Sparse interpolation and small primes in arithmetic progressions

Daniel S. Roche

Symbolic Computation Group
School of Computer Science
University of Waterloo

CMS Winter Meeting
Windsor, Ontario
December 5, 2009

Joint work with Mark Giesbrecht:

Preliminary version at MACIS, December 5–7, Paris
Accepted to *Computational Complexity*
Available on the arXiv

Interpolating an unknown polynomial

Example

$$f = (x-3)^{107} - 485(x-3)^{54}$$

Suppose we can evaluate $f(\theta)$ at any chosen point θ.

- Can we find a formula for f?

Interpolating an unknown polynomial

Example

$$f = (x-3)^{107} - 485(x-3)^{54}$$

Suppose we can evaluate $f(\theta)$ at any chosen point θ.

- Can we find a simple formula for f?

Interpolating an unknown polynomial

Example

$$f = (x-3)^{107} - 485(x-3)^{54}$$

Suppose we can evaluate $f(\theta)$ at any chosen point θ.

- Can we find a simple formula for f
  in a reasonable amount of time?

Shifted-Lacunary Interpolation

Want to interpolate an unknown polynomial $f \in \mathbb{Q}[x]$ into:

Definition (Shifted-Lacunary Representation)

$$f = c_0 + c_1(x-\alpha)^{e_1} + c_2(x-\alpha)^{e_2} + \cdots + c_t(x-\alpha)^{e_t},$$

where $e_1 < \cdots < e_t = n$ and t is minimal for any α

- Can be reduced to finding the sparsest shift α.
- No previous polynomial-time algorithm known.

Black Box Model

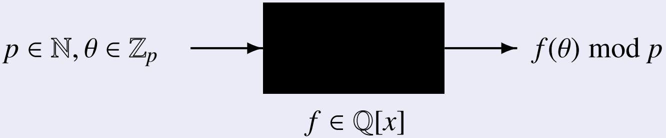
Arbitrary evaluations will usually be very large:

Example

$$
\begin{aligned}
f &= (x-3)^{107} - 485(x-3)^{54} \\
f(1) &= -162259276829222100374855109050368
\end{aligned}
$$

To control evaluation size, use modular arithmetic:

The "Modular Black-Box"



Modular-Reduced Polynomial

Definition (Modular-reduced Polynomial)

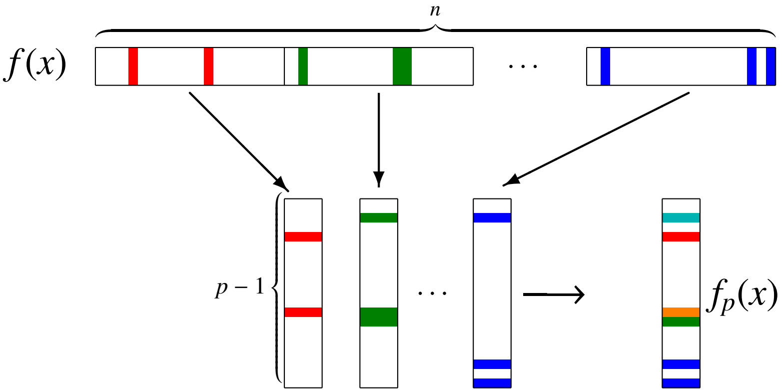
For $f \in \mathbb{Q}[x]$, $f^{(p)}$ is the unique polynomial in $\mathbb{Z}_p[x]$ with degree less than p such that $f \equiv f^{(p)} \bmod (x^p - x)$.

- $f(\theta) \operatorname{rem} p = f^{(p)}(\theta \operatorname{rem} p), \qquad \forall \theta \in \mathbb{Z}$ (Fermat's Little Theorem)
- α_p, the sparsest shift of $f^{(p)}$, is at worst a t-sparse shift of $f^{(p)}$

Example

$$
\begin{aligned}
f &= -4(x-2)^{145} + 14(x-2)^{26} + 3 \\
f^{(11)} &= 3x^6 + 4x^5 + 9x^3 + 6x^2 + 6x + 4 \\
&= 7(x-2)^5 + 3(x-2)^6 + 3
\end{aligned}
$$

Visual description of $f^{(p)}$



Uniqueness and Rationality of Sparsest Shift

Theorem (Lakshman & Saunders (1996))

If the degree is at least twice the sparsity,
then the sparsest shift is unique and rational.

