09-723 Proximal probe techniques. Name_____
Homework #5.

Due by 6:30 PM, Thursday, October 7, 2004

Download the ForceCurve.zip file which contains all necessary Matlab files by clicking on it and choosing to save it on your computer (e.g. on your desktop). Open the archive on your computer and extract it to some folder. Make sure that a path is set in Matlab to this folder. To set a path, choose the set path option in the file menu of Matlab. Click the Add with subfolders button and locate the folder to be added to the path in the pop-up window and click OK. Then click Save and Close.

As usual, please contact me if you have any problems with the installation or any other problems with using the program.

justinl@andrew.cmu.edu
phone: 8-9175

ForceCurveExplorer (Some assembly required)

The ForceCurveExplorer simulates simple AFM force curve experiments using the Lennard Jones potential.

All necessary parameters are entered through the provided Graphical User Interface (GUI), which also controls the execution of the Simulink model, which you will need to construct in Simulink, with some minor adjustments to your model from Homework#4. Save the new model as **ForceCurveModel.mdl**. Do not save over your old model. Save the model in the folder that you extracted the contents of **ForceCurve.zip** into. You should have already set a path to this folder. Make sure that in your model you call the variables as follows:

Mass: `Mtip` (As promised in previous homework, this is now the mass of the tip)
Spring constant: `Kspring`
Damping Coefficient: `Bdamp`
The simulation step (Fixed-step size): `SimStep`
The total simulation time (i.e. Stop Time)= `SimTime`
The initial velocity = `Zprime` (initial condition for the first integrator)
The initial position = `Zstart` (initial condition for the second integrator and the drive offset)
The position trajectory = `Ztraj`
The time = `time`
The driving force amplitude = `Fo`
The drive frequency (operating frequency in Hz) = `Foper`
The phase = `Phi`
The potential energy = Penergy
The kinetic energy = Kenergy
The damping power = Bpower
The damping energy = Benergy
The drive power = Dpower
The drive energy = Denergy
The ramp stop distance = Zend
The ramp position = RampPos
The Hamaker constant = Atip
The tip radius = Rtip
Sigma = sigma
The Lennard Jones force = FLJ
The ramping rate = RampRate

If these names are not used, the GUI will not work.
All other variables will be defined through the GUI.
Steps in construction of your model:

I. Creating a ramp for shifting drive position.

Place two **ramp** blocks (which are found under **Sources**) in your model. We need two ramps because we want to move the tip toward the surface for half of the simulation, and then we want to retract the tip in the second half of the simulation. In **Ramp1**, set the slope to **–RampRate**, the Start Time to **0**, and the Initial Output to **Zstart**. In **Ramp2**, set the slope to **RampRate*2**, the Start Time to **SimTime/2**, and the Initial Output to **0**. Add these two **Ramps** together using an **Add** block. Use a **Gain** to multiply the signal from the **Add** block by **Kspring/Mtip**. Use another **Add** block to add this to the **drive signal**. The signal from this new **Add** block then replaces the drive signal in the damped driven harmonic oscillator loop. Send the **RampPos** to the workspace using a **To Workspace** block. The **RampPos** should be taken from the signal from the **Add** block that added the two **Ramps** together, and before the **Gain**.

II. Adding the Lennard Jones Force to the model

First, add another signal to the **Add** block of the damped driven harmonic oscillator loop by placing a **+** in the List of Signs. Now, place a **Fcn** block (found under User-Defined Functions) in your model. Use the z signal as the input to this **Fcn** block. In the Expression box of the **Fcn** block, type exactly:

(Atip*Rtip)*((pow(sigma,6)-(30*pow(u,6)))/(180*pow(u,8)))

Use a **Gain** to multiply the signal from the **Fcn** block by **1/Mtip**. Send the signal from the **Gain** to the new input you create in the **Add** block of the damped driven harmonic oscillator loop. Also, send the signal from this **Gain** to the workspace as **FLJ** using a **To Workspace** block.

