Victor: Code comments Dec 6 2023

General

\*) Did not find anything wrong. Probably can be sped up by a factor of 6 or more.

1.) Should we be using pointers and vector operations? Passing by value versus reference.

2.) The huge nested structure in SpectralFunction.cpp and FunctionsSF make it difficult to debug individual portions of the code (in general).

3.) How many cores? It’s mostly matrix manipulations so could run fast on gpu potentially.

4.) Big speedup possible by taking advantage of the dummy indices. Also for lambda = 0 finite k results should all be the same...

Functions.cpp

1.) W is defined in the old way (as in the PRL)

2.) Matrix inversion is the slow part?

3.) Potential speedup in not having to calculate the condensate external lines as a function of momentum—these will all be the same so can be duplicated/collapsed.

SpectralFunction.cpp [Checked]

1.) Line 64 how does this cutoff loop work…? It seems like a lot of repition with the ground state calc each time.

2.) Self consistent solution in 109. Check it with the way I do it.

3.) Strange 10e6 in norm line 179

4.) 255: Super tiny steps of Epol—this must take forever but probably it’s necessary!!

5.) Purpose of Fixedpoint function