Overview

Definitions

Lemmas

Theorem

Conclusion

Citations

Factoring and TFNP Part 2

Akash Kumar[1], Yuriko Nishijima[2]

February 1, 2024

[1] Columbia University, New York. abk2187@columbia.edu
[2] Columbia University, New York. yn2411@columbia.edu

Where We Got Up To

Theme: How FACTORING fits into subclasses of TFNP.
First paper proved:

1. 4GOODINTEGERFACTORING $\in$ PPA
   - Reduction to LONELY
   - "4Good" means $N \equiv 1 \pmod 4$ and -1 is not quadratic residue mod N
2. GOODINTEGERFACTORING $\leq^{\mathsf{RP},\frac{1}{2}}$ PPP
   - Randomized reduction to PIGEON
   - "Good" means -1 is not quadratic residue mod N
   - ERH allows us to derandomize result by guarantee n.q.r. $a \in (1, 3 \cdot \log^2(N)]$

Where We're Going

Second paper builds on this by dropping the "4Good" or "Good" requirements.

1. FACTORING randomly reduces to $\mathrm{PWPP} \subseteq \mathrm{PPP}$
2. FACTORING randomly reduces to PPA

The Trajectory of (1):

$$\text{FACTORING} \leq^* \text{FACROOT} \leq^* \text{WEAKFACROOT} \leq \text{FACROOTMUL} \in \mathrm{PWPP}$$

To prove the above, we need to define the Jacobi Symbol and variations of the FACTORING problem...

($*$ denotes a randomized reduction)

Legendre Symbol

For an odd prime p, the Legendre Symbol is

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & \text{if } p \mid a \\ +1 & \text{if } p \nmid a \text{ and } a \text{ is quadratic residue mod } p \\ -1 & \text{if } p \nmid a \text{ and } a \text{ is not quadratic residue mod } p \end{cases}$$

It is efficiently computable (will follow from later discussion).

Jacobi Symbol

More generally, for odd $N = p_1^{\alpha_1} \cdots p_k^{\alpha_k}$, the Jacobi Symbol is

$$\left(\frac{a}{N}\right) = \prod_{i=1}^{k} \left(\frac{a}{p_i}\right)^{\alpha_i}$$

It is efficiently computable, even without knowing the prime factorization of N, due to Quadratic Reciprocity:

$$\left(\frac{M}{N}\right)\left(\frac{N}{M}\right) = (-1)^{\frac{(M-1)(N-1)}{4}} = \begin{cases} -1 & \text{if } M \equiv N \equiv 3 \pmod 4 \\ +1 & \text{otherwise} \end{cases}$$

along with two "base cases":

$$\left(\frac{-1}{N}\right) = (-1)^{\frac{N-1}{2}} \qquad \left(\frac{2}{N}\right) = (-1)^{\frac{N^2-1}{8}}$$

Interpreting the Jacobi Symbol

The meaning that the Jacobi Symbol tells you is more complicated than the Legendre Symbol, and is why determining if a is a quadratic residue mod N isn't easy.

- If $\left(\frac{a}{N}\right) = -1$, then you know a is not a q.r. mod N
- If $\left(\frac{a}{N}\right) = 1$, then a could or could not be a q.r. mod N

Why the uncertainty? Suppose $N = pq$. There are two cases.

- $\left(\frac{a}{N}\right) = \left(\frac{a}{p}\right)\left(\frac{a}{q}\right) = 1 \times 1 \implies a$ is a q.r.
- $\left(\frac{a}{N}\right) = \left(\frac{a}{p}\right)\left(\frac{a}{q}\right) = (-1) \times (-1) \implies a$ is not a q.r.

Follows that if we could factor N, then we could efficiently determine if a is a q.r. or not.

