Lifting/descending processes for polynomial zeros

Bernard Mourrain
SAGA, INRIA, 06902, Sophia Antipolis, Cedex, France
internet: mourrain@sophia.inria.fr

Victor Y. Pan*
Mathematics and Computer Science Department
Lehman College, CUNY, Bronx, NY 10468
internet: vpan@alpha.lehman.cuny.edu

Abstract

The recently proposed Chebyshev-like lifting map for the zeros of a univariate polynomial (cf. [BP96]) was motivated by its applications to splitting a univariate polynomial $p(z)$ numerically into factors, which is a major step of some most efficient algorithms for approximating polynomial zeros. We complement the Chebyshev-like lifting process by a descending process, decrease the estimated computational cost of performing the algorithm, demonstrate its correlation to Graeffe's lifting/descending process and generalize lifting from Graeffe's and Chebyshev-like maps to any fixed rational map of the zeros of the input polynomial.

Key words: polynomial zeros, numerical splitting a polynomial into factors, lifting polynomial zeros.

1991 Math. Subject Classification: 26C10, 30C15, 65H05, 68Q40.

1 Introduction and background.

Graeffe's map squares the zeros of a fixed univariate polynomial

$$p(z) = \sum_{i=0}^{n} p_i z^i = p_n \prod_{j=1}^{n} (z - z_j), \quad p_n \neq 0, \tag{1.1}$$

that is, transforms $p(z)$ into the polynomial

$$\overline{p}(x) = \overline{p}_n \prod_{j=1} (x - z_j^2) = (-1)^n p(z) p(-z)/p_n, \; x = z^2, \; \overline{p}_n = (-1)^n p_n. \tag{1.2}$$

Such a squaring, applied recursively as repeated squaring of polynomial zeros, is used in various algorithms for polynomial rootfinding (cf. e.g. [Sc82], [P96], [P97]). Actually, "Graeffe's map" was first discovered by Dandelin and soon thereafter independently by

*Supported by NSF Grants CCR 9625344 and 9732206 and PSC-CUNY Award 669363

Lobachevsky [H70]. The transformation (1.2) is simple (it essentially amounts to polynomial multiplication and costs $O(n\log n)$ operations). Hereafter, we will write "ops" as our abbreviation for "arithmetic operations".

In [BP96] map (1.2) was extended to the transformation of $p(z)$ into the polynomial

$$\hat{p}(x)=\hat{p}_n\prod_j(x-(z_j+z_j^{-1})/2)=p(z)p(z^{-1})),\ x=(z+z^{-1})/2,\ \hat{p}_n=p_0p_n, \tag{1.3}$$

with the zeros $(z_j+z_j^{-1})/2$, provided that $p_0p_n\neq 0$, and some applications were shown to splitting $p(z)$ numerically into factors, which is the basic step of some of the most efficient known polynomial rootfinders [P95], [P96]. In [BP96], this transformation was called *Chebyshev-like lifting* and the estimate $O(n\log^2 n)$ ops was shown for performing it.

In this paper we show that the actual cost is $O(n\log n)$ (cf. sections 2 and 3), complement Chebyshev-like lifting by the converse descending process (section 2), demonstrate correlation between Chebyshev-like and Graeffe's maps (section 3), and generalize Graeffe's and Chebychev-like maps to computing a polynomial $\tilde{p}_r(x)=p_n\prod_{j=1}^n(x-r(z_j))$ for any fixed rational function $r(z)$ (section 4).

Acknowledegements. We are greateful to the referee for helpful comments.

2 Chebyshev-like lifting/descending maps.

The Chebyshev-like lifting algorithm was introduced in [BP96] as means of the transition from the polynomial $p(z)$ of (1.1) with the zeros z_j to the polynomial $\hat{p}(x)$ of (1.3), with the zeros $\frac{z_j+z_j^{-1}}{2}$. For the sake of completeness of our presentation, we will next recall this algorithm, based on the equations of (1.3), and we will refine its computational cost estimate, versus the pessimistic one claimed in [BP96].

Algorithm 2.1, *Chebyshev-like lifting.*

Input. *Degree n and the coefficients $p_0,p_1,\dots,p_n$ of the polynomial $p(z)$ of (1.1) having zeros $z_1,\dots,z_n$, where $p_0p_n\neq 0$.*

Output. *The coefficients $\hat{p}_0,\hat{p}_1,\dots,\hat{p}_n$ of the polynomial $\hat{p}(x)$ of (1.3), with the zeros $\frac{z_j+z_j^{-1}}{2}$, $j=1,\dots,n$.*

Computations.

