhibition is the most potent at early times. At longer times non-competitive inhibition and uncompetitive inhibition have identical equivalent effects.

Total 8 points: 4 points for running code, 2 points for normalization using the concentration at the final time without inhibitor, 2 points for approximately the correct inhibitor concentrations.

28 points overall for problem 1.

MATLAB code for Problem 1

EnzymeInhibition.m:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%              20.320 Problem set 3
%
%                     Problem 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function Enzyme_Inhibition()
close all;
clc;

k1 = 1e7*1e-9;       % [nM^-1 s^-1]
kminus1 = 0.0005;    % [s^-1]
k2 = 0.2;            % [s^-1]
ki = 1e5*1e-9;       % [nM^-1 s^-1]
kminus_i = 1e-3;     % [s^-1]

E0 = 10;  % [nM]
S0 = 1e2; % [nM]
ES0 = 0;  % [nM]
P0 = 0;   % [nM]
ESI0 = 0; % [nM]
EI0 = 0;  % [nM]


x0 = [E0 S0 0 0];
P = [k2 k1 kminus1];
[T, Y] = ode15s(@(t,y)Enzyme_Kinetics_Equadiff(t, y, P), [0 5000], x0);

x0 = [E0 S0 500 ES0 EI0 P0];
P = [k1 kminus1 k2 ki kminus_i];
[T1, Y1] = ode15s(@(t,y)competitive(t, y, P), [0 5000], x0);

x0 = [E0 S0 500 0 0 0 P0];
[T2, Y2] = ode15s(@(t,y)non_competitive(t, y, P), [0 5000], x0);

x0 = [E0 S0 500 ES0 ESI0 P0];
[T3, Y3] = ode15s(@(t,y)uncompetitive(t, y, P), [0 5000], x0);

figure(1);
plot(T, Y(:,4), 'k');
hold on;
plot(T1, Y1(:,6), 'b');
plot(T2, Y2(:,7), 'g');
plot(T3, Y3(:,6), 'r');
xlabel('Time [s]');
ylabel('Product concentration [nM]');
legend('[I] = 0', 'Competitive', 'Non-competitive', 'Uncompetitive');
title('[I] = 500nM');
hold off;


figure(2);
I = logspace(1,7, 50);
x0_1 = [E0 S0 0 0];
P_1 = [k2 k1 kminus1];
```

```
Product_end = zeros(3,50);
time = logspace(2,5,4);
for t = 1:4
    for i = 1:50
        x0 = [E0 S0 I(i) ES0 EI0 P0];
        P = [k1 kminus1 k2 ki kminus_i];
        [T1, Y1] = ode15s(@(t,y)competitive(t, y, P), [0 time(t)], x0);

        x0 = [E0 S0 I(i) 0 0 0 P0];
        [T2, Y2] = ode15s(@(t,y)non_competitive(t, y, P), [0 time(t)], x0);

        x0 = [E0 S0 I(i) ES0 ESI0 P0];
        [T3, Y3] = ode15s(@(t,y)uncompetitive(t, y, P), [0 time(t)], x0);

        Product_end(1,i) = Y1(end,6);
        Product_end(2,i) = Y2(end,7);
        Product_end(3,i) = Y3(end,6);
    end
    [T, Y] = ode15s(@(t,y)Enzyme_Kinetics_Equadiff(t, y, P_1), [0 5000], x0_1);
    max = Y(end, 4);
    Product_end = Product_end./max;

    subplot(4,1,t);
    semilogx(I, Product_end(1,:));
    hold on;
    semilogx(I, Product_end(2,:), 'g');
    semilogx(I, Product_end(3,:), 'r');
    legend('Competitive', 'Non-competitive', 'Uncompetitive', 'Location', 'West');
    plot(log(I)/log(10), 0.5, 'k:');
    axis([1 1e7 0 1]);
    title(sprintf('Time = %0.f s', time(t)));
    hold off;
end
xlabel('Inhibitor concentration nM');
subplot(4,1,2);
ylabel('Product concentration as a fraction of non inhibited case');

