Comp. & Maths. with Appls. Vol. 8, No. 5, pp. 379–387, 1982
Printed in Great Britain.

0097–4943/82/050379–09$03.00/0

STABILIZING BAIRSTOW'S METHOD

R. ALT and J. VIGNES
Université P et M Curie, 4 Place Jussieu, 75730 Paris Cedex 05, France

Communicated by J. L. Lions

(Received December 1981)

Abstract—Bairstow's method has to face with numerical errors due to the termination criterion of Raphson–Newton iterations and to successive polynomial divisions. Here, an optimal termination criterion is proposed allowing to stop the iterations as soon as a good computed solution is obtained. Moreover, a simple formula to check the validity of a root of a polynomial is given. It is then possible to eliminate the numerical errors in Bairstow's method. Numerical examples are presented.

1. INTRODUCTION

Bairstow's method is used in searching for the real or complex roots of a polynomial having real coefficients of any degree. The search consists of successive divisions of the initial polynomial by second-degree polynomials, for which it is easy to obtain the roots.

The main difficulty in this process thus lies in finding a second-degree polynomial exactly dividing a given polynomial. This is done by canceling the two coefficients of the remainder of the division, which is a first-degree polynomial, by means of Raphson–Newton's method.

More exactly, let $P_N(x)$ be and N-degree polynomial with the following real coefficients

$$P_N(x) = \sum_{i=0}^{N} a_i x^{N-i} \quad a_i \in \mathbf{R}\ i = 0, 1, \ldots, N \tag{1}$$

$P_N(x)$ is written in the form

$$P_N(x) = (x^2 - sx + p).P_{N-2}(x) + b_{N-1}(x - s) + b_N \tag{2}$$

where

$$P_{N-2}(x) = \sum_{i=0}^{N-2} b_i x^{N-2-i} \quad b_i \in \mathbf{R}\ i = 0, 1, 2, \ldots, N.$$

We search for the numbers s and p which satisfy the following 2-dimensional non linear system

$$\begin{cases} b_{N-1}(s, p) = 0 \\ b_N(s, p) = 0. \end{cases} \tag{3}$$

This system is normally solved by Raphson–Newton's method. Knowing the solution for this system thus gives a second-degree polynomial which exactly divides the initial polynomial $P_N(x)$ and hence by solving the second-degree polynomial, we obtain two roots for $P_N(x)$.

The process also gives the coefficients b_i of the polynomial P_{N-2} and it can be begun over again with the dividend until this dividend becomes a second- or first-degree polynomial. The method is described in [5].

By examining the algorithm thus described, we see that its implementation in a computer raises various problems, which we will attempt to solve.

2. PROBLEMS LINKED TO THE COMPUTER IMPLEMENTATION OF BAIRSTOW'S METHOD

2.1 *Choice of criterion to stop the iterative process*

The iterations in Raphson–Newton's method used for solving system (3) are generally stopped in an arbitrary way. The most conventional termination criterion, described in [1] consists in stopping the iterative process on iteration k as soon as inequality (4) is satisfied and then four additional iterations are performed.

$$\left|\frac{s_k - s_{k-1}}{s_{k-1}}\right| + \left|\frac{p_k - p_{k-1}}{p_{k-1}}\right| < 10^{-2}. \tag{4}$$

It is obvious that this termination criterion, based on the fact that the convergence of Raphson–Newton's method is generally quadratic, may often fail and this failure is particularly likely to occur when the polynomial has multiple roots. The iterations are then broken off before b_{N-1} and b_N reach zero.

2.2 *Validity of roots found*

Once s and p have been obtained, what criterion must be used to decide whether the roots of the polynomial x^2-$sx+p$ are effectively the best roots of the initial polynomial $P_N(x)$ which the computer is capable of providing and, if this is not the case, how can they be improved?

2.3 *Accuracy of the solution*

When the roots obtained are the best possible ones with regard to the computer, what is the accuracy obtained for these roots?

The solutions to these problems are based on the Permutation–Perturbation method proposed by La Porte and Vignes [4] and which is summed up hereunder.

3. PERMUTATION-PERTURBATION METHOD

This method is used for automatically analyzing the propagation of computing errors caused by the floating-point arithmetic of the computer for any algorithm providing results in a finite number of computers.

Therefore, for each result, it provides the accuracy, i.e. a number of exact significant decimal figures.

