A New Solution Approach for Non-Linear Equation Systems with Grey Wolf Optimizer

Pakize Erdogmus[1]

[1]Corresponding Author; Duzce University, Computer Engineering Dept; Duzce, Turkey;
pakizeerdogmus@duzce.edu.tr ; +90 533 542 97 28

Abstract

The aim of this study is to bring a new perspective for the solutions of non-linear equation systems. So this study handles the non-linear equation systems as a constrained optimization problem, while generally is handled unconstrained optimization problem or multi objective optimization problem. The object is to minimize the sum of the squares of nonlinear equations under the nonlinear equality constraints. A recently developed heuristic optimization algorithm called Grey Wolf Optimizer (GWO) is proposed for the solution of nonlinear equation systems. Two results were obtained. Firstly, it has been seen that GWO can be an alternative solution technique for the solution of nonlinear equation systems. Secondly, modelling the systems of nonlinear equations as constrained optimization gives better results.

1. Introduction

In general, it can be usually found solutions to a system of equations when the number of unknown independent variables matches the number of equations. If the equations are linear, then the solution of the system is calculated with some methods based on matrix operations. But if the equations are non-linear, the solution of the system is not always a simple task as linear systems.

Nonlinear equation systems arise in many areas such as engineering, mechanics, medical science, chemistry and robotics [1]. Financial market analysis and network flow analysis are some applications of nonlinear equation systems [2].

Sometimes even a single nonlinear equation can be difficult to solve and it requires a lot of iteration and time. The success of the solution depends on the initial guess of the solution and the method used for the solution. The solution methods are divided into two main categories direct methods and iterative methods. Several iterative methods have been developed to solve the nonlinear system of equations F(x) = 0, by using essentially Taylor's polynomial, decomposition, homotopy perturbation method, quadrature formulas and other techniques [3]. Iterative methods are divided into two groups called interval methods and continuation methods. Interval methods guarantee solution because of the fact that it starts a solution with two initial values which the solution exists between the two values. Sometimes the convergence of these methods can be slow. Bisection, Regula-Falsi and Secant method are called as interval methods. The open methods start only one initial value and converges the solution rapidly since they use derivative. Newton-Raphson is an open method and used for the solution of both nonlinear equation and nonlinear equation systems [4].

A lot of numerical methods like Newton methods have been proposed for the solution of nonlinear equation systems. But these methods have some disadvantages such as they require derivative information, a good initial starting point and easily trapped in local minima [1]. These methods also find only one solution for the problem.

Besides numerical methods, some solution approaches are proposed in the literature. Nonlinear equation systems have been modelled as multi objective optimization problems[4], [5]. The advantage of modelling the nonlinear equation systems as multi objective optimization problem is that the user can find alternative solutions instead of finding only one solution. It is desirable to find all the

solutions for a given systems in real-life applications, since it presents alternatives for decision maker [5]. Some heuristic optimization algorithms have also been used for the solutions of the nonlinear equation systems. Genetic algorithms [2], parallel imperialist competitive algorithm [6], weed optimization algorithm [7], particle swarm optimization [8], immune genetic algorithm [9], leader glowworm swarm optimization [10] and hybrid social emotional optimization algorithm [11] have been used for the solution of the nonlinear equation systems.

GWO, one of the recent nature-inspired algorithms, was applied successfully for the solutions of lot of engineering problems from Electrical Engineering to machine learning. Training multilayer percepteron [12], optimal reactive power dispatch [13], economic emission dispatch [14], classification [15], optimal power flow [16], hyperspectral band selection [17], energy loss minimization [18], system identification and filter design [19], feature selection [20] are some of the engineering applications using GWO.

In this study, there are two main contributions. One of them is to propose a handling mechanism for the solutions of nonlinear equations systems as an optimization problem. Systems were solved both as an unconstrained and constrained problem. The results were compared and discussed. The second one is that upon our research, GWO has not been used for the solution of the systems of nonlinear equations. So in this study, it has been aimed to use GWO, and present its performance for the solution of the systems of nonlinear equations. Although many nature inspired optimization algorithms have been developed in recent years, GWO has been chosen especially among these algorithms. As far as it has been seen, GWO is quite successful for the solution of continuous optimization problems.

This article is organized as follows. Section 2, presents the Nonlinear Equation Systems, Section 3 presents GWO. The solutions of some nonlinear equation systems with GWO are presented in Section 4. Section 5 concludes the achievements of the study and gives some ideas for future studies.

