Intern. J. Computer Math., 1983, Vol. 14, pp. 59–71
0020-7160/83/1401-0059 $18.50/0
© Gordon and Breach Science Publishers Inc., 1983
Printed in Great Britain

# Polynomial Real Root Approximation Using Continued Fractions

ALKIVIADIS G. AKRITAS

*University of Kansas, Department of Computer Science, Lawrence, Kansas 66045, U.S.A.*

and

KING N. NG

*Shell Oil Company, P.O. Box 991, Houston, Texas 77001, U.S.A.*

(*Received June 1982; in final form December 1982*)

A method with polynomial computing time bound is presented for the approximation of real roots of polynomial equations using continued fractions; it is based on an idea by Lagrange [10] and Vincent's theorem [17], and it has been implemented using exact (infinite precision) integer arithmetic algorithms. A theoretical analysis of the computing time of this method is given along with some empirical results.

KEY WORDS: Analysis of (exact) algorithms, polynomial real root isolation and approximation, Cauchy's rule, Lagrange's method, Vincent's theorem.

C.R. CATEGORIES: I1, G.1.5.

## 1. INTRODUCTION

Recently, in the light of new discoveries, the process of isolating the real roots of a polynomial equation (or, simply, isolation) has been extensively studied [6, 2, 4]. Isolation is the process of finding real, disjoint intervals such that each contains exactly one real root and every real root is contained in some interval. According to Fourier this is only the first (or the two) step involved in the computation of the real roots of a polynomial equation; the second step consists of approximating these roots to any desired degree of accuracy $\varepsilon$, that is, making the length of the isolating intervals less than or equal to $\varepsilon$.

In this paper we shall restrict our attention to those approximation methods which have been implemented in computer algebra systems using exact (infinite precision) integer arithmetic algorithms. (For a survey of computer algebra systems see [13]). When we use exact integer arithmetic then, in analyzing an algorithm, the "cost" of an operation on two integers depends not only on the operation itself but on the length (number of bits) of the operands as well. If $A$ is an integer, we define $L(A)$, its length, by

$$L(A) = \begin{cases} 1 & A = 0 \\ \lfloor \log_b |A| \rfloor + 1 & A \neq 0 \end{cases}$$

where $b$ indicates the base of the number system in which the operand $A$ is represented when an operation is performed. If $P$ is a polynomial with integer coefficients, $|P|_\infty$ represents the maximum coefficient in absolute value.

Bisection is basically the only approximation method implemented in computer algebra systems. For a square-free polynomial $P$ and an isolating interval of the form $(a, b]$ bisection proceeds as follows [9]: Evaluate the sign of $P$ at $b$. If it is zero replace $(a, b]$ by $[b, b]$ and terminate, otherwise evaluate the sign of $P$ at $(a+b)/2$. If this sign is zero, replace $(a, b]$ by $[(a+b)/2, (a+b)/2]$ and terminate; if it has the same sign as $P$ does at $b$, then obviously the root is in the interval $(a, (a+b)/2]$, otherwise it is in $((a+b)/2, b]$. This process is repeated until the length of the current interval is less than or equal to $\varepsilon$. It has been shown [9] that the bisection method will isolate one real root in time

$$O(n^2(L(h/\varepsilon))(L(eh/\varepsilon))(L(eh|P|_\infty/\varepsilon))) \tag{1.1}$$

where $n$ is the degree of the square-free polynomial $P$, $h$ is the initial length of the interval, $\varepsilon$ is the degree of accuracy (limit of approximation) and $e = \max\{|a_1|, |a_2|, |b_1|, |b_2|\}$ where $(a_1/b_1, a_2/b_2]$ is the initial interval, and $a_1$, $a_2$, $b_1$, $b_2$ are integers. Empirical results showed that bisection is a very slow method: its performance was later improved when it was combined with Newton's method [16].

Quite recently, extending previous work by the first author [2, 4] the second author developed a method (actually, three versions of it) with polynomial computing time bound for the approximation of real roots using continued fractions [11]. This approach is based on an idea by Lagrange (1767) and Vincent's theorem of 1836 [10, 17]; as we will see, all versions of our method will isolate one real root in time

$$0\left(L\left(\frac{1}{\varepsilon}\right)(n^3(L(|P|_\infty))^3)\right). \tag{1.2}$$

From (1.2) it is obvious that, unlike bisection, this method does not depend on the length of the initial isolating interval. In the sequel we will study this method in detail. (It should be noted that the ratios $h/\varepsilon$, in (1.1) and $1/\varepsilon$, in (1.2), are integers.)