Fact: For $N = \prod_{i=1}^{k} p_i^{\alpha_i}$ and a s.t. $\gcd(a, N) = 1$,

$$a \text{ is a q.r. mod } N \text{ iff } \left(\frac{a}{p_i}\right) = 1 \text{ for all } i \in [k].$$

Variants of FACTORING

FACROOT(N, a): Given odd N, and a s.t. $\left(\frac{a}{N}\right) = 1$, find

- nontrivial divisor of N, or
- square root of a

FACROOTMUL(N, a, b): Given odd N, and $a, b \in \mathbb{Z}$, find

- nontrivial divisor of N, or
- square root of one of a, b, or ab

WEAKFACROOT(N, a, b): Given odd N, and a, b s.t. $\left(\frac{a}{N}\right) = 1$ and $\left(\frac{b}{N}\right) = -1$, find

- nontrivial divisor of N, or
- square root of a

FACTORING $\leq^*$ FACROOT

FACTORING $\leq^*$ FACROOT

If N is even or a perfect power, then factoring is easy; assume $N = \prod_{i=1}^{k} p_i^{\alpha_i}$, with $k \geq 2$.

FACTORING $\leq^*$ FACROOT

If N is even or a perfect power, then factoring is easy; assume $N = \prod_{i=1}^{k} p_i^{\alpha_i}$, with $k \geq 2$.

Choose random $a \in \{1, \ldots, N-1\}$. If $\gcd(a, N) \neq 1$, return the gcd as the factor. Else, feed (N, a) to the FACROOT oracle.

FACTORING $\leq^*$ FACROOT

If N is even or a perfect power, then factoring is easy; assume $N = \prod_{i=1}^{k} p_i^{\alpha_i}$, with $k \geq 2$.

Choose random $a \in \{1, \ldots, N-1\}$. If $\gcd(a, N) \neq 1$, return the gcd as the factor. Else, feed (N, a) to the FACROOT oracle.

We want a to satisfy $\left(\frac{a}{N}\right) = 1$ and a not a q.r. so that the FACROOT oracle is forced to return a factor of N.

Factoring $\leq^*$ FacRoot

If N is even or a perfect power, then factoring is easy; assume $N = \prod_{i=1}^{k} p_i^{\alpha_i}$, with $k \geq 2$.

Choose random $a \in \{1, \ldots, N-1\}$. If $\gcd(a, N) \neq 1$, return the gcd as the factor. Else, feed (N, a) to the FacRoot oracle.

We want a to satisfy $\left(\frac{a}{N}\right) = 1$ and a not a q.r. so that the FacRoot oracle is forced to return a factor of N.

First, what's the probability that $\left(\frac{a}{N}\right) = 1$? Among $a \in \mathbb{Z}_N^*$, there's a half chance that $\left(\frac{a}{N}\right) = 1$. In fact, we can improve from $1/2$ to 1 with the following trick: instead of randomly choosing a, now randomly choose $a, b \in [N-1]$. Among $c \in \{a, b, ab\}$, take the first so that $\left(\frac{c}{N}\right) = 1$. Now you are guaranteed to find an element with Jacobi Symbol equal to 1 because $\left(\frac{a}{N}\right) = \left(\frac{b}{N}\right) = -1 \implies \left(\frac{ab}{N}\right) = 1$.

FACTORING $\leq^*$ FACROOT

If N is even or a perfect power, then factoring is easy; assume $N = \prod_{i=1}^{k} p_i^{\alpha_i}$, with $k \geq 2$.

Choose random $a \in \{1, \ldots, N-1\}$. If $\gcd(a, N) \neq 1$, return the gcd as the factor. Else, feed (N, a) to the FACROOT oracle.

We want a to satisfy $\left(\frac{a}{N}\right) = 1$ and a not a q.r. so that the FACROOT oracle is forced to return a factor of N.

First, what's the probability that $\left(\frac{a}{N}\right) = 1$? Among $a \in \mathbb{Z}_N^*$, there's a half chance that $\left(\frac{a}{N}\right) = 1$. In fact, we can improve from $1/2$ to 1 with the following trick: instead of randomly choosing a, now randomly choose $a, b \in [N-1]$. Among $c \in \{a, b, ab\}$, take the first so that $\left(\frac{c}{N}\right) = 1$. Now you are guaranteed to find an element with Jacobi Symbol equal to 1 because $\left(\frac{a}{N}\right) = \left(\frac{b}{N}\right) = -1 \implies \left(\frac{ab}{N}\right) = 1$.