1. *Compute the values $\alpha_i=p(\omega^i)$, $i=0,\dots,2n+1$, where $\omega=\exp(\frac{2\pi\sqrt{-1}}{2n+2})$ is a primitive $(2n+2)$-nd root of 1.*

2. *Compute the values $\beta_i=\alpha_i\alpha_{2n+2-i}$, $i=0,\dots,n$, which the function $p(z)p(z^{-1})$ takes on at the $(2n+2)$-nd roots of 1. These values coincide with the values that the polynomial $\hat{p}(x)$ takes on at the (Chebyshev-like) points $x_i=(\omega^i+\omega^{-i})/2=\cos(\pi i/(n+1))$, $i=0,\dots,n$.*

3. *Recover the coefficients of $\hat{p}(x)$ by interpolating to the n-th degree polynomial having the values of β_i at the points $\cos(\pi i/(n+1))$, $i=0,\dots,n$.*

Correctness of this algorithm immediately follows from the observations that $\hat{p}(x)$ is a polynomial of a degree at most n and that the values of $p(z)p(z^{-1})$ at the $(2n+2)$-nd roots of 1 coincide with the values of the polynomial $\hat{p}(x)$ at the points $x_i=(\omega^i+$

$\omega^{-1})/2 = \cos(\pi i/(n+1))$, $i = 0, \dots, n$. Algorithm 2.1 belongs to the class of the evaluation-interpolation algorithms (cf. [BP94], ch. 1). It requires $O(n \log n)$, $n+1$, and $O(n \log n)$ ops at its stages 1, 2, and 3, respectively.

Stage 1 is handled by FFT, and the known algorithms of [CHQZ87], [For95], and [P98] for Chebyshev-like interpolation in $O(n \log n)$ ops can be used at stage 3; [BP96] relied on the pessimistic cost bound, $O(n \log^2 n)$, and extended it to the Chebyshev-like lifting algorithm. Now, with the refinement of the Chebyshev interpolation cost, the estimated overall cost of Chebyshev-like lifting immediatley goes down to $O(n \log n)$, versus the one claimed in [BP96], and in the next section, we will describe an alternative algorithm that yields the some goal by studying the correlation between Chebyshev-like and Graeffe's processes. Thus, in various ways we arrive at the next result:

Proposition 2.1 *$O(n \log n)$ ops suffice to perform Chebyshev-like lifting algorithm 2.1.*

Recursive application of algorithm 2.1 enables us to isolate from each other two groups of the zeros of a polynomial. Namely, such an application moves towards 1 the zeros of $p(z)$ having positive real parts and moves towards -1 the zeros of $p(z)$ having negative real parts. When the isolation is achieved, we will apply the known splitting algorithms of [BP96], [C96] or [Sc82], in order to split the resulting polynomial into two factors whose zeros are separated by the imaginary coordinate line L. We will use such a factorization as a springboard in order to yield a similar factorization of the original polynomial $p(z)$, that is, to split it over the line L. To simplify the notation, let us assume that already the polynomial $\hat{p}(x)$ has been split into two factors,

$$\hat{p}(x) = \hat{F}(x)\hat{G}(x), \tag{2.1}$$

where all the zeros of $\hat{F}(x)$ [respectively, $\hat{G}(x)$] have positive (respectively, negative) real parts. Then we will split $p(z)$ into two factors as follows:

Algorithm 2.2, *Chebyshev-like descending.*

Input. *Polynomials $p(z)$ of (1.1), $\hat{F}(x)$ and $\hat{G}(x)$of (2.1), of degrees n, k, and $n-k$, respectively.*

Output. *Two polynomials, $F(z)$ of degree k and $G(z)$ of degree $n-k$, satisfying $p(z) = F(z)G(z)$ where the real parts of all the zeros of $F(z)$ are positive and of all the zeros of $G(z)$ are negative.*

Computation. *Compute and output $F(z)$ being the greatest common divisor (gcd) of $F^*(z) = z^k \hat{F}(\frac{z+z^{-1}}{2})$ and $p(z)$. Then compute and output $G(z) = p(z)/F(z)$, so that $G(z) = \gcd(p(z), G^*(z))$, $G^*(z) = z^{n-k}\hat{G}(\frac{z+z^{-1}}{2})$.*