## 2. MATHEMATICAL BACKGROUND

The idea to approximate the real roots of a polynomial equation using continued fractions is due to Lagrange (1767) [10], whose objective was to develop a procedure free of the defects plaguing the well-known Newton's method of approximation. Lagrange's idea may be stated as follows (see also ([14], pp. 135–141) and ([8], p. 223)): Suppose a root of the polynomial equation $P(x)=0$ lies between the consecutive integers $a_1$ and $a_1+1$; diminish the roots of the equation by $a_1$ (i.e. $x \leftarrow x+a_1$) and take the reciprocal equation (i.e. $x \leftarrow 1/x$). Find, *by trial*, a root of the last equation lying between $a_2$ and $a_2+1$, diminish the roots by $a_2$ and take the reciprocal equation. Proceed in this way. Then the continued fraction

$$a_1+\cfrac{1}{a_2+\cfrac{1}{a_3+\ddots}}$$

approximates a root of the equation.

Clearly, Lagrange's idea has certain drawbacks. Notice that the partial quotients $a_i$ are computed by trial, which means, that this computation is exponential in the length of the $a_i$'s. Moreover, it should be observed that the procedure is straightforward if there is one, and only one, root between the consecutive integers $a_i$ and

$a_i+1$. However, there was no proof, at the time, that if there are two or more roots within $(a_i, a_i+1)$, the process will eventually separate them. This fact was proven in 1836 by the following:

THEOREM 2.1. *Let $P(x)=0$ be a polynomial equation of degree $n>1$, with rational coefficients and without multiple roots, and let $\Delta>0$ be the smallest distance between any two of its roots. Let $m$ be the smallest index such that*

$$F_{m-1}\frac{\Delta}{2}>1 \quad \text{and} \quad F_{m-1}F_m\Delta>1+\frac{1}{\varepsilon_n} \tag{2.1}$$

*where $F_k$ is the kth member of the Fibonacci sequence 1, 1, 2, 3, 5, 8, 13,... and*

$$\varepsilon_n=\left(1+\frac{1}{n}\right)^{\frac{1}{n-1}}-1.$$

*Then the transformation*

$$x=a_1+\cfrac{1}{a_2+\cfrac{1}{a_3+\cdots+\cfrac{1}{a_m+\cfrac{1}{y}}}} \tag{2.2}$$

*(which is equivalent to the series of successive transformations of the form $x=a_i+(1/y)$, $i=1,2,\ldots,m$) with arbitrary, positive, integral elements $a_1,a_2,\ldots,a_m$, transforms the equation $P(x)=0$ into the equation $\tilde{P}(y)=0$, which has not more than one sign variation in the sequence of its coefficients.*

The proof of the above theorem is very long, and it is omitted since it can be found in the literature [1, 15 pp. 298–304]. The original form of Theorem 2.1 (that is, without specifying the quantity $m$) is due to Vincent alone [17, 6] and appeared in 1836; Uspensky [15] extended it in a somewhat erroneous manner, which was corrected in [1].

Theorem 2.1 can be used to isolate the real roots of a polynomial equation; from its statement we know that a transformation of the form (2.2) with arbitrary, positive integer elements $a_1,a_2,\ldots,a_m$ transforms $P(x)=0$ into an equation $\tilde{P}(y)=0$ which has at most one sign variation. This transformation can also be written as

$$x = \frac{P_m y + P_{m-1}}{Q_m y + Q_{m-1}} \tag{2.3}$$

where $P_k/Q_k$ is the $k$th convergent to the continued fraction

$$a_1 + \cfrac{1}{a_2 + \ddots}$$

and as we recall

$$\begin{aligned} P_{k+1} &= a_{k+1} P_k + P_{k-1} \\ Q_{k+1} &= a_{k+1} Q_k + Q_{k-1} \end{aligned} \tag{2.4}$$

with

$$P_k Q_{k-1} - P_{k-1} Q_k = (-1)^k \tag{2.5}$$

and $k \geqq 0$, $P_0 = 1$, $P_{-1} = 0$, $Q_0 = 0$, $Q_{-1} = 1$.

