ANL-7344

ANL-7344

481

MASTER

Argonne National Laboratory

**THE ROOT-CUBING AND THE
GENERAL ROOT-POWERING METHODS FOR
FINDING THE ZEROS OF POLYNOMIALS**

by

Erwin H. Bareiss

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

The facilities of Argonne National Laboratory are owned by the United States Government. Under the terms of a contract (W-31-109-Eng-38) between the U. S. Atomic Energy Commission, Argonne Universities Association and The University of Chicago, the University employs the staff and operates the Laboratory in accordance with policies and programs formulated, approved and reviewed by the Association.

MEMBERS OF ARGONNE UNIVERSITIES ASSOCIATION

The University of Arizona	Kansas State University	The Ohio State University
Carnegie-Mellon University	The University of Kansas	Ohio University
Case Western Reserve University	Loyola University	The Pennsylvania State University
The University of Chicago	Marquette University	Purdue University
University of Cincinnati	Michigan State University	Saint Louis University
Illinois Institute of Technology	The University of Michigan	Southern Illinois University
University of Illinois	University of Minnesota	University of Texas
Indiana University	University of Missouri	Washington University
Iowa State University	Northwestern University	Wayne State University
The University of Iowa	University of Notre Dame	The University of Wisconsin

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

Printed in the United States of America

Available from

Clearinghouse for Federal Scientific and Technical Information
National Bureau of Standards, U. S. Department of Commerce
Springfield, Virginia 22151

Price: Printed Copy \$3.00; Microfiche \$0.65

ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, Illinois 60439

THE ROOT-CUBING AND THE
GENERAL ROOT-POWERING METHODS FOR
FINDING THE ZEROS OF POLYNOMIALS

by

Erwin H. Bareiss

Applied Mathematics Division

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

June 1967

THIS PAGE
WAS THIS PAGE
WAS INTENTIONALLY
LEFT BLANK

TABLE OF CONTENTS

	<u>Page</u>
ABSTRACT.	5
1. Motivation	5
<i>PART I. ROOT-CUBING AND ROOT-SQUARING METHODS.</i>	5
2. The Root-cubing Method	5
3. Comparison with the Root-squaring Method	7
4. Extracting Roots from Polynomial Equations	14
5. The Root-bisquaring Transformation	17
<i>PART II. THE GENERAL ROOT-POWERING METHODS</i>	18
6. Newton's and Waring's Formulas	18
7. The General Algorithm for the Root-powering Method	21
8. The p th-power-of-roots Algorithm for Polynomials of Degree $p - 1$, and the General Root-powering Method by Partitioning of the Polynomial	32
9. Circulant Representation of the Root-powering Algorithms	38
10. Resultants and General Root-powering. Algorithm R	46
11. General Root-powering by Elimination Methods. Algorithms E1 to E4.	58
<i>APPENDIXES</i>	
A. A Proof for (9.4)	75
B. Another Derivation of (9.10).	76
C. Remarks on the Expansion of the Circulant	78
D. Method of Expanding Circulants on High-speed Computers.	80
E. An Extension of the Circulant Method.	83
F. The Root-powering Transformation for Analytic Functions	84
ACKNOWLEDGMENTS	86
REFERENCES.	87

LIST OF TABLES

<u>No.</u>	<u>Title</u>	<u>Page</u>
I.	Numerical Values for Formulas (3.5)	9
II.	Number of Full Multiplications Required for Root-cubing Transformations of $P_n = A + B + C$ Using Different Methods . .	11
III.	Number of Full Multiplications to Perform Eight Root-squaring or Five Root-cubing Operations for Various Degrees n of the Polynomial and Different Methods of Calculation	12
IV.	Maximum Number of Full Multiplications Required per Arithmetic Unit to Perform Eight Root-squaring Transformations $AA - BB$, viz., $8M[AA]$; Five Root-cubing Transformations $AAA + BBB + CCC = 3ABC$ or $A^3 + B^3 + C^3 - 3ABC$, viz., $5M[ABC]$; and Eight Root-squaring Transformations $A^2 - B^2$, viz., $8M[A^2]$ on a Computer with Parallel Arithmetic	13
V.	Expansion of Circulants of Order p ($2 \leq p \leq 8$)	44

THE ROOT-CUBING AND THE GENERAL ROOT-POWERING METHODS FOR FINDING THE ZEROS OF POLYNOMIALS

by

Erwin H. Bareiss

ABSTRACT

The root-squaring and root-cubing method is generalized and called the general root-powering method. A simple representation, suitable for coding, is obtained by introducing partitioned polynomials. The method is extended to include analytic functions.

1. Motivation

The root-squaring method, in conjunction with the resultant procedures,²⁻⁵ has proven reliable and very fast in practical applications. However, to decide whether the real roots of a polynomial are positive or negative, we must enter a test routine. Such a test routine is extraneous to the elegance of the resultant procedures. Testing of the sign or real roots can (probably always) be eliminated when we replace the root-squaring process by a root-cubing process.

The general root-powering process, using partitioned polynomials, was developed for pure mathematical curiosity. Because of the harmonious result, it may well become a practical method in the near future. Also the algorithms presented in the last two sections may become of practical value.

PART I. ROOT-CUBING AND ROOT-SQUARING METHODS

2. The Root-cubing Method

Given is the polynomial equation

$$P_n(x) \equiv \sum_{i=0}^n a_i x^i = 0. \quad (2.1)$$

It is required that $P_n(x)$ be transformed into a polynomial $P_n^{(3)}(z)$, $z = x^3$, such that

$$P_n^{(3)}(z) \equiv \sum_{i=0}^n a_i^{(3)} x^{3i} = 0 \quad \text{whenever } P_n(x) = 0, \quad z = x^3. \quad (2.2)$$

The transformation is obtained as follows. Let

$$A \equiv \sum_{i=0}^{\left[\frac{n}{3}\right]} a_{3i} x^{3i},$$

$$B \equiv x \sum_{i=0}^{\left[\frac{n-1}{3}\right]} a_{1+3i} x^{3i},$$

$$C \equiv x^2 \sum_{i=0}^{\left[\frac{n-2}{3}\right]} a_{2+3i} x^{3i},$$
(2.3)

and

$$C \equiv x^2 \sum_{i=0}^{\left[\frac{n-2}{3}\right]} a_{2+3i} x^{3i},$$

where $[n/3]$ is the greatest integer not greater than $n/3$ and so on. Hence, by (2.1) and (2.3),

$$P_n(x) \equiv A + B + C = 0. \quad (2.4)$$

The set of all polynomials forms a commutative ring. Thus, applying the polynomial theorem to $(A + B + C)^3$ and then simplifying yields

$$\begin{aligned} (A + B + C)^3 &\equiv A^3 + B^3 + C^3 + 3[(A^2B + AB^2) + (A^2C + AC^2) + (B^2C + BC^2)] \\ &\quad + 6ABC \\ &\equiv A^3 + B^3 + C^3 + 3(AB + AC + BC)(A + B + C) - 3ABC. \end{aligned}$$

Since by assumption (2.4), $A + B + C = 0$, we have

$$A^3 + B^3 + C^3 - 3ABC \equiv P_n^{(3)}(z), \quad (2.5)$$

which is the desired polynomial, $P_n^{(3)}(z)$. This transformation was already known to Encke.¹³ Because of the definition of A, B, C by (2.3), it is easily seen that $P_n^{(3)}(z)$ is a polynomial of degree n in z , where $z = x^3$.

An even shorter derivation of (2.5) is as follows. As in (2.4) let

$$P_n(x) = A + B + C = 0.$$

Then

$$\begin{aligned} B + C &= -A \\ C + A &= -B \\ A + B &= -C. \end{aligned} \quad (2.6)$$

The product of the left sides of these equations yields

$$\begin{aligned}(A + B)(B + C)(C + A) &= A^2(B + C) + B^2(C + A) + C^2(A + B) + 2ABC \\ &= -(A^3 + B^3 + C^3) + 2ABC.\end{aligned}$$

Equating this result to the product of the right sides of equations (2.6), $-ABC$, and combining equal terms, yields

$$A^3 + B^3 + C^3 - 3ABC = 0,$$

which is again the root-cubing transformation, (2.5).

If we transform $P_n^{(3)}(z)$ in a similar way, we obtain a polynomial $P_n^{(9)}(z'')$ of order n in z'' , where $z'' = z^3 = (x^3)^3 = x^{3^2}$, and so on. Thus, if $(-\alpha)$ is a root of (2.1), then

$$P_n(-\alpha) = 0, P_n^{(3)}(-\alpha^3) = P_n^{(3^2)}(-\alpha^9) = 0, \dots, P_n^{(3^m)}(-\alpha^{3^m}) = 0.$$

The transformation defined by (2.5) is easily accomplished by a subroutine that multiplies polynomials. The polynomials A , B/x , C/x^2 , defined by (2.3), are only of order $[n/3]$, $[(n-1)/3]$, $[(n-2)/3]$, respectively, in x^3 . If the computer has at least four independent arithmetic units, the coefficients of

$$A^3 = AA^2, B^3/x^3 = (B/x)(B/x)^2, C^3/x^6 = (C/x^2)(C/x^2)^2, ABC/x^3 = A(B/x)(C/x^2)$$

can be calculated concurrently. Thus, the root-cubing process is suited also for parallel-controlled multiprocessors.

More sophisticated codes for the root-cubing transformation contain subroutines that calculate A^3 , B^3 , and C^3 , directly, thus reducing the numerical work. Such codes will be considered in the next section.

3. Comparison with the Root-squaring Method

To compare the root-cubing transformation as defined by (2.5) with the root-squaring transformation, we write, instead of (2.4),

$$P_n(x) \equiv A + B = 0, \quad (3.1)$$

where

$$A \equiv \sum_{i=0}^{\left[\frac{n}{2}\right]} a_{2i} x^{2i},$$

and

$$B = x \sum_{i=0}^{\left[\frac{n-1}{2}\right]} a_{1+2i} x^{2i}. \quad (3.2)$$

Since $A^2 - B^2 \equiv (A + B)(A - B) = 0$, it is easily shown that, for $y' = x^2$,

$$P_n^{(2)}(y') \equiv A^2 - B^2 = 0 \quad (3.3)$$

leads to the usual root-squaring algorithm, also known as the Graeffe process.* Here, the powers of the moduli in successive transformations are 2, 4, 8, 16, 32, ..., as opposed to 3, 9, 27, 81, 243, ..., for the root-cubing method. Thus to obtain equal separation in the moduli of the roots, the number of transformations M_2 and M_3 for the root-squaring and -cubing methods, respectively, are related by

$$2^{M_2} = 3^{M_3}$$

or

$$M_2 = \frac{\log 3}{\log 2} M_3 = 1.585 M_3. \quad (3.4)$$

For example, let $\rho = |\alpha|$ be the modulus of any root, then ρ^{256} is obtained in eight squaring transformations, while ρ^{243} is obtained in only five cubing transformations. Similarly, ρ^{2048} is obtained in 11 squaring transformations, while ρ^{2187} is obtained in only seven cubing transformations.

The polynomials A and B are of order $[n/2]$ and $[(n-1)/2]$ while A , B , C are only of order $[n/3]$, $[(n-1)/3]$, $[(n-2)/3]$, respectively. Thus, if we use a polynomial multiplier subroutine to perform the transformations, the root-squaring method (3.3) yields

$$\left(\left[\frac{n+2}{2}\right]\right)^2 + \left(\left[\frac{n+1}{2}\right]\right)^2$$

terms, whereas the root-cubing method (2.5) yields certainly less than

$$\left(\left[\frac{n+3}{3}\right]\right)^3 + \left(\left[\frac{n+2}{3}\right]\right)^3 + \left(\left[\frac{n+1}{3}\right]\right)^3 + \left[\frac{n+3}{3}\right] \cdot \left[\frac{n+2}{3}\right] \cdot \left[\frac{n+1}{3}\right]$$

terms. These expressions give, however, only a very rough estimate of the amount of arithmetic involved, since simplifications are possible in both cases.

Depending on the method of calculating the transformations, the number of multiplications will be different. To simplify our representation, we introduce the following convention. Let $M[\]$ be the number of *full multiplications*** necessary to calculate the coefficients of the polynomial resulting from the operation indicated in brackets. To illustrate, assume A , B , and C are polynomials of order $m-1$, therefore having m coefficients each. Then, $M[AB] = m^2$. Let us furthermore agree that $M[A^p]$ implies evaluation of A^p by the *polynomial theorem*, and not by repeated multiplication of A . Thus,

*Another derivation of (3.3) is as follows: Because of (3.1), $A = -B$; thus, $A^2 = B^2$, which implies (3.3).

**Multiplication of a polynomial by an integer is not a "full" multiplication. Thus, in (3.5), $M[AB] = M[2AB]$, and so on.

$$\begin{aligned}
M[AA] &= M[AB] = m^2, \\
M[A^2] &= m + \binom{m}{2} = m(m+1)/2 = \frac{1}{2}(m^2 + m), \\
M[AAA] &= M[ABC] = m^2 + m(2m-1) = (3m-1)m, \\
M[A^2A] &= M[AA^2] = M[AB^2] = m(m+1)/2 + m(2m-1) = (5m-1)m/2, \\
M[A^3] &= 2m + m(m-1) + \left[\binom{m-1}{2} + \binom{m}{3} \right] = (m^3 + 6m^2 - m + 6)/6.
\end{aligned} \tag{3.5}$$

Note that $M[AAA] = M[ABC] < 3m^2$, while $M[A^3] > m^3/6$, but $M[A^2A] = M[AA^2] < (5/2)m^2$. The formula for $M[ABC]$ is obvious, since BC is a polynomial with $2m-1$ coefficients. To calculate BC requires m^2 full multiplications. Thus to calculate $A(BC)$ requires an additional $m(2m-1)$ full multiplications.

To derive the formula for $M[A^3]$, let

$$A = \sum_{i=0}^{m-1} a_i x^i.$$

Then*

$$A^3 = \sum_{i=0}^{m-1} a_i^3 x^{3i} + 3 \sum_{i < j} (a_i^2 a_j x^{2i+j} + a_i a_j^2 x^{i+2j}) + 6 \sum_{i < j < k} a_i a_j a_k x^{i+j+k}.$$

The calculation of all a_i^2 requires m full multiplications. Thus the calculation of all a_i^3 under the first summation sign requires $2m$ full multiplications. There are $\binom{m}{2}$ terms, each of the types $a_i^2 a_j$ and $a_i a_j^2$. Thus the evaluation of the second summation sign requires an additional $2\binom{m}{2} = m(m-1)$ full multiplications. The third summation term includes $\binom{m}{3}$ different terms of type $a_i a_j a_k$. There are $\binom{m-1}{2}$ different terms $a_i a_j$ ($0 \leq i < j \leq m-2$) requiring one full multiplication each. To calculate all terms of type $(a_i a_j) a_k$ requires an additional $\binom{m}{3}$ full multiplications. This concludes the derivation of the formula.

Numerical values for the formulas (3.5) are given in Table I. It is seen that $M[A^3] < M[AA^2]$ for polynomials up to order 7 ($m=8$). Only for very large values of m will the asymptotic behavior (as indicated in the last column) apply.

TABLE I. Numerical Values for Formulas (3.5)
(A, B, and C are polynomials of order $m-1$)

	$m =$	1	2	3	4	5	6	7	8	9	$m^\dagger$
1	$M[AA] = M[AB]$	1	4	9	16	25	36	49	64	81	m^2
2	$M[A^2]$	1	3	6	10	15	21	28	36	45	$>m^2/2$
3	$M[AAA] = M[ABC]$	2	10	24	44	70	102	140	184	234	$<3m^2$
4	$M[AA^2] = M[AB^2]$	2	9	21	38	60	87	119	156	198	$<5m^2/2$
5	$M[A^3]$	2	6	14	27	46	72	106	149	202	$>m^3/6$

† Asymptotic values for large m .

*If A is identified with A of Eq. (2.3), i.e., $A = A$, then $a_i x^i \Rightarrow a_{3i} x^{3i}$.

We return to the calculation of the root-cubic transformation of $P_n = A + B + C$, $P_n^{(3)} = A^3 + B^3 + C^3 - 3ABC$. To simplify the formulas, we assume $n = 3m - 1$ ($m = 1, 2, \dots$). Then A , B , and C have m coefficients each. The *simplest* program, requiring only a subroutine that multiplies polynomials, demands for one complete transformation

$$\begin{aligned} M[AAA + BBB + CCC - 3ABC] &= M[AAA] + M[BBB] + M[CCC] + M[ABC] = 4M[ABC] \\ &= 4m(3m - 1) = \frac{4}{3}(n + 1)n \end{aligned} \quad (a)$$

full multiplications. A more efficient way to calculate the transformation demands

$$\begin{aligned} M[AAA + B(BB - 3AC) + CCC] &= 3M[ABC] + M[BB] = (10m - 3)m \\ &= \frac{(10n + 1)(n + 1)}{9} \end{aligned} \quad (b)$$

full multiplications. A formula requiring, in addition, a *polynomial squaring routine* demands*

$$M[AA^2 + B(B^2 - 3AC) + CC^2] = 3M[AA^2] + M[AC] = \frac{(17n + 8)(n + 1)}{18} \quad (c)$$

full multiplications. A formula requiring, in addition, a *polynomial cubing routine* demands

$$\begin{aligned} M[A^3 + B(B^2 - 3AC) + C^3] &= 2M[A^3] + M[B^2] + M[ABC] \\ &= \frac{2(n + 1)^3 + 99(n + 1)^2 - 45(n + 1)}{162} + 2 \end{aligned} \quad (d)$$

full multiplications. Finally,

$$\begin{aligned} M[A^3 + B^3 + C^3 - 3ABC] &= 3M[A^3] + M[ABC] \\ &= \frac{(n + 1)^3 + 36(n + 1)^2 - 27(n + 1)}{54} + 3. \end{aligned} \quad (e)$$

Numerical values for different degrees of P_n are given in Table II. The number of coefficients in the first four polynomials is a multiple of 6, so that later comparison with root-squaring transformations is simplified. Also, for comparison purposes, the last line gives the *least* number of full multiplications necessary if a method is used as proposed by Aitken.¹ He obtained the root-cubing transformation by

$$P_n^{(3)}(x^3) = P_n(x)[P_n(\omega x)P_n(\omega^2 x)],$$

*Note that here $\frac{17}{18}n^2 < M < \frac{17}{18}(n + 1)^2$, thus requiring less than $(n + 1)^2$ full multiplications.

where ω is a complex third root of unity. The formula that furnishes this minimum number of full multiplications is*

$$M[Aitken] = \frac{(n+1)(7n+8)}{6}. \quad (f)$$

For the most common polynomials (of degree < 26), method (d) requires the least number of full multiplications. This has to be weighted with the simplicity of coding the transformations. Method (a) is the most simple, next to method (b). Then comes method (c). Aitken's method (f) rates about equal to methods (d) and (e) in complexity. It therefore follows that methods (b) and (d) are of more practical interest than methods (e) and (f). Since the complete solution of most polynomials up to degree 15 can be accomplished within 1 sec on modern electronic computers (like the CDC 3600 or IBM 7090), simplicity of a code may often be more important than speed, and thus method (a) should not be discarded. It should be mentioned here also, that the numerical stability or error propagation is, for practical purposes, equal for all six methods. This point will be explained in the simpler case of root squaring below.

TABLE II. Number of Full Multiplications Required for
Root-cubing Transformations of $P_n = A + B + C$
Using Different Methods

($n = 3m - 1$)

Method	Degree n of Polynomial	5	11	17	23	26	Large
	m	2	4	6	8	9	
(a)	$M[AAA + BBB + CCC - 3ABC]$	40	176	408	736	936	$> \frac{4n^2}{3}$
(b)	$M[AAA + B(BB - 3AC) + CCC]$	34	148	342	616	783	$> \frac{10n^2}{9}$
(c)	$M[AA^2 + B(B^2 - 3AC) + CC^2]$	31	130	297	532	<u>675</u>	$> \frac{17n^2}{18}$
(d)	$M[A^3 + B(B^2 - 3AC) + C^3]$	<u>25</u>	<u>108</u>	<u>267</u>	<u>518</u>	683	$>> \frac{(n+1)^3}{81}$
(e)	$M[A^3 + B^3 + C^3 - 3ABC]$	28	125	318	631	840	$>> \frac{(n+1)^3}{54}$
(f)	$M[Aitken]$	43	170	381	676	855	$> \frac{7(n+1)^2}{6}$

*We omit the proof for this formula, since Aitken's method is in no way optimal. A modification of Aitken's method for electronic computers by Busk¹⁰ increases the number of full multiplications to more than $(5/2)n^2$.

We turn now briefly to the calculation of the root-squaring transformation of $P_n = A + B$, $P_n^{(2)} = A^2 - B^2$. To simplify the formulas, we assume $n = 2m - 1$ ($m = 1, 2, \dots$). Then A and B have m coefficients each. The *simplest* program, requiring only a subroutine that multiplies polynomials, demands for one complete transformation

$$M[AA - BB] = 2M[AA] = 2m^2 = \frac{(n+1)^2}{2}$$

full multiplications. A formula requiring, in addition, a *polynomial squaring routine* demands only

$$M[A^2 - B^2] = 2M[A^2] = m(m+1) = \frac{(n+1)(n+3)}{4}$$

full multiplications. Although the number of full multiplications in the second formula is only about half the number of those in the first formula, the numerical stability or error propagation is the same in both methods of calculation. To see this, let $A = a + bz + cz^2 + \dots$ ($z = x^2$). Then $AA = a^2 + abz + baz + b^2z^2 + \dots = a^2 + (ab + ba)z + \dots$, while $A^2 = a^2 + 2abz + \dots$. Assuming that in the computer used, $ab = ba$, and $ab + ab = 2ab$, the errors in both methods are the same.* These ideas can be applied to the calculations of AAA , AA^2 , and A^3 , and to the methods discussed in the previous section.

We are now prepared to compare the efficiency of the root-squaring and the root-cubing transformations. We have seen in (3.4) that theoretically, the number of squaring transformations, M_2 , is 1.585 (or approximately $8/5$) times the number of cubing transformations, M_3 . In practical applications, eight squaring transforms are sufficient in most cases. This would correspond to five cubing transformations. Thus, a meaningful comparison can be obtained by comparing, say $8M[AA - BB]$ with $5M[AAA + B(BB - 3AC) + CCC]$, assuming that only polynomial multiplication is used. Although the formulas are simple, numerical examples are more illuminating and are presented in Table III. In addition, the numerical values for

TABLE III. Number of Full Multiplications to Perform Eight Root-squaring or Five Root-cubing Operations for Various Degrees n of the Polynomial and Different Methods of Calculation (Sequential computation)

Method	Degree of Polynomial	5	11	17	23
12	$8M[AA - BB]$	144	576	1296	2304
13	$5M[AAA + B(BB - 3AC) + CCC]$	170	740	1710	3080
22	$8M[A^2 - B^2]$	96	336	720	1248
23	$5M[A^3 + B(B^2 - 3AC) + C^3]$	125	540	1335	2590

*In the IBM 360 system, using double precision, xy equals yx , but the rounding error $\epsilon(z + z) \leq \epsilon(2z)$, unless $2z$ is obtained by a shift operation, in which case $z + z = 2z$. Similarly, a factor 3 should be obtained either by $(z + z + z)$, or by one shift operation and one addition, to minimize the rounding error.

$8M[A^2 - B^2]$ and $5M[A^3 + B(B^2 - 3AC) + C^3]$ are listed. These last two formulas represent the minimum number of "full" multiplications required for respective equivalent root-squaring and -cubing transformations. The analyst has to weigh the different methods in the light of the purpose of the computer program. In particular, he must decide whether the increased number of multiplications in the root-cubing transformations justifies the savings obtained by the omission of a root-testing subroutine as always required by the root-squaring transformations.

