# 1.021, 3.021, 10.333, 22.00 Introduction to Modeling and Simulation - Spring 2018

**Part I - Continuum and particle methods: Problem set #2**

**Instructor:** Markus Buehler (1-235A&B, mbuehler@MIT.EDU)

**TA:** Francisco Martin-Martinez (fmartinm@mit.edu)

**Due: March 6, 2018**

---

**Important:** Specify all resources you use for your solution (lecture note, book, and web).

The following set of exercises is designed to train you in the basics of atomistic and molecular simulation, focusing on the development and application of interatomic potentials and Monte Carlo simulation. For each exercise, show us how you came to your answer and result. We highly encourage you to make drawings where appropriate. Also, please follow the sections in the Pset when providing the answers. **Note to carefully select the parameters before submitting your job to nanoHUB; the default parameters may not have been adapted to the case you are studying.**

---

## 1. **Lennard-Jones potential development**

Pair potentials are one of the simplest potentials used to describe metals. Here we develop a 12:6 Lennard-Jones (LJ) pair potential for face-centered cubic (FCC) copper based on structural and elastic properties of copper obtained from quantum mechanical calculations.

(a) Assuming a 12:6 Lennard-Jones potential (equations see lecture notes), derive an expression for the equilibrium position $r_0$ between pairs of atoms. This position corresponds to the nearest neighbor distance. Express the LJ length parameter $\sigma$ as a function of the lattice parameter $a_0$. Consider only nearest neighbor interactions between atoms.

(b) Identify the second derivative of the potential, $\varphi'' = k$, expressed at the equilibrium spacing as the spring constant, resulting in an expression of $k$ as a function of the potential parameters $\sigma$ and $\varepsilon$.

(c) Using the equations listed below, determine the parameters of the Lennard-Jones (LJ) potential based on the values for the lattice constant and bulk elastic properties of copper obtained from quantum mechanical calculations reported by Ogata, Li and Yip (data provided in the table below). **Express potential parameters in units of eV (for energy) and Angstrom (for length).**

To solve this problem, write an expression for the bulk modulus $K$ as a function of potential parameters, then determine the unknowns. Consider only nearest neighbor interactions.

**Note**: See additional equations below, which you can use to develop a relation between the Young's modulus and the potential parameters $\sigma$ and $\varepsilon$.).

(d) Compare your potential parameter values with the LJ copper potential reported by Cleri and coworkers (*Phys. Rev. Lett.*, 1997; paper posted on Stellar and reference included in lecture notes). Discuss possible disagreement in light of the potential formulation and the potential cutoff.

(e) Now we use the LJ potential developed above and carry out an MD simulation to determine elastic properties.

Using the web based code (nanoHUB module **stretchfcc**), calculate the bulk modulus of copper by plotting the stress tensor coefficients $\sigma_{11}$, $\sigma_{22}$ and $\sigma_{33}$. This module enables you to apply deformation (strain) to a crystal in periodic boundary conditions and measure the stress tensor under increasing deformation.

For the simulation analysis, use:

(i) the LJ potential with the parameters developed above (**note:** Consider only nearest neighbor interactions by choosing a proper cutoff radius, maybe 10 to 20% larger than the nearest neighbor distance), and

(ii) Cleri *et al*.'s potential.

Consider equitriaxial strain loading, that is, the same strain applied to all three directions of the unit cell.

For both cases make sure to convert units properly and express the computed bulk modulus in GPa.

Deformation is measured in strain $\varepsilon_{11} = \Delta u_x / L_x$,

$\varepsilon_2 = \Delta u_y / L_y$,

$\varepsilon_{11} = \Delta u_x / L_x$ ;

the relationship between strain and stress is the bulk modulus,

$(\sigma_{11} + \sigma_{22} + \sigma_{33})/3 = K(\varepsilon_{11} + \varepsilon_{22} + \varepsilon_{33})$

(Stress and strain are all the same in all three directions)

**(f)** For both potentials, calculate the critical strain when the slope of the stress-strain plot reaches zero (characterizing the point when the crystal becomes unstable).

