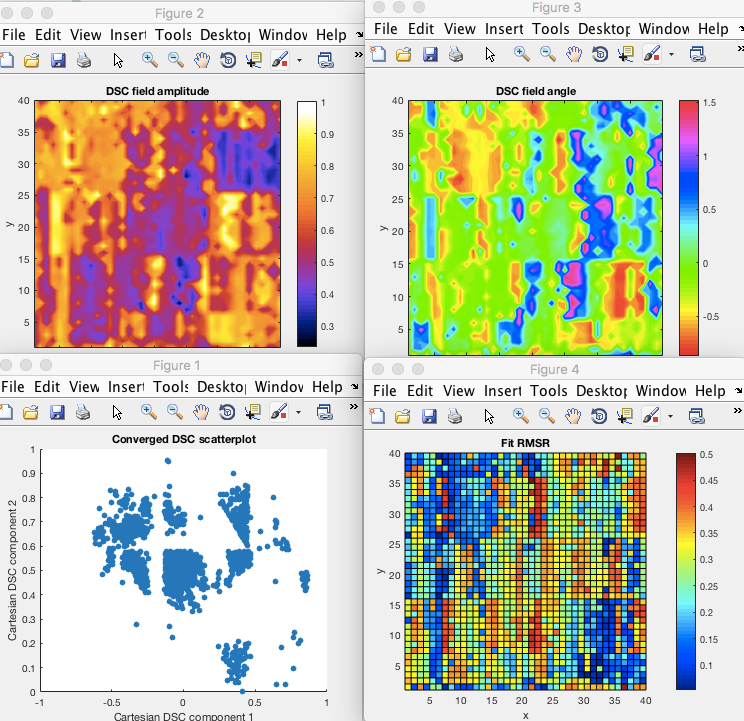
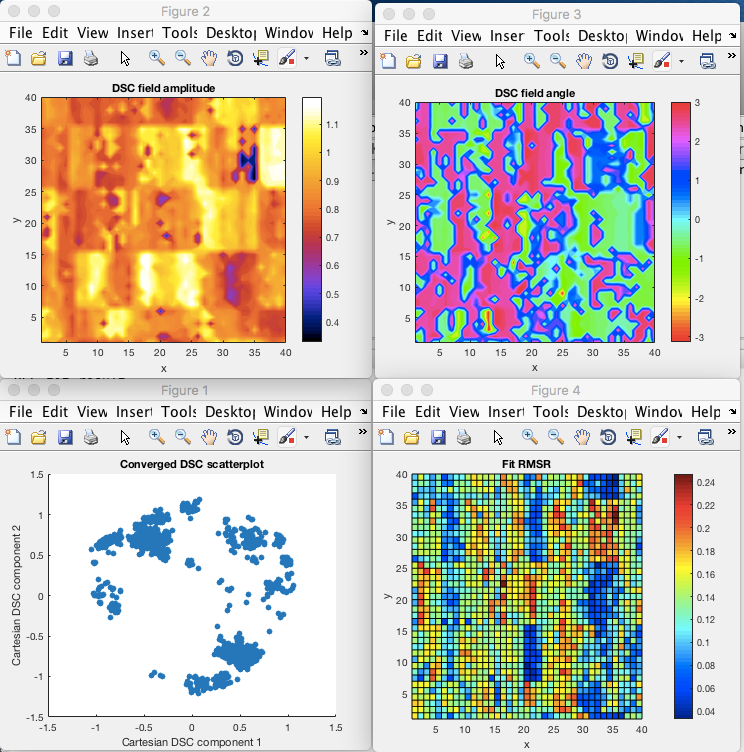
12232019 Notes on DSC fit

This is my first attempt (using zeroed, normalized disks and fitting function, eliminating disks 3 and 6 from the fitting function’s residuals).

I don’t believe this has sufficiently converged. Note how the only visible pattern in the DP is followed with a large residual band that is inordinately high for the data scaling.

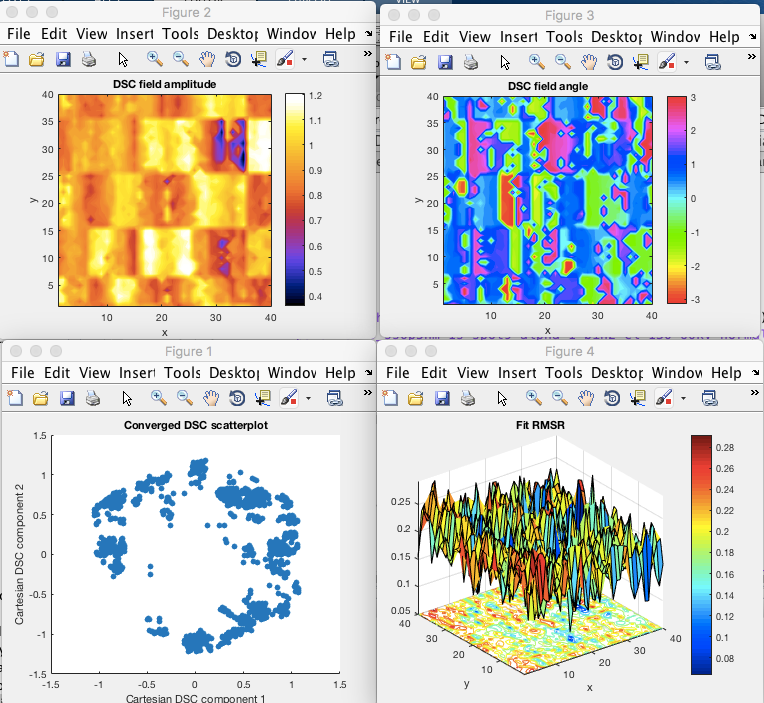


Trying again with a ninefold multistart on the lsqnonlin:  


This is looking (somewhat) better as far as the accuracy of multistart convergence, but it is not good for actually demonstrating a hexagonal mapping to the data. It looks as though the data is dominated by artifacts; witness the chopping up of the DSC convergence locations and the gridlike nature of the DSC field (and the RMSR, for that matter).

The weighting vector allows all but the third and sixth inner disks here.