We will briefly sum up this method by taking, as an algorithm, an algebraic expression defined by

$$y = f(d, +, -, \times, :, \text{funct}) \tag{5}$$

in which $d \subset \mathbf{R}$ is the set of data, and $y \in \mathbf{R}$ is the result of the algebraic expression, $+, -, \times, :$, funct are the exact mathematical operators.

To perform algebraic expression (5) on a computer, it is transcribed into a programming language and we obtain

$$Y = F(D, \oplus, \ominus, *, /, \text{FUNCT}) \tag{6}$$

in which $D \subset \mathbf{F}$ is the set of data, $Y \in \mathbf{F}$ is the result of procedure (6); $\oplus, \ominus, *, /$, FUNCT, are data-processing operators.

$\mathbf{F}$ is the set of floating-point values that can be represented in the computer.

3.1 *The permutation method*

Since the rules of algebra, such as the associativity of addition, are not valid in floating-point arithmetic, there is not just a single data-processing image expression of f but a set $P = \{F_i\}$ of data-processing image expressions obtained by performing all possible combinations of permutable arithmetic operators in the algebraic expression. We obtain

$$\text{Card } P = C_{\text{op}}$$

in which C_{op} is the number of combinations corresponding to all possible permutations of the arithmetic operators.

3.2 *The perturbation method*

Let us consider one of the data-processing expressions F_i and have it performed by the computer. At each arithmetic operation since the result contains an error (chopping or rounding-off), any result of a data-processing operation (assignment, arithmetic operations) must be considered to have two possible results, one by lack and the other by excess.

Therefore, if the algebraic expressions F_i require k data-processing operations, each F_i will provide a set

$$R = \{Y_i / Y_i \in \mathbf{F}) \tag{7}$$

so that each Y_i thus legitimately represents the exact result y.

3.3 *The permutation-perturbation method*

By applying the Perturbation method to each data-processing image F_i of f, a set $\mathcal{R} = \{Y_j / Y_j \in \mathbf{F}\}$ is obtained in which:

$$\text{Card } \mathcal{R} = 2^k C_{op}. \tag{8}$$

Each Y_j thus legitimately represents the exact result y.

3.4 *Evaluating the accuracy of* Y_j

It has been shown [3] that subpopulation of only three elements, obtained by having the computer perform any data-processing procedure F_i three times, by perturbing the result of each arithmetic operation, can be used to determine the accuracy of the result, i.e. its number of exact significant decimal figures.

It has been shown [3] that the data-processing result which best represents y is the average $\bar{Y}$ of three data-processing results obtained Y_j, $j = 1, 2, 3$, and that the exact number of significant decimal figures for $\bar{Y}$ is given by C so that

$$C = \log_{10} \frac{|\bar{Y}|}{S} \tag{9}$$

S Being the standard deviation of the three results Y_j.

4. SOLUTIONS TO PROBLEMS RAISED

4.1 *Optimum termination criterion for Raphson–Newton's iterations*

4.1.1 *Need for a good termination criterion.* It is quite obvious that the accuracy of Bairstow's method depends directly on the obtaining of the polynomial $x^2 - sx + p$ dividing the initial polynomial. Therefore, the choice of the termination criterion for Raphson–Newton's iterations is determinant with regard to the propagation of errors from one division to the next one. For example, a search for the roots of the polynomial

$$P(x) = \sum_{i=0}^{51} \frac{x^i}{i!} - e^{-10}$$

with a CDC 7600 computer using the termination criterion (4), gives as the 51st root

$$x_{51} = -0.9994\ldots \times 10^1$$

whereas the root is -10 with 14 exact decimal figures. The root is thus obtained with only three exact decimal figures, whereas the computer is working in binary floating-point arithmetic with a 48 bits mantissa and can thus give 14 exact decimal figures. This example shows that the propagation of numerical errors may be considerable or even catastrophic. We consider that all the termination criteria used to date are unsatisfactory, and we propose an optimum termination criterion.

4.1.2 *Expression of optimum termination criterion.* The only adequate termination criterion for this type of iterative method is that which stops the process as soon as a satisfactory dataprocessing solution is found [4].

Let us consider the 2-dimensional non-linear system

$$\mathscr{F}(x) = 0 \text{ with components } f_i,\ i = 1,2$$

and let $F(X)$ be one of its data-processing images.

