J. Basic. Appl. Sci. Res., 2(11)11136-11148, 2012
© 2012, TextRoad Publication

ISSN 2090-4304
Journal of Basic and Applied Scientific Research
www.textroad.com

Solution to Force-Free and Forced Duffing-Van der Pol Oscillator Using Memetic Computing

Suheel Abdullah Malik[a], Ijaz Mansoor Qureshi[a], Muhammad Zubair[a], Ihsanul Haq[a]

[a] Department of Electronic Engineering, Faculty of Engineering and Technology, International Islamic University, Islamabad, Pakistan

ABSTRACT

We propose memetic computing for the solution to force-free and forced Duffing-van der pol (DVP) oscillator. An approximate mathematical model based on the linear combination of some basis functions employing log sigmoid has been done. Fitness function (FF) which is a function of unknown weights is formulated. Genetic algorithm (GA) and hybrid approach of GA and interior point algorithm (IPA) have been employed for the optimization of the unknown weights. The proposed method has been effectively applied to force-free DVP, as well as single-well, double-well, and double-hump situations of forced DVP oscillator. The results obtained using this method are in good agreement with the numerical methods based on fourth order Runge-Kutta (RK), and Lindsted's method (LM).

KEYWORDS*:* Duffing-van der pol (DVP), nonlinear oscillator; heuristic computation; genetic algorithm; adomian decomposition, homotopy perturbation

I. INTRODUCTION

In the last few decades a considerable attention has been devoted to the study of nonlinear oscillators due to their potential applications in diverse areas of engineering and science [1]. The key issue in the study of nonlinear oscillators is to find the exact solutions of the nonlinear oscillator equations especially using traditional analytical techniques. In this view a rich variety of approximate methods, and the numerical techniques such as variational iteration method (VIM), adomian decomposition method (ADM), homotopy perturbation method (HPM), Runge Kutta method (RKM), Lindsted's method (LM), energy balance method (EBM) , differential transform method (DTM) have been proposed [2, 3]. Duffing van-der pol (DVP) oscillator is one of the most extensively studied dynamical system, which can be used as a model in engineering, electronics, physics, biology, neurology, and many other disciplines [1,4,5]. The chaotic behavior and coupling of the Duffing-van der pol oscillator (DVP) makes it useful in applications, such as chaos communication systems, synchronization in communication engineering, image processing, electrical and automation engineering [1, 2].

A great deal of attention has been devoted toward the solution of DVP equation and several approximate analytical and numerical techniques have been proposed. Cordshooli, and Vahidi [6] employed adomian decomposition method (ADM) to solve the DVP equation. Chen and Liu [7] applied homotopy analysis method (HAM) to study the limit cycle of DVP oscillator. Recently Vahidi et al. [8] applied restarted adomian decomposition method (RADM) to solve DVP equation. Sajadi et al. [9] used homotopy perturbation method (HPM) and variational iteration method (VIM) to study the problem of single-well, double-well and double-hump. Khan et al. [10] employed modified version of homotopy pertubation method (NHPM) to solve the force-free DVP equation. Kimiaeifar et al. [11] applied homotopy analysis method (HAM) to solve the DVP equation. Sweilam and Khader [2] employed He's parameter-expansion method (PEM) to solve the coupled chaotic Duffing-van der pol system. Kakmeni et al. [12] investigated the chaotic behavior of DVP system with two external periodic forces obtained by numerical methods such as bifurcation diagram, Lyapunov exponent and Poincare map. Njah [5] studied the synchronization and anti-synchronization behavior of double-hump DVP oscillator using active control. Recently amalgamation of neural networks and evolutionary computation techniques such as genetic algorithm (GA), particle swarm optimization (PSO), simulated annealing (SA), and interior point algorithm (IPA) have been exploited to solve nonlinear problems in engineering and science [13-15].The classical Van- der pol oscillator equation has been solved using neural network (NN) based approach by Khan et al. [16]. Caetano et al. [17] solved nonlinear differential equations in atomic and molecular physics using ANN tuned with GA.

The aim of this research work is to investigate the solution of the DVP oscillator equation using the heuristic computation approach. Since GA has high potential for global optimization it has been used to obtain the numerical solution of the DVP equation. To exploit the application of hybrid evolutionary computation we have solved DVP oscillator equation using the memetic heuristic approach of GA and IPA. We have also solved three special situations of DVP oscillator equation such as single-well, double-well, and double- hump. The results of the proposed method are compared with some popular analytical and numerical methods. The proposed method based on heuristic approach is stochastic in nature as compared to the conventional

***Corresponding Author:** Suheel Abdullah Malik, Department of Electronic Engineering, Faculty of Engineering & Technology, International Islamic University Islamabad, Pakistan, Email: suheel.abdullah@iiu.edu.pk

methods such as ADM, HPM, DTM etc. which are deterministic in nature. The proposed heuristic computation method can provide the solution to the problem on continuous grid of time while the deterministic methods give the solution on the predefined discrete points. Further most of the conventional deterministic methods provide the results for the time t between 0 and 1. On the other hand proposed heuristic computation method can provide the results for any value of t without repeating the whole procedure.

The major scientific contribution of this work is that an alternate stochastic method based on heuristic computation is proposed which is capable of solving nonlinear DVP oscillator problem. To the best of our knowledge nobody has solved the DVP oscillator problem using the proposed approach.

The remaining paper is organized as follows: In section II the governing equation of DVP oscillator is presented. In section III proposed method is described. Section IV gives a brief overview of GA, and IPA. In section V we present the results and discussion. Finally concluding remarks are given in section VI.

II. DVP Oscillator Equation

The governing equation of the forced DVP oscillator is given by the following second order differential equation [11].

$$\ddot{x}(t) - \mu(1 - x^2(t))\dot{x} + \alpha x(t) + \beta x^3(t) = g(f, \omega, t) \quad (1)$$

where $g(f, \omega, t) = f\, cos(\omega t)$ represents the periodic excitation function, and for a force-free Duffing-van der pol oscillator $g(f, \omega, t) = 0$. ω is the angular frequency of the driving force, f is the amplitude of the excitation, $\mu > 0$ is the damping parameter of the system, while α and β are constant parameters.

The DVP oscillator equation has three main physically fascinating situations, (a) single-well ($\alpha > 0$, $\beta > 0$), (b) double-well ($\alpha < 0$, $\beta > 0$), and (c) double- hump ($\alpha > 0$, $\beta < 0$).

III. METHODOLOGY

We may assume that the solution $x(t)$ and its first and second derivatives, $\dot{x}(t), and\ \ddot{x}(t)$, are a linear combination of some basis functions which can be represented by the following equations.

$$x(t) = \sum_{i=1}^{m} \propto_i \varphi(\omega_i t + \beta_i) \quad (2)$$

$$\dot{x}(t) = \sum_{i=1}^{m} \propto_i \omega_i \dot{\varphi}(\omega_i t + \beta_i) \quad (3)$$

$$\ddot{x}(t) = \sum_{i=1}^{m} \propto_i {\omega_i}^2 \ddot{\varphi}(\omega_i t + \beta_i) \quad (4)$$

where $\varphi(t)$ is the log sigmoid function given by

$$\varphi(t) = \frac{1}{1 + e^{-t}} \quad (5)$$

$\propto_i, \beta_i,\ and\ \omega_i$ are real valued unknown weights (chromosomes), and m is the number of basis functions.
To apply a heuristic algorithm fitness function (FF) is formulated for the given equation as follows

$$FF = \frac{1}{1 + \varepsilon_j} \quad (6)$$

$$\varepsilon_j = \varepsilon_1 + \varepsilon_2 \quad (7)$$

where ε_1 is mean square error linked with the given equation, and ε_2 is mean square error linked with the initial conditions while j is the cycle index.

