# 20.320 — Problem Set # 6

October 29th, 2010

*Due on November 5th, 2010 at 11:59am. No extensions will be granted.*

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.

**85 points + 2 EC for problem set 6.**

# 1 Protein engineering scaffolds

Molecules capable of specific, high affinity binding to a molecular target are vital to targeted therapeutics, diagnostics and other biotechnology applications. Antibodies have been largely the scaffold of choice, but also have a large number of shortcomings. In this problem, we will explore the structure of different protein engineering scaffolds: Ankyrin repeat protein (PDB ID: 2JAB), Affibody (PDB ID: 2B88) and $10^{th}$ Fibronectin domain 3 (PDB ID: 1TTG). Structural data for proteins can be obtained from the Protein Data Bank (PDB): www.rcsb.org

a) Give three disadvantages for the use of antibodies as therapeutics:

**Solution:**

- Very large size (150kDa)
- Complex glycosylation
- Multiple domains connected by disulfide bonds
- Reduced diffusivity
- Slow clearance (advantage for therapeutics, but not for imaging)

*3 points*

b) Write a Python program to parse the PDB files and extract the $\Phi$ and $\Psi$ angles.

**Solution:**

First, do not forget to clean the pdb file so that it corresponds to PyRosetta's standards. While in coyote, use the command below:

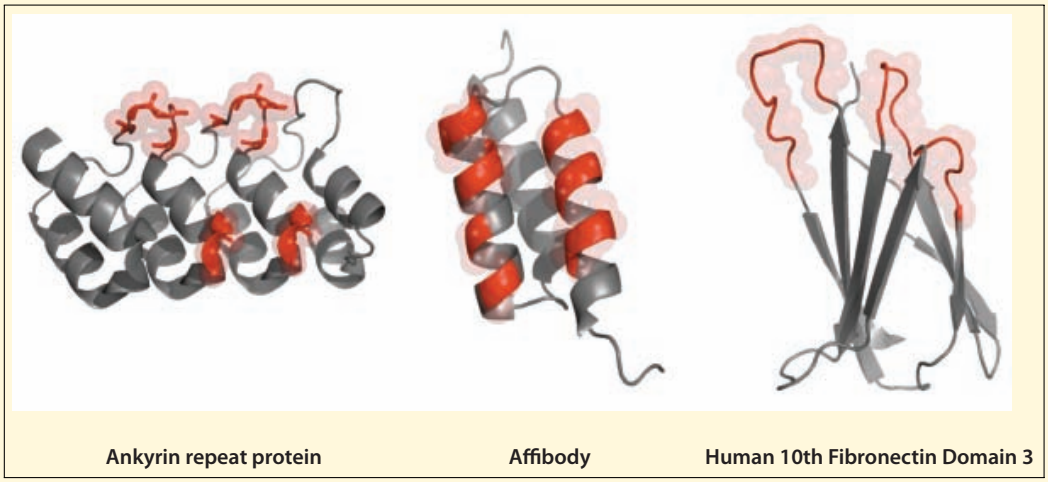
```
grep "^ATOM" 2JAB.pdb > 2JAB.nice.pdb
```

Also be careful that the PDB file for Affibody (2B88) contains multiple time the same protein. Therefore, you need to scan from residues 1 to 58. Likewise for Ankyrin (2JAB), you need to scan from residues 12 to 136 which are read as 1-124 by PyRosetta. For the rest, see code at the end.
*4 points*

c) Residues in alpha helices have $\phi$ = -75°± 30°and $\psi$ = -60 ± 15°while residues in beta-strands have $\phi$ = -100°± 40°and $\psi$ = 135 ± 45°. Given these allowed angle ranges, what fraction of residues are in an alpha-helical conformation? What fraction are in a beta-strand conformation?

**Solution:**

| Protein | $\alpha$ helix | $\beta$-strand |
|---|---|---|
| Affibody | 13/125 | 14/125 |
| Ankyrin repeat protein | 8/53 | 6/53 |
| Fibronectin | 1/94 | 47/94 |




*6 points: allow for some variability depending on the chosen angle criteria.*

d) If you were to engineer a protein with specific binding properties using these scaffolds starting from scratch, which regions in these proteins would you mutate?

**Solution:**

In general it is preferable to preserve the structured region because they tend to confer stability to the overall protein, although this is not always the taken approach. Typically the loops inbetween these structures are the ones targeted for mutagenesis. Take a look at the figure above where in red have been highlighted the mutagenized regions.
*2 points: Accept any coherent answer*

e) **Extra credit:** Tryptophan is one of the most commonly found residue in binding interfaces. What property of tryptophan makes it especially suited for this role?

