SOLVING NONLINEAR DIFFERENTIAL EQUATIONS USING HYBRID METHOD BETWEEN LYAPUNOV’S ARTIFICIAL SMALL PARAMETER AND CONTINUOUS PARTICLE SWARM OPTIMIZATION

OMAR SABER QASIM*, AHMED ENTESAH AND WALEED AL-HAYANI

Department of Mathematics
College of Computer Science and Mathematics
University of Mosul, Mosul, Iraq

(Communicated by Gerhard-Wilhelm Weber)

ABSTRACT. In this paper, Lyapunov’s artificial small parameter method (LASPM) with continuous particle swarm optimization (CPSO) is presented and used for solving nonlinear differential equations. The proposed method, LASPM-CPSO, is based on estimating the ε parameter in LASPM through a PSO algorithm and based on a proposed objective function. Three different examples are used to evaluate the proposed method LASPM-CPSO, and compare it with the classical method LASPM through different intervals of the domain. The results from the maximum absolute error (MAE) and mean squared error (MSE) obtained through the given examples show the reliability and efficiency of the proposed LASPM-CPSO method, compared to the classical method LASPM.

1. Introduction. Nonlinear problems occur in almost every field in science, engineering and others. The exact solutions are very difficult to obtain for most strong nonlinear differential equations. To solve these problems, the approximate solutions can be obtained through some analytical methods such as the traditional perturbation method [20], Lyapunov’s artificial small parameter method (LASPM) [17, 18], Adomian decomposition method (ADM) [1–3], Homotopy analysis method (HAM) [6, 15, 16], Homotopy perturbation method (HPM) [13] and Variational iteration method (VIM) [14] have been proposed.

In 1892 Lyapunov [17,18] proposed the so-called artificial small parameter method for solving nonlinear differential equations to obtain the solution series converges to the exact solution in the whole region.

In the recent literature, Zhang and Liang [26] have demonstrated ADM is a special case of Lyapunov’s artificial small parameter method. In previous works, Andrianov et al. in 2005 proposed a new method (Artificial small parameter method) to solve the boundary value problems and they got good results by solving some mechanical examples [8]. Ahmad et al. used the least-squares method to solve the linear and nonlinear differential equations where they used new parameters to

2010 Mathematics Subject Classification. Primary: 58F15, 58F17; Secondary: 53C35.
Key words and phrases. Dimension theory, Poincaré recurrences, multifractal analysis.
*Corresponding author.
obtain less error by using a system of nonlinear equations [4]. Rawashdeh and Maitama in 2015 used a new method (NDM) to solve several types of nonlinear differential equations where a convergent series of the exact solution was obtained [23]. In 2020, Moaaz et al. studied the behavior of the oscillation of a type of from differential equations and set sufficient conditions to obtain better results through this study [19]. Also with regard to the algorithms of swarms, Ouyang et al. in 2009 proposed a hybrid algorithm between the Nelder–Mead simplex method (NMSM) and the Particle Swarm Optimization (PSO) algorithm to overcome the difficulty in selecting a good initial guess of SM that is used to solve systems of nonlinear equations [21]. Babaei used PSO algorithm to solve nonlinear differential equations through used a general approach to approximate solutions [9]. Entesar et al. in 2019 proposed the hybrid genetic algorithm with the HAM for solving fractional PDE [12].

The main objective of this paper is to use LASPM to solve differential equations for various examples to obtain accurate approximate solutions, where the basic idea in this paper depends on using the continuous particle swarm optimization algorithm to find the best value for the $\varepsilon$ parameter in LASPM, which in turn leads to improving the solutions and reducing the error in the proposed method compared to the default LASPM method. Most of the analytical methods depend on setting a specific value for the $\varepsilon$ parameter in LASPM, while the proposed LASPM-CPSO method uses an optimal value for the parameter through the CPSO algorithm.

In what follows, we give a brief review of LASPM and Particle Swarm Optimization (PSO).

2. Basic Ideas of LASPM. Let us consider a nonlinear equation

$$N [w (r, t)] = f (r, t),$$

where $N$ is a nonlinear operator, $w$ is a dependent variable, $f (r, t)$ is a known function, and $r$ and $t$ denote the spatial and temporal variables, respectively. Suppose that the nonlinear operator $N$ can be divided into

$$N = L_0 + N_0,$$

where $L_0$ is the linear operator and $N_0$ is the nonlinear operator. Using the Eq. (2) and introducing the artificial small parameter $\varepsilon$, the original Eq. (1) becomes

$$L_0 [\phi (r, t; \varepsilon)] + \varepsilon N_0 [\phi (r, t; \varepsilon)] = f (r, t),$$

where $\phi (r, t; \varepsilon)$ is an unknown function, $0 < \varepsilon \leq 1 + \delta$, where $\delta \geq 0$ and when $\varepsilon = 1$; the Eq. (3) is the same as Eq. (1) so that

$$\phi (r, t; 1) = w (r, t).$$

Expanding $\phi (r, t; \varepsilon)$ in a power series of the artificial small parameter $\varepsilon$, we have

$$\phi (r, t; \varepsilon) = w_0 (r, t) + \sum_{n=1}^{+\infty} \varepsilon^n w_n (r, t).$$

