数理解析研究所講究録
第339巻 1978年 43-69

大域的収束性をもつ代数方程式の解法

AN ALGEBRAIC-EQUATION SOLVER WITH GLOBAL CONVERGENCE PROPERTY

東京大学工学部　伊理　正夫
東京大学工学部　山下　浩
東京大学工学部　寺野　隆雄
東京都立農芸高等学校　小野　令美

MASAO IRI*, HIROSHI YAMASHITA*, TAKAO TERANO* AND
HARUMI ONO**

* Department of Mathematical Engineering and Instrumentation Physics, Faculty of Engineering, University of Tokyo, Bunkyo-ku, Tokyo, Japan.
** Tokyo Metropolitan Nogei Agricultural Upper Secondary School, Suginami-ku, Tokyo, Japan.

1

Introduction

A practically usable algebraic-equation solver should desirably have the following properties:

(i) to be self-correcting;

(ii) to be rapidly convergent —— at least locally;

(iii) to be globally convergent, i.e. to afford all the roots whatever initial points we may start from;

(iv) to admit a sharp estimation of errors;

(v) to be computationally efficient.

Based upon the observation that, among the iterative algorithms which improve approximate points to all the roots simultaneously [1], [4], [6], [8] (see also the recent expositions [13] and [14]), there is one which can be endowed with property (iii), we propose a theoretically robust and practically powerful method for solving algebraic equations by incorporating a number of minor but substantial techniques in it. Among all, special attention has been paid to the way of calculating the value of a polynomial at a given value of the variable so as to avoid the overflow as far as possible and to get a rigorous estimate of rounding error. That enables us to expel the "ε" for convergence criterion which a user must ordinarily specify almost arbitrarily. (We need only machine constants such as the base of the number system adopted, the length of the mantissa and the information about the way of rounding.)

Properties (i) and (ii) are intrinsic to any algorithm of this kind, and property (iv) can be realized by B. T. Smith's method [11]. Experiments have shown that our method is not very slow in comparison with the ordinary routines for algebraic equations found in many computing centres (property (v)).

A detailed analysis has been carried out on the asymptotic behaviour of approximate solutions in the neighbourhood of a multiple root (or a "cluster" of roots too close to one another to be discriminated under the precision of computation), which revealed an interesting phenomenon to be utilized to enhance the efficiency of the method. The solutions of a series of equations of very high degrees (including those of degrees up to 200) are illustrated.

1. Problem

The problem we shall be concerned with is to find all the zeros $\alpha_1, \cdots, \alpha_n$ of a given polynomial of degree n over the complex field:

$$P(z) = z^n + a_1 z^{n-1} + \cdots + a_{n-1} z + a_n \quad (1.1)$$

$$= (z - \alpha_1)(z - \alpha_2) \cdots (z - \alpha_n). \quad (1.2)$$

It is assumed in the following that the polynomial is given in the form of (1.1), but, the method we shall develop in this paper is applicable to the case where the polynomial is defined by any algorithm of calculating the value of the polynomial itself as well as that of its first derivative for an arbitrarily given value of the independent variable z.

2. Iteration Schemata to Be Used

The following iteration schemata which have been well known for a long time (see, e.g., [4], [5], [8] etc.) will be investigated.

Setting

$$N = \{1, 2, \cdots, n-1, n\}, \quad (2.1)$$

we define functions

$$\varphi_i(z_1, \cdots, z_n) = -\frac{P(z_i)}{\prod_{j \in N-\{i\}} (z_i - z_j)} \quad (\forall i \in N) \quad (2.2)$$

and

$$\psi_i(z_1, \cdots, z_n) = -\frac{P(z_i)/P'(z_i)}{1 - P(z_i)/P'(z_i) \sum_{j \in N-\{i\}} \frac{1}{z_i - z_j}} \quad (\forall i \in N). \quad (2.3)$$

It is well known that, if the zeros of the polynomial are all simple and if $z_i^{(0)}$ is *sufficiently* close to α_i for every $i (\in N)$, then the sequence $(z_i^{(0)}, z_i^{(1)}, \cdots)$ generated by the iteration scheme

$$z_i^{(\nu+1)} = z_i^{(\nu)} + \varphi_i(z_1^{(\nu)}, \cdots, z_n^{(\nu)}) \quad (2.4)$$

converges *quadratically* to α_i $(\forall i \in N)$, and the sequence generated by the scheme

$$z_i^{(\nu+1)} = z_i^{(\nu)} + \psi_i(z_1^{(\nu)}, \cdots, z_n^{(\nu)}) \quad (2.5)$$

converges *cubically* to α_i $(\forall i \in N)$. (See, e.g., [2]. We shall reconsider those properties in the subsequent section from another viewpoint.)

3. Expressions of the Asymptotic Behaviour of Errors

Let us consider the case where

$$\alpha_1 = \cdots = \alpha_m \quad \text{and} \quad \alpha_j \neq \alpha_1 \quad (\forall j \in N-M) \tag{3.1}$$

with $N = \{1, \cdots, n\}$ and $M = \{1, \cdots, m\}$ $(\subseteq N)$.

Let us assume furthermore that $\varepsilon_j = z_j - \alpha_j$ are small enough for all $j \in N$, and evaluate $\varepsilon_i' = z_i' - \alpha_i$ in terms of ε_j, z_j, α_j, where

$$z_i' = z_i + \varphi_i(z_1, \cdots, z_n) \qquad (\forall i \in M) \tag{3.2}$$

or

$$z_i' = z_i + \psi_i(z_1, \cdots, z_n) \qquad (\forall i \in M). \tag{3.3}$$

For the φ-scheme of (3.2), the straightforward calculation will give

$$\begin{aligned}
\varepsilon_i' &= \varepsilon_i - \frac{P(z_i)}{\prod_{j \in N-\{i\}} (z_i - z_j)} \\
&= \varepsilon_i - \frac{\varepsilon_i^m}{\prod_{j \in M-\{i\}} (\varepsilon_i - \varepsilon_j)} \prod_{j_1 \in N-M} \frac{z_i - \alpha_{j_1}}{z_i - z_{j_1}} \\
&= \varepsilon_i - \frac{\varepsilon_i^m}{\prod_{j \in M-\{i\}} (\varepsilon_i - \varepsilon_j)} \prod_{j_1 \in N-M} \left(1 + \frac{\varepsilon_{j_1}}{\alpha_1 - z_{j_1}} - \frac{\varepsilon_i \varepsilon_{j_1}}{(\alpha_1 - z_{j_1})^2} + \frac{\varepsilon_i^2 \varepsilon_{j_1}}{(\alpha_1 - z_{j_1})^3} - \cdots \right) \\
&= \varepsilon_i - \frac{\varepsilon_i^m}{\prod_{j \in M-\{i\}} (\varepsilon_i - \varepsilon_j)} - \frac{\varepsilon_i^m}{\prod_{j \in M-\{i\}} (\varepsilon_i - \varepsilon_j)} \sum_{j_1 \in N-M} \frac{\varepsilon_{j_1}}{\alpha_1 - z_{j_1}} \\
&\quad + \frac{\varepsilon_i^{m+1}}{\prod_{j \in M-\{i\}} (\varepsilon_i - \varepsilon_j)} \left[\sum_{j_1 \in N-M} \frac{\varepsilon_{j_1}}{(\alpha_1 - z_{j_1})^2} + \sum_{j_1 \neq j_2;\, j_1, j_2 \in N-M} \frac{\varepsilon_{j_1} \varepsilon_{j_2}}{(\alpha_1 - z_{j_1})^2 (\alpha_1 - z_{j_2})} \right] \\
&\quad - \frac{\varepsilon_i^{m+2}}{\prod_{j \in M-\{i\}} (\varepsilon_i - \varepsilon_j)} \left[\sum_{j_1 \in N-M} \frac{\varepsilon_{j_1}}{(\alpha_1 - z_{j_1})^3} + \sum_{j_1 \neq j_2;\, j_1, j_2 \in N-M} \frac{\varepsilon_{j_1} \varepsilon_{j_2}}{(\alpha_1 - z_{j_1})(\alpha_1 - z_{j_2})} \right. \\
&\qquad \left. \times \left(\frac{1}{(\alpha_1 - z_{j_1})^2} + \frac{1}{(\alpha_1 - z_{j_1})(\alpha_1 - z_{j_2})} + \frac{1}{(\alpha_1 - z_{j_2})^2} \right) \right] \\
&\quad + \cdots .
\end{aligned} \tag{3.4}$$

If $m=1$, i.e. if α_1 is a simple zero, then we have

$$\varepsilon_1' = -\varepsilon_1 \cdot \sum_{j\in N-\{1\}} \frac{\varepsilon_j}{\alpha_1 - z_j} + O\left(\varepsilon_1^2 \cdot \varepsilon_{j(\neq 1)}\right). \tag{3.5}$$

It should be noted here that the old ε_1 is multiplied by the other ε_j's to yield the new ε_1'. If $m \geq 2$, i.e. if α_1 is a multiple zero, then the old ε_i's and the new ε_i''s with $i \in M$ are of the same order in magnitude. However, it is noteworthy that the deviation of the centre of gravity of z_i's ($i \in M$) from α_1, which is equal to $\frac{1}{m}\sum_{i\in M} \varepsilon_i$, shows a faster convergence similar to (3.5):

$$\begin{aligned}
\sum_{i\in M} \varepsilon_i' &= \sum_{i\in M} \varepsilon_i - \sum_{i\in M} \frac{\varepsilon_i^m}{\prod_{j\in M-\{i\}} (\varepsilon_i - \varepsilon_j)} - \sum_{i\in M} \frac{\varepsilon_i^m}{\prod_{j\in M-\{i\}} (\varepsilon_i - \varepsilon_j)} \sum_{j_1\in N-M} \frac{\varepsilon_{j_1}}{\alpha_1 - z_{j_1}} + \cdots \\
&= -\sum_{i\in M} \varepsilon_i \cdot \sum_{j_1\in N-M} \frac{\varepsilon_{j_1}}{\alpha_1 - z_{j_1}} + \left(\sum_{i\in M} \varepsilon_i^2 + \sum_{\substack{i_1<i_2 \\ i_1,i_2\in M}} \varepsilon_{i_1}\varepsilon_{i_2}\right) \cdot O\left(\varepsilon_{j(\in N-M)}\right) + \cdots \\
&= -\sum_{i\in M} \varepsilon_i \cdot \sum_{j_1\in N-M} \frac{\varepsilon_{j_1}}{\alpha_1 - z_{j_1}} + O\left(\varepsilon_i^2 \cdot \varepsilon_{j(\in N-M)}\right).
\end{aligned} \tag{3.6}$$