2. Nonlinear Equation Systems

Nonlinear equation systems are given in Equation 1.

$$f_1(x_1, x_2, \ldots, x_n) = 0,$$

$$f_2(x_1, x_2, \ldots, x_n) = 0,$$

$$\ldots\ldots\ldots\ldots\ldots\ldots$$

$$f_n(x_1, x_2, \ldots, x_n) = 0 \tag{1}$$

f_1, f_2 and f_n are nonlinear functions dependent to the variables x_1, x_2 and x_n. Some classical and heuristic solution methods have been proposed for the solution of the nonlinear equations system in the literature. Classical methods are generally use the initial estimates and try to solve the system with some iteration. The success of such algorithms like Newton is dependent to initial estimates. Since most of the classical methods require derivative and matrix operations, the complexity of the solution for a nonlinear equation system increases with the number of independent variable and the number of equations. So some heuristic methods as stated introduction are used for the solution of nonlinear equation systems.

The Nonlinear Equation Systems have been converted to an optimization problem with different ways. In some studies, nonlinear equation systems have been modelled and solved as unconstrained single objective optimization problem. In such problems, the sum of the squares of nonlinear equations is accepted as the objective function to be minimized. So this solution approach doesn't guarantee that

the total error is distributed to the each equation in the system, but it finds a solution making total error minimum.

In some other studies, each nonlinear equation has been accepted as an objective function, and nonlinear equation systems are modeled and solved as multi objective optimization problem. This approach evaluates each equation as an objective function to be minimized.

A new approach for the solution of Nonlinear Equation Systems is used in this study. Nonlinear equation systems have been accepted as constrained single objective optimization problems. It has been aimed to distribute the total error to each nonlinear equation equally. This formulation is given in Equation 2.

$$\text{Objective function}: fmin = \sum_{i=1}^{n} f_i (x_1, x_2, \ldots, x_n)^2 \tag{2}$$

$$\text{Constraints}: f_1(x_1, x_2, \ldots, x_n) = 0,$$
$$f_2(x_1, x_2, \ldots, x_n) = 0,$$
$$\ldots \ldots$$
$$f_n(x_1, x_2, \ldots, x_n) = 0$$

But since all the heuristic optimization algorithms find only unconstrained minimum value of an object function, the constrained optimization problem is converted to unconstrained optimization problem by penalizing the objectives.

Since the problem is continuous optimization variables, and since the possibility of creation a continuous value with random number is zero, a tolerance value must be specified for the equality constraints. So the penalty function is given in Equation 3 and 4.

$$P = \sum_{i=1}^{n} R_i \max(0, f_i((x_1, x_2, \ldots, x_n)^2 - tol)^2 \tag{3}$$

The fitness function is given in Equation 4.

$$F_{fitnes} = f_{min} + P \tag{4}$$

In this study, the test problems were solved with both method given in Equation 2 as unconstrained and with Equation 4 as constrained optimization problem. The system of nonlinear equations used in this study was given in Table 1. The problems were selected from some articles[1,5].

3. Grey Wolf Optimizer (GWO)

Grey Wolf Optimizer was firstly developed by Mirjalili in 2014[21] GWO is one of the nature inspired optimization algorithm developed recently. GWO simulates the hunting behavior of Grey Wolves for finding near optimal solution to optimization problems. GWO is one of the population based optimization algorithms. GWO was originally developed for the solution of continuous optimization problems. But it was also applied for the solutions of discrete optimization problems [22]. In order to increase the performance of the GWO, it was combined with other heuristic algorithms such as genetic operators [23,24], particle swarm [25], simulated annealing[26] and differential evolution[27].

Grey Wolves live as social groups. There is a hierarchy among them. So they are classified as alpha, beta, delta and omega. The wolf has some responsibility according to its level in the hierarchy. And their hunting strategy has three main phases. In the first phase, grey wolves track the prey. In the second phase, they encircle the prey until the prey stops moving and at last attack towards the prey. GWO mimics this hunting strategy for finding the optimal solution of an optimization problem.