Setting $\varepsilon = 1$ in the Eq. (5) and using the Eq. (4), we get

$$w (r, t) = w_0 (r, t) + \sum_{n=1}^{+\infty} w_n (r, t).$$
Substituting Eq. (5) into the Eq. (3), we have
\[
L_0 [w_0 (r, t)] - f (r, t) + \sum_{n=1}^{+\infty} \varepsilon^n L_0 [w_n (r, t)] + \varepsilon N_0 \left[ w_0 (r, t) + \sum_{n=1}^{+\infty} \varepsilon^n w_n (r, t) \right] = 0. 
\]  
(7)

We write
\[
N_0 \left[ w_0 (r, t) + \sum_{n=1}^{+\infty} \varepsilon^n w_n (r, t) \right] = \sum_{n=0}^{+\infty} \varepsilon^n A_n (r, t), 
\]  
(8)

where, \( N_0 \) is the nonlinear operator, \( w \) is the dependent variable and \( w_0 \) is the initial solution. Differentiating both sides of the Eq. (8) \( m \) times with respect to the artificial small parameter \( \varepsilon > 0 \) and when setting \( \varepsilon = 0 \), we obtain
\[
\left. \left\{ \frac{\partial^m}{\partial \varepsilon^m} N_0 \left[ w_0 (r, t) + \sum_{n=1}^{+\infty} \varepsilon^n w_n (r, t) \right] \right\} \right|_{\varepsilon=0} = m! w_m (r, t), 
\]  
(9)

which implies that
\[
w_m (r, t) = \frac{1}{m!} \left. \left\{ \frac{\partial^m}{\partial \varepsilon^m} N_0 \left[ w_0 (r, t) + \sum_{n=1}^{+\infty} \varepsilon^n w_n (r, t) \right] \right\} \right|_{\varepsilon=0} = A_m (r, t), 
\]  
(10)

where \( A_m (r, t) \) are polynomials (so-called Adomian polynomials) [4–6]. So, substituting
\[
N_0 \left[ w_0 (r, t) + \sum_{n=1}^{+\infty} \varepsilon^n w_n (r, t) \right] = \sum_{n=0}^{+\infty} \varepsilon^n A_n (r, t) \]  
(11)

into Eq. (7), we have
\[
\left\{ L_0 [w_0 (r, t)] - f (r, t) \right\} + \sum_{n=1}^{+\infty} \varepsilon^n \left\{ L_0 [w_n (r, t)] + A_{n-1} (r, t) \right\} = 0, 
\]  
(12)

which gives
\[
L_0 [w_0 (r, t)] - f (r, t) = 0, 
\]  
(13)

and
\[
L_0 [w_n (r, t)] + A_{n-1} (r, t) = 0, \quad n \geq 1. 
\]  
(14)

Solving the Eqs. (13) and (14), we get
\[
w_0 (r, t) = L_0^{-1} [f (r, t)], 
\]  
(15)

and
\[
w_n (r, t) = -L_0^{-1} [A_{n-1} (r, t)], \quad n \geq 1. 
\]  
(16)

Thus all components of \( w \) can be calculated once the \( A_{n-1} \) are given for \( n = 1, 2, \ldots \).

3. Continuous Particle Swarm Optimization. The continuous particle swarm optimization (CPSO) algorithm is a population-based global search algorithm and solutions exchange, proposed by Kennedy and Eberhart (1995). The idea of the algorithm is inspired by the mimicry of fish schooling, bird flocking, where researches solutions in a continuous space and requires minimum standards but is generally effective in reaching the solution [22,24].

