Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 2458-9403
Vol. 3 Issue 7, July - 2016

The Usage of OpenMp Platform in Solving Polynomial Equations

Rinela Kapçiu [1]
Computer Science Department
"Aleksandër Moisiu" University of Durrës
Durrës, ALBANIA
topalli_r78@hotmail.com

Fatmir Hoxha[2]
Applied Mathematics Department
University of Tirana
Tirana, ALBANIA
fat.hoxha51@gmail.com

***Abstract*— In this article we present a parallel implementation of the simultaneous methods algorithms for finding the roots of polynomials of high degree in OpenMP platform. We have implemented both a CPU version in C++ and a compatible version with OpenMP platform. The main result of our work is to emphasize the advantages of parallel implementation in solving higher degree polynomial equations. This platform has the benefit of using a personal computer at the optimal usage.**

Keywords*— Simultaneous method; root; polynomial; Durand – Kerner method; Börsch – Supan method; Ehrlich – Aberth method; parallelization.

I. INTRODUCTION

One of the most important problems in solving nonlinear equations is the construction of the initial conditions which provide rapid convergence of numerical algorithm. In this paper we present three methods Durand – Kerner, Börsch – Supan and Ehrlich – Aberth, which have some initial new conditions to ensure convergence of methods for solving algebraic equations. The stated initial conditions are of practical importance since they are computationally verifiable, they depend only on the coefficients of a given polynomial, its degree *n* and initial approximations to polynomial roots. [1]

One of the main problems in the solution of equations of the form *f(z) = 0* is the construction of the initial conditions which offer guaranteed convergence of numerical algorithms. These initial conditions include an initial approximation $z^{(0)}$ to the root of *f* with which starts the implementation of the algorithm to generate the sequence $\left\{z^{(m)}\right\}_{m=1,2,\ldots}$ of approximations tends to the root of *f*. The study of a general problem of the construction of the initial conditions and the choice of initial approximation to ensure convergence is very difficult and generally can not be solved in a satisfactory way, even for simple functions such as algebraic polynomials.

II. THE SIMULTANEOUS METHODS

In this paper we present three methods that provide improved conditions and fast convergence. These methods are: Durand – Kerner, Börsch – Supan and Ehrlich – Aberth. These conditions depend only on the coefficients of the given polynomial $P(z) = z^n + a_{n-1}z^{n-1} + \cdots + a_1 z + a_0$ of degree *n* and the vector of initial approximations $z^{(0)} = \left(z_1^{(0)}, \ldots, z_n^{(0)}\right)$. Most of iterative methods for the simultaneous determination of roots of o polynomial can be expressed as:

$$z_i^{(m+1)} = z_i^{(m)} - C\left(z_1^{(m)}, \ldots, z_n^{(m)}\right) \quad \left(i \in I_n, m = 0,1,\ldots\right) \quad (1.1)$$

Where $z_1^{(m)}, \ldots, z_n^{(m)}$ are distinct approximations to simple roots $\zeta_1, \ldots, \zeta_n$ respectively, obtained in the *m*-th step. The term $C_i^{(m)} = C_i\left(z_1^{(m)}, \ldots, z_n^{(m)}\right) \ \left(i \in I_n\right)$ will be called the *iterative correction term* or simply the *correction*. [1]

Let $\Lambda(\zeta_i)$ a close neighborhood of the root $\zeta_i \left(i \in I_n\right)$ and the function $\left(z_1, \ldots, z_n\right) \to F_i\left(z_1, \ldots, z_n\right)$ that satisfies the following conditions for each $i \in I_n$:

$$\boxed{\begin{array}{ll} (1) & F_i\left(\zeta_1, \ldots, \zeta_n\right) \neq 0, \\ (2) & F_i\left(z_1, \ldots, z_n\right) \neq 0 \quad \textit{for distinct approximations } z_i \in \Lambda\left(\zeta_i\right), \\ (3) & F_i\left(z_1, \ldots, z_n\right) \quad \textit{is continuous in } \mathbb{C}^n \end{array}}$$