To show correctness of the algorithm, recall that $\hat{F}(\frac{z+z^{-1}}{2})\hat{G}(\frac{z+z^{-1}}{2}) = \hat{F}(x)\hat{G}(x) = \hat{p}(x) = p(z)p(z^{-1})$; therefore, $z^k\hat{F}(\frac{z+z^{-1}}{2})(z^{n-k}\hat{G}(\frac{z+z^{-1}}{2})) = F^*(z)G^*(z)$. It remains to observe that the real parts of all the zeros of the polynomials $\hat{F}(x)$ and $F^*(z)$ are positive, whereas the real parts of all the zeros of $\hat{G}(x)$ and $G^*(z)$ are negative. □

We may obtain the polynomial $F(z)$ by computing the $(k, n-k)$ entry $(F(z), G^*(z)/G(z))$ of the Padé approximation table for the polynomial $(p(z)/G^*(z)) \bmod z^{n+1}$.

Computation of $F(z)$, both as the gcd and from the entry of the Padé table, can be done in $O(n \log^2 n)$ ops, by means of the Euclidean algorithm [BGY80]. The computation of $F(z)$ from the Padé table can be also reduced to solving a Toeplitz linear system of k or $n-k$ equations; the solution of such a system costs $O(k \log^2 k)$ or $O((n-k)\log^2(n-k))$ ops, respectively [BGY80].

Remark 2.1 *It is realistic to assume that the coefficients of the factors* $\hat{F}(x)$ *and* $\hat{G}(x)$ *of (2.1) are known with a high but finite precision, which implies some small but nonzero perturbation of the input* $(p(z)/G(z)) \bmod z^{n+1}$ *of the Padé approximation problem. How would such an input perturbation affect the output polynomial* $F(z)$*? Let* $\Delta(z)$ *and* $f(x)$ *be the perturbations of the input and output polynomials, respectively, and let* $||\sum_i u_i x^i|| = \sum_i |u_i|$ *be the* L_1*-norm of a polynomial. It is proved in [P96], Fact 12.1, that* $||f(z)|| \leq ||\Delta(z)||(2+\frac{1}{\phi-1})^{C_1 n}$ *provided that* $||\Delta(z)C_0|| \leq (2+1/\phi)^{-C_0 n}$, C_0 *and* C_1 *are two fixed constants, and* ϕ *is the isolation ratio of the maximal zero-free annulus on the complex plane* $\{w\}$ *that separates the zeros of* $F(\frac{1-w}{1+w})$ *from those of* $G(\frac{1-w}{1+w})$*. (The isolation ratio is defined as the ratio of the radii of the two boundary circles of the annulus (cf. [P95], [P96]).) An interesting open question is how to extend this result to estimate the output errors of such a computation of Padé entries where they are computed numerically, with rounding to a fixed finite precision.*

3 Correlation between Chebyshev-like and Graeffe's maps.

Chebyshev-like lifting of (1.3) is quite similar to Graeffe's lifting of (1.2). Let us formalize this similarity. Recall that the map (1.2) squares the zeros z_j of $p(z)$, whereas the map (1.3) transforms z_j into $(z_j + z_j^{-1})/2$, $j = 1, \ldots, n$. Let us map z into a variable w (not to be mixed with ω denoting a root of 1) according to the next map:

$$z = \frac{1-w}{1+w}, \quad w = \frac{1-z}{1+z}, \tag{3.1}$$

and let us similarly map x into y:

$$x = \frac{1+y}{1-y}, \quad y = \frac{x-1}{x+1}. \tag{3.2}$$

Then the map of z_j into z_j^2 can be obtained as a composition of the three maps, (3.1), (3.2), and $x = \frac{w+w^{-1}}{2}$. Namely, we may write

$$x_j = \frac{1+y_j}{1-y_j},$$

$$w_j = \frac{1-z_j}{1+z_j},$$

$$x_j = \frac{w_j + w_j^{-1}}{2} = (\frac{1-z_j}{1+z_j} + \frac{1+z_j}{1-z_j})/2 = \frac{1+z_j^2}{1-z_j^2},$$

which implies that $y_j = z_j^2$.

Likewise, the map of z_j into $(z_j + z_j^{-1})/2$ can be obtained as a composition of the three maps, (3.1), (3.2), and $y = w^2$. Indeed, we write

$$x_j = \frac{1+y_j}{1-y_j}, \; y_j = w_j^2, \; z_j = \frac{1-w_j}{1+w_j}$$

and similarly deduce that $x_j = \frac{z_j + z_j^{-1}}{2}$.