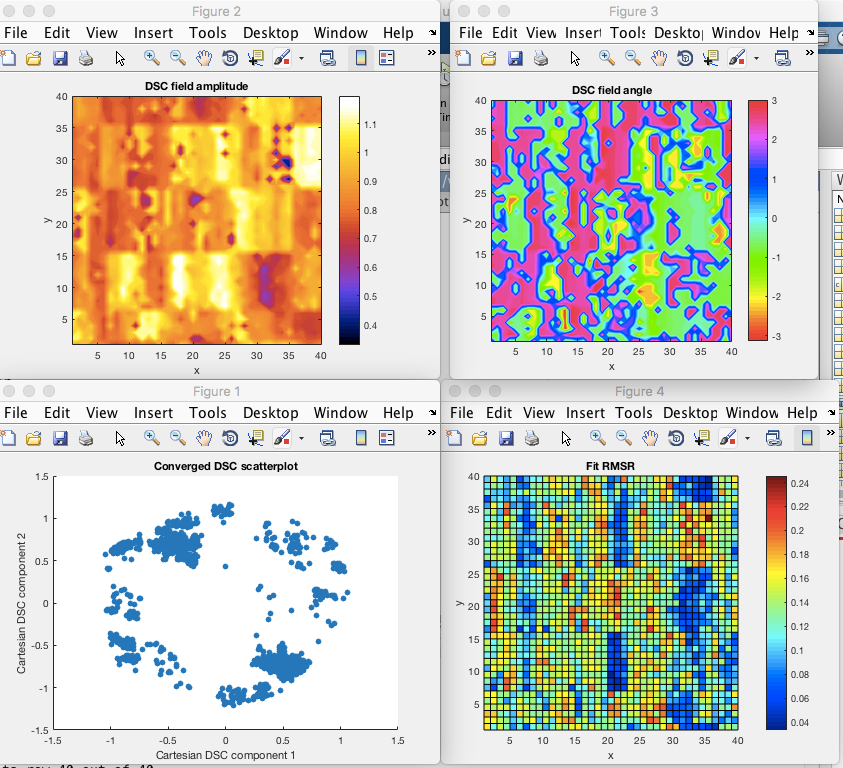
Now, let’s try it with all 12 of the disks active.



(This was still with a ninefold multistart.)

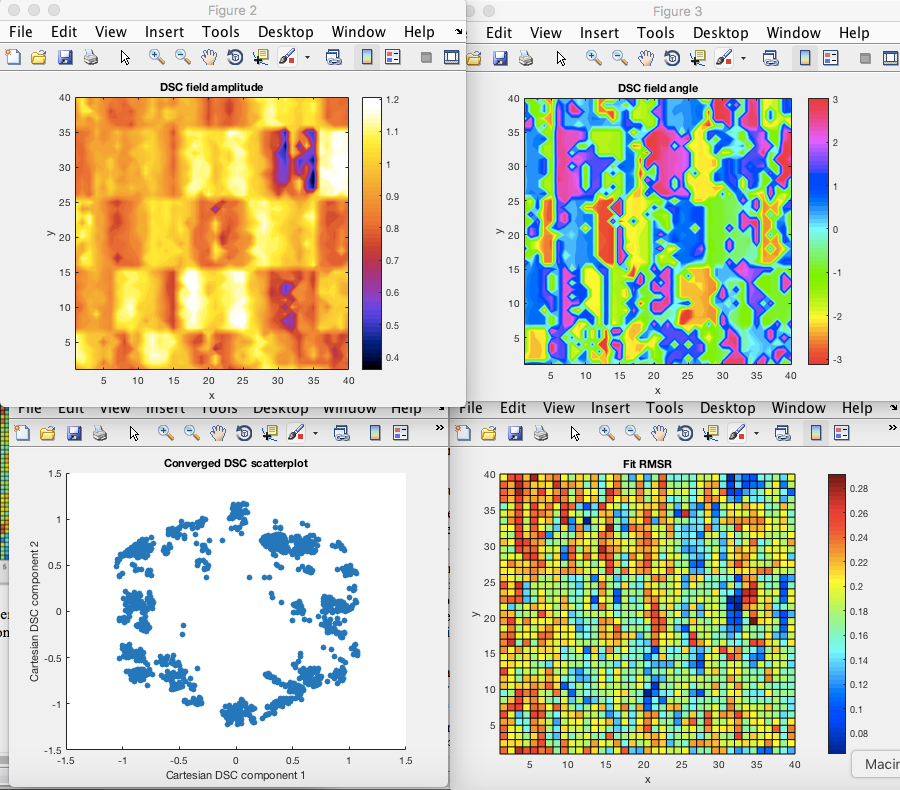
DISCOVERED THESE WERE SCREWED UP BECAUSE THE -/- QUADRANT GUESS WAS NOT EMPLOYED.

Try again with the all but 3 and 6 scheme.

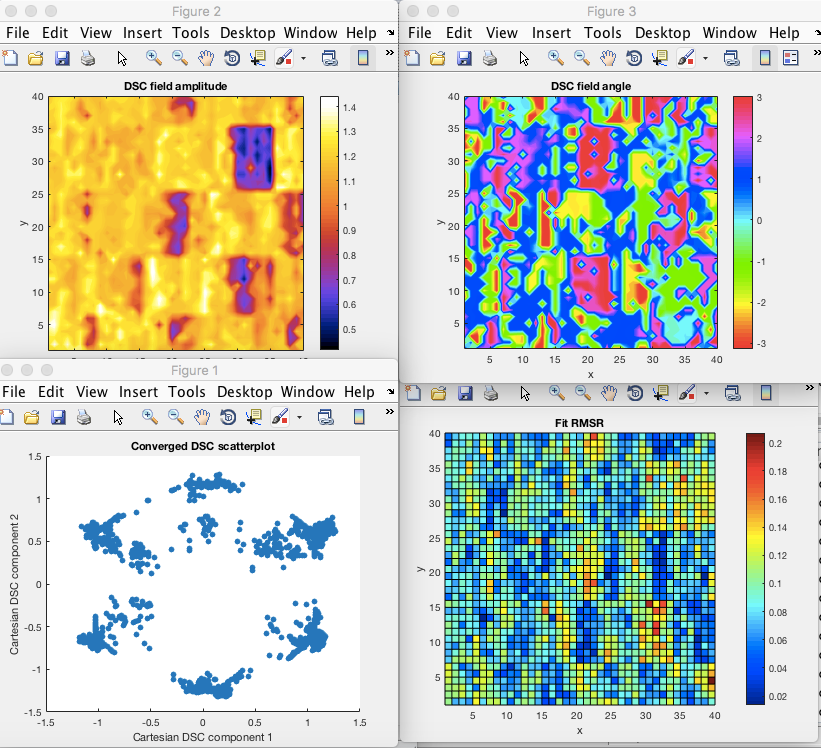


Still the DSC is only converging around the edges, even though there is a multistart planted near the middle. Is this significant for the reconstructed nature of the material? On an optimistic interpretation, this is because there are lots of AB and SP stacked regions present in the material.

