Faster Sparse Interpolation over Finite Fields and Complex Numbers

Mark Giesbrecht and Daniel S. Roche
Symbolic Computation Group, University of Waterloo

WATERLOO
CHERITON SCHOOL OF COMPUTER SCIENCE

The Problem

The basic **sparse interpolation** problem is as follows: Given a **black box** (i.e. way to evaluate) an unknown polynomial

$$f = c_1 x^{e_1} + c_2 x^{e_2} + \cdots + c_t x^{e_t},$$

determine the coefficients c_i and exponents e_i.

We are interested in two cases:

- Coefficients come from a large, unchosen finite field
- Coefficients are approximations to complex numbers

Remainder Black Box

A **remainder black box** takes a monic polynomial g and evaluates f rem g.

Example: unknown polynomial is

$$f = 5x^6 - 20x^{139} + 16x^{218} - 3x^{381}.$$

Given $g = x^{10} - 1$, the black box returns

$$f \text{ rem } g = -3x + 5x^6 + 16x^8 - 20x^9.$$

Observe: exponents reduced modulo 10.

Garg and Schost's Algorithm

Garg & Schost (TCS 2009): first polynomial-time algorithm for sparse interpolation over a large, unchosen finite field.

Overview: Given remainder black box for unknown

$$f = c_1 x^{e_1} + c_2 x^{e_2} + \cdots + c_t x^{e_t},$$

define the unknown integer polynomial

$$\Gamma(z) = (z - e_1)(z - e_2)\cdots(z - e_t) \in \mathbb{Z}[z].$$

For primes $p \in O(t^2 \log \deg f)$, evaluate f rem $x^p - 1$.
This gives us the set $\{e_1 \text{ rem } p, e_2 \text{ rem } p, \ldots e_t \text{ rem } p\}$, from which the coefficients of Γ mod p can be computed.

Repeating $O(t^2 \log d)$ times gives the coefficients of Γ, and we perform root finding over $\mathbb{Z}[z]$ to find the exponents e_i.

Diversification

- We call a polynomial with all coefficients distinct **diverse**.
- Diverse polynomials are easier to interpolate.
- We use randomization to create diversity.

Theorem. *If $q \gg t^2 \deg f$, $f \in \mathbb{F}_q[x]$, and $\alpha \in \mathbb{F}_q$ is chosen randomly, then $f(\alpha x)$ is probably diverse.*

Theorem. *If $f \in \mathbb{C}[x]$ has large coefficients and ζ is an order-$O(t^2)$ root of unity, $f(\zeta x)$ is probably diverse.*

Diversity in the latter case (approximate) means sufficiently separated coefficients.

Example over finite field $\mathbb{F}_{101}$

Let $f = 57 + 5x^{74} + 57x^{76} + 5x^{92} \in \mathbb{F}_{101}[x]$ be unknown. Note that f is *not* diverse.

Diversify. Randomly choose $\alpha \in \mathbb{F}_{101}$: $\alpha = 21$.
Also choose $p_1 \in O(t^2 \log \deg f)$: $p_1 = 11$, and evaluate

$$f(\alpha x) \text{ rem} (x^{11} - 1) = 57 + x^4 + 19x^8 + 15x^{10}.$$

This gives sparsity $t = 4$ and shows that $f(\alpha x)$ is diverse.

Further evaluations. Let $p_2 = 5$ and $p_3 = 7$. Evaluate

$$f(\alpha x) \text{ rem} (x^5 - 1) = 57 + 15x + x^2 + 19x^4$$
$$f(\alpha x) \text{ rem} (x^7 - 1) = 57 + x + 19x^4 + 15x^6.$$

Recover exponents. Because we know $p_1 p_2 p_3 > \deg f$, like terms are correlated **using the diverse coefficients**, and then exponents are found by Chinese remaindering:

$$e_1 = 0, \quad e_2 = 74, \quad e_3 = 76, \quad e_4 = 92.$$

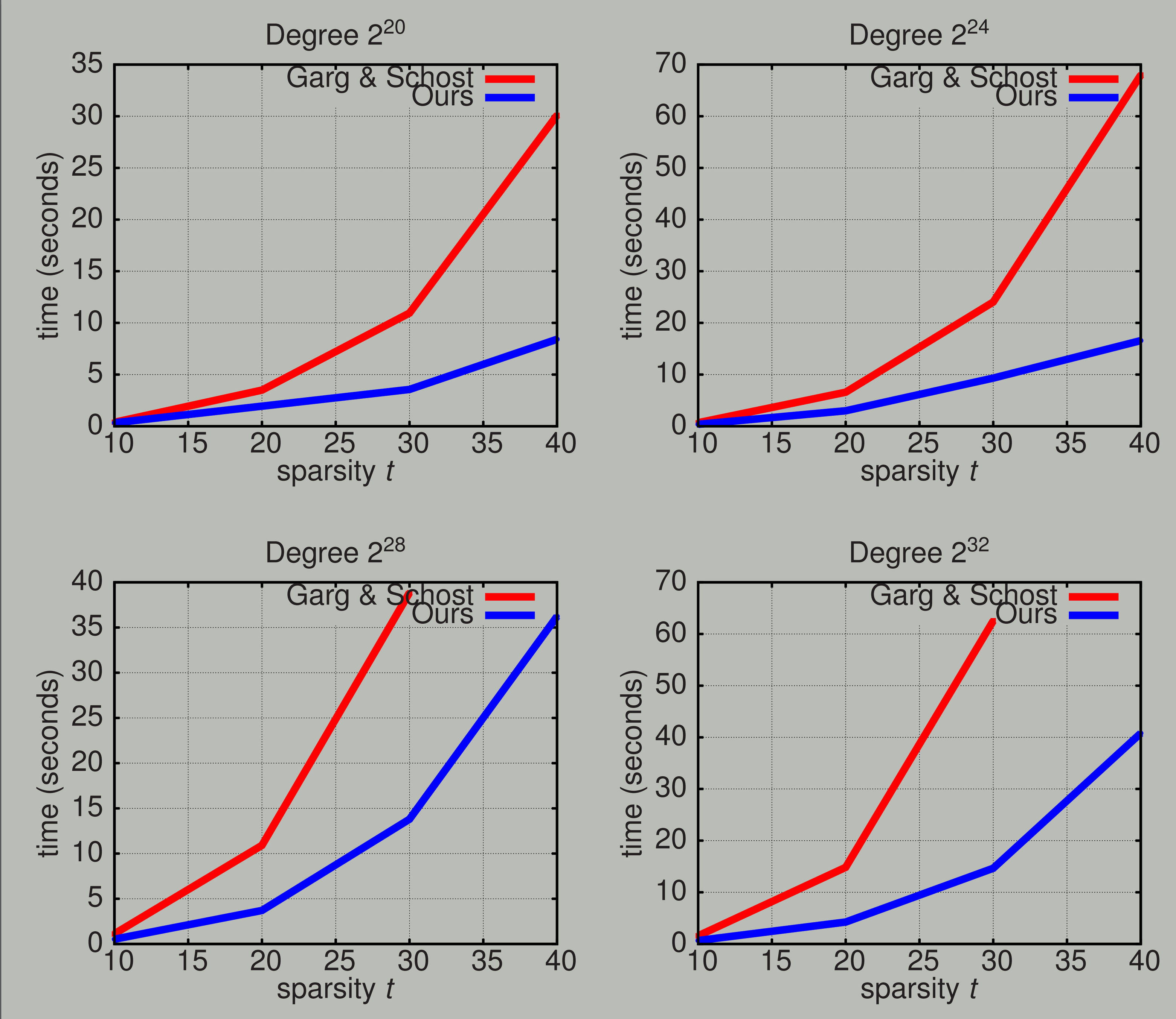
Recover coefficients. Once we know the exponents, the coefficients are determined from any modular evaluation.

Summary of results

Finite fields: Randomized cost is $\tilde{O}(t^2 \log^2 \deg f)$.

Approximate: In the same time, and with ϵ noise, we can compute a $g \in \mathbb{C}[x]$ such that $\|f - g\|_2 < \epsilon \|f\|_2$.

Finite field implementation experiments



Experimental stability in approximate algorithm

Noise	Mean Error	Median Error	Max Error
0	4.440 e−16	4.402 e−16	8.003 e−16
$\pm 10^{-12}$	1.113 e−14	1.119 e−14	1.179 e−14
$\pm 10^{-9}$	1.149 e−11	1.191 e−11	1.248 e−11
$\pm 10^{-6}$	1.145 e−8	1.149 e−8	1.281 e−8

Extending to multivariate

Now consider an unknown multivariate $f \in \mathsf{F}[x_1, \ldots, x_n]$.
We can perform sparse interpolation in one of two ways:

Kronecker substitution. Consider the polynomial

$$\hat{f} = f(y, y^d, y^{d^2}, \ldots, y^{d^{n-1}}).$$

If $d > \deg_{x_i} f$ for all i, then the terms of the *univariate* polynomial $\hat{f}$ correspond to those of f.

Zippel's method. Zippel's multivariate interpolation algorithm can be hybridized with our univariate algorithms. The method is randomized and works variable-by-variable, resulting in more univariate calls with lower degrees.