International Journal of Research In Science & Engineering e-ISSN: 2394-8299
Volume: 3 Issue: 5 Sep.-Oct. 2017 p-ISSN: 2394-8280

AN EFFICIENT ROOT FINDING METHOD IN MATHEMATICS: A SURVEY

Prof. Nita K. Choukade[1], Prof. Amit P. Mangalekar[2]
Asst. Professor Mathematics Deptt., J.C.O.E.T. Yavatmal
nitachoukade8@gmail.com
amitpmanglekar@gmail.com

ABSTRACT:

In mathematics root-finding algorithms arefor finding roots of continuous functions. Root of any function f(x), from real numbers to real numbers or from complex numbers to complex numbers, is a number x such that f(x) = 0. Generally roots of any function cannot be computed exactly nor even expressed in close form, Root finding algorithms always provides approximate roots and expressed as floating point numbers or small isolating intervals, or disks for the complex roots.

Here analysis and comparisonof several Efficient Root Finding MethodsIn Mathematics is done.This Paper also introduces Efficient Root Finding Method In Mathematics. In PastMany Root Finding Methods are not only derive from scholar's classic approach but are of equivalentapproach. The related computational experiments show the built-in multiplicity estimates can significantly decrease the number of iterations, while the error of these estimates may significantly increase. This work Help to reduce the efforts of newly scholar to find Efficient Roots.

Keywords:-Delta (δ), Matrix ([]), Less than (<),Rho (ρ).

INTRODUCTION: -

Most numerical root-finding methods use iteration, producing a sequence of numbers that hopefully converge towards the root as a limit [1][3][5]. They require one or more initial guesses of the root as starting values, then each iteration of the algorithm produces a successively more accurate approximation to the root [2][4][6]. Since the iteration must be stopped at some point these methods produce an approximation to the root, not an exact solution [7][9][11]. Many methods compute subsequent values by evaluating an auxiliary function on the preceding values [8][10][12]. The limit is thus a fixed point of the auxiliary function, which is chosen for having the roots of the original equation as fixed points, and for converging rapidly to these fixed points[13-20].

The behaviour of root-finding algorithms is studied in numerical analysis[21-24]. The efficiency of an algorithm may depend on the characteristics of the given functions [25-28]. For example, many algorithms use the derivative of the input function, while others work on every continuous function. In general, numerical algorithms cannot guarantee to find all the roots of a function, so failing to find a root does not prove that there is no root. However, for polynomials, there are specific algorithms that use algebraic properties for locating the roots in intervals (or disks for complex roots) that are small enough for insuring the convergence of numerical methods (typically Newton's method) to the unique root so located.

In numerical linear algebra, the Gauss–Seidel method, also known as the Liebmann method or the method of successive displacement, is an iterative method used to solve a linear system of equations. It is named after the German mathematicians Carl Friedrich Gauss and Philipp Ludwig von Seidel, and is similar to the Jacobi method. Though it can be applied to any matrix with non-zero elements on the diagonals, convergence is only guaranteed if the matrix is either diagonally dominant, or symmetric and positive definite. It was only mentioned in a private letter from Gauss to his student Gerling in 1823.[1] A publication was not delivered before 1874 by Seidel.

This article is about iterative methods for solving systems of equations. For other uses, see Relaxation Method(disambiguation).In numerical mathematics, relaxation methods are iterative methods for solving systems of equations, including nonlinear systems.[1]Relaxation methods were developed for solving large sparse linear systems, which arose as finite-difference discretization's of differential equations.[2][3] They are also used for the solution of linear equations for linear least-squares problems[4] and also for systems of linear inequalities, such as those arising in linear programming.[5][6][7] They have also been developed for solving nonlinear systems of equations.[1]Relaxation methods are important especially in the solution of linear systems used to model elliptic partial differential equations, such as Laplace's equation and its generalization, Poisson's equation. These equations describe boundary-value problems, in which the solution-function's values are specified on boundary of a domain; the problem is to compute a solution also on its interior. Relaxation methods are used to solve the linear equations resulting from a discretization of the differential equation, for example by finite differences. [4][3][2]These iterative methods of relaxation should not be confused with "relaxations" in mathematical optimization, which approximate a difficult problem by a simpler problem, whose "relaxed" solution provides information about the solution of the original problem.[7]

METHODOLOGY: -

Gauss Seidel Method: - This is the elementary elimination method & it reduces the system of equations to an equivalent upper –triangular system, which can be solved by Back substitution.