%----------------    ODE SYSTEMS    ------------------------------%

function xdot = Enzyme_Kinetics_Equadiff(t, x, P)
    % P = [kcat k1 k_minus1]
    % x = [x(1) x(2) x(3) x(4)] == [S E ES P]
xdot = [-P(2)*x(2)*x(1) + (P(3)+P(1))*x(3);... % dE/dt
        -P(2)*x(2)*x(1) + P(3)*x(3);... % dS/dt
        P(2)*x(2)*x(1) - (P(3)+P(1))*x(3);...%dES/dt
        P(1)*x(3)]; % dP/dt
end


function xdot = competitive(t, x, k)
% x = [E S I ES EI P]
% k = [k1 kminus_1 k2 ki kminus_i]

xdot = [-k(1)*x(1)*x(2) + (k(2) + k(3))*x(4) + k(5)*x(5) - k(4)*x(1)*x(3);...
    -k(1)*x(1)*x(2) + k(2)*x(4);...
    -k(4)*x(1)*x(3) + k(5)*x(5);...
    k(1)*x(1)*x(2) - k(2)*x(4) - k(3)*x(4);...
    k(4)*x(1)*x(3) - k(5)*x(5);...
    k(3)*x(4)];
```

```
114 end
115
116 function xdot = non_competitive(t, x, k)
117 % x = [E S I ES EI ESI P]
118 % k = [k1 kminus_1 k2 ki kminus_i]
119
120 xdot = [-k(1)*x(1)*x(2) + (k(2) + k(3))*x(4) + k(5)*x(5) - k(4)*x(1)*x(3);...
121     -k(1)*(x(1)*x(2) + x(2)*x(5)) + k(2)*(x(4) + x(6));...
122     -k(4)*(x(1)*x(3) + x(3)*x(4)) + k(5)*(x(5)+x(6));...
123     k(1)*x(1)*x(2) - k(2)*x(4) + k(5)*x(6) - k(4)*x(4)*x(3) - k(3)*x(4);...
124     k(4)*x(1)*x(3) - k(5)*x(5) + k(2)*x(6) - k(1)*x(2)*x(5);...
125     k(1)*x(2)*x(5) + k(4)*x(3)*x(4) - (k(5) + k(2))*x(6);...
126     k(3)*x(4)];
127 end
128
129 function xdot = uncompetitive(t, x, k)
130 % x = [E S I ES ESI P]
131 % k = [k1 kminus_1 k2 ki kminus_i]
132
133 xdot = [-k(1)*x(1)*x(2) + (k(2) + k(3))*x(4);...
134         -k(1)*x(1)*x(2) + k(2)*x(4);...
135         -k(4)*x(3)*x(4) + k(5)*x(5);...
136         k(1)*x(1)*x(2) - k(2)*x(4) - k(3)*x(4) - k(4)*x(3)*x(4) + k(5)*x(5);...
137         k(4)*x(3)*x(4) - k(5)*x(5);...
138         k(3)*x(4)];
139
140 end
141
142 end
```

2 The Huang-Ferrell Model of the MAPK Cascade

The Huang-Ferrell model of the mitogen-activated protein kinase cascade captures key emergent features of its function. You have been provided with a implementation of the model. In this problem, you will critically reflect on its assumptions, test its response to perturbations, and extend it to account for the effects of a drug candidate.

Figure reproduced from [1].

`KinaseCascade.m` contains the differential equation model of the cascade pictured above. `PS3_Huang_Ferrell.m` performs numerical integration using different sets of inputs. In Figure 1, it will plot the fractional activation of RAF, Erk, and MEK in response to an input.

a) The cascade exhibits an ultrasensitive, cooperative response to stimulus. In what ways is this cooperativity analogous to that seen in haemoglobin binding to oxygen? In what ways is it different? State and *briefly* explain two similarities and two differences.

Solution:

Similarities

- Same functional phenotype — steep change in output at a critical threshold level of input.
- Can be approximated by Hill equation.
- Ultrasensitivity requires coupling between several events (of binding or catalysis).

Differences

- Cooperative mechanism is atomic-mechanical in Hb, but emerges from mathematical relationship of independent soluble molecules in MAPK cascade.
- Emergent ultrasensitivity can be tuned at many points, but Hb cooperativity has its n_{H} determined by oligomeric structure (hard-wired).

4 points: 1 each for 2 reasonable similarities and 2 reasonable differences.

b) Both in Huang and Ferrell's paper and in this implementation, Michaelis-Menten kinetics were assumed. What is their reason for this? Do you agree with it?

Solution:

- This assumption makes the system computationally more tractable.