**Solution:**

Very large surface contact for hydrophobic interactions.
*2 extra credits*

**15 points + 2 EC overall for problem 1.**

## Python code for Problem 1

PS6Q1.py:

```
from rosetta import *
rosetta.init()

p1 = Pose("2B88.nice.pdb")
p2 = Pose("2JAB.nice.pdb")
p3 = Pose("1TTG.nice.pdb")

#for i in range(1, p1.total_residue() + 1):
    #print i, "phi = ", p.phi(i), "psi = ", p.psi(i)

# condition for alpha helical:
# phi = -105 to -45
# psi = -70 to -50
# condition for beta strand
# phi = -160 to -60
# psi = 90 to 180

p_alpha = 0
p_beta = 0
for i in range(1,58+1):
    phi = p1.phi(i)
    psi = p1.psi(i)
    if phi > -105 and phi < -45 and psi > -75 and psi < -45:
        p_alpha = p_alpha + 1
    if phi > -160 and phi < -60 and psi > 90 and psi < 180:
        p_beta = p_beta + 1

p2_alpha = 0
p2_beta = 0
for i in range(1,125+1):
    phi = p2.phi(i)
    psi = p2.psi(i)
    if phi > -105 and phi < -45 and psi > -75 and psi < -45:
        p2_alpha = p2_alpha + 1
    if phi > -160 and phi < -60 and psi > 90 and psi < 180:
        p2_beta = p2_beta + 1

p3_alpha = 0
p3_beta = 0
for i in range(1,p3.total_residue()+1):
    phi = p3.phi(i)
    psi = p3.psi(i)
    if phi > -105 and phi < -45 and psi > -75 and psi < -45:
        p3_alpha = p3_alpha + 1
    if phi > -160 and phi < -60 and psi > 90 and psi < 180:
        p3_beta = p3_beta + 1

print('Affibody has %d residues:'%(p1.total_residue())
print('%d alpha and %d beta\n' %(p_alpha, p_beta))
print('Ankyrin has %d residues:'%(p2.total_residue()))
print('%d alpha-helical and %d beta-strand\n' %(p2_alpha, p2_beta))
print('Fibronectin has %d residues:'%(p3.total_residue()))
print('%d alpha-helical and %d beta-strand\n' %(p3_alpha, p3_beta))
```

## 2 Surface energy

In this problem we will explore the conformation energy of two tri-peptide as a function of the $\chi_1$ angles of the middle reside. We will consider the AVA and ARA peptide.

a) Give 3 components of the scoring function and discuss their relative weights.

**Solution:**
After creation of the score function `scorefxn = create_score_function('standard')`, the command `print scorefxn` displays the componentns of the scoring function. Among many, we note:

| | Definition | Weight |
|---|---|---|
| `fa_atr` | Van der Waals net attractive energy | 0.8 |
| `fa_rep` | Van der Waals net repulsive energy | 0.44 |
| `hbond_sr_bb, hbond_lr_bb` | Hydrogen bonds, short and long-range (backbone-backbone) | 1.1, 1.17 |
| `hbond_bb_sc, hbond_sc` | Hydrogen bonds (backbone-side chain and side chain-side chain) | 1.17, 1.1 |

*2 points: more possible answers can be found on Appendix A of the PyRosetta tutorials.*

b) Set the angles of your petpides so that they are in alpha-helical state.

**Solution:**

- One approach is to have PyRosetta iterate through the different conformation and choose the one that has the minimal energy (see code). The phi and psi angles obtained are listed below:

| | Residue 1 | Residue 2 | Residue 3 |
|---|---|---|---|
| $\phi$ angles for ARA: | 0 | -150 | -150 |
| $\psi$ angles for ARA: | 150 | 150 | 0 |
| $\phi$ angles for AVA: | 0 | -150 | -150 |
| $\psi$ angles for AVA: | 150 | 150 | 0 |

  Interestingly, for such a small peptide, the optimal conformation is not alpha-helical but beta-strand.

- Another approach is to fix keep the side chain angles as that of default by the `make_pose_from_sequence` and change the $\phi$ and $\psi$ angles manually:

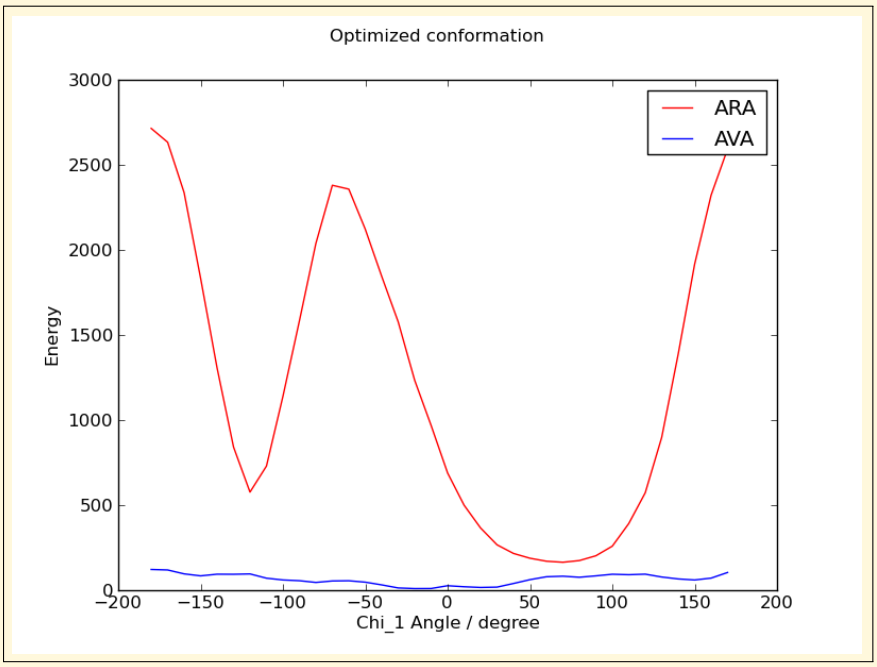
| | Residue 1 | Residue 2 | Residue 3 |
|---|---|---|---|
| $\phi$ angles for ARA: | -55 | -55 | -55 |
| $\psi$ angles for ARA: | -60 | -60 | -60 |
| $\phi$ angles for AVA: | -55 | -55 | -55 |
| $\psi$ angles for AVA: | -60 | -60 | -60 |

*4 points: either method is ok, allow for some variability in the chosen angles, especially for the peptide ends.*