Corollary

If $\deg f^{(p)} \geq 2t$, then $\alpha_p \equiv \alpha \pmod{p}$

Condition not satisfied means p is a “bad prime”,
or the polynomial is dense.

Outline of Algorithm

Input: Modular black box for $f \in \mathbb{Q}[x]$,
Bound B on the bit length of the lacunary-shifted representation

1. Choose a prime p
2. Evaluate $f(0), f(1), \ldots, f(p-1) \operatorname{rem} p$ to interpolate $f^{(p)}$.
3. If $\deg f^{(p)} \geq 2t$, then compute sparsest shift α_p of $f^{(p)}$
4. Repeat Steps 1–3 enough times to recover α
5. Use sparse interpolation to recover $f(x+\alpha)$

Outline of Algorithm

Input: Modular black box for $f \in \mathbb{Q}[x]$,
Bound B on the bit length of the lacunary-shifted representation

1. Choose a prime p
2. Evaluate $f(0), f(1), \ldots, f(p-1) \operatorname{rem} p$ to interpolate $f^{(p)}$.
3. If $\deg f^{(p)} \geq 2t$, then compute sparsest shift α_p of $f^{(p)}$
4. Repeat Steps 1–3 enough times to recover α
5. Use sparse interpolation to recover $f(x+\alpha)$

The challenge: Choosing primes on Step 1
so that Step 3 will succeed

Bad prime: Exponents Too Small

Sparsest shift of $f^{(p)}$ is not α_p

$$f = -4(x-2)^{145} + 14(x-2)^{26} + 3$$

$$p = 13$$

$$\begin{aligned} f^{(13)} &= 9(x-2)^{1} + (x-2)^{2} + 3 \\ &\equiv (x-4)^{2} + 12 \end{aligned}$$

Condition: $(p-1) \nmid e_t(e_t-1)(e_t-2)\cdots(e_t-(2t-2))$

Bad prime: Exponents Collide

Sparsest shift of $f^{(p)}$ is not α_p

$$f = 4(x-1)^{59} + 2(x-1)^{21} + 7(x-1)^{19} + 20$$

$$p = 11$$

$$\begin{aligned} f^{(11)} &= 4(x-1)^{9} + 2(x-1)^{1} + 7(x-1)^{9} + 9 \\ &= 2(x-1) + 9 \\ &\equiv 2(x-2) \end{aligned}$$

Condition: $(p-1) \nmid (e_t - e_1)(e_t - e_2)\cdots(e_t - e_{t-1})$

Bad prime: Coefficients Vanish

Sparsest shift of $f^{(p)}$ is not α_p

$$f = 69(x-5)^{42} - 12(x-5)^{23} + 4$$

$$p = 23$$

$$\begin{aligned} f^{(23)} &= 0(x-5)^{20} + 11(x-5)^{1} + 4 \\ &= 11(x-5) + 4 \\ &\equiv 11(x-13) \end{aligned}$$

Condition: $p \nmid c_t$

Sufficient Conditions

Definition

$$C = \prod_{i=1}^{t-1} e_i \cdot \prod_{i=1}^{t-1} (e_t - e_i) \cdot \prod_{i=0}^{2t-2} (e_t - i) \quad \leq 2^{4B^2}$$

Sufficient Conditions for Success

- $p \nmid c_t$
- $(p-1) \nmid C$

Generating Good Primes

The Problem:

Given $\beta_1 > \log_2 c_t$, $\beta_2 > \log_2 C$, and ℓ
find ℓ small primes p such that $p \nmid c_t$ and $(p-1) \nmid C$.

Primes-in-arithmetic-progressions approach

Choose primes $p = qk + 1$, for distinct primes q.
$q \mid (p-1)$, so $q \nmid C \Rightarrow (p-1) \nmid C$.

A brief history of primes

(in arithmetic progressions)

Definition

For $q \in \mathbb{Z}$, $S(q)$ is the smallest prime p such that $q|(p-1)$.