Let us rewrite the maps (3.1) and (3.2) as follows:

$$z = \frac{1-w}{1+w} = \frac{2}{1+w} - 1, \; w = \frac{1-z}{1+z} = \frac{2}{1+z} - 1,$$

$$x = \frac{1+y}{1-y} = \frac{2}{1-y} - 1, \; z = \frac{x-1}{x+1} = 1 - \frac{2}{x+1},$$

reducing them to scaling, shifts and inversion of the variable. For the respective transformations of a fixed polynomial $p(z)$, (implied by scaling, shift and inversion of the variable), we need $O(n)$, $O(n \log n)$ (cf. e. g. [BP94], page 15), and 0 ops, respectively (the inversion of the variable amounts to the reversion of the order of the polynomial coefficients). Squaring of the polynomial zeros is supported by (1.2), at the cost of performing $O(n \log n)$ ops. This enables us to achieve lifting (1.3) in $O(n \log n)$ ops, which is an alternative derivation of proposition 2.1.

4 Generalization of lifting process.

Graeffe's and Chebyshev-like maps can be generalized to define a polynomial $\tilde{p}(x) := \tilde{p}_r(x) := p_n \prod_{j=1}^{n}(x - r(z_j))$ of degree n with the zeros $x_j = r(z_j)$, $j = 1, \ldots, n$, for any fixed rational function $x = r(z) = \nu(z)/\delta(z)$, $\nu(z)$ and $\delta(z)$ being two given relatively prime polynomials, $\delta(z)$ is monic, so that $r(z)$ is a polynomial if and only if $\delta(z) = 1$. In particular $\tilde{p}_r(x)$ turns into $(-1)^n \overline{p}(x)$ of (1.2) for $r(z) = z^2$ and into $p_0 \hat{p}(x)$ of (1.3) for $r(z) = \frac{z+z^{-1}}{2}$. We will specify two expressions for $\tilde{p}(x) = \tilde{p}_r(x)$ via $p(z)$ and $r(z)$:

$$\tilde{p}(x) = R(p(z), \delta(z)x - \nu(z)), \tag{4.1}$$

$$\tilde{p}(x) = \det(x\,I - r(F_p)). \tag{4.2}$$

Here, $R(u(z), v(z))$ is the resultant of two polynomials $u(z)$ and $v(z)$ (cf. e.g. [BP94], page 149, for the definition of resultants), whereas F_p is the Frobenius (companion) matrix of the monic polynomial $p(z)/p_n$, that is,

$$F_p = \begin{pmatrix} 0 & 1 & & 0 \\ 0 & & \ddots & 0 \\ & & \cdots & \\ 0 & & & 1 \\ -p'_0 & & \cdots & -p'_{n-1} \end{pmatrix}, \quad p'_i = p_i/p_n, \quad i = 0, 1, \ldots, n-1.$$

It is immediately verifed that in both cases (defined by (4.1) and (4.2)) we have $\tilde{p}(x) = \prod_{j=1}^{n}(x - r(z_j))$.

We may evaluate the (resultant) polynomial $\tilde{p}(x)$ of (4.1) at the points $x_i = a\omega^i$, $i = 0, 1, \ldots, n$, where a is any fixed (complex) scalar and ω is a primitive 2^h-th root of 1, for $h = \lceil \log_2(n+1) \rceil$. Then we may compute the coefficients of $\tilde{p}(x)$ by applying the inverse of the FFT. The cost of the entire computation is dominated by the cost $O((d+n)n \log^2(d+n))$ of the 2^h evaluations of the resultant $\tilde{p}(x)$, where

$$d = \max\{\deg(\nu(z)), \deg(\delta(z))\}. \tag{4.3}$$

An alternative algorithm relies on (4.2). We first compute the matrix $r(F_p)$ (which belongs to the matrix algebra $\mathcal{A}_p$ generated by the matrix F_p) and then compute its characteristic polynomial, $\tilde{p}(x)$ of (4.2). An addition, a subtraction and a multiplication in $\mathcal{A}_p$

require $O(n \log n)$ ops, the inversion of a matrix in $\mathcal{A}_p$ takes $O(n \log^2 n)$ ops [C96]. Therefore, the computation of the matrix $r(F_p)$ involves $O((d+\mu \log n)n \log n)$ ops, for d of (4.3), where $\mu = 0$ if $\delta(z) = 1$, $\mu(z) = 1$ otherwise.