Table 1. The Systems of Nonlinear Equations used in this study for testing

Test problem Name and Reference	The system of Nonlinear Equation	The number of independent variable	Decision Space	The number of optimal solution
P1[5]	$x_1^2+x_2^2-1=0$ $x_1-x_2=0$	2	$[-1,1]^2$	2
P2[5]	$\sin(5\pi x_1)-x_2=0$ $x_1-x_2=0$	2	$[-1,1]^2$	11
P3[5]	$\cos(4\pi x_1)-x_2=0$ $x_1^2+x_2^2-1=0$	2	$[-1,1]^2$	15
P4[1]	$x_1^2+2x_2^2+\cos(x_3)-x_4^2=0$ $3x_1^2+x_2^2+\sin^2(x_3)-x_4^2=0$ $-2x_1^2-x_2^2-\cos(x_3)+x_4^2=0$ $-x_1^2-x_2^2-\cos^2(x_3)+x_4^2=0$	4	$[-2,2]^4$	2
P5[1]Interval Aritmethic Benchmark	$x_1-0.25428722-0.18324757x_4x_3x_9=0$ $x_2-037842197-0.16275449x_1x_{10}x_6=0$ $x_3-0.27162577-0.16955071x_1x_2x_{10}=0$ $x_4-0.19807914-0.15585316x_7x_1x_6=0$ $x_5-0.44166728-0.19950920x_7x_6x_3=0$ $x_6-0.14654113-0.18922793x_8x_5x_{10}=0$ $x_7-0.42937161-0.21180486x_2x_5x_8=0$ $x_8-0.07056438-0.17081208x_1x_7x_6=0$ $x_9-0.34504906-0.19612740x_{10}x_6x_8=0$ $x_{10}-0.42651102-0.21466544x_4x_8x_1=0$	10	$[-2,2]^{10}$	?
P6[1]Neuro-physology Application	$x_1^2+x_3^2=1$ $x_2^2+x_4^2=1$ $x_5 x_3^3+x_6 x_4^3=c_1$ $x_5 x_1^3+x_6 x_2^3=c_2$ $x_5 x_1 x_3^2+x_6 x_4^2 x_2=c_3$ $x_5 x_1^2 x_3+x_6 x_2^2 x_4=c_4$ $c_i=0, i=1,..4$	6	$[-1,1]^6$	?
P7[1] Chemical Equilibrium	$x_1 x_2 +x_1-3x_5 =0$ $2x_1x_2+x_1+x_2 x_3^2 +R_8x_2 - Rx_5+ 2R_{10}x_2^2+R_7 x_2 x_3+ R_9 x_2 x_4=0$ $2x_2 x_3^2 + 2R_5 x_3^2 -8 x_5 +R_6 x_3 +R_7 x_2 X_3=0$ $R_9 x_2x_4+2x_4^2 – 4Rx_5=0$ $x_1(x_2 +1)+ R_{10} x_2^2 + x_2 x_3^2+ R_8 x_2 +R_5 x_3^2 +x_4^2-1 +R_6 x_3 + R_7 x_2x_3+ R_9x_2x_4=0$ R values can be found ref[1].	5	$[-40,40]^5$	?

Prey represents the optimal solution while each wolf represents a solution. While real wolves change their position in such a way that they encircle and attack the prey, random initial solutions changes their position from one iteration to another in such a way that they converges optimal solution.

After initial positions for each wolf have been created and fitness function values for each wolf have been evaluated, the best three solutions have been chosen as alpha, beta and delta. And the rest of the wolves called delta, tracks the best three of them, assuming that the best three solutions have encircled

the prey. So the other wolves update their positions, according to the average of three updated positions.

For the encircling behavior of wolves, the following equations are proposed:

$$\vec{D} = \left| \vec{C}.\overrightarrow{X_p}(t) - \vec{X}(t) \right| \tag{5}$$

$$\vec{X}(t+1) = \overrightarrow{X_p}(t) - \vec{A}\vec{D} \tag{6}$$

X represents the position of grey wolf; Xp represents the position of prey, $\vec{A}$ and $\vec{C}$ are the coefficients, t represents current iteration and t+1 represents the next iteration. Equation 5 and 6 are generic equations. Since it has not been known what is the position of best solution(prey), these equations are applied for best three solutions as given in 9-14. The real updated position is the average of the distances among three best solution.

$\vec{A}$ and $\vec{C}$ are given in equation 7 and 8.

$$\vec{A} = 2\vec{a}.\overrightarrow{r_1} - a \tag{7}$$

$$\vec{C} = 2\vec{r_2} \tag{8}$$

a is a coefficient which is linearly decreased from 2 to 0, $\overrightarrow{r_1}$ and $\vec{r_2}$ are the random numbers between 0-1. Evenif the real wolves see their prey, since it has not got any idea about the optimum point(prey) in the solution space, the hunting behavior is simulated with the three best solution found among the wolves. So the following equations are proposed for hunting behavior:

$$\overrightarrow{D_\alpha} = \left| \vec{C_1}.\overrightarrow{X_\alpha} - \vec{X} \right| \tag{9}$$

$$\overrightarrow{D_\beta} = \left| \vec{C_2}.\overrightarrow{X_\beta} - \vec{X} \right| \tag{10}$$

$$\overrightarrow{D_\delta} = \left| \vec{C_3}.\overrightarrow{X_\delta} - \vec{X} \right| \tag{11}$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}(\overrightarrow{D_\alpha}) \tag{12}$$

$$\overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}(\overrightarrow{D_\beta}) \tag{13}$$

$$\overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}(\overrightarrow{D_\delta}) \tag{14}$$

$$\vec{X}(t+1) = \frac{\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3}}{3} \tag{15}$$

The success of the nature inspired algorithms is dependent to the equilibrum between exploration and exploitation process. So, in the first steps of the algorithm, in order to explore more efficient solutions the diameter of $\overrightarrow{A_1}$, $\overrightarrow{A_2}$ and $\overrightarrow{A_3}$ are big enough for exploration as it has been seen in equation 16. This means that wolves track the prey only fort he time being.