- It is justified because the functional form is what matters most, and the system is robust wrt. changes in the values of kinetic constants.

2 points.

c) Now consider the model in detail. What would happen if the cascade did not have any phospatases in it? State and justify a hypothesis. Then, test it computationally. Did you refute your hypothesis or is it consistent with the results? Comment.

Solution:

- **Hypothesis** (various reasonable examples):
  - In the absence of phosphatases, ultrasensitivity is preserved but the transition takes place at a lower stimulus.
  - In the absence of phosphatases, ultrasensitivity is abolished.
  - ...
- **Methods:**
  In `PS3_Huang_Ferrell.m`, set the initial concentrations of all phosphatases to zero, i.e.
  - ERKPase (lines 23–26),
  - MEKPase (lines 27–30),
  - and E2 (lines 31–33).

  This is the easiest approach, but it is of course also valid to edit the ODEs to remove all phosphatase terms.
- **Results:**
  - With phosphatases:

    □
  - Without phosphatases:

    □

Solution:

- **Conclusions:**
  - The removal of phosphatases from the cascade does not abolish ultrasensitivity (while ultrasensitivity can arise from different types of interactions, a deeper analysis than was undertaken here reveals that the *double* phosphorylation of members of the cascade is the critical cause of greater-than-Michaelis-Menten sensitivity).
  - However, the phosphatases are an important tuning point, and removing them shifts the cascade response far toward lower stimulus levels.

10 points total: 2 for stating a reasonable hypothesis, 3 points for a valid approach of taking out the phosphatases (set initial conc to zero, or remove all terms from ODEs) and implementing this in , 2 points for graphs, 3 points for conclusions (note that ultrasensitivity is preserved, and describe what does happen).

d) To interfere with pathologically upregulated cell proliferation, you consider developing an inhibitor of the MAPK cascade. You wonder if an inhibitor which binds the inactive, unphosphorylated form of MEK and prevents its phosphorylation by Raf with an IC_{50} of 2 μM would be an effective way of downregulating the cascade. Create a new model which extends the code you have been given to incorporate such inhibition. Evaluate its impact with the same large input stimulus used in figure 2 of the given code, and plot the steady-state (maximum) output of activated ERK for a range of inhibitor concentrations. How much inhibitor must you add for the level of activated ERK to be reduced by 90%?

HINT: You may approximate rate constants in the same way as has been done here and in the paper.

Solution:

- Use simplified terms for inhibitor influence – see code for details.
- Inhibition response curve:

□

- Calculate that 90% effective inhibition is achieved at [I] = 166.81 μM.
- Note that the IC_{50} is assumed to be implicitly included in the (all-normalized) rate constants. There is no need to explicitly account for it. The point of this problem is to illustrate the principle of simulating *network* responses to a drug to assess likely efficacy, rather than to make accurate predictions as part of this problem set.

7 points total: 5 for curve code and plot), 2 for [I] for 90% inhibition.
Graders: Check the code thoroughly. Apportion partial credit for the correct approach (see code for details), emphasizing the correct method over the specific numerical result. Do not deduct points if a valid calculation was performed, but the ODEs differ by a constant factor.

23 points overall for problem 2.

MATLAB code for Problem 2

PS3HuangFerrellInhibitor.m:

```
1
2  %E1 = Ras-GTP
3  %E2 = RAF Phosphatase
4  %cx = complex
5  %P = phosphate (PO4)
6  %PP = two phosphates
7  %* = activated
8  %Pase = phosphatase enzyme (so MEKPase is MEK phosphatase)
9
10 %initial conditions, all from Huang & Ferrell, 1996
11 RAF = 0.003;        %uM
12 RAFstar = 0;        %uM, initially no activated RAF
13 RAFstar_cx = 0;     %uM, initially no RAF* complex
14 RAFstar1_cx = 0;    %uM, initially no RAF* complex
15 MEK = 1.2;          %uM
16 MEKp = 0;           %uM, initially no phosphorylated MEK
17 MEKpp = 0;          %uM, initially no phosphorylated MEK
18 MEKpp_cx = 0;       %uM
19 MEKpp1_cx = 0;      %uM, initially no phosphorylated MEK complex
20 ERK = 1.2;          %uM
21 ERKp = 0;           %uM, initially no phosphorylated ERK
22 ERKpp = 0;          %uM, initially no phosphorylated ERK
23 ERKPase = 0.12;     %uM
24 ERKPase_cx = 0;     %uM, initially no complex
25 ERKPase1 = 0.12;    %uM
26 ERKPase1_cx = 0;    %uM, initially no complex
27 MEKPase = 0.3e-3;   %uM
28 MEKPase_cx = 0;     %uM, initially no complex
29 MEKPase1 = 0.3e-3;  %uM
30 MEKPase1_cx = 0;    %uM, initially no complex
31 E2 = 0.3e-3;        %uM, input stimulus, 10-fold less abundant than its
32                     %substrate Mos
33 E2_cx = 0;          %uM
34 E1 = 1e-2;          %uM, will vary this input stimulus below
35 E1_cx = 0;          %uM
36
37 %parameters
38 Km = 300;           %nM, Michaelis constant
39 Vmax = 150;         %nM s^-1, from Michaelis Menten
40
41 E1 = logspace(-6, -1, 100);     %uM
42
43 params = [E2,0,ERK,ERKp,ERKpp,MEK,MEKp,MEKpp,RAF,RAFstar,MEKPase, ...
44           MEKPase1, ERKPase,ERKPase1,E2_cx,E1_cx,MEKpp_cx,MEKpp1_cx, ...
45           RAFstar_cx, RAFstar1_cx,MEKPase_cx,MEKPase1_cx,ERKPase_cx, ...
46           ERKPase1_cx];
47
48 t = [0 100];
49
50 for j = 1:length(E1)
51     params(2) = E1(j);
52     [t,y] = ode23s(@KinaseCascade, t, params,[],Km,Vmax);
53     Y1 = y(:,5);
54     Y2 = y(:,8);
```

```
    Y3 = y(:,10);
    Activated_ERK(j) = Y1(length(t)); %just want steady state values
    Activated_MEK(j) = Y2(length(t));
    Activated_RAF(j) = Y3(length(t));

end

%normalize to get percent response
Activated_ERK = Activated_ERK/(Activated_ERK(length(Activated_ERK)));
Activated_MEK = Activated_MEK/(Activated_MEK(length(Activated_MEK)));
Activated_RAF = Activated_RAF/(Activated_RAF(length(Activated_RAF)));

semilogx(E1,Activated_RAF,'b', 'LineWidth', 2);
hold on
semilogx(E1,Activated_MEK,'g', 'LineWidth', 2);
semilogx(E1,Activated_ERK,'r', 'LineWidth', 2);
legend('activated RAF','activated MEK','activated ERK');
title('Ultrasensitivity in the MAPK cascade','FontSize', 16, ...
      'FontWeight', 'bold');
xlabel ('Input stimulus (E1)','FontSize', 12, 'FontWeight', 'bold');
ylabel ('predicted steady-state fractional activation','FontSize', 12, ...
        'FontWeight', 'bold');
set(gca,'FontSize',12, 'FontWeight', 'bold');
hold off;


E1 = 1e-1; %large input stimulus, uM
params(2) = E1;
[t,y] = ode23s(@KinaseCascade, t, params,[],Km,Vmax);
activatedERK = y(:,5);
figure(2)
plot(t,activatedERK, 'LineWidth', 2);
title('ERK output over time for large input stimulus','FontSize', 16, ...
      'FontWeight', 'bold');
xlabel ('time','FontSize', 12, 'FontWeight', 'bold');
ylabel ('active ERK concentration / nM', 'FontSize', 12, ...
        'FontWeight', 'bold');
set(gca,'FontSize',12, 'FontWeight', 'bold');


%part d, solution

%%%Inhibitor concentration at 90% reduction in activated ERK
%%%[I] = 166.8101 uM (see nested if-statements for how this was calculated)