Since the elements $a_1, a_2, \ldots, a_m$ are arbitrary, there is obviously an infinite number of transformations of the form (2.2). However, with the help of Budan's theorem [3] we can easily determine those that are of interest to us; namely, there is a finite number of them (equal to the number of the positive roots of $P(x)=0$) which lead to an equation with exactly one sign variation in the sequence of its coefficients. Suppose $\tilde{P}(y)=0$ is one of those equations; then from the Cardano–Descartes rule of signs we know that it has one root in the interval $(0, \infty)$; If $\hat{y}$ was this positive root then the corresponding root $\hat{x}$ of $P(x)=0$ could be easily obtained from (2.3). We only know though that $\hat{y}$ lies in the interval $(0, \infty)$; therefore, substituting $y$ in (2.3) once by 0 and once by $\infty$ we obtain for the positive root $\hat{x}$ its isolating interval whose unordered endpoints are $P_{m-1}/Q_{m-1}$ and $P_m/Q_m$. In this fashion we can isolate all the positive roots of $P(x)=0$. If we subsequently replace $x$ by $-x$ in the original equation, the negative roots become positive and, hence, they too can be isolated in the way mentioned above.

The calculation of the quantities $a_1, a_2, \ldots, a_m$ for the transformations of the form (2.2)—which lead to an equation with exactly one sign variation—constitutes the polynomial real root isolation procedure. There are two methods, Vincent's and the one due to the first author, corresponding to the two different ways in which the computation of the $a_i$'s may be performed.

Vincent's method basically consists of computing a particular $a_i$ by a series of unit incrementations (by trial); that is $a_i \leftarrow a_i + 1$, which corresponds to the substitution $x \leftarrow x+1$. This brute force approach results in a method which will behave exponentially when the values of the $a_i$'s are big. Examples of this approach can be found in [17] and in [15].

On the contrary, the method due to the first author consists of immediately computing a particular $a_i$ as the lower bound on the values of the positive roots of a polynomial; that is $a_i \leftarrow b$, which corresponds to the substitution $x \leftarrow x+b$ performed on the particular polynomial under consideration. This method is obviously independent of how big the values of the $a_i$'s are. An unsuccessful treatment of the big values of the $a_i$'s can be found in ([15], p. 136). (We can safely conclude that $b = \lfloor \alpha_s \rfloor$, where $\alpha_s$ is the smallest positive root.) The lower bound $b$ on the values of the positive roots is computed with the help of the following rule ([12], pp. 50–51); notice that we are computing the upper bound on the values of the positive roots of $P(1/x) = 0$.

THEOREM 2.2 (Cauchy's Rule). *Let $P(x) = x^n + c_{n-1}x^{n-1} + \ldots + c_1 x + c_0 = 0$ be an integral-coefficient, monic polynomial equation of positive degree n, and let $\lambda$ be the number of its negative coefficients. Then*

$$b = \max_{\substack{1 \leq k \leq n \\ c_{n-k} < 0}} |\lambda c_{n-k}|^{1/k}$$

*is an upper bound on the values of the positive roots of $P(x) = 0$.*

Cauchy's rule has been implemented using exact integer arithmetic and it has been shown that its computing time is [5]

$$O(n^2 L(|P|_\infty)). \tag{2.6}$$

Pursuing studies in the direction outlined above, it was observed that Theorem 2.1 can be also used to approximate the real roots to any desired degree of accuracy. This is easily achieved by extending (computing more partial quotients of) the continued fraction (2.2) which transforms the original polynomial equation into one with exactly one sign variation in the sequence of its coefficients. Notice that now the approximation method depends heavily on the isolation process; that is, it cannot work if it is provided only with the isolating intervals of the roots.

Suppose that the limit of approximation is $\varepsilon$, and that we have computed $k$ partial quotients. Then, from the preceding discussion it becomes obvious that the root lies between the consecutive convergents

$$\frac{P_{k-1}}{Q_{k-1}}, \frac{P_k}{Q_k}$$

(obtained from (2.4)) whose difference in absolute value is

$$\frac{1}{Q_{k-1}Q_k}.$$

(Use (2.5) to obtain the difference.) Hence if $x$ is the root we are approximating, we have

$$\left|\frac{P_k}{Q_k} - x\right| \leqq \frac{1}{Q_{k-1}Q_k} \leqq \frac{1}{Q_{k-1}^2}$$

and the method will terminate when

$$\frac{1}{Q_l^2} \leqq \varepsilon \tag{2.7}$$

for some $l$.

