

## On the Formulation of Metaheuristic Algorithm-Based Approximation Approach for Nonlinear Ordinary Differential Equations with Application to Heat Exchanger Problem

Ee Soong Low and Pauline Ong\*

*Faculty of Mechanical and Manufacturing Engineering, Universiti Tun Hussein Onn Malaysia (UTHM), 86400 Parit Raja, Batu Pahat, Johor, Malaysia*

### ABSTRACT

The problems that arise in multitudinous fields often involve solving complex nonlinear ordinary differential equations (ODE), and it remains challenging since the actual solutions to these problems are hard to obtain. In this regard, the solution strategy with the formulation of Fourier series expansion, calculus of variation and metaheuristic algorithm, was introduced to determine the approximate solution of the nonlinear ODE. The nonlinear ODE was formulated as an optimization problem, specifically, the moth-flame optimization (MFO) algorithm and flower pollination algorithm (FPA) were utilized to find the coefficients of the Fourier series. This paper aimed to determine the feasibility of the proposed method to solve the ODEs with different characteristics and compare the obtained results with other optimization algorithms. Moreover, the suitable number of terms (NT) of Fourier series were determined for different test problems for MFO and FPA. The quantitative analysis in terms of the generational distance (GD) metric demonstrated that the approximate solutions were reasonably accurate, with the low GD within the range

of 1E-03 to 1E-05 for all test problems. The comparative analysis showed that the approximate performances of MFO and FPA were superior to or comparable with the genetic algorithm, particle swarm optimization and water cycle algorithm.

### ARTICLE INFO

#### *Article history:*

Received: 6 May 2020

Accepted: 5 June 2020

Published: 21 October 2020

DOI: <https://doi.org/10.47836/pjst.28.4.06>

#### *E-mail addresses:*

[ongp@uthm.edu.my](mailto:ongp@uthm.edu.my) (Pauline Ong)

[low\\_es@hotmail.com](mailto:low_es@hotmail.com) (Ee Soong Low)

\*Corresponding author

*Keywords:* Flower pollination algorithm, Fourier series, metaheuristic algorithm, moth-flame optimization algorithm, ordinary differential equations

## INTRODUCTION

Ordinary differential equations (ODE) have found a widespread application in modelling various dynamic systems, for instance, the membrane distillation process (Perfilov et al., 2018) and Human Immunodeficiency Virus model (Tian et al., 2019), to name a few. A  $n$ -th order nonlinear ODE can be expressed as Equation 1:

$$F(x, y, y', y'', \dots, y^{(n)}) = 0 \quad (1)$$

where  $y$  is a function of a single independent variable  $x$  and  $y^{(n)}$  is the  $n$ -th order derivative with respect to  $x$ .

Solving the nonlinear ODE is a challenging task, especially using analytical approaches, in which the solutions of the complex nonlinear ODE may be too complicated to obtain. Therefore, numerical methods, such as Euler and Runge-Kutta, are applied to determine the approximate solutions of the nonlinear ODE. Numerical methods, however, have their operational limitations. For example, Euler's method is numerically unstable, especially for stiff equations (Lambert, 1986). A finer discretization is required for smaller approximation error, and unavoidably, leads to higher computation load (Hussain et al., 2016). Moreover, most of the numerical methods have large cumulative errors due to the rounding-off error, which increases quickly with finer discretization (Butcher, 2016).

In addition to numerical methods, variant of analytical methods, exemplarily, variational iteration method (VIM) (He, 1999), differential transformation method (DTM) (Hussin et al., 2016), homotopy perturbation method (HPM) (He, 2003), and Adomian decomposition method (ADM) (Adomian, 1988), have been utilized to determine the approximate solutions of the ODE. Each of these methods has its limitations, albeit it gains prominence in solving the modeling problems in various fields (Das & Kundu, 2019; El-Sayed & El-Mongy, 2019; Turkyilmazoglu, 2018).