I = logspace(0, 4, 100); %uM
MEKI_cx = 0; %none initially
params = [E2,E1,ERK,ERKp,ERKpp,MEK,MEKp,MEKpp,RAF,RAFstar,MEKPase, ...
          MEKPase1,ERKPase,ERKPase1,E2_cx,E1_cx,MEKpp_cx,MEKpp1_cx, ...
          RAFstar_cx,RAFstar1_cx,MEKPase_cx,MEKPase1_cx,ERKPase_cx, ...
          ERKPase1_cx, 0, 0];
printed = 0;
for j = 1:length(I)
    params(25) = I(j);
    [t,y] = ode23s(@KinaseCascadeInhibitor, t, params,[],Km,Vmax);
    Y1 = y(:,5);
    Activated_ERK(j) = Y1(length(t)); %just want steady state value
    if (j>1 & printed==0)
        if (Activated_ERK(j) < (0.1*Activated_ERK(1)))
```

```
            'printing inhibitor conc at 90% reduction in activated ERK'
            I(j)
            printed = 1;
        end
    end
 end

figure(3);
semilogx(I,Activated_ERK,'b', 'LineWidth', 2);
title('Predicted response to MEK-inhibitor','FontSize', 16, ...
      'FontWeight', 'bold');
xlabel ('[I] / uM','FontSize', 12, 'FontWeight', 'bold');
ylabel ('active ss ERK concentration / nM', 'FontSize', 12, ...
        'FontWeight', 'bold');
set(gca,'FontSize',12, 'FontWeight', 'bold');
```

KinaseCascadeInhibitor.m:

```
%KinaseCascade function: will use to return ERK, MEK, and RAF values

%E1 = Ras-GTP
%E2 = RAF Phosphatase
%cx = complex
%P = phosphate (PO4)
%PP = two phosphates
%* = activated
%Pase = phosphatase enzyme (so MEKPase is MEK phosphatase)

function myfun = KinaseCascadeInhibitor(t,y,Km,Vmax)
% y1 = dE2dt
% y2 = dE1dt
% y3 = dERKdt
% y4 = dERKPdt
% y5 = dERKPPdt
% y6 = dMEKdt
% y7 = dMEKPdt
% y8 = dMEKPPdt
% y9 = dRAFdt
% y10 = dRAF*dt
% y11 = dMEKPasedt
% y12 = dMEKPase1dt
% y13 = dERKPasedt
% y14 = dERKPase1dt
% y15 = dE2_cxdt
% y16 = dE1_cxdt
% y17 = dMEKPP_cxdt
% y18 = dMEKPP.MEKPP1_cxdt
% y19 = dRAF*_cxdt
% y20 = dRAF*.RAF*_cxdt
% y21 = dMEKPase_cxdt
% y22 = dMEKPase1.MEKPase_cxdt
% y23 = dERKPase_cxdt
% y24 = dERKPase1.ERKPase_cxdt

myfun(1,:) = - 1000 * y(10) * y(1)+ Km * y(15);
myfun(2,:) = - 1000 * y(9) * y(2)+ Km * y(16);
myfun(3,:) = - 1000 * y(3)* y(8)+ Vmax * y(17)+ Vmax * y(23);
myfun(4,:) = - 1000 * y(4)* y(8)+ Vmax * y(18)- 1000 * y(4)* y(13) ...
             + Vmax * y(23)+ Vmax * y(17)+ Vmax * y(24);
myfun(5,:) = - 1000 * y(5)* y(14)+ Vmax * y(24)+ Vmax * y(18);
% myfun(6,:) = - 1000 * y(6)* y(10)+ Vmax * y(19)+ Vmax * y(21);
% change dMEKdt: add another term for inhibitor binding MEK and for
% MEKI_cx becoming MEK
myfun(6,:) = - 1000 * y(6)* y(10) - 1000 * y(6) * y(25) + Vmax * y(19)...
             + Vmax * y(21) + Vmax* y(26);
myfun(7,:) = - 1000 * y(7)* y(10)+ Vmax * y(20)- 1000 * y(7)* y(11) ...
             + Vmax * y(21)+ Vmax * y(19)+ Vmax * y(22);
myfun(8,:) = - 1000 * y(8)* y(12)+ Vmax * y(22)+ Vmax * y(20) ...
             - 1000 * y(3) * y(8)+ Km * y(17)- 1000 * y(4) * y(8) ...
             +  Km * y(18);
myfun(9,:) = - 1000 * y(9)* y(2)+ Vmax * y(16)+ Vmax * y(15);
myfun(10,:) = - 1000 * y(10)* y(1)+ Vmax * y(15)+ Vmax * y(16) ...
              - 1000 * y(6) * y(10)+ Km * y(19)- 1000 * y(7) * y(10) ...
              + Km * y(20);
```