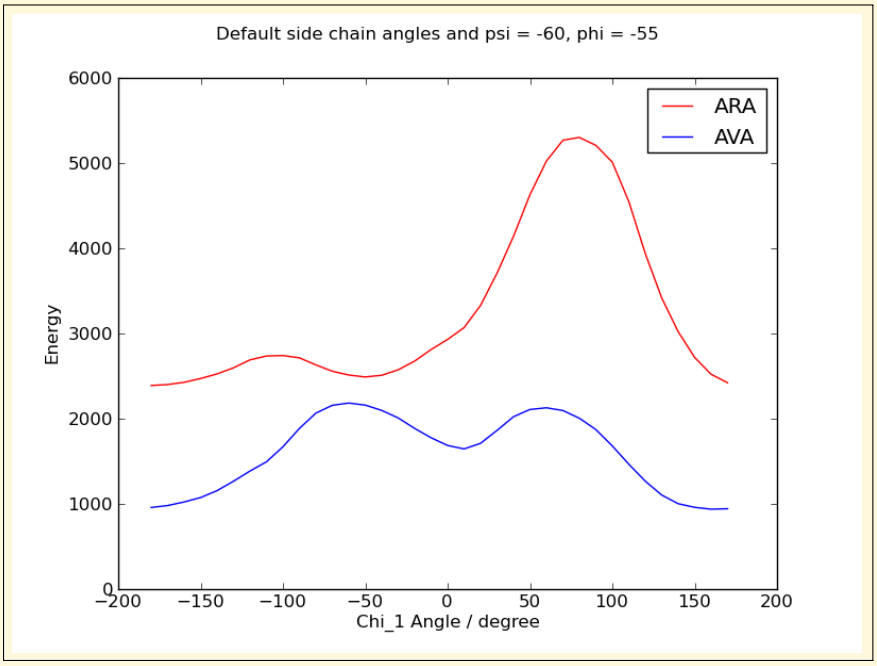
c) For both peptides, plot the energy as a function of the $\chi_1$ angle. To do so, vary the $\chi_1$ angle by 10°from -180°to 180°and calculate the energy of the peptide.

**Solution:**

With the optimized angles:




With the set alpha-helical angles:

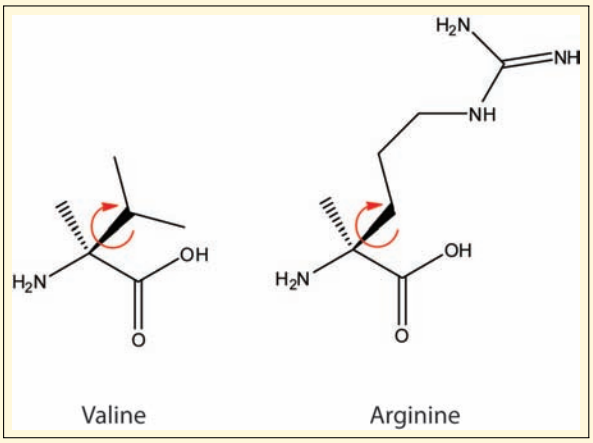



*6 points*

d) What do you observe?

**Solution:**

Given the size of the arginine side chain, its mobility around the $\chi_1$ angle is much more restricted and results thus in large energy differences between the conformation. On the other hand, valine's side chain is much smaller, therefore, it can rotate more freely.




*3 points*

**15 points overall for problem 2.**

## Python code for Problem 2

PS6Q2Code.py:

```
from rosetta import *
from matplotlib import pylab
rosetta.init()
import math
def minimize(p, first=1,last=3): # Minimization function
    minmover=MinMover()
    mmall=MoveMap()
    mmall.set_bb_true_range(first,last)
    minmover.movemap(mmall)
    minmover.tolerance(0.5)
    minmover.apply(p)

out = open('PS6Q2data.txt','w')
out2 = open('PS6Q2data2.txt','w')

p1 = Pose();
make_pose_from_sequence(p1, "ARA", "fa_standard");
p2 = Pose();
make_pose_from_sequence(p2, "AVA", "fa_standard");

scorefxn = create_score_function('standard');

# initialize angles to minimal energy state
minimize(p1)
minimize(p2)
print('Phi angles for ARA: %4.2f\t %4.2f\t %4.2f\n'%(p1.phi(1), p1.phi(2), p1.phi(3)))
print('Psi angles for ARA: %4.2f\t %4.2f\t %4.2f\n'%(p1.psi(1), p1.psi(2), p1.psi(3)))
print('Phi angles for AVA: %4.2f\t %4.2f\t %4.2f\n'%(p2.phi(1), p2.phi(2), p2.phi(3)))
print('Psi angles for AVA: %4.2f\t %4.2f\t %4.2f\n'%(p2.psi(1), p2.psi(2), p2.psi(3)))

# Create chi angles vector
chi = []
for i in range(36):
    chi.append(-180+i*10)

energy_p1 = []
energy_p2 = []
for i in range(len(chi)):
     energy_p1.append([])
     energy_p2.append([])
     p1.set_chi(1,2,chi[i]) # set chi_1 angle of residue 2 to value chi[i]
     p2.set_chi(1,2,chi[i])
     value1 = scorefxn(p1)
     value2 = scorefxn(p2)
     energy_p1[i].append(value1)
     energy_p2[i].append(value2)
     out.write('%4.2f\t%4.2f\t%4.2f\n'%(chi[i],value1,value2))

pylab.figure(1)
pylab.plot(chi,energy_p1, 'r')
pylab.plot(chi,energy_p2, 'b')
pylab.ylabel('Energy')
pylab.xlabel('Chi_1 Angle / degree')
pylab.suptitle('Optimized conformation', fontsize = 12)
```

```
pylab.legend(('ARA', 'AVA'))
pylab.savefig('PS6Q2.png', format='png')
pylab.close(1)

#-------------------------------------------------------------------------#
# Alternatively, we can use the default angles for side chains and
# change psi and phi for alpha-helical state. See the difference...

make_pose_from_sequence(p1, "ARA", "fa_standard");
make_pose_from_sequence(p2, "AVA", "fa_standard");

for k in range(3):
    p1.set_phi(k+1,-55)
    p1.set_psi(k+1,-60)
    p2.set_phi(k+1,-55)
    p2.set_psi(k+1,-60)

energy_p1 = []
energy_p2 = []
for i in range(len(chi)):
    energy_p1.append([])
    energy_p2.append([])
    p1.set_chi(1,2,chi[i]) # set chi_1 angle of residue 2 to value chi[i]
    p2.set_chi(1,2,chi[i])
    value1 = scorefxn(p1)
    value2 = scorefxn(p2)
    energy_p1[i].append(value1)
    energy_p2[i].append(value2)
    out2.write('%4.2f\t%4.2f\t%4.2f\n'%(chi[i],value1,value2))

pylab.figure(2)
pylab.plot(chi,energy_p1, 'r')
pylab.plot(chi,energy_p2, 'b')
pylab.ylabel('Energy')
pylab.xlabel('Chi_1 Angle / degree')
pylab.suptitle('Default side chain angles and psi = -60, phi = -55', fontsize = 12)
pylab.legend(('ARA', 'AVA'))
pylab.savefig('PS6Q2v2.png', format='png')
pylab.close(2)
```

