3 Approximate Polynomial Decomposition

Robert M. Corless, Mark W. Giesbrecht, David J. Jeffrey, Xianping Liu & Stephen M. Watt
University of Western Ontario

In this paper we establish a framework for the decomposition of approximate polynomials. We consider approximately known polynomials $f(z) \in \mathbb{C}[z]$ or $f(z) \in \mathbb{R}[z]$ and examine the problem of functional decomposition. That is, given f, we wish to compute polynomials g and h such that

$$(f + \Delta f)(z) = (g \circ h)(z) = g(h(z)),$$

where $\deg g < \deg f$, $\deg h < \deg f$, $\deg \Delta f \leq \deg f$ and Δf is "small" with respect to the 2-norm of the vector of coefficients. In practice if $\|f\|$ denotes the 2-norm of f, then we compute g and h such that $\|\Delta f\|$ is a local minimum with respect to variations in g and h.

This problem has been studied for exact polynomials and rational functions by several authors [1, 2, 5, 7, 9, 10]. There are several reasons why approximate polynomial decomposition interests us:

- Decomposition is a fundamental operation on polynomials. Posing a natural, well-defined interpretation of approximate polynomial decomposition and presenting an algorithm for its computation further advances the program to develop a full collection of symbolic-numeric algorithms for polynomials.
- Sometimes one knows *a priori* from the problem domain that polynomials should be compositions. This can occur when modelling a phenomenon which comprises a number of sequential algebraic steps, for example, the positions of a multiply articulated robot arm.
- The decomposed form of a polynomial can be substantially less expensive to evaluate than either an expanded or factored form. For example, a dense polynomial of degree n would take approximately $2n$ operations to evaluate in either expanded or factored form. A presentation as two composition factors, however, would take between $4\sqrt{n}$ and n arithmetic operations.

The main results of this paper are:

(a) an iterative method to compute a decomposition of a given approximate polynomial, given a starting point. The iteration scheme, which is linearly convergent, is analogous to quotient-divisor iteration for the approximate GCD problem [3]. Further, each iteration can be executed with $O(n \log^2 n)$ floating point operations.

(b) a theorem showing that a surprisingly good starting point is obtained from the initial step of the exact algorithm, except when the leading coefficient of f is too small.

(c) an experimental comparison of the method with Newton iteration.

(d) a prototype implementation that uses Aldor polynomial tools to set up the problems to be solved at each step of the iterations, and appropriate numerical linear algebra routines from the NAG library to carry out the solution efficiently.

Definitions and Design Choices

Notation. We write f to indicate the polynomial operator $f = z \to f_0 + f_1 z + \cdots + f_n z^n$, and its value at a as $f(a)$. We write the transpose of the vector of coefficients of f as $\boldsymbol{f}^t = [f_0, f_1, \ldots, f_n]$. By $\boldsymbol{f}^*$ we mean the conjugate transpose of $\boldsymbol{f}$. By signum(α) for $0 \neq \alpha \in \mathbb{C}$ we mean $\alpha/|\alpha|$. Henceforth let $\deg f = n$, $\deg g = m$ and $\deg h = d$. We write $[\, z^\ell \,]\,(p)$ for the coefficient of z^ℓ in p, following [4].

In this paper, the size of polynomials, and hence the distance between two polynomials, will be measured using the 2-norm. This norm can also be usefully expressed as a contour integral through Parseval's theorem (see [6]).

Lemma 3.1. *For a polynomial* $f(z) = \sum_{k=0}^{n} f_k z^k \in \mathbb{C}[z]$,

$$\begin{aligned} \|f\|^2 = \|f(z)\|^2 &= |f_0|^2 + |f_1|^2 + \cdots + |f_n|^2, && (1)\\ &= \frac{1}{2\pi}\int_0^{2\pi} f(e^{it})\overline{f}(e^{-it})\,dt && (2)\\ &= \frac{1}{2\pi i}\int_C f(z)\overline{f}(1/z)\frac{dz}{z} && (3)\\ &= [\, z^0 \,]\ f(z)\overline{f}(1/z)\,, && (4)\end{aligned}$$

where $z = e^{it}$ *parameterises* C*, the unit circle.*

This norm has the following advantages.

(a) It allows us to evaluate partial derivatives of the norm in terms of polynomial and series manipulations. These can be used to express a sequence of least squares problems, whose solutions usually converge to a minimum perturbation $||\Delta f|| = ||f - g \circ h||$. The derivatives can also be used for Newton's method.

(b) Minimising $\|\Delta f\|$ gives a near-Chebyshev minimum on the unit disk [8].

(c) It permits fast algorithms for the solution of subproblems at each iteration.

The expression of $\|f\|$ in the form (2) emphasises the importance of the size of the values of $f(z)$ on the unit disk. This highlights the need for the following assumptions regarding the formulation of the problem:

(a) The location of the origin has been chosen (thus making explicit an implied assumption in previous numerical polynomial algorithms),

(b) The scale of $|z|$ has been chosen.

In particular, we assume that the problem context precludes a change of variable by an affine transformation $z \to bz + a$.

References

[1] C. Alonso, J. Gutiérrez, and T. Recio. A rational function decomposition algorithm by near-separated polynomials. *J. Symb. Comp*, 19(6):527–544, 1995.

[2] D. R. Barton and R. Zippel. Polynomial decomposition algorithms. *J. Symb. Comp.*, 1:159–168, 1985.

[3] P. Chin, R. Corless, and G. Corliss. Optimisation strategies for the approximate gcd problem. In *Proc. ISSAC'98*, pages 228–235, Maui, Hawaii, 1998. ACM Press.

[4] R. Graham, D. Knuth, and O. Patashnik. *Concrete Mathematics.* Addison-Wesley, Reading, MA, USA, 1994.

[5] J. Gutiérrez, T. Recio, and C. Ruiz de Velasco. Polynomial decomposition algorithm of almost quadratic complexity. In *Proc. Applied algebra, algebraic algorithms and error-correcting codes*, volume 357 of *Lecture Notes in Comp. Sci.*, pages 471–475, Rome, 1988. Springer, Berlin.

[6] Thomas W. Körner. *Fourier Analysis.* Cambridge University Press, 1988.

[7] D. Kozen and S. Landau. Polynomial decomposition algorithms. *J. Symb. Comp.*, 7:445–456, 1989.

[8] T. Rivlin. *Chebyshev polynomials: from approximation theory to number theory.* Wiley, 1990.

[9] J. von zur Gathen. Functional decomposition of polynomials: the tame case. *J. Symb. Comp.*, 9:281–299, 1990.

[10] R. Zippel. Rational function decomposition. In *Proc. ISSAC'91*, pages 1–6. ACM Press, 1991.