Encircling prey is simulated decreasing the value of a. a is calculated with a formula given in equation 16. Since $\overrightarrow{A_1}$, $\overrightarrow{A_2}$ and $\overrightarrow{A_3}$ are dependent to a, related to a, they also decrease. So the other wolves converges to the best three solutions(prey). At last iteration, since iteration is the max_iteration a, will be zoro. This result simulates attacking prey.

a=a(1-iteration/max_iteration) (16)

Upon our knowledge, this is the first time that GWO is being applied to the solution of the systems of nonlinear equations.

4. The Solutions of the Systems of Nonlinear Equations with GWO

In this study, some systems of nonlinear equations are taken form literature were selected as testing for the performance of GWO on nonlinear equations systems. So the problems firstly were solved with Trust-Region Method. The number of independent variables, the number of iterations and solution values of independent variables and equations are given in Table 2. Since some systems of nonlinear equations have more than one solution, the average x values are not given.

Table 2. The solutions of systems of Nonlinear Equations with Trust-Region

The problem no	The number of variable	The number of iteration	Solution
P1	2	4	$x=(0.707107,0.707107)$ $Eq_1=0$ $Eq_2=2.01E-7$
P2	2	3	$x=(0.562006,0.562006)$ $Eq_1=-1.2E-8$ $Eq_2=0$
P3	2	4	$x=(0.909178,0.416408)$ $Eq_1=-4.9E-8$ $Eq_2=2.56E-7$
P4	4	44	$x=(-9.6E-08,-2.8E-09,1.394741,0.5349)$ $Eq_1=-0.11097$ $Eq_2=0.683206$ $Eq_3=-0.46126$ $Eq_4=0.110971$
P5	10	3	$x=(0.257833,0.381097,0.278745,0.200669,0.445251,$ $0.149184,0.43201,0.073403,0.345967,0.427326)$ $Eq_1=1.88E-11$, $Eq_2=-3E-14$,$Eq_3=-3.7E-14$, $Eq_4=-2E-13$, $Eq_5=1.09E-11$, $Eq_6=2.06E-12$ $Eq_7=4.17E-13$, $Eq_8=-2.2E-13$, $Eq_9=-1.3E-12$ $Eq_{10}=-2.3E-12$
P6	6	5	$x=(0.754488, 0.954881, 0.656314, 0.296989, -1.4E-11,4E-07)$ $Eq_1=6.88E-15$ $Eq_2=3E-09$ $Eq_3=1.05E-08$ $Eq_4=3,48E-07$ $Eq_5=3,37E-08$ $Eq_6=1,08E-07$
P7	5	16	$X=(0.003114,34.59792,0.065042,0.859378,0.036952)$ $Eq_1=3.25E-12$,$Eq_2=7.71E-12$,$Eq_3=2.34E-12$, $Eq_4=-1.1E-14$,$Eq_5=4.42E-12$

After the problems were solved using classical method, the systems of nonlinear equations were handled as unconstrained optimization problem using the objective function. The objective function is the sum of the equations as given in Equation 2. After the problems were solved as unconstrained problem, fitness functions were handled using penalty function. Solutions with both methods were found with GWO, Particle Swarm Optimization (PSO) and Genetic Algorithm (GA). The parameters used for the simulation were given in Table 3. Simulations were made with a computer using Intel® Core™ i5 3230M CPU 2.6GHz processor and Microsoft Windows 7 operating system. The average(Ave) and standard deviation(Std) of the solutions were presented in Table 4.

Table 3. GWO, PSO and GA parameters

Algorithm	Parameters	Values
GWO	Search Agent Number	30

	Maximum iteration	1500
	A	2 to 0 decreased linearly
PSO	Swarm Size	30
	Maximum iteration	1500
	Inertia Range	0.1-1.1
	SelfAdjustmentWeight	1.49
	SocialAdjustmentWeight	1.49
GA	Population Size	30
	Maximum Generation	1500
	CrossoverFraction	0.8
	EliteCount	10
	Selection	Roulette

Table 4. The solution of P1 with GWO. PSO and GA

Model	P1	GWO		PSO		GA	
		Ave	**Std**	**Ave**	**Std**	**Ave**	**Std**
Unconstrained	Eq_1	-2.1E-05	5.54E-05	-0.97959	0.142857	-0.25988	0.448116
	Eq_2	-1.3E-05	7.39E-05	3.43E-14	2.4E-13	-0.1148	0.368058
	Sum	0.000107	5.05E-05	0.979592	0.142857	0.716682	0.288713
	Sure	1.274965	0.007189	0.029086	0.023637	0.208054	0.080656
Constrained	Eq_1	**5.7E-05**	5.7E-05	-0.49971	1.1E-07	-0.03675	0.412518
	Eq_2	**6.01E-05**	6.01E-05	-0.70731	7.74E-08	-0.17288	0.491179
	Sum	**5.72E-05**	5.72E-05	0.75	6.35E-11	0.434269	0.582274
	Sure	0.014116	0.014116	0.030721	0.021259	0.187856	0.069329