The CPSO algorithm is generally uncomplicated and its equations simple, and does not contain the processes of selection, crossover, and mutation such as the genetic algorithm. The CPSO algorithm has been successfully related references to find the optimal solution that is closest to the actual solution.
The initial population of CPSO starts with a swarm of random particles, and each particle has two properties of position and velocity for the search issue. The movement of particles in the search space is controlled by velocity and position, where the velocity equation is as follows [7, 25].

\[
v_{ij}^{t+1} = w v_{ij}^t + C_1 k_1 (p_{best_{ij}}^t - x_{ij}^t) + C_2 k_2 (g_{best_{ij}}^t - x_{ij}^t)
\]

where \( p_{best} \) represents the personal best particle, and \( g_{best} \) represents the global best particle values are identified. As well, the position can be calculated from the following equation:

\[
x_{ij}^{t+1} = x_{ij}^t + v_{ij}^{t+1}
\]

where \( t \) is a number of iterations, \( w \) represents the inertia weight ranging which is between 0 and 1, \( C_1 \) and \( C_2 \) are the cognitive memory and social learning factors, respectively and \( k_i, i = 1, 2 \) represent the random numbers uniformly chosen between 0 and 1, \( X_i = \{x_{i1}, x_{i2}, \ldots, x_{im}\}, i = 1, 2, \ldots, N, X_i \) represent the position vector, \( V_i = \{v_{i1}, v_{i2}, \ldots, v_{im}\}, i = 1, 2, \ldots, N, V_i \) represent the velocity vector, \( P_{best_i} = \{p_{i1}, p_{i2}, \ldots, p_{im}\}, i = 1, 2, \ldots, N, P_{best_i} \) represents the best solution found yet by particles, \( G_{best_i} = \{g_1, g_2, \ldots, g_m\}, G_{best} \) represents the best particle found so far in the swarm.

The general form of the target function can be written as:

\[
\text{Fitness function} \ (x) = \min G \ (x)
\]

where \( G(x) \) is a function that is chosen according to the type of problem to be solved.

In each iteration, particles are updated in the swarm through the velocity and position equations, where \( P_{best} \) (personal best) and \( G_{best} \) (global best) values are identified, and the algorithm stops updating the positions of the particles when they reach the maximum number of iterations (which was chosen as 10 iterations). A pseudo code for the CPSO can be written as follows [10]:

```
Begin
    Create swarm with N particles
    Initialize the position and velocities for each particle randomly
    Set \( t = 0 \)

    Repeat
        Set \( t = t + 1 \)
        Calculate fitness function for each particle of swarm
        Select \( p_{best} \) for each particle
        Select \( g_{best} \) from \( P(t-1) \)
        Compute velocity each particle \( V(t+1) \) from Eq. (17)
        Compute position each particle \( X(t+1) \) from Eq. (18)

    END

    Until the termination criteria (maximum iteration or tolerance) of problem satisfied.

END
```

4. Description of the proposed algorithm. The proposed method LASPM-CPSO is based on finding the best value for the LASPM using the CPSO algorithm. The result of the LASPM, which is as a series of approximate solutions, is used as an objective function to be introduced into the CPSO algorithm as follows:

\[
OF (\varepsilon) = \min \left( \frac{1}{m} \sum_{i=1}^{m} (u_x (x_i) - \phi (x_i))^2 \right), \tag{20}
\]
where $OF$ represents the objective function of CPSO algorithm, $\varepsilon$ the parameter of LASPM, $m$ the total numbers of steps used in the domain of $x$, $\phi$ represents the approximate solutions resulting from the LASPM and $u_e$ represents the exact solution for the given examples. The pseudo code of the proposed algorithm is given as follows:

\begin{verbatim}
Input: CPSO parameters, Lyapunov's approximation of the problem
Output: The optimal artificial small parameter $\varepsilon$

Begin
    Initialize the positions and velocities randomly
    Evaluate the objective function $OF(\varepsilon)$
    While (not terminating satisfied) do
        Compute $V(t+1)$ from Eq. (17)
        Compute $X(t+1)$ from Eq. (18)
    End
    Give the optimal parameter of artificial small parameter $\varepsilon$
    Complete the solution by LASPM
End
\end{verbatim}

There are many analytical methods used to solve nonlinear differential equations such as (ADM, HAM, and HPM) and all of these methods contain parameters within the analytical part of the solution and contribute significantly to accelerate and accuracy of the solution if it is selected appropriately. The problem with standard methods is to determine the value of the parameters (for example the $\varepsilon$ parameter of LASPM), since the designation value of this parameter to 1 may not give good solutions compared to the exact solution, for this a hybrid method was introduced to estimates the parameter value for the LASPM based on the CPSO algorithm. In this study, three different examples were used to evaluate the proposed method LASPM-CPSO, and compare it with the classical method LASPM through different intervals of the domain as follows (Example 1, Example 2, and Example 3):

4.1. Example 1. Firstly, let us consider the following nonlinear initial value problem

\[ w''(x) - 3w^5(x) = 0, \quad 0 \leq x \leq 1.5 \]

\[ w(0) = \frac{1}{2}, \quad w'(0) = -\frac{1}{8}. \]  

