Improved the Convergence of Iterative Methods for Solving Systems of Equations by Memetics Techniques

Liviu Octavian Mafteiu-Scai
West University of Timisoara,
Timisoara, Romania

ABSTRACT

This work proposes proposed a technique inspired by memetic algorithm (MA) to improve the convergence of iterative methods for solving systems of equations. In the first phase the system of equations is transformed into an optimization problem. In this first phase, a memetics technique -ie a double optimization, local and global- is used to determine an initial vector favorable to a rapid convergence. In the second phase the system of equations is solved using an iterative method with the initial vector obtained in the previous phase. One can say that it is a hybrid method of solving systems of equations, both linear and nonlinear. The experimental results obtained with conjugate gradient, preconditioned conjugate gradient, Newton, Chebyshev and Broyden methods, serial and parallel versions, recommend the proposed method.

General Terms

Algorithms.

Keywords

systems of equations, memetic algorithms, iterative methods, convergence, intial vector, basin of attraction

1. THEORETICAL CONSIDERATIONS

1.1 Memetic algorithms and optimization problems

Memetic Algorithms (MAs) are computational intelligence structures combining multiple and various operators in order to address optimization problems [1]. The memetic algorithms (MAs) are inspired from genetic and cultural evolution. Memetic algorithms are in case of global optimization problems a good choice, that combine evolutionary techniques with other classical or intelligent optimization techniques. A memetic algorithm can be seen as an evolutionary algorithm that incorporates knowledge about the problem domain being solved.In general, a memetic algorithm use a double optimization: local and global. and represent a particular class of evolutionary algorithms that apply a local search in order to refine the current approximation of the global optimum. From another point of view, the MAs are population-based metaheuristics. A detailed description of memetic algorithms is made by Ferrante Neri et. all in *Handbook of Memetic Algorithms* [1]. Other nature-inspired intelligent algorithms are presented in the works [2], [3], [4], [5] and [6].

The main steps of a general MA are:

```
Step1: Initialize Population
Step2: Evaluate Curent Population
Step3: Select Parents
Step4: Crossover and Mutation to parents
Step5: Evaluate New Chromosomes
Step6: Improve with local search
Step7:If population converge then restart population
Step8: If termination criterion is not met, go to Step2
```

In the process of designing a memetic algorithm must be considered the following aspects:

-the choice of recombination operators [3];

-the separation and the balance between local and global optimization;

-refreshing the population when population converges, to avoid the exploration of the same search space, which would lead to obtaining the same solutions and unnecessary waste of processing time;

-a non-random initialization of initial population is recommended, which can direct the search into a particular regions that contain good or appropiate solutions. This can be achieved mainly by the inclusion in the initial population of good solutions previously known or by a process selected based on fitness from a large population generated randomly [7];

-in local search, the recombination and mutation operators must be different;

1.2 Solving systems of equations

A frequent problem in numerical analysis is solving the systems of equations (SE). There are classical numerical methods and methods inspired by techniques from artificial intelligence. Hybrid methods have been also proposed along the time [8].

The classical methods are usually divided into direct and iterative methods. In direct methods, like Cramer, Gaussian elimination, Gauss-Jordan elimination, LU factorization, QR decomposition etc., the solution is obtained after a fixed number of operations, a number that is directly proportional to system size. The solution is affected by rounding errors at each step, fact which not recommend the use of these methods for large systems. For large system of equation are prefered iterative methods. These methods are based on an iterative process that starts from an initial approximation of the solution. This is possible that the given system is well conditioned.

A major advantage of iterative methods is that in practice rounding errors and truncation errors can be insignificant sometimes even eliminated. Note that in terms of implementation the iterative methods are simpler. The basic process in iterative mthods consists in building a sequence x_k that converges to the exact solution of the system. The iterative process is stopped when reaching a specific precision imposed by the user. Below are described the characteristics of the most popular iterative methods.

The Conjugate Gradient (CG) method:

The method proposed by M.R. Hestenes and E. Stiefel in [9] is seen as a special case of Gaussian elimination. The method is effective for symmetric and positive definite systems. When it is generalized for unsymmetrical systems one of the advantages will be killed: the short recurrence or accuracy. Such generalizations are proposed in works such as [10] and [11]. The main features of the CG method are:

-a small number of recurrences to determine the search direction;

-the method involves small errors;
-due to small errors, the method will generate accurate solutions at most *n* steps;
-in practice, rounding errors can cause loss of orthogonality;
-the errors are reduced, on average, at each iteration by a factor $\frac{\sqrt{k}-1}{\sqrt{k}+1}$ where k=cond(A)=||A||||A^{-1}|| is the condition number of associated matrix A;
-the convergence tends to be faster if the system is well conditioned and can be arbitrary otherwise.

