Complexity of
Shifted-Lacunary Polynomial Interpolation

Daniel S. Roche

SIG Theory and Complexity Seminar
University of Delaware
9 January 2009
This is joint work with Mark Giesbrecht.

Complexity of Shifted-Lacunary Polynomial Interpolation

General Problem

Determining a function from its values.

Goals

- Find the simplest possible formula.
- Don’t take too long.

Necessities

- What type of function? (output type)
- How big can it be? (output size)
Example

\[ f = (x - 3)^{107} - 485(x - 3)^{54} \]

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

- Can we find a formula for \( f \)?
Example

\[ f = (x - 3)^{107} - 485(x - 3)^{54} \]

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

- Can we find a simple formula for \( f \)?
Suppose we can evaluate $f(\theta)$ at any chosen point $\theta$. Can we find a simple formula for $f$ in a reasonable amount of time?
Complexity of Shifted-Lacunary Polynomial Interpolation
Dense Methods

Definition (Dense Representation)

\[ f = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n, \]
where \( n = \deg(f) \) and \( a_0, a_1, \ldots, a_n \in \mathbb{R} \)

- Studied by Newton (1711), Waring (1779), \ldots
- Highly efficient implementations available
### Definition (Dense Representation)

\[ f = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n, \]

where \( n = \deg(f) \) and \( a_0, a_1, \ldots, a_n \in \mathbb{R} \)

### Example

For \( f = (x - 3)^{107} - 485(x - 3)^{54} \), we will have

\[
 f = x^{107} - 321x^{106} + 51039x^{105} - 5359095x^{104} + \cdots \\
 + 40200992749659079854837585152311792674303590144819373x \\
- 1127130637840908780976768693419860197828458989848152
\]

This is way too big! (twice exponential in the desired size)
Complexity of Shifted-Lacunary Polynomial Interpolation

(Lacunary polynomials are sometimes called sparse or supersparse.)

- Default representation in Maple, Mathematica, etc.
- Some things are hard (Plaisted 1977, 1984)
- Some things aren’t: Interpolation, finding low-degree factors
- Some things are unknown!
Complexity of Shifted-Lacunary Polynomial Interpolation

Sparse Methods

Definition (Lacunary Representation)

\[ f = b_0 + b_1 x^{d_1} + b_2 x^{d_2} + \cdots + b_s x^{d_s}, \]

where \( d_1 < d_2 < \cdots < d_s = n \) and \( b_1, \ldots, b_s \in \mathbb{R} \setminus \{0\} \)

- Baron de Prony (1795), Ben-Or & Tiwari (1988), Kaltofen, Lakshman, Wiley, Lee, Lobo, . . .
- Need to choose evaluation points
- \( \mathbb{R} \) must have a high-order element and a fast logarithm.
**Definition (Lacunary Representation)**

\[ f = b_0 + b_1 x^{d_1} + b_2 x^{d_2} + \cdots + b_s x^{d_s}, \]

where \( d_1 < d_2 < \cdots < d_s = n \) and \( b_1, \ldots, b_s \in \mathbb{R} \setminus \{0\} \)

**Example**

If \( f = (x - 3)^{107} - 485(x - 3)^{54} \),

this helps iff we know the sparsest shift 3, since \( f(x + 3) = x^{107} - 485x^{54} \) is 2-sparse.
**Complexity of Shifted-Lacunary Polynomial Interpolation**

**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 \( \alpha \)

- This is our problem.
- Can be reduced to finding the sparsest shift \( \alpha \).
- We restrict the domain to \( \mathbb{Q}[x] \).
- No previous polynomial-time algorithm known.
Complexity of Shifted-Lacunary Polynomial Interpolation

We give an algorithm with output-sensitive polynomial-time complexity, specifically, *bit complexity* polynomial in:

- Number of nonzero terms $t$
- Logarithm of the degree $n$
- Size of the coefficients $c_1, \ldots, c_t$
- Size of the sparsest shift $\alpha$

Black box calls are assumed to have constant cost.
Computing the Sparsest Shift

- **Borodin & Tiwari (1991)**
  Compute sparsest shift from evaluation points (open)

- **Grigoriev & Karpinski (1993)**
  Compute sparsest shift from a black-box function.
  State need for complexity *not* polynomial in $n$

- **Lakshman & Saunders (1996)**
  Compute sparsest shift from dense representation

- **Giesbrecht, Kaltofen, Lee (2003)**
  Current best results (deterministic & probabilistic)
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.