Table 5. The solution of P2 with GWO. PSO and GA

Model	P2	GWO		PSO		GA	
		Ave	**Std**	**Ave**	**Std**	**Ave**	**Std**
Unconstrained	Eq_1	-0.00022	0.000983	0	0	-0.19308	0.586802
	Eq_2	5.39E-05	0.000874	0	0	0.152401	0.37522
	Sum	0.000627	0.001327	0	0	0.849918	0.447619
	Sure	1.28619	0.01385	0.026712	0.01272	0.202238	0.083861
Constrained	Eq_1	-0.0001	0.000869	**0**	**0**	-0.20295	0.366526
	Eq_2	-0.00014	0.000645	**0**	**0**	0.045108	0.406401
	Sum	1.2E-06	3.31E-06	**0**	**0**	0.342726	0.462044
	Sure	0.963405	0.018266	0.023357	0.02122	0.174822	0.053142

Table 6. The solution of P3 with GWO. PSO and GA

Model	P3	**GWO**		**PSO**		**GA**	
		Ave	**Std**	**Ave**	**Std**	**Ave**	**Std**
Unconstrained	Eq_1	**1.49E-05**	**0.000206**	0.877551	0.331201	-0.11354	0.609738
	Eq_2	**-6.7E-06**	**0.000259**	-0.89668	0.282167	0.017597	0.733833
	Sum	**1.08E-07**	**3.56E-07**	1.774235	0.611899	0.535718	0.383857
	Sure	1.284663	0.00725	0.030963	0.021649	0.215209	0.066759
Constrained	Eq_1	2.38E-05	0.000207	-1.6E-09	1.57E-08	-0.08706	0.468391
	Eq_2	-6.3E-05	0.000382	-1.6E-09	2.25E-08	-0.06475	0.516686
	Sum	**1.89E-07**	9.72E-07	7.41E-16	4.69E-15	0.488202	0.516362
	Sure	0.979166	0.017728	0.072056	0.131582	0.200041	0.102071

Table 7. The solution of P4 with GWO. PSO and GA

Model	P4	GWO		PSO		GA	
		Ave	**Std**	**Ave**	**Std**	**Ave**	**Std**
Unconstrained	Eq_1	8.43E-06	3.24E-05	-0.00014	0.000999	0.098736	0.772846
	Eq_2	1.000012	0.000138	0.999566	0.002371	0.934627	1.601345
	Eq_3	-0.00017	0.000206	-0.00072	0.003793	-3.01563	3.391034
	Eq_4	5.81E-05	7.21E-05	0.000143	0.000999	0.745177	0.850028
	Sum	1.000258	0.000164	1.000573	0.003308	5.410317	4.753821
	Sure	1.329889	0.042571	0.090337	0.022326	0.41724	0.18847
Constrained	Eq_1	**-0.10998**	**9.34E-05**	0.749999	6.28E-06	-0.39997	0.289604
	Eq_2	**0.683386**	**8.03E-05**	-0.25	6.28E-06	0.552223	0.418849
	Eq_3	**-0.4615**	**0.000124**	-1.25	6.28E-06	-0.51954	0.857206
	Eq_4	**0.10999**	**9.55E-05**	-0.75	6.28E-06	0.605544	0.29847
	Sum	**0.704194**	**2.13E-05**	2.75	1.38E-09	2.162616	1.672967
	Sure	0.947775	0.011986	0.121347	1.4E-17	0.269673	0.135641