The Preconditioned Conjugate Gradient (PCG) method:

The CG method can be modified to reduce the computation time by choosing a matrix P^{-1}, called *preconditioning matrix* with the property $P^{-1}\approx A^{-1}$ followed by solving the system $P^{-1}Ax\approx P^{-1}b$, where $Ax=b$ is the equations system. The cases in which is recommended the choice (search) the matrix *P* are those when the system is poorly conditioned and has a big condition number. The simplest form for *P* is a diagonal matrix, method known as *Jacobi preconditioning*. In paper [12] are presented the most popular preconditioning methods used to solve large systems of equations.

The Newton-type methods

The Newton-type methods are in general for nonlinear systems of equations. A.Galant in [13] shows the most important theoretical aspects of Newton-type methods for solving systems of equations, in terms of convergence, error, stability and complexity of the method.

In [14] the author proposes an approximation that leads to a quasi-Newton algorithm for systems of linear equations. The proposed method is distinguished by a smaller number of iterations required for convergence than conjugate gradient when the condition number of the system of equations is very high.

The Broyden method

The Broyden method is a quasi-Newton-type method and has been first described in 1965 by Broyden [15]. The method is characterized by a very good convergence, reaching solution in *2n* steps, where *n* is the size of the system, demonstration made by DM Gay in [16].

The Chebyshev method

The method was named after the Russian mathematician Pafnuty Chebyshev. Its main advantage is that it can solve unsymmetrical systems of linear equations. Regarding the good convergence of this method, a complete treatment of the subject is made by Golub in [17] and by Manteuffel in [18]. Also in [17] is shown that a good choice of parameters leads to increased efficiency of the Chebyshev method compared with conjugate gradient. The numerical stability of the Chebyshev method for solving large systems of linear equations is analyzed by Wozniakowski in [19].

Note 1: Many works show the importance of the initial vector in solving systems of equations by iterative methods, an example being the paper [30]. This fact was mentioned because in our approach, the initial vector/approximation is a central point of interest.

Note 2: Another point of interest in our approach was the *basin of attraction* in the case of iterative methods for solving systems of equations. Many studies [31,32,33 etc) show that in iterative methods, especially for Krylov subspace methods, it is a very interesting but a dificult problem to determine the basin of attraction of a stable fixed point, defined as the set of all initial data that ends up converging to it. Has been shown that if the initial values are outside a basin of attraction can lead to divergent iterates.

In the last years, artificial intelligence techniques have been used to solve systems of equations. In paper [20] is proposed an algorithm that uses a genetic approach to solve linear systems of equations. The problem solving is viewed in terms of an optimization task. In paper [21], in terms of the objective function, the approach is similar to that in [20], but the problem is interpreted as a multiobjective optimization task. Approaches based on multiobjective optimization were proposed also in papers [22,23,24,25]. In [26], the author propose a method based on particle swarm optimization, designed for ill-conditioned linear systems equations. In paper [7] a method that use a memetic algorithm is proposed. The proposed method in [7] is able to determine solutions of a given linear system of equations, even in cases where clasical methods fail (determinant null, ill-conditioned systems, subdeterminate and supradeterminate systems, system does not satisfy the convergence conditions etc).

The interest in parallel solving systems of equations, especially those very large and sparse, has been very high, there are hundreds of papers that deal with this subject. The main aspects of parallel solving of large systems of equations can be found in papers [26], [27] and [28].

2. THE PROPOSED METHOD

In Figure 1 is represented the basic structure of the proposed hybrid method.

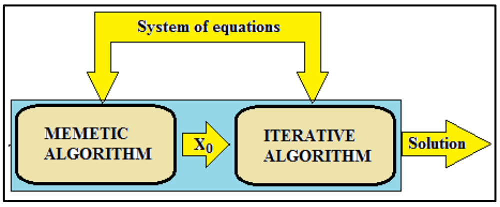

Figure 1 Structure of the proposed hybrid method

The memetic component role is to determine a good/optimal value for the initial vector X_0. This component is based on the algorithm proposed in [7] with some specific particularities which will be specified later. The iterative component role is to solve the system of equations based on the value of X_0. This component can be a classical iterative method, serial or parallel, such as for example, conjugate gradient, preconditioned conjugate gradient, Broyden or Chebyshev.