Before we give an algorithmic description of this method we would like to elaborate on the computation of the lower bound $b$ on the value of a positive root, during the approximation process. Clearly, Cauchy's rule can be used. However, this rule, in general, is somewhat inefficient due to the fact that several applications of it are needed in order to compute the value of $b$ (a particular partial quotient $a_i$). While during isolation Cauchy's rule is the only possible process, this is no longer true during approximation. That is, during approximation we can take advantage of the special nature of the polynomials whose lower bounds $b$ we are computing. These polynomials are special in the sense that they have one, and only one, positive root and hence one sign variation. Having made this observation we can now use the following theorem, which is a modern version of the one found in ([8], pp. 164–165) and ([14], p. 58).

THEOREM 2.3. *Let* $P(x)=c_n x^n+c_{n-1}x^{n-1}+\ldots+c_0=0$ *be an integral-coefficient polynomial equation of degree n. Then an upper bound on the values of its positive roots is given by*

$$b=\max_{\substack{0\le r\le n\\ c_r<0}}\left(\frac{|c_r|}{\sum\limits_{\substack{i=r+1\\ c_i>0}}^{n} c_i}\right)+1 \tag{2.8}$$

COROLLARY 2.1. *Let* $P(x)=c_n x^n+\ldots+c_{r+1}x^{r+1}-c_r x^r-\ldots-c_0=0$ *be an integral-coefficient polynomial equation of degree n, with only one sign variation in the sequence of its coefficients. Then an upper bound on its (only one) positive root is given by*

$$b=\frac{\max\limits_{0\le j\le r}(|c_j|)}{\sum\limits_{i=r+1}^{n} c_i}+1. \tag{2.9}$$

(*It is assumed that the* $c_i$*'s/*$i=0(1)n$ *are non-negative numbers.*)

Compared with Cauchy's rule, (2.9) is much simpler to implement. Its computing time bound is ([11], p. 39)

$$O(nL(|P|_\infty)+(L(|P|_\infty))^2). \tag{2.10}$$

In the approximation method we used a modified version of (2.9) and found it to take as much time as Cauchy's rule ([11], p. 16). Hence, in order to improve in efficiency, we compute $b$'s (the partial quotients $a_i$) in yet another way. That is, we use Corollary 2.1 to obtain an upper bound $\bar{b}$ on the positive root and then bisect the interval $(0, \bar{b})$ to find $b$. This is feasible because, now, there is only one positive root; the computing time of this approach is [11]

$$O(n^2(L(|P|_\infty))^3). \tag{2.11}$$

## 3. THE APPROXIMATION METHOD

In what follows we give an algorithmic description of the approximation method along with an analysis of its computing time.

*Description.* Let $\tilde{P}_c(x)=0$ be an integral-coefficient polynomial equation of degree $n$ with one sign variation in the sequence of its coefficients. $\tilde{P}_c(x)=0$ is obtained from an original equation $P(x)=0$ after a continued fraction transformation of the form (2.2). Let $P_c/Q_c$ be the convergent to the continued fraction from which $\tilde{P}_c(x)=0$ is derived and $P_0/Q_0$ the immediately preceeding one. From the previous discussion it is clear that $\tilde{P}_c(x)=0$ isolates one real root of $P(x)=0$, and we are going to approximate this root to within $\varepsilon$. Obviously, if $1/Q_0^2 \leqq \varepsilon$ then we have nothing to do (see also (2.7)).

*Step 1.* Compute the lower bound $b$ on the value of the positive root of $\tilde{P}_c(x)=0$. (It is assumed that $b=\lfloor\alpha\rfloor$, where $\alpha$ is the positive root; $b=a_i$ for some $i$).

*Step 2.* Obtain $\tilde{P}_{new}(x)=\tilde{P}_c(x+b)=0$ and the new convergent $P_{new}/Q_{new}$ using (2.4) and the pair $P_c/Q_c$, $P_0/Q_0$.

*Step 3.* If $\tilde{P}_{new}(0)=0$ we have computed the root exactly and we terminate.

*Step 4.* If $1/Q_c^2 \leqq \varepsilon$ we return an interval whose endpoints, $P_{new}/Q_{new}$ and $P_c/Q_c$, approximate the root to within the specified degree of accuracy $\varepsilon$, and we terminate.

