A deterministic algorithm for finding r-power divisors

15th ANTS, 12.08.2022, Bristol, UK

D. Harvey (UNSW), M. Hittmeir (SBA Research)

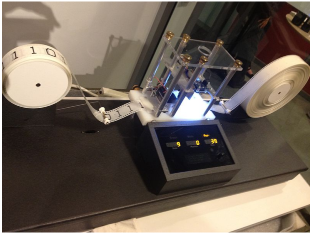
Introduction

Factorization Problem

Find all prime factors of natural numbers N.

The *theoretical* study of this problem concerns...

- ...algorithms for *deterministic* Turing machines.
- ...*rigorous* proofs for the worst-case runtime.



Credit: `https://commons.wikimedia.org/wiki/File:Model_of_a_Turing_machine.jpg`

Deterministic Integer Factorization

- Until 1974: Trial Division, $\widetilde{O}(N^{1/2})$
- 1974: Method of Lehman, $\widetilde{O}(N^{1/3})$
- 1974-1977: Pollard-Strassen approach, $\widetilde{O}(N^{1/4})$
- 2020-2022: Combining Lehman and Pollard-Strassen, $\widetilde{O}(N^{1/5})$

Q: What about divisors of certain shape?

r-Power Factorization Problem

For $r, N \in \mathbb{N}$, find all positive integers p such that $p^r \mid N$.

Previously best (rigorous) result due to Pollard and Strassen:

- All divisors of N less than B can be found in $O(B^{1/2+\varepsilon})$
- If $N = p^r q$, then either $p \leq N^{1/(r+1)}$ or $q \leq N^{1/(r+1)}$
- Hence: Problem can be solved in time $O(N^{1/2(r+1)+\varepsilon})$

For example: Square divisors $(r = 2)$ can be found in $O(N^{1/6+\varepsilon})$

Coppersmith and BDHG

Our improvement is based on *Coppersmith's method*:

1. Find all divisors of N in an interval via lattice methods
2. Choose a sequence of intervals that covers $[1, N^{1/2}]$

Boneh, Durfee and Howgrave-Graham: $N = p^r q$ with $p \approx q$

1. Adaptation of Coppersmith's method
2. Faster than ECM when $r \approx (\lg p)^{1/2}$

Our goal: Estimate worst-case complexity for arbitrary p, q, r

Main Result

Theorem 1

Let $N \geq 2$ and $r \leq \log_2 N =: \lg N$.
We can find all positive integers p with $p^r \mid N$ in time

$$O\left(N^{1/4r}\frac{(\lg N)^{10+\varepsilon}}{r^3}\right).$$

Our method finds square divisors $(r = 2)$ in $O(N^{1/8+\varepsilon})$
The space complexity is negligible

Searching one interval

Let $H, P \in \mathbb{N}$ with $H < P \leq N^{1/r}$.
We first discuss an algorithm that outputs a list of all integers p with $p^r \mid N$ and

$$P - H \leq p \leq P + H.$$

Strategy

1. Construct polynomials f_i, $i = 0, \ldots, d-1$, such that $f_i(p - P) \equiv 0 \mod p^{rm}$. Here we need $rm \leq d$.
2. Compute $g \in \mathrm{span}_{\mathbb{Z}}(f_i)$ with $|g(p - P)| < p^{rm}$.
3. We get $g(p - P) = 0$, hence $p - P$ is an integer root of g.

A key tool to achieve this:

LLL lattice reduction

Let $v_0, \ldots, v_{d-1} \in \mathbb{Z}^d$ be linearly independent. We may find a nonzero $w \in L := \mathrm{span}_{\mathbb{Z}}(v_0, \ldots, v_{d-1})$ such that

$$||w||_2 \leq 2^{(d-1)/4} (\det L)^{1/d}.$$

- We may take w as the first vector in a reduced basis for L
- The runtime complexity is polynomial w.r.t. the input size

Consider the polynomials $f_0, \ldots, f_{d-1}$ defined by

$$f_i(x) := \begin{cases} N^{m-\lfloor i/r \rfloor}(P+x)^i, & 0 \leq i < rm, \\ (P+x)^i, & rm \leq i < d. \end{cases}$$

Let $\tilde{f}_i(y) := f_i(Hy)$. Let v_i be the coefficient vector of $\tilde{f}_i$.

For $L := \operatorname{span}_{\mathbb{Z}}(v_0, \ldots, v_{d-1})$, we now compute $\det L$:

- Consider the *dxd*-matrix with the v_i as its rows
- Since $\deg f_i = i$, this is a lower triangular matrix
- Diagonal entries $\ldots \begin{cases} N^{m-\lfloor i/r \rfloor}H^i, & 0 \leq i < rm, \\ H^i, & rm \leq i < d. \end{cases}$

$$\det L = H^{1+2+\cdots+(d-1)} \underbrace{(N^m \cdots N^m)}_{r \text{ terms}} \underbrace{(N^{m-1} \cdots N^{m-1})}_{r \text{ terms}} \cdots \underbrace{(N \cdots N)}_{r \text{ terms}}$$
$$= H^{1+2+\cdots+(d-1)}(N^{1+2+\cdots+m})^r$$
$$= H^{d(d-1)/2} N^{rm(m+1)/2}$$

Applying LLL reduction to the v_i, we obtain $w \in L$ with

$$||w||_2 \leq 2^{(d-1)/4} H^{(d-1)/2} N^{rm(m+1)/2d} =: \Lambda.$$

This vector corresponds to a nonzero $\tilde{g}(y) = \tilde{g}_0 + \cdots + \tilde{g}_{d-1} y^{d-1}$.
Define $g(x) := \tilde{g}(x/H)$.