Example

\[ f = (x - 3)^{107} - 485(x - 3)^{54} \]

⇒ 3 is the only shift with \( \leq 54 \) terms

Condition not satisfied means polynomial is dense.
Black Box Model

Arbitrary evaluations will usually be very large:

Example

\[ f = (x - 3)^{107} - 485(x - 3)^{54} \]
\[ f(1) = -162259276829222100374855109050368 \]

To control evaluation size, use modular arithmetic:

The “Modular Black-Box”

\[ p \in \mathbb{N}, \theta \in \mathbb{Z}_p \rightarrow f(\theta) \mod p \]
\[ f \in \mathbb{Q}[x] \]
# Modular Reductions

**Definition (Rational remainder)**

\[ a \text{ rem } m = b \iff a \equiv b \mod m \text{ and } 0 \leq b < m. \]

**Definition (Shifted remainder)**

\[ a \text{ rem}_1 m = b \iff a \equiv b \mod m \text{ and } 1 \leq b \leq m. \]

**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 \( f \equiv f^{(p)} \mod (x^p - x) \).
**Modular-Reduced Polynomial**

### Definition

\[
f = c_0 + c_1 (x - \alpha)^e_1 + \cdots + c_t (x - \alpha)^e_t
\]

\[
f^{(p)} = (c_0 \text{ rem } p) + (c_1 \text{ rem } p)(x - \alpha_p)^{e_1 \text{ rem } 1(p-1)} + \cdots + (c_t \text{ rem } p)(x - \alpha_p)^{e_t \text{ rem } 1(p-1)}
\]

where \( \alpha_p \equiv \alpha \mod p \).

- \( f(\theta) \text{ rem } p = f^{(p)}(\theta \text{ rem } p), \quad \forall \theta \in \mathbb{Z} \) (Fermat’s Little Theorem)
- \( \alpha_p \) is at least a \( t \)-sparse shift of \( f^{(p)} \)
$f(x)$

$\cdots$

$p-1$

$n$

$\cdots$

$f_p(x)$
Red squares indicate nonzero terms in the polynomial.

The reel is the unit circle in $\mathbb{Z}_p$. 
Outline of Algorithm

**Input**: Bound $B$ on the bit length of the lacunary-shifted representation

1. Choose a prime $p$ with $p \in O(B^{O(1)})$
2. Evaluate $f(0), f(1), \ldots, f(p - 1) \text{rem} p$ to interpolate $f^{(p)}$.
3. Use a dense sparsest shift method to compute $\alpha_p$
4. Repeat Steps 1–3 enough times to recover $\alpha$
Example

**Unknown Polynomial in** \( \mathbb{Q}[x] \)

\[ f = (x - 3)^{107} - 485(x - 3)^{54} \]

1. Choose a prime \( p \) with \( p \in O(B^{O(1)}) \)

**Step 1**

\[ p = 11 \]
Example

Unknown Polynomial in $\mathbb{Q}[x]$

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

1. Choose a prime $p$ with $p \in O(B^{O(1)})$

2. Evaluate $f(0), f(1), \ldots, f(p - 1) \mod p$ to interpolate $f^{(p)}$

Step 2

$$f(0), f(1), \ldots, f(p - 1) \mod p = 9, 10, 9, 0, 0, 2, 5, 2, 5, 10, 3$$

$$f^{(p)} = x^7 + x^6 + 2x^5 + 9x^3 + 2x^2 + 8x + 9$$
Example

**Unknown Polynomial in \( \mathbb{Q}[x] \)**

\[
f = (x - 3)^{107} - 485(x - 3)^{54}
\]

1. Choose a prime \( p \) with \( p \in O(B^{O(1)}) \)
2. Evaluate \( f(0), f(1), \ldots, f(p - 1) \) \( \text{rem} \) \( p \) to interpolate \( f^{(p)} \)
3. Use a dense sparsest shift method to compute \( \alpha_p \)

**Step 3**

\[
f^{(p)} \equiv (x - 3)^7 + 10(x - 3)^4 \mod p
\]

\[\alpha_p = 3\]
**Example**

### Unknown Polynomial in $\mathbb{Q}[x]$  

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

1. Choose a prime $p$ with $p \in O(B^{O(1)})$  
2. Evaluate $f(0), f(1), \ldots, f(p - 1) \mod p$ to interpolate $f^{(p)}$  
3. Use a dense sparsest shift method to compute $\alpha_p$  
4. Repeat Steps 1–3 enough times to recover $\alpha$

### Step 4

$$\alpha_{11} = 3, \quad \alpha_{13} = 3, \quad \alpha_{17} = 3, \quad \ldots$$

$$\alpha = 3$$
Every step is guaranteed to succeed, except when $\alpha_p$ is not the unique sparsest shift of $f^{(p)}$.