Fitness function (FF) given by (6) is a function of unknown weights ($\propto_i, \beta_i, \omega_i$). An evolutionary algorithm is used to maximize this fitness function by suitably training the equations represented by (2) to (4). The values of unknown weights are acquired. Consequently the approximate solution $x(t)$ of the given problem at hand is determined using (2).

IV. Heuristic Search Algorithms

A. Genetic Algorithm (GA)

The Genetic algorithm (GA) is one of the well known global search algorithms based on the evolutionary ideas of natural selection, and genetics [18]. GA works on the Darwin's theory about evolution 'survival of the fittest'. The flow chart of the genetic algorithm is shown in Figure 1. GA begins with an initial population of individuals. Each individual with a population represents one possible solution to the problem. Each individual within a population is evaluated using a fitness value. The algorithm evolves population sequentially and iteratively using three fundamental operators: selection, crossover, and mutation. The general procedural steps of GA can be summarized as follows:

Algorithm 1: Genetic Algorithm

Step 1: Generate a population of N chromosomes (C_1, C_2, …, C_N) each of length M using random number generator.

Step 2: Determine the fitness of each individual in the current population.

$$F_1 = f(\bar{x} = C_1); \ldots\ldots; F_N = f(\bar{x} = C_N)$$

(The chromosomes are sorted according to their fitness)

Step 3: Selection of parents and production of offsprings (children).

Suppose the chromosomes are sorted in descending order. These are parents to the next generation. They will produce offsprings with a probability to their fitness through crossover operation.

Step 4: Populating the new generation.

Step 5: Mutation (optional)

Mutation operation is performed if there is no improvement in fitness in the generation or the problem is converging very fast or steady state is reached very easily.

Step 6: Stoppage criteria.

The algorithm terminates if the fitness reaches a certain value or a certain number of cycles has reached.
Else go to step 3.

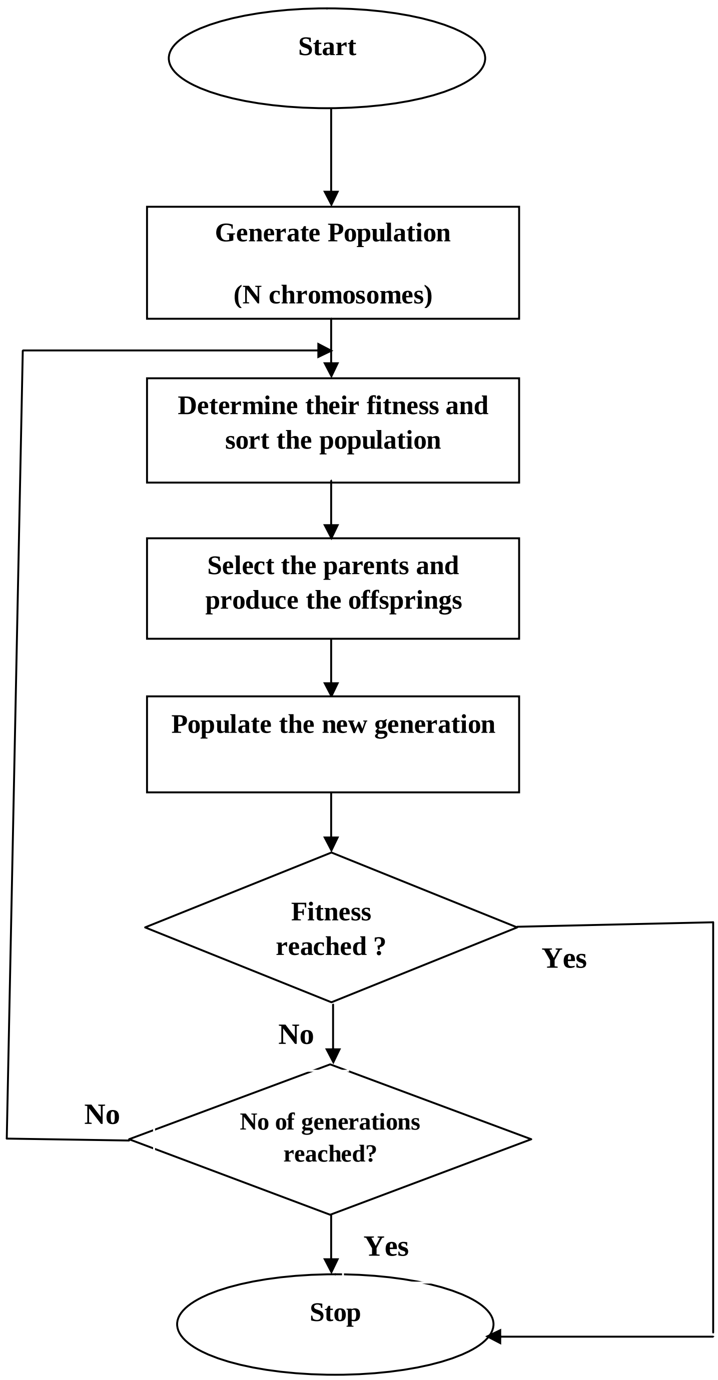

Figure 1. Flow Chart of GA

B. Interior Point Algorithm (IPA)

Interior-point methods (IPM) also called as barrier methods are among the most useful algorithms for solving broad range of optimization problems including linear, nonlinear, convex, and non-convex. Since the revolutionary work of Karmarkarin [19], IPMs have received a tremendous attention, and they have been extensively applied in various practical optimization problems in engineering. The algorithm solves a sequence of barrier subproblems by employing either Newton step or conjugate gradient (CG) step at each iteration. The Newton step is the default step taken by the algorithm which attempts to solve Karush-Kuhn-Tucker (KKT) equations. The algorithm decreases a problem specific merit function. If an attempted step does not decrease the merit function, it is rejected by the algorithm and a new step is attempted. Contrary to the simplex method IPA traverses through the interior feasible region until it reaches an optimal solution [20]. The algorithm iteratively traces the central path while reducing the barrier parameter μ at each iteration [21]. The sequence of barrier parameters must converge to zero [22]. The graphical representation of the IPM algorithm is given in Figure 2. [21].