So far, our considerations were focused only on *sequential arithmetic*. If we have a computer working with *parallel arithmetic*, the picture changes. Assume we have four arithmetic units at our disposition. Then, as stated in the previous section, A^3 , B^3 , C^3 , and ABC can be computed simultaneously. Since for $n \leq 25$, $M[ABC] \geq M[AA^2] \geq M[A^3]$, the significant number of multiplications is $M[ABC]$. In the root-squaring transformation, $A^2 - B^2$, the significant number of multiplications is either $M[AA]$ or $M[A^2]$, depending on the method chosen. Thus, to compare speed on computers with parallel arithmetic, the numbers of Table III must be replaced by the numbers of Table IV.

TABLE IV. Maximum Number of Full Multiplications Required per Arithmetic Unit to Perform Eight Root-squaring Transformations $AA - BB$, viz., $8M[AA]$; Five Root-cubing Transformations $AAA + BBB + CCC - 3ABC$ or $A^3 + B^3 + C^3 - 3ABC$, viz., $5M[ABC]$; and Eight Root-squaring Transformations $A^2 - B^2$, viz., $8M[A^2]$ on a Computer with Parallel Arithmetic (Parallel computation)

Degree of Polynomial, n	5	11	17	23
$8M[AA]$	72	288	648	1152
$5M[ABC]$	50	220	510	920
$8M[A^2]$	48	168	360	624

We see that when simplicity is preferred in parallel computation, i.e., if we restrict the program to the use of a polynomial multiplier subroutine only, the root-cubing transformation is faster indeed.

For the squaring and cubing processes, pivotal coefficients can be recognized by the same general method as developed in Ref. 2 or 17. However, a more detailed study of the root-cubing method may lead to substantial new insights, which can be applied in future codes for solving polynomial equations. The few examples in the following section clearly demonstrate important differences between root squaring and root cubing in the coefficient configuration of the transformed polynomials.

Although we consider the existing resultant procedures as the fastest and most reliable general codes for solving polynomial equations of degree up to 20 (and higher), the replacement of the root-squaring transformation by the root-cubing transformation opens the possibility of further improvement. In particular, when parallel computation is available, we may desire a simple subroutine that allows both root-squaring and root-cubing transformations.

4. Extracting Roots from Polynomial Equations.

The aim of successive root-cubing transformations is to separate roots of unequal moduli, so that after an appropriate number of transformations, Newton's symmetric functions can be approximated with only a few terms, preferably with only a single term. In this section, we denote by α_i ($i = 0, 1, 2, \dots, n-1$) the *negatives* of the roots of (2.1), so that $P_n(-\alpha_i) = 0$. Furthermore, assume $\alpha_n = 1$, and $P_n(x)$ to be real.

Then the coefficients a_i of the polynomial $P_n(x)$ of (2.1) are related to the roots by

$$\begin{aligned} a_n &= 1 \\ a_{n-1} &= \alpha_0 + \alpha_1 + \dots + \alpha_{n-1} \\ a_{n-2} &= \alpha_0\alpha_1 + \alpha_0\alpha_2 + \dots + \alpha_{n-2}\alpha_{n-1} \\ &\vdots \\ \alpha_0 &= \alpha_0\alpha_1\alpha_2 \dots \alpha_{n-1}. \end{aligned}$$

We discuss here only a few very simple cases.

a) Assume

$$|\alpha_{n-1}| > |\alpha_{n-2}| > \dots > |\alpha_0|. \quad (4.1)$$

Assume further that m cubing transformations have been performed such that

$$|\alpha_{n-1}|^{3m} \gg |\alpha_{n-2}|^{3m} \gg \dots \gg |\alpha_0|^{3m}.$$

Let

$$\beta_k = \alpha_k^{3m}, \text{ and } b_k = \alpha_k^{(m)}. \quad (4.2)$$

Then

$$\begin{aligned} b_n &= 1 \\ b_{n-1} &\approx \beta_{n-1} \\ b_{n-2} &\approx \beta_{n-1}\beta_{n-2} \\ b_{n-3} &\approx \beta_{n-1}\beta_{n-2}\beta_{n-3} \\ &\vdots \\ b_0 &= \beta_{n-1}\beta_{n-2} \dots \beta_0. \end{aligned} \quad (4.3)$$

Thus,

$$\beta_k \approx \frac{b_k}{b_{k+1}}, \quad (4.4)$$

and by (4.2),

$$\alpha_k = \beta_k^{1/(3^m)}. \quad (4.5)$$

A simple example will illustrate the results obtained so far. Take, for (2.1),

$$P_2 = 2 + 3x + x^2 = 0. \quad (4.6)$$

Then the decomposition (2.4) is given by (2.3) as

$$A = 2; \quad B = 3x; \quad C = x^2.$$

The transformation (2.5) yields

$$P'_2 = 8 + 9x^3 + x^6.$$

If P'_2 is transformed again, we have

$$P''_2 = 512 + 513x^9 + x^{18}. \quad (4.7)$$

At this point, $m = 2$. Thus by (4.4),

$$\beta_0 \approx \frac{b_0}{b_1} = \frac{512}{513} = 0.99805,$$

and

$$\beta_1 \approx \frac{b_1}{b_2} = \frac{513}{1} = 513.$$

Since $m = 2$, we have, by (4.5),

$$\alpha_0 = \beta_0^{1/(3^2)} = (0.99805)^{1/9} = 0.998,$$

and

$$\alpha_1 = \beta_1^{1/(3^2)} = 513^{1/9} = 2.004.$$

Therefore, after two cubing transformations, the approximate roots of (4.6) are -0.9998 and -2.004. The true roots are -1 and -2.

b) Assume that in a) $\alpha_{n-1} = \alpha_{n-2}$. Then

$$b_n = 1; \quad b_{n-1} \approx 2\beta_{n-1}; \quad b_{n-2} \approx \beta_{n-1}^2; \quad b_{n-3} \approx \beta_{n-1}^3 \beta_{n-3}; \quad \text{etc.} \quad (4.8)$$

Thus α_{n-1} is given by

$$\alpha_{n-1} \approx \left(\frac{b_{n-1}}{2} \right)^{1/(3^m)} \quad \text{and} \quad \alpha_{n-1} = \pm (b_{n-2})^{1/2(3^m)}.$$

c). Assume that, in a), $\alpha_{n-1} = -\alpha_{n-2}$. Then

$$b_n = 1; b_{n-1} \approx 0; b_{n-2} \approx -\beta_{n-1}^2; b_{n-3} \approx -\beta_{n-1}^2 \beta_{n-3}; \text{etc.} \quad (4.9)$$

Thus the configuration $b_{n-1} \approx 0$, $b_{n-2} < 0$ indicates that $\alpha_{n-2} = -\alpha_{n-1}$, where

$$\alpha_{n-1} \approx (-b_{n-2})^{1/2(3^m)}.$$

d) Assume that, in a), $\alpha_{n-1} = \overline{\alpha_{n-2}}$. Then

$$b_n = 1; b_{n-1} \approx 2\operatorname{Re}(\beta_{n-1}); b_{n-2} \approx |\beta_{n-1}|^2; \text{etc.} \quad (4.10)$$

Thus, $|\alpha_{n-1}| \approx (b_{n-2})^{1/2(3^m)}$. To obtain $\operatorname{Re}(\alpha_{n-1})$, we transfer to the resultant procedure. Case d) includes b), but it does not include c), because a pure imaginary α_{n-1} yields the configuration $b_{n-1} \approx 0$, $b_{n-2} > 0$.

In this manner, we can continue to investigate all possible cases as was done for root squaring in Ref. 3.

It is important to note the the configuration $b_n; b_{n-1}; b_{n-2}; \text{etc.}$, of (4.8) obtained in b) and characterized by $b_{n-1} = 1$, $b_{n-2} \approx (b_{n-1}/2)^2$ is only necessary but not sufficient to warrant the existence of real roots, since with every transformation extraneous roots are introduced in the new equation. Thus if, in d), $\alpha_{n-1} = \rho e^{i\phi}$ with

$$\phi = \ell\pi/3^k \quad (1 \leq k \leq m; 1 \leq \ell < 3^k),$$

then

$$\begin{aligned} \beta_{n-1} &= \alpha_{n-1}^{3^m} = \rho^{3^m} \exp(3^{m-k} \ell \pi i) \\ &= \begin{cases} \rho^{3^m} & \text{for } \ell \text{ even} \\ -\rho^{3^m} & \text{for } \ell \text{ odd} \end{cases} = \overline{\alpha_{n-1}^{3^m}}, \end{aligned}$$

and therefore

$$b_n = 1; b_{n-1} \approx \pm 2\rho^{3^m}; b_{n-2} \approx \rho^{2 \cdot (3^m)},$$

which is the configuration (4.8) of case b). The possible appearance of extraneous real roots in transformed equations is also present in the root-squaring method. After the acceptance test (as used in the existing resultant procedures) is eliminated, the recognition of the extraneous roots becomes important. A detailed investigation on this subject is needed.

5. The Root-bisquaring Transformation

The success of reducing the root-squaring and root-cubing transformations to simple multiplications and additions of partitioned polynomials defined by

$$A_{p,k} = x^k \sum_{i=0}^{\left[\frac{n-k}{p}\right]} a_{pi+k} x^{pi} \quad (k = 0, 1, \dots, p-1) \quad (5.1)$$

induces the question of whether similar transformations exist that yield polynomials whose zeros are the p th powers of the roots of (2.1), when $p \geq 4$.

For $p = 4$, this leads to the root-bisquaring transformation. To establish it, let, in (5.1),

$$A = A_{4,0}, B = A_{4,1}, C = A_{4,2}, D = A_{4,3}. \quad (5.2)$$

Then (2.1) is

$$P_n(x) = A + B + C + D = 0,$$

from which follows the root-squaring transformation,

$$(A + C)^2 - (B + D)^2 = 0. \quad (5.3)$$

Applying the root-squaring transformation on the explicit form of the polynomial (5.3) again, yields

$$P'_n(z) = (A^2 + C^2 - 2BD)^2 - (B^2 + D^2 - 2AC)^2 = 0, \quad (5.4)$$

where $z = x^4$. Equation (5.4) can be arranged in different ways, as for example,

$$P'_n(z) = (A^2 - C^2)^2 - (B^2 - D^2)^2 + 4(AB - CD)(BC - DA). \quad (5.5)$$

In a similar way, we can obtain formulas for $p = 6$ by combining root squaring ($p = 2$) and root cubing ($p = 3$); or formulas for any $p = 2^k 3^l$, ($k, l = 0, 1, 2, \dots$). These formulas will be of increasing complexity.

The generalization of the root-squaring and root-cubing transformations to any p th power of roots transformation can be achieved elegantly over the ring of the coefficients of $P_n(x)$, $\{a_0, a_1, \dots, a_n\}$ and over the ring of the partitioned polynomials (5.1). Since the generalized transformation is not of the same degree of immediate practical interest as the root-squaring and root-cubing process, and the method of proof uses different tools, we have developed the *general root-powering transformation* independently in Part II. We only note here that for $p > n$, the set of partitioned polynomials $A_{p,k}$ of (5.1) is simply

The sums of powers of the α 's are denoted by

$$s_p = \alpha_1^p + \alpha_2^p + \dots + \alpha_n^p \quad (p = 1, 2, 3, \dots). \quad (6.5)$$

Thus, in particular,

$$s_1 = c_1.$$

Newton's formulas* are

$$a_0 s_1 + 1a_1 = 0,$$

$$a_0 s_2 + a_1 s_1 + 2a_2 = 0,$$

$$a_0 s_3 + a_1 s_2 + a_2 s_1 + 3a_3 = 0, \quad (6.6)$$

$$\dots \quad \dots \quad \dots \quad \dots$$

$$a_0 s_p + a_1 s_{p-1} + \dots + a_{p-1} s_1 + p a_p = 0 \quad (p = 1, 2, 3, \dots).$$

They are obtained by comparing equal powers of x in

$$\frac{d}{dx} P(x) = n a_0 x^{n-1} + (n-1) a_1 x^{n-2} + \dots + a_{n-1}$$

and

$$\frac{d}{dx} P(x) = \frac{P(x)}{x - \alpha_1} + \dots + \frac{P(x)}{x - \alpha_n}$$

for $p = 1, 2, \dots, n-1$ and from

$$\sum_{i=1}^n \alpha_i^{p-n} (a_0 \alpha_i^n + a_1 \alpha_i^{n-1} + \dots + a_n) = 0$$

for $p = n, n+1, n+2, \dots$

Because of Newton's formulas, the sum of powers of the roots, s_p , is an algebraic function of the coefficients $a_0, \dots, a_n$ whose coefficients are integers, and can be written as a determinant of order p (see Ref. 9):

$$s_p = -\left(\frac{1}{a_0}\right)^p \begin{vmatrix} a_1 & a_0 & 0 & \dots & 0 \\ 2a_2 & a_1 & a_0 & \dots & \cdot \\ \cdot & \cdot & \cdot & \dots & \cdot \\ (p-1)a_{p-1} & a_{p-2} & a_{p-3} & \dots & a_0 \\ p a_p & a_{p-1} & a_{p-2} & \dots & a_1 \end{vmatrix}. \quad (6.7)$$

*These and the following formulas can be extended to negative powers of the α 's, $p < 0$. To do so, set $\beta_i = \alpha_i^{-1}$, replace $P(x)$ by $y^n P(1/y) = a_n y^n + \dots + a_0 = 0$, and proceed as above.

Conversely, the coefficients a_i can be expressed by the s_i :

$$(-1)^k k! a_k = a_0 \begin{vmatrix} s_1 & 1 & 0 & \dots & 0 \\ s_2 & s_1 & 2 & \dots & \cdot \\ \cdot & \cdot & \cdot & \cdot & (k-1) \\ s_k & s_{k-1} & s_{k-2} & \dots & s_1 \end{vmatrix}. \quad (6.8)$$

The explicit expressions of (6.7) and (6.8) are *Waring's formulas* (1762),

$$s_p = p \left(-\frac{1}{a_0} \right)^p \sum_{\{\lambda\}} \frac{(-1)^{\lambda_0} (p - \lambda_0 - 1)!}{\lambda_1! \dots \lambda_n!} a_0^{\lambda_0} a_1^{\lambda_1} \dots a_n^{\lambda_n}, \quad (6.9)$$

where the summation extends over all possible combinations of nonnegative integers $\lambda_0, \dots, \lambda_n$ which satisfy

$$\lambda_0 + \lambda_1 + \dots + \lambda_n = p \quad (6.10)$$

$$\lambda_1 + 2\lambda_2 + \dots + n\lambda_n = p$$

and

$$a_k = a_0 \sum \frac{(-1)^{\lambda_1 + \dots + \lambda_k}}{\lambda_1! \dots \lambda_k!} \left(\frac{s_1}{1} \right)^{\lambda_1} \dots \left(\frac{s_k}{k} \right)^{\lambda_k} \quad (k = 1, 2, \dots, m), \quad (6.11)$$

where the summation extends over all combinations of nonnegative integers $\lambda_1, \dots, \lambda_k$ such that

$$\lambda_1 + 2\lambda_2 + \dots + k\lambda_k = k. \quad (6.12)$$

Because of (6.3) and (6.4), Eqs. (6.7) and (6.8) can be written in terms of c 's only:

$$s_p = \begin{vmatrix} c_1 & 1 & 0 & 0 & \dots & 0 \\ 2c_2 & c_1 & 1 & 0 & \dots & 0 \\ 3c_3 & c_2 & c_1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ pc_p & c_{p-1} & c_{p-2} & \dots & c_1 \end{vmatrix} \quad (6.13)$$

$$k! c_k = \begin{vmatrix} s_1 & 1 & 0 & 0 & \dots & 0 \\ s_2 & s_1 & 2 & 0 & \dots & 0 \\ s_3 & s_2 & s_1 & 3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ s_k & s_{k-1} & s_{k-2} & \dots & s_1 \end{vmatrix}. \quad (6.14)$$

Then, (6.9) and (6.11) become

$$s_p = (-1)^p p! \sum \frac{(-1)^{\lambda_1 + \dots + \lambda_n} (\lambda_1 + \dots + \lambda_n - 1)!}{\lambda_1! \dots \lambda_n!} c_1^{\lambda_1} \dots c_n^{\lambda_n}, \quad (6.15)$$

under the condition $\lambda_1 + 2\lambda_2 + \dots + n\lambda_n = p$, where $\lambda_i \geq 0$ and integer;

$$c_k = (-1)^k \sum \frac{(-1)^{\lambda_1 + \dots + \lambda_k}}{\lambda_1! \dots \lambda_k!} \left(\frac{s_1}{1}\right)^{\lambda_1} \dots \left(\frac{s_k}{k}\right)^{\lambda_k},$$

under the condition $\lambda_1 + 2\lambda_2 + \dots + k\lambda_k = k$, where $\lambda \geq 0$ and integer.

7. The General Algorithm for the Root-powering Method

With the formulas of the preceding section it is easy to construct from (6.1) a polynomial whose zeros are the p th power of the α_i . The elementary symmetric functions of this polynomial are then

$$\begin{aligned} c_1^{(p)} &= \alpha_1^p + \alpha_2^p + \dots + \alpha_n^p, \\ c_2^{(p)} &= \alpha_1^p \alpha_2^p + \alpha_1^p \alpha_3^p + \dots + \alpha_{n-1}^p \alpha_n^p, \\ c_n^{(p)} &= \alpha_1^p \dots \alpha_n^p. \end{aligned} \quad (7.1)$$

The sum of the powers of the (α_i^p) 's are now, by (6.5),

$$s_{kp} = (\alpha_1^p)^k + \dots + (\alpha_n^p)^k, \quad (7.2)$$

whence $c_k^{(p)}$ is given by (6.8) as

$$k! c_k^{(p)} = \begin{vmatrix} s_p & 1 & 0 & 0 & \dots & 0 \\ s_{2p} & s_p & 2 & 0 & \dots & 0 \\ s_{3p} & s_{2p} & s_p & 3 & \dots & \\ \dots & \dots & \dots & \dots & \dots & \\ s_{kp} & s_{(k-1)p} & s_{(k-2)p} & \dots & \dots & s_p \end{vmatrix}. \quad (7.3)$$

For example,

$$c_1^{(p)} = s_p,$$

$$2c_2^{(p)} = s_p^2 - s_{2p},$$

and so on. Thus, we obtain the following procedure to construct an algorithm of the general root-powering method.

a) Compute s_{kp} ($k = 1, 2, \dots, n$) as a function of the coefficients of the given polynomial by (6.13), viz.,

$$s_{kp} = \begin{vmatrix} c_1 & 1 & 0 & \dots & 0 \\ 2c_2 & c_1 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ k_p c_{kp} & c_{kp-1} & c_{kp-2} & \dots & c_1 \end{vmatrix}, \quad (7.4)$$

where, by (6.4), $c_i = 0$ for $i > n$. Instead of (6.13), we can use any one of these equations, (6.7), (6.9), or (6.15).

b) Use (7.3) to compute $c_k^{(p)}$ ($k = 1, 2, \dots, n$). Then the polynomial equation of degree n ,

$$\tilde{P}_n^{(p)}(x) = x^n - c_1^{(p)} x^{n-1} + \dots + (-1)^n c_n^{(p)} = 0, \quad (7.5)$$

has the roots $\alpha_i^{(p)}$ ($i = 1, 2, \dots, n$). Instead of (7.3), we can use (6.16) with s_k replaced by s_{kp} , and c_k by $c_k^{(p)}$; i.e.,

$$c_k^{(p)} = (-1)^k \sum \frac{(-1)^{\lambda_1 + \dots + \lambda_k}}{\lambda_1! \dots \lambda_k!} \left(\frac{s_p}{1} \right)^{\lambda_1} \dots \left(\frac{s_{kp}}{k} \right)^{\lambda_k}, \quad (7.6)$$

where $\lambda_1 + 2\lambda_2 + \dots + k\lambda_k = k$.

Examples: To illustrate the process, let us determine $c_3^{(2)}$. By (7.3) or (7.6), we have

$$c_3^{(p)} \equiv \sum \alpha_1^p \alpha_2^p \alpha_3^p = \frac{1}{6} (s_p^3 - 3s_p s_{2p} + 2s_{3p}).$$

For $p = 2$, we must determine s_2, s_4, s_6 from (7.4) or one of its alternatives. We obtain

$$\begin{aligned} s_2 &= c_1^2 - 2c_2, \\ s_4 &= c_1^2 - 4c_1^2 c_2 + 4c_1 c_3 + 2c_2^2 - 4c_4, \\ s_6 &= c_1^6 - 6c_1^4 c_2 + 9c_1^2 c_2^2 - 2c_2^3 + 6c_1^3 c_3 - 12c_1 c_2 c_3 + 3c_3^2 + 6c_2 c_4 \\ &\quad + 6c_1 c_5 - 6c_6. \end{aligned} \quad (7.7)$$

Since

$$c_3^{(2)} = \frac{1}{6} (s_2^3 - 3s_2 s_4 + 2s_6), \quad (7.8)$$

we find, by substituting (7.7) in the above equation and collecting similar terms, that

$$c_3^{(2)} = c_3^2 + 2c_1 c_5 - 2c_2 c_4 - 2c_6. \quad (7.9)$$

The following transformation formulas are given as additional examples and will serve for later reference.

$p = 2$: *Root-squaring Algorithm*

$$\begin{aligned} c_1^{(2)} &= c_1^2 - 2c_2, \\ c_2^{(2)} &= c_2^2 - 2(c_1c_3 - c_4), \\ c_3^{(2)} &= c_3^2 - 2(c_2c_4 - c_1c_5 + c_6), \\ c_4^{(2)} &= c_4^2 - 2(c_3c_5 - c_2c_6 + c_1c_7 - c_8), \end{aligned} \quad (7.10)$$

and so on.

$p = 3$: *Root-cubing Algorithm*

$$\begin{aligned} c_1^{(3)} &= c_1^3 - 3\{c_1c_2 - c_3\}, \\ c_2^{(3)} &= c_2^3 - 3\{c_1c_2c_3 - c_1^2c_4 - c_3^2 + c_2c_4 + c_1c_5 - c_6\}, \\ c_3^{(3)} &= c_3^3 - 3\{c_2c_3c_4 - c_1c_4^2 + c_1c_3c_5 - c_2^2c_5 + c_4c_5 + c_1c_2c_6 \\ &\quad - 2c_3c_6 - c_1^2c_7 + c_2c_7 + c_1c_8 - c_9\}, \end{aligned} \quad (7.11)$$

and so on.

$p = 4$: *Root-bisquaring Algorithm*

$$\begin{aligned} c_1^{(4)} &= c_1^4 + 2c_2^2 - 4(c_1^2c_2 - c_1c_3 + c_4), \\ c_2^{(4)} &= c_2^4 + 2c_1^2c_3^2 - 4(c_1c_2^2c_3 - c_2c_3^2 + c_2^2c_4 - c_1^2c_2c_4 + c_1^3c_5 \\ &\quad + c_3c_5 - c_1^2c_6 + c_2c_6 + c_1c_7 - c_8) \\ &\quad + 6c_4^2 - 8(c_1c_3c_4 - c_1c_2c_5), \\ c_3^{(4)} &= c_3^4 + 2c_2^2c_4^2 - \dots, \\ c_4^{(4)} &= c_4^4 + \dots, \end{aligned} \quad (7.12)$$

and so on.

