Applied Mathematical Sciences, Vol. 2, 2008, no. 55, 2699 - 2703

A New Technique for Solving Systems of Nonlinear Equations

J. Biazar [1] and B. Ghanbary

Department of Mathematics, Faculty of Sciences
University of Guilan,P.O. Box 1914, Rasht, Iran
biazar@guilan.ac.ir, b.ghanbary@yahoo.com

Abstract

The aim of this paper is to construct a new method for solving systems of nonlinear equations. The new method is based on the idea of Gauss-Seidel method, which is a known method for solving systems of linear equations, and considering any equation, in such a system of nonlinear equations, as an equation of one variable.

Keywords: Systems of non-linear equations; Gauss-Seidel method

1 Introduction

Solving systems of nonlinear equations is of great importance, because these systems frequently arise in many branches of computational mathematics. Consider the following systems of nonlinear equations:

$$\begin{aligned} &f_1(x_1, x_2, ..., x_n) = 0,\\ &f_2(x_1, x_2, ..., x_n) = 0,\\ &\vdots\\ &f_n(x_1, x_2, ..., x_n) = 0. \end{aligned}$$

This system can be referred by $F(X) = 0$, where $F = (f_1, f_2, ..., f_n)^t$ and $X = (x_1, x_2, ..., x_n)^t$ from the n-dimensional space $\Re^n$ into $\Re$ .We assume that the system (1) admits a unique solution.

2 New method

J.H. He reached some iteration formulae via general Lagrange multiplier [1], for solving nonlinear equations $f(x) = 0$, with initial approximation x_0 for r ,

[1] Corresponding author

the simple root of equation, i.e.$f(r) = 0$ and $f^{'}(r) \neq 0$. One of these formulae is as the following, which we have used in this paper

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^{''}(x_n)f^2(x_n)}{2f'^3(x_n)} \quad (1)$$

This formula is applied to solve equations, with one variable which are obtained by the following procedure. Suppose that $x^{^0} = (x_1^0, x_2^0, ..., x_n^0)$ be an initial approximation to the solution of the system (1), let for each $i, 1 \leq i \leq n$, $f_i(x_i^1) = f_i(x_1^1, ...x_{i-1}^1, x_i^1, x_{i+1}^0, ..., x_n^0) = 0$ be an equation of one variable,x_i^1 . We solve $f_i(x_i^1) = 0$, with initial approximation x_i^0 via (2), to obtain x_i^1 by a few iterations. Continuing this procedure the values of $x_1^1, x_2^1, ..., x_n^1$ would be determined and $x^1 = (x_1^1, x_2^1, ..., x_n^1)$ would be the new approximation for the solution of $F(X) = 0$. Similarly for $j, j \geq 1$ (as iteration level) and $i, 1 \leq i \leq n$, define

$$f_i(x_i^{j+1}) = f_i(x_1^{j+1}, ...x_{i-1}^{j+1}, x_i^{j+1}, x_{i+1}^j, ..., x_n^j) = 0. \quad (2)$$

Which are n equations with one-variable,x_i^{j+1} . Solving these equations via (2), with initial approximation x_i^j , we obtain new approximation $x^{j+1} = (x_1^{j+1}, x_2^{j+1}, ..., x_n^{j+1})$ for the system (1).

3 Numerical examples

In this part, we consider some examples which are solved in [2]. The results of Newton and Adomian's decomposition methods for these examples are presented here for comparison purposes. The results and expected solutions for each example have been presented on the Tables 1 and 2.

Example 1. Consider the following system of non-linear equations:

$$\begin{aligned} x_1^2 - 10x_1 + x_2^2 + 8 = 0 \\ x_1x_2^2 + x_1 - 10x_2 + 8 = 0 \end{aligned}$$

The exact solution is $X = (1, 1)^t$.

First, we apply the new method to solve this Example with the initial approximation $(x_1^0, x_2^0)' = (0.8, 0.8)$. Continuing the procedure of the method that was illustrated yields the following equations:

$$\begin{aligned} f_1(x_1^1) = (x_1^1)^2 - 10(x_1^1) + 8.64 \\ f_2(x_2^1) = 0.8(x_2^1)^2 - 10(x_2^1) + 8.8 \end{aligned}$$

$$f_1(x_1^2) = (x_1^1)^2 - 10(x_1^1) + 8.907324032$$
$$f_2(x_2^2) = 0.9551452327(x_2^2)^2 - 10(x_2^2) + 8.955145233$$

$$f_1(x_1^3) = (x_1^1)^2 - 10(x_1^1) + 8.977970344$$
$$f_2(x_2^3) = 0.9884311059(x_2^3)^2 - 10(x_2^3) + 8.988431106$$

$$f_1(x_1^4) = (x_1^1)^2 - 10(x_1^1) + 8.994242560$$
$$f_2(x_2^4) = 0.9972472189(x_2^4)^2 - 10(x_2^4) + 8.997247219$$

$$f_1(x_1^5) = (x_1^1)^2 - 10(x_1^1) + 8.998625146$$
$$f_2(x_2^5) = 0.9992803845(x_2^5)^2 - 10(x_2^5) + 8.999280384$$

The solution of these equations via (2), with six significant digits, is presented in Table 1.