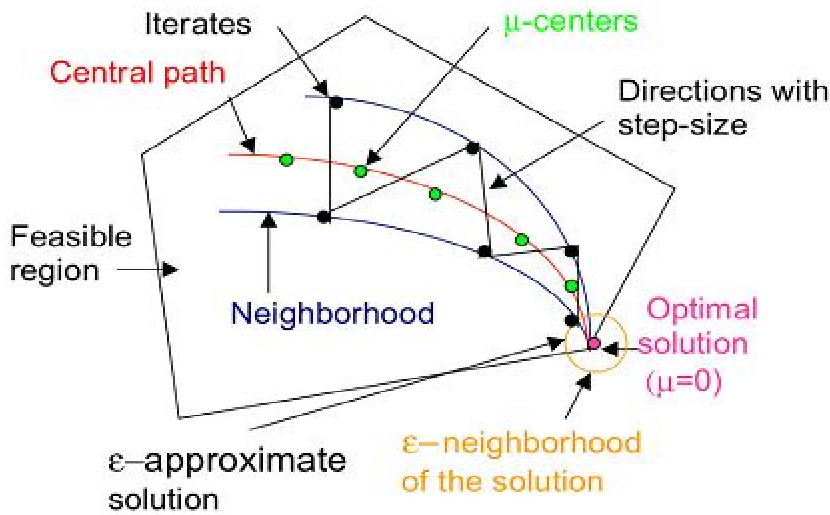

Figure 2. Graphical representation of IPM [21]

Memetic computation is a very recent growing area of evolutionary computation. The memetic computation is a hybridization approach of the global search technique with a local search method. The hybridization algorithms have been shown to be more accurate and fast convergent [23]. Due to the improved performance and fast convergence properties hybrid algorithms have been vastly used in diverse fields of engineering [24-25].

The genetic algorithm (GA) is a global optimizer but in some optimization problems it encounters the trouble of premature convergence. The memetic approach of GA and IPA (GA-IPA) can prevent the problem of premature convergence, and the improved performance can also be achieved [26]. In this paper hybridization of GA, and IPA has been used for the learning and optimization of the unknown weights by minimizing the fitness function. In this hybrid approach the best chromosome found by the GA is given as a starting point to the IPA which is a fine and fast local optimizer and brings down the error to acceptable levels. Thus improved performance with a fast and fine convergence is achieved.

V. RESULTS AND DISCUSSION

In this section we demonstrate the results of applying the proposed method to the force-free Duffing-van der pol (DVP) oscillator, and the forced Duffing-van der pol oscillator equations.

Problem 1: Consider the force- free Duffing-van der pol equation given by [6, 8]

$$\ddot{x}(t) - 0.1(1 - x^2(t))\dot{x} + x(t) + 0.01x^3(t) = 0 \tag{8}$$

with initial conditions,

$$x(0) = 2, \qquad \dot{x}(0) = 0$$

The solution of (8) using ADM, RADM, and LM is given by (9), (10), and (11) respectively [6, 8]

$$x_{ADM} = 2 - 1.04x^2 + 0.104x^3 + 0.089x^4 + \cdots + 0.00053x^8 \tag{9}$$

$$x_{RADM} = 2 - 1.04x^2 + 0.104x^3 + 0.0892667x^4 + \cdots + 2.57313 \times 10^{-14}x^{24} \tag{10}$$

$$x_{LM} = A\cos\omega t + \frac{\alpha}{4}\cos 3\omega t + \mu\left(\frac{3}{4}\sin\omega t - \frac{1}{4}\sin 3\omega t\right) + O(\mu^2) \quad (11)$$

with $A = 2 - \frac{1}{2}\alpha, \omega = 1 + \frac{3}{2}\alpha - \frac{27}{16}\alpha - \frac{1}{16}\mu^2 + O(\mu^2)$

The solution of (8) using the proposed method is obtained by formulating its fitness function (FF) as follows.

$$\varepsilon_j = \frac{1}{11}\sum_{i=1}^{11}(\ddot{x}(t_i) - 0.1(1-x^2)\dot{x} + x + 0.01x^3)^2 + \frac{1}{2}\{(x(0)-2)^2 + (\dot{x}(0))^2\}\Big|_j \quad (12)$$

where $x(t)$, $\dot{x}(t)$, *and* $\ddot{x}(t)$ are given by (2) to (4)

The fitness function given by (12) is minimized by applying heuristic methods. For simulations Matlab R2008a version 7.6.0 has been used in this work. Three different heuristic methods such Genetic algorithm (GA), Interior Point Algorithm (IPA), and the memetic heuristic approach of GA and IPA (GA-IPA) have been implemented for the training of equations (2) to (4) to minimize the fitness function given by (12).

The parameter settings in the optimization tool for the execution of GA, IPA, and GA-IPA algorithms used for this problem are given in Table 1. The equations of (2) to (4) are trained by a training set with inputs taken from $t \epsilon\{0,0.1,0.2,0.3, \ldots,1\}$. The number of basis functions is taken equal to 10. Therefore the number of (chromosomes) unknown weights ($\propto_i, \beta_i, \omega_i$) which have to be tailored is equal to 30. These unknown weights are restricted to the interval$[-5, +5]$. This was observed by simulations that we get better results if we bound unknown weights to this interval. After running GA, IPA, and GA-IPA according to the prescribed settings given in Table 1, the minimum fitness function values achieved for GA, IPA, and GA-IPA are 7.7956 x 10^{-8}, 1.2417 x $10^{-9,}$ and 3.0509 x 10^{-10} respectively. The values of unknown weights (best chromosomes) obtained corresponding to these fitness values are given in Table 2. The approximate solution $x(t)$of the DVP oscillator represented by (9) is achieved by using the values of unknown weights in (2). The results obtained using the proposed method by employing GA, IPA, and GA-IPA are given in Table 3.

The approximate solution $x(t)$obtained from the proposed method using GA, IPA, and GA-IPA are compared with the results of the Lindsted's method (LM) [6], and adomian decomposition method (ADM) [6]. It is evident from the comparison of the results that the proposed method based on the heuristic techniques GA, IPA, and GA-IPA provide much satisfactory results which are in good agreement with the numerical method based on LM. The comparison further reveals that the proposed method is more accurate than ADM. The effectiveness of the proposed approach is quite evident from the absolute errors which have been computed relative to the LM. The superior performance of memetic approach of GA and IPA (GA-IPA) is also quite evident from the results. Further it would be worth to mention that GA-IPA approach took only 990 iterations as compared to 1500 iterations taken by GA.

In Table 4 we present absolute errors of GA, IPA, and GA-IPA, computed relative to LM [6] and compare these with the errors obtained using restarted adomian decomposition method (RADM) [8]. From the comparison it is significant that the performance of the proposed method is comparatively better than RADM.

TABLE 1. SETTINGS OF THE ALGORITHMS

GA		IPA	
Parameters	**Setting**	**Parameters**	**Setting**
Population Size	240	Maximum Iterations	1000
Chromosome Size	30	Maximum function evaluations	60000
Creation function	Uniform	X tolerance	1e-6
Fitness scaling function	Proportional	Function tolerance	1e-24
Selection function	Stochastic Uniform	Nonlinear constraint tolerance	1e-12
Reproduction crossover fraction	0.6	Minimum perturbation	1e-8
Mutation function	Adaptive feasible	Maximum perturbation	0.1
Crossover function	Heuristic	Derivative type	Forward differences
Migration direction	Forward	Hessian	BFGS
Hybrid function	fmincon (IPA)	Subproblem algorithm	ldl factorization
No of generations	1500	Scaling	objective and constraints
Function tolerance	1e-22	Initial barrier parameter	0.1
Nonlinear constraint tolerance	1e-12		