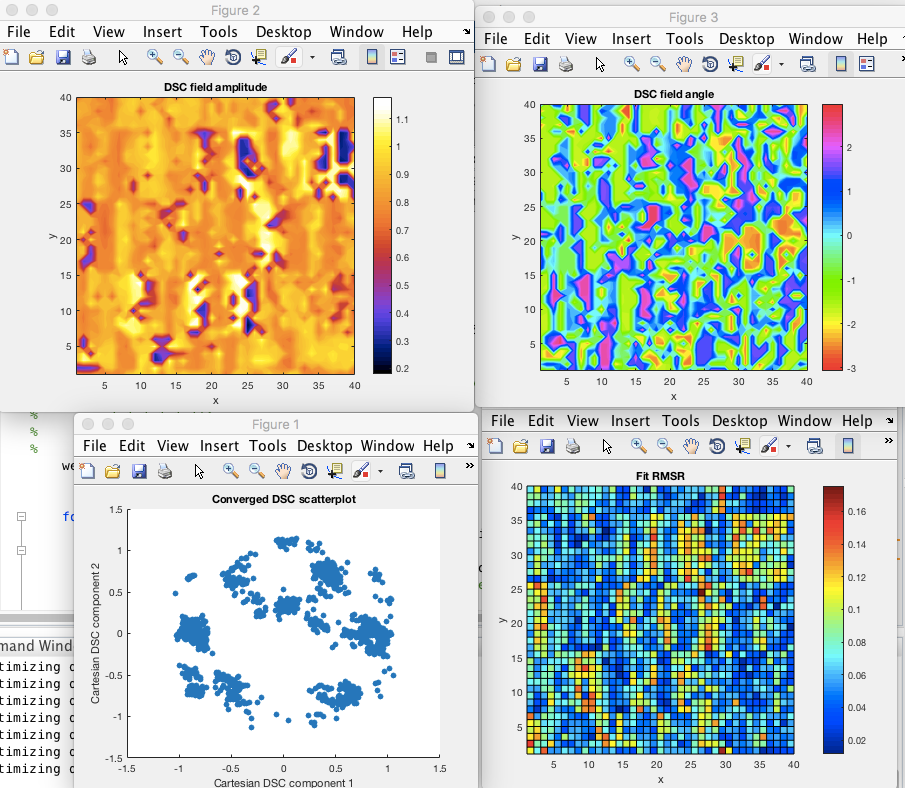
The following, then is with all of the disks allowed, including 3 and 6 in the inner ring.



The following is with only the inner ring of disks allowed:

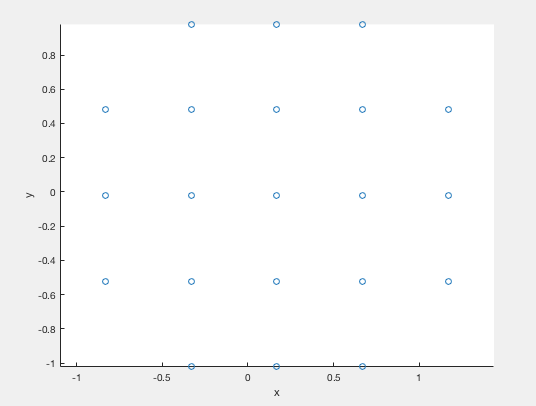


The following is with only the outer ring of disks allowed.



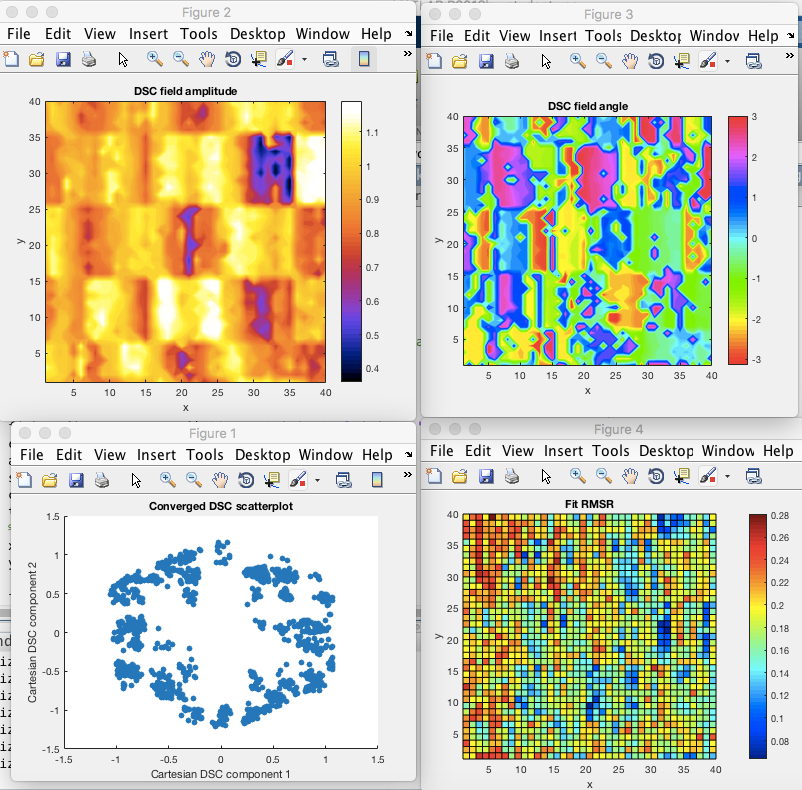
Huh, that’s a little different. Maybe those weird diagonals have something to do with it.

Try the multistart with hexagonally rastered starting points.