The exact solution for this problem is

\[ w_{\text{Exact}}(x) = \frac{1}{\sqrt{2x + 4}}. \]  

Introducing the artificial small parameter $\varepsilon$, the Eq. (21) becomes

\[ w''(x) - 3\varepsilon w^5(x) = 0. \]  

Substituting

\[ w(x) = \sum_{n=0}^{\infty} \varepsilon^n w_n(x), \]

into the Eq. (23), we get

\[ \sum_{n=0}^{\infty} \varepsilon^n w_n''(x) - 3\varepsilon \left( \sum_{n=0}^{\infty} \varepsilon^n w_n(x) \right)^5 = 0. \]  

(25)
From Eq. (25), collecting and comparing the coefficient of like powers of \( \varepsilon \), the following approximations are obtained:

- \( \varepsilon^0 : w''_0(x) = 0, \quad w_0(0) = \frac{1}{2}, \quad w'_0(0) = -\frac{1}{8} \)
- \( \varepsilon^1 : w''_1(x) = 3w^5_0, \quad w_1(0) = 0, \quad w'_1(0) = 0 \)
- \( \varepsilon^2 : w''_2(x) = 15w^4_0w_1, \quad w_2(0) = 0, \quad w'_2(0) = 0 \)
- \( \varepsilon^3 : w''_3(x) = 15w^3_0w_2 + 30w^2_0w^2_1, \quad w_3(0) = 0, \quad w'_3(0) = 0 \)
- \( \varepsilon^4 : w''_4(x) = 60w^3_0w_1w_2 + 15w^2_0w_3 + 30w^2_1w^2_1, \quad w_4(0) = 0, \quad w'_4(0) = 0. \)

Solving the system in Eq. (26), the first few components are given by:

\[
\begin{align*}
\varepsilon^0 & : w_0(x) = \frac{1}{2} - \frac{1}{8}x, \\
\varepsilon^1 & : w_1(x) = -\frac{458752}{16384}x^7 + \frac{3}{4096}x^6 - \frac{5}{1024}x^4 - \frac{5}{256}x^4 + \frac{3}{64}x^2, \\
\varepsilon^2 & : w_2(x) = -\frac{5}{97710505984}x^{13} + \frac{5}{1879048192}x^{12} - \frac{15}{234881024}x^{11} + \frac{275}{29360128}x^9 + \frac{495}{7340032}x^8 - \frac{325}{917504}x^7, \\
& \quad + \frac{43}{32768}x^6 - \frac{51}{16384}x^5 + \frac{15}{4096}x^4, \\
\varepsilon^3 & : w_3(x) = -\frac{145}{106459113847783424}x^{19} + \frac{145}{140077813786624}x^{18} - \frac{350194453446656}{1305}x^{17} + \frac{7395}{87548613361664}x^{16} - \frac{7395}{22185}x^{15} + \frac{4185}{1367947083776}x^{14} - \frac{48755}{765}x^{13} + \frac{48755}{541788335104}x^{12} - \frac{48755}{117440512}x^{11} + \frac{1835008}{1367947083776}x^{10} \\
& \quad - \frac{48755}{6215}x^9 + \frac{14680064}{58720256}x^8 - \frac{1695}{3670016}x^7 + \frac{51}{131072}x^6, \\
\varepsilon^4 & : w_4(x) = -\frac{55}{58720256}x^{10} - \frac{275}{29360128}x^9 + \frac{495}{7340032}x^8 - \frac{325}{917504}x^7 + \frac{43}{32768}x^6.
\end{align*}
\]

Thus the approximate solution in a series form is given by:

\[
\begin{align*}
\text{w}_{\text{Approx}} (x) &= \sum_{n=0}^{\infty} \varepsilon^n w_n (x) = w_0 (x) + \varepsilon w_1 (x) + \varepsilon^2 w_2 (x) + \varepsilon^3 w_3 (x) + O (\varepsilon^4) \\
&= \left( \frac{1}{2} - \frac{1}{8}x \right) + \varepsilon \left( -\frac{1}{458752}x^7 + \frac{1}{16384}x^6 - \frac{3}{4096}x^5 + \frac{5}{1024}x^4 - \frac{5}{256}x^4 + \frac{3}{64}x^2 \right) \\
&\quad + \varepsilon^2 \left( -\frac{5}{97710505984}x^{13} + \frac{5}{1879048192}x^{12} - \frac{15}{234881024}x^{11} + \frac{275}{29360128}x^9 + \frac{495}{7340032}x^8 - \frac{325}{917504}x^7 + \frac{43}{32768}x^6 \\
&\quad + \frac{55}{58720256}x^{10} - \frac{275}{29360128}x^9 + \frac{495}{7340032}x^8 - \frac{325}{917504}x^7 + \frac{43}{32768}x^6 \right).
\end{align*}
\]
The exact solution for this problem is 
\[ w(x) = 4.2. \]

Example 2.

