I feel like I am a bit lost as to what my goal is for the RNA-seq portion of this research. When Jing comes in today, I will need to ask her to explain it again to me. I feel like I generally understand what is happening, but I don’t know what my specific task is.

I spoke to Jing and am going to try and figure out how to interpret the data and how they normalized the data to account for the increase of reads at the shine dalgarno sequences. I want to use python to try plotting some of the data in the files because excel cannot handle it. For this I will need numpy and matplot lib for python. Except I can’t figure out how to install these without having to reinstall python all together.

I got numpy and matplotlib working, but Jing said that with the size of the data files, it will not be able to open the files in a reasonable time.

Jing sent me some instructions for how to open and read the files with MochiView that I will try to use. I’m still struggling a bit to understand where the numbers come from and what they represent even though Jing explained it to me multiple times. Following the instructions, I was able to open the files and get something to show up, but I still need help interpreting the information.

I am going to perform some comparisons between the working backwards way and the using lower levels to make higher levels. I expect there to be less circuits but still the same min number of gates for each of the truth values.

10111100 8 6

10111101 8 7

11011010 8 6

11011011 8 7

11100110 8 6

11100111 8 7

The first number is the truth value, the next is the number of gate from the going backward method, the next is the actual minimum number of gates.

I adjusted this piece of code:

for i in range(depthLimit+1):

unformatted = workBackward4(truthValue,i, availableCircuits=availableCircuits)

if unformatted != truthValue:

break

to start range at 3 so there is actually things to look through.

10111100

~~10111101~~

11011010

~~11011011~~

11100110

~~11100111~~

The crossed out truth values are the ones that were fixed by this. I have the remaining ones running, but they seem to be taking longer than I expected. They have only found as little as 7 gates so far

I just noticed, oddly, all the truth values that were supposed to be 7 gates were palindromic truth values. All the 6 gate ones were 1 less than the seven gate ones. Also the seven gate ones were like moving a pair of ones toward the outside. All 7-gate requiring truth values are isomorphs of each other while all 6-gate requiring circuits are isomorphs of each other.

I changed this line of code to use minCircuitFinder6 instead of minCircuitFinder5.

x = minCircuitFinder5(truthValue, num=num)

if type(x) == list:

print "done"

return x

to

x = minCircuitFinder5(truthValue, num=num)

if type(x) == list:

print "done"

return x

elif type(x) == dict:

x = minCircuitFinder6(truthValue, num=num)

This allowed it to find the 6 gate truth values that it missed before, although it does add like 5 seconds to the run time.

I am still running with the first modification to see if the second modification was necessary. It seems like it will take a while so I will just let it run. I copied the combine function and made combine4 which just has the second modification and not the first. It was able to find the 6 gate circuits. I will run it for all truth values and check to see if it finds all minimum circuits. Hopefully my computer will not overheat before it finishes.

I just noticed that these three truth values are probably the save truth values that were missed when we used 5 gates to find 6 gates and 7 gates. They were missed by 6 gates but then found at 7 gates. If this is the case, then this process is essentially the same as going up to 5 then using 5 to go to 6 gates and then 7.