In work [7] the problem of solving a linear system of equations is transformed into a multiobjective optimization problem, where each equation is used to define an objective function, named in term of evolutionary process, *fitness* function. The goal of these optimization functions is to minimize the difference between left side and right side for each equation in part, in absolute value, and show how good it is a particular solution. For example, if it considered a linear system of *n* equations with *n* unknows that can be written as $Ax=b$, with the unknowns $x=\{x_1, x_2, \ldots, x_n\}$,the coefficients $\{a_{11}, a_{12}, \ldots a_{nn}\}$ and the constant terms $b=\{b_1, b_2, \ldots, b_n\}$, the fitness functions considered in [7] were *abs(f*$_i$*(X))*, where $f_i(x_i) = a_{i1}x_1 + \cdots + a_{in}x_n - b_i$, *i=1,2,...,n*. I do not insist here on describing that algorithm and its advantages, a detailed description was made in [7]. It should be noted that the main disadvantage of the method proposed in [7] consists in the high execution time for medium and large systems of equations. This time becomes prohibitive if a high accuracy of the solution is desired, because it requires population growth in the evolutionary process and therefore increasing the processing time. The mentioned disadvantage can be removed by hybridization of the method proposed in [7] with an iterative method with a fast convergence, such as CG, PCG, Chebyshev or Broyden, through some adjustments to the method proposed in [7].

In the current approach the goal of the memetic component will not be to determine the solution/solutions of the system of equations. The new goal will be to find a good initial approximation inside the basin of attraction. Achieving this objective will ensure the convergence of iterative methods and more, it will improve the convergence. Below are listed the main changes made to the memetic algorithm proposed in paper [7] to improve the global performance of proposed hybrid method:

i) *Chromosomes.* A chromosome in the population is represented by a vector *S={x_1, x_2...x_n}*, with integer values in the search interval *[-r, + r]*. The *S* vector corresponds to a set of values assigned to the system unknowns *X={x_1, x_2...x_n}*.

ii) the relations for calculating the deviations (relation 4 and 5 in [7]) were modified to be valid and for systems of nonlinear equations ie:

$$dx_{ik} = \frac{b_i - \sum_{j=1}^{n}(a_{ij} \cdot \prod_{m=1}^{n} s_m^{p_{mj}})}{n * a_{ik}}$$

and

$$delta\big(f(X)\big) = \sum_{i=1}^{n} |b_i - \sum_{j=1}^{n}(a_{ij} \prod_{m=1}^{n} s_m^{p_{mj}})|$$

where p_{mj} is the exponent of the term s_m in equation *i*.

iii) *Evolutionary process.* Because it was observed that crossover does not help to much, to improve the efficiency of the memetic algorithm, it was excluded from the evolutionary process.

iv) *Termination Conditions:* In old form [7], the termination condition contains an iterator, initially set by the user, showing how many solutions are intended to be obtained, if there are infinite solutions. This condition has been eliminated for the proposed objective -finding a good value for X_0 vector- because we consider sufficient to find a single, even if approximate solution.

In [7] the minimum values for the *delta(f_i(x))* and *delta(x_i)* are also used in the *Termination Condition*. But the interval for *admissible approximation a_a*, specified there, was increased from [0,1] to [0,100] and can be preset by the user at the beginning of the computing process.

3. EXPERIMENTAL RESULTS

After the experiments, it was observed that the proposed hybrid method leads in most cases to an overall efficiency of the process of solving systems of equations.

A first set of experiments was aimed at memetic algorithm convergence. There was a rapid convergence in cases when solutions of the system of equations are integer values. Otherwise, the procedure revolves around the solution but does not "catch". The next example will show this. Consider the system of equations:

$$\begin{cases} 2x + y + 3z = 13 \\ x + 5y + z = 14 \\ 3x + y + 4 = 17 \end{cases}$$

with solution vector X = {1,2,3}.

Each equation determines a plan (green, blue and brown) in the space R^3, as can be seen in Figure 2. The intersection of the three planes represents the system solution, represented by a red dot in figures. In Figures 2,3 and 4 are represented graphically the values generated (X_0) by memetic algorithm in case of three different executions, E_1, E_2 and E_3.