$p = 5$: *Fifth-power-of-roots Algorithm*

$$\begin{aligned}
 c_1^{(5)} &= c_1^5 - 5\{c_1^3c_2 - c_1c_2^2 + c_2c_3 - c_1^2c_3 + c_1c_4 - c_5\}, \\
 c_2^{(5)} &= c_2^5 - 5\{c_1c_2^3c_3 - c_1^2c_2c_3^2 - c_2^2c_3^2 + c_1c_3^3 - c_1^2c_2^2c_4 + c_2^3c_4 \\
 &\quad + c_1^3c_3c_4 - c_1^2c_4^2 - c_2c_4^2 - c_3^2c_4 + c_1c_2c_3c_4 \\
 &\quad + c_1^3c_2c_5 - c_1^4c_6 - c_2^2c_6 + c_4c_6 \\
 &\quad + c_1^3c_1 + c_3c_7 - c_1^2c_8 + c_2c_8 + c_1c_9 - c_{10}\} \\
 &\quad + 10\{c_1c_2^2c_5 + c_1^2c_3c_5 + c_5^2 + c_1c_3c_6 + c_1c_2c_7\} \\
 &\quad - 15\{c_1^2c_2c_6 + c_2c_3c_5 - c_1c_4c_5\},
 \end{aligned} \tag{7.13}$$

and so on.

Although the method described above is straightforward and can be programmed for a computer, the following two theorems enable us to get the same results either quicker or in a different way:

Theorem 1. A homogeneous symmetric polynomial of degree m in the α 's, when expressed in terms of the c 's, is isobaric of weight m .

Theorem 2. A symmetric polynomial in $(\alpha_1, \dots, \alpha_n)$, when written in terms of the elementary functions $c_1, \dots, c_n$, will be of the same degree in the c 's as it was at first in any one of the α 's.

Simple proofs for these well-known theorems can be found in Ref. 8.

By definition (7.1), $c_k^{(p)}$ is a symmetric polynomial in $\alpha_1, \dots, \alpha_n$, of degree $m = kp$. Thus, by Theorems 1 and 2,

$$c_k^{(p)} = \sum_{(i)} \gamma_{k,i}^{(p)} c_1^{\lambda_1^{(i)}} c_2^{\lambda_2^{(i)}} \dots c_n^{\lambda_n^{(i)}}, \tag{7.14}$$

where the summation extends over all possible sets $\{\lambda_j^{(i)}\}$ of nonnegative integers $\lambda_j^{(i)}$ ($j = 1, 2, \dots, n$) such that

$$\begin{aligned}
 \lambda_1^{(i)} + 2\lambda_2^{(i)} + \dots + n\lambda_n^{(i)} &= kp \quad (\text{isobaric}) \\
 \lambda_1^{(i)} + \lambda_2^{(i)} + \dots + \lambda_n^{(i)} &\leq p \quad (\text{degree } p \text{ in } c\text{'s}),
 \end{aligned} \tag{7.15}$$

the equal sign being valid at least once. In addition, it can be shown that the γ 's are constant integers (see Ref. 21, page 149, Theorem 68) and independent of n (see Ref. 8, page 244, Theorem 2), but dependent on p and k , of course.

Example: For $c_3^{(2)}$, we have $m = kp = 6$; $p = 2$. Thus the possible sets $\{\lambda_j^{(i)}\}$ are given in the following table:

i	$\lambda_1^{(i)}$	$\lambda_2^{(i)}$	$\lambda_3^{(i)}$	$\lambda_4^{(i)}$	$\lambda_5^{(i)}$	$\lambda_6^{(i)}$	$\sum \lambda_j^{(i)}$	$\sum j \lambda_j^{(i)}$
1	0	0	2	0	0	0	2	6
2	0	1	0	1	0	0	2	6
3	1	0	0	0	1	0	2	6
4	0	0	0	0	0	1	1	6

Therefore, $c_3^{(2)}$ is the sum of four terms:

$$c_3^{(2)} \equiv \gamma_1 c_3^2 + \gamma_2 c_2 c_4 + \gamma_3 c_1 c_5 + \gamma_4 c_6. \quad (7.16)$$

The γ 's may be determined by the ordinary method of undetermined coefficients. We choose four different sets of α 's, which determine four different sets of c 's. This leads to a system of four linear equations. Since the γ 's are independent of n , we start with $n = 3$ and $\alpha_1 = \alpha_2 = \alpha_3 = 1$. Thus $c_3^{(2)} = 1$, $c_3 = 1$, $c_4 = c_5 = c_6 = 0$. Therefore, (7.16) yields $\gamma_1 = 1$.

Next, take $n = 4$ and $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 1$. Thus

$$c_3^{(2)} = 4, c_3 = \binom{4}{3} = 4, c_2 = \binom{4}{2} = 6, c_4 = 1, c_5 = c_6 = 0.$$

Therefore, from (7.16), $\gamma_2 = -2$. Similarly, we obtain the other coefficients and finally have

$$c_3^{(2)} = c_3^2 - 2c_2 c_4 + 2c_1 c_5 - 2c_6,$$

which agrees with (7.9).

Constructing a triangular matrix for the linear system will not always be as simple as in the above example. However, the aim will be to make as many c_i 's zero as possible by a judicious choice of the α 's.

We are now ready to prove the following theorem, which will be basic for the proof of the Main Theorem (see Section 8).

Theorem 3. Let

$$P_n^{(p)}(z) = a_0^{(p)} z^n + a_1^{(p)} z^{n-1} + \dots + a_n^{(p)} \quad (7.17)$$

be the polynomial whose zeros are the p th power of the zeros of

$$P_n(x) = a_0 x^n + a_1 x^{n-1} + \dots + a_n, \quad (7.18)$$

and let $a_0^{(p)} = a_0^p$. Then the coefficients $a_k^{(p)}$ are homogeneous polynomials in $(a_0, \dots, a_n)$ of degree p and weight kp with integer coefficients.

In fact, we shall prove that

$$a_k^{(p)} = \sum_{(i)} b_{k,i}^{(p)} a_0^{\lambda_{k,0}^{(i)}} a_1^{\lambda_{k,1}^{(i)}} \dots a_n^{\lambda_{k,n}^{(i)}}, \quad (7.19)$$

where the summation extends over all possible sets

$$\left\{ \lambda_{k,j}^{(i)} \right\}^{(p)} = \left\{ \lambda_{k,j}^{(i)} \ (j = 0, 1, \dots, n): \sum_{j=0}^n \lambda_{k,j}^{(i)} = p; \right. \\ \left. \sum_{j=1}^n j \lambda_{k,j}^{(i)} = kp; \lambda_{k,j}^{(i)} \geq 0, \text{ integer} \right\} \quad (7.20)$$

and the b 's are integers independent of n .

Proof. Define

$$a_k^{(p)} = (-1)^k c_k^{(p)} a_0^p \quad (k = 0, 1, 2, \dots, n). \quad (7.21)$$

By (6.3),

$$c_k = (-1)^k a_k / a_0 \quad (k = 1, 2, \dots, n). \quad (7.22)$$

Substitute (7.22) into (7.14) and remember from (7.15) that $kp = \lambda_1 + 2\lambda_2 + \dots + n\lambda_n$ to obtain

$$c_k^{(p)} = \sum_{(i)} \gamma_{k,i}^{(p)} (-1)^{kp} \frac{a_1^{\lambda_1^{(i)}} \dots a_n^{\lambda_n^{(i)}}}{a_0^{\lambda_1^{(i)} + \dots + \lambda_n^{(i)}}} \quad (k = 1, 2, \dots, n).$$

Multiply both sides by $(-1)^k a_0^p$ and define $\lambda_0^{(i)}$ by

$$p = \lambda_0^{(i)} + \lambda_1^{(i)} + \dots + \lambda_n^{(i)} \quad (7.23)$$

(where $\lambda_0 \geq 0$, integer, by (7.15)) to obtain

$$a_k^{(p)} = \sum_{(i)} (-1)^{k(p+1)} \gamma_{k,i}^{(p)} a_0^{\lambda_0^{(i)}} a_1^{\lambda_1^{(i)}} \dots a_n^{\lambda_n^{(i)}}, \quad (k = 1, 2, \dots, n). \quad (7.24)$$

Define

$$b_{k,i}^{(p)} = (-1)^{k(p+1)} \gamma_{k,i}^{(p)} \quad (7.25)$$

and introduce it in (7.24). Finally, denote $\lambda_j^{(i)}$ in (7.24) by $\lambda_{k,j}^{(i)}$ to indicate its dependence on k , to arrive at (7.19), thus concluding the proof.

Because of (7.21), we can rewrite (7.6) for $a_k^{(p)}$. From (7.6) alone, it is not obvious how $a_0^{(p)}$ should properly be defined. Thus

$$a_0^{(p)} = a_0^p,$$

$$a_k^{(p)} = a_0^p \sum_{(i)} \frac{(-1)^{\lambda_1^{(i)} + \dots + \lambda_k^{(i)}}}{\lambda_1^{(i)}! \dots \lambda_k^{(i)}!} \left(\frac{s_p}{1}\right)^{\lambda_1^{(i)}} \dots \left(\frac{s_{kp}}{k}\right)^{\lambda_k^{(i)}}, \quad (7.26)$$

where the summation extends over all $\lambda_k^{(i)}$ such that

$$\lambda_1^{(i)} + 2\lambda_2^{(i)} + \dots + k\lambda_k^{(i)} = kp.$$

The algorithms for the $a_k^{(p)}$ of (7.17) can now be obtained by (7.26), or from the $a_k^{(p)}$ by (7.21), or by the ordinary method of undetermined coefficients applying Theorem 3. The right-hand sides of the new formulas are both isobaric of weight kp and homogeneous of degree p . The formulas corresponding to equations (7.10) to (7.13) are given below, but for $p > 2$ only terms containing $a_k^{(p)}$'s and a_k 's with indices $k < p$ are printed. We have for

$p = 2$: *Root-squaring Algorithm*

$$a_0^{(2)} = a_0^2,$$

$$a_1^{(2)} = -a_1^2 + 2a_0a_2, \quad (7.27)$$

$$a_2^{(2)} = a_2^2 - 2a_1a_3 + 2a_0a_4,$$

... ..

$p = 3$: *Root-cubing Algorithm*

$$a_0^{(3)} = a_0^3,$$

$$a_1^{(3)} = a_1^3 - 3a_0a_1a_2 + \dots, \quad (7.28)$$

$$a_2^{(3)} = a_2^3 - \dots,$$

... ..

$p = 4$: Root-bisquaring Algorithm

$$a_0^{(4)} = a_0^4,$$

$$a_1^{(4)} = -a_1^4 - 2a_0^2a_2^2 + 4(a_0a_1^2a_2 - a_0^2a_1a_3) + \dots,$$

$$a_2^{(4)} = a_2^4 + 2a_1^2a_3^2 + 4(a_0a_2^2a_3 - a_1a_2^2a_3) + \dots, \quad (7.29)$$

$$a_3^{(4)} = -a_3^4 + \dots,$$

... ..

$p = 5$: Fifth-power-of-roots Algorithm

$$a_0^{(5)} = a_0^5;$$

$$a_1^{(5)} = a_1^5 - 5\{a_0a_1^3a_2 - a_0^2a_1a_2^2 + a_0^3a_2a_3 - a_0^2a_1^2a_3 + a_0^3a_1a_4\} + \dots,$$

$$a_2^{(5)} = a_2^5 + 5\{a_0a_2^2a_3^2 - a_0a_1a_3^3 + a_1^2a_2a_3^2 - a_1a_2^3a_3 + a_0a_1^2a_4^2 - a_0a_2^3a_4 \\ + a_0^2a_2a_4^2 - a_1^3a_3a_4 + a_1^2a_2^2a_4 - a_0a_1a_2a_3a_4 + a_0^2a_3^2a_4\}, \quad (7.30)$$

$$a_3^{(5)} = a_3^5 - 5\{a_0a_3^3a_4 - a_1a_3^2a_4^2 + a_1a_2a_4^3 - a_2^2a_3a_4^2 + a_2a_3^3a_4\} + \dots,$$

$$a_4^{(5)} = a_4^5 + \dots,$$

... ..

Theorem 4. Let $P_n(x)$ and $P_n^{(p)}(z)$ be given as in Theorem 3, and, in (7.19),

let the integers $b_{k,i}^{(p)}$ be the coefficient of $a_0^{\lambda_{k,0}^{(i)}} \dots a_n^{\lambda_{k,n}^{(i)}}$.

Then $b_{n-k,i}^{(p)} = (-1)^{n(p+1)} b_{k,i}^{(p)}$, where $b_{n-k,i}^{(p)}$ is the coefficient

$$a_0^{\lambda_{k,n}^{(i)}} a_1^{\lambda_{k,n-1}^{(i)}} \dots a_n^{\lambda_{k,0}^{(i)}}.$$

Proof. The proof is divided into two parts. First we show that $b_{n-k,i}^{(p)}$ exists, then that $b_{n-k,i}^{(p)} = (-1)^{n(p+1)} b_{k,i}^{(p)}$.

The set

$$\left\{ \lambda_{k,j}^{(i)} \right\}^{(p)},$$

associated with $b_{k,i}^{(p)}$, satisfies

$$\lambda_{k,0}^{(i)} + \lambda_{k,1}^{(i)} + \dots + \lambda_{k,n}^{(i)} = p \quad (\text{homogeneous of degree } p). \quad (7.31)$$

$$\lambda_{k,1}^{(i)} + 2\lambda_{k,2}^{(i)} + \dots + n\lambda_{k,n}^{(i)} = kp \quad (\text{isobaric of weight } kp). \quad (7.32)$$

The set

$$\left\{ \lambda_{n-k,j}^{(i)} \right\}^{(p)},$$

associated with $b_{n-k,i}^{(p)}$, contains by hypothesis the same elements as

$$\left\{ \lambda_{k,j}^{(i)} \right\}^{(p)}$$

but in reverse order; i.e.,

$$\lambda_{n-k,j}^{(i)} = \lambda_{k,n-j}^{(i)} \quad (j = 0, 1, \dots, n). \quad (7.33)$$

Hence,

$$\lambda_{k,n}^{(i)} + \lambda_{k,n-1}^{(i)} + \dots + \lambda_{k,0}^{(i)} = p \quad (\text{homogeneous of degree } p) \quad (7.34)$$

is satisfied. Multiply this equation by n and subtract (7.32) to obtain

$$\begin{aligned} np - kp &= n\lambda_{k,n}^{(i)} + n\lambda_{k,n-1}^{(i)} + \dots + n\lambda_{k,1}^{(i)} + n\lambda_{k,0}^{(i)} \\ &\quad - \left(n\lambda_{k,n}^{(i)} + (n-1)\lambda_{k,n-1}^{(i)} + \dots + \lambda_{k,1}^{(i)} \right) \\ &= \lambda_{k,n-1}^{(i)} + 2\lambda_{k,n-2}^{(i)} + \dots + n\lambda_{k,0}^{(i)} = (n-k)p \\ &\quad (\text{isobaric of weight } (n-k)p), \end{aligned} \quad (7.35)$$

which proves the existence of the term

$$a_0 \lambda_{n-k,0}^{(i)} \dots a_n \lambda_{n-k,n}^{(i)} = a_0 \lambda_{k,n}^{(i)} \dots a_n \lambda_{k,0}^{(i)}.$$

To prove that $b_{n-k,i}^{(p)} = (-1)^{n(p+1)} b_{k,i}^{(p)}$, consider

$$x^n P_n\left(\frac{1}{x}\right) = a_0 + a_1 x + \dots + a_n x^n. \quad (7.36)$$

The zeros of (7.35) are $\beta_i = \frac{1}{\alpha_i}$ ($i = 1, 2, \dots, n$). The corresponding polynomial

$$z^n P_n^{(p)}(z) = a_0^p + a_1^{(p)} z + \dots + a_n^{(p)} z^n \quad (7.37)$$

has the zeros $\beta_i^p = (1/\alpha_i)^p$, ($i = 1, 2, \dots, n$). The coefficient $a_n^{(p)}$ in (7.37) is easily determined from (7.21), (7.1), (6.2), and (6.3) to be

$$\begin{aligned} a_n^{(p)} &= (-1)^n c_n^{(p)} a_0^p = (-1)^n [c_n a_0]^p = (-1)^n [(-1)^n a_n]^p \\ &= (-1)^{n(p+1)} a_n^p. \end{aligned} \quad (7.38)$$

Applying (7.19) to (7.37) yields

$$a_{n-k}^{(p)} = (-1)^{n(p+1)} \sum_{(i)} b_{k,i}^{(p)} a_n^{\lambda_{k,0}^{(i)}} \dots a_0^{\lambda_{k,n}^{(i)}}, \quad (7.39)$$

where

$$\begin{aligned} \lambda_{k,0}^{(i)} + \lambda_{k,1}^{(i)} + \dots + \lambda_{k,n}^{(i)} &= p, \\ \lambda_{k,1}^{(i)} + 2\lambda_{k,2}^{(i)} + \dots + n\lambda_{k,n}^{(i)} &= kp. \end{aligned} \quad (7.40)$$

The factor in front of the summation symbol compensates for a possible change of sign in the coefficients. Since $a_{n-k}^{(p)}$ has the unique representation

$$a_{n-k}^{(p)} = \sum_{(i)} b_{n-k,i}^{(p)} a_0^{\lambda_{n-k,0}^{(i)}} \dots a_n^{\lambda_{n-k,n}^{(i)}},$$

where

$$\begin{aligned} \lambda_{n-k,0}^{(i)} + \dots + \lambda_{n-k,n}^{(i)} &= p, \\ \lambda_{n-k,1}^{(i)} + 2\lambda_{n-k,2}^{(i)} + \dots + n\lambda_{n-k,n}^{(i)} &= (n-k)p, \end{aligned} \quad (7.41)$$

it follows from the existence of (7.33) that we must have

$$b_{n-k,i}^{(p)} = (-1)^{n(p+1)} b_{k,i}^{(p)}. \quad (7.42)$$

Corollary. Assume that in (7.19) for the term $b_{k,i}^{(p)} a_0^{\lambda_{k,0}^{(i)}} \dots a_n^{\lambda_{k,n}^{(i)}}$ the exponents $\lambda_{k,j}^{(i)} = 0$ for $j > k + \ell \leq n$. Then there exists a term $b_{\ell,i}^{(p)} a_0^{\lambda_{\ell,k+\ell}^{(i)}} \dots a_{k+\ell}^{\lambda_{\ell,0}^{(i)}}$ and its coefficient is $b_{\ell,i}^{(p)} = (-1)^{(k+\ell)(p+1)} b_{k,i}^{(p)}$.

Proof. Since $b_{k,i}^{(p)}$ is independent of n , let $a_j = 0$ for $j > k + \ell$. But then we have the conditions for which Theorem 4 is valid, letting $n - k = \ell$ there. The symmetry property of (7.19) expressed by Theorem 4 will be applied in the next section.* The corollary may be used to reduce the numerical work in calculating the b 's. For example, in (7.30) we have

$$a_1^{(5)} = \dots - 5a_0^3 a_1^0 a_2^1 a_3^1 + \dots$$

Let $\ell = 2$, and obtain

$$a_2^{(5)} = \dots - 5a_0^1 a_1^1 a_2^0 a_3^3 + \dots$$

*This symmetry property is quite obvious when we replace the polynomial $P_n(x)$ in Eq. (6.1) by the binary form,

$$\begin{aligned} & a_0 x^n + a_1 x^{n-1} y + \dots + a_{n-1} x y^{n-1} + a_n y^n \\ & \equiv (xy_1 - x_1 y)(xy_2 - x_2 y) \dots (xy_n - x_n y), \end{aligned}$$

where

$$a_0 = y_1 y_2 \dots y_n,$$

$$-a_1 = \sum x_1 y_2 y_3 \dots y_n,$$

...

$$(-1)^{n-1} a_{n-1} = \sum x_1 \dots x_{n-1} y_n,$$

$$(-1)^n a_n = \sum x_1 x_2 \dots x_n.$$

Thus, $x_i/y_i = \alpha_i$ ($i = 1, 2, \dots, n$) are the roots of (6.1), and $y_i/x_i = \beta_i$ ($i = 1, 2, \dots, n$) are the roots of (7.35), assuming $x_i y_i \neq 0$.

For $l = 3$, we write

$$a_1^{(5)} = \dots - 5a_0^3 a_1^0 a_2^1 a_3^1 a_4^0 + \dots$$

to obtain

$$a_3^{(5)} = \dots - 5a_0^0 a_1^1 a_2^1 a_3^0 a_4^3 + \dots,$$

and so on.

8. The p th-power-of-roots Algorithm for Polynomials of Degree $p - 1$, and the General Root-powering Method by Partitioning of the Polynomial

If we set $a_k = 0$ for $k \geq p$ in Eqs. (7.27) and (7.28), we have for

$$p = 2:$$

$$P_1^{(2)}(z) = a_0^2 z - a_1^2,$$

$$P_1^{(2)}(1) = a_0^2 - a_1^2, \quad (8.1)$$

$$z P_1^{(2)}\left(\frac{1}{z}\right) = a_0^2 - a_1^2 z;$$

$$p = 3:$$

$$P_2^{(3)}(z) = a_0^3 z^2 + a_1^3 z + a_2^3 - 3a_0 a_1 a_2 z,$$

$$P_2^{(3)}(1) = a_0^3 + a_1^3 + a_2^3 - 3a_0 a_1 a_2, \quad (8.2)$$

$$z^2 P_2^{(3)}\left(\frac{1}{z}\right) = a_0^3 + a_1^3 z + a_2^3 z^2 - 3a_0 a_1 a_2 z.$$

Similar equations can be written for $p = 4$ and $p = 5$ from Eqs. (7.29) and (7.30), respectively. If we let $a_0 = A$ and $a_1 = B$ in the second equation of (8.1), and $a_0 = A$, $a_1 = B$, and $a_2 = C$ in the second equation of (8.2), we obtain the root-squaring transformations $A^2 - B^2$ of (4.3) and the root-cubing transformation $A^3 + B^3 + C^3 - 3ABC$ of (2.5), respectively. Similarly, we obtain from (7.29) for

$$p = 4:$$

$$\begin{aligned} P_3^{(4)}(1) &= a_0^4 - a_1^4 + a_2^4 - a_3^4 - 2(a_0^2 a_2^2 - a_1^2 a_3^2) \\ &\quad + 4(a_0 a_1^2 a_2 - a_0^2 a_1 a_3 + a_0 a_2 a_3^2 - a_1 a_2^2 a_3) \\ &\equiv (a_0^2 + a_2^2 - 2a_1 a_3)^2 - (a_1^2 + a_3^2 - 2a_0 a_2)^2, \end{aligned} \quad (8.3)$$

which corresponds to the transformation (5.4) of Part I.

These three examples illustrate the rule for obtaining the general root-power transformations by partitioned polynomials, which will be described in this section. For convenience, we return to the notation of Part I for $P_n(x)$ and $P^{(p)}_n(z)$. Because of Theorem 4, the results of the previous sections can be applied unchanged. Thus let

$$P_n(x) = \sum_{i=0}^n a_i x^i, \quad (8.4)$$

and

$$P^{(p)}_n(z) = \sum_{i=0}^n a_i^{(p)} z^i, \quad (8.5)$$

where $a_0^{(p)} = a_0^p$, and the zeros of (8.5) are the p th powers of the zeros of (8.4). If we substitute $z = x^p$ in (8.5), then $P^{(p)}_n(x^p)$ is a polynomial of degree pn in x whose np zeros are the zeros of $P_n(x)$ multiplied by the p th roots of unity.

Definition. The polynomial $P_n(x) = \sum_{i=0}^n a_i x^i$ is called p -partitioned if

$$P_n(x) = \sum_{k=0}^{p-1} A_{p,k} \quad \text{where the } p\text{-partitions are defined by}$$

$$A_{p,k} = x^k \sum_{i=0}^{\left[\frac{n-k}{p}\right]} a_{pi+k} x^{pi} \quad (k = 0, 1, 2, \dots, p-1). \quad (8.6)$$

Main Theorem. Let a given polynomial be p -partitioned. Then the polynomial whose zeros are the zeros of the given polynomial multiplied by the p th roots of unity is an algebraic homogeneous function of degree p in the p -partitions, with integer coefficients.