If the correction term of iterative method 1.1 has the form

$$C_i\left(z_1, \ldots, z_n\right) = \frac{P\left(z_i\right)}{F_i\left(z_1, \ldots, z_n\right)} \quad \left(i \in I_n\right) \quad (1.2)$$

for which conditions (1) – (3) hold and $z_1^{(0)}, \ldots, z_n^{(0)}$ are the initial approximations to the polynomial roots, in [5] proved that this method is convergent if there is a real number $\beta \in (0,1)$ such that satisfies the following inequalities::

$$(i) \quad \left|C_i^{(m+1)}\right| \leq \beta \left|C_i^{(m)}\right| \quad \left(m = 0,1,\ldots\right).$$

$$(ii) \quad \left|z_i^{(0)} - z_j^{(0)}\right| > g(\beta)\left(\left|C_i^{(0)} + C_j^{(0)}\right|\right) \quad \left(i \neq j, \quad i,j \in I_n\right)$$

A. THE DURAND – KERNER METHOD

One of the most useful simultaneous methods for solving a polynomial is the Durand - Kerner (Weierstrass) method expressed as follows:

$$z_i^{(m+1)} = z_i^{(m)} - W_i^{(m)} \quad (i \in I_n, m = 0,1,\dots) \qquad (1.3)$$

where

$$W_i^{(m)} = \frac{P\left(z_i^{(m)}\right)}{\prod_{\substack{j=1\\j\neq i}}^{n}\left(z_i^{(m)} - z_j^{(m)}\right)} \quad (i \in I_n, m = 0,1,\dots)$$

In this case the correction term is equal to Weierstrass's correction

$$C_i = W_i = \frac{P(z_i)}{F_i(z_1,\dots,z_n)}$$

ku $F_i(z_1,\dots,z_n) = \prod_{\substack{j=1\\j\neq i}}^{n}(z_i - z_j) \quad (i \in I_n)$.

The Durand – Kerner Algorithm

1. Compute initial values $\{z_0;\dots;z_{n-1}\}$
2. Let m=1;
3. *do*
4. $\Delta z_{\max} = 0$;
5. *for* j = 0, ..., n-1
6. $z_j^{m-1} = z_j^m$;
7. $z_j^{m-1} = H_i\left(z^{m-1}\right)$;
8. Set $\Delta z_{\max} = \frac{\left|z_j^m - z_j^{m-1}\right|}{z_j^m}$;
9. k=k+1;
10. *while* $z_{\max} > \varepsilon$

B. The Börsch – Supan Method

The Börsch – Supan method is a simultaneous method, which is determined by the iterative formula

$$z_i^{(m+1)} = z_i^{(m)} - \frac{W_i^{(m)}}{1 + \sum_{\substack{j=1\\j\neq i}}^{n} \frac{W_j^{(m)}}{\left(z_i^{(m)} - z_j^{(m)}\right)}} \quad (i \in I_n, m = 0,1,\dots) \qquad (1.4)$$

where $W_i^{(m)}$ is given by (1.2). This formula has term correction:

$$C_i(z_1,\dots,z_n) = \frac{P(z_i)}{F_i(z_1,\dots,z_n)} \quad (i \in I_n)$$

where

$$F_i(z_1,\dots,z_n) = \left(1 + \sum_{j\neq i} \frac{W_j}{z_i - z_j}\right) \prod_{j\neq i}(z_i - z_j) \quad (i \in I_n)$$

The Börsch – Supan algorithm

1. Compute initial values $\{z_0;\dots;z_{n-1}\}$
2. Let m=1;
3. *do*
4. $\Delta z_{\max} = 0$;
5. *for* j = 0, ..., n-1
6. $z_j^{m-1} = z_j^m$;
7. $z_j^{m-1} = H_i\left(z^{m-1}\right)$;
8. Set $\Delta z_{\max} = \frac{\left|z_j^m - z_j^{m-1}\right|}{z_j^m}$;
9. $z_i^{(m+1)} = z_i^{(m)} - \frac{W_i^{(m)}}{1 + \sum_{\substack{j=1\\j\neq i}}^{n} \frac{W_j^{(m)}}{\left(z_i^{(m)} - z_j^{(m)}\right)}}$;
10. k=k+1;
11. *while* $z_{\max} > \varepsilon$