**Additional equations**

Bulk modulus: $K = E/(3(1-2\nu))$

Young's modulus $E = 8/3\mu$

Shear modulus $\mu = r_0^2 k/2/V$ (nearest neighbor interactions)

Volume of unit cell $V = a_0^3/4$

Poisson's ratio assumed to be $\nu = 1/4$

Spring constant of potential defined by $k = \dfrac{\partial^2 \phi(r)}{\partial r^2} = \phi''$

**Results from quantum mechanical Density Functional Theory (DFT) calculations***

| | $a_0$ [Å] | $K$ [GPa] |
|---|---|---|
| Cu (DFT calculation) | 3.64 | 140.0 |

* Source: S. Ogata, J. Li and S. Yip, *Physical Review B,* Vol. 71, paper # 224102, 200

## 2. **Monte Carlo simulation of cell spreading**

The purpose of this problem set is to develop a simple Monte Carlo model to describe the migration or spreading of cells. Spreading of cells is an important mechanism in the development and maintenance of multicellular organisms, and plays a central role in embryonic development and wound healing. Cell spreading happens in a mechanism by which cells form so-called extensions. The details of this mechanism is not well understood, albeit recent work has shed more light onto this issues. Here, we assume a simplistic view of the process and think about ways to model this complex biological phenomenon.

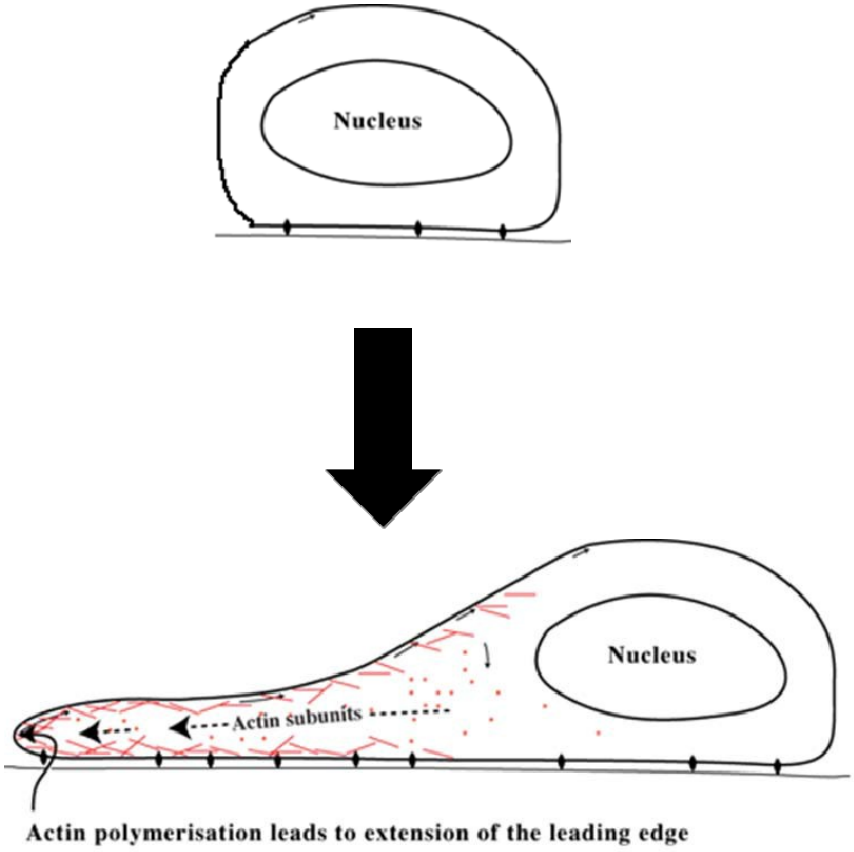


**Figure 1**: Mechanism of the formation of extensions (side view), which allow cells to put out extensions, or "feelers", to find new attachment sites. If unsuccessful, new extensions are put out in a different direction. Source: http://en.wikipedia.org/wiki/Cell_migration

In a paper (Bucaro, Aizenberg, *et al., ACS Nano*, 2012; paper posted with this problem set), a group of researchers showed that the geometry of cell spreading correlates with the structure of the substrate (see **Figure 2**). The substrate used in their study is composed of arrays of nanopillars (NPs), whose density can be controlled. They deposited stem cells on top of the nanopillars and observed that the density of nanopillars, controlled by the interpillar distance, can be used to tune the spreading behavior of the cells. They proposed that the substrate geometry is capable of guiding the growth and differentiation of stem cells.