Introducing the artificial small parameter \( \varepsilon \), the Eq. (29) becomes
\[ \frac{d^2 w}{dx^2} - 2e^{w(x)} \frac{d^2 w}{dx^2} - 2e^{w(x)} = 0. \]

The exact solution for this problem is
\[ w_{Exact}(x) = -2 \ln[\cos(x)]. \]

In Tables 1, 2 we present the maximum absolute and mean squared errors obtained by LASPM (4-iterations) and the Proposed Method with the exact solution.

| Domain | Proposed Method | LASPM, \( \varepsilon = 1 \) |
|--------|-----------------|-----------------------------|
| \([-1, 1]\) | \([-1, 0]\) \( 1 + 1.98 \varepsilon - 0.5 \) \( 3.60 \varepsilon - 0.5 \) \( 3.79 \varepsilon - 0.5 \) |
| \([-1, 1]\) | \([-0.5, 0]\) \( 1 + 2.60 \varepsilon - 0.9 \) \( 1.26 \varepsilon - 0.8 \) \( 1.27 \varepsilon - 0.8 \) |
| \([-0.5, 1]\) | \([0, 1]\) \( 1 + 3.10 \varepsilon - 0.7 \) \( 8.34 \varepsilon - 0.7 \) \( 8.45 \varepsilon - 0.7 \) |

| Domain | Proposed Method | LASPM, \( \varepsilon = 1 \) |
|--------|-----------------|-----------------------------|
| \([-1, 1]\) | \([-1, 0]\) \( 1 + 1.98 \varepsilon - 0.5 \) \( 1.26 \varepsilon - 10 \) \( 1.41 \varepsilon - 10 \) |
| \([-1, 1]\) | \([-0.5, 0]\) \( 1 + 2.60 \varepsilon - 0.9 \) \( 2.68 \varepsilon - 17 \) \( 2.70 \varepsilon - 17 \) |
| \([-0.5, 1]\) | \([0, 1]\) \( 1 + 3.10 \varepsilon - 0.7 \) \( 7.51 \varepsilon - 14 \) \( 7.78 \varepsilon - 14 \) |

4.2. Example 2. We consider the Bratu type initial value problem [5, 11]
\[ w''(x) - 2e^{w(x)} = 0, \quad 0 \leq x \leq 1 \]
\[ w(0) = 0, \quad w'(0) = 0. \]

The exact solution for this problem is
\[ w_{Exact}(x) = -2 \ln[\cos(x)]. \]

Introducing the artificial small parameter \( \varepsilon \), the Eq. (29) becomes
\[ w''(x) - 2\varepsilon e^{w(x)} = 0. \]
Substituting Eq. (24) into the Eq. (31), we get
\[
\sum_{n=0}^{\infty} \varepsilon^n w_n''(x) - 2\varepsilon \sum_{n=0}^{\infty} \frac{\varepsilon^n w_n(x)}{n!} = 0. \tag{32}
\]

Using the Taylor series expansion, which implies that
\[
\sum_{n=0}^{\infty} \varepsilon^n w_n''(x) - 2\varepsilon \left[ 1 + \sum_{n=0}^{\infty} \varepsilon^n w_n(x) + \frac{1}{2!} \left( \sum_{n=0}^{\infty} \varepsilon^n w_n(x) \right)^2 + \frac{1}{3!} \left( \sum_{n=0}^{\infty} \varepsilon^n w_n(x) \right)^3 + \cdots \right] = 0. \tag{33}
\]

From Eq. (33), collecting and comparing the coefficient of like powers of \( \varepsilon \), the following approximations are obtained
\[
\begin{align*}
\varepsilon^0 : & \quad w_0'' (x) = 0, \quad w_0 (0) = 0, \quad w_0' (0) = 0 \\
\varepsilon^1 : & \quad w_1'' (x) = 2 + 2w_0 + w_0^2 + \frac{1}{3} w_0^3, \quad w_1 (0) = 0, \quad w_1' (0) = 0 \\
\varepsilon^2 : & \quad w_2'' (x) = 2w_1 + w_0^2 w_1 + 2w_0 w_1, \quad w_2 (0) = 0, \quad w_2' (0) = 0 \\
\varepsilon^3 : & \quad w_3'' (x) = 2w_2 + w_0^2 + w_0 w_1^2 + w_0 w_1^2 + 2w_0 w_2, \quad w_3 (0) = 0, \quad w_3' (0) = 0 \\
\varepsilon^4 : & \quad w_4'' (x) = 2w_3 + w_0^2 w_3 + 2w_1 w_2 + 2w_0 w_1 w_2 + 2w_0 w_3 + \frac{1}{3} w_1^3, \quad w_4 (0) = 0, \quad w_4' (0) = 0.
\end{align*}
\]
\[
\vdots \tag{34}
\]