*Step 5.* Obtain $\tilde{P}'_{new}(x)=\tilde{P}_{new}(1/x)=0$, update $\tilde{P}_c(x)$, $P_0/Q_0$, $P_c/Q_c$ by $\tilde{P}'_{new}(x)$, $P_c/Q_c$, $P_{new}/Q_{new}$ respectively, and go back to Step 1.

THEOREM 3.1. *Let $\tilde{P}_c(x)=0$ be an integral-coefficient polynomial equation of degree n with one sign variation in the sequence of its coefficients (and hence with one positive root) such that $\tilde{P}_c(0)\neq 0$. If $\varepsilon$ is the required degree of accuracy, then the method described above will isolate the positive root of $\tilde{P}_c(x)=0$ in time*

$$0\left(L\left(\frac{1}{\varepsilon}\right)(n^3(L(|\tilde{P}_c|_\infty))^3)\right)$$

*(where $1/\varepsilon$ is integer).*

*Proof.* From [7] we know that the transformation $\tilde{P}_{new}(x)=\tilde{P}_c(x+b)=0$ is executed in time $0(n^3(L(b))^2+n^2L(b)L(|\tilde{P}_c|_\infty))$. Combining the last formula with the fact that $b=0(|\tilde{P}_c|_\infty)$ (see [4], p. 59) we obtain $0(n^3(L(|\tilde{P}_c|_\infty))^2)$ which is bounded by

$$0(n^3(L(|\tilde{P}_c|_\infty))^3). \tag{3.1}$$

Comparing (2.6), (2.10), and (2.11) with (3.1) we see that (3.1) dominates the computing times of all the steps (for one iteration of our method.)

To find $i$, the number of iterations of our method needed to approximate the root to within $\varepsilon$ we use (2.7), i.e. $1/Q_i^2 \leqq \varepsilon$, and the following facts: (a) $Q_i \geqq F_i$, where $F_i$ is the $i$th member of the Fibonacci sequence (see also (2.4)), and (b) $F_i=\phi^i/\sqrt{5}$ (rounded to the nearest integer), where $\phi=1.618\ldots$. Obviously, (2.7) yields

$$\frac{5}{\phi^{2i}} \leqq \varepsilon$$

from which we obtain

$$i=0\left(\log_\phi \frac{1}{\varepsilon}\right)=0\left(L\left(\frac{1}{\varepsilon}\right)\right). \tag{3.2}$$

The proof is now completed if we multiply (3.1) and (3.2). ∎

## 4. EMPIRICAL RESULTS AND CONCLUSIONS

In this section we present two tables comparing theoretical aspects and actual computing times for the methods of approximation by bisection and continued fractions.

Table I indicates that, in order to achieve a specified degree of accuracy, more bisections are needed than are partial quotients. The assumptions are that for the bisection method the initial isolating interval is $(0, 1]$, whereas for the continued fraction method it is $(0, \infty)$. Moreover, for the latter method we take the worst possible case, i.e. $a_i = 1$ for all $i$'s, and in this case $Q_i = F_i$, where $F_i$ is the $i$th member of the Fibonacci sequence (see also (2.4)).

TABLE I

Comparison of the number of partial quotients and bisections needed to obtain a specified degree of accuracy

| $m$ | Continued fraction method $(1/F_m^2)$ | Bisection method $(1/2^m)$ |
|---|---|---|
| 1 | 1.0 | 0.5 |
| 5 | 0.004 | 0.03125 |
| 10 | $3.3 \times 10^{-4}$ | $9.7 \times 10^{-4}$ |
| 15 | $2.7 \times 10^{-6}$ | $3.1 \times 10^{-5}$ |
| 20 | $2.2 \times 10^{-8}$ | $9.5 \times 10^{-7}$ |
| 25 | $1.7 \times 10^{-10}$ | $1.9 \times 10^{-8}$ |
| 30 | $1.4 \times 10^{-12}$ | $9.3 \times 10^{-10}$ |
| 35 | $3.1 \times 10^{-14}$ | $1.5 \times 10^{-11}$ |
| 40 | $9.5 \times 10^{-17}$ | $2.3 \times 10^{-13}$ |

In Table II we compare the actual computing times of the bisection method with all three versions of the continued fraction method. All times are in seconds and were obtained by using the SAC-1 computer algebra system, on the Honeywell 66/60 computer of the University of Kansas, to approximate all the real roots of the Chebyshev polynomials to within $\varepsilon = 10^{-15}$.