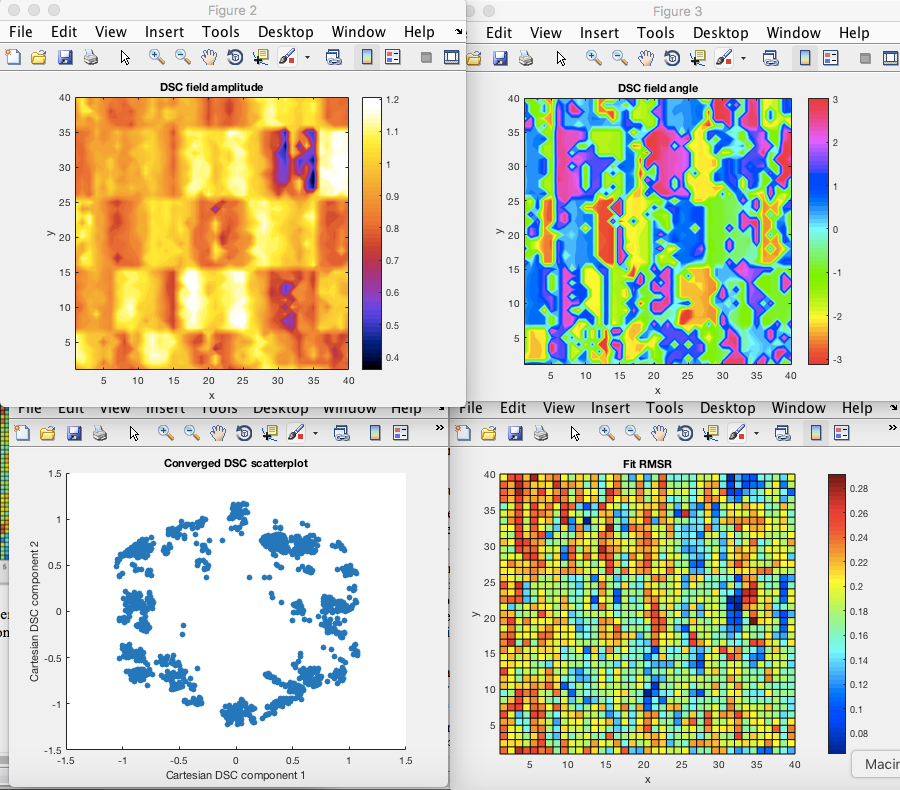
density 0.5: 

(Notice that this is not a symmetric covering with respect to the x-axis, which I had not noticed before.)

Results:



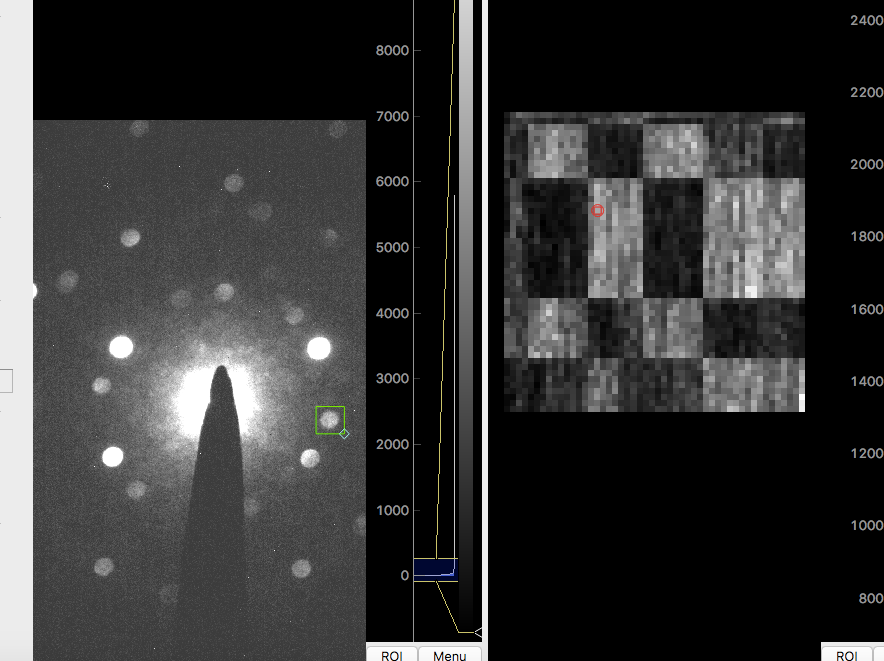
For comparison, what we had before:

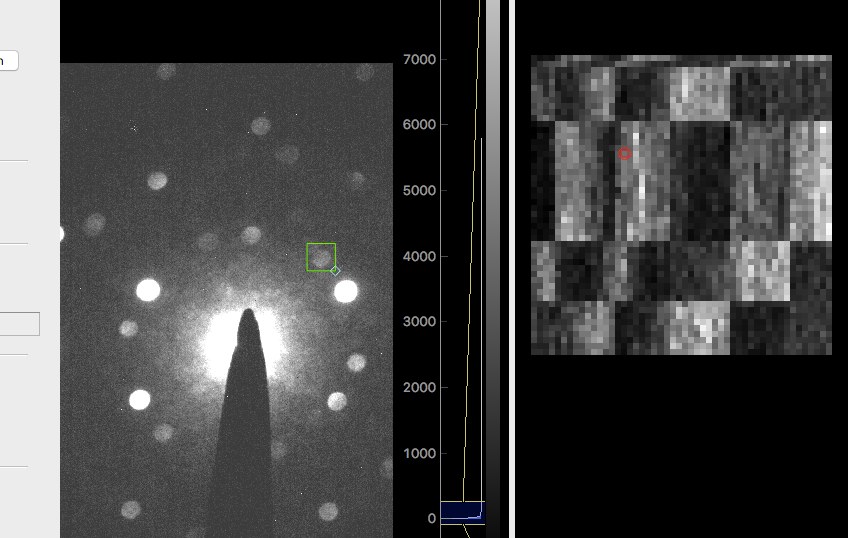


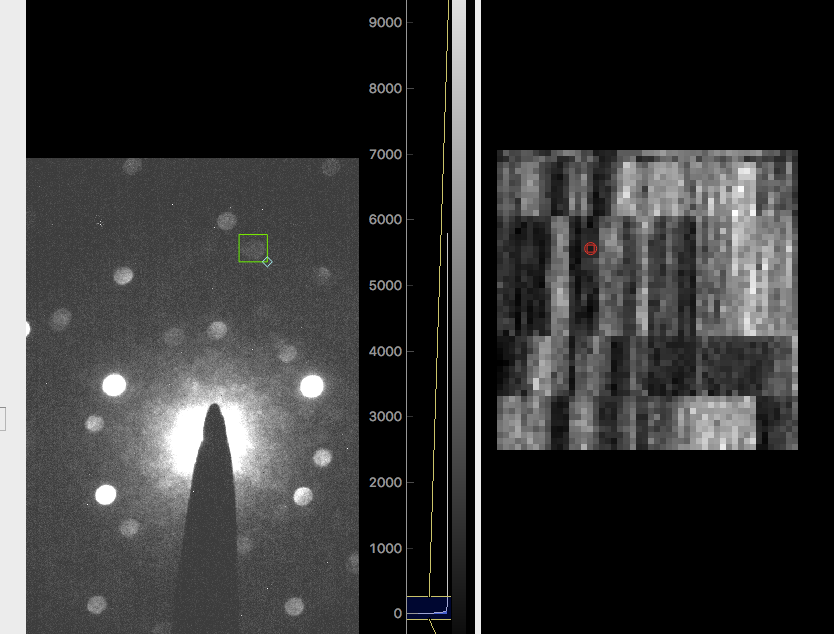
Note that these are very similar, though the addition of a few more middle optimization algorithm starting points thickens up some of the intermediate DSC vectors calculated. But certainly we never have anything approaching AA. For our hexagonal raster simulations, at AA, we are close to 90% on all of the disks. Note we are not at 100%. This may be a fault of the way that we did the raster, which didn’t place (I don’t think!) a query point right at [0,0]. But look into that.

Okay, so let’s consider moving on to dataset17. Will that work?

Preliminary virtual detector analysis suggests probably not. Here are a sampling of the images:



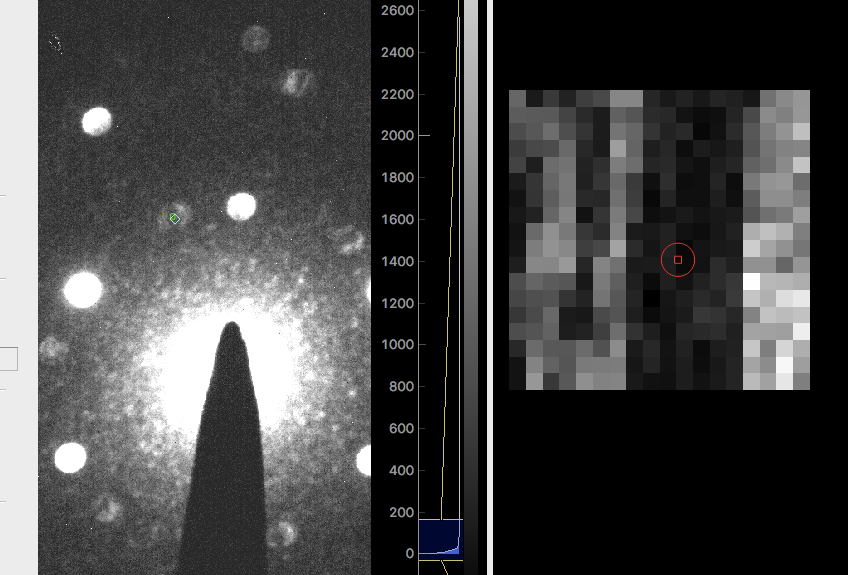


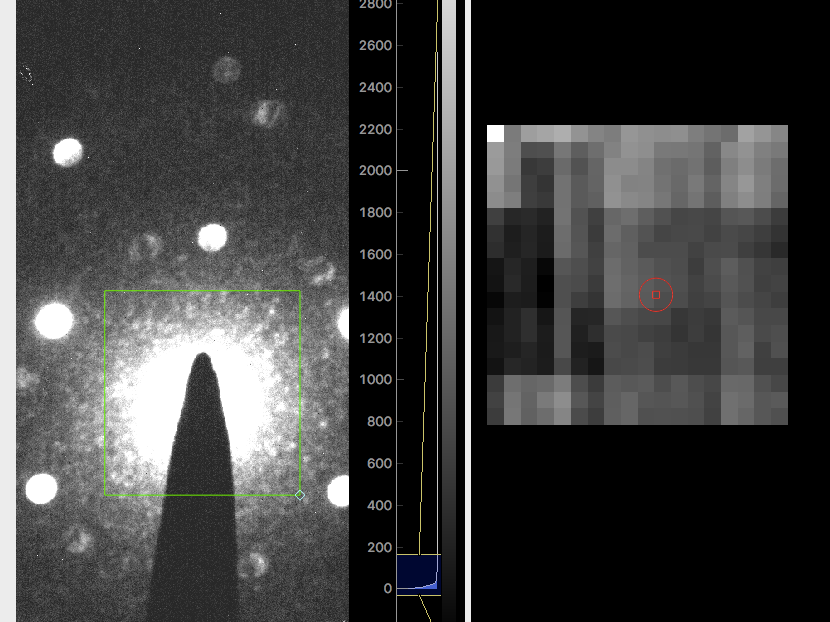


I don’t see any reason to think why the fit on 17 will come out any differently. Now, do note that my fitting routine has yet to be validated, but it seems really hard to believe you could get a hexagon with such square edges.

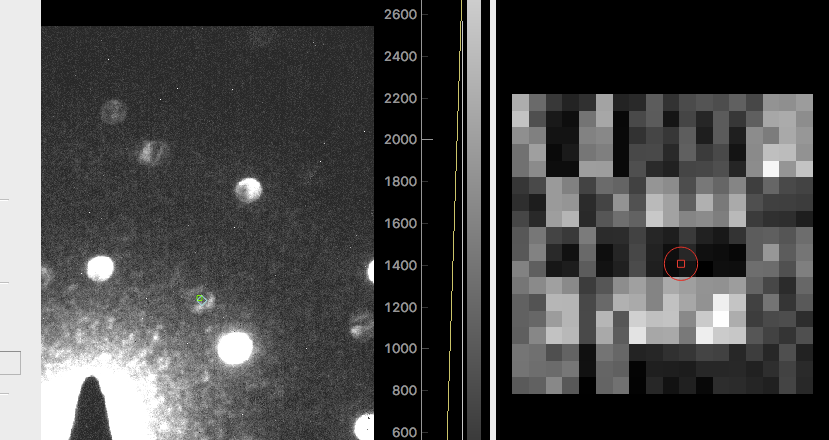
Probing some alternative (smaller) datasets:

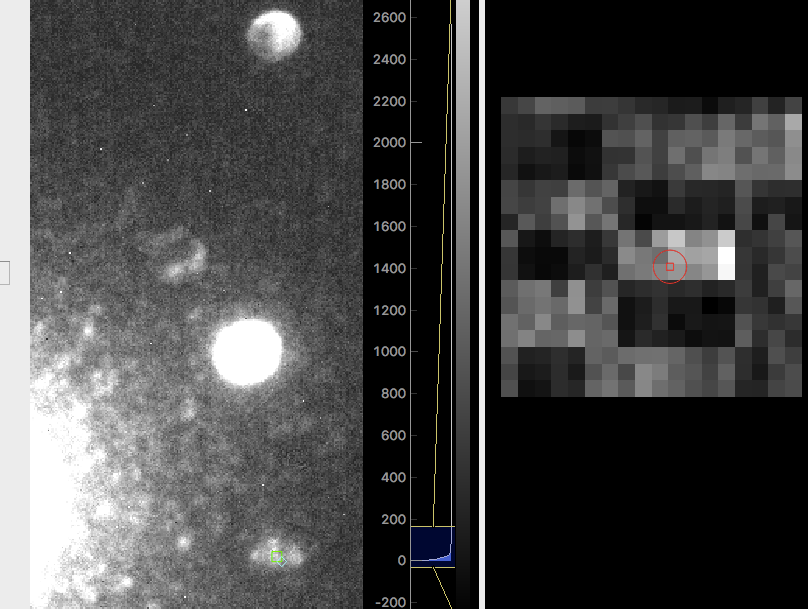
Dataset 3: example pictures.





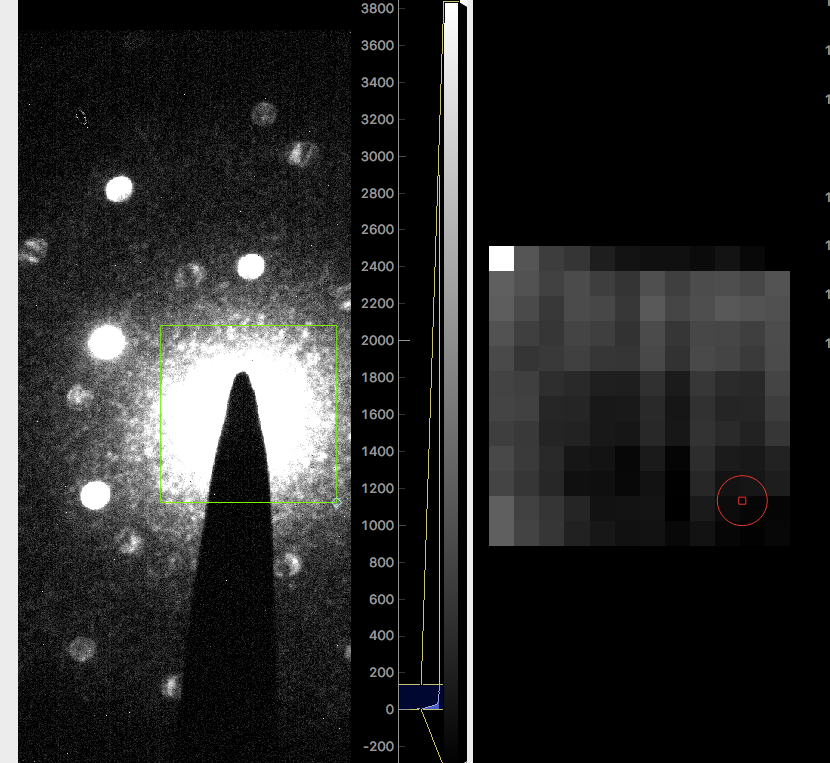
then normalized.



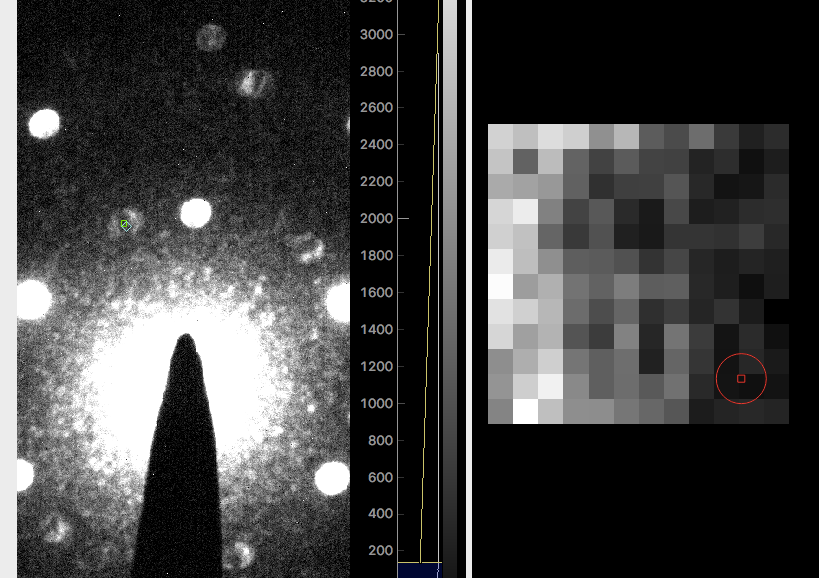


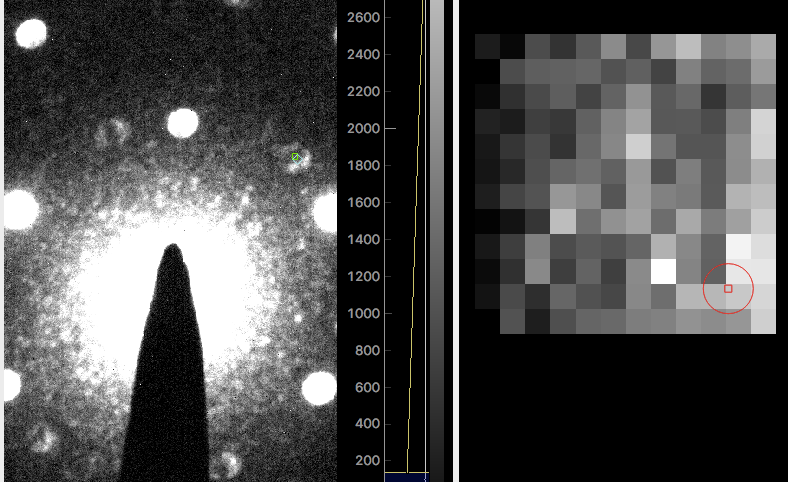
Yuck. This frankly looks just as bad, even though the data is on a much smaller grid. Is there a significance to that?