```
57 myfun(11,:) = - 1000 * y(7) * y(11)+ Km * y(21);
58 myfun(12,:) = - 1000 * y(8) * y(12)+ Km * y(22);
59 myfun(13,:) = - 1000 * y(4) * y(13)+ Km * y(23);
60 myfun(14,:) = - 1000 * y(5) * y(14)+ Km * y(24);
61 myfun(15,:) = 1000 * y(10) * y(1)- Km * y(15);
62 myfun(16,:) = 1000 * y(9) * y(2)- Km * y(16);
63 myfun(17,:) = 1000 * y(3) * y(8)- Km * y(17);
64 myfun(18,:) = 1000 * y(4) * y(8)- Km * y(18);
65 myfun(19,:) = 1000 * y(6) * y(10)- Km * y(19);
66 myfun(20,:) = 1000 * y(7) * y(10)- Km * y(20);
67 myfun(21,:) = 1000 * y(7) * y(11)- Km * y(21);
68 myfun(22,:) = 1000 * y(8) * y(12)- Km * y(22);
69 myfun(23,:) = 1000 * y(4) * y(13)- Km * y(23);
70 myfun(24,:) = 1000 * y(5) * y(14)- Km * y(24);
71
72 %also add an equation for dIdt
73 myfun(25,:) = -1000 * y(6) * y(25) + Vmax * y(26);
74 %and for MEKI_complex
75 myfun(26,:) = 1000 * y(6) * y(25) - Vmax * y(26);
```

3 Kinase Specificity and Competition

To parametrize physico-chemical models of cellular pathways, we often characterize their binding and catalytic properties *in vitro*. This, however, does not faithfully reflect the cellular context. The presence of a large number of competing potential substrates is particularly difficult to account for; conservative estimates put the number of amino acid sites which can potentially be phorphorylated by a protein kinase in the tens of thousands in an average human cell. Here, you will computationally investigate the ability of a protein kinase to discriminate against one particular non-cognate substrate.

Figure reproduced from [2].

The cognate substrate A (orange) and the non-cognate substrate B (green) are both bound by the kinase with equal rates and phosphorylated with equal rate. However, B dissociates much more readily from the kinase than does A. Both phosphorylated forms pA and pB are dephosphorylated by a constitutive phosphatase.

a) Write out chemical equations for all reactions.

Solution:

$$E + A \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} EA \xrightarrow{k_2} E + pA$$

$$E + B \underset{k_{-3}}{\overset{k_3}{\rightleftharpoons}} EB \xrightarrow{k_4} E + pB$$

$$pA \xrightarrow{k_p} A$$

$$pB \xrightarrow{k_p} B$$

2 points.

b) Provide ordinary differential equations for the time-evolution of the concentrations of all chemical species.

Solution:

$$
\begin{aligned}
[\dot{E}] &= -k_1[A][E] - k_3[B][E] + k_{-1}[EA] + k_2[EA] + k_{-3}[EB] + k_4[EB] \\
[\dot{A}] &= -k_1[A][E] + k_{-1}[EA] + k_p[pA] \\
[\dot{B}] &= -k_3[B][E] + k_{-3}[EB] + k_p[pB] \\
[\dot{EA}] &= +k_1[A][E] - k_{-1}[EA] - k_2[EA] \\
[\dot{EB}] &= +k_3[B][E] - k_{-3}[EB] - k_4[EB] \\
[\dot{pA}] &= +k_2[EA] - k_p[pA] \\
[\dot{pB}] &= +k_4[EB] - k_p[pB]
\end{aligned}
$$

3.5 points.

c) Formulate all necessary conservation laws (mass balance equations).

Solution:

$$
\begin{aligned}
E_0 &= [E] + [EA] + [EB] \\
A_0 &= [A] + [EA] + [pA] \\
B_0 &= [B] + [EB] + [pB]
\end{aligned}
$$

1.5 points.

d) Substitute the conservation relations into the rate equations.

Solution:

This problem was removed from the problem set. For numerical integration, it is not necessary to substitute the conservation relations. This would *be necessary in order to find the steady-state phosphorylated fractions by solving for the roots of the derivatives instead.*

e) In implement a function which encodes this modified system of differential equations. Parametrize it with the following values[1] for rate constants and initial conditions:

[1] The parameter values are mostly physiologically reasonable; however, some of the rate constants have been chosen to better illustrate a particular point. Note that in the caption of Figure 6 of the paper [2], the values given for k_{-1} and k_{-3} are 10^6 times too large.

Parameter	Description	Value and units
E	Initial kinase concentration	10^{-1}–10^{1} μM
A	Initial concentration of A	100 μM or 0 μM
B	Initial concentration of B	100 μM or 0 μM
k_1	Association rate constant for A and E	$1 \cdot 10^6$ $s^{-1}M^{-1}$
k_{-1}	Dissociation rate constant for E:A	1 s^{-1}
k_2	Catalytic rate constant for phosphorylation of A	3 s^{-1}
k_3	Association rate constant for B and E	$1 \cdot 10^6$ $s^{-1}M^{-1}$