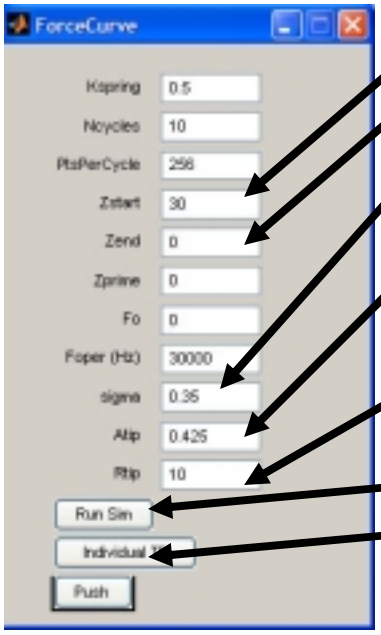
III. Use a Variable Step

In the **Simulation** menu, select **Configuration Parameters. . .** and set **Type**: to **Variable-Step** to and **Solver**: to **ode23 (Bogacki-Shampine).**

Before running the model you need to provide the values of other parameters using the provided GUI. To use the GUI provided, run `ForceCurve` from command line.

```
>> ForeCurve
```

The following interface should appear:



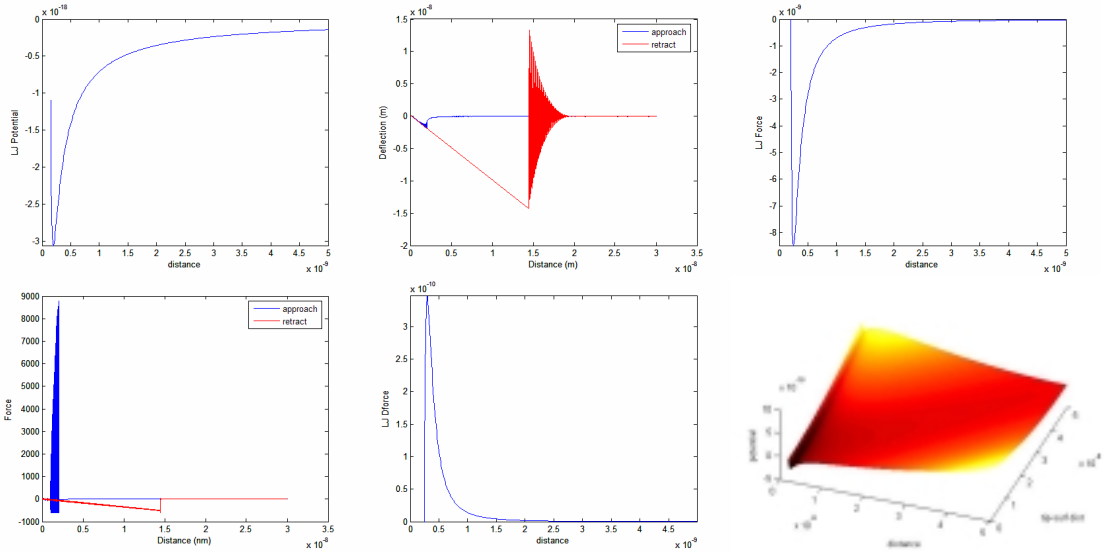
Zstart – defines the starting and ending distance from the surface
Zend – define force curve depth
sigma – define sigma (in nm)

Atip – define Hamaker constant (in J*10^-19)

Rtip – define tip radius (in nm)

Run Sim – run the sim
Individual TP – pull a single total potential curve at a specified tip-sample distance.

Click the **Run Sim** button and the following plots should appear (this may take a few minutes as there are several calculations being performed):



1. Click on the **Run Sim** button and copy the resulting deflection force curve into your report document. Identify all characteristic regions of the force curve and make sketches illustrating the position of a cantilever with respect to the sample and cantilever shape (undeformed, bent up, bent down, etc.).

2. By running several force curves with different values of Hamaker constant A_{tip} radius R_{tip} and reading out from the plot the values of the snap-out force F_{out}, determine the relationship

   $A \times R_{tip}$ vs. F_{out} .

   Justify this relationship based on the equation describing the interaction of a sphere with the surface. Assume that $\sigma = 0.35$ nm.

3. In this problem you are asked to provide the illustration of the energetic reasons of snap-in and snap-out behavior. Do this by generating and pasting into your report a series of plots of total energy as a function of cantilever base distance from the surface, P at some characteristic distances (far away, when the total potential is dominated by the elastic energy, then closer to the snap-in point, at the snap point, etc. Show what happens upon approach to the surface (snap-in) and upon moving away from the surface (snap-out). You do not need to generate more than 5 plots for approach and 5 plots for retraction. Describe in your own words the total energy balance at each distance you have chosen to show. To generate these plots you will have to click the **Individual TP** button and provide some separation distance. Due to resolution and array sizing issues, total energy plots are limited to separation distances ranging from 0.15 nm to 5 nm. So in order to answer this question use the following values:
   Kspring = 2.0, **Ncycles =** 10, **PtsPerCycle** = 256, **Zstart** = 10, **Zend** = 0, **Zprime** = 0, **Fo** = 0, **Foper** = 30000, **sigma** = 0.35, **Atip** = 0.425, and **Rtip** = 10.