Lecture 16 - Case Study Solution

1. Read ✓

2. (a) Class of problem. Single non-linear equation

   Use Newton's method, root, etc. I'll pick opt.root.

   (b). Standard notation:

$$\frac{Fb}{k_B T} = \frac{1}{2}\left(1 - \frac{x}{L}\right)^{-2} - \frac{1}{2} + 2\frac{x}{L}$$

let $x_0 = \frac{Fb}{k_B T}$, $x_1 = \frac{x}{L}$ ← Then these are unitless!

$$0 = -x_0 + \frac{1}{2}(1 - x_1)^{-2} - \frac{1}{2} + 2x_1$$

   (c) Convert to consistent units:

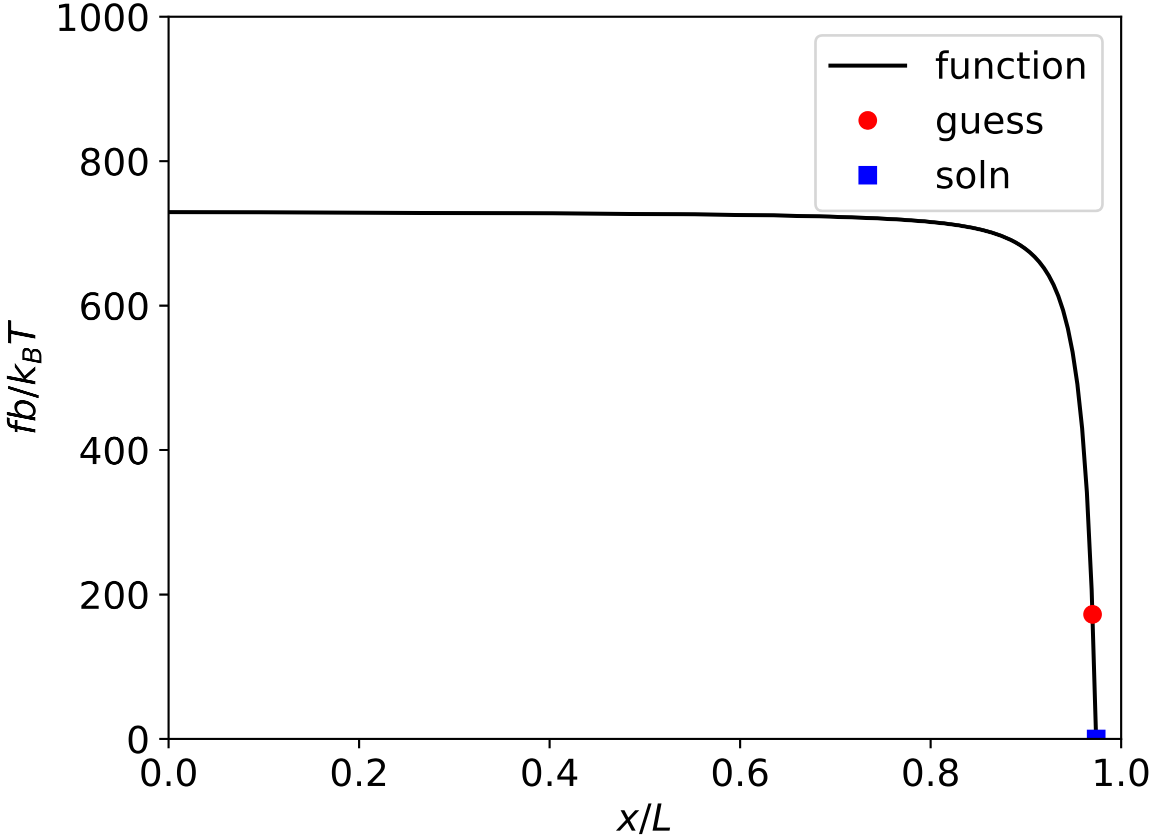
$$x_0 = \frac{Fb}{k_B T} = \frac{30 \cdot 10^{-12}\,N \cdot 100 \times 10^{-9}\,m}{1.38 \times 10^{-23}\,\frac{J}{K} \cdot 298\,K} \cdot \frac{J}{N \cdot m} = 729.5$$

$$L = 16.8\,\mu m = 1.68 \times 10^{-5}\,m$$

   (d) make a smart guess → see python plot

   (e) Solve problem: see Python file.

```python
#!/usr/bin/env python3

import numpy as np
import matplotlib.pyplot as plt
import scipy.optimize as opt

x0 = 30e-12*100e-9/1.38e-23/298 # dimensionless force
L = 16.8 # lambda DNA contour length in microns

def f(x1):
  return x0 - 0.5*(1 - x1)**(-2) + 0.5 - 2*x1

x_guess = 0.97 # use plot below to get this
x_soln = opt.root(f, x_guess).x[0]

# check solution
print('residual: ', f(x_soln))
print('fractional extension: ', x_soln)
print('max extension: ', x_soln*L, '(microns)')

# plot for getting a good guess
x1_plt = np.linspace(0, 0.999, 201) # x/L needs to be between 0 c
plt.rc('font', size=14)
plt.plot(x1_plt, f(x1_plt), 'k-', label='function')
plt.plot(x_guess, f(x_guess), 'ro', label='guess')
plt.plot(x_soln, f(x_soln), 'bs', label='soln')
plt.legend()
plt.xlabel('$x/L$')
plt.ylabel('$f b / k_{B} T$')
plt.xlim([0, 1])
plt.ylim([0, 1000])
plt.tight_layout()
#plt.show()
plt.savefig('Lec16_Supp-Case_Study_Key_Plot.pdf')
plt.close('all')
```



1000
800
600
400
200
0
0.0
0.2
0.4
0.6
0.8
1.0
fb/k_BT
x/L
function
guess
soln