Table 8. The solution of P5 with GWO. PSO and GA

Model	P5	GWO		PSO		GA	
		Ave	**Std**	**Ave**	**Std**	**Ave**	**Std**
Unconstrained	Eq_1	0.211065	0.035169	0.084407	1.33E-09	-0.26078	1.037777
	Eq_2	-0.00249	0.000132	-0.12885	5.27E-08	-0.52251	0.739745
	Eq_3	1.64E-05	0.000137	7.29E-10	5.61E-08	0.311641	0.305518
	Eq_4	-0.00344	0.034249	-4.3E-09	6.44E-08	0.334334	0.392504
	Eq_5	-0.00343	0.000134	-3.1E-09	5.12E-08	0.740494	0.612229
	Eq_6	-0.003	0.029341	-7E-10	5.38E-08	0.792778	1.449963
	Eq_7	-0.00299	0.000124	1.02E-09	4.29E-08	0.595109	0.228063
	Eq_8	-0.00807	0.03056	-1.7E-09	2.89E-08	0.149612	0.369496
	Eq_9	-0.00811	0.00012	-1E-09	6.3E-08	0.157793	0.518176
	Eq_{10}	-4.4E-06	9.9E-05	3.95E-09	4.79E-08	0.548074	0.7509
	Sum	0.017619	0.062108	0.128849	2.2E-07	4.444123	1.756659
	Sure	0.706921	0.03573	0.261802	0.033448	4.273988	0.131444
Constrained	Eq_1	**-7.2E-05**	0.000217	-0.25429	1.12E-16	-0.67827	0.214738
	Eq_2	**-6.8E-06**	0.00015	-0.37842	5.61E-17	-0.6815	0.196881
	Eq_3	**3.92E-06**	0.000135	-0.27163	0	-0.39389	0.236042
	Eq_4	**-2.8E-06**	0.000178	1.03E-06	5.72E-06	-0.17364	0.043134
	Eq_5	**2E-05**	0.000115	-8.3E-07	5.31E-06	-1.57352	0.752772
	Eq_6	**-0.00019**	0.000427	-5.1E-07	4.19E-06	-1.2027	0.560146
	Eq_7	**-2.5E-05**	0.000145	4.81E-07	5.27E-06	0.32745	0.118963
	Eq_8	**-0.02084**	0.033294	1.05E-07	4.95E-06	-0.6115	0.263702
	Eq_9	**7.08E-06**	0.000133	2.93E-07	8.05E-06	0.156045	0.152697
	Eq_{10}	**4.5E-06**	0.000159	7.82E-07	4.34E-06	-1.59127	1.123785
	Sum	**0.00152**	0.002535	0.281646	5.1E-10	10.41383	2.051221
	Sure	1.101325	0.013817	0.112471	0.049463	0.167137	0.080274

Table 9. The solution of P6 with GWO. PSO and GA

Model	P6	GWO		PSO		GA	
		Ave	**Std**	**Ave**	**Std**	**Ave**	**Std**
Unconstrained	Eq_1	1.46E-06	3.73E-06	-0.02041	0.142855	-0.05122	0.06901
	Eq_2	1.17E-06	3.78E-06	-2E-07	1.32E-06	0.123655	0.282989
	Eq_3	9.74E-08	5.27E-07	4.34E-07	2.67E-06	-0.34251	0.230471
	Eq_4	-4E-07	1.9E-06	-7.5E-07	4.37E-06	-0.18574	0.21498
	Eq_5	7.33E-08	4.56E-06	1.72E-06	1.23E-05	-0.14785	0.235251
	Eq_6	1.3E-06	4.81E-06	8.55E-06	5.67E-05	-0.26954	0.229667
	Sum	6.37E-06	1.03E-05	0.02042	0.142854	1.458507	0.664296

	Sure	0.887614	0.024446	0.123561	0.096898	0.166033	0.089815
Constrained	Eq$_1$	3.36E-06	9.32E-06	**-6.2E-07**	5.82E-06	-0.30134	-0.30134
	Eq$_2$	4.04E-06	1.33E-05	**-4.1E-07**	2.27E-06	-0.71446	-0.71446
	Eq$_3$	6.94E-07	3.82E-06	**6.22E-08**	1.14E-06	0.028763	0.028763
	Eq$_4$	1.51E-07	6.17E-07	**-5E-08**	1.74E-06	-0.07147	-0.07147
	Eq$_5$	-2.2E-07	1.22E-05	**-3.6E-07**	1E-05	-0.04267	-0.04267
	Eq$_6$	-2E-06	1.27E-05	**-1.5E-07**	4.17E-06	0.045682	0.045682
	Sum	1.55E-05	2.99E-05	**5.77E-06**	1.85E-05	1.309252	0.214584
	Sure	0.9321	0.03721	**0.129346**	0.032312	0.138676	0.052217

Table 10 . The solution of P7 with GWO. PSO and GA

Model	P7	GWO		PSO		GA	
		Ave	**Std**	**Ave**	**Std**	**Ave**	**Std**
Unconstrained	Eq$_1$	0.127037	0.083838	**0.090542**	0.052227	0.145894	0.803546
	Eq$_2$	-0.00188	0.002951	**-0.00222**	0.010148	-1.03864	1.84389
	Eq$_3$	-0.06244	0.118998	**2.4E-06**	1.5E-05	-1.29417	1.944347
	Eq$_4$	3.23E-05	0.000173	**6.68E-06**	5.4E-05	-7.57217	14.05573
	Eq$_5$	-2.1E-05	0.00017	**1.89E-05**	0.000139	14.92211	9.690871
	Sum	0.197969	0.174349	**0.092793**	0.050359	29.84503	11.34493
	Sure	0.827519	0.041963	0.184222	0.056142	0.263126	0.129591
Constrained	Eq$_1$	0.074048	0.057517	0.050266	0.022702	0.36486	1.578264
	Eq$_2$	-0.03889	0.037323	-0.00738	0.02104	1.262703	2.926319
	Eq$_3$	-0.03001	0.09159	0.000813	0.003339	1.809743	3.149888
	Eq$_4$	0.002484	0.008607	0.000104	0.000304	19.29885	13.1848
	Eq$_5$	-0.01029	0.031584	-0.00215	0.00446	-1.7976	3.888444
	Sum	0.385538	0.511875	0.096418	0.103426	5893.319	3837.592
	Sure	1.00075	0.017395	0.372549	0.145352	0.261271	0.115431