## 3 Interaction prediction

A number of crystallographic studies have shown that the binding interfaces between are generally large and include many intermolecular contacts. However, structural analysis alone cannot show whether all of these contacts are important for tight binding. For the determination of residues important for a binding interaction between two proteins, alanine screening is often performed. In this technique, each amino acid is substituted individually by an alanine. The impact in the interaction can then be measured and the importance of the residue can be assessed. In this problem, you will determine the importance of certain residues of hGH for binding to its receptor. Site 1 of hGH binds to its receptor with a $K_D$ of 0.3nM, corresponding to a binding free energy of $\Delta G$ of -12.3 kcal/mol.

a) Your alanine screening experiment gives the result listed in the table below. Calculate the change in free energy, $\Delta\Delta G$ associated with these mutations.

| Mutation | $K_D$ (nM) |
|---|---|
| R43A | 0.6 |
| E75A | 0.25 |
| I103A | 0.9 |
| W104A | 3.2 |
| I105A | 1.05 |

**Solution:**

The relationship between $K_D$ and the free energy of the complex formation is given by:

$$\Delta G = RT \ln(K_D)$$

We can thus compute the values for $\Delta\Delta G$:

| Mutation | $K_D$ (nM) | $\Delta G$ (kcal/mol) | $\Delta\Delta G$ (kcal/mol) |
|---|---|---|---|
| R43A | 0.6 | -11.93 | 0.39 |
| E75A | 0.25 | -12.43 | -0.10 |
| I103A | 0.9 | -11.71 | 0.62 |
| W104A | 3.2 | -10.99 | 1.33 |
| I105A | 1.05 | -11.62 | 0.70 |

*Total 4 points: 2 points for the $K_D$ - $\Delta G$ relationship, 2 points for numerical application (other units are ok).*

b) Given the $\Delta\Delta G$ for these mutations, which residues are the most important for binding?

**Solution:**

The top 3 are (in order) W104, I105, I103, since they have the most effect on the free energy change for complex formation (more positive).
*2 points*

c) A different mutation on a residue that is burried inside the protein, W23A, yields a $K_D$ of 120nM. Why is there such a dramatic effect on the $K_D$ when this residue is known not to interact with the the receptor?

**Solution:**

Given the large size of tryptophan and its location in the core of the protein, it may have serve stability purposes. Mutation of this residue could potentially disrupt the overall folding of the protein and therefore considerable loss of binding.
*3 points*

d) What does this tell you about the limitation of alanine scanning?

**Solution:**

Alanine scanning can detect residues important for the binding interaction but can also give false positives such as in the W23A scenario above. Therefore, these studies must be conducted along with structural information of the protein-ligand complex.
*3 points*

**12 points overall for problem 3.**

# 4 Protein folding

The p53 tumour suppressor protein is a sequence-specific transcription factor whose function is to maintain genome integrity. Inactivation of p53 by mutation is a key molecular event, which is detected in more than fifty percent of all human cancers. In response to stress insults, the p53 tumour suppressor protein activates a network of genes whose products mediate vital biological functions, the most critical of which are cell cycle arrest and apoptosis. The large majority of the mutations affect residues that are critical for maintaining the structural fold of this a highly conserved DNA-binding (core) domain.
In order to measure the free energy of unfolding, one often uses denaturing conditions such as urea treatment. The free energy change for unfolding, $\Delta G_u$ at each denaturant concentration can be calculated as follows:

$$\Delta G_u = -RT \ln K_u = -RT \ln \frac{[U]}{[F]}$$

Where [U] represents the concentration of unfolded protein and [F] of folded protein. However, we are interested in the unfolding free energy in absence of denaturing conditions. Denaturant-induced unfolding transitions are most commonly interpreted assuming a linear relationship for the denaturant (C):

$$\Delta G_u(C) = \Delta G_u^0 - m \cdot C$$

Given the data measured below at 25°C, what is the free energy of unfolding of this protein?

