Nullspace computation over rational function fields for symbolic summation

Burçin Eröcal
RISC
Johannes Kepler University
Linz, Austria, A-4040
burcin@erocal.org

Arne Storjohann
School of Computer Science
University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1
astorjoh@uwaterloo.ca

Introduction

Symbolic summation methods, either based on the WZ-Fasenmyer paradigm [2, 11, 10] or Karr's algorithm [6], reduce the given summation problem to finding vectors in the nullspace of a matrix over a rational function field with one or more variables. Direct methods to find such vectors use Gaussian elimination with heuristic pivoting strategies, but, due to intermediate expression swell, can fail to produce results for even moderate sized inputs. In this poster we focus on the case of one variable, and consider two related approaches that use homomorphic imaging and Chinese remaindering to compute a nullspace of a matrix over $\mathbb{Q}(x)$. We report on a preliminary implementation of these methods in the open source computer mathematics system Sage [8] and comment on their performance on actual input matrices obtained from the Mathematica implementation of Wegshaider's algorithm [10].

While conceptually simple, these approaches make essential use of many asymptotically fast methods for integers and polynomials, including not only multiplication, but also radix conversion, interpolation, rational function and number reconstruction, etc. The single problem of computing a nullspace over $\mathbb{Q}(x)$ thus provides a good test of, and should motivate the further development of, highly optimized libraries such as GMP [3], FFLAS [1], FLINT [4] and zn_poly [5].

Outline of approach

To use homomorphic imaging we need to ensure consistent images modulo the various primes. A straightforward preprocessing step involving Monte Carlo rank computation, and scaling of the rows to clear denominators, reduces the problem to computing the nullspace of full row rank matrix $[\,A\,|\,B\,] \in \mathbb{Z}[x]^{n\times(n+m)}$, where A is square nonsingular. A canonical nullspace basis is now given by

$$\left[\frac{sA^{-1}B}{-sI}\right] \in \mathbb{Q}[x]^{(n+m)\times m},$$

where the scaling polynomial $s \in \mathbb{Q}[x]$, a factor of $\det A$, is used to clear denominators (i.e., $\mathbb{Q}(x) \to \mathbb{Q}[x]$) from $A^{-1}B \in \mathbb{Q}(x)$. The pair $(s, sA^{-1}B)$ is computed modulo various word size primes p and the final result over $\mathbb{Q}[x]$ is recovered using Chinese remaindering and, if needed, rational number reconstruction. We implemented two approaches to compute a suitable $(s, sA^{-1}B)$.

An a priori degree bound for $\det s$ and $\det sA^{-1}B$ is nd, where d is a bound for the degrees of entries in A and B.

Nullspace via output sensitive x-adic lifting

We compute $(s, sA^{-1}B) \bmod p \in \mathbb{Z}_p[x]$ using x-adic lifting, with an output sensitive approach that performs lifting up to $\max(\deg s, \deg sA^{-1}B)$ instead of the a priori degree bound nd. Here, s will be a monic divisor, possibly proper, of $\det A$, and the final recovery of the result over $\mathbb{Q}[x]$ requires rational number reconstruction as well as Chinese remaindering. The lifting can be reduced to calling the FFLAS library to perform multiply-add operations, which are implemented efficiently using hardware floating point arithmetic. The asymptotic cost of computing each image is $O\tilde{}(n^3md)$ operations modulo p. Due to the structure of the input matrices, using an output sensitive approach both for the x-adic lifting and Chinese remaindering greatly improves the performance of this method. Our implementation instantly finds a solution to the system corresponding to $C_{7,2k}$ in [7], which is beyond the reach of the standard commands in Maple and Mathematica.

Nullspace via outer product adjoint formula

We compute $(\det A, (\det A)A^{-1}B) \bmod p \in \mathbb{Z}_p[x]$ using the outer product adjoint formula approach of [9]. The cost of computing each image is $O\tilde{}(n^3d + n^2md)$ operations modulo p. Even though the outer product method has better worst case complexity, and is considerably faster on random input, for the types of input matrices we encountered in the summation problem the direct x-adic algorithm gave better results. This is partly due to the fact that the outer product formula requires us to work with a preconditioned matrix whose adjoint has degrees $(n-1)d$, thus hiding the structure of the inputs and preventing effective use of an early termination strategy for the lifting.

References

[1] J.-G. Dumas, T. Gautier, and C. Pernet. Finite field linear algebra subroutines. In T. Mora, editor, *Proc. ISSAC '02*, pages 63–74. ACM Press, New York, 2002.

[2] M.C. Fasenmyer. *Some generalized hypergeometric polynomials.* PhD thesis, University of Michigan, November 1945.

[3] T. Granlund. The GNU multiple precision arithmetic library, 2004. Edition 4.1.4. `http://www.swox.com/gmp`.

[4] Bill Hart and David Harvey. Fast library for number theory. `http://www.flintlib.org/`.

[5] David Harvey. Faster polynomial multiplication via multipoint kronecker substitution. *Journal of Symbolic Computation*, 44(10):1502 – 1510, 2009.

[6] Michael Karr. Summation in finite terms. *J. ACM*, 28(2):305–350, 1981.

[7] F. Stan. On recurrences for Ising integrals. *Adv. in Appl. Math.*, 45(3):334–345, 2010.

[8] W. A. Stein et al. *Sage Mathematics Software.* The Sage Development Team. `http://www.sagemath.org`.

[9] A. Storjohann. On the complexity of inverting integer and polynomial matrices. Technical report, David R. Cheriton School of Computer Science, University of Waterloo, 2008.

[10] K. Wegschaider. Computer generated proofs of binomial multi-sum identities. Master's thesis, RISC, Johannes Kepler University, May 1997.

[11] H. S. Wilf and D. Zeilberger. An algorithmic proof theory for hypergeometric (ordinary and "q") multisum/integral identities. *Invent. Math.*, 108(3):575–633, 1992.