We reformulate this theorem in more specific language as

Theorem 5. Let

$$P_n(x) = \sum_{i=0}^n a_i x^i = \sum_{k=0}^{p-1} A_{p,k}, \quad (8.7)$$

where $A_{p,k}$ is defined by (8.6); and let

$$P^{(p)}_n(z) = \sum_{i=0}^n a_i^{(p)} z^i \quad (a_0^{(p)} = a_0^p) \quad (8.5)$$

be the polynomial whose zeros are the p th powers of the zeros of $P_n(x)$. If we set

$$z = x^p, \quad (8.8)$$

then

$$P_n^{(p)}(z) = P_n^{(p)}(x^p) = \sum_{k=0}^{p-1} \sum_{(i)} b_{k,i}^{(p)} A_{p,0}^{\lambda_{k,0}^{(i)}} A_{p,1}^{\lambda_{k,1}^{(i)}} \dots A_{p,p-1}^{\lambda_{k,p-1}^{(i)}}, \quad (8.9)$$

where the $b_{k,i}^{(p)}$ and $\lambda_{k,j}^{(i)}$ ($j = 0, 1, 2, \dots, p-1$) are the same integers that occur in (7.19) and (7.20).

Proof. Define

$$a_{p,k}(z) = \sum_{i=0}^{\left[\frac{n-k}{p}\right]} a_{pi+k} z^i \quad (k = 0, 1, 2, \dots, p-1), \quad (8.10)$$

so that

$$A_{p,k} = x^k a_{p,k}(x^p). \quad (8.11)$$

Then

$$P_n(x) = \sum_{k=0}^{p-1} x^k a_{p,k}(x^p). \quad (8.12)$$

Let α_j be any root of $P_n(x) = 0$. Then α_j^p is also a root of

$$P_{p-1}(x) = \sum_{k=0}^{p-1} x^k a_{p,k}(\alpha_j^p) = 0. \quad (8.13)$$

There may be $p-2$ additional roots $\alpha_{j,k}$ ($k = 1, 2, \dots, p-2$). Let $P_{p-1}^{(p)}(z)$ be the polynomial whose zeros are α_j^p and $\alpha_{j,k}^p$ ($k = 1, \dots, p-2$). Then

$$P_{p-1}^{(p)}(z) = \sum_{k=0}^{p-1} z^k a_{p,k}^{(p)}(\alpha_j^p), \quad (8.14)$$

where by (7.19) and (7.20) (because of the symmetry property of the coefficients $b_{k,i}^{(p)}$ (Theorem 4) and the normalization $a_{p,0}^{(p)}(\alpha_j^p) = [a_{p,0}(\alpha_j^p)]^p$),

$$a_{p,k}^{(p)}(\alpha_j^p) = \sum_{(i)} b_{k,i}^{(p)} [a_{p,0}(\alpha_j^p)]^{\lambda_{k,0}^{(i)}} \dots [a_{p,p-1}(\alpha_j^p)]^{\lambda_{k,p-1}^{(i)}} \quad (k = 0, 1, \dots, p). \quad (8.15)$$

We show that $a_{p,k}^{(p)}(\alpha_j^p)$ is at most of degree $n - k$ in α_j^p . The degree of $a_{p,k}^{(p)}(\alpha_j^p)$ is at most $[(n - k)/p]$ in α_j^p by (8.10). Each individual term on the right side of (8.15) is therefore at most of degree

$$\sum_{\ell=0}^{p-1} \left[\frac{n - \ell}{p} \right] \lambda_{\ell}^{(i)} \leq \sum_{\ell=0}^{p-1} \frac{n - \ell}{p} \lambda_{\ell}^{(i)} = \frac{n}{p} \sum \lambda_{\ell}^{(i)} - \frac{1}{p} \sum \ell \lambda_{\ell}^{(i)} = n - k$$

in α_j^p .

Thus, if we substitute z for α_j^p in (8.15), $a_{p,k}^{(p)}$ is at most of degree $n - k$ in z . Consequently, by (8.14), $P_n^{(p)}(z)$ becomes a polynomial $P'_n(z)$ of degree not higher than n in z ,

$$P'_n(z) = \sum_{k=0}^{p-1} z^k a_{p,k}^{(p)}(z). \quad (8.16)$$

Clearly, α_j^p is a zero of this polynomial, $P'_n(z)$. Since α_j^p is an arbitrary zero of $P_n(x)$, the same reasoning applies to all zeros α_j^p ($j = 1, 2, \dots, n$) of $P_n(x)$. The coefficients $b_{k,i}^{(p)}$ in (8.15) do not depend on the α_j^p 's. Thus $P'_n(z)$ vanishes for $z = \alpha_j^p$ ($j = 1, 2, \dots, n$). Therefore it is the unique polynomial (8.5) of degree n :

$$P'_n(z) \equiv P_n^{(p)}(z). \quad (8.17)$$

By using (8.10), we can verify that $\alpha_0^{(p)} = \alpha_0^p$ in $P'_n(z)$.

To bring (8.16) into final form, we combine (8.17), (8.16), and (8.15) to obtain

$$P_n^{(p)}(z) = \sum_{k=0}^{p-1} z^k \sum_{(i)} b_{k,i}^{(p)} [a_{p,0}(z)]^{\lambda_{k,0}^{(i)}} \dots [a_{p,p-1}(z)]^{\lambda_{k,p-1}^{(i)}}. \quad (8.18)$$

By (8.8) we have $z = x^p$, and from (7.20) we obtain

$$z^k = x^{kp} = [x^1]^{\lambda_{k,1}^{(i)}} [x^2]^{\lambda_{k,2}^{(i)}} \dots [x^{p-1}]^{\lambda_{k,p-1}^{(i)}}.$$

Thus, (8.18) becomes

$$P_n^{(p)}(x^p) = \sum_{k=0}^{p-1} \sum_{(i)} b_{k,i}^p [a_{p,0}(x^p)]^{\lambda_{k,0}^{(i)}} [x^1 a_{p-1}(x^p)]^{\lambda_{k,1}^{(i)}} \dots [x^{p-1} a_{p,p-1}(x^p)]^{\lambda_{k,p-1}^{(i)}}. \quad (8.19)$$

Making the substitution (8.11) in this equation leads to the final form, (8.9). Should some $A_{p,j} = 0$, (8.9) will still be valid if we define (as usual)

$$A_{p,j}^{\lambda_{k,j}^{(i)}} = \begin{cases} 0 & \text{if } \lambda_{k,j}^{(i)} > 0 \\ 1 & \text{if } \lambda_{k,j}^{(i)} = 0 \end{cases}.$$

We have therefore proved that it is always possible to generate a "pth-power-of-the-roots algorithm," using only multiplication and addition of the p-partitions of the given polynomial. Since our proof is constructive, a method to devise such an algorithm is simultaneously supplied.

Rule. To obtain the pth-power-of-the-roots algorithm by p-partitions,

- a) Compute the $b_{k,i}^{(p)}$ for $P_{p-1}^{(p)}(1) = \sum_{i=0}^{p-1} a_i^{(p)}$,
- b) Replace a_i by $A_{p,i}$ ($i = 0, 1, \dots, p-1$) in $P_{p-1}^{(p)}(1)$.

As we see from the example (8.3) for $p = 4$, these algorithms are not necessarily in their most efficient form. To give yet another example, for $p = 5$, we can also write, based on (7.30),

$$P_4^{(5)}(1) = \sum_{i=0}^4 a_i \{ [a_i^2]^2 - 5[a_{i-1}a_{i+1}(a_i - a_{i-1}a_{i+1}) + a_{i+2}a_{i-2}(a_i^2 - a_{i+2}a_{i-2})] \} - 5a_0a_1a_2a_3a_4, \quad (8.20)$$

where $i \pm j$ is to be taken modulo 5; e.g., a_0 appears as a_0 , a_{-2} appears as a_3 , etc.

We can always find algorithms that raise the power of the roots so that they differ by only one. For example, we can apply root squaring three times, then use root cubing twice. Assuming the roots are separated enough, we can calculate

$$\alpha_1 \approx \frac{c_1^{(9)}}{c_1^{(8)}}, \quad \alpha_1 \alpha_2 \approx \frac{c_2^{(9)}}{c_2^{(8)}}, \quad \dots \quad (8.21)$$

The above fact is based on finding the decomposition of two consecutive numbers in primes. Thus we have

$$\left\{ \begin{array}{l} 9 = 3^2 \\ 8 = 2^3 \end{array} \right\} \quad \left\{ \begin{array}{l} 625 = 5^4 \\ 624 = 13 \times 3 \times 2^4 \end{array} \right\} \quad \left\{ \begin{array}{l} 1225 = 7^2 \times 5^2 \\ 1224 = 17 \times 3^2 \times 2^3 \end{array} \right\}.$$

These few examples will suffice for most practical cases. However, more research may very likely lead to better decompositions. In the root-squaring method, a power of more than 1024 was rarely needed. Because of rounding errors, we cannot expect a high accuracy for the α 's by the method exemplified in (8.21). However, by comparing, for example,

$$1225 \sqrt[1225]{c_1^{(1225)}} \quad \text{and} \quad 1224 \sqrt[1224]{c_1^{(1224)}} \quad \text{etc.,}$$

we can obtain a good estimate for the magnitude of the rounding errors in the roots.

Even when the algorithms lead to powers of the roots that differ by a number other than one, we can construct codes for the direct computation of the moduli and arguments of the complex roots by combining two or more algorithms. Preliminary investigations indicate, however, that the resultant procedures are more advantageous than such general codes. For real polynomials and only one pair of complex roots, the situation is quite simple. The special case for root squaring and root cubing is worked out in Ref. 10:

Suppose that the pivotal stage of the coefficients $a_{n-s+1}^{(k)}$ and $a_{n-s}^{(k)}$ has been reached for $m = m_1$ or m_2 in the root-squaring and root-cubing methods, respectively. Then,

$$\cos p_1 \phi \approx \frac{1}{2} \rho^{-p_1} \frac{a_{n-s}^{(m_1)}}{a_{n-s+1}^{(m_1)}} = b_1 \quad (\text{root squaring, } p_1 = 2^{m_1}), \quad (8.22)$$

$$\cos p_2 \phi \approx \frac{1}{2} \rho^{-p_2} \frac{a_{n-s}^{(m_2)}}{a_{n-s+1}^{(m_2)}} = b_2 \quad (\text{root cubing, } p_2 = 3^{m_2}),$$

and if x and y are integral solutions of the diophantine equation,

$$p_1 x + p_2 y = 1, \quad (8.23)$$

it follows that

$$\phi = x \cos^{-1} b_1 + y \cos^{-1} b_2. \quad (8.24)$$

For example, $m_1 = 8$ root squarings and $m_2 = 5$ root cubings yield $p_1 = 256$, $p_2 = 243$, and $x = -56$, $y = 59$. If $m_1 = 11$ and $m_2 = 7$, then $x = 236$ and $y = -221$.

If (8.22) is extended to more than two equations, then (8.23) becomes

$$p_1 x_1 + p_2 x_2 + \dots + p_k x_k = 1, \quad (8.25)$$

and

$$\phi = x_1 \cos^{-1} b_1 + x_2 \cos^{-1} b_2 + \dots + x_k \cos^{-1} b_k. \quad (8.26)$$

As a simple example, take $k = 3$, and

$$p_1 = 2^3 \cdot 3^5 = 1944, \quad p_2 = 2^{11} = 2028, \quad p_3 = 3^7 = 2187.$$

It follows that (8.25) is satisfied for

$$x_1 = 4, \quad x_2 = -7, \quad x_3 = 3.$$

9. Circulant Representation of the Root-powering Algorithms

In this section we shall summarize our results by deriving them in a new way, leading to the circulant representation of the root-powering algorithms.

A circulant matrix of order p has the same p elements a_i arranged in cyclic order in each row and column in the following way:

$$(c_{ij}) = \begin{pmatrix} a_0 & a_1 & \dots & a_{p-2} & a_{p-1} \\ a_{p-1} & a_0 & \dots & a_{p-3} & a_{p-2} \\ \dots & \dots & \dots & \dots & \dots \\ a_1 & a_2 & \dots & a_{p-1} & a_0 \end{pmatrix}. \quad (9.1)$$

The determinant of (c_{ij}) is called circulant and will be denoted by C . Let

$$\omega_p^k = e^{\frac{2\pi i k}{p}} \quad (k = 0, 1, \dots, p-1)$$

be the p roots of unity satisfying

$$x^p - 1 = 0, \quad (9.2)$$

and define

$$f(x) = a_0 + a_1 x + \dots + a_{p-1} x^{p-1}. \quad (9.3)$$

Then it is known¹¹ that the determinant C is the product*

$$C = \prod_{k=0}^{p-1} f(\omega_p^k). \quad (9.4)$$

This representation of C leads us to reformulate our original problem. Let $P_n = P_n(x)$ be defined by (2.1), its zeros being α_j ($j = 0, 1, \dots, n-1$). Instead of asking for the polynomial $P_n^{(p)}(z)$ of order n whose zeros are α_j^p ($k = 0, 1, \dots, p-1$), as defined by (7.17) or (8.5); we ask for the polynomial $P_{pn}(x)$ of order pn whose zeros are $\alpha_j \omega_p^k$ ($j = 0, 1, \dots, n-1$; $k = 0, 1, \dots, p-1$). We have used this reformulation of the problem before in expressing our Main Theorem. It is easy to show that

$$P_{pn}(x) \equiv P_n^{(p)}(x^p). \quad (9.5)$$

Since, by (9.2),

$$\prod_{k=0}^{p-1} (x - \alpha_j \omega_p^k) = x^p - \alpha_j^p.$$

we have

$$P_{pn}(x) = \prod_{j=0}^{n-1} \prod_{k=0}^{p-1} (x - \alpha_j \omega_p^k) = \prod_{j=0}^{n-1} (x^p - \alpha_j^p) = P_n^{(p)}(x^p).$$

Define now the p -partitions of $P_n(x)$ by

$$A_k = x^k \sum_{i=0}^{\left[\frac{n-k}{p} \right]} a_{pi+k} x^{pi}, \quad (9.6)$$

so that

$$P_n = A_0 + A_1 + \dots + A_{p-1} = \sum_{k=0}^{p-1} A_k. \quad (9.7)$$

(Here, $A_k \equiv A_{p,k}$ of definition (8.6). The subscript p can be omitted because it is implied in the circulant representation (9.10) given below.) Define also, corresponding to (9.3),

$$F(\omega) = A_0 + A_1 \omega + A_2 \omega^2 + \dots + A_{p-1} \omega^{p-1}. \quad (9.8)$$

*For a connection with resultants, see Appendix A.

Then the zeros of $F(\omega_p^k)$ as polynomials in x are $\alpha_j \omega_p^{p-k}$ ($j = 0, 1, \dots, n-1$). This statement is easily proved by letting $x = \alpha_j \omega_p^{p-k}$ in (9.6) and observing that $\omega_p^{kp} = 1$. Then, for $A_\ell = A_\ell(x)$, being a function of x , we have

$$A_\ell(\alpha_j \omega_p^{p-k}) = \omega_p^{\ell(p-k)} A_\ell(\alpha_j) \quad (\ell = 0, 1, \dots, p-1).$$

Now, let $x = \alpha_j \omega_p^{p-k}$ in $F(\omega_p^k)$, and obtain

$$\begin{aligned} F(\omega_p^k) &= A_0 + \omega_p^k \omega_p^{p-k} A_1 + \omega_p^{2k} \omega_p^{2(p-k)} + \dots + \omega_p^{(p-1)k} \omega_p^{(p-1)(p-k)} \\ &= A_0 + A_1 + \dots + A_{p-1} = P_n(\alpha_j) = 0. \end{aligned}$$

Thus, if we let $\omega = \omega_p^k$ ($k = 0, 1, 2, \dots, p-1$) in (9.8), the polynomial $P_{pn}(x)$ whose zeros are $\alpha_j \omega_p^k$ ($j = 0, 1, \dots, n-1; k = 0, 1, \dots, p-1$) is the product

$$\begin{aligned} P_{pn}(x) &= \prod_{k=0}^{p-1} F(\omega_p^k) = (A_0 + A_1 + \dots + A_{p-1}) \dots \\ &\quad (A_0 + \omega_p^{p-1} A_1 + \dots + \omega_p^{(p-1)^2} A_{p-1}). \end{aligned} \quad (9.9)$$

But by (9.4) and (9.1), this is the circulant of $A_0, A_1, \dots, A_{p-1}$. Therefore we have

Theorem 6. Let P_n be a polynomial of degree n , and let A_k ($k = 0, 1, \dots, p-1$) be its p -partitions defined by (9.6) so that

$$P_n = A_0 + A_1 + \dots + A_{p-1}.$$

Then the polynomial P_{pn} of order pn whose (pn) zeros are the n zeros of P_n multiplied by the p p th-roots of unity is given by the circulant of the p -partitions, A_k . As a function of the independent variable of P_n , P_{pn} is also a polynomial $P_n^{(p)}$ of degree n of the p th power of this independent variable; i.e.,

$$P_{pn} = P_n^{(p)} = \begin{vmatrix} A_0 & A_1 & \dots & A_{p-1} \\ A_{p-1} & A_0 & \dots & A_{p-2} \\ \dots & \dots & \dots & \dots \\ A_1 & A_2 & \dots & A_0 \end{vmatrix}. \quad (9.10)^*$$

From the circulant representation (9.10), we have immediately the following corollaries:

*For another derivation of (9.10), see Appendix B.

Corollary 1. $P_n^{(p)}(x^p)$ is a homogeneous algebraic function of degree p in $A_0, \dots, A_{p-1}$.

$$P_n^{(p)} = \sum_{k=0}^{p-1} \sum_{(i)} b_{k,i} A_0^{\lambda_{k,0}^{(i)}} \dots A_{p-1}^{\lambda_{k,p-1}^{(i)}}, \quad (9.11)$$

where

$$\lambda_{k,0}^{(i)} + \lambda_{k,1}^{(i)} + \dots + \lambda_{k,p-1}^{(i)} = p. \quad (9.12)$$

Corollary 2. There are at most $p!$ terms in (9.11).

Corollary 3. The weight of each term $b_{k,i} A_0^{\lambda_{k,0}^{(i)}} \dots A_{p-1}^{\lambda_{k,p-1}^{(i)}}$ of (9.11) is a multiple of p only; i.e.,

$$\lambda_{k,1}^{(i)} + 2\lambda_{k,2}^{(i)} + \dots + (p-1)\lambda_{k,p-1}^{(i)} = kp \quad (k = 0, 1, \dots, p-1). \quad (9.13)$$

Corollary 1 is self-evident. To prove Corollary 2, note that the determinant of the matrix (c_{ij}) ($i, j = 1, 2, \dots, p$) is the sum of the product of all signed permutations,

$$\det (c_{ij}) = \prod \pm c_{1,j_1} c_{2,j_2} \dots c_{p,j_p}. \quad (9.14)$$

Since in a circulant several terms may be equal, or equal but of opposite sign, there are at most $p!$ different terms.

In the circulant of (9.10), the elements of the determinant are

$$c_{ij} = A_{j-i+\epsilon p} \quad (i, j = 1, 2, \dots, p) \quad (9.15)$$

when we define

$$\epsilon = \begin{cases} 0 & \text{for } j \geq i \\ 1 & \text{for } j < i \end{cases} \quad (9.16)$$

But then the terms in (9.14) become

$$\pm A_{j_1-1+\epsilon_1 p} A_{j_2-2+\epsilon_2 p} \dots A_{j_p-p+\epsilon_p p}, \quad (9.17)$$

where the $j_1, \dots, j_p$ are permutations of $1, 2, \dots, p$. The weight of (9.17) is, by definition,

$$\sum_{r=1}^p (j_r - r + \epsilon_r p) = \sum_{r=1}^p j_r - \sum_{r=1}^p r + \sum_{r=0}^p \epsilon_r p = p \sum_{r=1}^p \epsilon_r = kp,$$

where $0 \leq k < p - 1$. This proves Corollary 3.

Corollary 4.

- a) The coefficient of A_k^p in (9.11) is $(-1)^{k(p-1)}$.
- b) There exist no terms $A_k^{p-1} A_\ell$ ($\ell \neq k$).
- c) There exist no terms $A_k^2 A_{k+1} \dots A_{k+p-2}$.

The proof is obvious either from the definition of the determinants or from (9.9).

Corollary 5. If $P_n = a_0 + a_1 x + \dots + a_n x^n$, then for $p > n$, $P_n^p(x^p) = a_0^{(p)} + a_1^{(p)} x^p + \dots + a_n^{(p)} x^{pn}$ is the circulant of p elements

$$a_0, a_1 x, a_2 x^2, \dots, a_n x^n, 0, 0, \dots, 0.$$

The terms of $a_k^{(p)}$ are homogeneous of degree p and of weight kp in $a_0, a_1, \dots, a_n$.

Proof. If $n = p - 1$, then $A_k = a_k x^k$, and the terms $A_0^{\lambda_0} \dots A_{p-1}^{\lambda_{p-1}}$ of weight kp are

$$\begin{aligned} & a_0^{\lambda_0} (a_1 x^1)^{\lambda_1} (a_2 x^2)^{\lambda_2} \dots (a_{p-1} x^{p-1})^{\lambda_{p-1}} \\ &= x^{\lambda_1 + 2\lambda_2 + \dots + (p-1)\lambda_{p-1}} a_0^{\lambda_0} a_1^{\lambda_1} a_2^{\lambda_2} \dots a_{p-1}^{\lambda_{p-1}} \\ &= x^{kp} a_0^{\lambda_0} a_1^{\lambda_1} \dots a_{p-1}^{\lambda_{p-1}}. \end{aligned}$$

If $n < p - 1$, let $a_{n+1} = \dots = a_{p-1} = 0$, and the desired result is obtained. Note that Corollary 5 is also a direct consequence of (9.5).

For application of the circulant method, it should be observed:

Corollary 6. With each term there are associated at most $p - 1$ other terms which follow by cyclic permutation in $A_0, \dots, A_{p-1}$ (note change of sign for p even).

By reflecting the elements on the main diagonal of the circulant of $A_0, \dots, A_{p-1}$ and then making one cyclic permutation, we obtain the circulant for the elements $A_{p-1}, \dots, A_0$. Thus

Corollary 7. The coefficients of $A_0^{\lambda_0} \dots A_{p-1}^{\lambda_{p-1}}$ and of $A_{p-1}^{\lambda_0} \dots A_0^{\lambda_{p-1}} = A_0^{\lambda_{p-1}} \dots A_{p-1}^{\lambda_0}$ are the same for odd p , and differ only in sign for even p .

For a more general result, see Appendix C.

Examples. Let $p = 2$ and $P_n(x) = A + B$; then

$$P_n^{(2)} = \begin{vmatrix} A & B \\ B & A \end{vmatrix} = A^2 - B^2$$

is the root-squaring algorithm by partitioned polynomials (4.3).

Let $p = 3$ and $P_n(x) = A + B + C$; then

$$P_n^{(3)} = \begin{vmatrix} A & B & C \\ C & A & B \\ B & C & A \end{vmatrix} = A^3 + B^3 + C^3 - 3ABC$$

is the root-cubing algorithm by partitioned polynomials (2.5). According to Corollary 2, there are at most $3!$ terms. The absolute sum of the coefficients is $1 + 1 + 1 + 3 = 6$. In accordance with Corollary 1, the terms are of degree 3; and in accordance with Corollary 3, their weights are 0, 3, 6, and 3, respectively. The formula is cyclic in A, B , and C (Corollary 6) and symmetric (Corollary 7).

For $p = 5$, we obtain (7.30) by letting $a_k = A_k$ ($k = 0, 1, \dots, 4$), $a_k = 0$ for $k > 4$. There are $1 + 6 + 12 + 6 + 1 = 26$ different terms. The absolute sum of the coefficients is 110. By Corollary 2, this sum is at most $5! = 120$. The difference of 10 is due to the terms $A_0 A_1 A_2 A_3 A_4$. There are five even permutations and 10 odd permutations, making a total of 15 permutations that yield $A_0 A_1 A_2 A_3 A_4$. Thus their coefficient is $5 - 10 = -5$.