For the Ψ-scheme of (3.3), we have, in general,

$$\begin{aligned}
\varepsilon_i' &= \varepsilon_i - \left[P'(z_i)/P(z_i) - \sum_{j\in N-\{i\}} \frac{1}{z_i - z_j} \right]^{-1} \\
&= \varepsilon_i - \left[\sum_{j\in N} \frac{1}{z_i - \alpha_j} - \sum_{j\in N-\{i\}} \frac{1}{z_i - z_j} \right]^{-1} \\
&= \varepsilon_i - \left[\frac{1}{\varepsilon_i} - \sum_{j\in N-\{i\}} \frac{\varepsilon_j}{(z_i - z_j)(z_i - \alpha_j)} \right]^{-1} \\
&= \varepsilon_i - \varepsilon_i \left[1 - \sum_{j\in M-\{i\}} \frac{\varepsilon_j}{\varepsilon_i - \varepsilon_j} - \varepsilon_i \sum_{j\in N-M} \frac{\varepsilon_j}{(z_i - z_j)(z_i - \alpha_j)} \right]^{-1}.
\end{aligned} \tag{3.7}$$

If $m=1$, we have

$$\varepsilon_1' = -\varepsilon_1^2 \cdot \sum_{j\in N-\{1\}} \frac{\varepsilon_j}{(z_1 - z_j)(z_1 - \alpha_j)} + O\left(\varepsilon_1^3 \cdot \varepsilon_{j(\neq 1)}\right), \tag{3.8}$$

i.e. the new ε_1' is equal, in the order of magnitude, to the square of the

old ε_1 multiplied by the other ε_j's. However, if $m \geq 2$, the new ε'_i's are of the same order as the old ε_i's, so that we cannot expect more than a linear convergence in this case.

4. Differential Process Corresponding to the Ψ-Scheme and an Algorithm with Global Convergence Property

As is easily noted, the differential process correspondig to the finite difference scheme (3.3) has an integral which converges to zero, and we shall take advantage of this fact to establish an algorithm which yields a sequence of n-tuples of approximate points converging almost surely to exact zeros whatever initial points we may start from.

Specifically, we consider the system of differential equations:

$$\left.\begin{aligned} \frac{dz_i}{dt} &= \Psi(z_1, \cdots, z_n) \qquad (\forall\, i \in I), \\ \frac{dz_j}{dt} &= 0 \qquad\qquad\quad (\forall\, j \in J), \end{aligned}\right\} \tag{4.1}$$

where $N = I \cup J$ $(I \cap J = \phi)$. For any solution of the system (4.1), the scalar function

$$\nabla_I(z) = \frac{\prod_{i \in I} P(z_i)}{\prod_{\substack{i<j \\ i,j \in I}} (z_i - z_j) \prod_{\substack{i \in I \\ j \in J}} (z_i - z_j)} \tag{4.2}$$

satisfies

$$\begin{aligned} \frac{d}{dt}(\log \nabla_I(z)) &= \sum_{i \in I} \left(\frac{P'(z_i)}{P(z_i)} - \sum_{j \in I-\{i\}} \frac{1}{z_i - z_j} - \sum_{j \in J} \frac{1}{z_i - z_j} \right) \frac{dz_i}{dt} \\ &= -\sum_{i \in I} \frac{1}{\Psi_i(z)} \frac{dz_i}{dt} = -|I| \end{aligned} \tag{4.3}$$

or

$$\nabla_I(z) = c \cdot \exp(-|I| t) \,. \tag{4.4}$$

It will be seen (and we shall discuss about it in detail elsewhere) that the set of critical points of the vector field defined by the right-hand side of (4.1) constitutes a subvariety of dimension at most n-1 in the n-dimensional space with coordinates $(z_1, \cdots, z_n)$, and the critical points are not stable excepting those for which the z_i's with $i \in I$ are

equal to α_j's with distinct j's. Therefore, starting from any "general" initial points $(z^{(0)}_1, \cdots, z^{(0)}_n)$, we shall have $V_I(z)$ tend to zero as $t \to \infty$, which means that at least one of z_i's $(i \in I)$ will tend to a zero of $P(z)$.

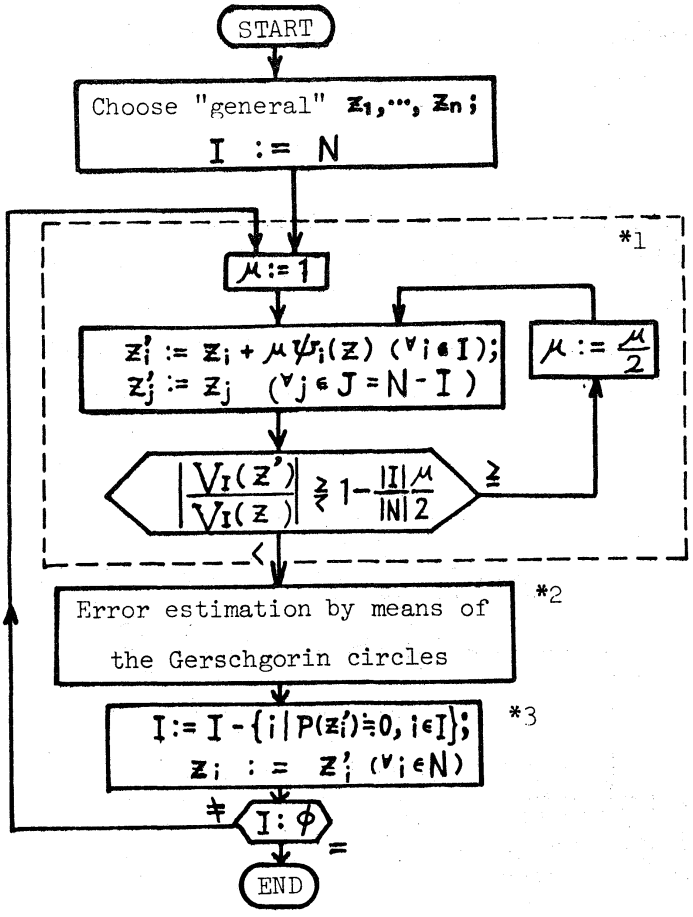

Fig. 1. Outline of the algorithm

---

*1. In order to guarantee the convergence, this loop is necessary. But, in practice, we need seldom make μ smaller than 1.

*2. See §6.

*3. The meaning of $P(z) \fallingdotseq 0$ will be defined in §5. Actually, we set $z_i := z'_i + \varphi_i(z)$ instead of $z_i := z'_i$ when letting i out of I.

This implies further that, if μ is taken to be small enough, the sequence of $(z_1^{(\nu)}, \cdots, z_n^{(\nu)})$'s $(\nu = 0,1,2,\cdots)$ generated by

$$\left.\begin{aligned} z_i^{(\nu+1)} &= z_i^{(\nu)} + \mu \psi_i(z^{(\nu)}) \quad (\forall i \in I), \\ z_j^{(\nu+1)} &= z_j^{(\nu)} \quad (\forall j \in J) \end{aligned}\right\} \tag{4.5}$$

will find at least one of the zeros of $P(z)$.

We can combine this fact with the local convergence property of the iteration scheme of (3.3) to get an algorithm for finding all the zeros of $P(z)$, as is outlined in the flow-chart of Fig. 1.

5. Remarks on the Computation of the Value of a Polynomial

It is widely recognized that the way of computing the value of a polynomial is the most crucial problem in solving an algebraic equation, and that there are a number of issues which have been studied thereupon. We took measures against the two most significant issues.

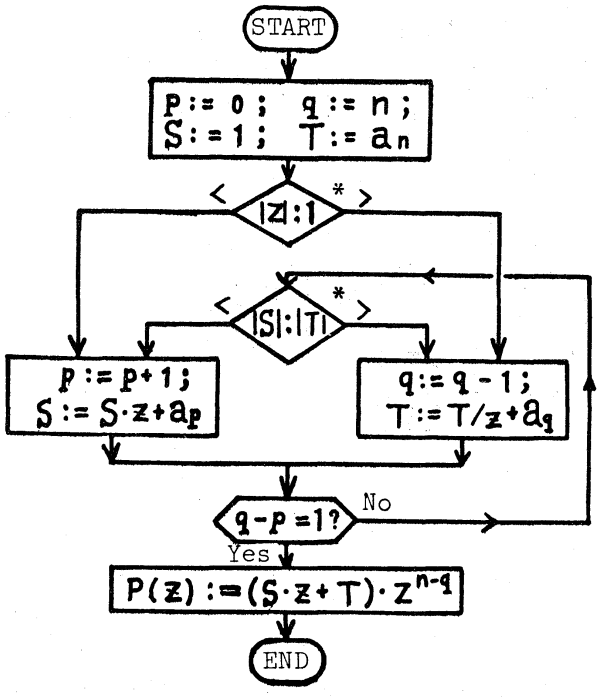

Fig. 2. Computation of the value of a polynomial $(n \geq 2)$

---

* $|\zeta|$ is substituted for by $\max(|\mathrm{Re}\,\zeta|, |\mathrm{Im}\,\zeta|)$ in order to reduce the computation time.

First, we adopted the two-sided nesting [11]:

$$P(z) = [(\cdots((z+a_1)z+a_2)z+\cdots+a_{m-1})z$$
$$+(\cdots((a_n/z+a_{n-1})/z+a_{n-2})/z+\cdots+a_{m+1})/z+a_m]\times z^{n-m} \quad (5.1)$$

in order to suppress the overflow that may take place during the computation and to keep the loss of significant digits to the minimum. Actually, we followed the expedient means illustrated in the flow-chart of Fig. 2 (for $n\geq 2$).