Consider the equations

$$a_1x+b_1y+c_1z = d_1 \ldots\ldots\ldots\ldots(1)$$
$$a_2x+b_2y+c_2z = d_2 \ldots\ldots\ldots\ldots(2)$$
$$a_3x+b_3y+c_3z = d_3 \ldots\ldots\ldots\ldots(3)$$

Where a_1,$b_{2,}c_3$ are called the elements of the principal diagonal of the system and they must be greater than the sum of the other two elements in each equation. If the problem given for solution is not in a diagonal system, thenarrange it in diagonal system andsolve. Equations are such as

$$x_1 = \frac{1}{a_1}(d_1 - b_1y - c_1z) \ldots\ldots(a)$$
$$y_1 = \frac{1}{b_2}(d_2 - a_2x - c_2z) \ldots\ldots(b)$$
$$z_1 = \frac{1}{c_3}(d_3 - a_3x - b_3y) \ldots\ldots(c)$$

Put the values of y = 0 & z = 0in the equation (a), the first iteration gives$x_1 = \frac{d_1}{a_1}$.

Use this value x = x_1,z = 0 in equation (b), we get y = y_1

Then put x = x_1, y = y_1 in equation (c), we get z = z_1

Method use x_1, y_1& $z_1$are as initial values &can calculate x_2, y_2 & z_2.

Similarly, Method uses x_2 ,y_2, z_2 *are* as initial values&calculate x_3, y_3 & z_3.

Relaxation Method: -

In Relaxation Method a solution of all unknowns is obtained simultaneously. The solution obtained is an approximation to a certain number of decimals. Consider the equations

a_1 x+b_1y+c_1z = d_1…………(1)
a_2 x+b_2y+c_2z = d_2…………(2)
a_3 x+b_3y+c_3z = d_3…………(3)

Then the quantities

$$R_x = d_1 - a_1x - b_1y - c_1z,\ R_y = d_2 - a_2x - b_2y - c_2z,\ R_z = d_3 - a_3x - b_3y - c_3z$$

Are called residues of the n equations. Thesolution of the n equations is a set of numbers x, y…,z. That makes all the R equal to zero. We shall obtain an approximate solution by using an iterative method which makes the R smaller & smaller at each step so that we get closer & closer to the exact solution. At each stage, the numerically largest residual is reduced to almost zero.To start with we assume x = y = z = 0 & calculate the initial residuals. Then the residuals are reduced step by step by giving increments to the variables. For thispurpose, method construct the following operation table.

			δR_X	δR_y	δR_z
$\delta_x = 1$	$\delta_y = 0$	$\delta_z = 0$	$-a_1$	$-a_2$	$-a_3$
$\delta_x = 0$	$\delta_y = 1$	$\delta_z = 0$	$-b_1$	$-b_2$	$-b_3$
$\delta_x = 0$	$\delta_y = 0$	$\delta_z = 1$	$-c_1$	$-c_2$	$-c_3$

Table1:- For Value Put Up & Residuals

If x is increased by 1 (keeping y & z constant) R_x, R_y, R_z decreases by a_1, a_2, a_3 respectively. This is shown in the above table along with the effects on the residuals when y & z are given unit increments. The table is the transpose of the coefficient matrix.

At each step, the numerically largest residual is reduced to almost zero. To reduce a particular residual, the value of the corresponding variable is changed,e.g.to reduce)R_x by p,x should be increased by $\frac{p}{a_1}$. When all the residuals have been reduced to almost zero, the increments in x,y,z are added separately to give the desired solution urgent.

