

Reprinted from IEEE TRANSACTIONS
ON COMPUTERS
Volume C-19, Number 1, January, 1970
pp. 79-80

COPYRIGHT © 1970—THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS, INC.
PRINTED IN THE U.S.A.

A Simple Convergent Algorithm for Rapid Solution of Polynomial Equations

Abstract—Extensions to a straightforward, always convergent method for solving polynomial equations given in a previous paper are considered. The extensions consist of additional simple calculations and logic instructions which considerably improve convergence rate for the cases when multiple roots exist or when roots are close together. It is believed that in terms of simplicity and convergence properties, the approach is more efficient than presently available methods.

Index Terms—Polynomial equations, roots of polynomial equations, zeros of polynomials.

This correspondence considers an extension to the always convergent methods for solving polynomial equations given in [1].¹ The extension consists of but a few simple calculations and logic instructions. These extra calculations are not used when convergence is rapid but are used to accelerate convergence if the convergence is

Manuscript received August 22, 1968. This work was supported by Australian Research Grants Committee.

¹[1] J. B. Moore, "A convergent algorithm for solving polynomial equations," *J. ACM*, vol. 14, pp. 311-315, April 1967.

slow due to the existence of multiple roots or clusters of roots close together.

The approach used in [1] is now briefly reviewed. Consider the entire function $f(z)$ of the complex variable $z = x + iy$ with real part $u(x, y)$ and imaginary part $v(x, y)$. In the neighborhood of a simple zero of $f(z)$, the equations

$$\Delta x \approx \frac{-u \frac{\partial u}{\partial x} - v \frac{\partial v}{\partial x}}{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2}; \quad \Delta y \approx \frac{u \frac{\partial v}{\partial x} - v \frac{\partial u}{\partial x}}{\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2} \quad (1)$$

may be used to calculate approximately the distances Δx and Δy in the x and y directions from a point in the neighborhood of $f(z)$ to the zero of $f(z)$. Successive application of (1) in the region of a simple zero of $f(z)$ for the case when $u, v, \partial u/\partial x$, and $\partial v/\partial x$ are readily calculated gives rapid convergence to the zero location.

It is observed that the direction of each iteration is

$$\frac{\Delta y}{\Delta x} = \frac{-u \frac{\partial u}{\partial x} - v \frac{\partial v}{\partial x}}{u \frac{\partial v}{\partial x} - v \frac{\partial u}{\partial x}} = \frac{\frac{\partial}{\partial x} |f(z)|}{\frac{\partial}{\partial y} |f(z)|}$$

which is in fact the direction of steepest descent of the function $|f(z)|$. It is further observed that $|f(z)|$ satisfies the following properties:

- 1) $|f(z)|$ is nonnegative;
- 2) the derivatives $\partial/\partial x |f(z)|$ and $\partial/\partial y |f(z)|$ exist;
- 3) the zeros of $|f(z)|$ are located at the zeros of $f(z)$;
- 4) these zeros are the only minima of $|f(z)|$.

(The fourth property is readily proved using the maximum modulus theorem.) It is concluded that if an iterative process using (1) is adopted even when the initial approximation is not in the region of a zero of $f(z)$, the convergence to a zero location of $f(z)$ may be guaranteed simply by monitoring $f(z)$ at each iteration and then proceeding as follows. If $|f(z)|$ is reduced, proceed with a further application of (1); if $|f(z)|$ is not reduced, reduce the step size given by (1) (but not changing the step direction) until $|f(z)|$ is decreased, then proceed with a further application of (1).

Extension: We first observe that when any of the successive approximations are in the vicinity of multiple roots (saddle points of $|f(z)|$) the convergence rate of the method of [1] can be considerably improved by increasing (decreasing) the step size taken as calculated using (1). One simple way of adjusting the step size systematically is to multiply Δx and Δy by a scale factored with the following properties. The factor S is initially unity and is also unity when converging towards a simple zero of the polynomial. It is a fraction when the zero approximation is in a saddle point region of $|f(z)|$, and is 2, 3, 4, etc. when converging towards a multiple zero of $f(z)$. (The actual rule adopted for the modification of S is as follows. The factor S is reduced to one-quarter of its magnitude if $|f(z)|$ is increased at any iteration, and this is repeated until $|f(z)|$ is decreased in magnitude. For the case when $|f(z)|$ is not reduced by a factor chosen as $(0.15 + 0.1S)$ at each iteration, S is increased depending on its value. If S is less than one-quarter then it is doubled, if S is between one-quarter and unity then it is set to unity, and if it is equal to or greater than unity then it is increased by unity.

The flow diagram shown in Fig. 1 is the core of the program for solving polynomial equations. Since the method may be recast as a synthetic division method with the addition of but a few logic instructions and simple calculations, the iteration time and program length are of the same order as those for synthetic division methods such as Newton-Bairstow's method or Lin's method. The program has been used for finding the zeros of many polynomials (mostly about tenth order) including those with multiple roots (up to sixth order) and those with many roots close together. For the more straightforward cases each zero is found to within the accuracy of the computer in about six iterations or less. For the less straightforward cases about ten iterations are required, and in no instance have more than fourteen iterations been required. This represents a considerable improvement over the method of [1].

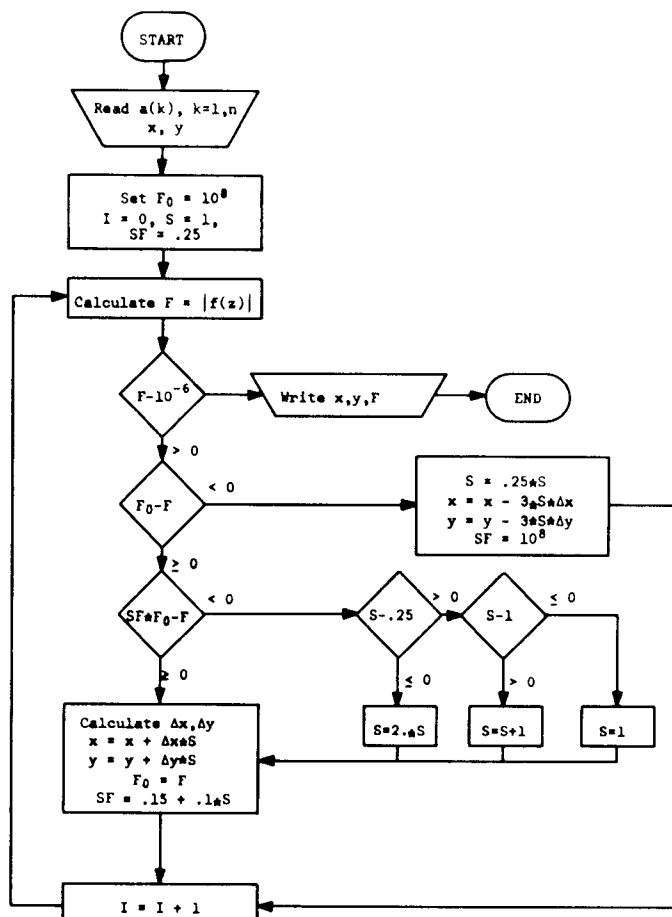


Fig. 1. Flow diagram.

In conclusion, it is believed that in terms of simplicity and convergence properties, the approach discussed here is more efficient than presently available methods—although the problem of ill-conditioning remains. We note further that the method is applicable to finding zeros of polynomials with complex coefficients and of analytic transcendental functions involving polynomials.

J. B. MOORE
K. T. CLARK
Dept. of Elec. Engrg.
University of Newcastle
New South Wales 2308
Australia