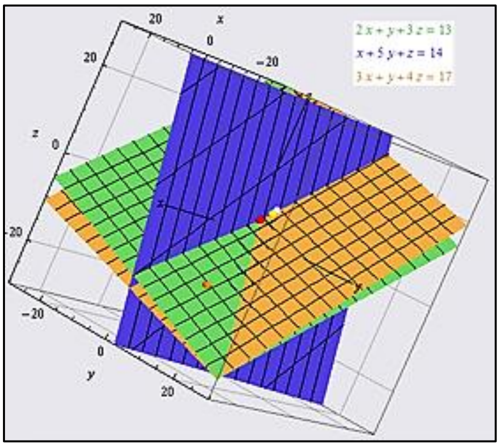

Figure 2 Execution E_1

E_1: after the first iteration is obtained X_0={19,10,-12} (orange dot) after the second iteration X_0={-2,2,5} (yellow dot) and after the third iteration X_0={-3,2,6} (white dot). The solution X={1,2,3} (red dot) was obtained in thr fourth iteration.

E_2: after the first iteration is obtained X_0={13,8,9} (yellow dot) and after the second iteration X_0={0,2,4} (white dot). The solution X={1,2,3} (red dot) was obtained in the third iteration

E_3 after the first iteration is obtained X_0={2,-3,3} (yellow dot) and after the second iteration X_0={5,2,0} (white dot). The solution X={1,2,3} (red dot) was obtained in the third iteration.

After this first set of first experiments it was observed that the memetic algorithm has a good convergence in the case of linear systems. The problem is when the system solution consists of real numbers. In this case, the new version (with integer numbers for chromosomes) can not converge to the exact solution. But this is not a problem, because in the proposed approach we are interested only in an approximating X_0, good enough to be used further by an iterative method.

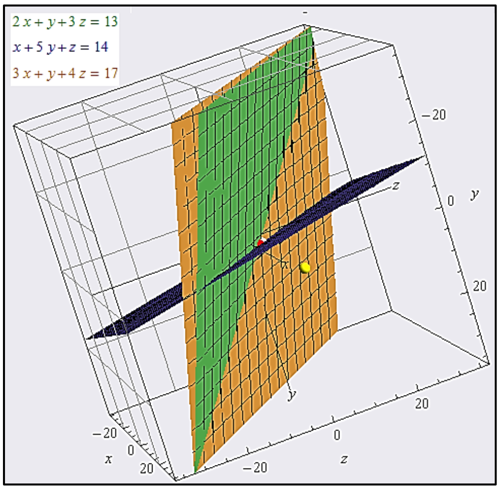

Figure 3 Execution E_2

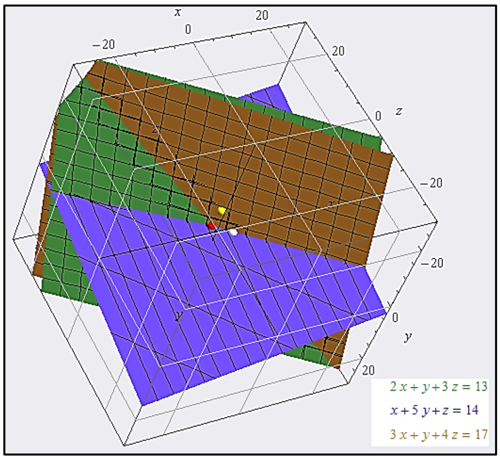

Figure 4 Execution E_3

A second set of experiments aimed the behavior of memetic algorithm for systems of nonlinear equations. We mention that the memetic variant with real numbers for chromosomes was used. More specifically, the values obtained by memetic algorithm were used as initial values X_0 for iterative methods, to see if they are in the basins of attraction of the iterative methods. Figures 5, 6 and 7 partially represent such an experiment, ie how is influenced the convergence of iterative methods by initial vector X_0 for the nonlinear equation $x^2+2y^2+2xy(x+1)=0$. The red dot represents the solution X and the black dot represents the initial vector X_0. On these figures it is mentioned the X_0 value and the number of iterations/steps required for each iterative method in part to reach solution from X_0 to X. Straight lines (yellow, green and blue) are the "routes" for convergence.

