Motivation for a $\mathbb{Z}[i]$-sieve

Zeb Engberg

There are several immediate methods that can be used to find all prime numbers in $\mathbb{Z}[i]$ up to norm x. We outline three below.

1. Use the sieve of Eratosthenes in $\mathbb{Z}$ to generate all prime numbers up to x. Hold these rational primes with a sieve array A. Note that a Gaussian integer $a+bi$ with $a>0$ and $b\geq 0$ is prime in $\mathbb{Z}[i]$ if and only if $\mathrm{N}(a+bi)$ is prime in $\mathbb{Z}$ or $b=0$ and a is a prime in $\mathbb{Z}$. This sieving process takes $O(x\log\log x)$ steps whereas checking each Gaussian integer $a+bi$ with $\mathrm{N}(a+bi)\leq x$ for primality through the sieve array A takes $O(x)$ steps. Hence the entire runtime of this algorithm is $O(x\log\log x)$.

2. Start as in the previous algorithm by generating all rational primes $p\leq x$. If $p\equiv 3 \mod 4$, then p is also prime in $\mathbb{Z}[i]$. If $p\equiv 1 \mod 4$, then p factorizes in $\mathbb{Z}[i]$. Finally, if $p=2$ we have the factorization $2=-i(1+i)^2$.

   To factor those $p\equiv 1 \mod 4$, first compute a primitive fourth root r of 1 in $\mathbb{Z}/p\mathbb{Z}$. Such an r satisfies $r^2+1\equiv 0 \mod p$. Using a random algorithm, we can expect to find such an r in $O(1)$ time per prime. Once r is found, calculating the greatest common divisor of p and $r+i$ in $\mathbb{Z}[i]$ will yield a factor $a+bi$ of p. Calculating this gcd will take at most $O(\log p)$ steps. Summing this over all primes $p\equiv 1 \mod 4$ with $p\leq x$, the number of sieving steps is still the dominate term in the complexity estimate, and this algorithm requires $O(x\log\log x)$ steps just as in the previous algorithm.

3. Rather than starting with a sieve in the rational integers $\mathbb{Z}$ and using information about the splitting behavior of primes $p\in\mathbb{Z}$ lifted to $\mathbb{Z}[i]$, it is possible to sieve directly in $\mathbb{Z}[i]$. The argument giving the runtime of the sieve of Eratosthenes in $\mathbb{Z}$ also shows that this $\mathbb{Z}[i]$-based sieve requires $O(x\log\log x)$ steps.

In this project, we implement the third algorithm described above. Of the three algorithms, the third is the most difficult to design because it requires stepping through a 2-dimensional array rather than the usual 1-dimensional array in the standard sieve of Eratosthenes. If all three algorithms have the same runtime, (and because the primesieve project renders the second algorithm the fastest in practice), why would sieving directly in $\mathbb{Z}[i]$ be of any use?

Sieving directly in the Gaussian integers via the third algorithm becomes useful when working within *segments* of the complex plane. Consider the following example: Given x, y, z, and w, suppose we wish to find all Gaussian primes $a+bi$ with $a\in[x, x+z)$ and $b\in[y, y+w)$. Here we are thinking of x and y as large and z and w as small relative to x and y. Sieving directly in $\mathbb{Z}[i]$ to find such primes would require

$$O(B\log\log B)+O\left(\sum_{N(\pi)\leq B}\frac{zw}{\mathrm{N}(\pi)}\right)$$

steps, where $B=((x+z)^2+(y+w)^2)^{1/2}$. The first term above corresponds to finding all primes $\pi\in\mathbb{Z}[i]$ with norm up to the square-root of the largest element in the sieve array B, and the second term corresponds to crossing off multiples of these primes π in the sieve array indexed by $[x, x+z)\times[y, y+w)$. If x is larger than y, z, w (say), this big-O estimate simplifies to

$$O((x+zw)\log\log x). \tag{1}$$

An analog of one of the first two $\mathbb{Z}$-based sieving algorithms from above would require the computation of rational primes $p \equiv 1 \mod 4$ in the interval

$$\left[x^2 + y^2, (x+z)^2 + (y+w)^2\right).$$

Once the dust settles, such an algorithm would require

$$O(xz \log\log x) \tag{2}$$

steps. If x is large and z is moderate in size, the estimate in line (2) is significantly larger than the estimate in line (1) obtained from sieving directly in $\mathbb{Z}[i]$.

In addition to employing segmented sieving in rectilinear boxes $[x, x+z) \times [y, y+w)$, we can also use segmented sieving to generate Gaussian integer primes in sectors in the complex plane. Given angle measures $0 \leq \alpha < \beta < \pi/2$, consider the primes $\pi \in \mathbb{Z}[i]$ with $\mathrm{N}(\pi) \leq x$ and $\alpha \leq \arg \pi \leq \beta$. Rather than generating all primes π with $\mathrm{N}(\pi) \leq x$ then filtering by angle, we can instead sieve directly in the sector defined by angle measure in the interval $[\alpha, \beta]$. As $\beta - \alpha$ becomes small, $\mathbb{Z}[i]$-sieving will become increasingly more efficient than sieving in $\mathbb{Z}$.

To summarize:

- If one wishes to generate all Gaussian integer primes up to some norm bound, use a $\mathbb{Z}$-based sieve and split the primes $p \equiv 1 \mod 4$ into Gaussian integer primes.
- If one wishes to generate Gaussian integer primes in some smaller subset of the complex plane (a box, a sector), direct sieving in $\mathbb{Z}[i]$ becomes more efficient.