5. Conclusions and Remarks

The first objective of this study was to test GWO performance on the systems of nonlinear equations. For this aim, seven systems of nonlinear equations were solved with GWO, PSO and GA. Average and standard deviation of the equation values for fifty run were submitted. Average and standard deviation of the solution times were also submitted. According to the tables (Table 4-Table10), GWO is quite successful for the solution of the systems of nonlinear equations. GWO gives minimum value in four of seven systems in view of sum of the squares of the equations. The second algorithm is PSO. PSO is also quite successful for the solutions of systems of nonlinear equations. As a result, it was proposed to use GWO for the solutions of the systems of nonlinear equations, since GWO is derivative free solution method and easily applied for the systems of nonlinear equations.

The second objective of this study is to offer a new approach for the solution of the systems of nonlinear equations as an optimization problem. In the studies, the systems of nonlinear equations were generally was modelled as unconstrained optimization model taking the sum of squares of each equation. But in this study, the systems of nonlinear equations were modelled as constrained single objective optimization problem. Since the nature inspired optimization algorithms solve unconstrained problems, fitness functions have been prepared using penalty function. According to the results, modelling the systems of nonlinear equations as constrained optimization problems give better results, since the total error was distributed to each equation.

Lastly, PSO is the fastest algorithm among the three optimization algorithms. GWO is the second. As it has been seen GA is not very successful both in view of solution time and best minimum solution.

References

[1] C. Grosan and A. Abraham, "A New Approach for Solving Nonlinear Equations Systems," in IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans, vol. 38, no. 3, pp. 698-714, May 2008.

[2] G. Joshi and M. B. Krishna, "Solving system of non-linear equations using Genetic Algorithm," *2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, New Delhi, 2014, pp. 1302-1308.

[3] Muhammad Aslam Noor, Muhammad Waseem, Some iterative methods for solving a system of nonlinear equations, In Computers & Mathematics with Applications, Volume 57, Issue 1, 2009, Pages 101-106, ISSN 0898-1221, https://doi.org/10.1016/j.camwa.2008.10.067.

[4] S. Qin, S. Zeng, W. Dong and X. Li, "Nonlinear equation systems solved by many-objective Hype," 2015 IEEE Congress on Evolutionary Computation (CEC), Sendai, 2015, pp. 2691-2696.

[5] W. Song, Y. Wang, H. X. Li and Z. Cai, "Locating Multiple Optimal Solutions of Nonlinear Equation Systems Based on Multiobjective Optimization," in IEEE Transactions on Evolutionary Computation, vol. 19, no. 3, pp. 414-431, June 2015.

[6] A. Majd et al., "Multi-population parallel imperialist competitive algorithm for solving systems of nonlinear equations," 2016 International Conference on High Performance Computing & Simulation (HPCS), Innsbruck, 2016, pp. 767-775.

[7] E. Pourjafari and H. Mojallali, "Solving nonlinear equations systems with a new approach based on invasive weed optimization algorithm and clustering," Swarm Evol. Comput., vol. 4, pp. 33–43, Jun. 2012.

[8] M. Jaberipour, E. Khorram, and B. Karimi, "Particle swarm algorithm for solving systems of nonlinear equations," Comput. Math. Appl., vol. 62, no. 2, pp. 566–576, 2011.

[9] J. Wang, "Immune genetic algorithm for solving nonlinear equations," in Proc. 2011 Int. Conf. Mechatr. Sci., Electr. Eng. Comput., Jilin, China, pp. 2094–2097.

[10] Y. Zhou, J. Liu, and G. Zhao, "Leader glowworm swarm optimization algorithm for solving nonlinear equations systems," Electr. Rev., vol. 88, no. 1, pp. 101–106, 2012.

[11] J. Wu, Z. Cui, and J. Liu, "Using hybrid social emotional optimization algorithm with metropolis rule to solve nonlinear equations," in Proc. 10th IEEE Int. Conf. Cogn. Inf. Cognit. Comput., Banff, AB, USA, 2011, pp. 405–411.

[12] Mirjalili, S. How effective is the grey wolf optimizer in training multi-layer perceptrons. Appl. Intell. 2015, 43, 150–161.

[13] Mohd Herwan Sulaiman, Zuriani Mustaffa, Mohd Rusllim Mohamed, Omar Aliman, Using the gray wolf optimizer for solving optimal reactive power dispatch problem, In Applied Soft Computing, Volume 32, 2015, Pages 286-292, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2015.03.041.