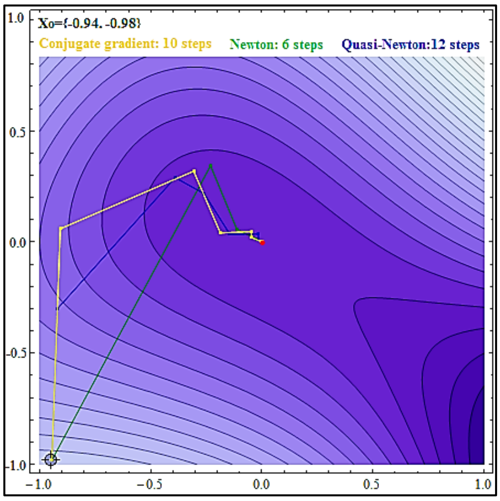

Figure 5

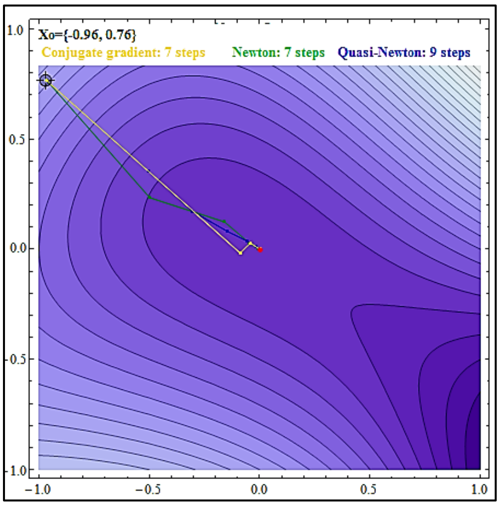

Figure 6

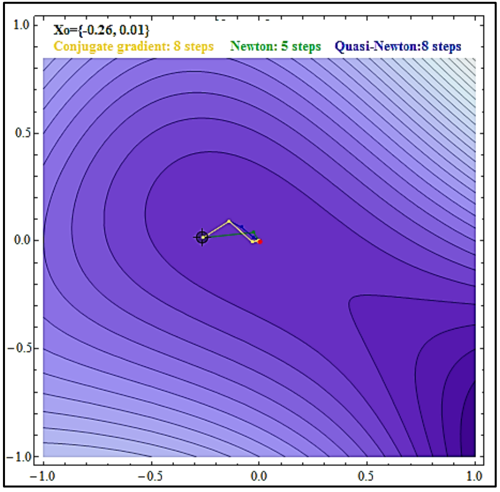

Figure 7

The third set of experiments studied the effect of memetic algorithm in the case of parallel variants of iterative methods. It was also observed the beneficial effect of memetic algorithm in improving the convergence of parallel methods. We mention that were used memetic variant with integer values for chromosomes. Some examples are given below. In Tables 1 and 2 are marked in red random values selected by the user for the initial vector X_0.

Example 1. The linear system of equations:

$$\begin{cases} 2x+y+3z=13 \\ x+5y+z=14 \\ 3x+y+4=17 \end{cases}$$

with solution vector $\{1,2,3\}$.

The imposed precision for the iterative method (parallel preconditioned conjugate gradient) was 10^{-15}.

Table 1 Experiments with parallel PCG, ex. 1

X_0 after first iteration	The iterations and the precision obtained	X_0 after second/third iteration	The iterations and the precision obtained
19,10,-12 (generated by memetic)	1.481866e-11 2.864289e-14 8.853316e-14	-2,2,5	6.342504e-12
		-3,2,6	4.885401e-13 3.972054e-15 7.944109e-15 5.329070e-15
		1,2,3 (the solution)	0
13,8,9 (generated by memetic)	1.551742e-11 1.776356e-15 3.87962e-14	0,2,4	1.685144e-12 1.776356e-15
		1,2,3	0
0,0,0 (random by user)	7.557119e-12 4.351167e-15		
100,100, 100 (random by user)	5.197698e-10 1.945901e-14 1.387379e-13 1.820224e-14 3.182604e-14 1.096459e-13 6.404745e-15 3.045817e-14 7.324106e-15 5.024295e-15 1.811535e-14 6.404745e-15 7.561523e-14		

Example 2. The linear system of equations:

$$\begin{cases} x + y + z = 102 \\ \quad 2y + 3z = 5 \\ x + 2y + z = 103 \end{cases}$$

with solution vector {100,1,1}.

The imposed precision for the iterative method (parallel preconditioned conjugate gradient) was 10^{-15}.