The iterative solution process must be stoped at iteration n as soon as each of the components of the residue

$$\rho^{(n)} = F(X^{(n)})$$

satisfies one of the following conditions

- either $\rho^{(n)}$ is zero (which almost never occurs because computing is done with a limited precision arithmetic),
- or it represents only the cumulative effect of the errors in the computing of the components $\rho_i^{(n)}$, $i = 1,2$.

The Permutation–Perturbation method of La Porte and Vignes described in Section 3 can be used to estimate the error resulting from the error propagation of computing and the exact number of significant figures of the result of an algebraic expression, its application to each of the equations in the non-linear system will enable us to estimate the number of exact significant figures $C_i^{(n)}$ in each of the components $\rho_i^{(n)}$, $i = 1, 2$ of the residual vector $\rho^{(n)}$.

Therefore, if one of the two $C_i^{(n)}$ is greater than or equal to 1, the solution has not been reached. But if both $C_i^{(n)}$ are less than 1, then it can be confirmed that the $\rho_i^{(n)}$ represent only the cumulative effects of the computing errors, and hence the two equations in the system are verified.

The result found can thus be considered to be a satisfactory data-processing solution for the system.

In practice, the optimum termination criterion is as follows:

At the end of each iteration p, the Permutation–Perturbation method is applied to each of the equations in the non-linear system, and the exact number of significant figures $C_i^{(p)}$, $i = 1, 2$, is evaluated for each residue $\rho_i^{(p)}$.

Either of two cases is possible (i) at least one of the $C_i^{(p)} \geq 1$, and so the iterative process must be continued because the solution has not yet been reached, or (ii) both $C_i^{(p)} < 1$, $i = 1, 2$, in which case both $\rho_i^{(p)}$ are non significant values, and the vector $X^{(p)}$ is a satisfactory data-processing solution for the system. The iterative process must then be broken off.

Comment

In practice, the termination of the process must also be considered if two successive iterations are equal from the data-processing standpoint, even if the $C_i^{(p)}$ are not less than 1. In this case the solution has not been reached, but the convergence of the process toward a more accurate solution is impossible.

4.1.3. *Numerical example.* Let us consider the following tenth-degree algebraic equation

$$x^{10} - 20x^9 + 175x^8 - 882x^6 + 2835x^6 - 6072x^5 + 8777x^4 - 8458x^3 + 5204x^2 - 1848x + 288 = 0.$$

This equation is proposed in [1] and has the following exact solutions

$$x_1 = x_2 = x_3 = x_4 = 1 \qquad x_5 = x_6 = x_7 = 2$$

$$x_8 = x_9 = 3 \qquad x_{10} = 4.$$

It has been solved with Bairstow's method on a CDC 7600 computer using the conventional termination criterion (4) followed by the optimum termination criterion. The results obtained are given in Tables 1 and 2.

It has seen that the use of the optimum termination criterion quite appreciably decreases the error propagation from one division to the next. This is due to the fact that at each division the non-linear system

$$\begin{cases} b_{N-1}(s, p) = 0 \\ b_N(s, p) = 0 \end{cases} \tag{12}$$

is solved with the best possible accuracy. Table 2 gives the number of iterations required for Raphson–Newton's method and the numerical values attained for b_{N-1} and b_N at the end of the iterations using the two preceding termination criteria; in both cases the initializations for s and p were $s_0 = p_0 = 0$.

This example clearly shows the advantage of the optimum termination criterion. When the conventional termination criterion is used the iterations are stopped, although the values of b_{N-1} and b_N are not yet zero.

The values are given in Table 2 with two exact significant figures. With the optimum termination criterion, the values of b_{N-1} and b_N, which are known with zero significant figures, must here be considered as exact mathematical zeros.

4.2 *Checking the validity of the root found*

To be certain that a value x_0 is effectively the root of a given polynomal $P_N(x)$, it is theoretically sufficient to check and see whether $P_N(x_0) = 0$. Unfortunately, when a computer

Table 1.