Secondly, in order to strictly evaluate the rounding error occurring in the computation, we resorted to a kind of "complex interval arithmetic". Although there have already been a few proposals for the extension of the interval arithmetic over the real field to the complex field [7], [9], [10], we used an easy one as follows. We shall represent a complex number contaminated with some kind of error by a circular disk with centre z and radius $\rho(\geq 0)$, which we shall denote as (z,ρ). Moreover, we make use of the quantity

$$u = \gamma\cdot M^{-(L-1)} \quad (5.2)$$

to express the "precision" of computation, where M is the base of the number system used, L is the length of the mantissa of the normalized floating-point expression of numbers, and $\gamma=\frac{1}{2}$ or 1 according as the result of computation is rounded or chopped. Then, we define the operations of addition/subtraction, multiplication and division on (z,ρ)'s by the formulae:

$$(z_1,\rho_1)\pm(z_2,\rho_2) = (z_1\pm z_2,\rho_3)$$

where

$$\rho_3 = \rho_1+\rho_2+u\sqrt{[\max(|\mathrm{Re}\,z_1|,|\mathrm{Re}\,z_2|,|\mathrm{Re}(z_1\pm z_2)|)]^2+[\max(|\mathrm{Im}\,z_1|,|\mathrm{Im}\,z_2|,|\mathrm{Im}(z_1\pm z_2)|)]^2}, \quad (5.3)$$

$$(z_1,\rho_1)(z_2,\rho_2) = (z_1\cdot z_2,\rho_4)$$

where

$$\rho_4 = |z_1|\cdot\rho_2+|z_2|\cdot\rho_1+\rho_1\cdot\rho_2+|z_1|\cdot|z_2|\cdot u, \quad (5.4)$$

and

$$(z_1,\rho_1)/(z_2,\rho_2) = (z_1/z_2,\rho_5)$$

where $|z_2|>\rho_2$ and

$$\rho_5 = \frac{\rho_1}{|z_2|-\rho_2}+\frac{|z_1|\cdot\rho_2}{|z_2|(|z_2|-\rho_2)}+\frac{|z_1|}{|z_2|}\cdot u\,. \quad (5.5)$$

These formulae guarantee that the result of an operation performed on any pair of complex numbers in the disks on the left-hand side certainly lies in the disk on the right-hand side.

When computing the value of a polynomial $P(z)$, we assume the variable z and the coefficients a_i's as circular disks with radius 0, i.e. we set them as $(z,0)$ and $(a_i,0)$'s, and, by means of the above complex interval arithmetic performed according to the algorithm of Fig. 2, we get the value of the polynomial in the form of a disk $(P(z), \Delta P(z))$. We conclude that $P(z) \ni 0$ whenever we have $|P(z)| < \Delta P(z)$. In this way, we can dispense with the threshold value which is ordinarily to be prescribed by the user on a not very sound bases.

It is said that, in general, the interval arithmetic enormously overestimates errors, but, for the evaluation of a polynomial in the above-explained manner, it gives a sufficiently sharp bound for the error in the computed polynomial value.

6. Estimation of Errors

For rigorous and considerably sharp estimation of the errors of approximate values for the zeros of a polynomial $P(z)$ the method proposed by B. T. Smith [11] seems to be most effective. Although Smith developed his method in a fairly general setting by means of highly sophisticated arguments, it is possible to derive his result in a much simpler manner.

In fact, for an arbitrarily given set of n distinct complex numbers $z_1, \cdots, z_n$, let us consider the matrix A of the form

$$A = \begin{bmatrix} z_1 + \varphi_1(z) & \varphi_2(z) & \cdots & \varphi_n(z) \\ \varphi_1(z) & z_2 + \varphi_2(z) & \cdots & \varphi_n(z) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_1(z) & \varphi_2(z) & & z_n + \varphi_n(z) \end{bmatrix}. \tag{6.1}$$

The characteristic polynomial $\Phi_A(\lambda) = \det(\lambda I - A)$ of A is seen, by direct expansion of the determinant, to be equal to

$$\Phi_A(\lambda) = \prod_{i \in N} (\lambda - z_i) + \sum_{i \in N} \left(\prod_{j \in N - \{i\}} \frac{\lambda - z_j}{z_i - z_j} \right) P(z_i), \tag{6.2}$$

and, since $\Phi_A(\lambda)$ of (6.2) coincides in value with $P(\lambda)$ at $n+1$ distinct points $\lambda = z_1, \cdots, z_n$ and ∞, we have $\Phi_A(\lambda) = P(\lambda)$.

Hence, the zeros of $P(\lambda)$ coincide with the eigenvalues of A, so that we can apply the famous theorem of Gerschgorin to show that all the zeros

of $P(z)$ lie in the union

$$K = \bigcup_{i \in N} K_i \tag{6.3}$$

of the n Gerschgorin circles:

$$K_i = \left\{ z \,\middle|\, |z - (z_i + \varphi_i(z))| \leq (n-1)|\varphi_i(z)| \right\} \quad (i \in N), \tag{6.4}$$

and that each connected component of K_i contains as many zeros of $P(z)$ as the circles constituting the component.

It is important to be careful enough when we determine the circles K_i's, i.e. we have to resort to the complex interval arithmetic to compute the numerator of $\varphi_i(z) = -P(z_i) / \prod_{j \in N-(i)} (z_i - z_j)$ and to adopt $z_i + \varphi_i(z)$ as the centre of the circle K_i and $(n-1)(|P(z_i)| + \Delta P(z_i)) / |\prod_{j \in N-(i)} (z_i - z_j)|$ as its radius in (6.4). (Otherwise, we might underestimate the radius of a circle due to the effect of the rounding error which might be equal to $\Delta P(z)$ in the order of magnitude.)

7. Choice of the Initial Approximate Values

Since our algorithm produces a sequence of n-tuples converging to the set of zeros of $P(z)$ so long as we start from any initial n-tuple of distinct complex numbers which are in "general" position, the choice of starting values is not very substantial from the theoretical viewpoints. However, the better choice would give the faster convergence, so that the choice is of practical importance.

Experiments on a number of problems have shown that the starting values recommended by O. Aberth [1] are good. Aberth defined the polynomial:

$$S(w) = w^n - |c_2| w^{n-2} - \cdots - |c_{n-1}| w - |c_n|, \tag{7.1}$$

where c_i's are determined from the given polynomial by the relation:

$$\begin{aligned} P(z) &= z^n + a_1 z^{n-1} + \cdots + a_{n-1} z + a_n \\ &= \left(z + \frac{a_1}{n}\right)^n + c_2 \left(z + \frac{a_1}{n}\right)^{n-2} + \cdots + c_n . \end{aligned} \tag{7.2}$$

Based on the fact that $S(w)=0$ has a unique positive root r_0 and that all the zeros of $P(z)$ lie in the circle with centre $-a_1/n$ (which is equal to the centre of gravity of the zeros) and radius r_0, he recommended to choose $z^{(0)}_k$'s $(k \in N)$ as follows [1]:

$$z^{(0)}_k = -\frac{a_1}{n} + r_0 \exp\left[\sqrt{-1}\left(2\pi \frac{k-1}{n} + \frac{\pi}{2n}\right)\right] \quad (\forall k \in N). \tag{7.3}$$

However, in order to choose the initial values as "general" as possible, we would recommend to start from

$$z^{(0)}_k = -\frac{a_1}{n} + r_0 \exp\left[\sqrt{-1}\left(2\pi\frac{k-1}{n} + \frac{3}{2n}\right)\right] \quad (\forall k \in N). \qquad (7.4)$$

(Note that "3" is used in (7.4) instead of "π" in (7.3) because $z^{(0)}_k$'s are then arranged on the periphery of the circle "transcendentally" with respect to the real and imaginary axes.) Furthermore, we do not need the exact value of r_0 but any value not less than it, so that we may set, for example, $w^{(0)} = \max\{|(n-1)c_m|^{1/m} \mid m = 2, \cdots, n\}$ and take $w^{(1)} = w^{(0)} - [S(w^{(0)}) / S'(w^{(0)})]$ or $w^{(2)} = w^{(1)} - [S(w^{(1)}) / S'(w^{(1)})]$ for r_0. (It is not difficult to see that $w^{(0)} \geq r_0$ and that, for any $w \geq r_0$, all the derivatives of $S(w)$ are nonnegative.)

8. Numerical Examples

8.1. Comparison with different methods. Our method is basically the iteration of the Jacobi type (i.e. of the simultaneous-replacement type) with the ψ-scheme. Fig. 3 shows the comparison of this method with other related methods — those with the φ-scheme instead of the ψ-scheme and/or the Gauss-Seidel type (i.e. the successive-replacement type) instead of the Jacobi type — from the viewpoints of speed of convergence.

It is observed that the convergence of the ψ-iteration is about two times as fast as that of the φ-iterations and that, although the Gauss-Seidel-type iterations are somewhat faster than the Jacobi-type iterations, the number of the steps necessary for attaining the approximation of the same accuracy seems to differ from each other at most one or so. (This observation does not contradict G. Alefeld and J. Herzberger's theoretical analysis [2].) Thus, the iteration of the Jacobi-type with the ψ-scheme upon which our method is based is nearly the best from the viewpoint of convergence, not to mention the fact that it has the global convergence property whereas the other methods do not.