Table 2. Experiments with parallel PCG, ex. 2

X_0 after first iteration	The iterations and the precision obtained	X_0 after second/third iteration	The iterations and the precision obtained
89,29,-21 (generated by memetic)	1.421196e-13 1.776356e-15	92,13,10	3.465136e-13 2.309263e-14
		88,8,-2	3.074496e-13 2.568049e-14
		90,9,-3	2.207851e-13 1.421085e-14
		95,4,0	1.284024e-13 3.640448e-14 2.624262e-14 2.664535e-15
		96,0,2	2.241379e-13 7.105427e-15
		94,8,-4	1.128545e-13
		99,2,2	1.303425e-12 3.552713e-15
		98,1,3	2.287037e-13 1.517719e-14 1.675811e-14 3.019806e-14 3.197442e-14 1.421085e-14
0,0,0 (random by user)	2.469174e-12 3.256115e-14 2.363289e-14 5.329070e-15		
-1, 100000, 100000 (random by user)	5.556293e-9 2.934965e-12 1.620676e-13 2.411204e-14 5.464282e-14 4.726578e-14		

4. CONCLUSION AND FUTURE WORK

The main conclusion is that MAs can be very helpful in finding solution for system of equations when iterative methods like CG, PCG, Newton or Quasi-Newton, in serial and parallel cases are used.

It was observed that a single iteration in memetic algorithm is enough to determine a value X0, good for iterative methods, without moving the computing effort from the iterative method to the memetic method.

The proposed memetic algorithm is able to discover solutions/initial vectors inside or very close to attraction basins of iterative methods. This is a good/trusted alternative in comparison with that random chosen by user, the last being often wrong as can be seen in Tables 1 and 2.

The proposed hybrid method is simple to implement in iterative methods, but it still requires some further implementation refinements.

The hybrid schema described above can also be implemented on parallel machines by allotting different equations to different processors in memetic phase and this will be our preoccupation in the future.

5. REFERENCES

[1] F. Neri, C. Cotta, P. Moscato (Eds.), *Handbook of Memetic Algorithms*, Studies in Computational Intelligence, 2012 Springer-Verlag Berlin Heidelberg, ISBN 978-3-642-23246-6, 2012

[2] J. Brownlee, *Clever Algorithms: Nature-Inspired Programming Recipes*, 2011, ISBN:978-1-4467-8506-5, 2011

[3] M. Gendreanu, J.Y.Potvin, *Handbook of Metaheuristics*, Springer 2010, ISBN 978-1-4419-1663-1, 2010

[4] T. F. Gonzales, *Handbook of Approximation Algorithms and Metaheuristics*, Chapman&Hall/CRC 2007

[5] N. Krasnogor, *Studies in the Theory and Design Space of Memetic Algorithms*, PhD thesis, Univ. of the West of England, 2002

[6] W.E. Hart, N. Krasnogor,J.E. Smith, *Recent Advances in Memetic Algorithms*, Springer-Verlag Berlin Heidelberg, ISBN 3-540-22904-3, 2005

[7] L.O. Mafteiu-Scai, *Solving Linear Systems of Equations using a Memetic Algorithm*, , International Journal of Computer Applications (0975-8887), volume 58-No.13 November 2012

[8] Ya-Zhong Luo , Guo-Jin Tang, Li-Ni Zhou, *Hybrid approach for solving systems of nonlinear equations using chaos optimization and quasi-Newton method*, Elsevier, Applied Soft Computing, Volume 8 Issue 2, pp: 1068-1073 March, 2008

[9] M.R. Hestenes, E. Stiefel, *Methods of Conjugate Gradients for Solving Liniar Systems*, Journal of Research of the National Bureau of Standards, Vol. 49, No.6, December 1952

[10] J.O. Omolehin, M.K.A Abdulrahman, K. Rauf, *Conjugate gradient method for non-positive definite matrix operator*, African Journal of Mathematics and Computer Science Research Vol. 1(2), pp: 028-031, October, 2008

[11] R. Fletcher, *Conjugate gradient methods for indefinite sistems*, Springer, Numerical Analysis Lecture Notes in Mathematics Volume 506, pp: 73-89, 1976

[12] Michele Benzi, *Preconditioning Techniques for Large Linear Systems: A Survey*, Journal of Computational Physics 182**,** pp: 418–477, 2002