Solving the system in Eq. (34), the first few components are given by
\[
\begin{align*}
w_0 (x) & = 0, \\
w_1 (x) & = x^2, \\
w_2 (x) & = \frac{1}{6} x^4, \\
w_3 (x) & = \frac{2}{45} x^6, \\
w_4 (x) & = \frac{17}{1260} x^8, \\
\vdots & 
\end{align*} \tag{35}
\]

Thus the approximate solution in a series form is given by
\[
\begin{align*}
w_{\text{Approx}} (x) & = \sum_{n=0}^{\infty} \varepsilon^n w_n (x) = w_0 (x) + \varepsilon w_1 (x) + \varepsilon^2 w_2 (x) + \varepsilon^3 w_3 (x) \\
& \quad + \varepsilon^4 w_4 (x) + O \left( \varepsilon^5 \right) \\
& = \varepsilon x^2 + \frac{1}{6} \varepsilon^2 x^4 + \frac{2}{45} \varepsilon^3 x^6 + \frac{17}{1260} \varepsilon^4 x^8 + O \left( \varepsilon^5 \right). \tag{36}
\end{align*}
\]

In Tables 3, 4 we present the maximum absolute and mean squared errors obtained by LASPM (4-iterations) and the Proposed Method with the exact solution.
Substituting Eq. (24) into the Eq. (39), we get

\[ w' (x) + \sin (w (x)) = 0, \quad 0 \leq x \leq 1 \]

The exact solution for this problem is

\[ w (0) = c, \quad c = \frac{\pi}{2} \]  

\[ (37) \]

Introducing the artificial small parameter \( \varepsilon \), the Eq. (37) becomes

\[ w' (x) + \varepsilon \sin (w (x)) = 0. \]  

\[ (39) \]

Substituting Eq. (24) into the Eq. (39), we get

\[ \sum_{n=0}^{\infty} \varepsilon^n w'_n (x) + \varepsilon \sin \left( \sum_{n=0}^{\infty} \varepsilon^n w_n (x) \right) = 0. \]  

\[ (40) \]

Using the Taylor series expansion, which implies that

\[ \sum_{n=0}^{\infty} \varepsilon^n w'_n (x) + \varepsilon \left[ \sum_{n=0}^{\infty} \varepsilon^n w_n (x) - \frac{1}{3!} \left( \sum_{n=0}^{\infty} \varepsilon^n w_n (x) \right)^3 + \frac{1}{5!} \left( \sum_{n=0}^{\infty} \varepsilon^n w_n (x) \right)^5 - \cdots \right] = 0. \]  

\[ (41) \]
From Eq. (41), collecting and comparing the coefficient of like powers of \( \varepsilon \), the following approximations are obtained

\[ \varepsilon^0 : \quad w_0' (x) = 0, \quad w_0 (0) = c \]

\[ \varepsilon^1 : \quad w_1' (x) = -w_0 + \frac{w_0^3}{6} - \frac{w_0^5}{120} + \frac{w_0^7}{5040}, \quad w_1 (0) = 0 \]

\[ \varepsilon^2 : \quad w_2' (x) = -w_1 + \frac{w_0^2 w_1}{2} - \frac{w_0^4 w_1}{24} + \frac{w_0^6 w_1}{720}, \quad w_2 (0) = 0 \]

\[ \varepsilon^3 : \quad w_3' (x) = -w_2 - \frac{w_0^3 w_1^2}{12} + \frac{w_0^5 w_1^2}{24} - \frac{w_0^7 w_1^2}{240} + \frac{w_0^9 w_1^2}{12} - \frac{w_0^{11} w_1^2}{240} + w_0^1 w_1 w_2 + \frac{w_0^3 w_3}{2}, \quad w_3 (0) = 0 \]

\[ \varepsilon^4 : \quad w_4' (x) = -w_3 + \frac{w_0^3 w_3}{6} + \frac{w_0^5 w_3}{720} - \frac{w_0^7 w_3}{12} + \frac{w_0^9 w_3}{240} + \frac{w_0^{11} w_3}{12} - \frac{w_0^{13} w_3}{240} + w_0^1 w_1 w_2 + \frac{w_0^3 w_1 w_2}{2} + \frac{w_0^5 w_1 w_2}{120} + \frac{w_0^7 w_1 w_2}{144} - \frac{w_0^9 w_1 w_2}{24}, \quad w_4 (0) = 0 \]

\[ \vdots \quad (42) \]