VIM – a modified Lagrange multiplier method based on the variational theory, was initially introduced by He to solve the wave equation (He, 1999). This method is of high calculation complexity due to the repetitive calculation of unnecessary terms (Abassy et al., 2007) and subjected to inherent inaccuracy in solving the Lagrange multiplier (Mohammadi & Hosseini, 2011). DTM, though based on the Taylor series, is distinct in the sense that it does not require the explicit calculation of high-order derivatives as in the Taylor series, which may not always formidable (Chen & Liu, 1998). The DTM, instead, iteratively calculates the derivatives using the differential transformation, but some drawbacks arise, such as the inefficiency in handling the oscillatory systems (Ebaid, 2011) and differential equation with complex nonlinearities (Ebaid, 2013). HPM is an improved classic perturbation method based on the idea of homotopy in topology (He, 2003). The HPM is limited in the way that the initial parameter has to be properly assigned to achieve a good approximation (Babolian et al., 2009). The salient feature of the ADM lies in its ability

to solve nonlinear differential equations without system linearization due to the utilization of the Adomian polynomials (Adomian, 1988). This method is somehow computationally complicated due to the calculation of the Adomian polynomials (Mohammadi & Hosseini, 2011), and prone to divergence when solving ODEs of Emden–Fowler type (Biazar & Hosseini, 2017).

In heat exchanger problems, finned surfaces are often used to increase the rate of heat transfer. In one-dimensional fin analysis, the energy transfer process can be modelled as an ODE if there is no heat variation in the fin cross-section. Otherwise, a partial differential equation will be formed (Torabi et al., 2013). The study of thermal conductivity of fins has been conducted by many researchers. Solving the nonlinear fin problems using numerical approach was studied by Bouaziz et al. (2001). In the analysis of thermal conductivity of convective radial and straight fins, VIM was used and compared with finite element analysis (Coşkun & Atay, 2007; Coşkun & Atay, 2008). Straight fins with different heat transfer coefficients problems were approximately solved using the VIM (Fouladi et al., 2010). To assess the temperature distribution of fins, Domairry and Fazeli (2009) applied the homotopy analysis method (HAM) to solve the differential equations formed by the nonlinear straight fins. Besides, HAM was used to find the approximate solutions for fin efficiency and temperature distribution in the study conducted by Khani et al. (2009). Other than rectangular fins, concave parabolic and trapezoidal fins were studied for space and airborne applications (Torabi et al., 2013).

The limitations of numerical methods and analytical methods prompt the investigation on the suitability to apply the metaheuristic optimization technique to solve the nonlinear ODE. Literature review reveals the credibility of using genetic algorithm (GA) (Mateescu, 2006), particle swarm optimization (PSO) (Babaei, 2013; Nemati et al., 2015), water cycle algorithm (WCA) (Sadollah et al., 2015a), harmony search algorithm (Sadollah et al., 2017), imperialist competitive algorithm (Almasi et al., 2016) and to the more recent, orthogonal colliding bodies optimization (OCBO) (Panda & Pani, 2017) in this regard.

Solving the ODE using the metaheuristic approach, generally, requires the formulation of the ODE as an optimization problem. A base approximate function is used as the approximate solution of the ODE. Subsequently, a population of self-organized agents collaboratively works to find the approximate solution of the ODE based on some rules. Various functions can be applied as the base function of the approximate solution, such as Runge-Kutta method (Mateescu, 2006), method of bilaterally bounded (Lee, 2006), finite element method (Mastorakis, 2006), constrained integration approach (Rudd & Ferrari, 2015), collocation method (Beidokhti & Malek, 2009) and Fourier series (Babaei, 2013). In this study, the Fourier series is chosen as it can decompose any periodic function into the sum of sine and cosine functions. Hence, it can approximate any form of ODE using the same procedure. Moreover, in comparison with the  $n$ -th order polynomial base function

which can only be differentiated up to  $n$ -th times, the Fourier series can be infinitely differentiated since it only involves the sine and cosine functions.

The Fourier series is made up of the sum of sine and cosine functions along with the unknown coefficients. In this study, the unknown coefficients of the Fourier series are optimized using the moth-flame optimization (MFO) algorithm and flower pollination algorithm (FPA), in which minimizing the weight error function of the ODE is formed as the objective function of the metaheuristic algorithms. The obtained approximate solutions in solving the well-known ODEs and the longitudinal heat transfer fins problem using the MFO and FPA are compared with the results from other optimizers, namely, PSO, WCA, GA and OCBO, in terms of the generational distance (GD) metric and mean squared error (MSE).