**Theorem**

*The method succeeds whenever $\deg f^{(p)} \geq 2t$.***

Next we develop sufficient conditions on $p$ to avoid failure.
Sparsest shift of $f^{(p)}$ is not $\alpha_p$

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

$$p = 13$$

$$f^{(13)} = 9(x - 2)^1 + (x - 2)^2 + 3$$

$$\equiv (x - 4)^2 + 12$$

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

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

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

\[
p = 11
\]

\[
f^{(11)} = 4(x - 1)^9 + 2(x - 1)^1 + 7(x - 1)^9 + 9
\]

\[
= 2(x - 1) + 9
\]

\[
\equiv 2(x - 2)
\]

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

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

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

$p = 23$

\[
f^{(23)} = 0(x - 5)^{20} + 11(x - 5)^1 + 4
= 11(x - 5) + 4
≡ 11(x - 13)
\]

Condition: $p \nmid c_t$
Sufficient Conditions

**Definition**

\[
C = \max\{e_1, 1\} \cdot \prod_{i=2}^{t-1} e_i \cdot \prod_{i=1}^{t-1} (e_t - e_i) \cdot \prod_{i=0}^{2t-2} (e_t - i) \leq 2^{4B^2}
\]

**Sufficient Conditions for Success**

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

**Approach**

Choose primes \(p = qk + 1\), for distinct primes \(q\).

\(q \mid (p - 1)\), so \(q \nmid C \Rightarrow (p - 1) \nmid C\).
Definition

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

Theorem (Mikawa 2001)

There exists a constant $\mu$ such that, for all $n > \mu$, and for most of the integers $q \in \{n, n + 1, \ldots, 2n\}$, $S(q) < q^{1.89}$.
Prime Choosing Algorithm

1. \( q \leftarrow \text{next smallest prime} \)
2. If \( S(q) < q^{1.89} \), add \( S(q) \) to \( \mathcal{P} \)
3. Repeat until \(|\mathcal{P}|\) is sufficiently large.

\( \mathcal{P} \) must contain \( \Omega(B^2) \) “good” primes, so we only iterate \( O^{\sim}(B^2) \) times.
Step 3 (computing sparsest shift of $f^{(p)}$) dominates.

Cost of this step is $O(B^2 M(B^3 p^2))$ bit operations
(using deterministic algorithm from Giesbrecht et al. 2003)

Bit Complexity

$$O(B^3 \cdot M(B^{10.56} \log^{15.12} B) \cdot M(\log^2 B) = O^\sim(B^{13.56})$$

Can be faster by using probabilistic methods and/or heuristics.
Problem

Given $\alpha \in \mathbb{Q}$, we have a modular black box for the $t$-sparse polynomial $f(x + \alpha) \in \mathbb{Q}[x]$. 

Question

How to recover $f \in \mathbb{Q}[x]$ from $\alpha \in \mathbb{Q}$ and images $f^{(p)} \in \mathbb{Z}_p[x]$?

Example

\[
\begin{align*}
  f &= 12x^{46} + 3x^{29} + 9x^12 \\
  f^{(7)} &= 3x^6 + 3x^5 + 5x^4 \\
  f^{(11)} &= 3x^9 + x^6 + 9x^2
\end{align*}
\]

How can we match up the exponents?
## Summary of Sparsest Shift Computation Techniques

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Deterministic Algorithm</strong></td>
<td>Actual complexity only $O^\sim(B^{29})$ unless ERH is false.</td>
</tr>
<tr>
<td><strong>Probabilistic Algorithm</strong></td>
<td>Always correct and (provably) probably much faster.</td>
</tr>
<tr>
<td><strong>Heuristic</strong></td>
<td>Faster in practice and provably never wrong, but might not terminate</td>
</tr>
</tbody>
</table>
Interpolation

Once $\alpha$ is known, we can construct a modular black box for evaluating $f(x + \alpha)$.

Then use lacunary interpolation along the lines of Kaltofen, Lakshman, & Wiley (1990) and Avendaño, Krick, & Pacetti (2006).
Conclusions

- 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 $\mathbb{Q}[x]$?
- What about multivariate rational polynomials?