TABLE 2. UNKNOWN WEIGHTS ACQUIRED BY THE ALGORITHMS

Algorithm	index(i)	α_i	ω_i	β_i
GA	1	-0.177793228724768	1.372143885594660	0.267269199684203
	2	-3.049739835713970	1.563871029800120	-3.254523880756310
	3	-2.874952834310280	-0.405584227198786	2.271394966928810
	4	-0.349735472562995	2.760102458375770	2.455955703923160
	5	-0.627598796531771	1.740348775679930	-4.799283367298240
	6	1.845313381842820	3.609775852071270	3.439964856722690
	7	2.348621061435780	-1.521669443019670	1.961009551059900
	8	-1.546577907838180	1.140333235413910	0.012979425823147
	9	2.554300772541020	2.206712789642280	1.116526783509830
	10	2.382873095545740	-1.451357186416530	-2.668660904156950
IPA	1	-1.121799848865420	0.420143531084565	0.034323237619862
	2	-0.508611137824402	0.746024565990766	1.804652145488360
	3	1.584952571402940	2.998952520449050	2.696161332877600
	4	-0.874019196843631	-1.479609888608340	-0.778953119440877
	5	-0.824407746358772	0.320247653958491	0.545579678464943
	6	-1.152516096018750	0.834305530936738	-0.374731349720112
	7	1.734747474779950	-1.383761471401220	2.641836670355020
	8	2.215353452581230	-1.501236084993330	2.798800354505280
	9	-1.505673801930070	-2.071985708808530	-0.928420142542375
	10	-1.683042566769000	1.169726402658540	-0.883378881913459
GA-IPA	1	-1.234744607081430	0.232944210747082	1.457483658480350
	2	2.592612422029060	-1.567763207476860	2.153321905631980
	3	2.208874828179050	2.027267452377460	0.842639594114534
	4	-1.455291413720960	-3.045586200700630	-4.137726344376620
	5	-0.823008658068831	2.365036090673610	4.026104317205500
	6	1.533762855190490	-2.144640828654960	4.310907228423700
	7	-0.439278165377514	-0.696639801223637	0.680993653140154
	8	1.131061904093790	3.165451409082820	2.486680936175640
	9	-3.368197116088810	0.976129223253480	0.017384897958785
	10	-0.608813786434817	1.532770791383090	4.372269147581950

TABLE 3. COMPARISON OF THE RESULTS OF GA, IPA, GA-IPA, LM [6], and ADM [6]

	x(t)					Absolute Error			
t	**LM**	**GA**	**IPA**	**GA-IPA**	**ADM**	**GA**	**IPA**	**GA-IPA**	**ADM**
0	2.00000	2.00000	2.00000	2.00000	1.9975	0.000E+00	0.000E+00	0.000E+00	2.500E-03
0.1	1.98971	1.98971	1.98971	1.98971	1.98724	0.000E+00	0.000E+00	0.000E+00	2.470E-03
0.2	1.95936	1.95936	1.95936	1.95936	1.95697	0.000E+00	0.000E+00	0.000E+00	2.390E-03
0.3	1.9098	1.90981	1.9098	1.9098	1.90758	1.000E-05	0.000E+00	0.000E+00	2.220E-03
0.4	1.84202	1.84203	1.84202	1.84202	1.84008	1.000E-05	0.000E+00	0.000E+00	1.940E-03
0.5	1.75702	1.75703	1.75702	1.75702	1.75552	1.000E-05	0.000E+00	0.000E+00	1.500E-03
0.6	1.65586	1.65587	1.65586	1.65585	1.65493	1.000E-05	0.000E+00	1.000E-05	9.300E-04
0.7	1.53958	1.53959	1.53958	1.53957	1.53937	1.000E-05	0.000E+00	1.000E-05	2.100E-04
0.8	1.40922	1.40925	1.40923	1.40922	1.40982	3.000E-05	1.000E-05	0.000E+00	-6.000E-04
0.9	1.26581	1.26588	1.26586	1.26586	1.26726	7.000E-05	5.000E-05	5.000E-05	1.450E-03
1.0	1.11033	1.11056	1.11053	1.11053	1.11267	2.300E-04	2.000E-04	2.000E-04	2.340E-03
1.1	0.94373	0.94437	0.94435	0.94434	0.94704	6.400E-04	6.200E-04	6.100E-04	3.310E-03
1.2	0.76686	0.76848	0.76845	0.76845	0.77147	1.620E-03	1.590E-03	1.590E-03	4.610E-03
1.3	0.58037	0.5841	0.58407	0.5841	0.58715	3.730E-03	3.700E-03	3.730E-03	6.780E-03
1.4	0.38462	0.39254	0.39255	0.39267	0.39545	7.920E-03	7.930E-03	8.050E-03	1.083E-02
1.5	0.17946	0.19518	0.19532	0.1957	0.19795	1.572E-02	1.586E-02	1.624E-02	1.849E-02

TABLE 4. COMPARISON OF ABSOLUTE ERRORS OF GA, IPA, GA-IPA, and RADM [8]

	x(t)	Absolute Error			
t	**LM**	**GA**	**IPA**	**GA-IPA**	**RADM**
0	2.00000	0.0000E+00	0.0000E+00	0.0000E+00	2.5000E-03
0.1	1.98971	0.0000E+00	0.0000E+00	0.0000E+00	2.4751E-03
0.2	1.95936	0.0000E+00	0.0000E+00	0.0000E+00	2.3902E-03
0.3	1.9098	1.0000E-05	0.0000E+00	0.0000E+00	2.2202E-03
0.4	1.84202	1.0000E-05	0.0000E+00	0.0000E+00	1.9337E-03
0.5	1.75705	2.0000E-05	3.0000E-05	3.0000E-05	1.5052E-03
0.6	1.65598	1.1000E-04	1.2000E-04	1.3000E-04	9.2768E-04
0.7	1.53999	4.0000E-04	4.1000E-04	4.2000E-04	2.2146E-04
0.8	1.41039	1.1400E-03	1.1600E-03	1.1700E-03	5.6169E-04
0.9	1.26872	2.8400E-03	2.8600E-03	2.8600E-03	1.3411E-03
1	1.11696	6.4000E-03	6.4300E-03	6.4300E-03	2.0159E-03

Problem 2: Consider DVP oscillator represented by (1) with the following parameter values [9]

a) α =0.5, β =0.5, μ=0.1, ω=0.79, f=0.5 α >0, β >0 (single-well situation)
b) α = -0.5, β =0.5, μ=0.1, ω=0.79, f=0.5 α<0, β >0 (double- well situation)
c) α =0.5, β = -0.5, μ=0.1, ω=0.79, f=0.5 α >0, β<0 (double-hump situation)

with the following initial conditions

$$x(0) = 1, \qquad \dot{x}(0) = 0$$

To apply the proposed method fitness function for each of the three cases is formulated. The fitness function for single well situation is as follows.

$$\varepsilon_j = \frac{1}{11}\sum_{i=1}^{11}\left(\ddot{x} - 0.1(1-x^2)x + 0.5\dot{x} + 0.5x^3 - 0.5\cos(0.79t)\right)^2 + \frac{1}{2}\left\{(x(0)-1)^2 + \left(\dot{x}(0)\right)^2\right\}\Bigg|_j \tag{13}$$

Similarly fitness functions for double well and double hump are formulated. The parameters settings for algorithms GA, IPA, and GA-IPA for single-well are given in Table 5 and for double-well and double- hump in Table 6.

The algorithms are executed according to the prescribed settings. The values of unknown weights attained for three situations, single-well, double-well, and double-hump are given in Table 7, Table 8, and Table 9 respectively.