Next, what's the probability that a random residue $c \in [N-1]$ s.t. $\left(\frac{c}{N}\right) = 1$ is a quadratic residue? By the "Fact" from earlier, it's $\frac{1}{2^k}$. Hence, our success probability is $1 - \frac{1}{2^k} \geq \frac{1}{2}$.

Akash Kumar, Yuriko Nishijima

Overview
Definitions
Lemmas
Theorem
Conclusion
Citations

FacRoot $\leq^*$ WeakFacRoot

Recall that the input to FacRoot is (N, a), and the input to WeakFacRoot is (N, a, b), so all we need to do is find b s.t. $\left(\frac{b}{N}\right) = -1$.

To do so, we pick a random $b \in [N-1]$, and this shall succeed with probability $\frac{1}{2}$.

By succeed, we mean

- $\gcd(b, N) \neq 1$, so return that factor; or,
- $\left(\frac{b}{N}\right) = -1$

WeakFacRoot $\leq$ FacRootMul

Recall that WeakFacRoot takes (N, a, b) as input, and so does FacRootMul. I claim that to solve WeakFacRoot, one can simply pass the given input (N, a, b) to the FacRootMul oracle.

FacRootMul(N, a, b) could never return a square root of b or ab since $\left(\frac{b}{N}\right) = \left(\frac{ab}{N}\right) = -1$. Hence, the output of FacRootMul(N, a, b) works.

FacRootMul $\in$ PWPP

We are given, as input, (N, a, b). If either a or b shares a factor with N, return it; so we assume $\gcd(a, N) = \gcd(b, N) = 1$. Consider the polytime-computable function $f : \{0, 1, 2\} \times \left\{1, \ldots, \frac{N-1}{2}\right\} \to \{1, \ldots, N-1\}$:

$$f(i, x) = \begin{cases} a_i x^2 \pmod{N} & \text{if } \gcd(x, N) = 1 \\ x & \text{otherwise} \end{cases}$$

where $a_0 = 1, a_1 = a, a_2 = b$.

The domain of f is $3/2$ times larger than its codomain, so the WeakPigeon oracle gives us a collision: (i, x) and (j, y) s.t. $f(i, x) = f(j, y)$ and $(i, x) \neq (j, y)$.

Again, we assume $\gcd(x, N) = \gcd(y, N) = 1$, as otherwise we can factor N.

FacRootMul $\in$ PWPP

With the collision, there are two cases to consider.

Case 1: $i = j$ (good case)
Then $f(i, x) = f(j, y) \implies x^2 \equiv y^2$. In addition, $x \not\equiv \pm y$ since $(i, x) \neq (j, y)$, which means that $\gcd(N, x - y)$ returns a nontrivial factor of N.

Case 2: $i < j$ (cop out case)
Then $f(i, x) = f(j, y) \implies (xy^{-1})^2 = a_j a_i^{-1}$.

- $(xy^{-1})^2 = a$
  Return xy^{-1}
- $(xy^{-1})^2 = b$
  Return xy^{-1}
- $(xy^{-1})^2 = ba^{-1} \implies (axy^{-1})^2 = ab$
  Return axy^{-1}

$\textsc{Factoring} \in^* \mathsf{PWPP} \cap \mathsf{PPA}$

$$\textsc{Factoring} \leq^* \textsc{FacRoot} \leq^* \textsc{WeakFacRoot} \leq \textsc{FacRootMul} \in \mathrm{PWPP}$$

Via the chain of reductions above, we have shown that $\textsc{Factoring}$ is randomly reducible to $\textsc{WeakPigeon}$.

The paper additionally shows that $\textsc{Factoring}$ is randomly reducible to $\textsc{Lonely}$, i.e. $\textsc{Factoring} \in^* \mathsf{PPA}$.

Hence,

$$\textsc{Factoring} \in^* \mathsf{PWPP} \cap \mathsf{PPA}$$

Citations

1. [1]
2. [2]

Joshua Buresh-Oppenheim.
On the TFNP complexity of factoring.
2006.

Emil Jerabek.
Integer factoring and modular square roots.
2015.