Solving the system in Eq. (42), the first few components are given by

\[ w_0 (x) = c, \]

\[ w_1 (x) = -0.9998431014x \]

\[ w_2 (x) = -0.0004471913x^2 \]

\[ w_3 (x) = 0.1673681439x^3 \]

\[ w_4 (x) = -0.0006819336x^4 \]

\[ w_5 (x) = -0.0413229106x^5 \]

\[ \vdots \]

Thus the approximate solution in a series form is given by

\[ w_{\text{Approx}} (x) = \sum_{n=0}^{\infty} \varepsilon^n w_n (x) = w_0 (x) + \varepsilon w_1 (x) + \varepsilon^2 w_2 (x) + \varepsilon^3 w_3 (x) + \varepsilon^4 w_4 (x) + O (\varepsilon^6) \]

\[ = c + \varepsilon (-0.9998431014) + \varepsilon^2 (-0.0004471913x^2) + \varepsilon^3 (0.1673681439x^3) \]

\[ + \varepsilon^4 (-0.0006819336x^4) + \varepsilon^5 (-0.0413229106x^5) + O (\varepsilon^6) \].

In Tables 5, 6 we present the maximum absolute and mean squared errors obtained by LASPM (5-iterations) and the Proposed Method (CPSO) with the exact solution.

5. Conclusion. In this paper, a new approach has been introduced by hybridizing the LASPM with the CPSO algorithm. The optimum value of the \( \varepsilon \) parameter in LASPM is determined by the CPSO procedures, which in turn positively affects the improvement of solutions. The optimal value of the \( \varepsilon \) parameter in LASPM is determined by the CPSO procedure (mainly by relying on the objective function suggested in Eq. (23)) which in turn produces better solutions. In order to demonstrate the efficiency of the proposed LASPM-CPSO method, three different examples have been tested to illustrate the excellence of the LASPM-CPSO on the LASPM by relying on maximum absolute error (MAE) and mean squared error.
(MSE) criteria. The results in Tables (1-6) demonstrated that LASPM-CPSO is superior to LASPM in all examples of this study.

Table 5. Maximum absolute error for example 3

| Domain | Proposed Method | LASPM, $\varepsilon = 1$ |
|--------|-----------------|--------------------------|
| $[-1,1]$ | $\varepsilon$ | MAE | MAE |
| $[-1,0]$ | 0.9969649575 | $4.77E-03$ | $6.90E-03$ |
| $[-1,0.5]$ | 0.9974640403 | $5.12E-03$ | $6.90E-03$ |
| $[-1,1]$ | 0.9960515183 | $6.36E-03$ | $9.16E-03$ |
| $[-0.5,0]$ | 1.0004736382 | $3.69E-05$ | $2.44E-04$ |
| $[-0.5,0.5]$ | 1.0002174260 | $1.62E-04$ | $2.44E-04$ |
| $[-0.5,1]$ | 0.9960267479 | $6.34E-03$ | $9.16E-03$ |
| $[0,0.5]$ | 0.9999618650 | $4.83E-05$ | $6.52E-05$ |
| $[0,1]$ | 0.9951507666 | $5.72E-03$ | $9.16E-03$ |

Table 6. Mean squared error (MSE) for example 3

| Domain | Proposed Method | LASPM, $\varepsilon = 1$ |
|--------|-----------------|--------------------------|
| $[-1,1]$ | $\varepsilon$ | MSE | MSE |
| $[-1,0]$ | 0.9969649575 | $3.27E-06$ | $5.22E-06$ |
| $[-1,0.5]$ | 0.9974640403 | $2.47E-06$ | $3.59E-06$ |
| $[-1,1]$ | 0.9960515183 | $4.50E-06$ | $7.97E-06$ |
| $[-0.5,0]$ | 1.0004736382 | $6.48E-10$ | $1.80E-08$ |
| $[-0.5,0.5]$ | 1.0002174260 | $6.28E-09$ | $1.03E-08$ |
| $[-0.5,1]$ | 0.9960267479 | $4.10E-06$ | $6.88E-06$ |
| $[0,0.5]$ | 0.9999618650 | $7.32E-10$ | $8.45E-10$ |
| $[0,1]$ | 0.9951507666 | $4.96E-06$ | $9.99E-06$ |

Acknowledgments. This paper was supported by the College of Computer Science and Mathematics, University of Mosul, Mosul, Iraq.

