**Problem 2a**

**The PDF attached shows the details**

**Problem 2b**

**The Matlab code is attached**

%Symbolic variables

syms A B C v1 v2 v3 v4;

%Parameters and equations for this problem

I1\_conc = logspace(-2,3,10);

I2\_conc = logspace(-2,3,10);

A\_conc = zeros(numel(I1\_conc),numel(I2\_conc));

I1\_count = 1;

for i= I1\_conc

I2\_count=1;

for j = I2\_conc

I1 = i;

I2 = j;

Vmax1 = 5.0;

Vmax2 = 5.0;

Vmax3 = 1.0;

Vmax4 = 1.0;

Ks1 = 5.0;

Ks2 = 5.0;

Ks3 = 5.0;

Ks4 = 5.0;

KI1 = 1.0;

KI2 = 1.0;

S\_total = 100.0;

v1 = (Vmax1\*A)/((1+(I1/KI1))\*(Ks1+A));

v2 = (Vmax2\*A)/((1+(I2/KI2))\*(Ks2+A));

v3 = (Vmax3\*B)/(Ks3+B);

v4 = (Vmax4\*C)/(Ks3+C);

eqn1 = v1 - v3 == 0;

eqn2 = v2 - v4 == 0;

eqn3 = A + B + C == S\_total;

eqn4 = B >= 0;

eqn5 = C >= 0;

eqn6 = A >= 0;

eqns = [eqn1,eqn2,eqn3,eqn4,eqn5,eqn6];

S = solve(eqns,[A,B,C]);

value\_A = real(double(S.A));

A\_conc(I2\_count,I1\_count) = value\_A;

I2\_count=I2\_count+1;

end

I1\_count = I1\_count+1;

end

%3D-Plot for [A] vs. [I1] and [I2]

[X,Y]=meshgrid(log10(I1\_conc),log10(I2\_conc));

surf(X,Y,A\_conc)

colormap autumn;

title('change of [A] with [I1] and [I2] for Ks = 5.0')

xlabel('[I1]')

ylabel('[I2]')

zlabel('[A]')



**Problem 2c**

If [A] is considered the output and [I1] and [I2] the inputs, the system is a OR logic gate.

**Problem 2d**

**The Matlab code is attached**

%Symbolic variables

syms A B C v1 v2 v3 v4;

%Parameters and equations for this problem

I1\_conc = logspace(-2,3,10);

I2\_conc = logspace(-2,3,10);

A\_conc = zeros(numel(I1\_conc),numel(I2\_conc));

I1\_count = 1;

for i= I1\_conc

I2\_count=1;

for j = I2\_conc

I1 = i;

I2 = j;

Vmax1 = 5.0;

Vmax2 = 5.0;

Vmax3 = 1.0;

Vmax4 = 1.0;

Ks1 = 35.0;

Ks2 = 35.0;

Ks3 = 35.0;

Ks4 = 35.0;

KI1 = 1.0;

KI2 = 1.0;

S\_total = 100.0;

v1 = (Vmax1\*A)/((1+(I1/KI1))\*(Ks1+A));

v2 = (Vmax2\*A)/((1+(I2/KI2))\*(Ks2+A));

v3 = (Vmax3\*B)/(Ks3+B);

v4 = (Vmax4\*C)/(Ks3+C);

eqn1 = v1 - v3 == 0;

eqn2 = v2 - v4 == 0;

eqn3 = A + B + C == S\_total;

eqn4 = B >= 0;

eqn5 = C >= 0;

eqn6 = A >= 0;

eqns = [eqn1,eqn2,eqn3,eqn4,eqn5,eqn6];

S = solve(eqns,[A,B,C]);

value\_A = real(double(S.A));

A\_conc(I2\_count,I1\_count) = value\_A;

I2\_count=I2\_count+1;

end

I1\_count = I1\_count+1;

end

%3D-Plot for [A] vs. [I1] and [I2]

[X,Y]=meshgrid(log10(I1\_conc),log10(I2\_conc));

surf(X,Y,A\_conc)

colormap autumn;

title('change of [A] with [I1] and [I2] for Ks = 35.0')

xlabel('[I1]')

ylabel('[I2]')

zlabel('[A]')



**Why might this gate be referred to as a fuzzy operator?**

Because when we change Ks from 5 to 35, the results of change of [A] with [I1] and [I2] in 3D plot do not change largely.

**Problem 2e**

Zero-order ultrasensitivity means zero-order in the protein substrate, which saturates the enzyme surface, and ultrasensitive because it is more sensitive than a Michaelis-Menten response to stimulus. The importance of zero-order ultrasensitivity is described in Goldbeter and Koshland (1981). Zero-order ultrasensitivity can used in bicyclic cascade. Additional cycles in a cascade provide the potentiality for increasing the sensitivity of an individual cycle. Mathematical equations can be derived to show that Zero-order ultrasensitivity cannot occur when the converter enzymes operate entirely in the domain of the first-order kinetics. Zero-order ultrasensitivity occurs even when a single effector acts in a noncooperative manner on one of the converter enzymes. With higher turnover numbers for the converter enzymes, lower concentrations of the target proteins, or both, a millisecond time course could be achieved. zero-order ultrasensitivity," in which converter enzymes operating under saturating conditions amplify the response to a signal. A given pathway or cascade can use any one of these mechanisms or all three to enhance its sensitivity.