A Maple implementation of a parametric linear system solver using sparse rational function interpolation

Michael Monagan, Mantej Sohki, and Archit Srivastava.
Department of Mathematics, Simon Fraser University, BC, V5A 1S6, Canada

Abstract

Let $Ax = b$ be an n by n linear system over $\mathbb{Z}[y_1, y_2, ..., y_m]$, that is, $Ax = b$ is a parametric linear system in m parameters $y_1, y_2, ..., y_m$. In general, the solutions are rational functions in the parameters, that is, x_i is in $\mathbb{Q}(y_1, y_2, ..., y_m)$. If we let A_i be the matrix A with column i replaced by b, Cramer's rule says the solutions are given by

$$x_i = \frac{\det(A_i)}{\det(A)}.$$

An algorithm for computing $\det(A)$ and $\det(A_i)$ for $1 \le i \le n$, simultaneously, is the Lipson [5] fraction-free algorithm. This algorithm row reduces the augmented matrix $[A|b]$ to triangular form using the Bareiss-Edmonds [1, 3] fraction-free Gaussian elimination algorithm which obtains $\det(A)$, then it computes $\det(A_i)$ using a fraction-free back substitution. The final step is to simplify the fractions $\det(A_i)/\det(A)$ for $1 \le i \le n$ by computing and dividing out by $g_i = \gcd(\det(A_i), \det(A))$ to obtain $x_i = f_i(y_1, y_2, ..., y_m)/g_i(y_1, y_2, ..., y_m)$ for some $f_i, g_i \in \mathbb{Z}[y_1, y_2, \ldots, y_m]$. This approach does $O(n^3)$ polynomial multiplications, $O(n^3)$ exact divisions plus n gcd computations in the polynomial ring $\mathbb{Z}[y_1, y_2, \ldots, y_m]$. It suffers from expression swell as it creates intermediate polynomials which are larger than the determinants $\det(A)$ and $\det(A_i)$ which can be very large.

We have experimented with using the Kaltofen-Yang sparse rational function interpolation algorithm from [4] to interpolate the rational function solutions x_i from values of x_i. The Kaltofen-Yang algorithm interpolates the polynomials $\mu_i f_i$ and $\mu_i g_i$ for some fixed scalars μ_i using a sparse polynomial interpolation algorithm. To improve efficiency, we do this modulo a sequence of primes $p_1, p_2, p_3, \ldots$ and recover the rational coefficients of $\mu_i f_i$ and $\mu_i g_i$ in $\mathbb{Q}[y_1, y_2, ..., y_m]$ using Chinese remaindering and rational number reconstruction [7, 6]. For each prime $p \in \{p_1, p_2, \ldots\}$ we apply the Kaltofen-Yang algorithm. For the sparse polynomial interpolations we use the Ben-Or/Tiwari algorithm from [2] modulo the prime p. If $M(y_1, y_2, \ldots, y_m)$ is any monomial in f_i and g_i, this approach requires $p > M(2, 3, 5, \ldots, p_m)$ where p_m denotes the m'th prime.

The Kaltofen-Yang chooses many points $\alpha_j \in \mathbb{Z}_p^m$, solves the linear systems $A(\alpha_j)x = b(\alpha_j)$ for $x \in \mathbb{Z}_p^n$, and interpolates $y_1, y_2, ..., y_m$ in the numerators μf_i and denominators μg_i separately. Letting $\#f$ denote the number of terms of a polynomial f, the number of points needed by Kaltofen-Yang per prime is $O(DT)$ where

$$D = \max_{i=1}^{n}(\deg(f_i) + \deg(g_i) + 1) \quad \text{and} \quad T = \max_{i=1}^{n} \max(\#f_i, \#g_i).$$

This approach should be faster than the Lipson method when the h_i are non-trivial, equivalently, the size of the f_i and g_i are smaller than $\det(A_i)$ and $\det(A)$. This interpolation approach avoids the

expression swell in the Lipson fraction-free algorithm and avoids all arithmetic with polynomials in $\mathbb{Z}[y_1, y_2, ..., y_m]$.

We have implemented the algorithm in Maple. To get good performance, our Maple implementation makes use of Maple's foreign function; we have implemented several subalgorithms in C, including the Berlekamp Massey algorithm, univariate rational function interpolation in $\mathbb{Z}_p(x)$, and Zippel's transposed Vandermonde solver from [8]. Many subalgorithms in Maple are already implemented in C (e.g. polynomial root finding modulo a prime, linear system solving modulo a prime, and rational number reconstruction).

In the talk we will describe the Kaltofen-Yang algorithm and our implementation. We will demonstrate our code on a b-spline problem from computer graphics. It is a system of 21 linear equations in five parameters y_1, y_2, y_3, y_4, y_5. For this system $\det(A)$ has 1023 terms and the largest numerator f_i and denominator g_i has only 26 terms. We need only 973 points to interpolate all solutions x_i for $1 \le i \le 21$, that is, we solve 973 linear systems $A(\alpha_j)x = b(\alpha_j)$ for $1 \le j \le 973$ using ordinary Gaussian elimination over the field $\mathbb{Z}_p$ instead of solving one linear system $Ax = b$ over $\mathbb{Z}[y_1, y_2, y_3, y_4, y_5]$ using Lipson's fraction-free algorithm.

References

[1] Erwin H. Bareiss. Sylvester's identity and multistep integer-preserving Gaussian elimination. *Math. Comput.* **22**:565–578, 1968.

[2] Michael Ben-Or and Prasoon Tiwari. A deterministic algorithm for sparse multivariate polynomial interpolation. *Proceedings of STOC 1888*, pp. 301–309, ACM, 1988.

[3] Jack Edmonds. Systems of distinct representatives and linear algebra. *J. Res. Nat. Bur. Standards*, Sect. B, **71**:241–245, 1967.

[4] Erich Kaltofen and Zhengfeng Yang. On Exact and Approximate Interpolation of Sparse Rational Functions. *Proceedings of ISSAC 2007*, pp. 203–210, ACM, 2007.

[5] John D Lipson. Symbolic methods for the computer solution of linear equations with applications to flowgraphs. *Proceedings of Summer Institute on Symbolic Mathematical Computation*, pages 233—303, 1969.

[6] Michael Monagan. Maximal quotient rational reconstruction: an almost optimal algorithm for rational reconstruction. *Proceedings of ISSAC 2004*, pp. 243–249, ACM, 2004.

[7] Paul S. Wang, M. T. Guy, James H. Davenport. P-adic reconstruction of rational numbers. *SIGSAM Bulletin* **16**(2):2–3, ACM, 1982.

[8] Richard Zippel. Interpolating Polynomials from their Values. *J. Symbolic Computation* **9**(3):375–403, Elsevier, 1990.