[14] H.M. Song, M.H. Sulaiman, M.R. Mohamed. An application of Grey Wolf Optimizer for solving combined economic emission dispatch problems, Int Rev Model Simul (IREMOS), 7 (2014), pp. 838-844.

[15] Zhang, Y., Phillips, P., Wang, S., Ji, G., Yang, J., and Wu, J. (2016) Fruit classification by biogeography-based optimization and feedforward neural network. Expert Systems, 33: 239–253. doi: 10.1111/exsy.12146.

[16] El-Fergany Attia A, Hasanien Hany M. Single and multi-objective optimal power flow using grey wolf optimizer and differential evolution algorithms. Electr Power Compon Syst 2015;43(13):1548–59.

[17] S.A. Medjahed, T. Ait Saadi, A. Benyettou, M. Ouali, Gray Wolf Optimizer for hyperspectral band selection, In Applied Soft Computing, Volume 40, 2016, Pages 178-186, ISSN 1568-4946, https://doi.org/10.1016/j.asoc.2015.09.045.

[18] Y. T. K. Priyanto and L. Hendarwin, "Multi objective optimal power flow to minimize losses and carbon emission using Wolf Algorithm," *2015 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, Surabaya, 2015, pp. 153-158. doi: 10.1109/ISITIA.2015.7219971.

[19] Akhilesh Gotmare, Sankha Subhra Bhattacharjee, Rohan Patidar, Nithin V. George, Swarm and evolutionary computing algorithms for system identification and filter design: A comprehensive review, In Swarm and Evolutionary Computation, Volume 32, 2017, Pages 68-84, ISSN 2210-6502, https://doi.org/10.1016/j.swevo.2016.06.007.

[20] E. Emary, Hossam M. Zawbaa, Aboul Ella Hassanien, Binary grey wolf optimization approaches for feature selection, In Neurocomputing, Volume 172, 2016, Pages 371-381, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2015.06.083.

[21] Seyedali Mirjalili, Seyed Mohammad Mirjalili, Andrew Lewis, Grey Wolf Optimizer, In Advances in Engineering Software, Volume 69, 2014, Pages 46-61, ISSN 0965-9978, https://doi.org/10.1016/j.advengsoft.2013.12.007.

[22] Chao Lu, Shengqiang Xiao, Xinyu Li, Liang Gao, An effective multi-objective discrete grey wolf optimizer for a real-world scheduling problem in welding production, In Advances in Engineering Software, Volume 99, 2016, Pages 161-176, ISSN 0965-9978, https://doi.org/10.1016/j.advengsoft.2016.06.004.

[23] T. Jayabarathi, T. Raghunathan, B.R. Adarsh, Ponnuthurai Nagaratnam Suganthan, Economic dispatch using hybrid grey wolf optimizer, In Energy, Volume 111, 2016, Pages 630-641, ISSN 0360-5442, https://doi.org/10.1016/j.energy.2016.05.105.

[24] Zhang, Sen & Luo, Qifang & Zhou, Yong-Quan. (2017). Hybrid Grey Wolf Optimizer Using Elite Opposition-Based Learning Strategy and Simplex Method. International Journal of Computational Intelligence and Applications. 1750012. 10.1142/S1469026817500122.

[25] Narinder Singh and S. B. Singh, "Hybrid Algorithm of Particle Swarm Optimization and Grey Wolf Optimizer for Improving Convergence Performance," Journal of Applied Mathematics, vol. 2017, Article ID 2030489, 15 pages, 2017. https://doi.org/10.1155/2017/2030489.

[26] . D. Gupta, C. Anand and T. Dewan, "Enhanced heuristic approach for Traveling Tournament Problem based on Grey Wolf Optimizer," *2015 Eighth International Conference on Contemporary Computing (IC3)*, Noida, 2015, pp. 235-240. doi: 10.1109/IC3.2015.7346685

[27] D. Jitkongchuen, "A hybrid differential evolution with grey wolf optimizer for continuous global optimization," *2015 7th International Conference on Information Technology and Electrical Engineering (ICITEE)*, Chiang Mai, 2015, pp. 51-54. doi: 10.1109/ICITEED.2015.7408911

Pakize Erdogmus was born in Erzurum in 1972. She received the B.S. in Electronic and Communication Engineering from Yildiz Technical University, Kocaeli Engineering Faculty in 1993, and M.S. and Ph.D degree Computer Sciences and Numerical Methods from Ataturk University in 1997, 2003 respectively.

From 2003 to 2010, she was Assistant Professor in Duzce University, Technical Education Faculty. From 2010 she has been working in Computer Engineering Department of Duzce University, Engineering Faculty. Her research interests are heuristic algorithms, nature-inspired optimization algorithms and their performance analysis and image processing.