An International Journal
computers & mathematics with applications

PERGAMON Computers and Mathematics with Applications 41 (2001) 1559–1560
www.elsevier.nl/locate/camwa

A New Proximity Test for Polynomial Zeros

V. Y. PAN
Mathematics and Computer Science Department
Lehman College, CUNY, Bronx, NY 10468, U.S.A.
vpan@alpha.lehman.cuny.edu

(Received June 2000; accepted July 2000)

Abstract—We combine some known techniques and results of Turan and Schönhage to improve substantially numerical performance of the computation of the minimum and the maximum distances from a fixed complex point to roots (zeros) of a fixed univariate polynomial.

Keywords—Polynomial roots, Proximity test.

Given the coefficients of a polynomial

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

we seek the value

$$r = \max_j |z_j|. \tag{2}$$

The solution of the same problem applied to the reverse polynomial $r(x) = x^n p(1/x)$ produces $\min_j |z_j|$. By shifting the origin to a fixed complex point X, that is, by solving the same problem for the polynomials $q(y) = r(y + X)$ or $s(y) = p(y + X)$, we may compute $\min_j |X - z_j|$ (this is called *proximity test* at X) or $\max_j |X - z_j|$. Such tasks are important in polynomial computations, in particular, for rootfinding [1].

We propose the following substantial improvement of the numerical performance of the known algorithms.

ALGORITHM 1.

Input: the coefficients of a polynomial $p(x)$ of (1) and $\epsilon > 0$.

Output: a real r^* satisfying

$$r^* \leq r \leq 5(1 + \epsilon) r^*, \tag{3}$$

for r of (2).

Computation.

1. Compute $t = \max_{k \geq 1} |p_{n-k}/p_n|^{1/k}$.

Supported by NSF Grant CCR 9732206 and PSC-CUNY Award 669363.

0898-1221/01/$ - see front matter Typeset by AMS-TeX
PII: S0898-1221(01)00120-1

2. Compute the coefficients of the polynomial
$$q(x) = p(4tx) = (4t)^n p_n \prod_j (x - y_j), \qquad y_j = \frac{z_j}{(4t)}, \qquad j = 1, \dots, n.$$
3. Compute $\delta = \epsilon/(40(1+\epsilon))$ and $M = 2^l$, $l = \lceil \log_2 \log_2(n/\epsilon) \rceil$.
4. Compute the values
$$s_k^* = \frac{1}{M} \sum_{m=0}^{M-1} \omega^{m(k+1)} \frac{q'(\omega^m)}{q(\omega^m)}, \qquad m = 1, \dots, M-2,$$

where $\omega = \exp(2\pi\sqrt{-1}/M)$ is the M^{th} root of 1.

5. Compute
$$r_1^* = \max_{k=1,\dots,n} \left(\frac{(|s_k^*| - \delta)}{n} \right)^{1/k}, \qquad r^* = \frac{r_1^*}{(4t)},$$

and output r^*.

Let us deduce (3), that is, prove correctness of the algorithm. Recall that $t/n \le r \le 2t$ [2] and deduce that
$$\frac{1}{(4n)} \le r_1 = \max_j |y_j| \le \frac{1}{2}. \tag{4}$$
By applying a bound of [3], we obtain that
$$|s_k^* - s_k| \le \frac{n2^{-M-k}}{(1 - 2^{-M})} \le n2^{-M} \le \epsilon, \tag{5}$$
for $k = 1, \dots, n$, where $s_k = \sum_{j-1}^n y_j^k$. Clearly, $|s_k^*| - \delta \le |s_k| \le r_1^k n$ for all k, and therefore, $r_1^* \le r_1$, and $r^* \le r$ (cf. (3)). It remains to prove that $r \le 5(1+\epsilon)r^*$.

By Turan's theorem [4], we have
$$r_1 \le 5s, \qquad s = \max_{k=1,\dots,n} \left(\frac{|s_k|}{n} \right)^{1/k}.$$
Let $s^h = |s_h|/n$ for some $h \ge 1$. Then we have
$$r_1^h \le 5^h \frac{|s_h|}{n} \le 5^h \frac{(|s_h^*| + \delta)}{n} \le 5^h \frac{((r_1^*)^h n + 2\delta)}{n} = 5^h \left((r_1^*)^h + \frac{2\delta}{n} \right).$$
Therefore,
$$r_1 \le 5 \left(r_1^* + \frac{2\delta}{n} \right).$$
Substitute $r_1 \ge 1/(4n)$ (cf. (4)) and obtain that
$$r_1 \le 5r_1^* + 40\delta r_1, \qquad r_1 \le 5r_1^*(1 - 40\delta) \le 5(1+\epsilon)r^*.$$
This completes the proof of (3).

The algorithm requires $O(n)$ flops except for Stage 4, which is reduced to performing three discrete Fourier transforms at the M^{th} roots of 1, that is, to $O(M \log M)$ flops, due to FFT. To achieve $\epsilon = O(1/n^c)$ for any fixed c, it is sufficient to choose $M = O(\log n)$. Finally, by applying s Graeffe's root squaring steps, we may shift from $p(x)$ to the polynomial $p_n \prod_j (x - z_j^S)$, where $S = 2^s$. Application of Algorithm 1 to this polynomial enables us to replace the factor of 5 in (3) by $5^{1/S}$, $S = 2^s$.

REFERENCES

1. V.Y. Pan, Approximating complex polynomial zeros: Modified quadtree (Weyl's) construction and improved Newton's iteration, *J. of Complexity* **16** (1), 213–264 (2000).
2. P. Henrici, *Applied and Computational Complex Analysis*, Wiley, New York, (1974).
3. A. Schönhage, The fundamental theorem of algebra in terms of computational complexity, Math. Dept., University of Tübingen, Tübingen, Germany, (1982).
4. P. Turan, Power sum method and approximative solution of algebraic equations, *Math. Comp.* **29**, 311–318 (1975).