C. The Ehrlich – Aberth Method

The Ehrilch - Aberth method is one of the most efficient methods for finding simple roots simultaneously [17, 18]. Its iterative formula [8] is:

$$z_i^{(m+1)} = z_i^{(m)} - \frac{1}{\frac{P'\left(z_i^{(m)}\right)}{P\left(z_i^{(m)}\right)} - \sum_{\substack{j=1\\j\neq i}}^{n} \frac{1}{\left(z_i^{(m)} - z_j^{(m)}\right)}} \quad (i \in I_n, m = 0,1,\dots) \qquad (1.5)$$

As the Börsch – Supan method, the Ehrlich – Aberth method (1.5) has cubic convergence.
In [9] Carstensen demonstrated the identity:

$$\frac{P'(z_i)}{P(z_i)} - \sum_{j\neq i} \frac{1}{z_i - z_j} = \frac{1}{W_i}\left(\sum_{j\neq i} \frac{W_j}{z_i - z_j} + 1\right)$$

As compared with the iterative formulas (1.4) and (1.5), we conclude that they are equivalent. Theoretically, these two formulas should give the same results, but in practice, the effect of round - off error due to the use of a single – precision arithmetic brings that approaches computed by (1.4) and (1.5) to have a small difference. However, this effect is negligible in the first iterations.

The Ehrlich – Aberth algorithm

1. Compute initial values $\{z_0;\dots;z_{n-1}\}$
2. Let m=1;
3. *do*
4. $\Delta z_{\max} = 0$;
5. *for* j = 0, ..., n-1
6. $z_j^{m-1} = z_j^m$;

7. $z_j^{m-1} = H_i\left(z^{m-1}\right)$;

8. Set $\Delta z_{\max} = \dfrac{\left|z_j^m - z_j^{m-1}\right|}{z_j^m}$;

9. $z_i^{(m+1)} = z_i^{(m)} - \dfrac{1}{\dfrac{1}{W_i}\left(\sum_{j \neq i} \dfrac{W_j}{z_i^{(m)} - z_j^{(m)}} + 1\right)}$;

10. k=k+1;

11. *while* $z_{\max} > \varepsilon$

III. APPLICATION

In this section will compare the execution time in sequential algorithms (in C++ programming language) and in parallel algorithms (in OpenMP platform) mentioned above.

We consider the cubic polynomial $x^3 - 6x^2 + 11x - 6$, with roots 1, 2 and 3. Dispensing with the parameterization that was introduced in [19] for special purposes, we take directly the following initial approximations, which add up to zero and allow comparison

$$x_1^{(0)} = 0.5, \quad x_2^{(0)} = 1.5, \quad x_3^{(0)} = 2.5$$

TABLE I.

The Method	The Execution Time		Example $\left(x^3 - 6x^2 + 11x - 6\right)$	
	C++	OpenMP	The Roots	The iteration
Durand – Kerner	8.174s	6.786s	{1,2,3}	6-th
Börsch – Supan	7.831s	5.741s	{1,3,2}	5-th
Ehrlich – Aberth	7.238s	6.053s	{1,3,2}	5-th

IV. CONCLUSIONS

As seen from *Table 1*, OpenMP execution time is more quail than the execution of the same algorithm in C++. The execution of the Durand - Kerner algorithm in OpenMP approximates the roots in the sixth iteration and for a time 1.2045 times faster than its execution in C++. The execution of the algorithms of three other methods in OpenMP approximate the roots in the fifth iteration and the time of execution in OpenMP of Börsch – Supan algorithm is 1,364 times faster than the time of its execution in C++, while the execution time of the Ehrlich - Aberth algorithm in OpenMP is 1.1978 times faster than its execution time in C++. So we concluded that:

By testing these three simultaneous methods we see that the OpenMP platform is more qualitative than the sequential execution. As seen on the platform OpenMP implemented in our algorithms, their performance increases and this happens in the same drive hardware, with the same parameters, just exploiting parallelism and increasing the use of all potential multithread processor.