The paper is organized as follows. A brief introduction of the metaheuristic algorithms used in this study – MFO and FPA, is presented in metaheuristic algorithm section. Methods section 3 describes the solution strategy of formulating the approximate solution of ODE using the Fourier series. In results and discussion section, the experimental simulations on solving the well-known ODEs and the longitudinal heat transfer fins problem are presented. The relationship of using the different number of terms in Fourier series to the approximation accuracy is also analyzed. Lastly, some conclusions are drawn.

## METAHEURISTIC ALGORITHM

### Flower Pollination Algorithm

The flower pollination process inspires the development of FPA (Yang, 2012). There exist two types of pollination process: self-pollination, which does not require a pollinator to transfer the pollen; and cross-pollination, which needs the pollinator such as bees and birds to travel for a long-distance while obeying the Lévy flight behaviour (Dash et al., 2016). To simplify the search process in the FPA, the following four rules are assumed:

(i) The cross-pollination corresponds to the global search using Lévy flight, which is modelled as Equation 2:

$$x_i^{t+1} = x_i^t + \gamma L(\lambda)(x_i^t - g_*) \quad (2)$$

where  $x_i^t$  is pollen  $i$  at iteration  $t$ ,  $g_*$  is the best solution at current iteration,  $\gamma$  is a scaling factor to control the step size and  $L(\lambda)$  is a parameter corresponding to the strength of pollination. The Lévy distribution is described as Equation 3:

$$L \propto \frac{\lambda \Gamma(\lambda) \sin(\pi\lambda/2)}{\pi} \frac{1}{s^{1+\lambda}}, (s \propto s_o > 0) \quad (3)$$

where  $\Gamma(\lambda)$  is a standard gamma function.

(ii) The self-pollination acts as the local search, which may involve the same flower or different flowers of the same plant. This exploitation process is expressed as Equation 4:

$$x_i^{t+1} = x_i^t + \mathcal{E} (x_j^t - x_k^t) \tag{4}$$

where  $x_j^t$  and  $x_k^t$  are pollens from different flowers and  $\mathcal{E}$  is a random number from uniform distribution  $[0,1]$ .

(iii) The degree of similarity between two flowers is proportional to the reproduction ratio, which is considered as the flower constancy.

(iv) The switching probability  $p \in (0,1)$  is used to control the probability of local and global search.

The pseudocode of FPA is summarized in Figure 1.

```

Flower Pollination Algorithm
Determine the objective function,  $f(x), x = (x_1, x_2, \dots, x_d)$ 
Define population size,  $n$  and switch probability,  $p$ 
Define maximum iteration,  $max\_iter$ 
Initialize a population of  $n$  flowers randomly
Identify the global best solution,  $g$ .
Define iteration,  $t = 1$ 
while ( $t < max\_iteration$ )
    for  $i = 1:n$  (all  $n$  flowers in the population)
        if  $rand > p$ 
            Draw a (d-dimensional) step vector  $L$  which obeys a Lévy distribution
            Do global search using  $x_i^{t+1} = x_i^t + L(x_i^t - g)$ 
        else
            Pick  $e$  between 0 and 1 randomly
            Randomly choose  $j$  and  $k$  among all the solutions
            Do local search using  $x_i^{t+1} = x_i^t + \mathcal{E}(x_j^t - x_k^t)$ 
        end if
        Evaluate fitness of new solution ( $fmin$ )
        If  $fmin$  is better than previous solution, update the  $i$  solution and  $fmin$  in the population
        If  $fmin$  is better than global best solution, update global best solution and its fitness
    end for
    Record the current global best solution
     $t = t + 1$ 
end while
    
```

Figure 1. The pseudocode of FPA

## Moth Flame Optimization

The moth performs a spiral path around a light source in the night. Inspired by the transverse orientation of such navigation method, the MFO was proposed (Mirjalili, 2015).

In the MFO, the moths are the search agents (candidates of the solutions), the positions of moths represent the variables of the underlying problem, while the flame is the best position found by the moth. Through constantly changing the position vector, the moths can fly in multiple dimensional spaces. After the best position is identified, it acts as a flame and moth will start searching around the flame to search for a better solution.