8.2. Examples of the trajectories of approximate points from different initial points. Fig. 4 shows two cases of computation for a polynomial of degree 10 with four distinct zeros — one simple, one double, one triple and the fourth quadruple; in one case we start from "random" initial values, and in the other from the points on the periphery of a very small circle

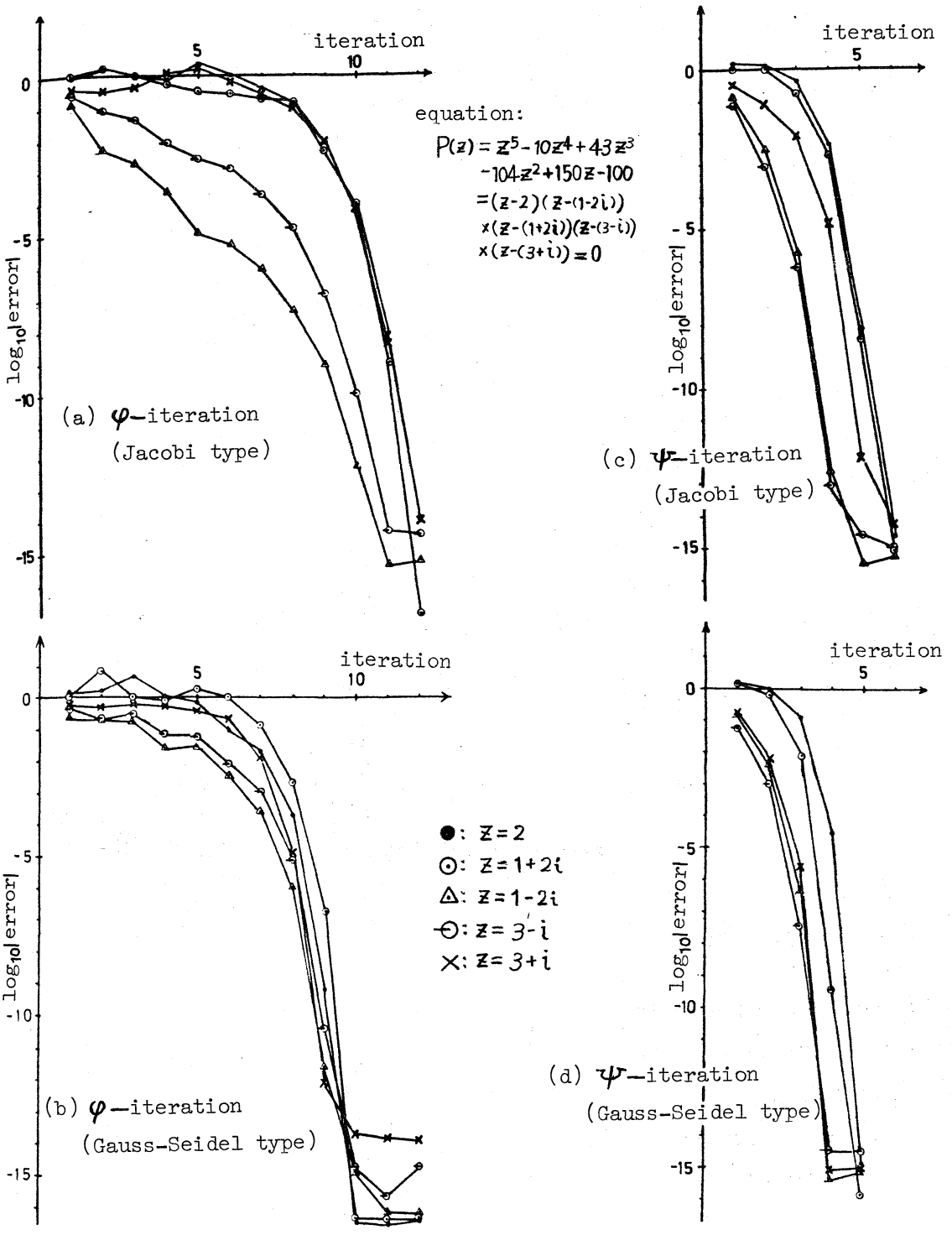

Fig. 3. Comparison of the φ-iterations with the ψ-iterations and of the iterations of the Jacobi-type with those of the Gauss-Seidel type

near the centre of gravity of the zeros.

Fig. 5 shows the trajectory for the polynomial $P_{15}(z)$ (to be defined in §8.4) with the starting values of (7.4).

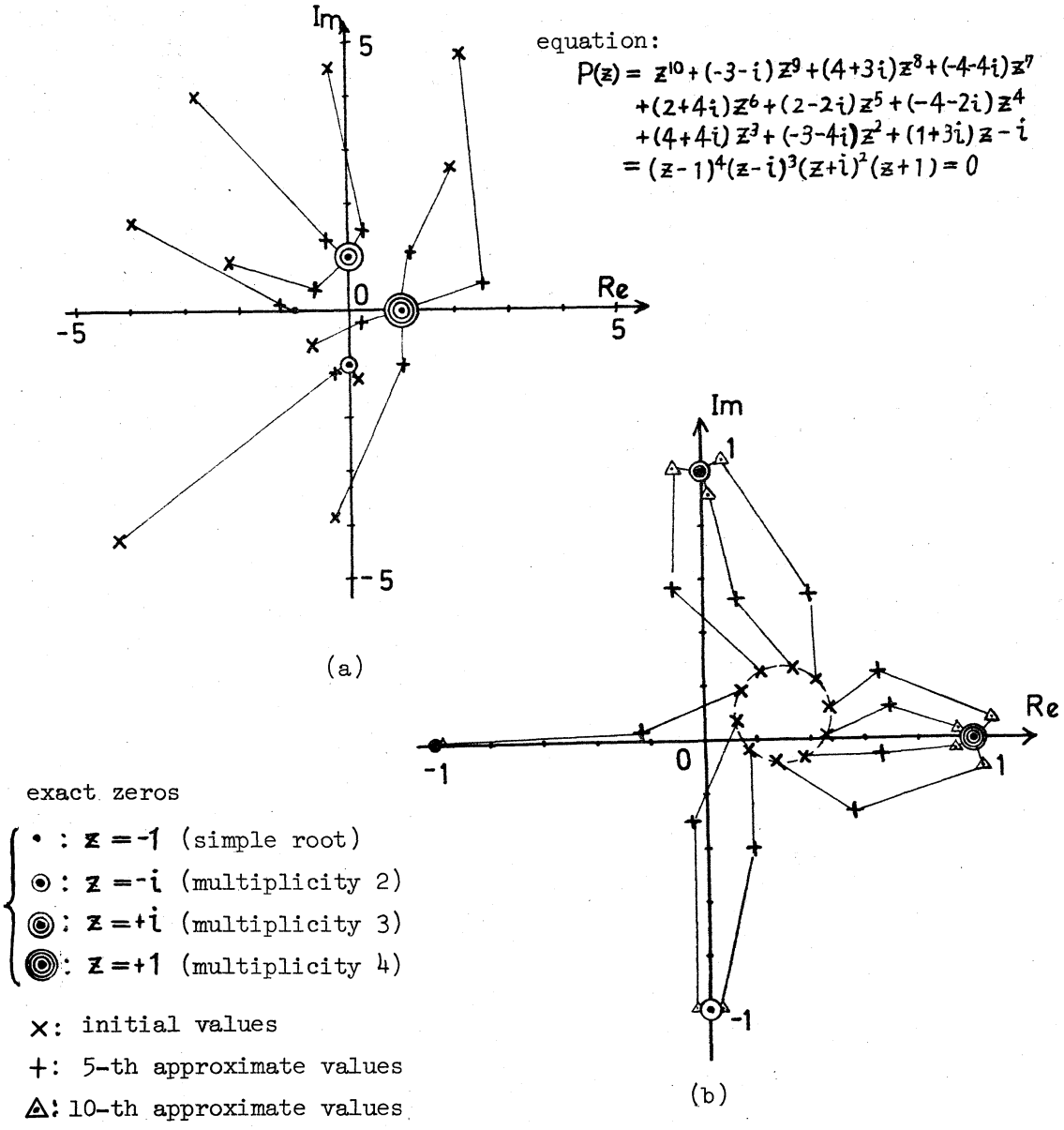

Fig. 4. Trajectories of the approximate points from different initial points

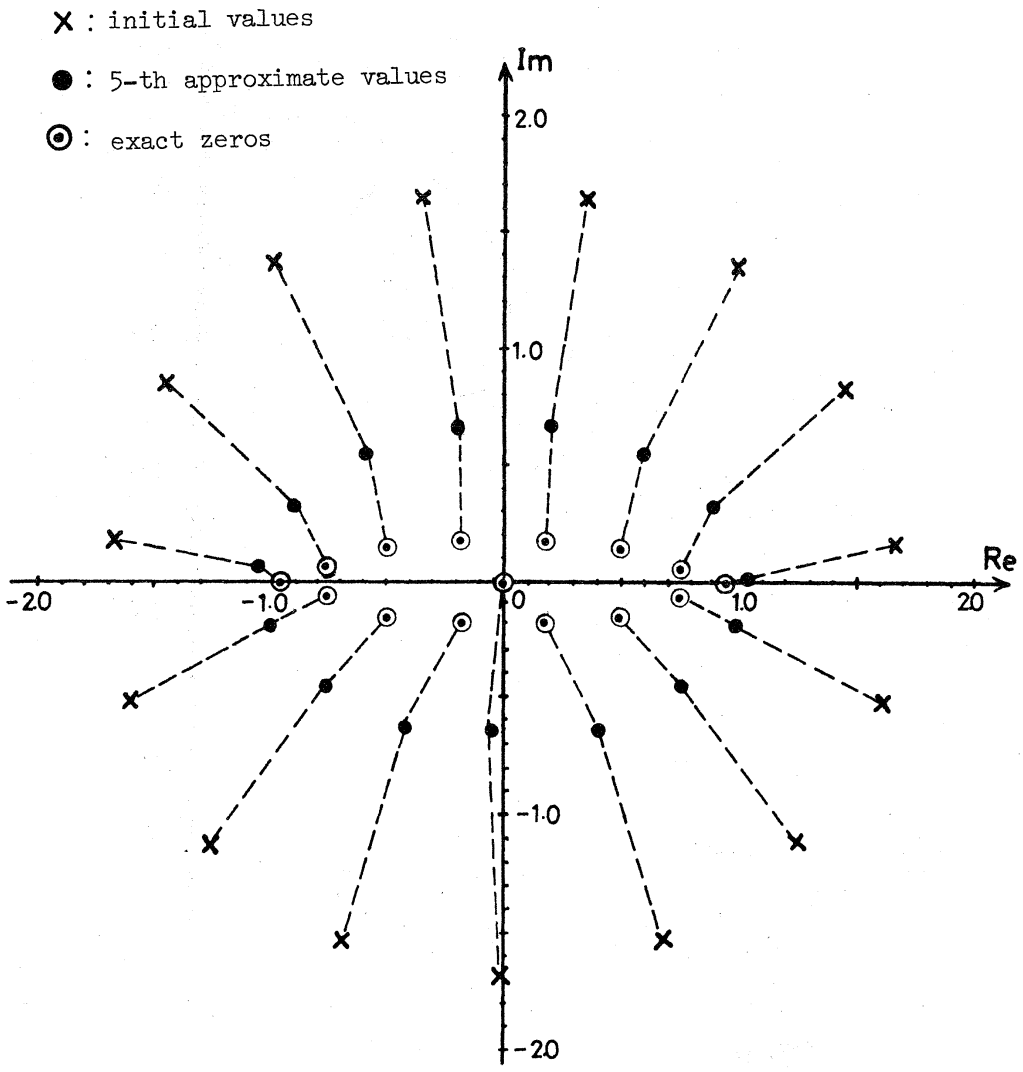

Fig. 5. Zeros of $P_{15}(z)$ and the approximate points approaching them

Fig. 6 shows the trajectory for the polynomial $P_{80}(z)$ (also to be defined in §8.4), where we started from a supposedly good approximate set of values. It is interesting to see the points move as if they were struggling for seats.