If $d^{1/2} \cdot \Lambda < (P-H)^{rm}$, then $x_0 := p - P$ is a root of g.

Proof.

We first show that $p^{rm} \mid g(x_0)$ by proving $p^{rm} \mid f_i(x_0)$ for all i:

- For $0 \leq i < rm$, we have $f_i(x_0) = N^{m-\lfloor i/r \rfloor} p^i \equiv 0 \mod p^{rm}$.
- For $i \geq rm$, we have $f_i(x_0) = p^i \equiv 0 \mod p^{rm}$.

Now $-H \leq x_0 \leq H$ implies that

$$\begin{aligned}|g(x_0)| \leq |h_0| + \cdots + |h_{d-1}|H^{d-1} &= |\tilde{g}_0| + \cdots + |\tilde{g}_{d-1}| \\ &\leq d^{1/2}||w||_2 < (P-H)^{rm} \leq p^{rm}.\end{aligned}$$

We obtain $g(x_0) = 0$. $\square$

Root-finding step

The last step of this section is about finding all integer roots of g.

Theorem 2

For $b, n \in \mathbb{N}$, let $f \in \mathbb{Z}[x]$ with $\deg f = n$ and $||f||_\infty \leq 2^b$.
We may find all integer roots of f in time $O(n^{2+\varepsilon} b^{1+\varepsilon})$.

This is proved in the *appendix* of our paper.

Proof of the main result

- In our proof above, we assumed $d^{1/2} \cdot \Lambda < (P-H)^{rm}$.
- Hence, it only works for *small* intervals $[P-H, P+H]$.
- For proving Theorem 1, we want to cover the range $[1, N^{1/r}]$.

Strategy

1. Consider a general interval $T \leq p \leq T'$.
2. Cover it with a sequence of subintervals $[P-H, P+H]$.
3. Minimize the number of subintervals by maximizing

$$H < \frac{1}{d^{1/(d-1)}2^{1/2}} \cdot \frac{T^{2rm/(d-1)}}{N^{rm(m+1)/d(d-1)}} =: \tilde{H}.$$

One finds that $\tilde{H}$ is largest for $m/d \approx \lg T / \lg N$.

Let $T = N^{\theta/r} > 4^{\sqrt{\lg N/r}}$, where $\theta \in [0, 1]$.

- Set $d := \lceil \lg N \rceil + 1$ and $m := \lfloor (d-1) \lg T / \lg N \rfloor$.

  One can show that this implies

  $$\tilde{H} > \frac{N^{\theta^2/r} N^{-1/\lg N}}{3} = \frac{N^{\theta^2/r}}{6} > 2.$$

  Compute $H := \lceil \tilde{H} \rceil - 1$.

- Invoke the algorithm of the previous section for $P = T + H, P = T + 3H, \ldots$ until $[T, T']$ is covered.

- The number of subintervals dominates the complexity:

  $$\left\lceil \frac{T' - T}{2H} \right\rceil \in O\left(\frac{T' - T}{N^{\theta^2/r}} \right) = O\left(\frac{T' - T}{T} \cdot N^{\theta(1-\theta)/r} \right)$$

To finish the proof, we now do the following:

- Check the numbers up to $4^{\sqrt{\lg N/r}}$ for p with $p^r \mid N$.
- The remaining range $[4^{\sqrt{\lg N/r}}, N^{1/r}]$ is divided into intervals of the form $[T, T'] = [2^k, 2^{k+1}]$.
- Each of this intervals can be searched in

$$O\left(\frac{T' - T}{T} \cdot N^{\theta(1-\theta)/r+\varepsilon}\right) \subseteq O(N^{1/4r+\varepsilon}),$$

  where we have used that $\theta(1-\theta) \leq 1/4$.
- The number of intervals is bounded by $\lceil \lg(N^{1/r}) \rceil$.

  All other steps are negligible.

$\square$

Discussion

- The maximum value $1/4$ of $\theta(1-\theta)$ is reached for $\theta = 1/2$. Hence, the "hardest" case is $p \approx N^{1/2r}$.
- $\theta(1-\theta)$ is much smaller than $1/4$ for most $\theta \in [0,1]$.

  We may thus improve the logarithmic factors in the bound.
- Our result on integer root finding yields an explicit power of $\log N$ in the bound of Coppersmith's method.
- We wanted to apply ideas from the $N^{1/5}$-improvement to r-power factorization, but without success.

References

1. R. S. Lehman, *Factoring Large Integers*, Math. Comp. 28, 1974, 637-646.
2. J. M. Pollard, *Theorems on factorization and primality testing*, Proc. Cambridge Philos. Soc. 76, 1974, 521–528.
3. V. Strassen, *Einige Resultate über Berechnungskomplexität*, Jber. Deutsch. Math.-Verein. 78(1), 1976/77, 1-8.
4. D. Harvey, M. Hittmeir, *A log-log speedup for exponent one-fifth deterministic integer factorization*, Math. Comp. 91, 2022, 1367-1379.
5. D. Coppersmith, *Small solutions to polynomial equations, and low exponent RSA vulnerabilities*, J. Cryptology 10(4), 1997, 233–260.
6. D. Boneh, G. Durfee, and N. Howgrave-Graham, *Factoring $N = p^r q$ for large r*, Advances in cryptology—CRYPTO '99 (Santa Barbara, CA), Lecture Notes in Comput.Sci. 1666, 1999, 326–337.

Questions?

mhittmeir@sba-research.org