Division n°	Conventional Termination criterion		Optimum Termination criterion	
	Re(x)*	im(x)**	Re(x)	im(x)
1	0.995	0.335×10^{-2}	0.998	0.113×10^{-2}
	0.995	-0.335×10^{-2}	0.998	0.113×10^{-2}
2	1.006	0.732×10^{-2}	1.001	0.167×10^{-2}
	1.006	-0.732×10^{-2}	1.001	-0.167×10^{-2}
3	1.993	0.116×10^{-1}	1.998	0.258×10^{-2}
	1.993	-0.116×10^{-1}	1.998	-0.258×10^{-2}
4	2.013	0.	2.002	0.
	2.998	0.	2.99990	0.
Remainder	3.001	0.	3.00009	0.
	3.999997	0.	3.999999999	0.

* Re(x): signifies the real part of x.
** im(x): signifies the imaginary part of x.

Table 2.

Division	Conventional termination criterion			Optimum termination criterion		
	Niter	b_{N-1}	b_N	Niter	b_{N-1}	b_N
1	24	$0.11\ 10^{-11}$	$0.7\ 10^{-12}$	28	0.	0.
2	16	$0.10\ 10^{-10}$	$0.40\ 10^{-11}$	13	0.	0.
3	20	$0.54\ 10^{-9}$	$0.63\ 10^{-11}$	23	0.	0.
4	16	$0.26\ 10^{-8}$	0.	22	0.	0.

is used, even if x_0 is the exact root, chopping or rounding-off errors are such that $P_N(x_0)$ is not zero but is equal to a number ρ which we call the residue

$$P_N(x_0) = 0. \tag{13}$$

If ρ is non-significant, meaning that it represents only the cumulative error effect in the computing $P_N(x_0)$, it must be considered as equal to mathematical zero, and hence x_0 must be considered the root of $P_N(x)$.

If ρ is significant, $P_N(x_0) \neq 0$ and x_0 is thus not a solution for $P_N(x) = 0$.

Yet, with the Permutation–Perturbation method, it is possible to compute the number of exact decimal figures C in the residue ρ. If C is not less than 1, ρ is significant and x_0 is not a root. If C is less than 1, then ρ is not significant and must be considered as zero, and x_0 is thus a solution for $P_N(x) = 0$. This method is the most general can be used for any type of equation. But for a polynomial equation or for a linear system, it is possible to obtain a statistical expression for the theoretical value of ρ if the value found for x_0 is a root of $P_N(x) = 0$. This has been demonstrated in [2], and we set forth the results here.

If $P_N(x)$ is a polynomial with real coefficients, $P_N(x) = \sum_{i=0}^{N} a_i x^{N-i}$ and A_i the data-processing images of coefficients a_i

$$A_i \in \mathbf{F} \quad i = 0, 1, \ldots, N.$$

If x^* is an exact real root of the polynomial $P_N(x)$ and X^* is its data-processing representation, then Horner's rule can be used with a computer to calculate $P_N(X^*)$, and will obtain a residue of the following quadratic mean value

$$\hat{\rho} = 2^{-m} \sqrt{\left[\left(X^* \frac{\mathrm{d}P^*}{\mathrm{d}X}\right)^2 + N \sum_{i=0}^{N} (A_i X^{*N-i})^2\right]} \tag{14}$$

m being the number of bits in the mantissa of the numbers expressed in normalized floating-point arithmetic.

If x^* is a complex root of polynomial $P_N(x)$ and X^* is its data-processing representation, then the computation of $P_N(X^*)$ using Horner's rule gives a complex residue for which the values of the real and imaginary parts are respecitvely noted by $\hat{\rho}_r$ and $\hat{\rho}_i$. Theses values are defined by

$$\hat{\rho}_r = 2^{-m} \sqrt{\left[\left\{\mathcal{R}(X^*) \times \mathcal{R}\left(\frac{\mathrm{d}P^*}{\mathrm{d}X}\right)\right\}^2 + \left\{\mathcal{T}(X^*) + \mathcal{T}\left(\frac{\mathrm{d}P^*}{\mathrm{d}X}\right)\right\}^2 + N \sum_{i=0}^{N} \mathcal{R}(A_i X^{*N-i})^2\right]}$$

$$\hat{\rho}_i = 2^{-m} \sqrt{\left[\{\mathcal{R}(X^*) \times \mathcal{T}\left(\frac{\mathrm{d}P^*}{\mathrm{d}X}\right)\right\}^2 + \left\{\mathcal{T}(X^*) \times \mathcal{T}\left(\frac{\mathrm{d}P^*}{\mathrm{d}X}\right)\right\}^2 + N \sum_{i=0}^{N} \mathcal{T}(A_i X^{*N-i})^2\right]} \tag{15}$$

$\mathcal{R}(u)$ signifies the real part of u; $\mathcal{T}(u)$ signifies the imaginary part of u; m being the number of bits in the mantissa.