REFERENCES

[1] G. Adomian, *Nonlinear Stochastic Operator Equations*, Academic Press, New York, 1986.
[2] G. Adomian, *Nonlinear Stochastic Systems Theory and Applications to Physics*, Kluwer Academic Publishers, Dordrecht, 1989.
[3] G. Adomian, *Solving Frontier Problems of Physics: The Decomposition Method*, Kluwer Academic Publishers, Dordrecht, 1994.
[4] K. Ahmad, A. Qaiser and A. H. Soori, Nonlinear least squares solutions of ordinary differential equations, *Mitteilungen Klosterneuburg*, 2 (2020).
[5] Y. Aksoy and M. Pakdemirli, New perturbation solutions for Bratu-type equations, *Computers and Mathematics with Applications*, 59 (2010), 2802–2808.
[6] W. Al-Hayani, L. Alzubaidy and A. Entesar, Solutions of singular IVP’s of Lane-Emden type by Homotopy analysis method with genetic algorithm, *Appl. Math. Inf. Sci.*, 11 (2017), 1–10.
[7] N. A. Al-Thanoon, O. S. Qasim and Z. Y. Algamal, Tuning parameter estimation in SCAD-support vector machine using firefly algorithm with application in gene selection and cancer classification, *Computers in Biology and Medicine*, 2018.
[8] I. V. Andrianov, J. Awrejcewicz and A. Ivankov, Artificial small parameter method-solving mixed boundary value problems, *Mathematical Problems in Engineering*, 2005 (2005), 325–340.
[9] M. Babaei, A general approach to approximate solutions of nonlinear differential equations using particle swarm optimization, *Applied Soft Computing*, 13 (2013), 3354–3365.
[10] N. Delgarm, B. Sajadi, F. Kowsary and S. Delgarm, Multi-objective optimization of the building energy performance: A simulation-based approach by means of particle swarm optimization (PSO), *Applied Energy*, 170 (2016), 293–303.
[11] S. Deniz and N. Bildik, Optimal perturbation iteration method for Bratu-type problems, *Journal of King Saud University-Science*, 30 (2018), 91–99.
[12] A. Entesar, O. Saber and W. Al-Hayani, Hybridization of genetic algorithm with homotopy analysis method for solving fractional partial differential equations, *Eurasian Journal of Science & Engineering*, 4 (2019).
[13] J.-H. He, *Homotopy perturbation technique*, *Comput. Methods Appl. Mech. Engrg.*, 178 (1999), 257–262.
[14] J.-H. He, Variational iteration method-a kind of non-linear analytical technique: some examples, *International Journal of Non-Linear Mechanics*, 34 (1999), 699–708.
[15] S. Liao, *Beyond Perturbation-Introduction to the Homotopy Analysis Method*, CRC Press LLC, 2004.
[16] S. Liao, *Homotopy Analysis Method in Nonlinear Differential Equations*, Springer, 2012.
[17] A. M. Lyapunov, *General Problem on Stability of Motion*, Taylor & Francis, London, 1992.
[18] A. M. Lyapunov, The general problem of the stability of motion, *Int. J. Control*, 553 (1992), 531–534.
[19] O. Moaaz, I. Dassios, O. Bazighifan and A. Muhib, Oscillation theorems for nonlinear differential equations of fourth-order, *Mathematics*, 8 (2020), 520.
[20] A. H. Nayfeh, *Perturbation Methods*, John Wiley & Sons, New York, 2000.
[21] A. Ouyang, Y. Zhou and Q. Luo, Hybrid particle swarm optimization algorithm for solving systems of nonlinear equations, in *2009 IEEE International Conference on Granular Computing*, (2009), 460–465.
[22] O. S. Qasim and Z. Y. Algamal, Feature selection using particle swarm optimization-based logistic regression model, *Chemometrics and Intelligent Laboratory Systems*, 182 (2018), 41–46.
[23] M. S. Rawashdeh and S. Maitama, Solving nonlinear ordinary differential equations using the NDM, *J. Appl. Anal. Comput.*, 5 (2015), 77–88.
[24] Y. Shi, Particle swarm optimization: developments, applications and resources, in *Evolutionary Computation, Proceedings of the 2001 Congress* (2001), 81–86.
[25] Y. Shi and R. C. Eberhart, Parameter selection in particle swarm optimization, in *International Conference on Evolutionary Programming*, (1998), 591–600.
[26] X. Zhang and S. Liang, Adomian decomposition method is a special case of Lyapunov’s artificial small parameter method, *Applied Mathematics Letters*, 48 (2015), 177–179.

Received March 2020; 1st revision August 2020; Final revision December 2020.

E-mail address: omar.saber@uomosul.edu.iq
E-mail address: ahmed_entesar84@uomosul.edu.iq
E-mail address: waleedalhayni@uomosul.edu.iq