Simultaneous Equation by Matrix Method: -

A system of linear equations

$a_{11}x_1 + a_{12}x_2 + a_{1n}x_n = b_1$……..(1)

$a_{21}x_1 + a_{22}x_2 + a_{2n}x_n = b_2$……..(2)

$a_{m1}x_1 + a_{m2}x_2 + a_{mn}x_n = b_m$…...(3)

Can be expressed in the matrix form as AX = B ………..(a)

$$\text{Where } A = \begin{pmatrix} a_{11}a_{12} \dots\dots a_{1n} \\ a_{21}b_{22} \dots\dots \quad a_{2n} \\ a_{m1}a_{m2} \dots\dots \quad a_{mn} \end{pmatrix}$$

$$X = \begin{pmatrix} x_1 \\ \\ \\ \end{pmatrix} \qquad B = \begin{pmatrix} b_1 \\ \\ \\ \end{pmatrix}$$

x_2 b_2

x_3 b_3

The system of equations 1, 2 & 3 is said to be consistent if they have one or more solution. Other-wise it is said to be inconsistent.

To find the solution, if exist.by matrix method we define the matrix [A:B] calledas augmented matrix C.

$$C = [\ A: B] = \begin{bmatrix} a_{11}a_{12} \dots \dots . . a_{1n}b_1 \\ a_{21}a_{22} \dots \dots . . a_{2n}b_2 \\ a_{m1}a_{m2} \dots \dots . . a_{mn}b_n \end{bmatrix}$$

Find the rank of the augmented matrix.

If Rank A= Rank C, then the given system (1) is consistent otherwise inconsistent.
Case I: - If $\rho(A) = \rho(C)$ = Number of unknown variables. Then system has unique solution.
Case II: - If $\rho(A) = \rho(C) <$ Number of unknown variables. Then system has infinite solution. (i.e. number of set of solution)

LIMITATIONS: -

In all three Methods if Complex values are given then it will not provide the roots of the equations. In case of Solution of simultaneous method by using matrix method, if auxiliary roots are not found then in method Matrix formation is not doe & the system become inconsistence.

CONCLUSION & FUTURE SCOPE: -

This work analyse three roots finding methods such as Gauss Seidal method, Relaxation method & Solution of simultaneous method by using matrix method. All methods find number of iterations and provides approximate values of three roots. Comparatively Relaxation method provides the roots easily rather than the other methods. As the approximate roots are consider again Relaxation method is beneficiary. In case of Solution of simultaneous method by using matrix method it is more complex & make this method tedious. In future if Solution of simultaneous method by using matrix method, provides rank easily then it is helpful to mathematicians.

REFERENCES:-

1.Bodewig, E.: On types of convergence and behaviour of approximations in the neighbourhood of a multiple root of an equation. Q. Appl. Math. **7**, 325–333 (1949) Math Sci Net MATH Google Scholar
2.Channabasappa, M.: A note on the computation of multiple zeros of polynomial by Newton's method. BIT **19**, 134–135 (1979) Cross Ref Math Sci Net MATH Google Scholar
3.Crouse, J.E., Putt, C.W.: Roots of Polynomials by Ratio of Successive Derivatives. Technical Report, NASA TN D-6793, NASA, Washington (1972) Google Scholar
4.Delahaye, J.-P.: Sequence Transformations. Springer, New York (1988) MATH Google Scholar
5.Hansen, E., Patrick, M.: Estimating the multiplicity of a root. Numer. Math. **27**, 121–131 (1976) Cross Ref Math Sci Net MATH Google Scholar