The MFO algorithm consists of three main functions: I, P and T. The function I is used to generate a random position within the search space. The function P moves the moths in a spiral pattern, while the function T terminates the operation when the stopping conditions are achieved.

The logarithmic spiral within the function P is modelled as Equation 5:

$$S(M_i, F_j) = D_i \cdot e^{bt} \cdot \cos(2\pi t) + F_j \quad (5)$$

where  $D_i$  is the distance between the  $i$ -th moth and the  $j$ -th flame,  $b$  is a constant used to define the shape of the logarithmic spiral and  $t$  is a random number within the range  $[-1, 1]$ . The spiral or the motion path of the moth should start from the initial position of a moth and end at the position of a flame. The distance between both flame and moth can be determined by manipulating the parameter  $t$  in Equation 5, where  $t$  varies from -1 (closest) to 1 (farthest).

For each iteration, the flame list is sorted from the best to the worst solution. The moth is confined to update its position by referring to only one of the flames. This is to avoid the solution from getting trapped in the local optima. Besides, to preserve better convergence towards the end of the search process, the number of flames decreases as the number of iteration increases, according to Equation 6:

$$Flame\_no = round\left(N - I * \frac{N-1}{T}\right) \quad (6)$$

Here,  $I$  is the current iteration number,  $N$  is the maximum number of flames and  $T$  is the maximum number of iterations. The flow chart of MFO algorithm is illustrated in Figure 2.

## METHODS

### Solving the Ordinary Differential Equation using Fourier Series

Consider an ODE of order  $n$  as in Equation 1 with initial conditions (Equation 7):

Moth Flame Optimization Algorithm
Initialize random moths' position
<b>while</b> ( $I < T$ )
Evaluate each moth's fitness
Sort moths and moths' fitness according to fitness
Update $t$
Calculate D
Update moths' position using logarithmic spiral
Update number of flame
$I = I + 1$
<b>end while</b>

Figure 2. The pseudocode of MFO

$$\begin{aligned}
 y(x_o) &= y_o \\
 y'(x_o) &= y'_o \\
 &\vdots \\
 y^{(n)}(x_o) &= y_o^{(n)}
 \end{aligned}
 \tag{7}$$

or boundary conditions (Equation 8):

$$\begin{aligned}
 y(x_o) &= y_o; y(x_n) = y_n \\
 y'(x_o) &= y'_o; y'(x_n) = y'_n \\
 &\vdots \\
 y^{(n)}(x_o) &= y_o^{(n)}; y^{(n)}(x_n) = y_n^{(n)}
 \end{aligned}
 \tag{8}$$

Let the approximate solution of the ODE represented by the Fourier series and centered at  $x_0$  to be expressed in the general form of in Equation 9:

$$y(x) \approx Y_{app}(x) = a_o + \sum_{m=1}^{NT} [a_m \cos(\frac{m\pi(x-x_o)}{L}) + b_m \sin(\frac{m\pi(x-x_o)}{L})]
 \tag{9}$$

where  $a_o$ ,  $a_m$  and  $b_m$  are the Fourier coefficients, and NT is number of terms.  $L$  represents the length of interval span  $x_0$  to  $x_n$ .

The derivatives of Equation 9 are thus given by Equation 10:

$$\begin{aligned}
 y'(x) &\approx Y_{app}'(x) \\
 &= \sum_{m=1}^{NT} \left[ -\frac{m\pi}{L} a_m \sin\left(\frac{m\pi(x-x_o)}{L}\right) + \frac{m\pi}{L} b_m \cos\left(\frac{m\pi(x-x_o)}{L}\right) \right] \\
 y''(x) &\approx Y_{app}''(x) \\
 &= \sum_{m=1}^{NT} \left[ -\left(\frac{m\pi}{L}\right)^2 a_m \cos\left(\frac{m\pi(x-x_o)}{L}\right) - \left(\frac{m\pi}{L}\right)^2 b_m \sin\left(\frac{m\pi(x-x_o)}{L}\right) \right]
 \end{aligned} \tag{10}$$