- Dirichlet (1837): $S(q)$ exists
- Linnik (1944): $S(q) < cq^L$ for some $L > 0$
- Heath-Brown (1992): $S(q) < cq^{5.5}$

For most q:

- Bombieri, Friedlander, Iwaniec (1987): $S(q) < cq^2$
- Rousselet (1988): $S(q) < q^2$ (more explicit)
- Baker & Harman (1996): $S(q) < q^{1.93}$
- Mikawa (2001): $S(q) < q^{1.89}$

Mikawa's Result

Fact (Mikawa 2001)

There exists a constant μ such that, for all $n > \mu$, and for most of the integers $q \in \{n, n+1, \ldots, 2n\}$, with less than $\mu n / \log^2 n$ exceptions, $S(q) < q^{1.89}$.

Theorem

For any $k \in \mathbb{N}$, we can construct a set $\mathcal{Q}$ of primes such that the set $\mathcal{P} = \{S(q) : q \in \mathcal{Q}\}$ has at least k distinct elements, and each $p \in \mathcal{P}$ is $O(k^{1.89} \cdot \log^{1.89} k)$.

Proof of Prime Generation Theorem

Proof sketch

For convenience, define $\Upsilon(n) = \dfrac{3n}{5\log n} - \dfrac{\mu n}{\log^2 n}$

Let $n \in \mathbb{N}$ such that $n > 21$, $n > \mu$, and $\Upsilon(n) > k$. (μ is just a guess.)

Define $\mathcal{Q} = \left\{q \text{ prime: } n \le q < 2n \text{ and } S(q) < q^{1.89}\right\}$

Since $n > 21$, $\#\left\{q \text{ prime: } n \le q < 2n\right\} \ge 3n/(5\log n)$.

Therefore $\#\mathcal{P} = \#\mathcal{Q} > \Upsilon(n) > k$.

(If not, then μ was too small, so double it.)

To make $\Upsilon(n) > k$, we have $n \in O(k \log k)$.
Each $p \in \mathcal{P}$ is $O(n^{1.89})$, so $p \in O(k^{1.89} \cdot \log^{1.89} k)$.

Complexity Implications

Theorem

Given a modular black box for unknown $f \in \mathbb{Q}[x]$ *and a bound* B *on the bit-length of the shifted-lacunary representation, we can compute the sparsest shift* α *with* $O(B^{8.56} \log^{6.78} \cdot (\log\log B)^2 \cdot \log\log\log B)$ *bit operations.*

In fact, we have confirmed for all word-sized q that $S(q) < 2q \log^2 q$.
Proving this would improve the complexity to $\tilde{O}(B^5)$.

A different idea

We have chosen $\mathcal{P}$ so that the following is sufficiently large:

$$\operatorname{lcm}\left(\{p-1 : p \in \mathcal{P}\}\right) \geq \prod_{q \in Q} q$$

How about using this directly?

- Use the first n primes for the algorithm: $p_1, p_2, \ldots, p_n$.
- Define $\Psi(n) = \operatorname{lcm}(p_1 - 1,\ p_2 - 1,\ \ldots,\ p_n - 1)$.
- If $\Psi(n) > C \cdot p_n^k$, then at least k of the first n primes satisfy $(p_i - 1) \nmid C$.

Lower bounds on $\Psi(n)$

Notice that $\Psi(n) = \lambda(P_n)$, where
$\lambda(x)$ is the Carmichael lambda function,
and $P_n = \prod_{i=1}^{n} p_i$ is the nth primorial number.

There are lower bounds on $\lambda(x)$, but they are either
too weak to be useful, or have too many exceptions.

Experimentally, we see strong evidence that $\Psi(n) \gg 2^n$.
This would also give a $\tilde{O}(B^5)$ complexity
(with smaller log factors)

Summary

- Shifted-lacunary interpolation can be performed in polynomial time, for rational polynomials given by a modular black box.
- How to apply these techniques to other problems on lacunary polynomials?
- What about domains other than $\mathbb{Q}[x]$?
- Prove that $S(q) \ll q \log^2 q$.
- Prove that $\Psi(n) \ll 2^n$.