6.Henrici, P.: Elements of Numerical Analysis. Wiley, New York (1964) MATH Google Scholar
7.Hernández Verón, M.A.: A note on Halley's method. Numer. Math. **59**, 273–276 (1991) Cross Ref Math Sci Net MATH Google Scholar
8.Hernández Verón, M.A.: Newton-Raphson's method and convexity. Univ. u NovomSadu, Zb. Rad. Prirod. Mat. Fak., Ser. Mat. **22**(1), 159–166 (1992) MATH Google Scholar
9.Hernández M.A., Salanova, M.A.: Indices of convexity and concavity. Application to Halley method. Appl. Math. Comput.**103**, 27–49 (1999) MATH Google Scholar
10.Hull, T.E., Mathon, R.: The mathematical basis and a prototype implementation of a new polynomial rootfinder with quadratic convergence. ACM Trans. Math. Softw. **22**, 261–280 (1996) Cross Ref Math Sci Net MATH Google Scholar
11.Igarashi, M., Ypma, T.: Relationships between order and efficiency of a class of methods for multiple zero of polynomials. J. Comput. Appl. Math. **60**, 101–113 (1995) Cross Ref Math Sci Net MATH Google Scholar
12.Igarashi, M., Ypma, T.: Empirical versus asymptotic rate of convergence of a class of methods for solving polynomial equations. J. Comput. Appl. Math. **82**, 229–237 (1997) Cross Ref Math Sci Net MATH Google Scholar
13.Jenkins, M., Traub, J.: Principles for testing polynomial zerofinding programs. ACM Trans. Math. Softw. **1**, 26–34 (1975) Cross Ref Math Sci Net MATH Google Scholar
14.Kahan, W.: Lecture notes on real root-finding. http://www.cs.berkeley.edu/~wkahan/Math128/RealRoots.pdf (2004)
15.King, R.: Improving the Van de Vel Root-Finding method. Computing **30**, 373–378 (1983) Cross Ref Math Sci Net MATH Google Scholar
16.Kravanja, P., Haegemans, A.: A modification of Newton's method for analytic mappings having multiple zeros. Computing **62**, 129–145 (1999) Cross Ref Math Sci Net MATH Google Scholar
17.Krishnamurty, E.V., Sen, S.K.: Computer Based Numerical Algorithms. Affiliated East-West Press Pvt. Ltd., New Delhi (1976) Google Scholar
18.Lang, M., Frenzel, B.C.: A New and Efficient Program for Finding All Polynomial Roots. Technical Report #9308, Rice University, Houston (1994) Google Scholar
19.Madsen, K.: A root-finding algorithm based on Newton's method. BIT **13**, 71–75 (1973) Cross Ref Math Sci Net MATH Google Scholar
20.Madsen, K., Reid, J.K.: FORTRAN Subroutines for Finding Polynomial Zeros. Technical Report A.E.R.E. R.7986, Computer Science and System Division, A.E.R.E. Harwell, Oxfordshire (1975) Google Scholar
21.Mathews, J.H.: An improved Newton's method. AMATYC Rev. **10**, 9–14 (1989) Google Scholar
22.McNamee, J.: A comparison of methods for accelerating convergence of Newton's method for multiple polynomial roots. ACM SIGNUM Newsl. **33**, 17–22 (1998) Cross Ref Google Scholar
23.McNamee, J.M. : Numerical Methods for Roots of Polynomials, Part I. Elsevier, Amsterdam (2007) MATH Google Scholar
24.A.A.Chincholkar &C.H.Thakare: GA Approach to solve the Dyanamic Channel Assignment Problem, International Journal od Advent Research in Computer & Electronics, ISSN 2348-5523(2014) Cross Ref Google Scholar
25.Ostrowski, A.: Solution of Equations and Systems of Equations. Academic, London (1960) MATH Google Scholar
26.Rall, L.: Convergence of the Newton process to multiple solutions. Numer. Math. **9**, 23–37 (1966) Cross Ref Math Sci Net MATH Google Scholar

27.Schröder, E.: Überunen dlichviele Algorithm enzur Auflösung der Gleichungen. Math. Ann. **2**, 317–365 (1870). English translation by G.W. Stewart: On Infinitely Many Algorithms for Solving Equations. Technical Report TR-92-121, University of Maryland, Department of Computer Science (1992) Cross Ref Math Sci Net Google Scholar
28.Steiglitz, K., Dickinson, B.: Phase unwrapping by factorization. IEEE Trans. Acoust. Speech Signal Process. **30**, 984–991 (1982) Cross Ref Math Sci Net MATH Google Scholar