| [Urea] (M) | Fraction of unfolded protein |
|---|---|
| 2.0 | 0.15 |
| 2.5 | 0.28 |
| 3.0 | 0.35 |
| 3.5 | 0.60 |
| 4.0 | 0.72 |
| 5.0 | 0.83 |

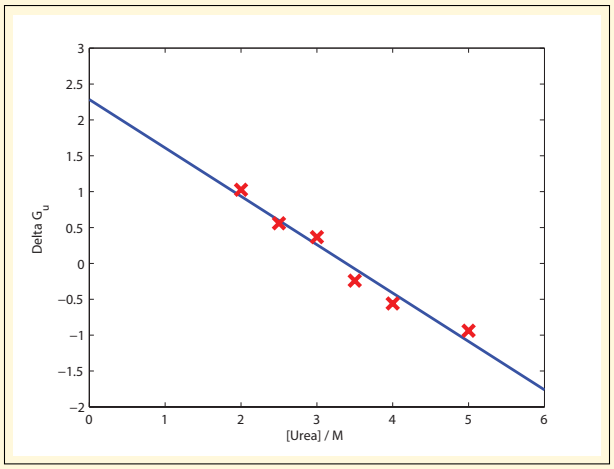
**Solution:**

- The free energy of unfolding can be extrapolated from the free energy of unfolding under denaturing condtions given the linear relation ship between unfolding energy and denaturant concentration. Thus the first step is to calculate the free energy of unfolding under the conditions listed in the table above. The relationship for $\Delta G_u$ can be rewritten as a function of $f_U$, the fraction of unfolded protein:

$$\Delta G_u = -RT \ln \frac{[U]}{[F]} = -RT \ln \frac{f_U}{1 - f_U}$$

- The second step is to perform a linear fit of the free energy of unfolding as a function of denaturant conditions.

- Finally, the free energy of unfolding for the protein corresponds to the intersection with the y-axis, which corresponds to the constant term in the linear fit. The result is: $\Delta G_u^0$ = 2.28 kcal/mol.




*10 points*

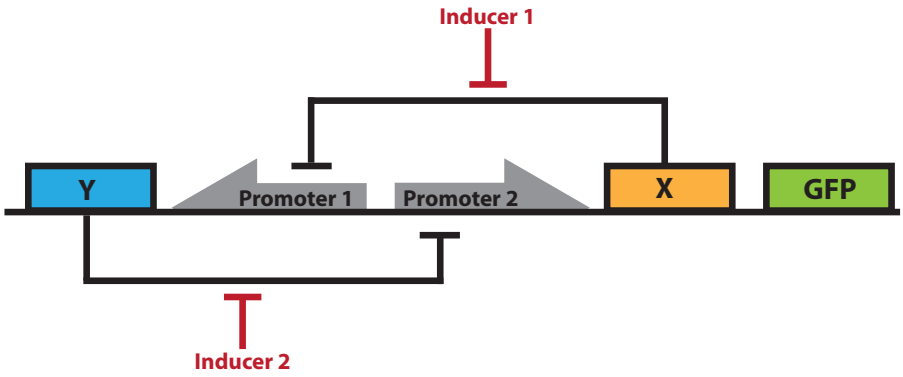
**10 points overall for problem 4.**

## MATLAB code for Problem 4

unfolding.m:

```
function unfolding
clear all;
clc;

U = [2.0 2.5 3.0 3.5 4.0 5.0];
fraction = [0.15 0.28 0.35 0.6 0.72 0.83];

DeltaG = zeros(1,6);
for i=1:length(U)
    DeltaG(i) = -1.986*298*log(fraction(i)/(1-fraction(i)))/1000; % kcal/mol
end


f = inline('a(1)*x + a(2)', 'a', 'x');

constants = nlinfit(U, DeltaG, f, [-2 2]);



nice_plot([0 6], f(constants, [0 6]), '[Urea] / M', 'Delta G_u', '',[0 0 1])
hold on;
plot(U, DeltaG, 'rx', 'MarkerSize', 12, 'LineWidth', 3)
axis([0 6 -2 3]);
hold off;

sprintf('Delta G is = %f', constants(2))



function nice_plot(x,y, Xlab, Ylab, Title, ColorCode)
    plot(x,y, 'Color', ColorCode, 'LineWidth', 2);
    xlabel(Xlab, 'Fontsize', 10);
    ylabel(Ylab, 'Fontsize', 10);
    title(Title, 'Fontsize', 10);
end

end
```

# 5 Qualitative analysis of a nonlinear biochemical circuit




Consider the synthetic genetic circuit above[1]. X and Y are homooligomeric transcription factors mutually repressing each other's expression, with $x = [\text{pX}]$ and $y = [\text{pY}]$. Green fluorescent protein is a readout for the state of the system. Inducers 1 and 2 can be added to perturb the system.

a) Recall this differential equation for transcriptional self-activation with a monomeric active form of the TF:

$$\dot{x} = \beta_0 + \beta_x \frac{x}{\mathrm{K_D} + x} - \alpha x$$

Write down a set of two analogous differential equations describing homooligomeric, mutual repression of two TFs (no derivation required).