k_{-3}	Dissociation rate constant for E:B	30 s^{-1}
k_4	Catalytic rate constant for phosphorylation of B	3 s^{-1}
k_p	Dephosphorylation rate constant	0.1 s^{-1}

For a range of kinase concentrations from 10^{-1}–10^{1} μM, plot (on the same graph) the phosphorylated *steady-state* fractions of substrate A and of substrate B as a function of kinase concentration with no competition (only one substrate present at a time). Do you think the kinase will be able to discriminate between these two substrates *in vivo*?

HINT: As units, use μM for concentrations and 10^6 M^{-1} for inverse concentrations. The ODE solvers in will encounter difficulties if you stay in standard SI units because the quantities they see will differ by too many orders of magnitude.

Solution:

□

- Over most of the [kinase] range, $\frac{[pA]/A_0}{[pB]/B_0}$ as one reasonable measure of specificity is $\ll 2$.
- This is not very good for effective discrimination between substrates.
- But *in vivo* context may be different ...

13 points: 10 for implementation, 3 for figure and interpretation (not very good discrimination).

f) Now repeat your analysis with competition (both substrates present at once). How does the result differ — do you see surprising features?

Solution:

□

- Now the phosphorylated fraction of cognate substrate A as a function of kinase concentration barely changes, but that of noncognate substrate B is significantly shifted to the right.
- Over a wide window of kinase concentrations, $\frac{[pA]/A_0}{[pB]/B_0}$ is $\gg 1$, indicating that while most of A is phosphorylated, most of B is not, i.e. that phosphorylation procedes with high apparant specificity.
- Note that in a scenario where [kinase] increases with time, this would allow to phosphorylate A first and then, after a delay, also B, as discussed in class.

3 points (for figure and interpretation).

g) The interaction between the substrates here is conceptually analogous to that between a substrate and an inhibitor. What type of inhibition does it most resemble?

Solution:

Competitive inhibition, since no EAB complex is formed.

1 point.

h) While numerical integration is a powerful tool to predict the time-evolution of a system, it often does not immediately clarify the parameter dependence of the system's behavior. Based on the above analogy and your knowledge of Michaelis-Menten-type rate laws for enzyme inhibition, how do you think the presence of substrate A alters the kinetics of phosphorylation of substrate B? In your answer, clearly state which parameters remain unchanged and which increase or decrease as the concentration of A is increased (assuming a *constant* concentration of the kinase).

Solution:

There will be an increase in K_{M} and no change in $v_{\max}$.
1 point.

25 points overall for problem 3.

MATLAB code for Problem 3

KinaseCompetition.m:

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
%  SOLUTION FOR 20.320 PROBLEM SET 3
%  FALL 2010
%
%  KINASE SPECIFICITY AND COMPETITION
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function KinaseCompetition
clc;
close all;

[k,yo,t,kinase] = kc_init();

% Without competition
yo(2) = 0;
yo(3) = 100;
[fracpa,fracpb] = simulate_ss(k,yo,t,kinase);
fractionpb = fracpb;
yo(2) = 100;
yo(3) = 0;
[fracpa,fracpb] = simulate_ss(k,yo,t,kinase);
fractionpa = fracpa;
plotresults('Without competition',kinase,fractionpa,fractionpb);

% With competition
yo(2) = 100;
yo(3) = 100;
[fractionpa,fractionpb] = simulate_ss(k,yo,t,kinase);
plotresults('With competition',kinase,fractionpa,fractionpb);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%
% Initializes parameters
function [k,yo,t,kinase] = kc_init
k = [1;       % k1    = 1.0 x 10^6 s^(-1) M^(-1)
     1;       % k-1   = 1.0 s^(-1)
     3;       % k2    = 3 s^(-1)
     1;       % k3    = 1.0 x 10^6 s^(-1) M^(-1)
     30;      % k-3   = 30.0 s^(-1)
     3;       % k4    = 3 s^(-1)
     0.1];    % kp    = 0.1 s^(-1)
yo = [2;      % [E1]o = 0.1 uM
      100;    % [A]o  = 100 uM
      100;    % [B]o  = 100 uM
      0;      % e1a
      0;      % e1b
      0;      % pa
      0];     % pb
t = [0 200]; % sufficient to reach steady state
kinase = logspace(-1,1,100);