For a real or complex root X obtained by Bairstow's method, we must compute the real or complex standardized residues defined by

$$\rho^* = \frac{|P(X)|}{\hat{\rho}_N}, \quad \rho_r^* = \frac{|\mathcal{R}(P(X))|}{\hat{\rho}_r}, \quad \hat{\rho}_i^* = \frac{|\mathcal{T}(P(X))|}{\hat{\rho}_i}$$

$$\mathcal{R}(P(X)) = \sum_{i=0}^{N} \mathcal{R}(A_i * X^{N-i}) \tag{16}$$

$$\mathcal{T}(P(X)) = \sum_{i=0}^{N} \mathcal{T}(A_i * X^{N-i}).$$

If the root is real, either ρ^* is close to or less than 1, or it is greater than 1. In the former case, ρ is merely the result of cumulative errors in the computing of $P(X)$ and X must then be considered as a satisfactory data-processing root of $P_N(x)$. In the latter, X is not a root of the polynomial $P_N(x)$.

If the root is complex, either ρ_r^* and ρ_i^* are both close to or less than 1, or ρ_r^* (and/or) ρ_i^* is higher than 1. If the former is true, X must be considered as satisfactory data-processing root of $P_N(x)$. If the latter is true, X is not a root of $P_N(x)$.

4.3 *Evaluation and eventual improvement of accuracy*

When the Raphson–Newton's method is applied to system (3) to give two numbers s and p such that $x^2 - sx + p$ exactly divides the polynomial P, it is nonetheless possible that the roots obtained will not satisfy the criterion of the standardized residue with regard to $P_N(x)$. It is then possible to improve them, e.g. by Newton's method in **R** or in **C** until the standardized residues are less than 1. Indeed, this very simple criterion in the case of a polynomial equation is a specific form of the optimum termination criterion described above. It can thus be used to break off the iterations in Newton's method when a satisfactory data-processing solution has been obtained for equation $P_N(x) = 0$. When these two roots have been obtained, they then provide us with two new values for their sum s and their product p and thus of coefficients b_i for the dividend which will be more nearly accurate than values given directly by solving system (3). Thus at each kth division,we obtain the best numbers s_k and p_k and the best polynomial P_{k-2} from the data-processing standpoint as would be given by

$$P_k(x) = (x^2 - s_k x + p_k).P_{k-2}(x). \tag{17}$$

The propagation of numerical errors from one division to another is thus greatly reduced, almost to the point of elimination.

There is a second advantage in using Newton's method to improve a root in **R** or in **C**. It lends itself very well to the implementation of permutations and perturbations for computing the number of significant figures in the solutions obtained. By means of the algorithm that we have developed, at each division we obtain the best possible roots from the data-processing standpoint. Equally important, we know the accuracy with which these roots have been computed. Computing the number of significant figures in the solution then enables us to decide whether a root for which the imaginary part was found to be small is a true complex root, or a real root, the imaginary value of which is due to computing errors.

Where z is a root computed with our algorithm, and C_r and C_i the numbers of significant figures found for the real value $Re(z)$ and the imaginary value $\mathrm{Im}(z)$, either

(i) C_i is greater than 1 (in which case we can affirm that z is effectively complex), or

(ii) C_i is less than 1. Here the value found for $\mathrm{Im}(z)$ is non-significant, and 0 is a possible value. The root that is found may be real, in which case it suffices to compute the numerical value of $P_N(z)$ with $z = (\mathrm{Re}(z), 0)$ and then to compute the standardized residue of $P_N(z)$.

Let $\rho^*(z)$ be this standardized residue. If $\rho^*(z)$ is less than 1, then $s = (\mathrm{Re}(z), 0)$ is a root of $P_N(z)$. Hence this root is effectively real. But if $\rho^*(z)$ is greater than 1, it can be said that $(\mathrm{Re}(z), 0)$ is not a solution. The root $z = (\mathrm{Re}(z), \mathrm{Im}(z))$ is thus effectively complex, and its imaginary part is not zero but is non-significant.

Example 1

Table 3 gives the rounded-off real and imaginary values found for the roots of the tenth-degree equation mentioned above as well as the number of exact decimal significant figures.