Table V lists the expansion coefficients for the circulants

$$C(A_0, A_1, \dots, A_{p-1}) = \begin{vmatrix} A_0 & A_1 & \dots & A_{p-1} \\ A_{p-1} & A_0 & \dots & A_{p-1} \\ \dots & \dots & \dots & \dots \\ A_1 & A_2 & \dots & A_0 \end{vmatrix}$$

$$= \sum_{(\lambda)} c_{\lambda_0 \lambda_1 \dots \lambda_{p-1}} A_0^{\lambda_0} A_1^{\lambda_1} \dots A_{p-1}^{\lambda_{p-1}}$$

and the corresponding exponents λ_k of A_k for $p = 1$ to 8. All terms

$c_{\lambda} A_0^{\lambda_0} \dots A_{p-1}^{\lambda_{p-1}}$ are obtained by cyclic permutations of the exponents given in the table. As an example, take $p = 4$:

$$\begin{aligned} C(A_0, A_1, A_2, A_3) &= A_0^4 - A_1^4 + A_2^4 - A_3^4 \\ &\quad - 4[A_0^2 A_1 A_3 - A_0 A_1^2 A_2 + A_1 A_2^2 A_3 - A_0 A_2 A_3^2] \\ &\quad - 2[A_0^2 A_2^2 - A_1^2 A_3^2]. \end{aligned}$$

The method of calculating the entries of Table V is discussed in Appendix D. An extension of the circulant method for root finding of polynomials is given in Appendix E, and an extension to analytic functions in Appendix F.

TABLE V. Expansion of Circulants of Order p ($2 \leq p \leq 8$)

$$C(A_0, A_1, \dots, A_{p-1}) = \sum_{(\lambda)} c_{\lambda_0 \lambda_1 \dots \lambda_{p-1}} A_0^{\lambda_0} A_1^{\lambda_1} \dots A_{p-1}^{\lambda_{p-1}}$$

Terms obtained by cyclic permutation have the same coefficients, but alternating sign when p is even:

$$c_{\lambda_0 \lambda_1 \dots \lambda_{p-1}} = (-1)^{p-1} c_{\lambda_1 \lambda_2 \dots \lambda_0} = c_{\lambda_2 \lambda_3 \dots \lambda_1} = \dots$$

Order	c_{λ}	λ_0 λ_1 λ_2 λ_3 λ_4 λ_5	Order	c_{λ}	λ_0 λ_1 λ_2 λ_3 λ_4 λ_5 λ_6
$p = 2$	± 1	2 0	$p = 6$	± 2	3 0 3 0 0 0
$p = 3$	1	3 0 0	Contd.	± 6	3 0 1 0 0 2
	-3	1 1 1 1		± 12	3 0 0 1 1 1
$p = 4$	± 1	4 0 0 0		∓ 9	2 2 2 0 0 0
	∓ 4	2 1 0 1		∓ 18	2 1 0 1 2 0
	∓ 2	2 0 2 0 2		± 9	2 0 2 0 2 0 2
$p = 5$	1	5 0 0 0 0	$p = 7$	1	7 0 0 0 0 0 0
	-5	3 1 0 0 1		-7	5 1 0 0 0 0 1
	-5	3 0 1 1 0		-7	5 0 1 0 0 1 0
	5	2 2 0 1 0		-7	5 0 0 1 1 0 0
	5	2 1 2 0 0		7	4 2 0 0 0 1 0
	-5	1 1 1 1 1 1		14	4 1 1 0 1 0 0
$p = 6$	± 1	6 0 0 0 0 0		7	4 1 0 2 0 0 0
	∓ 6	4 1 0 0 0 1		7	4 0 2 1 0 0 0
	∓ 6	4 0 1 0 1 0		7	4 0 1 0 0 0 2
	∓ 3	4 0 0 2 0 0		14	4 0 0 1 0 1 1
	± 6	3 2 0 0 1 0		7	4 0 0 0 2 0 1
	± 12	3 1 1 1 0 0		7	4 0 0 0 1 2 0
				-7	3 3 0 0 1 0 0
				-21	3 2 1 1 0 0 0

TABLE V (Contd.)

Order	c_λ	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	Order	c_λ	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7
p = 7	.14	3	2	0	0	0	0	2		p = 8	∓ 2	4	0	4	0	0	0	0	0
Contd.	-7	3	1	3	0	0	0	0		Contd.	∓ 24	4	0	2	0	0	1	0	1
	7	3	1	1	0	0	1	1			± 20	4	0	2	0	0	0	2	0
	7	3	1	0	1	1	0	1			∓ 48	4	0	1	1	1	0	0	1
	-21	3	1	0	1	0	2	0			± 16	4	0	1	1	0	1	1	0
	-21	3	1	0	0	2	1	0			± 8	4	0	1	0	2	0	1	0
	-21	3	0	2	0	1	0	1			∓ 24	4	0	1	0	1	2	0	0
	14	3	0	2	0	0	2	0			∓ 8	4	0	0	3	0	0	0	1
	-21	3	0	1	2	0	0	1			∓ 24	4	0	0	2	1	0	1	0
	7	3	0	1	1	1	1	0			± 20	4	0	0	2	0	2	0	0
	-7	3	0	1	0	3	0	0			± 8	4	0	0	1	2	1	0	0
	14	3	0	0	2	2	0	0			∓ 8	4	0	0	1	0	0	0	3
	-21	3	0	0	0	1	1	2			± 6	4	0	0	0	4	0	0	0
	-14	2	2	1	0	1	0	1			∓ 24	4	0	0	0	1	0	1	2
	-7	2	2	1	0	0	2	0			∓ 12	4	0	0	0	0	2	0	2
	-7	2	2	0	2	0	0	1			∓ 24	4	0	0	0	0	1	2	1
	35	2	2	0	1	1	1	0			± 32	3	3	1	1	0	0	0	0
	35	2	1	2	1	0	0	1			∓ 32	3	3	0	0	0	0	1	1
	35	2	1	1	2	0	1	0			± 16	3	2	3	0	0	0	0	0
	-14	2	1	2	0	1	1	0			∓ 32	3	2	1	0	0	1	0	1
	-14	2	1	1	1	2	0	0			∓ 16	3	2	1	0	0	0	2	0
	-105	1	1	1	1	1	1	1	1		∓ 32	3	2	0	1	1	0	0	1
p = 8	± 1	8	0	0	0	0	0	0	0		± 32	3	2	0	1	0	1	1	0
	∓ 8	6	1	0	0	0	0	0	1		± 16	3	2	0	0	2	0	1	0
	∓ 8	6	0	1	0	0	0	1	0		± 48	3	2	0	0	1	2	0	0
	∓ 8	6	0	0	1	0	1	0	0		± 32	3	1	2	0	1	0	0	1
	∓ 4	6	0	0	0	2	0	0	0		∓ 32	3	1	2	0	0	1	1	0
	± 8	5	2	0	0	0	0	1	0		± 32	3	1	1	2	0	0	0	1
	± 16	5	1	1	0	0	1	0	0		± 64	3	1	1	1	1	0	1	0
	± 16	5	1	0	1	1	0	0	0		∓ 32	3	1	1	1	0	2	0	0
	± 8	5	0	2	0	1	0	0	0		± 32	3	1	1	0	2	1	0	0
	± 8	5	0	1	2	0	0	0	0		± 32	3	1	0	3	0	0	1	0
	± 8	5	0	1	0	0	0	0	2		∓ 32	3	1	0	2	1	1	0	0
	± 16	5	0	0	1	0	0	1	1		∓ 32	3	1	0	1	3	0	0	0
	± 16	5	0	0	0	1	1	0	1		∓ 32	3	1	0	1	0	0	1	2
	± 8	5	0	0	0	1	0	2	0		∓ 32	3	1	0	0	1	1	0	2
	± 8	5	0	0	0	0	2	1	0		± 32	3	1	0	0	1	0	2	1
	∓ 8	4	3	0	0	0	1	0	0		± 32	3	1	0	0	0	2	1	1
	∓ 24	4	2	1	0	1	0	0	0		± 32	3	0	3	1	0	0	0	1
	∓ 12	4	2	0	2	0	0	0	0		∓ 32	3	0	3	0	1	0	1	0
	± 20	4	2	0	0	0	0	0	2		± 16	3	0	3	0	0	2	0	0
	∓ 24	4	1	2	1	0	0	0	0		∓ 16	3	0	2	2	0	0	1	0
	± 16	4	1	1	0	0	0	1	1		± 32	3	0	2	1	1	1	0	0
	± 16	4	1	0	1	0	1	0	1		∓ 16	3	0	2	0	3	0	0	0
	∓ 24	4	1	0	1	0	0	2	0		∓ 16	3	0	2	0	0	0	1	2
	± 8	4	1	0	0	2	0	0	1		∓ 32	3	0	1	3	0	1	0	0
	∓ 48	4	1	0	0	1	1	1	0		± 16	3	0	1	2	2	0	0	0
	∓ 8	4	1	0	0	0	3	0	0		± 32	3	0	1	1	0	1	0	2
											∓ 32	3	0	1	1	0	0	2	1

TABLE V (Contd.)

Order	c_λ	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	Order	c_λ	λ_0	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7
p = 8	± 16	3	0	1	0	2	0	0	2	p = 8	± 56	2	2	0	2	2	0	0	0
Contd.	± 64	3	0	1	0	1	1	1	1	Contd.	± 80	2	2	0	1	0	0	2	1
	∓ 16	3	0	1	0	0	2	2	0		± 32	2	2	0	0	1	1	11	1
	± 48	3	0	0	2	1	0	0	2		∓ 16	2	1	2	1	2	0	0	0
	∓ 32	3	0	0	2	0	1	1	1		∓ 64	2	1	1	1	0	1	1	1
	± 32	3	0	0	1	2	0	1	1		∓ 160	2	1	1	0	2	0	1	1
	∓ 32	3	0	0	1	1	2	0	1		∓ 32	2	1	1	0	1	2	0	1
	± 32	3	0	0	1	1	1	2	0		± 32	2	1	1	0	1	1	2	0
	± 16	3	0	0	0	2	2	1	0		∓ 32	2	1	0	2	1	0	1	1
	∓ 48	2	2	2	1	0	0	0	1		∓ 16	2	1	0	2	0	2	0	1
	± 16	2	2	2	0	1	0	1	0		∓ 48	2	1	0	2	0	1	2	0
	± 8	2	2	2	0	0	2	0	0		± 96	2	1	0	1	2	1	0	1
	∓ 80	2	2	1	2	0	0	1	0		∓ 16	2	1	0	1	2	0	2	0
	∓ 32	2	2	1	1	1	1	0	0		± 56	2	0	2	0	2	0	2	0

^{1,2} The order of each cyclic permutation is p, except for the last coefficient ^{4,2}with cycle of order 1 when p is odd and 2 when p is even for p < 8.

If p = 8, the order of each cyclic permutation is 8, except for the coefficients marked with indices 4 and 2, where the order is 4 and 2, respectively.

10. Resultants and General Root-powering. Algorithm R.

In Appendix A we have shown how the circulants are connected with the theory of resultants. In this section we shall derive yet another method which lends itself to algorithms for general root-powering based on the theory of resultants. We establish our notation by presenting first some pertinent facts of the resultant theory, and then applying the theory to the special case at hand. A concise treatment of the resultant theory is contained in Ref. 11 (Fifth book, p. 368 ff).

Let

$$f = \sum_{j=0}^n a_j x^j \quad (10.1)$$

and

$$g = \sum_{j=0}^m b_j x^j \quad (10.2)$$

be polynomials of degree n and m, respectively, which have a common divisor $\mathcal{D}$ of degree i. Define f_i and g_i by $f \equiv f_i \mathcal{D}$ and $g \equiv g_i \mathcal{D}$, where

$$f_i = \sum_{j=0}^{n-i} \alpha_j x^j; \quad g_i = \sum_{j=0}^{m-i} \beta_j x^j. \quad (10.3)$$

The following theorem is evident.

Theorem. Necessary and sufficient that the polynomials f and g have a common divisor of degree not less than i is the identity

$$fg_i + gf_i = 0. \quad (10.4)$$

The identity (10.4) can be written in matrix notation as

$$(\beta_0, \beta_1, \dots, \beta_{m-i}, \alpha_0, \alpha_1, \dots, \alpha_{n-i}) M_{i-1}^{(m,n)} (1, x, x^2, \dots, x^{m+n-i})^+ = 0, \quad (10.5)$$

where

$$M_{i-1}^{(n,m)} = \begin{bmatrix} a_0 & a_1 & a_2 & \dots & 0 \\ 0 & a_0 & a_1 & \dots & 0 \\ 0 & 0 & a_0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & a_n \\ b_0 & b_1 & b_2 & \dots & 0 \\ 0 & b_0 & b_1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & b_m \end{bmatrix} \quad (10.6)$$

is a matrix with $m + n - 2(i - 1)$ rows and $n + m - (i - 1)$ columns. The first $m - (i - 1)$ rows contain a band of nonzero elements $a_0, a_1, \dots, a_n$; the remaining $n - (i - 1)$ rows contain a similar band of nonzero elements $b_0, b_1, \dots, b_m$. The determinant of the $(m + n) \times (m + n)$ matrix $M_0^{(n,m)}$ is called the resultant of f and g , $R = |M_0^{(n,m)}|$. The determinant of the first $m + n - 2i$ columns of $M_i^{(m,n)}$ is called the i th subresultant, $R_i^{(0)}$; and the determinants $R_i^{(k)}$ are obtained from $R_i^{(0)}$ if we replace column $m + n - 2i$ in $R_i^{(0)}$ by column $m + n - 2i + k$ of $M_i^{(n,m)}$. Obviously, necessary and sufficient that the identity (10.5) holds is

$$R_{i-1}^{(0)} = R_{i-1}^{(1)} = \dots = R_{i-1}^{(i-1)} = 0. \quad (10.7)$$

It can also be shown that the i roots of the common divisor $\mathcal{D}$ of f and g are given by

$$\text{const. } \mathcal{D} = R_i^{(0)} + R_i^{(1)}x + R_i^{(2)}x^2 + \dots + R_i^{(i)}x^i = 0. \quad (10.8)$$

It follows also from (10.7) and (10.8) that $R = R_1^{(0)} = \dots = R_{i-1}^{(0)} = 0$, $R_i^{(0)} \neq 0$ is necessary and sufficient for f and g to have i common (equal or unequal) zeros, provided $a_0 b_0 \neq 0$.

To apply the above theory to the general root-powering problem, let $f = P_n(x)$ and $g = P_n^{(p)}(x^p)$. Therefore, $m = np$, and

$$\left. \begin{aligned} b_{kp} &= a_k^{(p)} & (k = 0, 1, 2, \dots, n); \\ b_j &= 0 & (j \neq kp). \end{aligned} \right\} \quad (10.9)$$

Furthermore, let $\mathcal{D} = f$, which means $i = n$, $f_n = 1$. Then (10.6) becomes $M_{n-1}^{(n,np)}$ with $np - n + 1$ rows containing a_k 's and with only one row containing the b_k 's. The common divisor is derived from $M_n^{(n,np)}$ and (10.8). After some obvious simplifications, we have

$$\mathcal{D} = f = \text{const. } a_0^{n(p-1)} (a_0 + a_1 x + \dots + a_n x^n), \quad (10.10)$$

as expected.

Denote the transpose of $M_{n-1}^{(n,np)}$ by M_{n-1}^+ . Thus M_{n-1}^+ is a matrix with $np + 1$ rows and $n(p - 1) + 2$ columns, and the identity (10.5) is equivalent to

$$M_{n-1}^+ \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{np-n} \\ 1 \end{bmatrix} = \begin{bmatrix} a_0 & 0 & 0 & \dots & 0 & 0 & b_0 \\ a_1 & a_0 & 0 & \dots & 0 & 0 & 0 \\ a_2 & a_1 & a_0 & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ a_p & a_{p-1} & a_{p-2} & \dots & 0 & 0 & b_p \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & a_n & a_{n-1} & 0 \\ 0 & 0 & 0 & \dots & 0 & a_n & b_{np} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{np-n} \\ 1 \end{bmatrix} = 0, \quad (10.11)$$

which is satisfied because we know that (10.7) is true in our case. Therefore the $a_k^{(p)} = b_{kp}$ are given by

$$\begin{bmatrix} a_0^{(p)} \\ a_1^{(p)} \\ \vdots \\ a_k^{(p)} \\ \vdots \\ a_n^{(p)} \end{bmatrix} = - \begin{bmatrix} a_0 & 0 & 0 & \dots & 0 & \dots & 0 & \dots & 0 & 0 \\ a_p & a_{p-1} & a_{p-2} & \dots & a_0 & \dots & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \dots & \vdots & \dots & \vdots & \vdots \\ (kp-n \text{ zeros}) & a_n & a_{n-1} & \dots & a_0 & (m-k)p-n \text{ zeros} & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & \dots & 0 & \dots & 0 & a_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{np-n} \end{bmatrix}, \quad (10.12)$$

or explicitly by

$$a_k^{(p)} = - \sum_{j=\max(0, kp-n)}^{\min(kp, (n-1)p)} a_{kp-j} \beta_j \quad (k = 0, 1, 2, \dots, n). \quad (10.13)$$

The remaining $n(p-1)$ coefficients $b_j = 0$ in (10.11) impose constraints on the $\beta_0, \dots, \beta_{n(p-1)}$ in the form of a matrix equation

$$N_n^{(p)} \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_{np-n} \end{bmatrix} = 0, \quad (10.14)$$

where

$$N_n^{(p)} = \begin{bmatrix} a_1 & a_0 & 0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ a_2 & a_1 & a_0 & \dots & 0 & 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots \\ a_{p-1} & a_{p-2} & a_{p-3} & \dots & a_1 & a_0 & 0 & 0 & \dots & 0 & 0 & 0 \\ a_{p+1} & a_p & a_{p-1} & \dots & a_3 & a_2 & a_1 & a_0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots \\ a_{2p-1} & a_{2p-2} & a_{2p-3} & \dots & a_{p+1} & a_p & a_{p-1} & a_{p-2} & \dots & 0 & 0 & 0 \\ a_{2p+1} & a_{2p} & a_{2p-1} & \dots & a_{p+3} & a_{p+2} & a_{p+1} & a_p & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & \vdots & \vdots & \vdots & \vdots & \dots & a_{n-1} & a_{n-2} & a_{n-3} \\ 0 & 0 & 0 & \dots & \vdots & \vdots & \vdots & \vdots & \dots & a_n & a_{n-1} & a_{n-2} \\ 0 & 0 & 0 & \dots & \vdots & \vdots & \vdots & \vdots & \dots & 0 & a_n & a_{n-1} \end{bmatrix} \quad (10.15)$$

$$\begin{bmatrix} 0 & \dots & 0 & c_{0n} & c_{0,n-1} & \dots & c_{0,1} & 1 & 0 & \dots & 0 \\ 0 & \dots & 0 & c_{1n} & c_{1,n-1} & \dots & c_{1,1} & 0 & 1 & \dots & 0 \\ \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & \dots & 0 & c_{p-2,n} & c_{p-2,n-1} & \dots & c_{p-2,1} & 0 & 0 & \dots & 1 \end{bmatrix}, \quad (10.21)$$

where

$$c_{0k} = r_0 a_k \quad (k = 1, \dots, n)$$

$$c_{jn} = r_j a_n \quad (j = 1, \dots, p-2) \quad (10.22)$$

$$c_{jk} = r_j a_k + c_{j-1,k+1} \quad (k = 1, \dots, n-1; j = 1, \dots, p-2).$$

Let

$$c_k = \begin{bmatrix} c_{0,k} \\ \vdots \\ c_{p-2,k} \end{bmatrix} \quad (k = 1, 2, \dots, n); \quad c_k = 0 \quad (k > n), \quad (10.23a)$$

and define

$$u_k = c_{kp+1}, \quad S_k = [c_{kp}, c_{kp-1}, \dots, c_{(k-1)p+2}] \quad (k = 1, 2, \dots, n-1). \quad (10.23b)$$

Then (10.21) can be represented in block matrix form as

$$[u_{n-1} S_{n-1} u_{n-2} S_{n-2} \dots u_1 S_1 u_0 I]. \quad (10.24)$$

Multiplying the augmented matrix of $N_n^{(p)}$ in (10.15) by the block diagonal matrix

$$\begin{bmatrix} R & & \\ & \ddots & \\ & & R \end{bmatrix}$$

yields the partitioned matrix

$$N_n^{(p)} = \begin{bmatrix} u_0 & I & & & & \\ u_1 & S_1 & u_0 & I & & \\ u_2 & S_2 & u_1 & S_1 & u_0 & I \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ u_{n-1} & S_{n-1} & \cdot & \cdot & \cdot & u_0 & I \end{bmatrix}. \quad (10.25)$$

Similarly, the augmented vector $[\beta_0, \dots, \beta_{np-1}]^T$ is partitioned in n elements β_{kp} and n vectors

$$v_k = \begin{bmatrix} \beta_{kp+1} \\ \beta_{kp+2} \\ \vdots \\ \beta_{(k+1)p-1} \end{bmatrix} \quad (k = 0, 1, 2, \dots, n-1). \quad (10.26)$$

If we remember (10.17), the matrix equation (10.14) assumes now the form

$$N_n^{(p)} \begin{bmatrix} \beta_0 \\ v_0 \\ \vdots \\ \beta_{(n-1)p} \\ v_{n-1} \end{bmatrix} = 0. \quad (10.27)$$

Define

$$S = \begin{bmatrix} I & & & \\ S_1 & I & & \\ \vdots & & \ddots & \\ S_{n-1} & S_1 & & I \end{bmatrix}; \quad U = \begin{bmatrix} u_0 & & & \\ \vdots & \ddots & & \\ u_{n-1} & & & u_0 \end{bmatrix}; \quad (10.28)$$

Contd.

$$v = \begin{bmatrix} v_0 \\ v_1 \\ \vdots \\ v_{n-1} \end{bmatrix}; \quad \beta = \begin{bmatrix} \beta_0 \\ \beta_p \\ \vdots \\ \beta_{(n-1)p} \end{bmatrix};$$

Contd.
(10.28)

and (10.27) can be written as

$$Sv + U\beta = 0. \quad (10.29)$$

Let

$$U = SW; \quad (10.30)$$

then

$$v = -W\beta. \quad (10.31)$$

Because of the special form of the triangular block matrices S and U , the block matrix W has also the form of U , namely

$$W = \begin{bmatrix} w_0 & & & \\ & w_1 & & \\ & & \ddots & \\ & & & w_{n-1} & w_{n-2} & \dots & w_0 \end{bmatrix}. \quad (10.32)$$

It follows from (10.30) that

$$\left. \begin{aligned} w_0 &= u_0 \\ w_k &= u_k - \sum_{j=0}^{k-1} S_{k-j} w_j \quad (k = 1, 2, \dots, n-1) \end{aligned} \right\} \quad (10.33)$$

and from (10.31),

$$v_k = - \sum_{j=0}^k w_{k-j} \beta_{jp} \quad (k = 0, 1, \dots, n-1). \quad (10.34)$$

Therefore, the problem is reduced to determine the components β_{jp} of β . We proceed as follows. Define the nonnegative integers r and h by

$$n(p-1) = rp + h \quad (0 \leq h < p). \quad (10.35)$$

From (10.17) follows

$$\beta_{kp} = 0 \text{ for } r < k < n, \quad (10.36)$$

and from (10.26) follows

$$v_r = \begin{bmatrix} \beta_{rp+1} \\ \vdots \\ \beta_{np-n} \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad v_{r+1} = \dots = v_{n-1} = \begin{bmatrix} 0 \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ 0 \end{bmatrix}. \quad (10.37)$$

The vector v_r contains h nonzero components. Therefore, the number of zero components in $v_r, v_{r+1}, \dots, v_{n-1}$ equals

$$(n - r)(p - 1) - h = r. \quad (10.38)$$

Equating these r zero components to the right side of (10.34) for $k = r, r + 1, \dots, n - 1$ yields a linear system of r equations for the $(r + 1)$ unknown parameters β_{kp} ($0 \leq k \leq r$). Because of (10.36), the summation in (10.34) extends from $j = 0$ to $j = r$ only. From (10.35) follows

$$r = n - \frac{n + h}{p} \leq n - 1. \quad (10.39)$$

Thus, the general root-powering problem has been reduced to the solution of a system of linear equations of order less than n . By (10.37) and (10.34), this system of linear equations takes the block matrix form

$$\begin{bmatrix} w'_r & w'_{r-1} & w'_{r-2} & \dots & w'_0 \\ w'_{r+1} & w'_r & w'_{r-1} & \dots & w'_1 \\ w'_{r+2} & w'_{r+1} & w'_r & \dots & w'_2 \\ \vdots & \vdots & \vdots & \dots & \vdots \\ w'_{n-1} & w'_{n-2} & w'_{n-3} & \dots & w'_{n-r-1} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_p \\ \beta_{2p} \\ \vdots \\ \beta_{rp} \end{bmatrix} = 0, \quad (10.40)$$

where w'_j is a vector obtained from w_j by omitting the first h components in accordance with the remarks made above and relating to the components of v_r . The block matrix in (10.40) is a *vector Toeplitz matrix* and an algorithm for its solution is given in Ref. 27. There, it is also shown that this system can be solved by about $\frac{1}{2} r^2 p$ multiplications.

