1 Introduction

Dixon’s algorithm is an improvement over Fermat’s factorization method which finds integers \(x\) and \(y\) such that \(n = x^2 - y^2 = (x + y)(x - y)\) and \(n\) gets factored. Dixon’s algorithm tries to find \(x\) and \(y\) efficiently by computing \(x, y \in \mathbb{Z}_n\) such that \(x^2 = y^2 \pmod{n}\). Then with probability \(\geq \frac{1}{2}\), \(x \neq \pm y \pmod{n}\), and hence \(\gcd(x - y, n)\) produces a factor of \(n\) with probability \(\geq \frac{1}{2}\).

2 Algorithm

Here are the steps of the algorithm.

1. Randomly select \(a \in \mathbb{Z}_n\).
2. Let \(b = a^2 \pmod{n}\).
3. Check if \(b\) is \(k\)-smooth [\(k\) to be defined later].
4. If YES, let \(b = \prod_{i=1}^{t} p_i^{\alpha_i}\) where \(\{p_1, \cdots, p_t\}\) is the set of primes \(\leq k\).
5. Collect \(t+1\) such pairs \((a_1, b_1), (a_2, b_2), \cdots, (a_{t+1}, b_{t+1})\).
6. Let \(b_j = \prod_{i=1}^{t} p_i^{\alpha_{ij}}\).
7. Find \(\beta_j\)'s such that \(\sum_{j=1}^{t+1} \beta_j \alpha_{ij}\) is even for each \(i\).
8. \(x = \prod_{j=1}^{t+1} a_j^{\beta_j}\) and \(y = (\prod_{j=1}^{t+1} b_j^{\beta_j})^{\frac{1}{2}}\).
3 Analysis

First we discuss why the step 7 is necessary.

Consider
\[
\prod_{j=1}^{t+1} b_j^{\beta_j} \text{ for } \beta_j \in \{0, 1\}
\]

\[
= \prod_{j=1}^{t+1} \prod_{i=1}^{t} p_i^{\beta_j a_{ij}}
\]

\[
= \prod_{i=1}^{t} \sum_{j=1}^{t+1} \beta_j a_{ij}
\]

If the term exponent \( \sum_{j=1}^{t+1} \beta_j a_{ij} \) is even for all \( i = 1 \) to \( t \), then the number is a perfect square over integers.

Now, to find \( \beta_j \)'s such that \( \sum_{j=1}^{t+1} \beta_j a_{ij} \) is even for each \( i \)

\[
\equiv \text{ to find vector } \vec{\beta} \text{ such that } \vec{\beta} \cdot \vec{\alpha}_i = 0 \pmod{2} \text{ for each } i
\]

\[
\equiv \text{ to find } \vec{\beta} \text{ such that } \vec{\beta} \cdot [\vec{\alpha}_1 \vec{\alpha}_2 \cdots \vec{\alpha}_t]^{(t+1) \times t} = 0
\]

which is easy given \( \vec{\alpha}_1, \vec{\alpha}_2, \cdots, \vec{\alpha}_t \).

Also it is easy to check that the \( x \) and \( y \) satisfy

\[
x^2 = \prod_{j=1}^{t+1} a_j^{2\beta_j}
\]

\[
= \prod_{j=1}^{t+1} b_j^{\beta_j} \pmod{n}
\]

\[
= y^2 \pmod{n}
\]

Now the problem is to find

How many \( k \)-smooth \( b \)'s exist in \( Z_n \) of the form \( a^2 \pmod{n} \)?

Let \( T \) be the number of \( b \)'s of the above kind.

Recall that \( \Psi(n,k) = \{ m \leq n \mid m \text{ is } k\text{-smooth} \} \) and \( \psi(n,k) = |\Psi(n,k)| \).

Then it is easy to see that,

\[
T \geq \psi(\sqrt{n},k) \left[ \text{taking all } k\text{-smooth numbers upto } \sqrt{n} \text{ as } a's \right]
\]

\[
\approx \left( \frac{1}{\ln k} \right)^{\frac{1}{2} \ln n} \frac{1}{\ln n}
\]

But we need to find a better lower bound of \( T \).

[ To be continued in the next lecture ].