The results obtained using GA, IPA, and GA-IPA for the three situations, single-well, double-well, and double-hump are given in Table 10, Table 11, and Table 12 respectively. For comparison we use the results of this problem obtained using fourth order Runge-Kutta (RK), variational iteration method (VIM), and homotopy perturbation method (HPM) given in [9]. It is significant from comparison of the results that the proposed heuristic method employing GA, IPA, and GA-IPA provide the solution of DVP oscillator (1) quite accurately for all the three situations, single-well, double-well, and double-hump. Further the results obtained using heuristic methods are more precise than HPM and VIM. The superiority of GA-IPA is quite evident from the results in all the three situations. The absolute errors given in Tables 10-12 have been computed relative to fourth order Runge-Kutta (RK) method [9].

TABLE 5. SETTINGS OF THE ALGORITHMS (SINGLE-WELL SITUATION)

GA		IPA	
Parameters	**Setting**	**Parameters**	**Setting**
Population Size	240	Start point	Weights from GA
Chromosome Size	30	Maximum Iterations	1000
Creation function	Uniform	Maximum function evaluations	200000
Fitness scaling function	Proportional	X tolerance	1e-06
Selection function	Stochastic Uniform	Function tolerance	1e-20
Reproduction crossover fraction	0.6	Nonlinear constraint tolerance	1e-08
Mutation function	Adaptive feasible	Mimimum perturbation	1e-08
Crossover function	Heuristic	Maximum perturbation	0.1
Migration direction	Both	Derivative type	Forward differences
Hybridization	IPA	Hessian	BFGS
No of generations	1000	Subproblem algorithm	ldl factorization
Function tolerance	1e-20	Scaling	objective and constraints
Nonlinear constraint tolerance	1e-20	Initial barrier parameter	0.1
Bounds	[-10,10]		

TABLE 6. SETTINGS OF THE ALGORITHMS (DOUBLE-WELL AND DOUBLE-HUMP SITUATIONS)

GA			IPA		
Parameters	**Settings**		**Parameters**	**Settings**	
	Double well	**Double hump**		**Double well**	**Double hump**
Population size	240	240	Start point	Weights from GA	Weights from GA
Chromosome size	30	30	Maximum iterations	1000	1000
Creation function	Uniform	Uniform	Maximum function evaluations	60000	150000
Fitness scaling function	Proportional	Proportional	X-tolerance	1e-6	1e-6
Selection function	Stochastic	Remainder	Function tolerance	1e-18	1e-22
Reproduction elite count	3	3	Nonlinear constraint tolerance	1e-18	1e-8
Reproduction crossover fraction	0.8	0.6	Minimum perturbation	1e-8	1e-8
Mutation function	Adaptive feasible	Adaptive feasible	Maximum perturbation	0.1	0.1
Crossover function	Heuristic	Heuristic	Derivative type	Forward differences	Central differences
Migration direction	Forward	Both	Hessian	BFGS	BFGS
Migration fraction	0.3	0.2	Subproblem algorithm	ldl factorization	ldl factorization
Migration interval	20	25	Scaling	Objective & constraints	Objective & constraints
Hybridization	IPA	IPA	Initial barrier parameter	0.1	0.1
No. of generations	1500	1500			
Function tolerance	1e-22	1e-24			
Nonlinear constraint tolerance	1e-10	1e-18			
Bounds	-10,+10	-20,+20			

TABLE 7. UNKNOWN WEIGHTS ACQUIRED BY THE ALGORITHMS (SINGLE-WELL SITUATION)

Algorithm	index(i)	α_i	ω_i	β_i
	1	-1.265889985090810	1.092793964017490	-2.521966427031720
	2	-0.080971147083984	-2.684965763982810	0.905133471080701
	3	1.672197087063020	-1.081914936624450	1.736136122217580
	4	-0.093971878638577	-3.104703481247070	-0.622790297096618
GA	5	-0.439331493429415	1.836044050217470	-0.928773178627185
	6	1.755029922105900	1.898392937479290	3.146558994004050
	7	0.005681627445785	-0.485072682968863	0.297801672544738
	8	-1.433917964568520	-0.303711713634846	4.651312091661520
	9	0.957187053339987	0.422228256448854	-1.498945085712250
	10	-0.741816693762730	-1.300739919289930	1.075096282269960
	1	-3.366540927777840	-0.695904078565238	-1.488560550141650
	2	-0.411968135837980	-0.105196898567083	0.292091550953972
	3	-0.215184226168917	-0.610945247008287	-0.773890022421320
	4	-0.517829846659449	0.551353210856498	-0.672422637825685
IPA	5	0.825115934559723	-1.398190441496490	0.687082018158296
	6	-0.201694410257641	1.403691206472150	3.135842593748000
	7	0.184231905202183	-0.072863528166909	-2.251822700851530
	8	1.297996489143470	1.065908756517320	0.470375391738961
	9	-2.675318568537380	0.619339557275293	-1.009069504675560
	10	1.712944895119460	-0.885757699083415	3.132104765412600
	1	-1.595699014585090	0.899241933683106	-2.162074100442170
	2	0.432572864151938	-1.769831224648860	1.065357373198600
	3	1.126073804646040	-0.882516492699905	1.750722316695450
	4	-0.025244391637788	-2.007871782280050	-0.848465670646516
GA-IPA	5	0.000034735803093	0.994711535812835	-0.503432012659814
	6	0.975813639648169	1.772990532096380	1.785700502978950
	7	0.072215645721349	-0.324572085099750	0.214266334263812
	8	-0.768876133296560	-0.041942988918680	2.935571080170680
	9	0.549393650027235	-0.184094122075768	-1.101351967757240
	10	-0.621710189709648	-1.405835778101230	0.531252411316486

TABLE 8. UNKNOWN WEIGHTS ACQUIRED BY THE ALGORITHMS (DUBLE-WELL SITUATION)

Algorithm	index(i)	α_i	ω_i	β_i
	1	-1.106389539379750	0.519952229380371	0.242684564980686
	2	2.032805933329850	1.218187143719400	2.429174237626060
	3	2.218196182261100	0.860222077787026	-2.877871963618480
	4	-0.216153098472222	1.226845901227490	-1.404100652461520
GA	5	-0.377820612000670	-1.487628264794860	2.596208990975920
	6	-2.369491469777590	0.331846429807915	-1.426311568822080
	7	-0.780496783025413	1.933051189824190	1.519948850215280
	8	-1.411227587992220	1.930967910874740	3.812887646930820
	9	2.457084518681930	1.390376926173040	-1.918754174759930
	10	2.268310865063920	-1.544546845906380	3.377408282567570
	1	-2.839197309899250	0.991562030993229	-2.507273023768120
	2	2.234996026290390	-1.041514939623520	-0.540180298782562
	3	0.652073746506419	0.287640053784605	1.686564563192340
	4	-1.516036800417300	1.162871457856940	-1.101891729106960
IPA	5	-1.706366226758330	0.110448488312742	-0.814302803964632
	6	0.875952746862778	1.488257006439010	-0.092706760870700
	7	-0.628111866299581	0.277794689093435	0.337009077245961
	8	-0.159153031009649	1.051869027750690	0.737956971716954
	9	3.626299968558260	1.394934331357520	-1.804899016671840
	10	0.738577797503401	1.272027159628920	-0.452987891014406
	1	-0.284054804086385	-0.236811726194791	0.245224201579831
	2	0.625612650321309	1.129169963239380	0.714869967947955
	3	1.264753325031940	-0.640177640006530	-1.096403143439970
	4	-0.042522121548590	0.553883204697940	-0.639565158724306
GA-IPA	5	-0.562386901970556	-1.680767593091950	0.790083450489341
	6	-1.164716663024460	0.572546538855296	-0.675382864004003
	7	0.272585804352962	-0.167925205450166	1.572699543193290
	8	-1.045323005459440	1.344159010173530	1.517380197856290
	9	1.548382075356330	1.683205761905120	-2.232225797171660
	10	1.785847840135400	-1.002300500792990	2.967357367425730