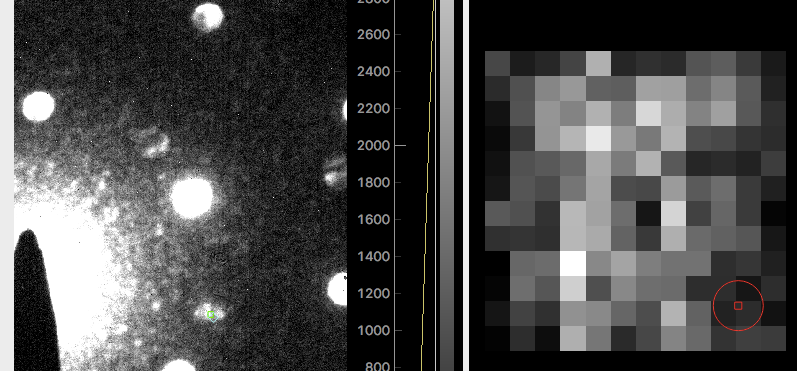
Dataset 4, before renormalization:

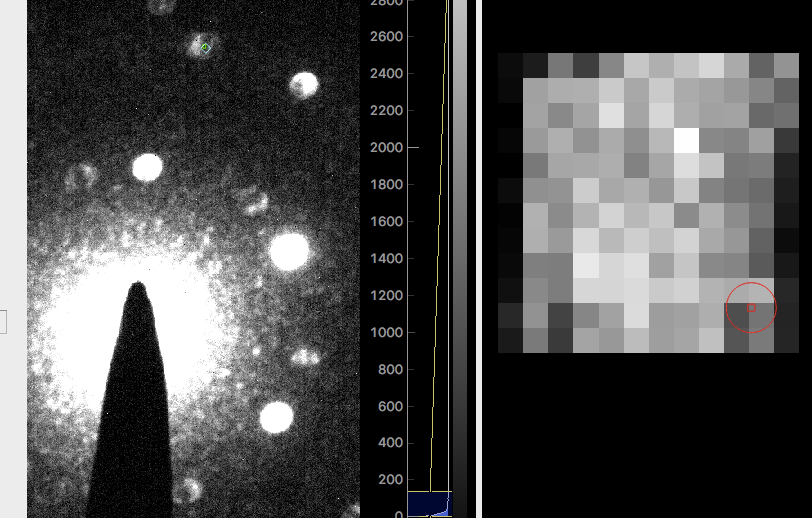


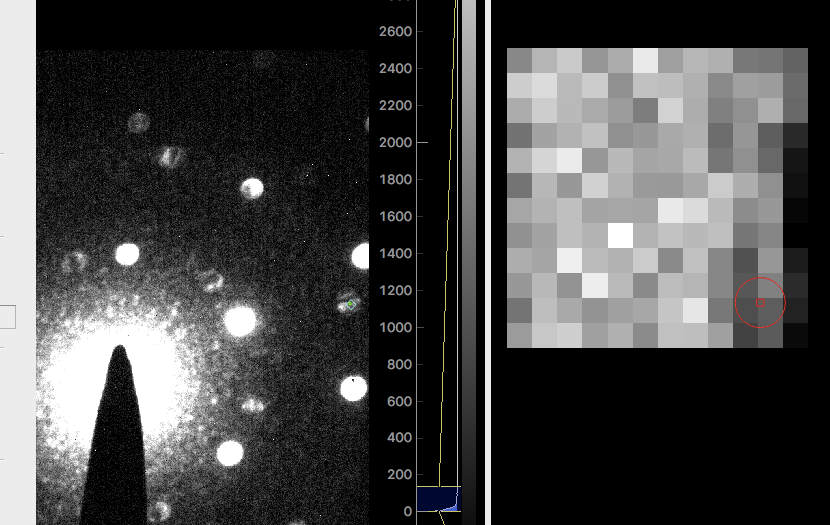
After normalization:

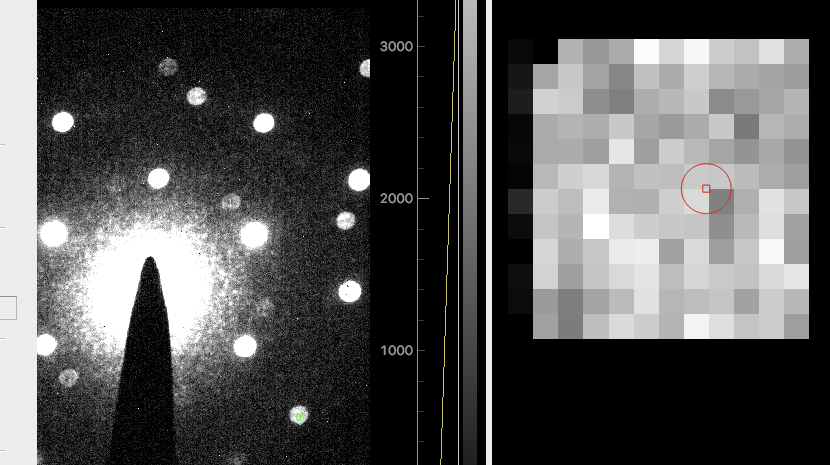










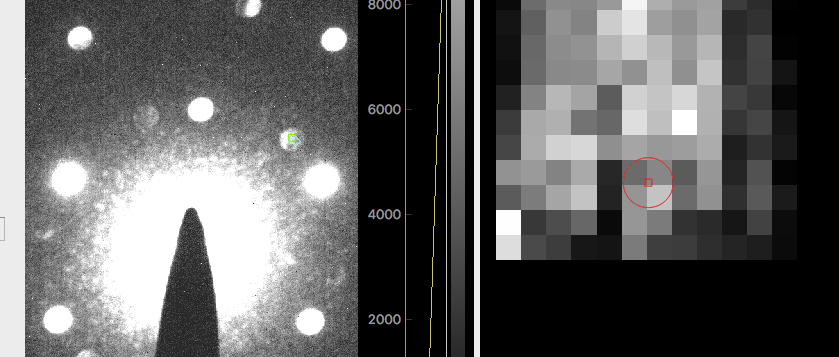


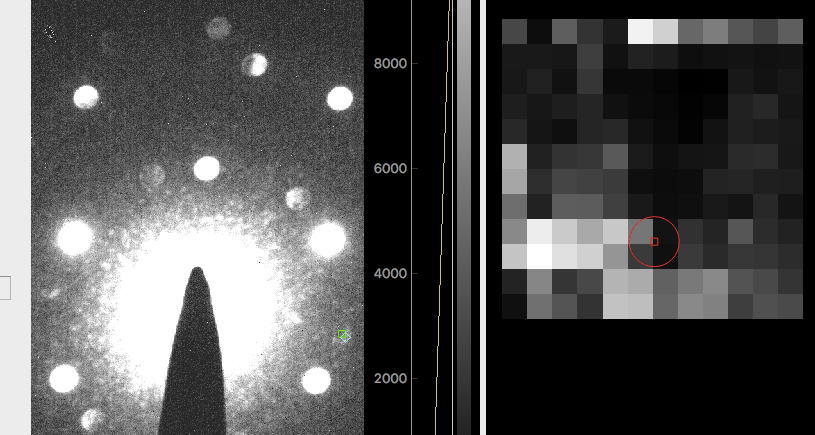
These data seem to line up in a coherent way! I’m surprised the outer disks are doubling the inner disks, but then again it is defocused, so I’m playing a bit fast and loose with the model anyways.

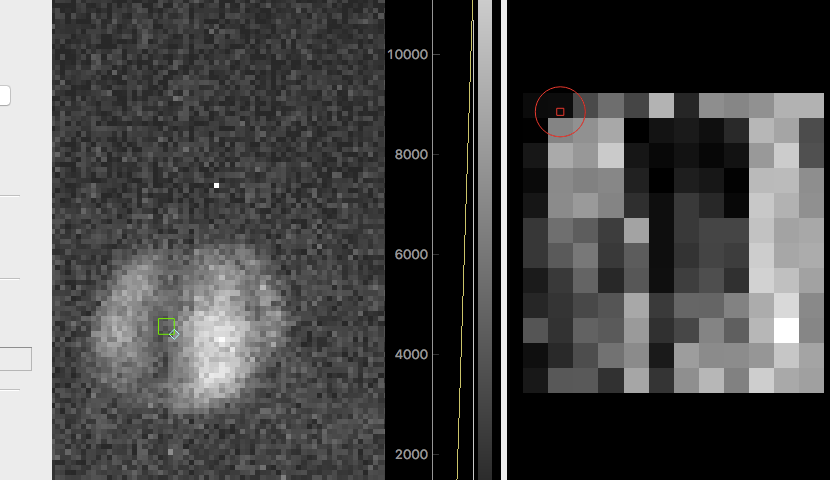
We could try to DSC model this one. Might need some simulations with the defocus? Hard to say.

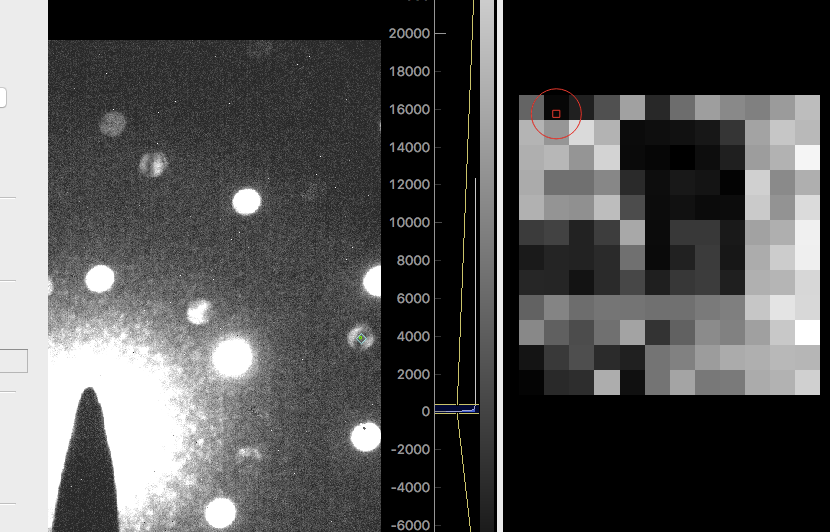
Dataset 5 after normalization:

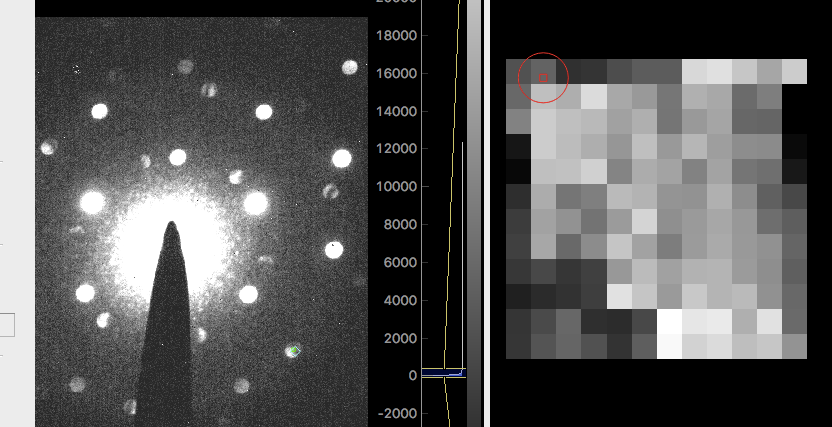












So this is an interesting dataset with some clear trends. Particularly, there aren’t too many hard edges. But we might need to do some defocus simulations to address this.

Datasets 6 and 7 are so defocused that it’s pretty obvious it won’t work to use the simple model for them.

So, we return to some directions:

1. From the long TBLG simulation that I ran, stored, and have never looked at since, verify that we get the striping patterns I would predict on the basis of the TRBLG studies.
2. Validate my fitting method against that simulation. Compute the DSC field (which I didn’t do before, but can), and compare to the recovered DSC field.
3. If satisfactory, return to analyzing the datasets 4 and 5 to see if anything interesting comes out of them.
4. Work out the theory for converting DSC into a strain field. The rate of change of the DSC field relative to the pure Moiré should allow us to do that.