1 The Overall View

For today, we want to find out some things:

1. What is the meaning of the $\log_2 n$-bit block size in the proofs for the block-save adder? It seems that we could have larger block sizes, say $\log_2^2 n$ and still have a polynomial number of input variables for each output block?

2. How can we calculate the L_1 norm for the bits of multiplication function, since this seems to depend still, in the worst case, of n partial product bits, for up to 2 bits.

3. Why is it that a polynomial with a L_1 norm whose difference from that of the desired target function is inverse polynomially bounded is a good approximation for the given function? It seems that two functions could have similar spectral coefficients, and therefore similar L_1 norms, but we very different, for example, just permuting the spectral coefficients of one polynomial representation.

4. What is the relationship between MAJORITY and linear threshold elements?

However, the scheme that has become clear from yesterday is that we have a lower-bound on the number of threshold gates (circuit size) for a particular boolean function, and that comes from calculating its L_1 norm. We have examples of how to do this for the COMPARE and ADD functions, which might give us clues for how to do this for MULTIPLY and DIVIDE. However, the proofs for these seem to involve finding polynomial threshold functions which approximate the boolean function, by bounding the difference in spectral norms. I don't seem to understand this part. There something about needing to approximate taking the logical AND of two approximating polynomials.

However, to make a true and useful comparison to our polylogarithmic depth case, we need also upper bounds. This can be done without much understanding by using the explicit function for ADDITION given in the Bruck-Alon paper. There are ways to unify layers rather than just naive combination. How does that affect circuit size? Unknown. Eliminating layers might change what multinomial terms there are, or polynomial threshold functions, where each one that has m terms and n inputs takes mn threshold gates to implement.

So the two prongs of our approach are constructive (upper bounding) and non-constructive (lower bounding).

2 Lower Bounding Terms in the Polynomial Representation

3 Polynomial Weights in Threshold Functions

The way that the quantum threshold gate works in Hoyer-Spalek, and Takahashi-Tani, is that the rotations by Hamming weight work on a phase out of 2π. Therefore, an exponentially-large weight function, such as COMPARE $\in LT_1$, would need to be simulated in three layers of poly-weighted functions ($\in \hat{LT}_3$). However, it is difficult to get exponential precision in the phase, so we would prefer it be polynomially-weighted. In fact, it is difficult to get more than linear precision in the phase. Rounding off is basically the same approach that allows us to truncate terms on the QFT.

For the explicit constructions of ADD, we can tell that the weights are simply $+1$ and -1 because they are in class MAJ_k. So this doesn't seem to be a huge problem.

4 The Activities of a Bodhisattva

By which I mean, a thesis writer. Keeping notecards is a good idea, writing what I learned each day, as a tangible reminder. I should date them too. They make a huge difference in writing, in giving me something to write about at the end of the day. God it's not even 30 minutes yet.

Other things to write about. Trying to figure out how something works by bullshitting into LaTeX. Even trying to write your thesis is writing your thesis.

5 DIVISION, EXPONENTIATION, Chinese Remainder Theorem

It seems really useful to understand this mixed radix representation. Why would it be easier to work in the coefficients r_i of the mixed radix m_i. Let's try to recreate something from memory.

We have an exponentially large number Z that is larger than a prime P_n. We are going to need to fact-check the hell out of this.

$$P_n = \prod_{i=1}^{n} p_i^{\alpha_i} \tag{1}$$

We define the resides of Z modulo p_i.

$$z_i = Z \bmod p_i \tag{2}$$

6 Literature Search

Tomorrow I should also figure out more recent citations by the Bruck papers, to see if someone has improved these in recent years.