**Solution:**

Let $K_x = K_{\mathrm{D,eff,x}}$ and $K_y = K_{\mathrm{D,eff,y}}$. Then,

$$\begin{cases} \dot{x} = \beta_{0,x} + \beta_x \frac{K_y^n}{K_y^n + y^n} - \alpha_x x \\ \dot{y} = \beta_{0,y} + \beta_y \frac{K_x^m}{K_x^m + x^m} - \alpha_y y \end{cases}$$

*4 points.*

b) To simplify the equations, make the following assumptions:

- Both proteins are degraded or diluted at the same rate $\alpha_x = \alpha_y = 1$.
- For both promoters, the maximum rate of expression is $\beta_x = \beta_y = 10$.
- Basal (leaky) expression for both promoters is $\beta_{0,x} = \beta_{0,y} = 0$.
- The effective $\mathrm{K_D}$ is $K_{\mathrm{D,eff},x} = K_{\mathrm{D,eff},y} = 1$.
- The active forms of both repressors are homodimeric.

Write down the simplified set of ODEs.

[1] This problem is inspired by Gardner TS, Cantor CR, and Collins JJ (2000): *Construction of a genetic toggle switch in Escherichia coli.* Nature 403 (6767): 339-342.

**Solution:**

$$\begin{cases} \dot{x} = \frac{10}{1+y^2} - x \\ \dot{y} = \frac{10}{1+x^2} - y \end{cases}$$

*2 points.*

c) Find the critical points $(x_0, y_0)$ of the system where $(\dot{x}, \dot{y})|_{x_0,y_0} = (0, 0)$.
**HINTS:** 1. You will come across a higher-order polynomial equation which you will need to solve. It is solvable by hand, but do feel free to use or an online equation solver such as http://www.numberempire.com/equationsolver.php.
2. Only positive, real-valued roots of that polynomial are biochemically meaningful steady states. Solutions which are negative or complex are mathematically legitimate, but such concentrations of the species in question will never occur; thus, these solutions can be disregarded in the rest of this problem.

**Solution:**

At critical points, $\dot{x} = \dot{y} = 0$. Thus,

$$x_0 = \frac{10}{1 + y_0^2} \tag{1}$$

$$y_0 = \frac{10}{1 + x_0^2} \tag{2}$$

Substitute (2) into (1):

$$x_0 = \frac{10}{1 + \left(\frac{10}{1+x_0^2}\right)^2}$$

$$x_0 \left(1 + \frac{100}{1 + 2x_0^2 + x_0^4}\right) = 10$$

$$x_0 + \frac{100x_0}{1 + 2x_0^2 + x_0^4} = 10$$

$$x_0 \left(1 + 2x_0^2 + x_0^4\right) + 100x_0 = 10 + 20x_0^2 + 10x_0^4$$

$$x_0^5 - 10x_0^4 + 2x_0^3 - 20x_0^2 x_0 + 100x_0 - 10 = 0$$

Solve using an online symbolic equation solver:

$$\boxed{x_0 = 2 \quad \vee \quad x_0 = 5 \pm 2\sqrt{6} \quad \vee \quad (x_0 = -1 \pm 2i)}$$

*(continued on next page)*

For $x_0 = 2$, $y_0 = \frac{10}{1+2^2} = 2$.
For $x_0 = 5 + 2\sqrt{6}$,

$$y_0 = \frac{10}{1 + \left(25 + 20\sqrt{6} + 24\right)} = \frac{10}{50 + 20\sqrt{6}} = \frac{1}{5 + 2\sqrt{6}}$$
$$\frac{y_0}{5 - 2\sqrt{6}} = \frac{1}{\left(5 + 2\sqrt{6}\right)\left(5 - 2\sqrt{6}\right)} = \frac{1}{25 - 24} = 1$$
$$y_0 = 5 - 2\sqrt{6}$$

Analogously, for $x_0 = 5 - 2\sqrt{6}$, $y_0 = 5 + 2\sqrt{6}$.

So altogether, the critical points are

$$(x_0, y_0) = (2, 2)$$
$$(x_0, y_0) = (5 + 2\sqrt{6}, 5 - 2\sqrt{6}) \approx (9.899, 0.101)$$
$$(x_0, y_0) = (5 - 2\sqrt{6}, 5 + 2\sqrt{6}) \approx (0.101, 9.899)$$

*6 points — 2 for each critical point.*

d) Find the Jacobian matrix $\mathbf{A} = \begin{pmatrix} f_x & f_y \\ g_x & g_y \end{pmatrix}$, where $\dot{x} = f(x, y)$ and $\dot{y} = g(x, y)$.

**Solution:**

$$\mathbf{A} = \begin{pmatrix} f_x & f_y \\ g_x & g_y \end{pmatrix} = \begin{pmatrix} -1 & \frac{-20y}{(1+y^2)^2} \\ \frac{-20x}{(1+x^2)^2} & -1 \end{pmatrix}$$

*2 points.*

e) Find the trace $\tau$ and the determinant $\Delta$ of the Jacobian matrix. Then, evaluate both at each of the critical points.

**Solution:**

$$\tau = -2$$
$$\Delta = 1 - \frac{400xy}{(1+x^2)^2(1+y^2)^2}$$

At $(x_0, y_0) = (2, 2)$:

$$\Delta = 1 - \frac{400 \cdot 4}{(1+4)^2(1+4)^2} = 1 - \frac{1600}{625} = \frac{625 - 1600}{625} = -\frac{39}{25}$$