You are asked to develop a model of this process using a Monte Carlo approach, and write a code to simulate the process that is able to ***qualitatively*** describe the cell spreading. The model should describe how the cell starts from a single point and spreads out to reach more peripheral NPs, while capturing

the characteristics found in Aizenberg's experiment, *i.e.*, how the NP distance leads to the different cell spreading behavior.

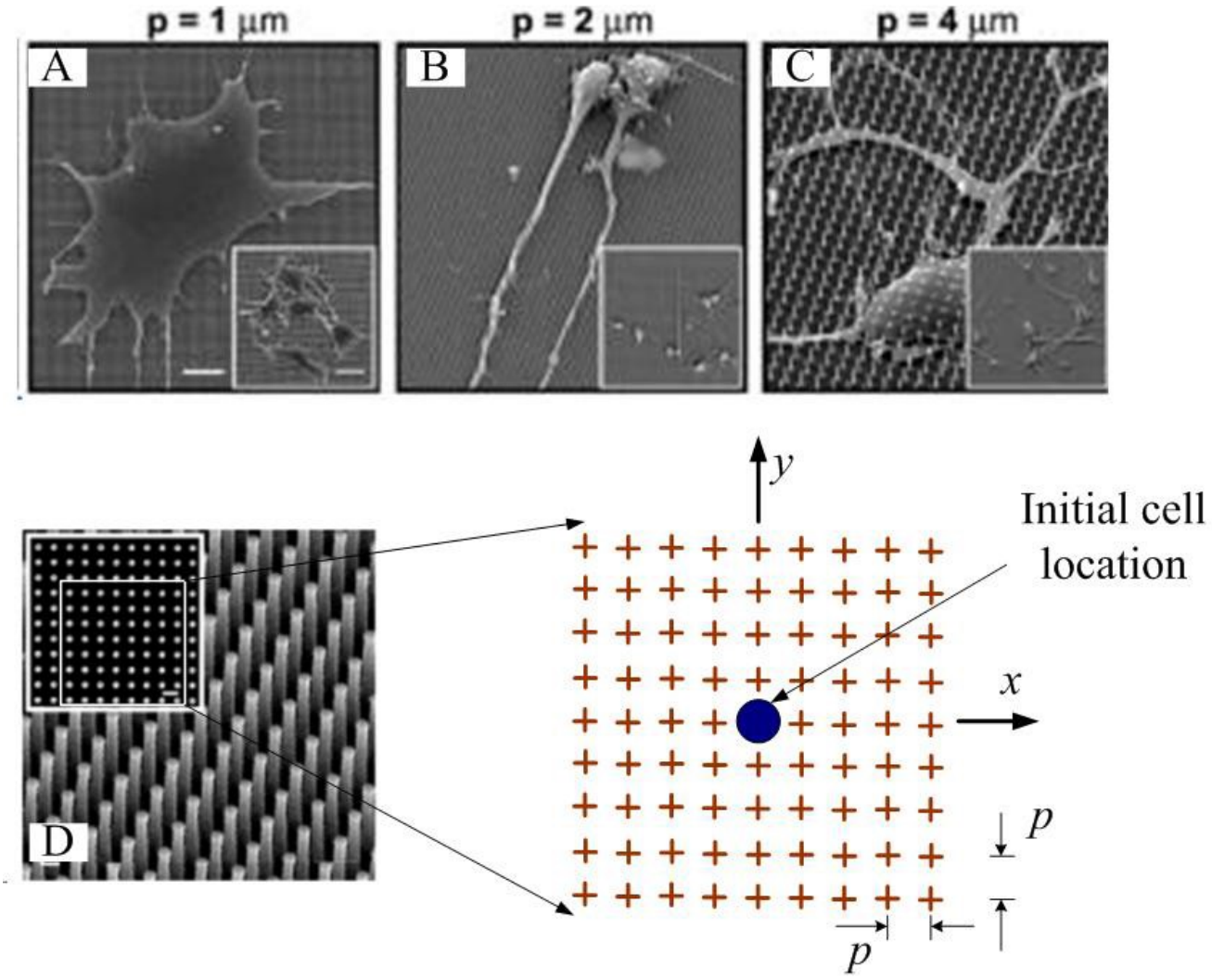


**Figure 2**. Mouse embryo-derived stem cells grown for 1 day on nanopillar (NP) arrays with a spectrum of NP spacings. Representative SEM images of cells on arrays of 1 μm (A), 2 μm (B), and 4 μm interpillar distance (C) show different stem cell morphologies as a function of NP density. Insets are lower magnification images. (A) Cells grown on regions with p = 1 μm spread similarly to cells on polished silicon. (B) A dramatic change to neuron-like morphologies occurs at p = 2 μm. The majority of cells develop a single extension that spreads across multiple NPs. (C) Cells on NPs of p=4μm extend past the NPs and spread at the bottom of the surface and develop multiple, highly branched extensions. Scale bars (A, B, C) 10 μm, insets 20 μm. (D) Picture and schematic of the NP substrate, as suggested for the model developed here. Here we consider each NP as a single point in-plane with fixed (*x, y*) coordinates. A cell initially rests on one single NP, as illustrated.

To solve this problem, use Matlab or any other programming language. Please submit your code along with your solution to the problem (submit the code in the native programming language, such as a .m file, and indicate which language you used so we can run it). Use commenting to explain the steps taken in the code, variables named, etc.

**Hint:** To make the problem easier to model and to ensure that the model captures the essential mechanism, several assumptions are to be included in the model, including:

- The substrate is modeled as 2-D plan with nanopillars locates at (*n*p, m*p*), where *p* is the interpillar distance, and *n* and *m* are integers;
- For each step in the Monte Carlo algorithm, randomly pick a single point occupied by the cell (use an data structure or array to keep track of all the occupied points) from which it makes spreading attempts;
