Shor's Algorithm

Elliott Pryor, Benjamin Bushnell, Shengnan Zhou

18 November, 2019

1 Example

To better understand how Shor's algorithm works, let's walk through an example.

Let's say $N = 21$, and we want to find a, b such that $N = a \cdot b$ and a, b are prime numbers. First, we take a random guess $g = 11$. We want to turn g into a better guess $g^{p/2} \pm 1$. Now we want to find p such that $g^p = m \cdot N + 1$ for some m.

We can find p by calculating the period at which a^{x+yp} (mod N) repeats. On a quantum computer this is very fast; however, we can easily do this by hand as 21 is a very small number (and our power is also small). Below, is 11 raised to the powers 0-19 (mod 21).

$$1, 11, 16, 8, 4, 2, 1, 11, 16, 8, 4, 2, 1, 11, 16, 8, 4, 2, 1, 11$$

We can see that the cycle repeats every 6 iterations. Therefore, we know that $p = 6$.

We find $11^6 = 84360 \cdot 21 + 1$. Now we can have the better guess $g^{p/2} + 1 = a \cdot s$ for some factor s, and $g^{p/2} - 1 = b \cdot t$ for some factor t. To find a, b, we need to use Euclid's algorithm to find the common factors. $g^{p/2} + 1 = 1332$, $gcd(1332, 21) = 3$. And $g^{p/2} - 1 = 1330$, $gcd(1330, 21) = 7$.

Now we found the two prime factors of $N = 21$ are $a = 3$ and $b = 7$.