Examination of this table and comparison with Table 1, clearly show that our algorithm can be used to determine whether a root is real or complex.

For the equation $\sum_{i=0}^{i=51} x^i/i! - e^{-10} = 0$ the solution $x = -10$ is found by the method described above with 7 exact significant decimal figures.

Example 2

Let us consider the following algebraic equation

$$a_1x^6 + a_2x^3 + a_3x + a_4 = 0 \tag{18}$$

Table 3.

Root N°	Re(x)	C_r	Im(x)	C_i
10	1.00	3	0.	14
9	1.00	3	0.	14
8	1.0	2	0.	14
7	1.00	3	0.	14
6	2.000	4	0.	14
5	2.00	3	0.	14
3	2.000	4	0.	14
4	3.000	5	0.	14
1	3.0000	5	0.	14
2	4.000000000	10	0.	14

in which

$$a_1 = 0.170522639876489\ 10^{+18}$$
$$a_2 = -0.180900824489071\ 10^{-4}$$
$$a_3 = -0.101133169642264\ 10^{-20}$$
$$a_4 = -0.383826436411105\ 10^{-26}.$$

This equation has been solved on a CDC 7600 computer with Bairstow's method using (i) the conventional termination criterion and (ii) the optimum termination criterion.

The results obtained are given in Table 4.

The results show very clearly that Bairstow's method using the conventional termination criterion gives false results because the iterations are broken off before attaining $b_N = b_{N-1} = 0$.

On the other hand, the results obtained with Bairstow's method using the optimum termination criterion give exact results to 14 decimal significant figures.

5. CONCLUSION

The Permutation–Perturbation method applied here to Raphson–Newton's method has made it is possible to determine the best possible second-degree polynomials which divide $P_N(x)$, almost

Table 4.

	Results obtained by using the conventional termination criterion	Results obtained by using the optimum termination criterion	C
X_1	$-0.47269433382398\ 10^{-7}$	$-0.47207606833448\ 10^{-7}$	14
X_2	$0.61009670860586\ 10^{-7}$	$0.597473339666347\ 10^{-7}$	14
X_3	$0.22924610660685\ 10^{-7}$ $-i0.48533909011179\ 10^{-7}$	$0.23604348981133\ 10^{-7}$ $-i0.41110574719183\ 10^{-7}$	14 14
X_4	$0.22924610660685\ 10^{-7}$ $+i0.48533909011179\ 10^{-7}$	$0.23604348981133\ 10^{-7}$ $+i0.41110574719183\ 10^{-7}$	14 14
X_5	$-0.33917452460500\ 10^{-7}$ $-i0.49366319284117\ 10^{-7}$	$-0.29874212547582\ 10^{-7}$ $-i0.51562637073113\ 10^{-7}$	14 14
X_6	$-0.33917452460500\ 10^{-7}$ $+i0.49366319284117\ 10^{-7}$	$-0.29874212547582\ 10^{-7}$ $+i0.51562637073113\ 10^{-7}$	14 14

entirely eliminating the propagation of errors due to successive polynomial divisions. Application of the method to Bairstow's algorithm made it possible to determine the number of exact significant figures in the solutions. In short, we can say that the method described here serves to stabilize Bairstow's method and to check the accuracy of the solution.

REFERENCES

1. E. Durand, *Solutions Numériques des Equations Algégriques*, Tome 1. Masson, Paris (1960).
2. M. La Porte and J. Vignes, Etude statistique des erreurs dans l'arithmétique des ordinateurs. Application au contrôle des résultats d'algorithmes numériques. *Numer. Math.* **23**, 63–72 (1974).
3. M. Maille, Méthodes d'évaluation de la précision d'une mesure ou d'un calcul numérique. LITP Report, Institut de Programmation, Université P. et M. Curie, Paris (1979).
4. J. Vignes, New methods for evaluating the validity of the results of mathematical computations. *Math. Comput.* Simul. **XX**, 227–249 (1978).
5. J. Vignes, M. Pichat and R. Alt, *Algorithmes Numériques, Analyse et Mise en Oeuvre.* Editions Technip, Paris (1980).
6. J. Vignes and M. La Porte, *Error Analysis in Computing*, pp. 610–614. Proce of IFIP Congress, Stockholm, (Aug. 1974).