Given β , we obtain $v_1, \dots, v_r$ by (10.34) and therefore, by (10.26), know all components $\beta_0, \dots, \beta_{n(p-1)}$. The coefficients $a_j^{(p)}$ of the transformed polynomial are then given by (10.12).

To obtain a measure for the efficiency of the algorithm, we shall count the number of multiplications needed for the entire transformation. We note that (10.20) is simplified if $a_0 = 1$, which can always be achieved by not more than n divisions. Thus, the algorithm, called algorithm R, is in summary as follows.

Algorithm R.

Let $1, a_1, a_2, \dots, a_n$, be given, and define

$$t_{jk} = r_j a_k. \quad (10.41a)$$

Step 1. Compute t_{jk} ($j = 0, 1, \dots, p-2$; $k = 0, 1, \dots, n$). From (10.20) it follows that $r_0 = 1$. Obviously, $t_{0k} = a_k$, and

$$r_j = - \sum_{i=0}^{j-1} t_{i,j-i}. \quad (10.41b)$$

Therefore we obtain the following matrix:

$$T = \begin{bmatrix} t_{01} & t_{02} & \dots & t_{0n} \\ t_{11} & t_{12} & \dots & t_{1n} \\ \dots & \dots & \dots & \dots \\ t_{p-2,1} & t_{p-2,2} & \dots & t_{p-2,n} \end{bmatrix}, \quad (10.42)$$

where the elements are determined row by row, using (10.41b) for r_j and (10.41a) for $t_{j1}, \dots, t_{jn}$ ($j = 1, \dots, p-2$). Since the first row is known, only $(p-2)n$ multiplications are needed to determine T .

Step 2. Compute c_{jk} of the matrix C ,

$$C = \begin{bmatrix} c_{01} & c_{02} & \dots & c_{0n} \\ c_{11} & c_{12} & \dots & c_{1n} \\ \dots & \dots & \dots & \dots \\ c_{p-2,1} & c_{p-2,2} & \dots & c_{p-2,n} \end{bmatrix}. \quad (10.43)$$

From (10.22) follows*

*A relation useful in debugging a code is $c_{j1} = -r_{j+1}$ ($j = 0, \dots, p-3$).

$$\left. \begin{aligned} c_{0k} &= a_k = t_{0k}; \quad c_{jn} = t_{nk} \quad (k = 1, \dots, n; \quad j = 1, \dots, p-2), \\ c_{jk} &= t_{jk} + c_{j-1, k+1} \quad (j = 1, \dots, p-2, \quad k = 1, \dots, n-1). \end{aligned} \right\} \quad (10.44)$$

Thus, no multiplications (but $(p-2)(n-1)$ additions) are needed to obtain C.

Step 3. Compute $w_0, w_1, \dots, w_{n-1}$.

Let $r = n - t$; then by (10.39),

$$tp = n + h \quad (0 \leq h \leq p-1). \quad (10.45)$$

From (10.23), we conclude that

$$\left. \begin{aligned} u_k &\neq 0 \text{ for } kp + 1 \leq n \text{ or } k \leq t-1 \\ S_k &\neq 0 \text{ for } (k-1)p + 2 \leq n \text{ or } k \leq \begin{cases} t & (h < p-1) \\ t-1 & (h = p-1) \end{cases} \end{aligned} \right\} \quad (10.46)$$

Therefore, (10.33) can be presented in block matrix form as

$$\begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ \vdots \\ w_{n-1} \end{bmatrix} = - \begin{bmatrix} c_1 \\ c_{p+1} \\ \vdots \\ c_{(t-1)p+1} \\ 0 \\ \vdots \\ 0 \end{bmatrix} \begin{bmatrix} S_1 & & & & \\ & S_1 & & & \\ & & S_{t-1} & & \\ & & S_t & S_{t-1} & \\ & & & S_{t-1} & S_1 \\ & & & & S_1 \end{bmatrix} \begin{bmatrix} -1 \\ w_0 \\ w_1 \\ \vdots \\ \vdots \\ w_{n-2} \end{bmatrix} \quad (10.47)$$

To arrive at w_{t-1} , we have, not counting (-1) as a factor,

$$(1 + 2 + \dots + t-1)(p-1)^2 = \frac{t(t-1)(p-1)^2}{2} \quad (10.48a)$$

multiplications. To determine $w_t, \dots, w_{n-1}$, we need

$$r(p-1)[t(p-1) - h] = rt(p-1)^2 - rh(p-1) \quad (10.48b)$$

multiplications. This expression holds for any h , since if $h < p-1$, S_t has h zero columns; and if $h = p-1$, $S_t \equiv 0$ and again has h zero columns. Adding (10.48a) to (10.48b) yields a total number of multiplications of

$$\frac{2r + t - 1}{2} t(p - 1)^2 - rh(p - 1) = \frac{n + r - 1}{2} t(p - 1)^2 - rh(p - 1)$$

$$= \frac{1}{2} (n + r) np - \text{lower-order terms.}$$

(10.48c)

Step 4. Calculate $\beta_0, \beta_p, \dots, \beta_{rp}$. Let $\beta_0 = 1$, and solve (10.40): If $r \leq p$, the number of multiplications required for the solution is

$$\frac{1}{3} r^3 + \text{lower-order terms} \quad (r \leq p). \quad (10.49a)$$

If $r > p$, and using the method given in Ref. 27, the number of multiplications needed is

$$\frac{p}{6} (3r^2 - p^2) + \text{lower-order terms} \quad (r > p). \quad (10.49b)$$

Step 5. Calculate β_j ($j = 1, \dots, n(p - 1); j \neq kp$).

If we let

$$w_k^\dagger = [w_{0k}, w_{1k}, \dots, w_{p-2,k}] \quad (k = 0, 1, \dots, r),$$

it follows from (10.34), (10.26), and $\beta_0 = 1$,

$$\left. \begin{aligned} \beta_i &= -w_{i-1,0} & (k = 0) \\ \beta_{kp+i} &= -(w_{i-1,k} + \sum_{j=1}^k w_{i-1,k-j} \beta_{jp}) & (k = 1, 2, \dots, r) \end{aligned} \right\} \quad (10.50)$$

where

$$1 \leq i \leq \begin{cases} p - 1 & \text{for } k = 0, 1, \dots, r \\ h & \text{for } k = r. \end{cases}$$

The number of multiplications needed in (10.50) is

$$\begin{aligned} (p - 1)(1 + 2 + \dots + r - 1) + hr &= \frac{1}{2} r(r - 1)(p - 1) + hr \\ &= \frac{r^2 p}{2} + \text{lower-order terms.} \end{aligned} \quad (10.51)$$

Step 6. Compute $a_j^{(p)}$ ($j = 1, 2, \dots, n$); $a_0^{(p)} = 1$.

We use (10.12). Define

$$\ell \equiv \left[\frac{n}{p} \right] = \begin{cases} t & (h = 0) \\ t - 1 & (h > 0) \end{cases} \quad (10.52)$$

To compute $a_0^{(p)}, \dots, a_\ell^{(p)}$ we need

$$0 + p + 2p + \dots + \ell p = \frac{p}{2} \ell(\ell + 1) \quad (10.53a)$$

multiplications. To compute $a_n^{(p)}, \dots, a_{n-\ell}^{(p)}$ we need

$$1 + (p + 1) + (2p + 1) + \dots + (\ell p + 1) = \ell + 1 + \frac{p}{2} \ell(\ell + 1) \quad (10.53b)$$

multiplications. To compute the remaining coefficients, if any, $a_{\ell+1}^{(p)}, \dots, a_{n-\ell-1}^{(p)}$, we need

$$(n - 2\ell - 1)(n + 1) = (n - \ell)(n + 1) - (\ell + 1)(n + 1) \quad (10.53c)$$

multiplications. Adding the last three expressions yields, for the total number of multiplications in Step 6,

$$(n - \ell)(n + 1) - (n - p\ell)(\ell + 1) < (r + 1)(n + 1) + (p - h)t. \quad (10.54)$$

Adding the number of multiplications in Steps 1 to 6 gives the approximate number of multiplications for the complete transformation,

$$\frac{p}{2} (n^2 + rn + r^2) + \begin{cases} \frac{1}{3} r^3 & \text{if } r \leq p \\ \frac{p}{6} (3r^2 - p^2) & \text{if } r > p \end{cases}, \quad (10.55)$$

where the lower-order terms are not included.

11. General Root-powering by Elimination Methods. Algorithms E1 to E4.

In (10.11) the coefficients $b_0, b_p, \dots, b_{np}$ are unknown. Therefore we may write (10.11) as

$$[A \ B]y = 0, \quad (11.1)$$

where

$$A = \begin{bmatrix} a_0 & & & 0 \\ & \ddots & & \\ & & a_n & \\ & & & \ddots \\ 0 & & & & a_n \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}, \quad y = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_{np-n} \\ b_0 \\ \vdots \\ b_{np} \end{bmatrix}. \quad (11.2)$$

The rows A and B will be numbered 0, 1, 2, ..., np. The columns of A will be numbered from 0 to np - n, and the columns of B from 0 to n. Thus [A B] is an (np + 1) x (np + 2) matrix, and (11.1) is a homogeneous equation for the (np + 2) unknowns β_k ($k = 0, 1, \dots, np - n$) and b_{kp} ($k = 0, 1, \dots, n$). We shall develop four algorithms which solve (11.1) efficiently for $b_{kp} = a_k^{(p)}$ only. To this end, we introduce a square matrix R, similar to the matrix R of (10.19) but of order (np + 1), which is defined by

$$R = \begin{bmatrix} r_0 & & & & \\ & r_1 & & & \\ & & r_0 & & \\ & r_2 & & r_1 & \\ & & & & r_0 \\ & \vdots & & \vdots & \\ & \vdots & & \vdots & \\ & \vdots & & \vdots & \\ r_{np} & r_{np-1} & r_{np-2} & \dots & r_0 \end{bmatrix}; \quad R^{-1} = \begin{bmatrix} a_0 & & & & 0 \\ & \ddots & & & \\ & & a_n & & \\ & & & \ddots & \\ 0 & & & & a_n \end{bmatrix}. \quad (11.3)$$

To compute r_j ($j = 0, \dots, np$), we use the following recurrence relations:

$$\begin{aligned} s_{00} &= 1, \quad r_0 = a_0^{-1} \\ s_{i,j-i} &= a_i r_{j-i} \text{ for } i = 1, 2, \dots, j. \\ s_{0j} &= - \sum_{i=1}^{\min(j,n)} s_{i,j-i} \left. \vphantom{\sum_{i=1}^{\min(j,n)}} \right\} \text{ for } j = 1, 2, \dots, np. \\ r_j &= -r_0 s_{0j} \end{aligned} \quad (11.4)$$

The elements s_{ij} calculated in (11.4) can be arranged as follows:

Diagram illustrating the calculation of the matrix element $s_{0,np}$ using the chain rule. The diagram shows a grid of points s_{ij} and a path of dashed lines representing the chain rule decomposition. The path starts at $s_{0,np}$ and goes down to $s_{1,np-1}$, then down to $s_{1,n-1}$, then down to $s_{1,j-1}$, then down to $s_{j,0}$, then down to $s_{n,np-n}$, then right to x , and finally down to $s_{n,np-n}$ again. The region to the right of the path is labeled "not calculated".

(11.5)

and A_1 and R_1 are rectangular matrices with n rows and $(np - n + 1)$ columns

$$A_1 = \begin{bmatrix} 0 & \dots & 0 & a_n & \dots & a_1 \\ \vdots & & & \vdots & & \vdots \\ 0 & \dots & 0 & & & a_n \end{bmatrix}, \quad R_1 = \begin{bmatrix} r_{np-n+1} & \dots & r_1 \\ \vdots & & \vdots \\ r_{np} & \dots & r_n \end{bmatrix}. \quad (11.8b)$$

We also introduce A_2 in correspondence to R_2 , which are square matrices of order n :

$$A_2 = \begin{bmatrix} a_0 & & 0 \\ \vdots & \ddots & \\ a_{n-1} & \dots & a_0 \end{bmatrix}; \quad R_2 = \begin{bmatrix} r_0 & & 0 \\ \vdots & \ddots & \\ r_{n-1} & \dots & r_0 \end{bmatrix}. \quad (11.8c)$$

B_0 is a matrix with $(np - n + 1)$ rows and $(n + 1)$ columns, B_1 is a matrix with n rows and $(n + 1)$ columns:

$$B_0 = \begin{bmatrix} 1 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \vdots & & \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \vdots & 0 & & \vdots & \vdots & & \vdots \\ \vdots & 1 & & \vdots & \vdots & & \vdots \\ \vdots & 0 & & \vdots & \vdots & & \vdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ \vdots & \vdots & & 0 & \vdots & & \vdots \\ \vdots & \vdots & & 1 & \vdots & & \vdots \\ \vdots & \vdots & & 0 & \vdots & & \vdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 & \dots & 0 & 0 & \dots & 0 & 0 \\ \vdots & & \vdots & 0 & & \vdots & \vdots \\ \vdots & & \vdots & 1 & & \vdots & \vdots \\ \vdots & & \vdots & 0 & & \vdots & \vdots \\ \vdots & & \vdots & \vdots & & \vdots & \vdots \\ \vdots & & \vdots & \vdots & & 0 & \vdots \\ \vdots & & \vdots & \vdots & & 1 & \vdots \\ \vdots & & \vdots & \vdots & & 0 & \vdots \\ \vdots & & \vdots & \vdots & & \vdots & 0 \\ 0 & \dots & 0 & 0 & \dots & 0 & 1 \end{bmatrix}. \quad (11.8d)$$

Let r , t , h be defined as in (10.35) and (10.45),

$$\left. \begin{aligned} r &= \left\lfloor \frac{n(p-1)}{p} \right\rfloor; & rp + h &= n(p-1); & t-1 &= \left\lfloor \frac{n-1}{p} \right\rfloor; \\ tp &= n+h; & r+t &= n. \end{aligned} \right\} \quad (11.9)$$

The last t columns of B_0 and the $r+1$ first columns of B_1 contain only zero elements. Column k ($k = 0, \dots, r$) in B_0 contains kp zero elements followed by a unit element 1 and $(np - n - kp)$ zero elements. In particular, column r of B_0 has rp zeros followed by a unit element and h zeros. Column $(n-k)$ ($k = 0, \dots, t$) in B_1 ends with kp zeros, preceded by a unit element 1 and $(n - kp - 1)$ zero elements. In particular, column

$r + 1 = n - (t - 1)$ of B_1 has $(p - h - 1)$ zeros, followed by a unit and $(t - 1)p$ zeros. From (11.3) and (11.8b,c), it follows that

$$R_0 A_0 = I, \quad R_1 A_0 + R_2 A_1 = 0, \quad A_1 R_0 + A_2 R_1 = 0, \quad A_2 R_2 = I. \quad (11.10)$$

Finally, we partition y such that

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \quad y_2 = \begin{bmatrix} b_0 \\ \vdots \\ b_{np} \end{bmatrix}. \quad (11.11a)$$

Then (11.1) takes the form

$$\begin{bmatrix} A_0 & B_0 \\ A_1 & B_1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \dot{0}. \quad (11.11b)$$

The problem is to determine y_2 efficiently by suitable algorithms.

Algorithm E1.

We multiply (11.11b) by R . Then, by (11.7) and (11.10),

$$\begin{bmatrix} R_0 & 0 \\ R_1 & R_2 \end{bmatrix} \begin{bmatrix} A_0 & B_0 \\ A_1 & B_1 \end{bmatrix} = \begin{bmatrix} I & R_0 B_0 \\ 0 & R_1 B_0 + R_2 B_1 \end{bmatrix}. \quad (11.12a)$$

Thus the problem (11.11b) is reduced to

$$[R_1 B_0 + R_2 B_1] y_2 = 0, \quad (11.12b)$$

or explicitly

$$\begin{bmatrix} r_{np-n+1} & r_{n(p-1)-n+1} & \dots & r_{n+1} & 0 & \dots & 0 & 0 \\ \vdots & \vdots & & \vdots & 0 & & \vdots & \vdots \\ \vdots & \vdots & & \vdots & 0 & & \vdots & \vdots \\ \vdots & \vdots & & \vdots & r_0 & & \vdots & \vdots \\ \vdots & \vdots & & \vdots & \vdots & & 0 & \vdots \\ \vdots & \vdots & & \vdots & \vdots & & r_0 & \vdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\ \vdots & \vdots & & \vdots & \vdots & & \vdots & 0 \\ r_{np} & r_{(n-1)p} & \dots & r_{tp} & r_{(t-1)p} & \dots & r_p & r_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_p \\ \vdots \\ b_{np} \end{bmatrix} = 0. \quad (11.12c)$$

The matrix in (11.12c) is formed from R in (11.3) by taking the last n elements of the rows kp ($k = 0, \dots, n$). Note that $tp = (n - r)p$ and $h + 1 = n(p - 1) - rp + 1 = n(p - r) - n + 1$. Define now

$$v_0 = \begin{bmatrix} r_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad v_k = \begin{bmatrix} r_{kp} \\ r_{kp-1} \\ \vdots \\ r_{kp-p+1} \end{bmatrix}; \quad w_0 = \begin{bmatrix} r_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad w_k = \begin{bmatrix} r_{kp} \\ r_{kp-1} \\ \vdots \\ r_{kp-p+1} \end{bmatrix}, \quad (11.13).$$

(k = 0, 1, 2, ..., n).

With this notation, (11.12c) can be written in the form of a vector Toeplitz matrix in which the vector elements v_k (k < 0) are zero,

$$\begin{bmatrix} v_0 & v_1 & v_2 \dots v_{t-2} & v_{t-1} & \dots & v_n \\ & v_0 & v_1 \dots v_{t-3} & v_{t-2} \dots v_{n-1} & & \\ & & \ddots & \ddots & \ddots & \\ & & & v_1 \dots v_r & & \\ & & & & w_0 & \dots & w_{r+1} \end{bmatrix} \begin{bmatrix} b_{np} \\ b_{np-p} \\ \vdots \\ b_p \\ b_0 \end{bmatrix} = 0 \quad (11.14)$$

In its explicit form, the matrix in (11.14) is centrosymmetric to the matrix in (11.12c). Equation (11.14) can be efficiently transformed into an equation with an upper triangular matrix (see Ref. 27).

$$\begin{bmatrix} r_0^{(0)} & r_p^{(0)} & r_{2p}^{(0)} & \dots & r_{(n-1)p}^{(0)} & r_{np}^{(0)} \\ 0 & r_{p-1}^{(1)} & r_{2p-1}^{(1)} & \dots & r_{(n-1)p-1}^{(1)} & r_{np-1}^{(1)} \\ 0 & 0 & r_{2(p-1)}^{(2)} & \dots & r_{(n-1)p-2}^{(2)} & r_{np-2}^{(2)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & r_{(n-1)(p-1)}^{(n-1)} & r_{np-n+1}^{(n-1)} \end{bmatrix} \begin{bmatrix} b_{np} \\ b_{(n-1)p} \\ b_{(n-2)p} \\ \vdots \\ b_0 \end{bmatrix} = 0 \quad (11.15)$$

the algorithm for this transformation is given by the recurrence relations

$$r_{kp}^{(0)} = r_{kp}, \quad r_{kp-1}^{(1)} = r_{kp-1} \quad (k = 0, 1, \dots, n); \quad (11.16a)$$

and for $i = 1, 2, \dots, n - 2$,

$$r_{kp-i-j}^{(i+1)} = r_{kp-i-j}^{(i)} - \frac{r_{i(p-1)-j}^{(i)}}{r_{i(p-1)}^{(i)}} r_{kp-i}^{(i)} \quad (k = i + 1, \dots, n; \quad 1 \leq j < \min(p, n - i)). \quad (11.16b)$$

If $p < n$, the number of multiplications to carry out the step $i \rightarrow i + 1$ of (11.16b) is $(p - 1)(n - i)$ for $i \leq n - p$; for $i = n - p + 1, \dots, n - 2$, we need $(p - 2) \cdot (p - 1), \dots, (1) \cdot (2)$ multiplications. Therefore the total number of multiplications needed in (11.16) is

$$\sum_{i=1}^{n-p} (p - 1)(n - i) + \sum_{i=1}^{p-2} i(i + 1) = (p - 1) \left\{ \frac{(n - p)(n + p - 1)}{2} + \frac{p}{3}(p - 2) \right\} \\ = (p - 1) \left\{ \frac{n^2 - n}{2} - \frac{p^2 + p}{6} \right\} < \frac{pn^2}{2} \quad (p < n). \quad (11.17a)$$

If $p \geq n$, the matrix in (11.14) reduces to $[v_0, v_1, \dots, v_n]$, and, because of the special form of v_0 , the triangularization requires

$$\sum_{i=1}^{n-2} i(i + 1) = \frac{n(n - 1)(n - 2)}{2} < \frac{n^3}{3} \quad (p \geq n) \quad (11.17b)$$

multiplications. The number of subtractions is equal to the number of multiplications, and the number of divisions is $n - 2$. To obtain $a_k^{(p)} = b_{kp}$ from (11.15), one chooses b_0 conveniently. If we assume $b_0 = 1$, back substitution in (11.15) requires

$$\sum_{i=1}^{n-1} i = \frac{n(n - 1)}{2} \quad (11.18)$$

multiplications (and additions) and n ($n - 1$ if $a_0 = 1$) divisions.

In summary, Algorithm E1 consists of:

Step 1. Compute r_j ($j = 0, 1, \dots, np$) by (11.4).

Step 2. Compute $r_{kp-i}^{(i)}$ ($i = 0, \dots, n - 1$; $k = i, i + 1, \dots, n$) by (11.16).

Step 3. Compute $a_k^{(p)} = b_{kp}$ by back substitution in (11.15).

The total number of multiplications needed for the entire transformation, obtained by combining the numbers given in (11.6), (11.17), and (11.18) is

$$\frac{3n^2p}{2} + \text{lower-order terms}, \quad (11.19a)$$

for $p < n$,

and

$$n^2\left(p + \frac{n}{3}\right) + \text{lower-order terms} \quad (11.19b)$$

for $p \leq n$.

