

Method description for applying a spacially varying force on surface atoms

Kristian Kuppert

April 2, 2015

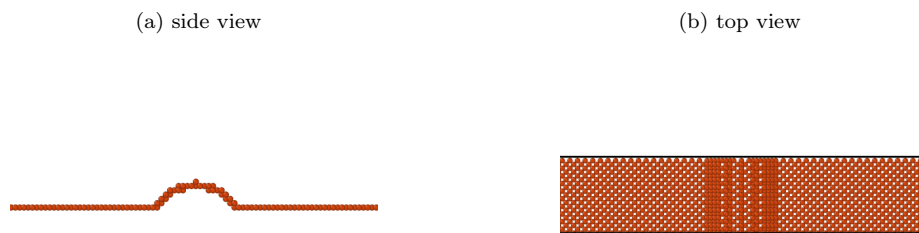
Introduction

Here I describe the process how I assign the forces on my surface atoms.

1 Creating an atomistic surface from bulk data

First I have to define what atoms of my system belong to the surface. For that, I use coordination analysis. This parameter basically counts the number of atoms that surround a particular atom. As it can change due to thermal fluctuations, as large sphere as possible should be taken here. The maximal sphere size is the cut-off of the potential, so I use that. Second parameter that has to be chosen, is the coordination number under which, I consider my atom to belong to the surface. The selection criteria for this number is that first, all of the surface should be covered, e.g there should not be any holes in the surface. Second one is, that it should be as small as possible, so to choose only the topmost atoms. The best way to make sure this parameter is OK, is by visual examination of the resulting atomistic surface. As it is possible to dynamically update the surface during the simulation, I do not have to worry that the whole surface would not be covered when the structure changes.

Figure 1: Atomic surface of a test case with cylindrical bump



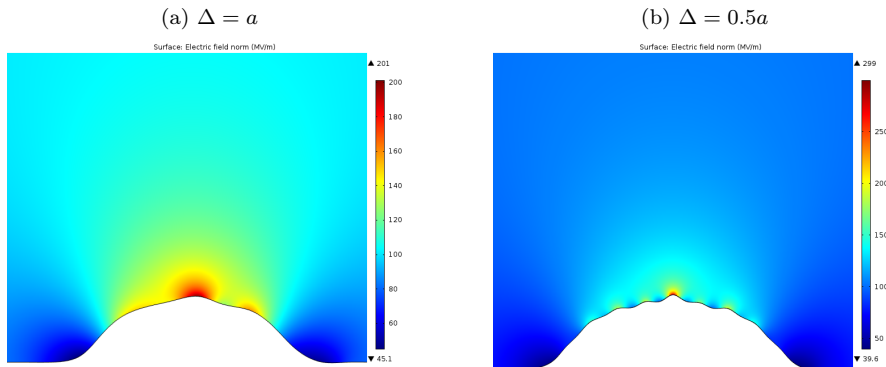
2 Creating a smooth surface for FEM calculations

As the surface is quite random, it is not possible to simply interpolate a smooth line through the coordinates of the surface atoms and some smoothing is necessary. The method I use is as follows: I divide the whole length of the surface into slices with thickness d . For each slice, I average the coordinates of the atoms in that slice and create an interpolation point with the coordinates:

$$\vec{x}_n = \frac{\sum_i^N \vec{x}_i}{N}$$

Where \vec{x}_n is the position vector of the point corresponding to the n -th slice with N atoms. Smoothing is necessary also to make the surface more physical electrostatics wise, for it to resemble the actual isopotential surface which would be created by the induced charges by the external electric field. The problem of how to do this physically correctly is in itself a very complex task[1] and it depends on the material, crystal orientation, applied field, etc. Nevertheless, for my application this method seems to suffice. The free parameter in this step, Δ is what needs to be optimized. Right now it seems that Δ of about 1 lattice constant is optimal. This also corresponds best with the results from HELMOD code.

Figure 2: resulting FEM surfaces corresponding to different Δ



3 Electric field calculations with FEM

The smooth surface can be extrapolated through the points obtained by the method described in the last paragraph. Piecewise-cubic splines are used to interpolate a smooth mathematical surface through the interpolation points. With the surface in place, the electric field calculations are quite straightforward. The boundary conditions are put in place so, that my created surface is an

isopotential (potential value of 0) and the electric field on the top boundary of the simulation domain is specified to be uniform and perpendicular to the boundary. From FEM I get two necessary variables - the electrostatic stress at each point on the surface and the normal vector components at each point.

4 Applying forces on the surface atoms

Let us view a surface element corresponding to the slice with width d , which for simplicity's sake has been taken to be the same as Δ from paragraph 2, where there is defined outwards electrostatic stress $p = \frac{\epsilon_0 E^2}{2}$ and normal vector components. Let α be the angle between \vec{n}_y and \vec{n} . Then the total force on this surface element is

$$\Delta F = \Delta p S$$

where S is the area of the surface element and Δp is the average stress on the surface. The components of the force:

$$\Delta F_y = \Delta F \cos \alpha = \Delta p S \cos \alpha = \left(\frac{1}{d} \int_{x_0}^{x_0+d} p \, dx \right) \frac{zd}{\cos \alpha} \cos \alpha = \left(\frac{1}{d} \int_{x_0}^{x_0+d} p \, dx \right) zd$$

$$\Delta F_x = \Delta p S \sin \alpha = \left(\frac{1}{d} \int_{x_0}^{x_0+d} p \, dx \right) \frac{zd}{\cos \alpha} \sin \alpha = \left(\frac{1}{d} \int_{x_0}^{x_0+d} p \, dx \right) zd \frac{n_x}{n_y}$$

$$\text{where } n_x = \int_{x_0}^{x_0+d} n_x \, dx \text{ and } n_y = \int_{x_0}^{x_0+d} n_y \, dx$$

So for each surface element I have to average the stress over that element. Let it be noted, that because stress depends on the electric field as $p \sim E^2$, one should not average the electric field and then calculate the stress from this average field. To get the force on each atom in a particular slice, we simply divide the total force with the number of atoms corresponding to that slice:

$$f_a = \frac{\Delta \vec{F}}{N}$$

I have implemented the above scheme like so: First I export the stress and normal vector data from

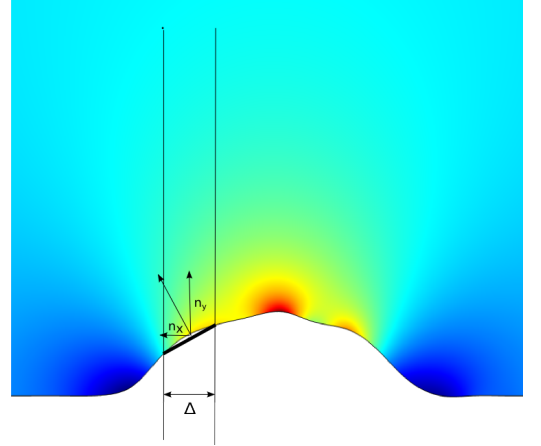


Figure 3: surface element corresponding to the slice with width d

COMSOL into text files. Then I use a python script to average the stress and normal vector components and get the forces corresponding to each slice. The resulting data file with the force components is read into LAMMPS, where simulation domain is divided into N regions, $N = \frac{L}{d}$ where L is the extent of atoms and d is the smoothing window size. At each timestep, lammmps counts the number of atoms in each region, and divides the force on that slice with the number of atoms.

References

- [1] A. Kiejna. *Metal surface electron physics*. Elsevier Science Ltd. ; Elsevier Science Inc, Kidlington, Oxford : Tarrytown, New York, 1st ed edition, 1996.