[13] A. Galantai, *The theory of Newton's method*, Journal of Computational and Applied Mathematics 124 (2000) pp 25-44, Elsevier, 2000

[14] Yixun Shi, *Modified Quasi-Newton Methods for Solving Systems of Linear Equations*, Int. J. Contemp. Math. Sci., Vol. 2, 2007, no. 15, pp: 737 – 744, 2007

[15] C.G. Broyden, *A Class of Methods for Solving Nonlinear Simultaneous Equations*, JSTOR vol. 19 no. 92 pp: 577–593, 1965, Mathematics of Computation, publ. By American Mathematical Society, 1965

[16] D.M. Gay, *Some convergence properties of Broyden's method*, SIAM Journal of Numerical Analysis (SIAM) vol. 16 no. 4, pp: 623–630, 1979

[17] Gene H. Golub, Michael L. Overton, *The convergence of inexact Chebyshev and Richardson iterative methods for solving linear systems*, Numerische Mathematik 1988, Vol. 53, Issue 5, pp 571-593, Springer 1988

[18] T.A. Manteuffel, *The Tchebyshev iteration for nonsymmetric linear systems*, Numerische Mathematik 1977, vol 28, 307–327 ,1977

[19] H. Woźniakowski *Numerical stability of the Chebyshev method for the solution of large linear systems*, Numerische Mathematik 1977, vol 28, 191–209 (1977)

[20] Al Dahoud Ali, Ibrahiem M. M. Emary, Mona M. Abd El-Kareem, *Application of Genetic Algorithm in Solving Linear Equation Systems*, MASAUM Journal of Basic and Applied Science, Vol.1, No.2 Sept. 2009

[21] Ikotun Abiodun M., Lawal Olawale N., Adelokun Adebowale P. *The Effectiveness of Genetic Algorithm in Solving Simultaneous Equations*, International Journal of Computer Applications (0975 – 8887) Volume 14– No.8, February 2011

[22] Crina Grosan, Ajith Abraham, *Multiple Solutions for a System of Nonlinear Equations*, International Journal of Innovative Computing, Information and Control ICIC International , 2008 ISSN 1349-4198,2008

[23] Ibrahiem M.M. El-Emary,Mona M. Abd El-Kareem, *Towards Using Genetic Algorithm for Solving Nonlinear Equation Systems*, World Applied Sciences Journal 5 (3): pp. 282-289, 2008, ISSN 1818-4952, 2008

[24] Crina Grosan, Ajith Abraham, *A New Approach for Solving Nonlinear Equations Systems*, IEEE Transaction on Systems, Man and Cybernetics-part A: Systems and Humans, vol. 38, no. 3, May 2008

[25] Yong Zhou, Huajuan Huang, Junli Zhang, *Hybrid Artificial Fish Swarm algorithm for Solving Ill-Conditioned Linear Systems of Equations*, ICICIS 2011 Proceedings, Part 1, Springer 2011, pp. 656-662, 2011

[26] Selim G. AkM, *The Design of Parallel and Analysis Algorithms*, ISBN 0-13-200056-3, Prentice-Hall, Inc., 1989

[27] S. Maruster, V. Negru, L.O. Mafteiu-Scai, *Experimental study on parallel methods for solving systems of equations*, IEEE CPS, Proc. of 14th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing, 2011

[28] T.J. Dekker, W. Hoffmann, K. Potma, *„Parallel algorithms for solving large linear systems”*, Journal of Comput. and Applied Mathematics 50 (1994) 221-232, Elsevier, 1994

[29] Tsu-Chien Cheu, C. Philip Johnson, Roy R. Craig Jr, *Computer algorithms for calculating efficient initial vectors for subspace iteration method*, Int**.** Journal for Numerical Methods in Engineering Vol**.** 24, **-** 10, pp**:** 1841–1848, October 1987

[30] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, SIAM 1995, ISBN:0-89871-352-8, 1995

[31] B.I. *Epureanu*, H.S. Greenside, *Fractal basins of attraction associated with a damped Newton's method*, SIAM Rev 1998, 40:102-109, 1998

[32] M. Scott, B. Neta, C. Chun, *Basin attractors for various methods*, Elsevier, Applied Mathematics and Computation 218 (2011) 2584–2599, 2011

[33] O. Cira, *Numerical experiments on attraction basin*, AMO Advanced modeling and optimization, vol. 2 no. 3, 2000