Algorithm E2

Instead of (11.12a), we take

$$\begin{bmatrix} R_0 & 0 \\ A_2 R_1 & I \end{bmatrix} \begin{bmatrix} A_0 & B_0 \\ A_1 & B_1 \end{bmatrix} = \begin{bmatrix} I & R_0 B_0 \\ 0 & A_2 R_1 B_0 + B_1 \end{bmatrix}. \quad (11.20a)$$

We used (11.10) to derive

$$A_2 R_1 A_0 + A_1 = A_2 [R_1 A_0 + R_2 A_1] = 0.$$

Thus the problem (11.11b) is reduced to

$$[A_2 R_1 B_0 + B_1] y_2 = 0. \quad (11.20b)$$

This equation can also be obtained by premultiplying (11.12b) with A_2 . The last t columns of the matrix in (11.20b) are identical to the last t columns of B_1 . The first $r+1$ columns of the matrix in (11.20b) can be obtained without multiplications from the submatrix

$$\begin{bmatrix} s_{0,h+1} & \dots & s_{0,np-n+1} & \dots & s_{0,np} \\ \vdots & & \vdots & & \vdots \\ s_{n-1,h+1} & \dots & s_{n-1,np-n+1} & \dots & s_{n-1,np} \end{bmatrix} \quad (11.21)$$

of (11.5) as follows. To obtain the elements of the first column, select the cotriangular matrix from the last n columns of (11.21) and add the elements along the parallels to the codiagonal. Thus the elements are

$$s_{0,np-n+1}; s_{1,np-n+1} + s_{0,np-n+2}; \dots; s_{n-1,np-n+1} + \dots + s_{0,np}.$$

For the next column we take the cotriangular matrix in (11.21) p places to the left and proceed as before, and so on. The last triangle so obtained from (11.21) is

$$\begin{bmatrix} s_{0,h+1} & \dots & s_{0,h+n} \\ \vdots & & \vdots \\ s_{n-1,h+1} & & \vdots \end{bmatrix}$$

where, by (11.9), $s_{0,h+n} = s_{0,tp}$. The matrix $A_2 R_1 B_0 + B_1$ of (11.20b) is not a vector Toeplitz matrix anymore. However, by appropriate rearrangement of the rows, we obtain an equation of the form

$$\begin{bmatrix} G & 0 \\ F & I \end{bmatrix} y_2 = 0, \quad (11.20c)$$

where G is only an $(r) \times (r+1)$ matrix, F is a $(t) \times (r+1)$ matrix, and I is the $t \times t$ identity matrix. The triangularization of G can be accomplished by

$$\sum_{i=1}^{r-1} i(i+1) = \frac{1}{3} r(r+1)(r-1) \quad (11.22)$$

multiplications. Letting $b_0 = 1$, the back substitution requires, for $b_1, \dots, b_{rp}; b_{(r+1)p}, \dots, b_{np}$,

$$\sum_{i=1}^{r-1} i + tr = \frac{r}{2} (n + t - 1) \quad (11.23)$$

multiplications.

In summary, Algorithm E2 consists of:

Step 1. Compute matrix (11.5) by (11.4).

Step 2. Form $A_2 R_1 B_0$ from (11.21).

Step 3. Solve (11.20b) to obtain $a_k^{(p)} = b_{kp}$.

The total number of multiplications needed for the entire transformation is obtained by combining the numbers given by (11.6), (11.22), and (11.23) and is

$$n^2 p + \frac{r^3}{3} + \text{lower-order terms.} \quad (11.24)$$

Algorithm E3

Instead of (11.12a) or (11.20a), we take

$$\begin{bmatrix} R_0 & 0 \\ A_1 R_0 & -I \end{bmatrix} \begin{bmatrix} A_0 & B_0 \\ A_1 & B_1 \end{bmatrix} = \begin{bmatrix} I & R_0 B_0 \\ 0 & A_1 R_0 B_0 - B_1 \end{bmatrix}. \quad (11.25a)$$

Thus the problem (11.11b) is reduced to

$$[A_1 R_0 B_0 - B_1] y_2 = 0. \quad (11.25b)$$

This equation can also be obtained from (11.20b) since, by (11.10), $A_1 R_0 = -A_2 R_1$. The first $r + 1$ columns of the matrix in (11.25b) can be obtained without multiplications from the submatrix,

$$\begin{array}{cccc} s_{1,0} & \cdots & s_{1,n(p-1)} & \\ \vdots & & \vdots & \\ s_{n,0} & \cdots & s_{n,n(p-1)} & \end{array} \quad (11.26a)$$

The elements of the first column are obtained from the cotriangular matrix

$$\begin{array}{c} s_{1,n(p-1)} \\ \vdots \\ s_{n,n(p-2)+1} \cdots s_{n,n(p-1)} \end{array} \quad (11.26b)$$

by taking the sum of the s_{ij} 's along the direction of the codiagonal. Thus, these elements are

$$s_{n,n(p-2)+1} + \cdots + s_{1,n(p-1)}; \cdots; s_{n,n(p-1)-1} + s_{n-1,n(p-1)}; s_{n,n(p-1)}.$$

For the calculations of the elements of the second column, we shift the triangle formed by (11.26b) by p places to the left and proceed as before, and so on. Eventually the cotriangular matrices will be replaced by matrices of trapezoidal form. The elements of the $(r + 1)$ st column are determined from the matrix

$$\begin{array}{c} s_{1,h} \\ \vdots \\ s_{h+1,0} \\ \vdots \\ s_{n,0} \cdots s_{n,h} \end{array} \quad (11.26c)$$

The remaining t columns of the matrix in (11.25b) are those of $-B_1$. Therefore, we can rearrange the rows so that we obtain an equation of the form (11.20c). The advantage of Algorithm E3 over Algorithm E2 is that (11.26a) can be obtained by

$$n \cdot n(p - 1) = n^2(p - 1)$$

multiplications, assuming as previously, $a_0 = 1$; this results in a saving of $n(n - 1)/2$ multiplications.

In summary, Algorithm E3 consists of:

Step 1. Compute matrix (11.26a)

$$\left. \begin{aligned} r_0 &= a_0^{-1}, \quad s_{i,0} = a_i r_0 \quad (i = 1, \dots, n) \\ r_j &= -r_0 \sum_{k=1}^{\min(j,n)} s_{k,j-k} \\ s_{ij} &= a_i r_j \quad (i = 1, 2, \dots, n) \end{aligned} \right\} \quad j = 1, 2, \dots, n(p-1). \quad (11.27)$$

Step 2. Form $A_1 R_0 B_0$ from (11.26).

Step 3. Solve (11.25b).

The number of multiplications needed for the entire transformation is of the same order as given in (11.24), but less than that for Algorithm E2.

Algorithm E4.

We introduce the matrices G and E ,

$$G = \begin{bmatrix} g_n & g_{n-1} & \dots & g_1 \\ & g_n & \dots & g_2 \\ & & \ddots & \vdots \\ & & & g_n \end{bmatrix}, \quad E = \begin{bmatrix} 0 & \dots & 0 & 1 & \dots & 0 \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ \vdots & & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 0 & \dots & 1 \end{bmatrix}, \quad (11.28a)$$

defined by

$$GA_1 = E, \quad (11.28b)$$

$$g_n = a_n^{-1}; \quad g_k = -g_n \sum_{j=0}^{n-k-1} g_{n-j} a_{k+j}. \quad (11.28c)$$

Instead of (11.25a), we take

$$\begin{bmatrix} I & 0 \\ 0 & G \end{bmatrix} \begin{bmatrix} R_0 & 0 \\ A_1 R_0 & -I \end{bmatrix} \begin{bmatrix} A_0 & B_0 \\ A_1 & B_1 \end{bmatrix} = \begin{bmatrix} R_0 & 0 \\ ER_0 & -G \end{bmatrix} \begin{bmatrix} A_0 & B_0 \\ A_1 & B_1 \end{bmatrix} = \begin{bmatrix} I & R_0 B_0 \\ 0 & ER_0 B_0 - GB_1 \end{bmatrix}. \quad (11.29a)$$

Thus, the problem (11.11b) is reduced to

$$[ER_0 \quad -G] \begin{bmatrix} B_0 \\ B_1 \end{bmatrix} y_2 = [ER_0 B_0 - GB_1] y_2 = 0. \quad (11.29b)$$

Explicitly, the block matrix

$$[ER_0 - G] = \begin{bmatrix} r_{n(p-2)+1} & \dots & r_0 & 0 & \dots & 0 & g_n & \dots & g_1 \\ \vdots & & & & & & & & \\ r_{n(p-1)} & \dots & r_0 & 0 & \dots & 0 & g_n & & \end{bmatrix}, \quad (11.29c)$$

and $ER_0B_0 - GB_1$ is obtained by extracting columns 0, p, 2p, ..., np from (11.29c); i.e.,

$$C_0 = ER_0B_0 - GB_1 = \begin{bmatrix} r_{h+rp-n+1} & \dots & r_{h+tp-n+1} & 0 & \dots & 0 & g_{t(p-1)+1} & \dots & g_1 \\ \vdots & & \vdots & & & & g_n & & \\ & & & & & 0 & 0 & & \\ & & & & & r_0 & & & \\ & & & & & & & & \\ r_{h+rp} & \dots & r_{h+tp} & \dots & r_h & 0 & \dots & 0 & g_n \end{bmatrix}, \quad (11.30a)$$

where we used $rp + h = n(p - 1)$ and $tp = n + h$ to simplify the notation of the key elements. Furthermore, let $t^* = t$ if $2h < p - 1$, and $t^* = t - 1$ if $2h \geq p - 1$. Thus,

$$r_{h+t^*p-n+1} = \begin{cases} r_{2h+1} & \text{if } 2h < p - 1, t^* = t \\ r_{2h-p+1} & \text{if } 2h \geq p - 1, t^* = t - 1 \end{cases}$$

is never identically zero, but stands to the left of a zero element in the matrix (11.30a). Hence, each row contains at least $(t - 1)$ zeros between the r - and g -elements. The matrix (11.30a) has n rows and $n + 1$ columns. It is a vector Toeplitz matrix and can be inverted as described in Ref. 27. Because of the special form of (10.30a), we suggest the following method.

In the first t iterations as given below, we bring (11.30a) into the form

$$C_t = \begin{bmatrix} r_{h+rp-n+1}^{(0)} & \dots & r_{2h+1}^{(0)} & \dots & 0 & g_{tp-p+1}^{(0)} & \dots & g_1^{(0)} \\ r_{h+rp-n+2}^{(1)} & \dots & r_{2h-2}^{(1)} & \dots & 0 & 0 & g_{tp-2p+2}^{(1)} & \dots & g_2^{(1)} \\ \vdots & & \vdots & & \vdots & & & & \vdots \\ r_{h+rp-n+t}^{(t-1)} & \dots & r_{2h+t}^{(t-1)} & \dots & 0 & 0 & \dots & 0 & g_t^{(t-1)} \\ r_{h+rp-n+1}^{(t)} & \dots & r_{2h+t+1}^{(t)} & \dots & 0 & 0 & \dots & 0 & 0 \\ \vdots & & \vdots & & \vdots & & & & \vdots \\ & & & & 0 & & & & \\ & & & & r_0^{(t)} & & & & \\ & & & & \vdots & & & & \\ r_{h+rp}^{(t)} & \dots & r_{tp+h}^{(t)} & \dots & r_n^{(t)} & 0 & \dots & 0 & 0 \end{bmatrix}. \quad (11.30b)$$

All elements below

$$g_{tp-ip+i}^{(i-1)} \quad (i = 1, 2, \dots, t)$$

are made zero. In the following $r - 1$ iterations, all elements above

$$r_{h+ip-i}^{(t+i)} \quad (i = 0, \dots, r - 1)$$

in the last r rows are made zero. Therefore, the last r rows of (11.30b) are transformed into

$$\begin{array}{ccccccc} r_{h+rp-r+1}^{(t+r-1)} & r_{h+(r-1)p-(r-1)}^{(t+r-1)} & 0 & \dots & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ & & r_{h+2p-2}^{(t+2)} & & 0 & & & \\ & & & & r_{h+p-1}^{(t+1)} & & & \\ r_{h+rp}^{(t)} & r_{h+(r-1)p}^{(t)} & \dots & \dots & r_h^{(t)} & 0 & \dots & 0 \end{array} \quad (11.30c)$$

and after $t + (r - 1) = n - 1$ iterations, the transformed matrix (11.30a) takes the form

$$C_{n-1} = \left[\begin{array}{c|c|c} \begin{array}{c} \text{vertical bar} \\ \text{vertical bar} \end{array} & \begin{array}{c} \text{upper trapezoid} \\ \text{lower triangle} \end{array} & \begin{array}{c} \text{upper triangle} \\ \text{lower triangle} \end{array} \end{array} \right] \quad (11.30d)$$

with $t + r = n$ rows and $1 + r + t = n + 1$ columns. Hence, the problem (11.29b) is reduced to solving

$$C_{n-1} y_2 = 0. \quad (11.31)$$

If we assume b_0 to be given, b_{kp} ($k = 1, \dots, n$) are calculated by "back substitution" in an obvious way.

An algorithm that performs the transformation

$$C = C_1 \rightarrow C_2 \rightarrow \dots \rightarrow C_t \rightarrow \dots C_{n-1}$$

is obtained as follows. First define

$$r_j = 0 \quad (0 > j > h - p); \quad r_j = c_{n+p-h+j}^{(0)} \quad \text{for } j = 0, \dots, n(p-1), \quad (11.32a)$$

$$c_j^{(0)} = \begin{cases} g_j & \text{for } j = 1, 2, \dots, n \\ 0 & \text{for } j = n+1, \dots, n+p-h-1 \\ r_{h-p-n+j} & \text{for } j = n+p-h, \dots, (r+1)p+n. \end{cases} \quad (11.32b)$$

There are $np+1$ nonzero elements $c_j^{(0)}$. The elements $c_1^{(0)}$ are arranged in a matrix with n rows and $r+1$ columns,

$$C^{(0)} = \begin{bmatrix} c_{kp+j}^{(0)} \end{bmatrix} = \begin{bmatrix} c_{(r+1)p+1}^{(0)} & \dots & c_1^{(0)} \\ \vdots & & \vdots \\ c_{(r+1)p+n}^{(0)} & \dots & c_n^{(0)} \end{bmatrix}. \quad (11.33a)$$

Thus, if we omit, in C_0 of (11.30a), the $(t-1)$ zeros to the left of the g elements, we obtain $C^{(0)}$ of (11.33a). Hence we have a simple one-to-one correspondence established between $C_0 \leftrightarrow C^{(0)}$, $\dots$, $C_{n-1} \leftrightarrow C^{(n-1)}$ calculated below. It is clear that $C^{(0)}$ is again a Toeplitz matrix. On the elements of $C^{(0)}$ we perform the following transformations:

The recurrence relation is, for $i = 1, 2, \dots, t$,

$$c_{kp+j}^{(i)} = c_{kp+j}^{(i-1)} - \frac{c_{(t-i)p+j}^{(i-1)}}{c_{(t-i)p+i}^{(i-1)}} c_{kp+j}^{(i-1)} \quad (11.33b)$$

for

$$j = \begin{cases} 2, 3, \dots, p-h & \text{for } i = 1 \\ i+1, \dots, i+p-1 & \text{for } i > 1 \end{cases}$$

and

$$k = 0, 1, \dots, t-i-1; \quad t, \dots, \left(r+1 + \left\lceil \frac{n-j}{p} \right\rceil\right).$$

The elements

$$c_{(t-i)p+j}^{(i-1)}$$

are eliminated (made zero) and are not calculated. Note that for $i = 1$,

$$c_{(t-1)p+h}^{(0)} = c_n^{(0)} = g_n.$$

Thus we eliminate successively

$$c_n^{(0)}, \dots, c_{n-p+h+2}^{(0)}; \quad c_{n-p+h+3}^{(1)}, \dots, c_{n-2p+h+5}^{(1)}; \quad \dots; \quad c_{t+p-1}^{(t-1)}, \dots, c_{t+1}^{(t-1)},$$

or loosely expressed $g_n, g_{n-1}, \dots, g_{t+1}$. Therefore, (11.33b) brings (11.30a) into (11.30b). For the next $(r-1)$ iterations, we have only to consider the matrix

$$\begin{bmatrix} c_{(r+1)p+t+1}^{(t)} & \dots & 0 \\ \vdots & & \vdots \\ c_{(r+1)p+n}^{(t)} & \dots & c_{p+n}^{(t)} \end{bmatrix} = \begin{bmatrix} r_{rp+h-r+1}^{(t)} & \dots & 0 \\ \vdots & & \vdots \\ r_{rp+h}^{(t)} & \dots & r_h^{(t)} \end{bmatrix}.$$

The following recurrence relation is, for $(t+i)$, $i = 1, 2, \dots, r-1$,

$$c_{n+kp-j}^{(t+i)} = c_{n+kp-j}^{(t+i-1)} - \frac{c_{n+ip-j}^{(t+i-1)}}{c_{n+1+(p-1)i}^{(t+i-1)}} c_{n+1+kp-i}^{(t+i-1)} \quad (11.33c)$$

for

$$j = \begin{cases} 1, 2, \dots, h & \text{if } i = 1 \\ i, i+1, \dots, p+i-2 & \text{if } i < 1 \end{cases}; \quad j < r,$$

$$k = i+1, i+2, \dots, r+1.$$

The elements

$$c_{n+ip-j}^{(t+i-1)}$$

are eliminated and therefore are not calculated. Thus for $i = 1$,

$$c_{n+p-1}^{(t)} = r_{h-1}^{(t)}, \dots, c_{n+p-h}^{(t)} = r_0^{(t)}$$

are eliminated; for $i = 2$,

$$c_{n+2p-2}^{(t+1)} = r_{h+p-2}^{(t+1)}, \dots, c_{n+2p-p}^{(t+1)} = r_h^{(t+1)}$$

are eliminated, and so on, until for $i = r - 1$ ($t = n - 1$), we arrive at (11.30c), disregarding the last t zero columns. Therefore, we can construct (11.30d). Assuming $b_0 = 1$, then from

$$c^{(n-1)} \rightarrow c_{n-1}; \quad c_{n-1}y_2 = 0, \quad (11.34)$$

$b_p, b_{2p}, \dots, b_{np}$ are obtained by back substitution, starting with

$$c_{(r+1)p+t+1}^{(n-1)} + c_{rp+t+1}^{(n-1)} b_p = 0.$$

In summary, Algorithm E4 consists of:

Step 1. Compute $c_j^{(0)}$ for $j = 1, 2, \dots, (n+1)p - h$, by (11.32b), where g_j is determined by (11.28c) for $j = n, \dots, 1$; and r_j by (11.4) for $j = 0, 1, \dots, n(p-1)$.

Step 2. Compute $c_j^{(i)}$ for $i = 0, \dots, n-1$, by (11.33b,c).

Step 3. Compute $a_k^{(p)} = b_{kp}$ by back substitution in $c_{n-1}y_2 = 0$.
(See (11.34) and (11.30d).)

The computation of the number of multiplications needed for the entire transformation is more complicated than for Algorithm E1. We must consider that the $p - h - 1$ elements $c_j^{(0)}$ ($j = n+1, \dots, n+p-h-1$) are identically zero. Furthermore, we triangularize first a matrix of order t , then a matrix of order r . Finally, the back substitution involves the matrix (11.30d). In Steps 1 and 2, Algorithm E4 requires less multiplications than Algorithm E1. Instead of giving the rather complicated formulas for the exact number of total multiplications, we describe the approximate number of multiplications saved in each step as compared with Algorithm E1. In Step 1, the calculation of $g_1, \dots, g_n$ saves about $n^2/2$ multiplications against the calculations of $r_{n(p-1)}, \dots, r_{np}$ of Algorithm E1. In Step 2, the construction of (11.30d) requires about the same number of multiplications as the construction of (11.15). In Step 3, because the matrix (10.30d) contains approximately $t(t-1)(2p-1)/2p$ elements less than (11.15), an equal number of multiplications are saved. Therefore, the total saving of multiplications in Algorithm E4 over Algorithm E1 is

$$\frac{n^2 + t^2}{2} + \text{lower-order terms.} \quad (11.35)$$

To simplify the description of the elimination algorithms (Step 2 for Algorithms E1 and E2) we assumed that the pivoted elements do not vanish. If a pivotal element is zero, we proceed as in Gaussian elimination, interchanging the appropriate rows; however, we extend this permutation to all subsequent rows modulo p . In practice, this is no problem, since the elimination algorithms for vector Toeplitz matrices suggest to arrange the coefficients $r_0, \dots, r_{pn}$ of Algorithm E1 in the form (compare with (11.12c)).

$$\begin{array}{cccccc} r_0 & r_p & r_{2p} & \dots & r_{np} & \\ 0 & r_{p-1} & r_{2p-1} & & 0 & \\ \vdots & \vdots & \vdots & & \vdots & \\ 0 & r_1 & r_{p+1} & \dots & 0 & \end{array}$$

and the coefficients $c_1, \dots, c_{(r+1)p+n}$ of Algorithm E4 in the form (compare with (11.33a))

$$\begin{array}{cccccc} c_1 & c_{p+1} & \dots & c_{(n-1)p+1} & c_{np+1} & \\ \vdots & \vdots & & \vdots & \vdots & \\ \vdots & \vdots & & \vdots & c_{np+p-h} & \\ \vdots & \vdots & & \vdots & 0 & \\ \vdots & \vdots & & \vdots & \vdots & \\ c_p & c_{2p} & \dots & c_{np} & 0 & \end{array}$$

for computational purposes. In these arrangements, a single interchange of two rows is all that is needed to obtain a nonzero pivot. Since our problem has a well-defined solution, nonzero pivotal elements can always be obtained.

APPENDIX A

A Proof for (9.4)

We note that the circulant C is the resultant (or eliminant) of $f(x) = a_0 + a_1x + \dots + a_{p-1}x^{p-1}$ and $g(x) = -1 + x^p$; viz.,

$$R(f, g) = \begin{vmatrix} \overbrace{a_0 \ a_1 \ a_2 \ \dots \ a_{p-1}}^p & \overbrace{0 \ \dots \ 0}^{p-1} \\ 0 & \overbrace{a_0 \ a_1 \ \dots \ a_{p-2} \ a_{p-1}}^p \dots 0 \\ \dots & \dots \\ 0 & 0 \ 0 \ \dots \ a_0 \ a_1 \ \dots \ a_{p-1} \\ -1 & \dots \ 0 \ 1 \\ \dots & \dots \\ -1 & \dots \\ \dots & \dots \\ -1 & \dots \\ \dots & \dots \\ 0 & 1 \end{vmatrix} = \begin{vmatrix} a_0 & \dots & a_{p-1} \\ a_{p-1} & a_0 & \dots & a_{p-2} \\ \dots & \dots & \dots \\ a_1 & a_2 & \dots & a_0 \end{vmatrix} = C.$$

Making the substitution $a_k \alpha_j^k \rightarrow a_k$ will result in $C = 0$, because then $f(\alpha_j x)$ has a zero $x = 1$. But any substitution $a_k \alpha_j^k \omega_p^{\ell k} \rightarrow a_k$ will also result in $C = 0$, since $f(\alpha_j \omega_p^\ell \cdot \omega_p^{p-\ell}) = 0$. Thus the circulant of $(a_0, a_1 x, \dots, a_{p-1} x^{p-1})$ as a function of x has the $p(p-1)$ zeros $\alpha_j \omega_p^\ell$, and can be represented as a polynomial of degree $p(p-1)$ in x . Let $x = 1$, and obtain (9.4).

This proof connects the circulant to the theory of resultants. This connection was already known to Baltzer (1864)¹⁹ (Vol. III, p. 374).