- Only consider 4 possible directions with same likelihood for the cell to spread toward (*x+, y+, x-, y-*);
- The cell spreading cannot occur via 'jumps' across very large distances, *i.e.*, the point can only spread by a single lattice length in each Monte Carlo step, and if the targeting point is occupied, it pushes the occupied point one lattice length forward (repeat if the new target is occupied again).

Use the following questions to guide you through the solution of the problem:

1) Purely based on **Figure 2** and any information you can gather from reading the *ACS Nano* paper, use your own language to concisely summarize how cells behaves differently on different substrates, and why. Provide clear reasoning.
2) Considering the insights developed from the experiment, develop an algorithm to simulate the spreading of cells for different pillar spacings. Explain the rationale behind your algorithm. Think carefully about variables used, and clearly explain their physical meanings. If you do not have specific numerical values (very likely!), assume appropriate ones.
3) Develop a pseudocode to sketch how you will program the algorithm and write it out, with comments. This step is optional, you can also go straight to point 4) and write the code.
4) Develop a code in a programming language that carries out the simulation. Show the results of simulations for different spacings and relate it to the experimental findings. Include a detailed presentation of the results including assumptions made, and how the data is analyzed.
5) Discuss your results critically (how can the algorithm be improved, what kind of predictions could be made, are their behaviors you can identify in the model that could be tested experimentally, etc.). Present your result (snapshots obtained using your code) to demonstrate how it is able to reproduce experiment observations; include a side-by-side comparison (**Figure 2**).

**Notes regarding nanoHUB website and Molecular Dynamics codes**

You will need the **stretchfcc** module for this problem set (https://nanohub.org/tools/stretchfcc).

**Visualization:** As for pset #1 you may use the program Visual Molecular Dynamics or "VMD" (download at http://www.ks.uiuc.edu/Research/vmd/) to visualize your results. You can plot graphs and other data also using Matlab, which may be most appropriate for Problem 1).