These numerical examples would evidence that our method is insensitive to and robust against the choice of initial points and has a good convergence property.

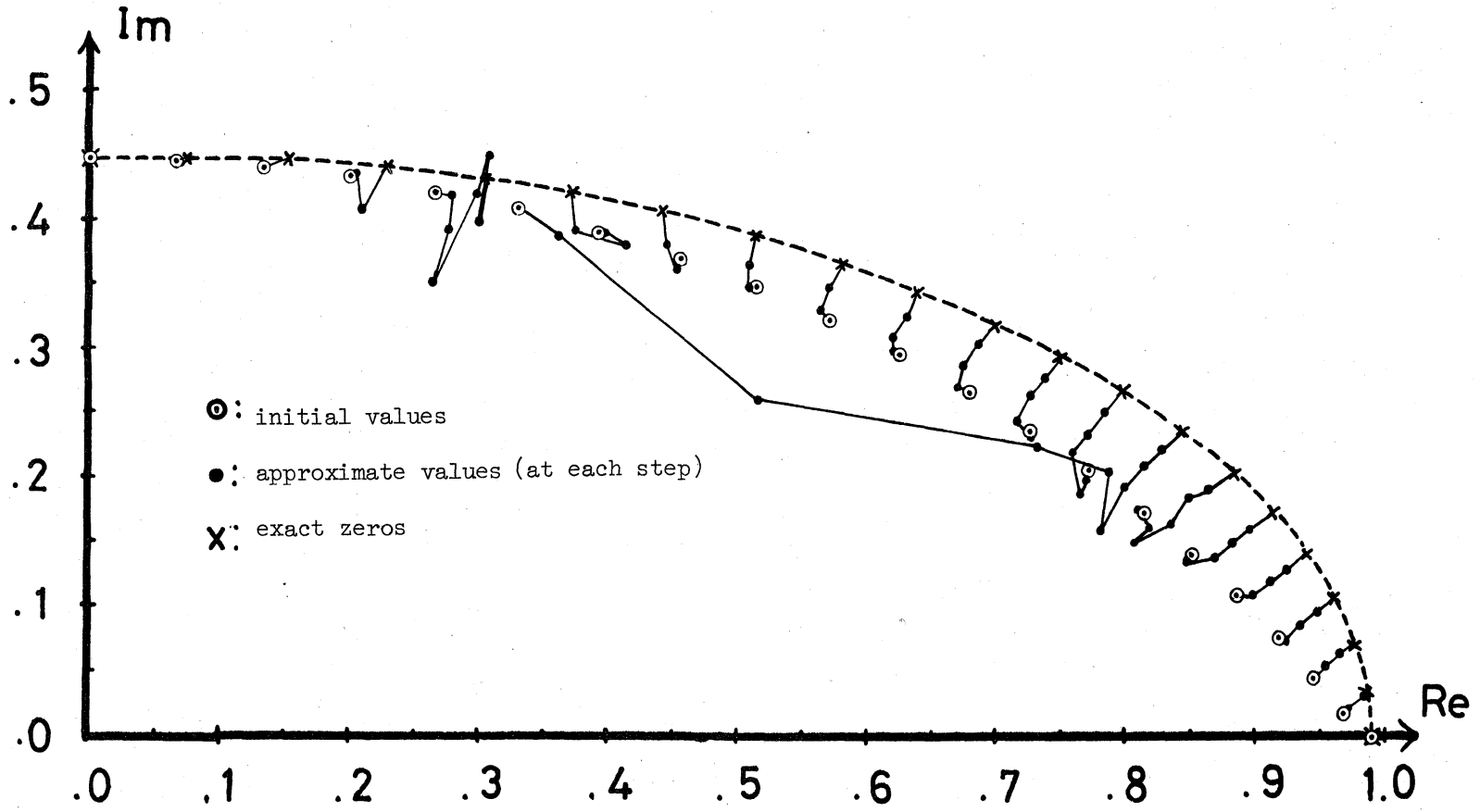

Fig. 6. Zeros of $P_{80}(z)$ obtained by starting from fairly good initial points

8.3. Behaviour around a multiple zero (or a cluster of zeros mutually very close). As was already noticed in §3, we can expect no better than linear convergence to a multiple zero (or a cluster of "nearly multiple" zeros). Moreover, it is our nowadays common sense that, roughly speaking, the number of significant digits of approximate values to an m-ple zero decreases to about one m-th of that of approximate values to simple zeros. These phenomena are clearly seen in the example of a seventh-degree polynomial with a triple zero and two complex-conjugate pairs of simple zeros in Fig. 7. There, the errors of the approximate values to simple zeros diminish very rapidly (cubically) to the order of $10^{-4}\sim10^{-5}$, $10^{-13}\sim10^{-15}$ and $10^{-30}\sim10^{-32}$ by single-precision, double-precision and quadruple-precision computations, respectively, whereas the errors of the three approximate values to the triple zero "2" diminish slowly (linearly) to the order of 10^{-1}, 10^{-4} and 10^{-10} by computations of the respective precisions.

What is the most remarkable in this example is the behaviour of the centre of gravity of the points $z_i^{(\nu)} + \varphi_i(z^{(\nu)})$ where $z_i^{(\nu)}$'s are the ν-th approximate values to the triple zero. The deviation of the centre from the true zero diminishes as rapidly as the errors of the approximate values to simple roots at the earlier stages of iteration. Equation (3.6) can explain this phenomenon well. The behaviour of the centre of gravity at the later stages of iteration is quite curious; its deviation from the triple zero takes the minimum at a certain stage, and, subsequently, grows gradually up to the order of magnitude of the errors of the separate approximate values. This phenomenon may be explained in terms of the rounding errors creeping in the process of computation as follows. While the approximate values are far from a multiple zero, they have almost as many significant digits as those used in computation, so that the behaviour of the centre of gravity is subject to equation (3.6) fairly well. However, as the approximate values approach the multiple zero nearer and nearer, they become contaminated by rounding errors more and more, so that their centre of gravity itself is diturbed by rounding errors to the same order of magnitude.

Fig. 8 illustrates a case with more than one multiple zero. This case is similar to that of Fig. 7, but it is seen that, although the centre of gravity approaches the multiple zeros more rapidly than the approximate values themselves, the convergence speed of the centre does not coincide with that of the approximate values to simple zeros. This is also in accord with (3.6).