TABLE 9. UNKNOWN WEIGHTS ACQUIRED BY THE ALGORITHMS (DOUBLE-HUMP SITUATION)

Algorithm	Index(i)	α_i	ω_i	β_i
	1	1.192870162603730	-0.711750462375938	1.043756398746230
	2	2.049766539550740	0.117622360536661	-0.145243339469248
	3	0.001667790450654	4.305225834970370	1.530014216481870
	4	1.553411780292540	1.609616750654910	-1.884653012672490
GA	5	-1.511832268651550	-0.655468812639403	0.093273323765845
	6	1.440488121127990	-1.071568775623080	1.797079394614090
	7	-1.392510050750330	1.679695649258460	2.056357865402550
	8	0.245722857624671	-1.326733813053720	2.086613245436550
	9	2.529430061638120	2.462287338798440	-5.333049517074870
	10	-0.482163869644241	2.090049645243980	5.256888620896470
	1	3.817834054554160	0.187141566168683	-3.395667211155050
	2	5.332124314907600	1.418442751529810	-5.250722964441450
	3	-2.158398571262080	-3.085465757381100	7.702584988451890
	4	-1.398343911120740	-0.569859957706764	1.084998239776930
IPA	5	2.196781234981200	-0.747976191759508	-0.446891476473399
	6	2.195789773353360	0.161503469808081	3.997387116500730
	7	-1.781231800390960	-0.919056686262346	1.469629415468130
	8	2.577483208690370	0.738746510553374	0.974268969697450
	9	8.923469343649400	-0.772901997544750	-2.505731671932690
	10	-0.273816082761995	-0.500401548803586	-1.301671726083450
	1	1.240642347032380	-0.722210559793231	1.095492035294410
	2	2.182861468112850	0.194394889547447	-0.159073730082925
	3	0.009961705724485	4.594886963565530	1.628338562228540
	4	1.699569959674330	1.587109469725120	-2.100989111588310
GA-IPA	5	-1.621222064807400	-0.652063496901351	0.125520376141866
	6	1.482422396313390	-1.085897551765640	1.910359610722290
	7	-1.452053698302400	1.721863375285820	1.919656215302390
	8	0.203457102639309	-1.406990651854360	2.222745302475970
	9	2.791641323983200	2.569536416700710	-5.874991502569490
	10	-0.484778160718981	2.195006812720280	5.637864072860240

TABLE 10. COMPARISON OF THE RESULTS OF GA, IPA, GA-IPA, RK [9], HPM [9], and VIM [9] (SINGLE-WELL SITUATION)

	t	0.2	0.4	0.6	0.8	1
	RK	0.99004	0.9607	0.91341	0.85024	0.77352
	HPM	0.99004	0.96075	0.91383	0.85216	0.77973
x(t)	VIM	0.99004	0.9607	0.91341	0.85025	0.77353
	GA	0.99005	0.96071	0.91342	0.85025	0.77352
	IPA	0.99005	0.9607	0.91342	0.85025	0.77352
	GA-IPA	0.99004	0.9607	0.91341	0.85025	0.77352
	HPM	0.000E+00	5.000E-05	4.200E-04	1.920E-03	6.210E-03
Absolute	VIM	0.000E+00	0.000E+00	0.000E+00	1.000E-05	1.000E-05
Error	GA	1.000E-05	1.000E-05	1.000E-05	1.000E-05	0.000E+00
	IPA	1.000E-05	0.000E+00	1.000E-05	1.000E-05	0.000E+00
	GA-IPA	0.000E+00	0.000E+00	0.000E+00	1.000E-05	0.000E+00

TABLE 11. COMPARISON OF THE RESULTS OF GA, IPA, GA-IPA, RK [9], HPM [9], and VIM [9] (DOUBLE-WELL SITUATION)

	t	0.2	0.4	0.6	0.8	1
	RK	1.00994	1.03911	1.08544	1.14538	1.21377
	HPM	1.00994	1.03918	1.08621	1.14937	1.22785
x(t)	VIM	1.00994	1.03911	1.08544	1.14539	1.21382
	GA	1.00995	1.03912	1.08545	1.14539	1.21379
	IPA	1.00995	1.03912	1.08545	1.14539	1.21378
	GA-IPA	1.00994	1.03911	1.08545	1.14538	1.21378
Absolute	HPM	0.000E+00	7.000E-05	7.700E-04	3.990E-03	1.408E-02
Error	VIM	0.000E+00	0.000E+00	0.000E+00	1.000E-05	5.000E-05
	GA	1.000E-05	1.000E-05	1.000E-05	1.000E-05	2.000E-05
	IPA	1.000E-05	1.000E-05	1.000E-05	1.000E-05	1.000E-05
	GA-IPA	0.000E+00	0.000E+00	1.000E-05	0.000E+00	1.000E-05

TABLE 12. COMPARISON OF THE RESULTS OF GA, IPA, GA-IPA, RK [9], HPM [9], and VIM [9] (DOUBLE-HUMP SITUATION)

	t	0.1	0.2	0.5	0.75	1
	RK	1.0025	1.01001	1.063	1.14346	1.26039
	HPM	1.0025	1.01001	1.06296	1.14209	1.25055
x(t)	VIM	1.0025	1.01001	1.063	1.14346	1.26035
	GA	1.0025	1.01001	1.063	1.14346	1.26039
	IPA	1.0025	1.01001	1.06301	1.14347	1.2604
	GA-IPA	1.0025	1.01001	1.06301	1.14347	1.26039
	HPM	0.000E+00	0.000E+00	4.000E-05	1.370E-03	9.840E-03
Absolute	VIM	0.000E+00	0.000E+00	0.000E+00	0.000E+00	4.000E-05
Error	GA	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
	IPA	0.000E+00	0.000E+00	1.000E-05	1.000E-05	0.000E+00
	GA-IPA	0.000E+00	0.000E+00	1.000E-05	1.000E-05	0.000E+00

Problem 3: Consider the following force-free Duffing-van der pol oscillator equation [10]

$$\ddot{x}(t) + \left(\frac{4}{3} + 3x^2\right)\dot{x} + \frac{1}{3}x + x^3 = 0 \tag{14}$$

With initial conditions,

$$x(0) = -0.28868, \qquad \dot{x}(0) = 0.12$$

The DVP oscillator represented by (14) is solved using the proposed method by employing GA, IPA, and GA-IPA algorithms. The fitness function is formulated as follows

$$\varepsilon_j = \frac{1}{11}\sum_{i=1}^{11}\left(\ddot{x} - \left(\frac{4}{3} + 3x^2\right)\dot{x} + \frac{1}{3}x + x^3\right)^2 + \frac{1}{2}\{(x(0) + 0.28868)^2 + (\dot{x}(0) - 0.12)^2\}\bigg|_j \tag{15}$$

The parameters settings used for the execution of the algorithms GA, IPA, and GA-IPA are given in Table 13. The unknown weights achieved and the results obtained using GA, IPA, and GA-IPA are given in Table 14, and Table 15 respectively. The absolute errors computed relative to the fourth order Runge-Kutta (RK) method are given in Table 16.