At $(x_0, y_0) = (5 + 2\sqrt{6}, 5 - 2\sqrt{6})$ and, by symmetry, $(x_0, y_0) = (5 - 2\sqrt{6}, 5 + 2\sqrt{6})$:

$$\begin{aligned}
\Delta &= 1 - \frac{400\left(5+2\sqrt{6}\right)\left(5-2\sqrt{6}\right)}{\left(1+\left(5+2\sqrt{6}\right)^2\right)^2\left(1+\left(5-2\sqrt{6}\right)^2\right)^2} \\
&= 1 - \frac{400(25-24)}{\left(1+\left(25+20\sqrt{6}+24\right)\right)^2\left(1+\left(25-20\sqrt{6}+24\right)\right)^2} \\
&= 1 - \frac{400}{\left(50+20\sqrt{6}\right)^2\left(50-20\sqrt{6}\right)^2} \\
&= 1 - \frac{400}{\left(\left(50+20\sqrt{6}\right)\left(50-20\sqrt{6}\right)\right)^2} \\
&= 1 - \frac{400}{(2500-2400)^2} \\
&= 1 - \frac{4}{100} \\
&= 0.96
\end{aligned}$$

*4 points.*

f) For each critical point,

i) Calculate the eigenvalues $\lambda$ of the Jacobian matrix at that point.

ii) Use the eigenvalues to classify the critical point as a stable or unstable equilibrium, a stable or unstable spiral, the center of a periodic orbit, or a saddle.

**Solution:**

$$\lambda_{1,2} = \frac{\tau \pm \sqrt{\tau^2 - 4\Delta}}{2}$$

At $(x_0, y_0) = (2, 2)$:

$$\begin{aligned}
\lambda_{1,2} &= \frac{-2 \pm \sqrt{4 - 4\left(-\frac{39}{25}\right)}}{2} \\
&= -1 \pm \sqrt{1 + \frac{39}{25}} \\
&= -1 \pm \sqrt{\frac{64}{25}} \\
&= -1 \pm \frac{8}{5} \\
\lambda_1 = \frac{3}{5} &\text{ and } \lambda_2 = -\frac{13}{5} \\
\mathrm{Im}\{\lambda_{1,2}\} = 0 &\Rightarrow \text{not a spiral or center;} \\
\mathrm{Re}\{\lambda_1\} > 0 \text{ and } \mathrm{Re}\{\lambda_2\} < 0 &\Rightarrow \text{saddle point.}
\end{aligned}$$

At $(x_0, y_0) = (5 + 2\sqrt{6}, 5 - 2\sqrt{6})$ and, by symmetry, $(x_0, y_0) = (5 - 2\sqrt{6}, 5 + 2\sqrt{6})$:

$$\begin{aligned}
\lambda_{1,2} &= -1 \pm \frac{1}{2}\sqrt{4 - 4 \cdot 0.96} \\
&= -1 \pm \sqrt{1 - 0.96} \\
&= -1 \pm \sqrt{0.04} \\
&= -1 \pm 0.2 \\
\lambda_1 = -1.2 &\text{ and } \lambda_2 = -0.8 \\
\mathrm{Im}\{\lambda_{1,2}\} = 0 &\Rightarrow \text{not a spiral or center;} \\
\mathrm{Re}\{\lambda_{1,2}\} < 0 &\Rightarrow \text{these two critical points are stable steady states.}
\end{aligned}$$

*6 points: 3 for analysis of (0,0) and 3 for* $(5 \pm 2\sqrt{6}, 5 \pm 2\sqrt{6})$.

g) By hand or with the help of (but *without* numerical simulation), sketch a phase portrait of the system:

i) Use $x$ as your horizontal and $y$ as your vertical axis.

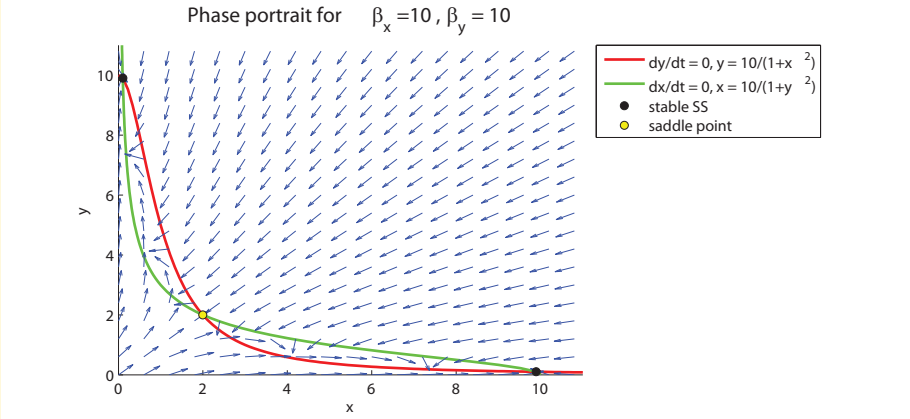
ii) Sketch the nullclines $f(x, y) = 0$ and $g(x, y) = 0$ and mark all critical points. might help to do this accurately.

iii) Use your earlier analysis of the stability of each critical point to sketch local, nearby trajectories around each critical point (best done by hand).

iv) Connect the local trajectories to extend throughout the phase plane.

v) Looking at this phase portrait, can you guess why this circuit might be preferable to the bistable switch implemented via sigmoidal self-activation in the last problem set?