equation: $P(z)=(z-2)^3(z-(1+2i))(z-(1-2i))(z-(3+i))(z-(3-i))=0$

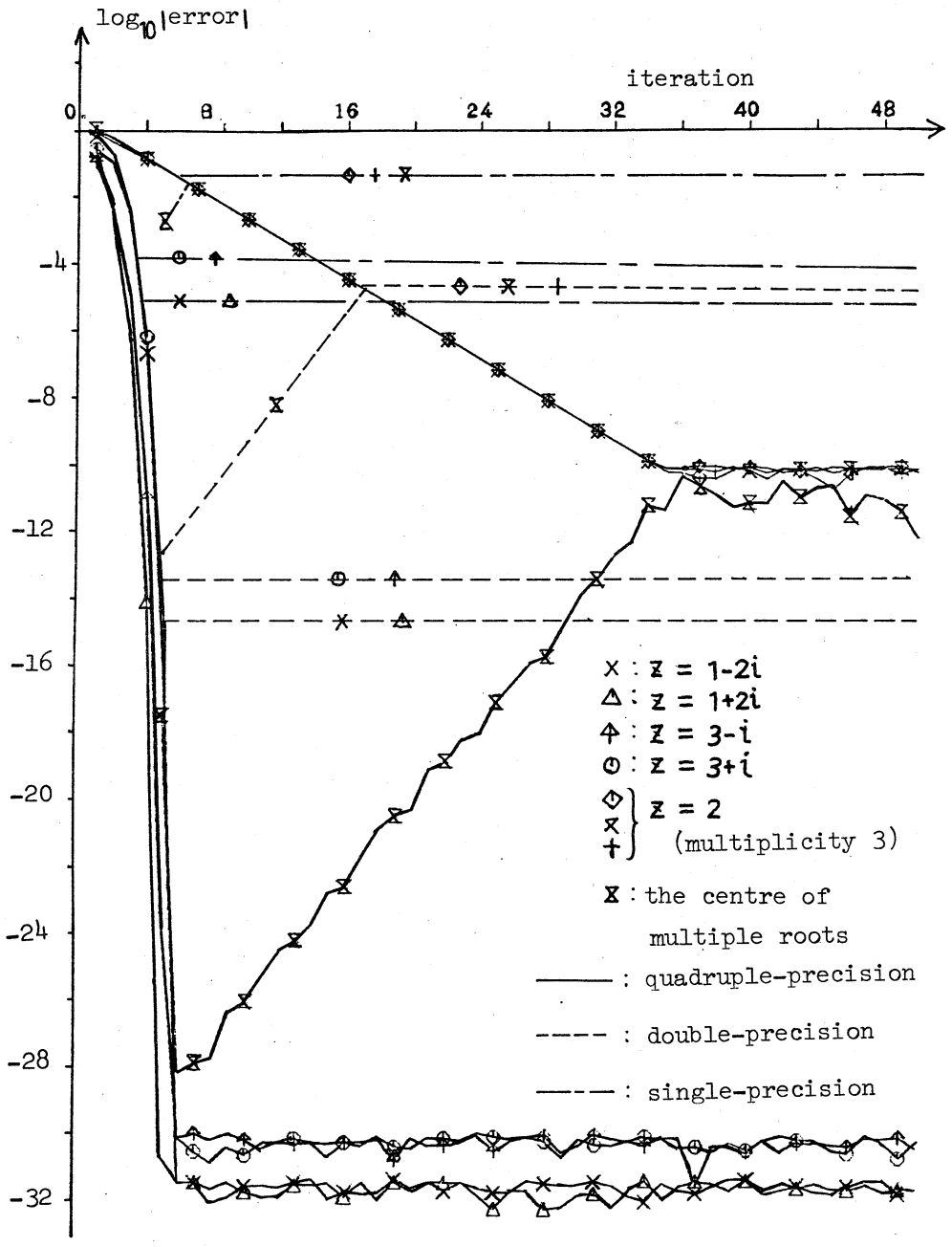

Fig. 7. Behaviour of the errors when there is one multiple zero

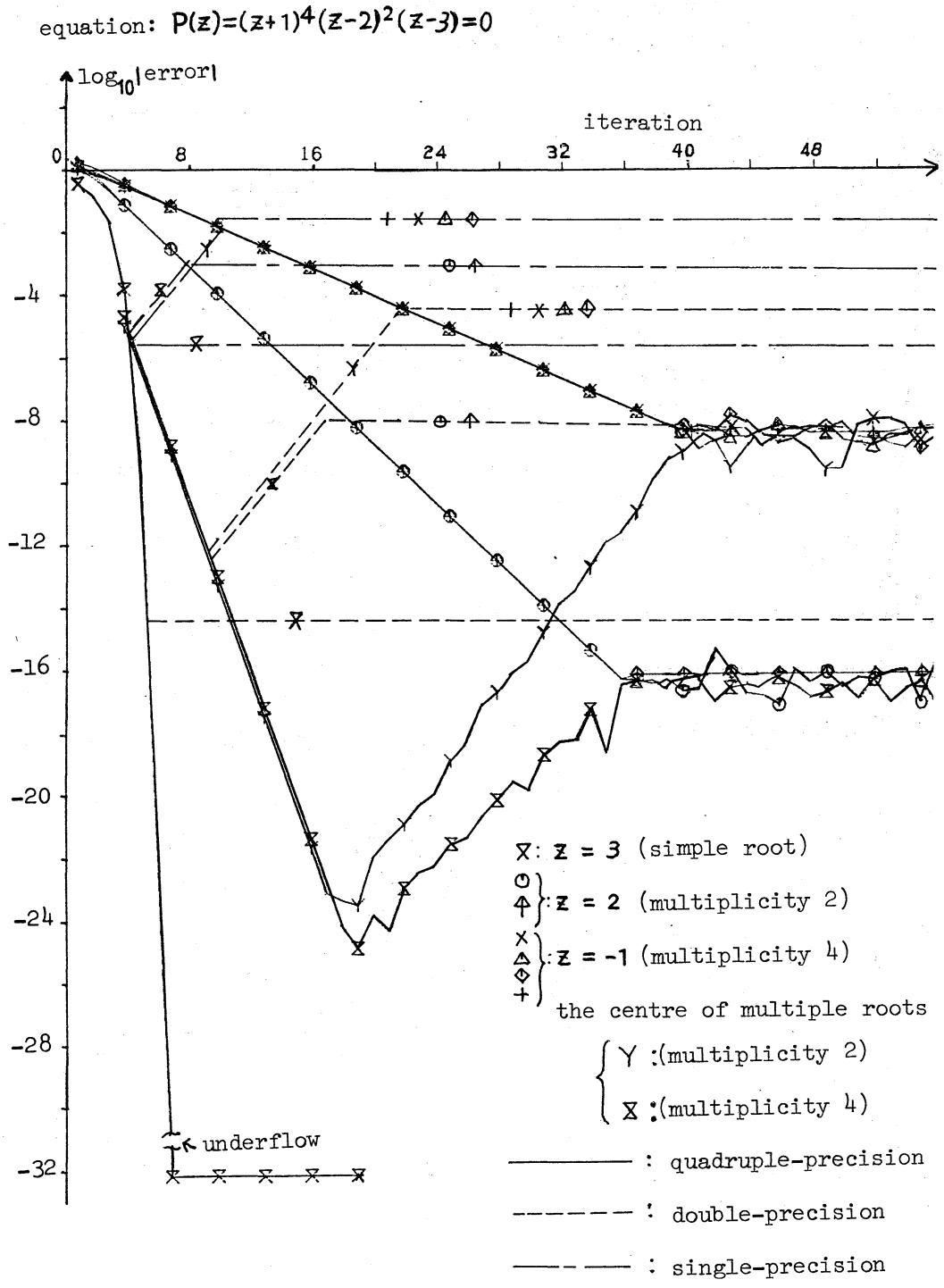

Fig. 8. Behaviour of the errors when there are more than one multiple zero

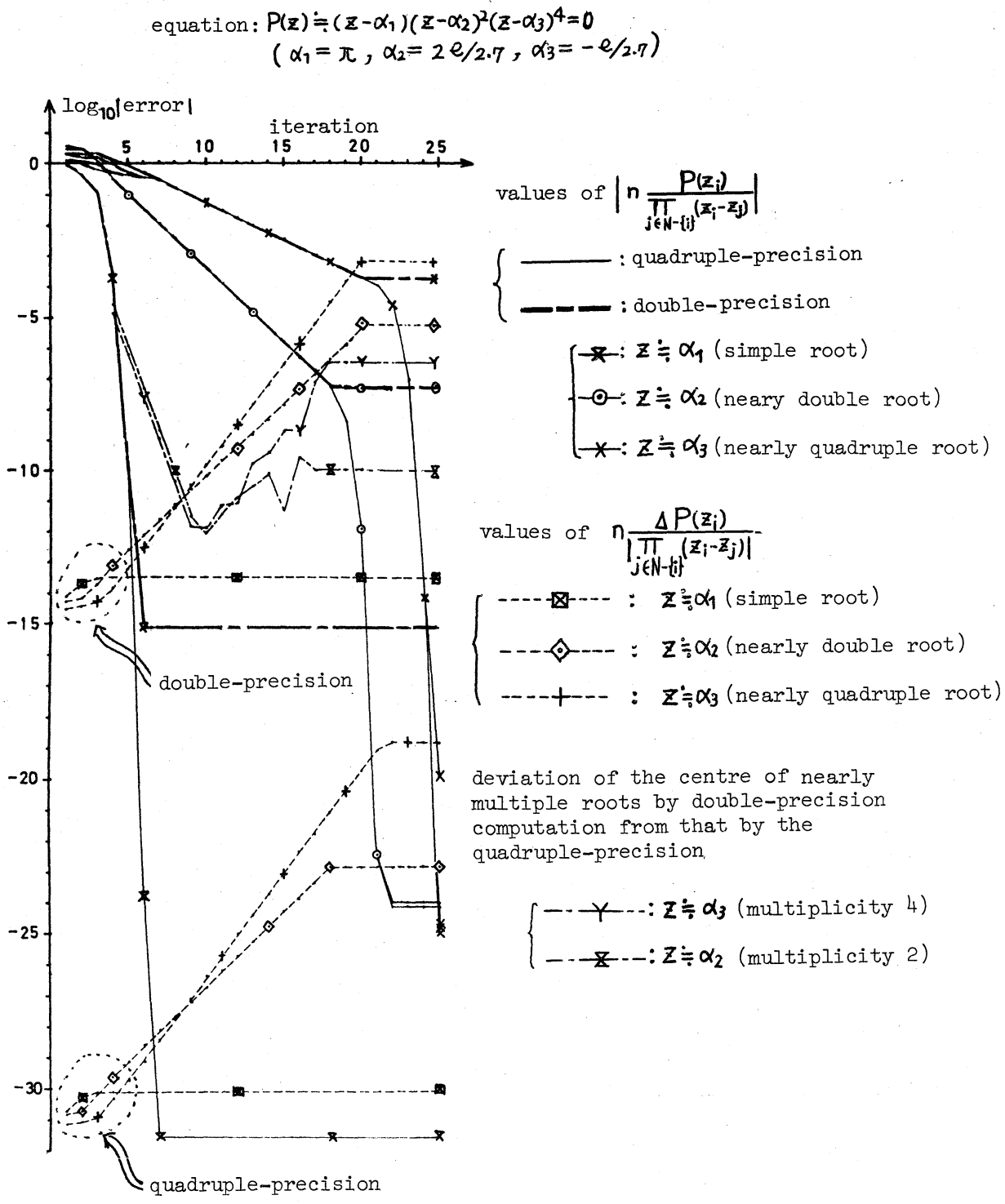

Fig. 9. Comparison of the resolution powers of computation of different precisions when there are clusters of zeros located very close to one another

The example shown in Fig. 9 is more subtle. The polynomial of this example was formed by first putting $\alpha_1=\pi$, $\alpha_2=2e/2.7$ and $\alpha_3=-e/2.7$, expanding $(z-\alpha_1)(z-\alpha_2)^2(z-\alpha_3)^4$ into the sum of powers of z to get a polynomial of degree 7 by sufficiently high-precision computation, and then truncating the coefficients of the powers of z to the double-precision numbers. By double-precision computation, the simple zero near α_1 was obtained after 5 iterations with the error of about 10^{-15}, the two zeros near α_2 were obtained after 15 iterations with the errors of about 10^{-7}, and the four zeros near α_3 were obtained after 20 iterations with the errors of 10^{-3}. In the case of the two (resp. four) zeros near α_2 (resp. α_3), their Gerschgorin circles do not separate from one another. The deviations of the centre of gravity of $z_i^{(\nu)} + \varphi_i(z^{(\nu)})$ (where $z_i^{(\nu)}$'s are the ν-th approximate values for the two zeros near α_2, and those for the four zeros near α_3) from the centres of gravity of the corresponding zeros obtained by quadruple-precision computation, first decrease and then grow up as shown in the figure. Thus, by double-precision computation, the approximate values behave themselves as if the polynomial had a simple zero α_1, a double zero α_2 and a quadruple zero α_3.

However, by quadruple-precision computation, seven zeros turned out to be all simple, with their errors estimated to be $10^{-20}\sim10^{-30}$ by means of the Gerschgorin circles which do not intersect one another.

8.4. Numerical solution of a series of algebraic equations of very high degrees. For the porpose of numerically backing up some theoretical conjecture we have tried to solve a series of algebraic equations of very high degrees.