OpenMP is well adapted to intensive computing. We parallelized the Durand-Kerner algorithm, Börsch – Supan algorithm and Ehrlich – Aberth algorithm for polynomial roots - finding and we obtained encouraging results. Indeed, the experimental study confirms that our program determines the same roots than the sequential version for high degrees. The contribution of the parallel solution allows us to accelerate the execution time and to study even more important degrees of polynomial.

REFERENCES

[1]. M. S. Petkovic[a,*], Đ. Herceg, Point estimation of simultaneous methods for solving polynomial equations: a survey, J. Computational and Applied Mathematics 136 (2001) 283 – 307

[2]. P. Batra, Improvement of a convergence condition for the Durand – Kerner iteration, J. Comput. Appl. Math. 96 (1998) 117 – 125.

[3]. M. S. Petkovic, On initial conditions for the convergence of simultaneous root - finding methods, Computing 57 (1996) 163 – 177.

[4]. M. S. Petkovic, C. Carstensen, M. Trajkovic, Weierstrass' formula and zero - finding methods, Numer. Math. 69 (1995) 353 – 372.

[5]. M. S. Petkovic, Đ. Herceg, S. Ilic, Safe convergence of simultaneous methods for polynomial zeros, Numer. Algorithms 17 (1998) 313 – 331.

[6]. A. W. M. Nourein, An iteration formula for the simultaneous determination of the zeroes of a polynomial, J. Comput. Appl. Math. 4 (1975) 251 – 254.

[7]. M. S. Petkovic[a,*], Đ. Herceg, Point estimation and safe convergence of root - finding simultaneous methods, Sci. Rev. 21 – 22 (1996) 117 – 130.

[8]. W. Börsch – Supan, A posteriori error bounds for the zeros of polynomials, Numer. Math. 5 (1963) 380 – 398.

[9]. C. Carstensen, On quadratic – like convergence of the means for two methods for simultaneous root – finding of polynomials, BIT 33 (1993) 64 – 73.

[10]. F. L. Lucio On the convergence of a parallel algorithm for finding polynomial zeros, 1997.

[11]. M. Ben et al, A fast parallel algorithm for determining all roots of a polynomial with real roots, SIAM J. Comp. 11, 1988.

[12]. D. Bini, L. Germignani, On the complexity of polynomial zeros, SIAM, J. Comp. 21, 1992.

[13]. T. Auckenthaler, et al, Parallel solution of partial symmetric eigenvalue problems from electronic structure calculations, Parallel Comp. 2011.

[14]. E. R. Alcalde, Parallel implementation of Davidson – type methods for large – scale eigenvalue problem, 2012.

[15]. C. Caretensen and M. S. Petkovic, On iterative methods without derivatives for the simultaneous determination of polynomial zeros, *J. Comp. & Appl. Math.*, **45**, pp 251 – 266, 1993 E238

[16]. Z. Bai, Demmel, J. Dongarra, A. Ruhe, H. Van der Vorst, *Templates for the solution of algebraic eigenvalue problems: a practical guide*, SIAM, Philadelphia, 2000. E238

[17]. O. Aberth, Iteration methods for finding all zeros of a polynomial simultaneously, Math.Comp.27 (1993) 339–344.

[18]. L. W. Ehrlich, A modified Newton method for polynomials, Comm. ACM 10 (1967) 107–108.

[19]. M. Hopkins, B. Marshall, G. Schmidt and S. Zlobec, On a method of Weierstrass for the simultaneous calculation of the roots of a polynomial, ZAMM 74 (8), 295-306 (1994).