We use fourth order Runge- Kutta (RK) method, modified homotopy perturbation method (NHPM), and Differential Transform method (DTM) for comparison of the results given in [10]. The comparison of results clearly illustrates the competency of the proposed method. The results obtained from the proposed method are satisfactory and in a good agreement with the numerical method based on fourth order Runge- Kutta (RK). It is clear from the results that the performances of GA, IPA, and GA-IPA are significantly better than the differential transform method (DTM). The remarkably good performance of GA-IPA is quite evident in this problem also. Further the results of IPA, and GA-IPA are comparatively better than NHPM.

TABLE 13. SETTINGS OF THE ALGORITHMS

GA		IPA	
Parameters	**Setting**	**Parameters**	**Setting**
Population Size	240	Start point	Weights from GA
Chromosome Size	30	Maximum Iterations	1000
Creation function	Uniform	Maximum function evaluations	200000
Fitness scaling function	Proportional	X tolerance	1.00E-06
Selection function	Stochastic Uniform	Function tolerance	1.00E-24
Reproduction crossover fraction	0.6	Nonlinear constraint tolerance	1.00E-10
Reproduction Elite count	3		
Mutation function	Adaptive feasible	mimimum perturbation	1.00E-08
Crossover function	Heuristic	maximum perturbation	0.1
Migration direction	Forward	derivative type	Forward differences
Migration fraction	0.3		
Mifration Interval	25		
Hybridization	IPA	Hessian	BFGS
No of runs	1500	subproblem algorithm	ldl factorization
Function tolerance	1.E-24	scaling	objective and constraints
Nonlinear constraint tolerance	1.E-10	initial barrier parameter	0.1

TABLE 14. UNKNOWN WEIGHTS ACQUIRED BY THE ALGORITHMS

Algorithm	index(I)	α_i	ω_i	β_i
	1	-1.91306516522	0.02175192151	1.29904899010
	2	1.70543263956	0.98854294435	0.14041737022
	3	1.43216420823	1.34875958181	2.66144272062
	4	0.77864220343	-0.27375397568	1.17238262963
GA	5	0.31009987831	-0.61181223325	-1.17654805869
	6	-0.86386475139	1.02024670131	1.82682599667
	7	0.07840918331	-0.69676627753	-0.26524559724
	8	-1.50822160178	0.96930100531	0.07388755505
	9	1.03044873446	0.19994355052	-1.37000284459
	10	-0.90397715307	-0.50116607023	-0.13660866351
	1	-1.47699748839	-0.75481030473	-1.11814642191
	2	0.08576367111	0.68642599788	0.36183184369
	3	-0.12680159035	1.41770408305	-0.38619987951
	4	0.16446402067	-1.95905490219	-0.93291827889
IPA	5	-0.24907914864	-1.65865427115	-1.26206961004
	6	-0.11915962203	-0.03681371193	-0.52649409703
	7	0.00105567820	-0.72484245418	0.36438882369
	8	0.40538553888	-0.90135083075	-1.17188997320
	9	0.02264743289	1.71360046672	-0.22189100676
	10	0.04475885194	-0.72603614550	-0.02238247734
	1	-0.75958061636	-0.12008938922	0.77292721647
	2	0.23004821405	1.34518813058	-0.02635950119
	3	1.02338291601	1.24638664911	2.22166706939
	4	0.02642539682	0.28946605634	0.56203668275
GA-IPA	5	0.12403534086	-0.35834293517	-0.55929926728
	6	-0.40195043859	1.10618287666	0.37127397210
	7	0.09951879349	-0.30950893860	-0.13914380752
	8	-1.24768968326	0.02250125920	0.28154548452
	9	0.44819179642	0.76064803340	-0.70035768911
	10	-0.24297156050	0.57411047343	-0.12721846243

TABLE 15. COMPARISON OF THE RESULTS OF GA, IPA, GA-IPA, RK [10], NHPM [10], and DT [10]

			x(t)			
t	RK	GA	IPA	GA-IPA	NHPM	DT
0	-0.28868000000	-0.28868029738	-0.288680019	-0.28867998436	-0.28868000000	-0.28868000000
0.01	-0.28748349253	-0.28748376209	-0.287483494	-0.28748346243	-0.28748347499	-0.28748522585
0.02	-0.28629387437	-0.28629413304	-0.286293874	-0.28629384488	-0.28629385687	-0.28630661206
0.03	-0.28511109904	-0.28511134719	-0.285111099	-0.28511106933	-0.28511108168	-0.28514428684
0.04	-0.28393510355	-0.28393534221	-0.283935105	-0.28393507400	-0.28393508631	-0.28399838456
0.05	-0.28276582565	-0.28276605654	-0.282765832	-0.28276579778	-0.28044716121	-0.28286904593
0.06	-0.28160320408	-0.28160342933	-0.281603218	-0.28160318022	-0.27929767140	-0.28175641828
0.07	-0.28044717833	-0.28044740045	-0.280447205	-0.28044716152	-0.27814565867	-0.28066065588
0.08	-0.27929768854	-0.27929791050	-0.279297732	-0.27929768251	-0.27701806494	-0.27951892013
0.09	-0.27815467572	-0.27815490080	-0.27815474	-0.27815468472	-0.27815465866	-0.27852037991
0.1	-0.27701808173	-0.27701831337	-0.277018171	-0.27701811031	-0.27701806494	-0.27747621187

TABLE 16. ABSOLUTE ERRORS OF GA, IPA, GA-IPA, NHPM, and DT

t	Error of GA	Error of IPA	Error of GA-IPA	Error of NHPM	Error of DT
0	2.9738E-07	1.9220E-08	1.5640E-08	0.0000E+00	0.0000E+00
0.01	2.6956E-07	1.1200E-09	3.0100E-08	1.7540E-08	1.7333E-06
0.02	2.5867E-07	1.0000E-10	2.9490E-08	1.7500E-08	1.2738E-05
0.03	2.4815E-07	4.0000E-10	2.9710E-08	1.7360E-08	3.3188E-05
0.04	2.3866E-07	1.3200E-09	2.9550E-08	1.7240E-08	6.3281E-05
0.05	2.3089E-07	6.0200E-09	2.7870E-08	2.3187E-03	1.0322E-04
0.06	2.2525E-07	1.4250E-08	2.3860E-08	2.3055E-03	1.5321E-04
0.07	2.2212E-07	2.6430E-08	1.6810E-08	2.3015E-03	2.1348E-04
0.08	2.2196E-07	4.2980E-08	6.0300E-09	2.2796E-03	2.2123E-04
0.09	2.2508E-07	6.4070E-08	9.0000E-09	1.7060E-08	3.6570E-04
0.1	2.3164E-07	8.9680E-08	2.8580E-08	1.6790E-08	4.5813E-04