The polynomials to be considered are those whose zeros determine the abscissae x_i's of the Chebyshev numerical integration formulae:

$$\int_{-1}^{1} f(x)dx \approx \frac{2}{n}\sum_{i=1}^{n} f(x_i). \tag{8.1}$$

It is known that the x_i's for the n-points Chebyshev formula are the zeros of the polynomial

$$\begin{aligned} P_n(z) &= (z-x_1)(z-x_2)\cdots(z-x_n) \\ &= z^n + a_2 z^{n-2} + a_4 z^{n-4} + \cdots + \begin{cases} a_{n-1}z \\ a_n \end{cases}, \end{aligned} \tag{8.2}$$

where a_j's are determined by the recurrence relations:

$$\left.\begin{aligned} &a_0 = 1, \quad a_2 = -n/6, \\ &a_{2k} = -\frac{n}{2k}\sum_{j=1}^{k}\frac{1}{2j+1}a_{2(k-j)} \quad (k=2,3,\cdots), \end{aligned}\right\} \tag{8.3}$$

and that $P_n(z)$'s have complex zeros except for n=1, 2, 3, 4, 5, 6, 7 and 9 [3].

S. Moriguti et al. studied the locations of the zeros of the $P_n(z)$'s for large n's theoretically and got to the conjecture that, as $n\to\infty$, all the zeros of $P_n(z)$ (except possibly $z=0$ for n odd) would be arranged densely on a closed curve which approaches the closed curve:

$$\left|(z+1)^{(z+1)/2}(z-1)^{-(z-1)/2}\right| = 2. \tag{8.4}$$

We first computed the coefficients of the polynomials by very high-precision computation and then rounded them according to the precision of computation used for finding the zeros of the polynomials. As an example we show in Fig. 10 the set of coefficients of $P_{200}(z)$, which were computed with 136 decimal digits and then rounded at the 41st decimal digits.

We could have all the zeros with the errors less than 10^{-5} for the polynomials of degrees not exceeding 60 by double-precision computation, and for the polynomials of degrees not exceeding 200 by quadruple-precision computations with the errors less than 10^{-2}. The zeros of $P_{20}(z)$, $P_{50}(z)$, $P_{100}(z)$ and $P_{200}(z)$ are illustrated in Fig. 11 together with the curve of (8.4).* In performing these computations, the initial values $z_k^{(0)} = x_k + \sqrt{-1}\,y_k$ were chosen carefully, using the following equations:

$$\left.\begin{aligned} &x_k = \frac{1}{\cos\theta}\cos\left[\left(\frac{\pi}{2}-\theta\right)\frac{n}{4}k+\theta\right], \quad \theta = \left(-\frac{n}{3600}+\frac{7}{60}\right)\pi, \\ &y_k = \frac{1}{2}\left[\frac{1}{2}\cos\frac{\pi}{2}x_k + c(1-x_k)\tan^{-1}\frac{1}{2c(1-x_k)}\right], \quad c = \frac{n}{180}+\frac{1}{3}. \end{aligned}\right\} \tag{8.5}$$

---

* The zeros of $P_{200}(z)$ were computed by the quadruple-precision floating-point arithmetic with the 131-bit mantissa ($\fallingdotseq$ 39 decimal digits) in binary expression. As for the other computations, see the concluding remarks in §10.

$a_0 =$	E	1 *	0,1000000000	0000000000	0000000000	0000000000	
$a_2 =$	-E	2 *	0,3333333333	3333333333	3333333333	3333333333	
	E	3 *	0,5455555555	5555555555	5555555555	5555555556	
	-E	4 *	0,5844268077	6014109347	4426807760	1410934744	
	E	5 *	0,4609072604	3503821281	5990593768	3715461493	
	-E	6 *	0,2853809089	1276076461	2616464468	3163201682	
	E	7 *	0,1444767599	4091044061	6495878106	6952730680	
	-E	7 *	0,6149930612	2747706130	0302530951	1690072145	
	E	8 *	0,2246452943	8147084437	4249906558	8520849363	
	-E	8 *	0,7151835230	5900750635	6050321509	7702226604	
$a_{20} =$	E	9 *	0,2008733417	5438414999	6088999037	3308971181	
	-E	9 *	0,5026559020	6220224897	7158655166	3745957739	
	E	10 *	0,1129689112	3264707881	6854632333	3011976145	
	-E	10 *	0,2295634032	0240096490	9990901869	5362736571	
	E	10 *	0,4241956734	2479044111	8193424484	4845533343	
	-E	10 *	0,7162347938	2322224077	2620282672	0897400079	
	E	11 *	0,1109650600	0762569042	2357492402	8302498982	
	-E	11 *	0,1583205468	0242867801	0166133042	2235595646	
	E	11 *	0,2086842603	6734981609	9739467520	9560788963	
	-E	11 *	0,2548348395	9750022835	8229765777	8517678955	
$a_{40} =$	E	11 *	0,2890140791	2210898809	9851728015	2554798541	
	-E	11 *	0,3050870917	9951481106	7735021955	6788245917	
	E	11 *	0,3003463086	9217730875	4689732714	1651202773	
	-E	11 *	0,2762308725	0727998264	7235063880	8045237642	
	E	11 *	0,2377120997	0285517285	0588006703	1741676781	
	-E	11 *	0,1916753825	4820733857	6024872066	4868899763	
	E	11 *	0,1449976796	5928360943	3710531194	9287354063	
	-E	11 *	0,1030202330	1750736062	8866495111	3826474518	
	E	10 *	0,6881559030	0474503522	0486461541	1076126613	
	-E	10 *	0,4325574417	4131250281	0129737501	7369807609	
$a_{60} =$	E	10 *	0,2560595456	3148636971	5963767938	0412645543	
	-E	10 *	0,1428523152	3381990994	6896823588	5396275923	
	E	9 *	0,7515466499	3012807760	0358327510	2997637466	
	-E	9 *	0,3730672253	4819098189	1289081626	0286452751	
	E	9 *	0,1748195842	5473540644	1201454883	9708120849	
	-E	8 *	0,7736515253	7736990988	4726081604	2453400698	
	E	8 *	0,3234493862	0701213799	5266293093	5687084530	
	-E	8 *	0,1277910815	2883366766	7534222083	9192434573	
	E	7 *	0,4772338242	8182358753	1471677826	7346591381	
	-E	7 *	0,1684915679	3060147251	3480737228	3685695846	
$a_{80} =$	E	6 *	0,5624692582	0670712186	3688727005	3136851898	
	-E	6 *	0,1775533133	0547284173	0849770934	7739861571	
	E	5 *	0,5300084694	4827375921	9602282245	9394321319	
	-E	5 *	0,1496080702	3993650945	6106307044	4036999153	
	E	4 *	0,3993185058	6413495549	9819207138	9029562600	
	-E	4 *	0,1007697665	6597416498	5671296878	3259555094	
	E	3 *	0,2403931153	2667595061	8251846798	6265096454	
	-E	2 *	0,5420130933	7119772766	3783139515	3975149863	
	E	2 *	0,1154752720	6931897135	4131325141	2937387733	
	-E	1 *	0,2323996838	2372704269	3847723044	4973560105	
$a_{100} =$	E	0 *	0,4416776704	0182501463	2254383388	7967111412	

$a_{102} =$	-E	-1 *	0,7923821642	5954345668	1268734061	4238765625	
	E	-1 *	0,1341339731	2047638205	7742971833	9334574987	
	-E	-2 *	0,2141471643	1187089733	4635243336	1601018361	
	E	-3 *	0,3222763334	2693300590	5018805258	4980721307	
	-E	-4 *	0,4569177098	7669795188	0580907510	6929129466	
	E	-5 *	0,6099146395	9960482356	3585599501	9764798105	
	-E	-6 *	0,7659946861	6754639771	7370481460	7024919354	
	E	-7 *	0,9044557877	5082739640	8120910133	0052731244	
	-E	-7 *	0,1003244987	0331252095	2637946230	3287930403	
$a_{120} =$	E	-8 *	0,1044504962	7138056210	4276881459	9115332781	
	-E	-9 *	0,1019752425	6424542844	0619250829	7150965657	
	E	-11 *	0,9326656746	5053987793	3968564432	5411196935	
	-E	-12 *	0,7982483345	0746900617	9882589382	2159500517	
	E	-13 *	0,6386015365	5571803054	4961752192	8987422801	
	-E	-14 *	0,4769401510	0700185072	7139699536	5568520366	
	E	-15 *	0,3320948286	8183462434	5431459409	5169026133	
	-E	-16 *	0,2152802262	8888736390	8436922107	9257003831	
	E	-17 *	0,1297248577	1513976028	6889604764	2118820773	
	-E	-19 *	0,7254414475	8485364061	8077675116	7758749594	
$a_{140} =$	E	-20 *	0,3758115322	9174506089	7013699482	2733328316	
	-E	-21 *	0,1800094352	5658685710	4793126100	1307632575	
	E	-23 *	0,7955753119	1475884603	9639555789	4740023041	
	-E	-24 *	0,3237135503	1507283385	6425698508	1204796691	
	E	-25 *	0,1209638077	8674303809	7635008769	0623825634	
	-E	-27 *	0,4148666410	0707425684	3652124695	0389988123	
	E	-28 *	0,1219409090	8052455079	2740119557	3969645668	
	-E	-29 *	0,1119189036	4159150487	6802700554	0904007927	
	-E	-30 *	0,7371274208	8416856926	9252082065	9120452476	
	-E	-30 *	0,7430508155	5536175632	4257256598	3323420348	
$a_{160} =$	-E	-30 *	0,7390241996	5701816461	0442238303	2570168923	
	-E	-30 *	0,7352560077	6985494630	3017966573	7146099475	
	-E	-30 *	0,7315091436	7710913353	9797734699	0872234167	
	-E	-30 *	0,7277883605	2356426598	9872867784	0725742447	
	-E	-30 *	0,7240933674	0741902826	9764971431	6586010790	
	-E	-30 *	0,7204239689	3563646499	2909612559	7846159570	
	-E	-30 *	0,7167799692	6048837436	0732249911	8410485870	
	-E	-30 *	0,7131611736	8942616185	0739908106	6876221117	
	-E	-30 *	0,7095673886	7095711418	1789388442	1517809098	
	-E	-30 *	0,7059984218	0384974285	2816161223	1705478776	
$a_{180} =$	-E	-30 *	0,7024540818	4540007134	4224933231	9248407845	
	-E	-30 *	0,6989341787	1909355210	7232341700	7064607733	
	-E	-30 *	0,6954385235	2167736524	3210837232	3627224090	
	-E	-30 *	0,6919669285	2966109691	4457341245	2385122492	
	-E	-30 *	0,6885192072	0526337268	2011943143	7736577989	
	-E	-30 *	0,6850951742	0182163808	1472154185	1035272310	
	-E	-30 *	0,6816946453	6868189943	9432725947	6441468938	
	-E	-30 *	0,6783174377	5558486270	0393516314	0474974857	
	-E	-30 *	0,6749633696	1656453704	7100886750	7743033215	
	-E	-30 *	0,6716322604	1337500351	4006348439	9209195995	
$a_{200} =$	-E	-30 *	0,6683239308	1846068675	7228446974	3671783813	