```

```
%%%%%%%%%%%%%%%%%%%%%%%%
% ODE model
function dydt = kinmodel(t,y,k)
dydt=[-k(1)*y(1)*y(2)-k(4)*y(1)*y(3)+k(2)*y(4)+k(3)*y(4)+k(5)*y(5)+k(6)*y(5);
      % d(E1)/dt
      -k(1)*y(1)*y(2)+k(2)*y(4)+k(7)*y(6);    % d(A)/dt
      -k(4)*y(1)*y(3)+k(5)*y(5)+k(7)*y(7);    % d(B)/dt
      +k(1)*y(1)*y(2)-k(2)*y(4)-k(3)*y(4);    % d(E1:A)/dt
      +k(4)*y(1)*y(3)-k(5)*y(5)-k(6)*y(5);    % d(E1:B)/dt
      +k(3)*y(4)-k(7)*y(6);                   % d(pA)/dt
      +k(6)*y(5)-k(7)*y(7)];                  % d(pB)/dt

%%%%%%%%%%%%%%%%%%%%%%%%
% Performs simulation and returns steady-state phosphorylated fractions
function [fractionpa,fractionpb] = simulate_ss(k,yo,t,kinase)
   for j = 1:length(kinase)
       %iterate through kinase concentrations
       yo(1) = kinase(j);
       [t,y] = ode15s(@kinmodel, t, yo, [], k);
       % calculate phosphorylated fraction
       if(yo(2)≠0)
           fractionpa(j) = y(end,6)/yo(2);
       else
           fractionpa(j)=0;
       end
       if(yo(3)≠0)
           fractionpb(j) = y(end,7)/yo(3);
       else
           fractionpb(j)=0;
       end
   end

%%%%%%%%%%%%%%%%%%%%%%%%
% Create plots
function plotresults(titletext,kinase,fractionpa,fractionpb)
    figure()
    semilogx(kinase,fractionpa,'r-',kinase,fractionpb,'b-', 'LineWidth', 2);
    legend('A','B','Location','NorthWest');
    title(titletext,'FontSize', 16, 'FontWeight', 'bold');
    xlabel ('Kinase concentration / uM','FontSize', 12, 'FontWeight', 'bold');
    ylabel ('Phosphorylated fraction', 'FontSize', 12, 'FontWeight', 'bold');
    set(gca,'FontSize',12, 'FontWeight', 'bold');
```

References

[1] C. Y. Huang and J. E. Ferrell. Ultrasensitivity in the mitogen-activated protein kinase cascade. *Proceedings of the National Academy of Sciences of the United States of America*, 93(19):10078–83, September 1996.

[2] J. A. Ubersax and J. E. Ferrell. Mechanisms of specificity in protein phosphorylation. *Nature Reviews Molecular Cell Biology*, 8(7):530–41, July 2007.

20.320 Analysis of Biomolecular and Cellular Systems
Fall 2012