From Table II we see that using Cauchy's rule, is much slower than the bisection method, because the former has to be applied several times in order to compute the integer part of the root. However, the new method improves when we combine Corollary 2.1 with bisection

TABLE II

Times in seconds for the approximation of *all* the real roots of the Chebyshev polynomials to within $\varepsilon = 10^{-15}$

| | | Continued fractions using: | | |
|---|---|---|---|---|
| Degree | Bisection | Cauchy's rule | Corollary 2.1 with bisection | Preconditioning |
| 2 | 17.2 | 11.5 | 6.7 | 5.4 |
| 3 | 17.9 | 10.3 | 4.9 | 3.8 |
| 4 | 42.3 | 38.7 | 15.7 | 10.3 |
| 5 | 45.8 | 40.0 | 16.4 | 10.8 |
| 6 | 83.1 | 99.8 | 46.2 | 29.2 |
| 7 | 90.9 | 105.1 | 44.6 | 27.0 |
| 8 | 146.3 | 277.8 | 93.0 | 50.2 |
| 9 | 170.6 | 257.6 | 106.2 | 62.2 |
| 10 | 243.2 | 524.3 | 202.8 | 116.2 |

to compute each partial quotient. The times in the last column were obtained assuming that a list of partial quotients is supplied as input (preconditioning). These times reflect the optimum time for the approximation of the real roots using continued fractions; they also indicate that a lot of time will be saved if a faster method is devised for computing the partial quotients $a_i$.

## Acknowledgement

We thank the unknown referee for his careful remarks and constructive criticism which improved the presentation of our paper.

## References

[1] A. G. Akritas, A correction on a theorem by Uspensky, *Bulletin of the Greek Mathematical Society* **19**, 1978, 278–285.

[2] A. G. Akritas, On the solution of polynomial equations using continued fractions, *Information Processing Letters* **9**, 1979, 182–184.

[3] A. G. Akritas, Reflections on a pair of theorems by Budan and Fourier, *Mathematics Magazine* **55**, No. 5, 1982, 292–298.

[4] A. G. Akritas, An implementation of Vincent's theorem, *Numerische Mathematik* **36**, 1980, 53–62.

[5] A. G. Akritas, Exact algorithms for the implementation of Cauchy's rule, *International Journal of Computer Mathematics* **9**, 1981, 323–333.

[6] A. G. Akritas and S. D. Danielopoulos, On the forgotten theorem of Mr. Vincent, *Historia Mathematica* **5**, 1978, 427–435.

[7] A. G. Akritas and S. D. Danielopoulos, On the complexity of algorithms for the translation of polynomials, *Computing* **24**, 1980, 51–60.

[8] W. S. Burnside and A. W. Panton, *Theory of Equations*, Dublin University Press Series 1886, Dublin and London, Second edition.

[9] L. E. Heindel, Integer arithmetic algorithms for polynomial real zero determination, *Journal of the Association for Computing Machinery* **18**, 1971, 533–548.

[10] J. L. Lagrange, *Traité de la Résolution des Équations Numériques*, Paris, 1798.

[11] K. H. Ng, Polynomial real root approximation using continued fractions, *M.S. Research Report*, University of Kansas, Department of Computer Science, Lawrence, Kansas, 1980.

[12] N. Obreschkoff, *Verteilung und Berechnung der Nullstellen Reeller Polynome*, VEB Deutscher Verlag der Wissenschaften, Berlin, 1963.

[13] S. R. Petricle (ed.), *Proceedings of the Second Symposium on Symbolic and Algebraic Manipulation*, ACM, 1971.

[14] I. Todhunter, *Theory of Equations*, Macmillan and Co., London, 1882.

[15] J. V. Uspensky, *Theory of Equations*, McGraw-Hill, New York, 1948.

[16] P. Verbaeten, Computing real zeros of polynomials with SAC-1, *SIGSAM* **9**, No. 2, 1975, 8–10.

[17] A. J. H. Vincent, Sur la Résolution des Équations Numériques, *Journal de Mathématiques Pures et Appliquées* **1**, 1836, 341–371.