Fig. 10. Coefficients of $P_{200}(z)$ (all the coefficients of the odd powers of z vanish)

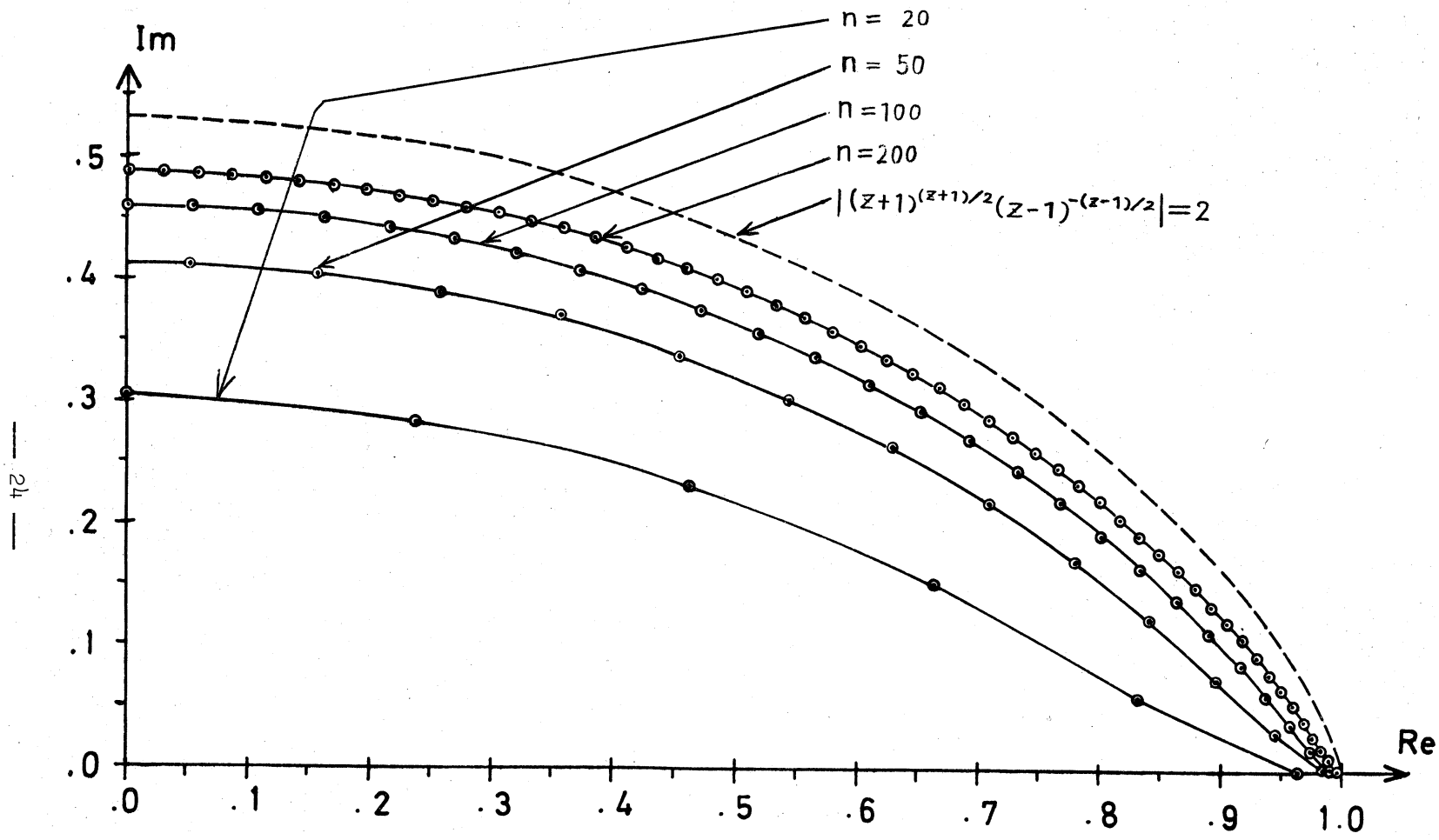

Fig. 11. Computed zeros of $P_n(z)$ for n=20, 50, 100 and 200, and the outline of the curve of (8.4), in the first quadrant

9. A Device for Enhancing the Efficiency of Computation of Multiple or Nearly Multiple Zeros

If a polynomial has multiple or nearly multiple zeros, the convergence to those zeros is slowed down so much that the computational efficiency of our method is affected seriously. In order to avoid this difficulty, we may make use of the property which we discovered in §8.3 as follows.

If some of the Gerschgorin circles for the approximate values z'_i $(i \in M)$ intersect one another after a certain prescribed number, say 3 or 4, of iterations (see Fig. 1), we compute

$$\bar{z}_M = \sum_{i \in M} (z'_i + \varphi_i(z')) \Big/ m \qquad (m = |M|). \tag{9.1}$$

If we have

$$P(\bar{z}_M) \fallingdotseq 0, \; P'(\bar{z}_M) \fallingdotseq 0, \; \cdots, \; P^{(m-1)}(\bar{z}_M) \fallingdotseq 0, \tag{9.2}$$

then we put

$$z_i := \bar{z}_M \; (\forall i \in M) \quad \text{and} \quad I := I - M \; ; \tag{9.3}$$

otherwise, we continue the iteration further.

In case of (9.3), we may make rough estimate of errors, to the effect that

m zeros are located within the circle of centre $\bar{z}_M$ and radius

$$\left[\frac{m! \left(|P(\bar{z}_M)| + \Delta P(\bar{z}_M) \right)}{|P^{(m)}(\bar{z}_M)|} \right]^{\frac{1}{m}} . \tag{9.4}$$

Applying this device to the example of Fig. 9, we could stop computation by double-precision after 5 (resp. 6) iterations for the zeros near α_3 (resp. α_2) and could obtain the approximate solutions with the estimated errors of the same order in magnitude as those we obtained after 20 (resp. 18) iterations according to the more basic algorithm of Fig. 1. If we resort to quadruple-precision computation, the conditions (9.2) (specifically, the first one) were not satisfied so that we had to continue until all the zeros were separated from one another.

10. Concluding Remarks and Acknowledgements

The method for calculating the zeros of polynomials proposed in the present paper has many advantages over other methods found in the existing literature. The authors tend to think it might be an ultimate one for that purpose.

Many of the ideas incorporated in the method, especially that in §4, arose during the discussions in the class on "Approximate Mathematics" of the Graduate School of the University of Tokyo conducted by Professor S. Moriguti, and the authors cordially thank him for his inspiring guidance and valuable suggestions.

Most of the numerical computations were carried out on HITAC 8700/8800 under the operating system OS/7 of the Computer Centre of the University of Tokyo. The programmes were written in FORTRAN IV. Floating-point numbers have the hexadecimal expression; the mantissa of single-precision numbers has 3 bytes or 6 hexadecimal digits ($\fallingdotseq$ 7 decimal digits), that of double-precision 7 bytes or 14 hexadecimal digits ($\fallingdotseq$ 16 decimal digits) and that of the quadruple-precision 14 bytes or 28 hexadecimal digits ($\fallingdotseq$ 33 decimal digits). The arithmetic operations of addition, subtraction, multiplication and division are followed by the operation of chopping.

References

[1] Aberth, O.: Iteration Methods for Finding All Zeros of a Polynomial Simultaneously. *Mathematics of Computation*, vol. 27, pp. 339-344 (1973).

[2] Alefeld, G., and Herzberger, J.: On the Convergence Speed of Some Algorithms for the Simultaneous Approximation of Polynomial Roots. *SIAM Journal on Numerical Analysis*, vol. 11, pp. 237-243 (1974).

[3] Bernstein, S.: Sur les formules de quadrature de Cotes et Tchebycheff. *Comptes Rendus (Doklady) de l'Académie des Sciences de l'URSS*, vol. XIV, no. 6, pp. 323-326 (1937).

[4] Börsch-Supan, W.: A Posteriori Error Bounds for the Zeros of Polynomials. *Numerische Mathematik*, vol. 5, pp. 380-398 (1963).

[5] Durand, E.: *Solutions numériques des équations algébriques*. Tome I: *Équations du type F(x)=0; Racines d'un Polynôme*, Masson, Paris, 1960.

[6] Farmer, M. R., and Loizou, G.: A Class of Iteration Functions for Improving, Simultaneously, Approximations to the Zeros of a Polynomial. *BIT*, vol. 15, pp. 250-258 (1975).

[7] Gargantini, I., and Henrici, P.: Circular Arithmetic and the Determination of Polynomial Zeros. *Numerische Mathematik*, vol. 18, pp. 305-320 (1972).

[8] Kerner, I. O.: Ein Gesamtschrittverfahren zur Berechung der Nullstellen von Polynomen. *Numerische Mathematik*, vol.8, pp. 290-294 (1966).

[9] Nickel, K.: Zeros of Polynomials and Other Topics. Hansen, E. R.(ed.): *Topics in Interval Analysis*, Oxford University Press, London, 1969, pp. 25-34.

[10] Rokne, J., and Lancaster, P.: Complex Interval Arithmetic. *Communications of the Association for Computing Machinery*, vol. 14, no. 2, pp. 111-112 (1971).

[11] Smith, B. T.: Error Bounds for Zeros of a Polynomial Based upon Gerschgorin's Theorems. *Journal of the Association for Computing Machinery*, vol. 17, pp. 661-674 (1970).

[12] Wilkinson, J. H.: *Rounding Errors in Algebraic Processes*. Her Majesty's Stationary Office, London, 1963.

[13] 山本哲朗：ある代数方程式解法と解の事後評価法. 数理科学, no. 157, pp. 52-57 (1976年7月).

[14] 山本哲朗，古金卯太郎，野倉久美：代数方程式を解く Durand-Kerner 法と Aberth 法. 情報処理, vol. 18, pp. 566-571.