APPENDIX B

Another Derivation of (9.10)

Once the connection with the resultant is given, the following method of proof is obvious. Let

$$P_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n = 0, \quad (2.1)$$

and let

$$z = x^p. \quad (8.8)$$

Define

$$a_k(z) = \sum_{i=0}^{\left[\frac{n-k}{p} \right]} a_{pi+k} z^i. \quad (8.10)$$

Then, letting $a_k(z) = a_k$, we obtain

$$\begin{aligned} P_n(x) &= \sum_{k=0}^n a_k x^k = \sum_{k=0}^n a_k z^{k/p} \\ &= \sum_{k=0}^{p-1} (a_k + a_{k+p} z + a_{k+2p} z^2 + \dots) z^{k/p} \\ &= \sum_{k=0}^{p-1} a_k z^{k/p} = 0. \end{aligned}$$

To remove the fractional exponents, we use Cayley's method of rationalization, thus multiplying the above equation by $z^{\ell/p}$ ($\ell = 0, 1, \dots, p-1$):

$$z^{\ell/p} \sum_{k=0}^{p-1} a_k z^{k/p} = \sum_{k=0}^{p-1} a_k z^{(k+\ell)/p} = 0 \quad (\ell = 0, 1, \dots, p-1),$$

or explicitly, arranging the terms in order of increasing fractional exponents,

$$a_0 + a_1 z^{1/p} + a_2 z^{2/p} + \dots + a_{p-1} z^{(p-1)/p} = 0$$

$$a_{p-1} z + a_0 z^{1/p} + a_1 z^{2/p} + \dots + a_{p-2} z^{(p-1)/p} = 0$$

$$a_{p-2} z^2 + a_{p-2} z \cdot z^{1/p} + a_0 z^{2/p} + \dots + a_{p-3} z^{(p-1)/p} = 0$$

$$\dots \quad \dots \quad \dots \quad \dots$$

$$a_1 z + a_2 z \cdot z^{1/p} + a_3 z \cdot z^{2/p} + \dots + a_0 z^{(p-1)/p} = 0.$$

Thus,

$$P_n^{(p)}(z) = \begin{vmatrix} a_0 & a_1 & a_2 & \dots & a_{p-1} \\ a_{p-1}z & a_0 & a_1 & \dots & a_{p-2} \\ a_{p-2}z & a_{p-1}z & a_0 & \dots & a_{p-3} \\ \dots & \dots & \dots & \dots & \dots \\ a_1z & a_2z & a_3z & \dots & a_0 \end{vmatrix} = 0. \quad (\text{B1})$$

Weihrauch used this technique as long ago as 1880 for $p = 4$ on a polynomial of order 3, and noted that it can be used also for a polynomial of order 4 (Ref. 19), Vol. IV, p. 359).

Multiplying in the determinant of (B1) columns 1 to p by 1, x , ..., x^{p-1} , respectively, and dividing rows 1 to p by 1, x , ..., x^{p-1} , respectively, yields

$$P_n^{(p)}(z) = \begin{vmatrix} a_0 & a_1x & \dots & a_{p-1}x^{p-1} \\ \dots & \dots & \dots & \dots \\ a_1x & a_2x^2 & \dots & a_0 \end{vmatrix} = C(a_0, \dots, a_{p-1}x^{p-1}) \quad (9.10)$$

$$= C(A_0, \dots, A_{p-1}),$$

where A_k is defined by (9.6).

APPENDIX C

Remarks on the Expansion of the Circulant

The following theorem will greatly supplement our other theorems and corollaries on the expansion of the circulant:

Theorem 7. When

$$bA_0^{\lambda_0} A_1^{\lambda_1} \dots A_{p-1}^{\lambda_{p-1}}$$

is a term in the expansion of the circulant

$$C = C(A_0, A_1, \dots, A_{p-1}),$$

then

$$bA_0^{\lambda_0} A_{j_1}^{\lambda_1} \dots A_{j_{p-1}}^{\lambda_{p-1}}$$

is also a term, where j_ℓ ($\ell = 1, 2, \dots, p-1$) are integers less than p , such that

$$\ell j_1 \equiv j_\ell \pmod{p}, \quad (C1)$$

j_1 being relatively prime to p .

To prove this theorem, we must show only that

$$C(A_0, A_1, \dots, A_{p-1}) = C(A_0, A_{j_1}, \dots, A_{j_{p-1}}). \quad (C2)$$

To this end, we use (9.4) (known as Spottiswoode's theorem and derived in 1853; see Ref. 19, Vol. II, p. 405). As before, define

$$F(\omega) = A_0 + A_1\omega + A_2\omega^2 + \dots + A_{p-1}\omega^{p-1}. \quad (9.8)$$

Let ω be a primitive p th-root of unity. Then

$$C(A_0, A_1, \dots, A_{p-1}) = \prod_{k=0}^{p-1} F(\omega^k).$$

In particular,

$$\begin{aligned} F(\omega) &= A_0 + A_1\omega + \dots + A_\alpha\omega^\alpha + \dots, \\ F(\omega^\beta) &= A_0 + A_1\omega^\beta + \dots + A_\alpha\omega^{\alpha\beta} + \dots, \end{aligned} \quad (C3)$$

where α, β satisfy $1 \leq \alpha, \beta \leq p-1$, but are otherwise arbitrary integers. Because j_1 is assumed to be relatively prime to p , an ℓ exists such that $j_\ell = 1$. Therefore, using the definition (C1) of j_ℓ given in Theorem 7 yields

$$\left. \begin{aligned} \omega^{\ell j_1} &= \omega^{j_\ell} = \omega, \\ \omega^{\ell j_\alpha} &= \omega^{\alpha \ell j_1} = \omega^\alpha, \\ \omega^{\beta \ell j_1} &= \omega^\beta, \\ \omega^{\beta \ell j_\alpha} &= \omega^{\beta \alpha \ell j_1} = \omega^{\alpha \beta}. \end{aligned} \right\} \quad (C4)$$

Since $\ell j_1 \equiv 1 \pmod{p}$, ℓ is also relatively prime to p . But then for $\beta = 1, 2, \dots, p-1$, $\omega^{\beta \ell}$ will assume all values $\omega^1, \omega^2, \dots, \omega^{p-1}$, but not necessarily in this order. Now, we again write the linear factor $F(\omega)$, but emphasize different terms. In addition, we write the factors $F(\omega^\ell)$ and $F(\omega^{\beta \ell})$, where $\beta \ell$ is to be taken modulo p ,

$$\begin{aligned} F(\omega) &= A_0 + \dots + A_{j_1} \omega^{j_1} + \dots + A_{j_\alpha} \omega^{j_\alpha} + \dots, \\ F(\omega^\ell) &= A_0 + \dots + A_{j_1} \omega^{\ell j_1} + \dots + A_{j_\alpha} \omega^{\ell j_\alpha} + \dots, \\ F(\omega^{\beta \ell}) &= A_0 + \dots + A_{j_1} \omega^{\beta \ell j_1} + \dots + A_{j_\alpha} \omega^{\beta \ell j_\alpha} + \dots. \end{aligned}$$

But by (C4), the last two equations are

$$\begin{aligned} F(\omega^\ell) &= A_0 + \dots + A_{j_1} \omega + \dots + A_{j_\alpha} \omega^\alpha + \dots, \\ F(\omega^{\beta \ell}) &= A_0 + \dots + A_{j_1} \omega^\beta + \dots + A_{j_\alpha} \omega^{\alpha \beta} + \dots. \end{aligned} \quad (C5)$$

Thus, comparing (C5) with (C3), we see that we can rearrange the coefficients $A_0, A_1, \dots, A_\alpha, \dots, A_{p-1}$ according to the permutation

$$\begin{pmatrix} 0 & 1 & \dots & \alpha & \dots & p-1 \\ 0 & j_1 & \dots & j_\alpha & \dots & j_{p-1} \end{pmatrix}$$

without changing the value of the individual linear factors. But then (C2) follows. A property similar to that expressed in our theorem was known to Muir in 1884 (Ref. 19, Vol. IV, p. 372) for the special case in which $p = 7$. Its extension to, and formulation for, arbitrary p and partitioned polynomials seem to be new.

APPENDIX D

Method of Expanding Circulants on High-speed Computers

The determinant in (9.10) and (9.18) can be made symmetric by $(p-1)(p-2)/2$ permutations of the rows so that

$$C(A_0, A_1, \dots, A_{p-1}) = \begin{vmatrix} A_0 & A_1 & \dots & A_{p-1} \\ A_{p-1} & A_0 & \dots & A_{p-2} \\ \dots & \dots & \dots & \dots \\ A_1 & A_2 & \dots & A_0 \end{vmatrix} = (-1)^{\frac{(p-1)(p-2)}{2}} \begin{vmatrix} A_0 & A_1 & \dots & A_{p-1} \\ A_1 & A_2 & \dots & A_0 \\ \dots & \dots & \dots & \dots \\ A_{p-1} & A_0 & \dots & A_{p-2} \end{vmatrix} \quad (D1)$$

Except for the sign, the term on the right side is the determinant of a symmetric matrix.

Method: To obtain the expansion of this determinant, integer-preserving Gaussian elimination is used. The general theory is developed in Ref. 7; the following algorithm is an adoption of the single-step method to symmetric matrices. Let

$$D = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1p} \\ a_{21} & a_{22} & \dots & a_{2p} \\ \dots & \dots & \dots & \dots \\ a_{p1} & a_{p2} & \dots & a_{pp} \end{vmatrix}, \quad (D2)$$

where $a_{ij} = a_{ji}$ (symmetric matrix). Then the transformations are given by

$$\left. \begin{aligned} a_{00}^{(-1)} &= 1, \quad a_{ij}^{(0)} = a_{ij} \\ a_{ij}^{(k)} &= (a_{kk}^{(k-1)} a_{ij}^{(k-1)} - a_{kj}^{(k-1)} a_{ki}^{(k-1)}) / a_{k-1, k-1}^{(k-2)} \\ &\quad (i = k+1, \dots, p; j = i, i+1, \dots, p) \\ &\quad (k = 1, \dots, p-1). \end{aligned} \right\} \quad (D3)$$

It is important that the division is performed last, after all the algebraic forms have been ordered and similar terms collected.

After k transformations, the determinant $D = D^{(0)}$ has the form

$$D^{(k)} = \begin{array}{c} \begin{array}{ccccccc} a_{11}^{(0)} & a_{12}^{(0)} & \dots & a_{1,k+1}^{(0)} & \dots & a_{1p}^{(0)} \\ & a_{22}^{(1)} & \dots & a_{2,k+1}^{(1)} & \dots & a_{2p}^{(1)} \\ & & \dots & & \dots & \\ & & & \dots & & \\ & & & & \dots & \\ & & & & & \dots \end{array} \\ \begin{array}{c} \text{(zeros, not} \\ \text{computed)} \end{array} \end{array} \begin{array}{c} \begin{array}{ccc} \dots & a_{k+1,k+1}^{(k)} & \dots & a_{k+1,p}^{(k)} \\ & \dots & & \\ & & \dots & \\ & & & \dots & a_{ij}^{(k)} \\ & & & & \dots & \\ & & & & & \dots & a_{pp}^{(k)} \end{array} \\ \begin{array}{c} \text{(not} \\ \text{computed)} \end{array} \end{array} \end{array} \quad (D4)$$

If the matrix of C is substituted for the matrix of D , then the $a_{ij}^{(l)}$'s are homogeneous algebraic forms of grade $l + 1$ in $A_0, A_1, \dots, A_{p-1}$. In particular, $a_{pp}^{(p-1)}$ is the desired expansion of $C(A_0, \dots, A_{p-1})$.

Formally, the elements a_{ij} of D are given by

$$a_{ij} = A_{i+j-2-\epsilon p}, \quad (D5)$$

where

$$\epsilon = \begin{cases} 0 \\ 1 \end{cases} \text{ for } i + j - 2 \begin{cases} < p \\ \geq p \end{cases}.$$

Examples:

$$p = 1 \quad a_{11}^{(0)} = A_0;$$

$$p = 2 \quad a_{22}^{(1)} = A_0^2 - A_1^2;$$

$$p = 3 \quad a_{33}^{(3)} = A_0^3 + A_1^3 + A_2^3 - 3A_0A_1A_2.$$

The method described above was used to calculate the expansion of the circulants of order two to eight, and the results are tabulated in Table V. To determine all expansion terms for $p = 2$ to 7 required less than 1 min on the CDC-3600 computer. To compute the terms for $p = 8$, the fast-access memory (32K bits) was not sufficient, and disc storage had to be used. The time requirement increased to several hours. For this reason, no calculations were performed for $p > 8$.

Should a code be required for $p > 8$, it is recommended to use Theorem 7 of Appendix C and all appropriate corollaries of Section 9, and then apply the method of undetermined coefficients to reduce the problem to a system of linear equations. In addition, we can combine this method with the method of undetermined coefficients for the root-powering transformation as described in the example following Theorems 1 and 2 and leading to the solution of (7.16).

APPENDIX E

An Extension of the Circulant Method

Let $P_n(x) = A_0 + A_1 + \dots + A_{p-1}$ be a partitioned polynomial as in (9.7), and let

$$\begin{aligned} Q_n(x) &= n x P_n(x) - x^2 \frac{d}{dx} P_n(x) \\ &= B_0 + B_1 + \dots + B_{p-1}, \end{aligned} \quad (E1)$$

where the B_k 's are the partitions of $Q_n(x)$. Further define

$$P^0(x) = P_n(x), \quad Q^0(x) = Q_n(x),$$

$$P^{k+1}(x^p) = \begin{vmatrix} A_0^k & A_1^k & \dots & A_{p-1}^k \\ A_{p-1}^k & A_0^k & \dots & A_{p-2}^k \\ \dots & \dots & \dots & \dots \\ A_1^k & A_2^k & \dots & A_0^k \end{vmatrix}, \quad (E2)$$

$$Q^{k+1}(x^p) = \begin{vmatrix} B_0^k & B_1^k & \dots & B_{p-1}^k \\ A_{p-1}^k & A_0^k & \dots & A_{p-2}^k \\ \dots & \dots & \dots & \dots \\ A_1^k & A_2^k & \dots & A_0^k \end{vmatrix}, \quad (E3)$$

where the A_j^k 's are the partitions of $P^k(x) = A_0^k + \dots + A_{p-1}^k$ and the B_j^k 's are the partitions of $Q^k(x) = B_0^k + \dots + B_{p-1}^k$ for $k = 0, 1, 2, \dots$.

Let $u \neq 0$ be a simple root of $P^m(x) = 0$ ($m \geq 0$). Then there exists a simple root α for the equation $P_n(x) = 0$ such that $u = \alpha^{p^m}$. This root is given by

$$\alpha = - \left[\frac{Q^m(x)}{x \frac{dP^m(x)}{dx}} \right]_{x=u}. \quad (E4)$$

A proof for this interesting result is given by Fiedler.¹⁵

APPENDIX F

The Root-powering Transformation for Analytic Functions

Theorem 8. If an analytic function is expanded in a power series,

$$f(z) = \sum_{i=0}^{\infty} a_i z^i,$$

and

$$A_k = z^k \sum_{i=0}^{\infty} a_{k+pi} z^{pi} \quad (k = 0, 1, \dots, p-1)$$

converges absolutely for $|z| < \rho$, then the function $F(w)$ whose zeros are p th powers of the zeros of $f(z)$ in $|z| < \rho$, is given by

$$F(z^p) = C(A_0, \dots, A_{p-1})$$

where

$$|w| = |z^p| < \rho^p.$$

The proof is straightforward (because C is a determinant of finite order) and therefore is omitted here.

Thus the concept of partitioned polynomials can be expanded to partitioned analytic functions. Theorem 8 seems to be new. In particular, it gives a simple condition under which the Gräffe transformation and the root-cubing method can be applied to power series. Pólya²³ and Ostrowski²⁰ treated the case $p = 2$ directly.

As a simple example, we apply this theorem to demonstrate that e^z has no zeros. Define

$$\begin{aligned} e^z &= \cosh z + \sinh z \\ &= A_0 + A_1. \end{aligned}$$

Then we have

$$F(z^2) = A_0^2 - A_1^2 = \cosh^2 z - \sinh^2 z = 1$$

by geometric reasons. Similarly, defining

$$e^{iz} = \cos z + i \sin z = A_0 + A_1$$

leads to

$$F(z^2) = A_0^2 - A_1^2 = \cos^2 z + \sin^2 z = 1$$

by geometric reasons. Thus $F(w)$ has no zeros in the finite w -plane, implying e^z and e^{iz} have no zeros in the finite z -plane.

An equally simple example is

$$f(z) = \frac{1}{1-z} = 1 + z + z^2 + \dots, \quad (\text{F1})$$

which leads to $F(w) = 1$ and not to $F(z^2) = (1 - z^2)^{-1}$ as we would have expected. Here, we have the relationship $A_1 = zA_0$. Note, however, that truncating $f(z)$, say, for example, to

$$f(z) \doteq P_3(z) = 1 + z + z^2 + z^3,$$

also yields $A_1 = zA_0$, but $F(w) = P_3^{(2)}(z^2) = -1 - z^2 + z^4 + z^6$ turns out to be as expected. This last example is of interest because the unit circle contains the dense limiting set of the zeros of the truncated function of (F1), namely of the cyclotomic polynomial,

$$P_{n-1}(z) \equiv \frac{1-z^n}{1-z} = 1 + z + z^2 + \dots + z^{n-1}$$

for $n \rightarrow \infty$, and thus illustrates the necessity of the conditions in Theorem 8.

The following theorem on poles and zeros of a meromorphic function $f(z)$ is well known. If $f(z)$ is analytic and is not zero on a closed contour C , and has inside C zeros $\alpha_1, \alpha_2, \dots, \alpha_n$ and poles $\beta_1, \beta_2, \dots, \beta_m$, then

$$\frac{1}{2\pi i} \int_C z^p \frac{f'(z)}{f(z)} dz = \sum_{i=1}^n \alpha_i^p - \sum_{i=1}^m \beta_i^p \quad (p = 0, 1, 2, \dots), \quad (\text{F2})$$

where a zero or pole is repeated according to its multiplicity.

If $f(z)$ has no poles inside C , then the right side of (F2) reduces to s_p ($p = 1, 2, \dots$), the sum of the p th powers of all zeros as defined by (6.5). For $p = 0$, (F2) yields the number n of zeros inside C . Thus, knowing $s_1, s_2, \dots, s_n$, we can use either Newton's formulas (6.6) or (6.8) to calculate the coefficients of a polynomial, the zeros of which are exactly the zeros $\alpha_1, \alpha_2, \dots, \alpha_n$ of $f(z)$ inside C . Numerical experimentation with this method has been claimed successful.¹²

The above remarks also imply that it is theoretically possible to break a high-order polynomial down into factors of (arbitrary) lower order. Whether an ingenious implementation of this idea can be superior to already existing methods is an unsolved problem.

ACKNOWLEDGMENTS

This report is a slightly enlarged version of Ref. 6. I wish to thank J. K. S. McKay, for informing me that similar research is being done at Edinburgh; Professor A. N. Kostovskii¹⁸ of Lvov State University (USSR), who let me see his unpublished thesis containing related results; Professor M. Fiedler, Prague, for discussing some of his research; and Mrs. Nancy Clark, who prepared the code for Table V.

REFERENCES

1. A. C. Aitken, *Note on Solving Algebraic Equations by Root-cubing*, The Mathematical Gazette, Vol. XV (1931), pp. 490-491.
2. E. H. Bareiss and M. A. Fisherkeller, *Resultant Procedure, A Method for Finding the Zeros of Real Polynomials*, ANL-5997 (June 1959).
3. Erwin H. Bareiss, *Resultant Procedure and the Mechanization of the Graeffe Process*, J.A.C.M. 7 (1960), pp. 346-386.
4. Erwin H. Bareiss, "The Numerical Solution of Polynomial Equations and the Resultant Procedures," Chapter 10 in Anthony Ralston and Herbert S. Wilf, *Mathematical Methods for Digital Computers*, Vol. 2, John Wiley & Sons (1967).
5. Erwin H. Bareiss and Ronald Hamelink, *RSSR Routine, A Root-squaring and Subresultant Procedure for Finding Zeros of Real Polynomials*, ANL-6987 (Oct 1965).
6. Erwin H. Bareiss, *The Root-cubing Process and the General Root-powering Method for Finding the Zeros of Polynomials on Present-day and Future Generation Computers*, Argonne National Laboratory, Applied Mathematics Division, Tech. Memo No. 114 (Jan 1966).
7. Erwin H. Bareiss, *Multi-step Integer-preserving Gaussian Elimination*, ANL-7213 (May 1966).
8. Maxime Bôcher, *Introduction to Higher Algebra*, The Macmillan Company, New York (1954).
9. F. Faà di Bruno, *Einleitung in die Theorie der Binären Formen*, Leipzig, B. G. Teubner (1881).
10. T. Busk and B. Svejgaard, "Polynomial Equations," Chapter IV in Ch. Gram, Ed., *Selected Numerical Methods*, Regnecentralen, Copenhagen (1962), pp. 262-295.
11. Ernesto Cesàro, *Einleitung in die Infinitesimalrechnung*, B. G. Teubner, Leipzig, Berlin (1922).
12. L. M. Delves and J. N. Lyness, *A Numerical Method for Locating the Zeros of an Analytic Function*, to be published in Math. Comp. (1967).
13. J. F. Encke, *Allgemeine Auflösung der numerischen Gleichungen*, Journal für die Reine und Angewandte Mathematik (Crelle), Berlin (1841).
14. D. K. Faddeev and I. S. Sominskii, *Problems in Higher Algebra*, W. H. Freeman and Company, San Francisco and London (1965).
15. M. Fiedler, *O Zoběcněné Graeffeho Metodě a Její Modifikaci*, Čas. pro Pěst. Mat. sv. 88, č.2, str. 194-199 (1963).
16. A. N. Kostovskii, *Generalized Transformation Formulae of the Lobachevskii-Graeffe Method for the Numerical Determination of the Zeros of Entire and Holomorphic Functions*, Z. Vyscisl. Mat. i Mat. Fiz. 1 (1961), pp. 345-349 (in Russian).

17. A. N. Kostovskii, *The Determination of Strict Inequalities between the Moduli of Roots in the Transformation of Algebraic Equations by the Lobačevskii-Graeffe Method*, Soviet Mathematics, Vol. 3 (1962), pp. 1587-1589.
18. A. N. Kostovskii, DAN Ukrainian SSR, No. 1 (1966), 6-9.
19. Thomas Muir, *The Theory of Determinants in the Historical Order of Development*, four volumes bound as two, Vol. I: 1693 to 1841; Vol. II: 1841 to 1860; Vol. III: 1861 to 1880; Vol. IV: 1880 to 1900, Dover Publication, New York (1960).
20. Alexander Ostrowski, *Recherches sur la Méthode de Graeffe et les Zéros des Polynômes et des Series de Laurent*, Acta Math. 72 (1940), pp. 99-257.
21. Oskar Perron, *Algebra, I Die Grundlagen*, Walter de Gruyter & Co., Berlin (1951).
22. G. Pólya, *Über das Graeffesche Verfahren*, Zeitschrift für Mathematik und Physik 63 (1915), pp. 275-290.
23. G. Pólya, *Remarks on Power Series*, Acta Scientiarum Mathematicarum, Szeged 12 (1950), Part B, pp. 199-203.
24. G. Pólya, *Graeffe's Method for Eigenvalues*, Stanford University Tech. Rep. No. CS63 (1967).
25. George Salmon, *Vorlesungen zur Einführung in die Algebra der linearen Transformationen*, 3rd ed. (1876), German translation by Fiedler, Leipzig.
26. B. L. Van der Waerden, *Moderne Algebra*, Vol. 1, 2nd ed., Springer (1940), p. 86.
27. E. H. Bareiss, *Numerical Inversion of Finite Toeplitz Matrices and Vector Toeplitz Matrices*, ANL-7440.