DSC Correspondence

Email to myself: “Code for the hexagonal tiling”

[https://www.mathworks.com/matlabcentral/answers/136902-how-to-create-a-hexagonal-meshgrid](https://www.mathworks.com/matlabcentral/answers/136902-how-to-create-a-hexagonal-meshgrid" \t "_blank)

Most likely, my internal coloring is going to have to be custom code. I might even want to remove the Cartesian axes and (overload?) a plot method that takes primitive lattice vectors as arguments. Then figure out the range we are trying to plot and convert against an internal colormap. That's really the right way to do this. Also, could possibly add an interpolation method. However, the place where interpolation is really key is on the boundary in the real space image, but even thought the lattice is hexagonal, the pixels are not, rather being Cartesian, so I should use the standard plotting functions there.

Make this a class. Then we can have a stored figure handle, easy data storage and manipulation, etc. Much more stable I think.

Email to Maddie: “Useful Paper and Idea”

Hi Maddie,

Thought you might want to see this one if you haven't already. The authors come up with some nice mathematical ways of describing the various components of a Moiré superlattice in twisted bilayer graphene, which I think will be useful for some of our studies. What I really liked was the idea of the DSC lattice as a way to describe pseudostacking (i.e., as a way of mathematically demarcating the "AA", "AB", and "saddle point" regions on the TBLG Moiré). The DSC lattice is generated for each atom in (say) the top lattice by asking which atom in the bottom lattice is closest to it. That makes a vector, so you get a vector field describing the orientation of the two graphene lattices (Figure 3c). You can then demarcate regions in DSC space that are within a certain distance of the DSC vector for ideal AA, AB, or SP stacking, and that allows you to assign the size of "psuedostacking" regions in the TBLG lattice (Figure 3d). (There's a nuance here in that the atoms must be in the same sublattice in both of the lattices when computing the DSC.)

Maybe you know all this already, but the whole thing about the DSC lattice gave me a new idea as follows. It seems probable to me that, for a large enough convergence angle, the diffraction pattern obtained from our 4DSTEM experimental data is very similar to what it would be if the entire region were structurally homogeneous, possessing only a single value of the DSC across the entirety of the beam. Via Colin's earlier argumentation, it is really the positioning on the DSC lattice that determines whether we are 'blinking' or not (since the DSC lattice in this paper is just a mathematical tool for describing whether and how the lattices are out of phase with each other).

Thus it may be possible to run simulations using Colin's code for all possible DSC values, where we don't actually build a Moiré for the simulation, but just make a structurally-homogenous lattice. Then, we will be able to say what blinking looks like at every point in DSC space. Then, under the assumption that the probe size is small, we can fit the blinking observed in the experiments to the blinking predicted by each DSC psuedostacking region.

The result would be a reconstruction of the DSC lattice of the sample, which would tell us where all of the AA, AB, and SP stacking regions are without the need to assume a Gaussian rotation model or anything like that. This is basically the same as the SAED images in the Yoo paper with the Moiré triangles, I believe, but rather than just probing antisymmetry in one direction, we can probe it in all directions at once because we have the fully separated diffraction pattern.

Let me know what you think. I'll be pretty busy for the next couple days because I have to pack and clean my apartment, but I'd like to try this at some point, and I think I can do it independently from Colin.

Thanks,

Nathanael

(Here the attached paper was the Scientific Reports one entitled “Interfacial Atomic Structure of Twisted Few Layer Graphene.”)