Since every matrix of $\mathcal{A}_p$ has a displacement rank at most 2 [C96], the computation of the characteristic polynomial of such a matrix can be performed with $O(n^2 \log n)$ ops (cf. [P92] or [BP94], pages 189–190). In particular, this cost estimate applies to the matrix $r(F_p) \in \mathcal{A}_p$, so that $O((n+d)n \log n)$ ops suffice for computing the coefficients of $\tilde{p}(z)$ based on (4.2).

The descending process, from the factorization of $\tilde{p}(z)$ of (4.1), (4.2) to the factorization of $p(z)$ of (1.1), is more involved in the case of the general rational function $r(z)$ than the descending process of algorithm 2.2, from factorization of $\hat{p}(x)$ of (2.1) to the one of $p(z)$. There is an important special case where d is small relative to n and all the n zeros $x_j = r(z_j)$ of $\tilde{p}(x)$ have been approximated. In this case the descending is much simpler because we have the n additional polynomial equations in z_j, that is, $\delta(z_j)x_j = \nu(z_j)$ for $j = 1, \ldots, n$. For each j, such an equation has at most d roots $z_j^{(i)}$, $i = 1, \ldots, d_j$, $d_j \leq d$, and we may test and/or refine them as the candidates for the approximation of the zeros of $p(z)$, by using the known techniques for multipoint polynomial evaluation and polynomial rootfinding (cf. [BP94], [P97], [BP,a]). This direction leads us to various interesting open problems of error analysis, numerical stability and the trade-off between stability and complexity of numerical computations. We postpone the study of this topic to a subsequent paper.

References

[BGY80] R. P. Brent, F. G. Gustavson, D. Y. Y. Yun, Fast Solution of Toeplitz Systems of Equations and Computation of Padé Approximations, *Journal of Algorithms*, 1, pp. 259–295, 1980.

[BP94] D. Bini, V. Y. Pan, *Polynomial and Matrices Computations, vol. 1: Fundamental Algorithms,* Birkhäuser, Boston, 1994.

[BP,a] D. Bini, V. Y. Pan, *Polynomial and Matrix Computations, vol. 2: Selected Topics,* Birkhäuser, Boston, to appear.

[BP96] D. Bini, V. Y. Pan, Graeffe's, Chebyshev-like, and Cardinal's Processes for Splitting a Polynomial into Factors, *Journal of Complexity*, 12, pp. 492–511, 1996.

[C96] J. P. Cardinal, On Two Iterative Methods for Approximating the Roots of a Polynomial, *Proc. of Workshop, Math. of Numerical Analysis: Real Number Algorithms*, (Park City, Utah, 1995), J. Renegar, M. Shub, and S. Smale editors, vol. 32 of *Lecture Notes in Applied Math.*, pp. 165–188, Amer. Math. Society Press, Providence, R.I. 1996.

[CHQZ87] C. Canuto, M. Y. Hussaini, A. Quateroni, T. A. Zang, *Spectral Methods for Fluid Dynamics*, Springer, 1987.

[For95] B. Fornberg. *A Practical Guide to Pseudospectral Methods.* Cambridge University Press, 1995.

[H70] A.S. Householder, *The Numerical Treatment of a Single Nonliear Equation*, McGraw-Hill, New York, 1970.

[P89] V. Y. Pan, Fast Evaluation and Interpolation at the Chebyshev Set of Points, *Applied Math. Letters*, 2, 3, pp. 255–258, 1989.

[P92] V. Y. Pan, Parametrization of Newton's Iteration for Computations with Structured Matrices and Applications, *Computers & Mathematics* (with Applications), 24, 3, pp. 61–75, 1992.

[P95] V. Y. Pan, Optimal (up to Polylog Factor) Sequential and Parallel Algorithms for Approximating Polynomial Zeros, *in Proc. 27th Annual ACM Symposium on the Theory of Computing*, pp. 741–750, ACM Press, New York, 1995.

[P96] V. Y. Pan, Optimal and Nearly Optimal Algorithms for Approximating Polynomial Zeros, *Computers and Math. (with Applications)*, 31, 12, pp. 97–138, 1996.

[P97] V. Y. Pan, Solving a Polynomial Equation: Some History and Recent Progress, *SIAM Review,* 30, 2, pp. 187-220, 1997.

[P98] V. Y. Pan, New Fast Algorithms for Polynomial Interpolation and Evaluation on the Chebyshev Node Set, *Computers & Math. (with Applications)*, 35, 3, 125–129, 1998.

[Sc82] A. Schönhage, The Fundamental Theorem of Algebra in Terms of Computational Complexity, Manuscript, *Dept. of Math., Univ. of Tübingen*, Tübingen, Germany, 1982.