**Solution:**



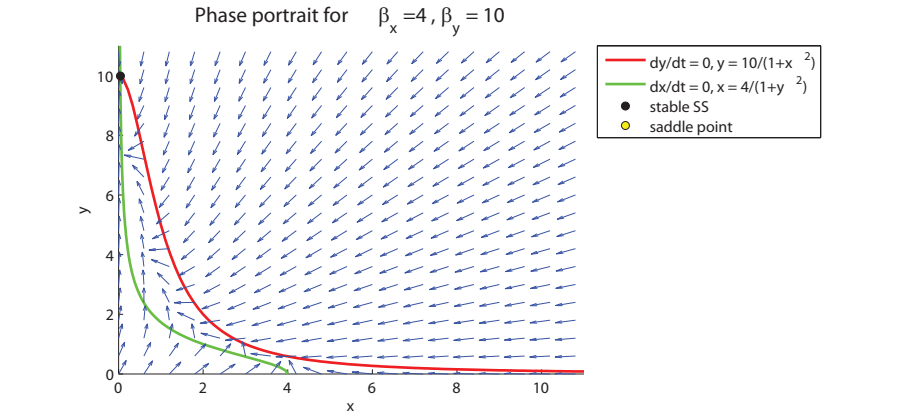

This system is preferable for two reasons: First, its stable points represent very clearly distinct transcriptional programs with X high / Y almost zero and X almost zero / Y high, respectively; whereas for a Hill coefficient of two, the self-activation switch had relatively nearby stable states. Second, this system is much more robust; each stable point drains a large basin of attraction in the phase plane, and *both* concentrations $x$ and $y$ would have to be strongly perturbed away from equilibrium to move a system from one of the stable teady states past the separatrix in the middle into the basin of attraction of the other. The self-activating TF, by constrast, can switch states following a modest perturbation of a single concentration.

*6 points: 3 for the phase portrait and 3 for (any) correct and substantiated discussion of advantages.*

h) All else remaining unchanged, what happens to the system if $\beta_y = 4$?

**Solution:**

Again plot the (new) nullclines (and here, for clarity, a full phase portrait; although that is not required for full credit):




Two of the three critical points vanish (this is called a saddle-node bifurcation), and we are left with a monostable system. Thus, the system's ability to function as a bistable switch is sensitive to maximal rates of transcription from the two promoters.

*3 points — for the realization that the system loses its bistability. This conclusion can be reached by different paths - plotting the nullclines is the easiest, but re-doing the entire problem is also legitimate.*

**33 points overall for problem 5.**

## MATLAB code for Problem 5

ps6q5.m:

```
1
2  % 20.320 Fall 2010
3  % Pset 6, problem 5
4  % Performs stability analysis for toggle and plots phase portrait
5
6  function ps6q5()
7
8  clc;
9  close all;
10
11 plotPhasePortrait(10, 10);
12 plotPhasePortrait(4, 10);
13
14 function plotPhasePortrait(betax, betay)
15
16 x = linspace (0,11,100);
17 ynull = betay ./ (1 + x.^2);
18 xnull = betax ./ (1 + x.^2);
19 [X,Y] = meshgrid(0:0.6:11);
20 DX = betax./(1+Y.^2) - X;
21 DY = betay./(1+X.^2) - Y;
22 sz = sqrt(DX.^2 + DY.^2); % The length of each arrow.
23 DXX = DX./sz;
24 DYY = DY./sz;
25
26 figure()
27 hold on;
28 hy = plot(x,ynull,'r-', 'LineWidth',2);
29 hx = plot(xnull,x,'g-', 'LineWidth',2);
30 hv = quiver(X,Y,DXX,DYY,.6);
31
32 c = [1, -betax, 2, -2*betax, (1+betay^2), -betax ]
33 ssx = roots(c)
34 ssy = betay ./ (1 + ssx.^2)
35
36 stablenodes = [];
37 saddles = [];
38 for i=1:length(ssx)
39    if imag(ssx(i)) == 0 && imag(ssy(i)) == 0
40       Asym = [-1,  2*betax*ssy(i)/((1+(ssy(i))^2)^2);
41           2*betay*ssx(i)/((1+(ssx(i))^2)^2),-1]; % [fx, fy; gx, gy]\
42       A = double(Asym);  % convert from symbolic to numeric
43       evalues = eig(A);
44       if ¬any(imag(evalues(:)))  % no imaginary eigenvalues
45           if all(evalues(:) < 0) % stable node
46               stablenodes = [stablenodes; [ssx(i), ssy(i)]];
47           elseif ¬all(evalues(:) > 0) % saddle
48               saddles = [saddles; [ssx(i), ssy(i)]];
49           end
50       end
51    end
52  end
53
54 pnodes = double(stablenodes)
```

```
55 psaddles = double(saddles)
56
57 for i=1:size(pnodes,1) % for each stable node
58   hss = plot(pnodes(i,1),pnodes(i,2),'ko', 'MarkerFaceColor', 'k');
59 end
60
61 for i=1:size(psaddles,1) % for each stable node
62   hsp = plot(psaddles(i,1),psaddles(i,2),'ko', 'MarkerFaceColor', 'y');
63 end
64
65 hsp = plot(-1,-1,'ko', 'MarkerFaceColor', 'y'); % so the handles are defined
66 hss = plot(-1,-1,'ko', 'MarkerFaceColor', 'k'); % even when there are no nodes
67
68     legend([hy hx hss hsp], ['dy/dt = 0, y = ', num2str(betay), ...
69         '/(1+x^2)'], ['dx/dt = 0, x = ' num2str(betax) '/(1+y^2)'], ...
70         'stable SS', 'saddle point', 'Position', -1);
71     title(['Phase portrait for \beta_x =' num2str(betax), ...
72         ' ,\beta_y = ' num2str(betay)] ,'FontSize', 16, ...
73         'FontWeight', 'bold');
74     xlabel ('x', 'FontSize', 12, 'FontWeight', 'bold');
75     ylabel ('y', 'FontSize', 12, 'FontWeight', 'bold');
76     set(gca,'FontSize',12, 'FontWeight', 'bold');
77     axis([0 11 0 11]);
78 hold off;
```

20.320 Analysis of Biomolecular and Cellular Systems
Fall 2012