Example 2. Let's solve the following system of non-linear equations:

$$15x_1 + x_2^2 - 4x_3 = 13,$$
$$x_1^2 + 10x_2 - \exp(-x_3) = 11,$$
$$x_2^3 - 25x_3 = -22$$

With the exact solution $X = (1.042149561, 1.031091272, 0.9238481549)^t$.
Following the procedure of the method with
$(x_1^0, x_2^0, x_3^0)^t = (0.86666667, 1.10000002, 0.88000000)^t$ yields to following equations:

$$f_1(x_1^1) = 15(x_1^1)^2 - 15.31(x_1^1)$$
$$f_2(x_2^1) = 10(x_2^1) - 10.40529900$$
$$f_3(x_3^1) = -25(x_3^1)^2 + 23.12658430(x_3^1)$$

$$f_1(x_1^2) = 15(x_1^2)^2 - 15.61647252(x_1^2)$$
$$f_2(x_2^2) = 10(x_2^2) - 10.31294542$$
$$f_3(x_3^2) = -25(x_3^2)^2 + 23.09685232(x_3^2)$$

$$f_1(x_1^3) = 15(x_1^3)^2 - 15.63192795(x_1^3)$$
$$f_2(x_2^3) = 10(x_2^3) - 10.31094626$$
$$f_3(x_3^3) = -25(x_3^3)^2 + 23.09621457(x_3^3)$$

$$f_1(x_1^4) = 15(x_1^4)^2 - 15.63223820(x_1^4)$$
$$f_2(x_2^4) = 10(x_2^4) - 10.31091327$$
$$f_3(x_3^4) = -25(x_3^4)^2 + 23.09620405(x_3^4)$$

$$f_1(x_1^5) = 15(x_1^5)^2 - 15.63224333(x_1^5)$$
$$f_2(x_2^5) = 10(x_2^5) - 10.31091273$$
$$f_3(x_3^5) = -25(x_3^5)^2 + 23.09620388(x_3^5)$$

The results of solving these equations via (2), with eight significant digits, are shown in Table 2.

Table 1. The results of different methods for example 1.

Iteration	Method	x_1	x_2
1	New method	0.95525031	0.98895007
	Newton method	0.98166470	0.99171726
	2	0.92800002	0.93120001
2	New method	0.99725389	0.99931400
	Newton method	0.99901628	0.99994708
	2	0.9694202	0.96898561
3	New method	0.9998256	0.99995714
	Newton method	0.99999999	0.99999999
	2	0.98551298	0.98462127
4	New method	0.99998928	0.99999732
	Newton method	1.00000000	1.00000000
	2	0.99263442	0.99191131
5	New method	0.9999993	0.99999998
	Newton method	1.00000000	1.00000000
	2	0.99607595	0.99556087

Table 2.The results of different methods for example 2.

Iteration	Method	x_1	x_2	x_2
1	New method	1.02066666	1.03730224	0.92464532
	Newton method	1.11364025	1.011779832	1.01887013
	2	1.02066667	1.06636720	0.93324000
2	New method	1.04150568	1.03119379	0.92386123
	Newton method	1.04305725	1.10314838	0.91687004
	2	1.03979682	1.02246038	0.92835651
3	New method	1.04213895	1.03109296	0.9238437
	Newton method	1.04230666	1.03103551	0.92440087
	2	1.04485749	1.02748608	0.92202779
4	New method	1.04214938	1.03109130	0.92384815
	Newton method	1.04221301	1.03109806	0.92377882
	2	1.04223720	1.01570328	0.92314446
5	New method	1.04214955	1.03109127	0.92384815
	Newton method	1.04215194	1.03109043	0.92385665
	2	1.04223716	1.01568782	0.92314446

4 Conclusion

In this paper an efficient iterative method was built up to solve systems of nonlinear equations. As it can been seen in Tables 1 and 2, Newton method for example 1 converges rapidly to exact solution. But as the number of equations and variables increases in example 2, the efficiency of Newton method decreases because of the huge computations needed. This new method seems to be very easy to employ with reliable results. The computations associated with examples were performed via Maple10.

References

[1] J.H. He, Non-perturbative method for strongly nonlinear equations, dissertation. de-Verlag im Internet Gmbh, Berlin, 2006.

[2] E. Babolian, J. Biazar, A.R. Vahidi, Solution of a system of nonlinear equations by Adomian decomposition method, Applied Mathematics and Computation. 150 (2004) 847-854.

Received: December 8, 2007