VI. Conclusions

In this work heuristic computing approach has been exploited to investigate the solution of forced and force-free Duffing-van der pol (DVP) oscillator. On the basis of the simulation results and the comparisons made with some popular numerical and analytical methods we can conclude that the heuristic approach is an effective method for solving DVP oscillator equation. The results obtained using our proposed method are in excellent agreement with the numerical methods based on fourth order Runge-Kutta (RK), and Lindsted's method (LM). The reliability and the efficacy of the proposed method using heuristic approaches of GA, IPA, and GA-IPA are also demonstrated by solving three interesting situations, single-well, double-well, and double-hump of the forced DVP oscillator. It has been observed that proposed method shows ascendancy on differential transform method (DTM), adomian decomposition method (ADM), and homotopy perturbation method (HPM) in comparison with numerical methods based on fourth-order Runge Kutta (RK) and Lindsted's method (LM). Moreover it has also been shown that proposed method can approximate the solution with the accuracy comparable to RADM, NHPM, and VIM or even better in some cases. The potency of the proposed method is that it can provide the approximate solution on the continuous time domain.

REFERENCES

[1] K. Kyamakya, C. Ngoy, M. Tamasala, and J. Chedjou, "A novel image processing approach combining a 'coupled nonlinear oscillators'-based paradigm with cellular neural networks for dynamic robust contrast enhancement," ISAST Transactions on Computers and Intelligent Systems, No. 1, Vol. 2, 2010.

[2] N. H. Sweilam, and M. M. Khader, "A Note on He's parameter-expansion method of coupled Van der pol–Duffing oscillators," Applications and Applied Mathematics, no.1, pp. 94-100, Aug. 2010,

[3] D. D. Ganji, M. Gorji, S. Soleimani, and M. Esmaeilpour, " Solution of nonlinear cubic-quintic Duffing oscillators using He's Energy Balance Method," Journal of Zhejiang University SCIENCE A, 10(9), pp. 1263-1268, July 2009.

[4] X. Li, J.C. Ji, C. H. Hansen, and C. Tan, "The response of a Duffing–van der pol oscillator under delayed feedback control," Journal of Sound and Vibration 291, pp. 644–655, 2006.

[5] A. N. Njah," Synchronization and anti-synchronization of double hump Duffing-Van der pol oscillators via active control," Journal of Information and Computing Science, Vol. 4, no. 4, pp. 243-250, Aug 2009.

[6] G. A. Cordshooli, and A. R. Vahidi," Solution of Duffing-van der pol equation using decomposition method," Adv. Studies Theor. Phys., Vol. 5, no. 3, pp. 121-129, 2011.

[7] Y.M. Chen and J.K. Liu," Uniformly valid solution of limit cycle of the Duffing–van der Pol equation," Mechanics Research Communications 36, pp. 845–850, June 2009.

[8] A. R. Vahidi , Z. Azimzadeh, and S. Mohammadifar," Restarted Adomian Decomposition Method for Solving Duffing-van der Pol Equation," Applied Mathematical Sciences, Vol. 6, no. 11, pp. 499 – 507, 2012.

[9] H. Sajadi, D.D. Ganji, and Y. V. Shenas," Application of numerical and semi-analytical approach on Van der Pol–Duffing oscillators," Journal of Advanced Research in Mechanical Engineering, Vol.1, Iss. 3, pp. 136-141, 2010.

[10] N. A. Khan, M. Jamil, S. A. Ali, and N. A. Khan," Solutions of the force-free Duffing-van der Pol oscillator equation," International Journal of Differential Equations, 2011.

[11] A. Kimiaeifar, A.R. Saidi, G.H. Bagheri, M. Rahimpour, and D.G. Domairry, "Analytical solution for Van der Pol–Duffing oscillators," Chaos, Solitons and Fractals 42, pp. 2660-2666, March 2009.

[12] F.M. M. Kakmenia, S. Bowong, C. Tchawoua, and E. Kaptouom," Strange attractors and chaos control in a Duffing–Van der Pol oscillator with two external periodic forces," Journal of Sound and Vibration, 277, pp. 783-799, 2004.

[13] J. A. Khan, M. A. Z. Raja, and I. M. Qureshi,"An application of evolutionary computational technique to nonlinear singular system arising in polytrophic and isothermal sphere," Global Journal of researches in engineering numerical methods,Vol. 12, Iss. 1, Ver. 1.0, March 2012

[14] M. A. Z. Raja, J. A. Khan, I. M. Qureshi,"A new stochastic approach for solution of Riccati differential equation of fractional order," Annals of Mathematics and Artificial Intelligence, Jan 2011.

[15] J. A. Khan, M. A. Z. Raja, and I. M. Qureshi,"Swarm intelligence for the problems of nonlinear ordinary differential equations and its application to well known Wessinger's equation," European Journal of Scientific Research, Vol.34 no.4, pp. 514-525, 2009.

[16] J. A. Khan, I. M. Qureshi, and M. A. Z. Raja," Hybrid evolutionary computational approach: application to van der pol oscillator," International Journal of the Physical Sciences Vol. 6(31), pp. 7247 - 7261, Nov. 2011.

[17] C. Caetano, J. L. Reis Jr., J. Amorim, M. Ruvlemes, and A. Dalpino Jr.," Using neural networks to solve nonlinear differential equations in atomic and molecular physics," International Journal of Quantum Chemistry, Vol. 111, pp. 2732–2740, 2011.

[18] P. Bajpai, and M. KUMAR, " Genetic algorithm – an approach to solve global optimization problems," Indian Journal of Computer Science and Engineering, Vol. 1, no 3, 199-206.

[19] N. K. Karmarkar," A new polynomial-time algorithm for linear programming," Combinatorica, 4(4), pp. 373-395, Nov. 1984.

[20] R. H. Byrd, M. E. Hribar, and J. Nocedal, "An interior point algorithm for large-scale nonlinear programming," SIAM Journal on Optimization," SIAM Journal on Optimization, Vol 9, no. 4, pp. 877–900, 1999.

[21] G. Lesaja ," Introducing interior-point methods for introductory operations research courses and/or linear programming courses," The Open Operational Research Journal, 3, pp. 1-12, 2009.

[22] R. A. Waltz, J. L. Morales, J. Nocedal, and D. Orban, "An interior algorithm for nonlinear optimization that combines line search and trust region steps," Mathematical Programming, Vol 107, no. 3, pp. 391–408, 2006.

[23] C. Grosan, and A. Abraham," Hybrid Evolutionary Algorithms: methodologies, architectures, and reviews," Studies in Computational Intelligence (SCI) 75, pp. 1–17, 2007.

[24] M. Nasiraghdam, and M. Nafar," New approach based on hybrid GA and PSO as HGAPSO in low-frequency oscillation damping using UPFC controller," Journal of basic and Applied Scientific Research, 1(11), pp. 2208-2218, 2011.

[25] A. Ezzatneshan," A hybrid algorithm for vehicle routing problem," Journal of Basic and Applied Scientific Research, 1(11), pp. 1910-1919, 2011.

[26] P. Garg," A comparison between memetic algorithm and genetic algorithm for the cryptanalysis of simplified data encryption standard algorithm," International Journal of Network Security & Its Applications, Vol.1, no 1, April 2009.