The approximate solution (Equation 9) and its derivatives (Equation 10) are then substituted into Equation 1, creating an error function of  $E(x)$  as Equation 11:

$$E(x) = F(x, Y_{app}, Y_{app}', Y_{app}'', \dots, Y_{app}^{(n)}) \tag{11}$$

Subsequently, the weighted error function (WEF) given as Equation 12:

$$WEF = \int_D |W(x)| \times |E(x)| dx \tag{12}$$

is formulated, where  $W(x)$  is named as the weight function. The WEF is then utilized as the objective function of the metaheuristic algorithm, in which the metaheuristic algorithm attempts to find the optimal combination of Fourier coefficients such that the error is minimized. In this study, the evaluation of integration in Equation 12 is solved using the Trapezoidal integration method. The weight function is assumed to be 1 for simplicity. Intuitively, the smaller the value of WEF, the better the approximation accuracy is.

The approximate solution should satisfy the ODE on one hand, and the initial or boundary conditions on another hand. Since solving the ODE using the metaheuristic algorithm is formulated as an optimization problem, the initial or boundary conditions are formed as the constraints of the optimization problem. For homogeneous conditions, the constraints for initial conditions are expressed as Equation 13:

$$\begin{aligned}
 y(x_0) = 0 &\Rightarrow g_1(x_0) = |Y_{app}(x_0)| \\
 y'(x_0) = 0 &\Rightarrow g_2(x_0) = |Y_{app}'(x_0)| \\
 &\vdots \\
 y^{(n)}(x_0) = 0 &\Rightarrow g_{n+1}(x_0) = |Y_{app}^{(n)}(x_0)|
 \end{aligned} \tag{13}$$

or the constraints for boundary conditions are expressed as Equation 14:

$$\begin{aligned}
 y(x_0) = 0 &\Rightarrow g_1(x_0) = |Y_{app}(x_0)|, y(x_n) = 0 \Rightarrow g_1(x_n) = |Y_{app}(x_n)| \\
 y'(x_0) = 0 &\Rightarrow g_2(x_0) = |Y_{app}'(x_0)|, y'(x_n) = 0 \Rightarrow g_2(x_n) = |Y_{app}'(x_n)| \\
 &\vdots \\
 y^{(n)}(x_0) = 0 &\Rightarrow g_{n+1}(x_0) = |Y_{app}^{(n)}(x_0)|, y^{(n)}(x_n) = 0 \Rightarrow g_{n+1}(x_n) = |Y_{app}^{(n)}(x_n)|
 \end{aligned} \tag{14}$$

For nonhomogeneous conditions, the constraints for initial conditions are written as Equation 15:

$$\begin{aligned}
 y(x_0) = y_0 &\Rightarrow g_1(x_0) = |Y_{app}(x_0) - y_0| \\
 y'(x_0) = y_0' &\Rightarrow g_2(x_0) = |Y_{app}'(x_0) - y_0'| \\
 &\vdots \\
 y^{(n)}(x_0) = y_0^{(n)} &\Rightarrow g_{n+1}(x_0) = |Y_{app}^{(n)}(x_0) - y_0^{(n)}|
 \end{aligned}
 \tag{15}$$

or the constraints for boundary conditions are written as Equation 16:

$$\begin{aligned}
 y(x_0) = y_0 &\Rightarrow g_1(x_0) = |Y_{app}(x_0) - y_0|, \\
 y(x_n) = y_n &\Rightarrow g_1(x_n) = |Y_{app}(x_n) - y_n| \\
 y'(x_0) = y_0' &\Rightarrow g_2(x_0) = |Y_{app}'(x_0) - y_0'|, \\
 y'(x_n) = y_n' &\Rightarrow g_2(x_n) = |Y_{app}'(x_n) - y_n'| \\
 &\vdots \\
 y^{(n)}(x_0) = y_0^{(n)} &\Rightarrow g_{n+1}(x_0) = |Y_{app}^{(n)}(x_0) - y_0^{(n)}|, \\
 y^{(n)}(x_n) = y_n^{(n)} &\Rightarrow g_{n+1}(x_n) = |Y_{app}^{(n)}(x_n) - y_n^{(n)}|
 \end{aligned}
 \tag{16}$$

The penalty approach is employed for constraint handling in this study. The penalty function value (PFV) of the violated solution is multiplied with a penalty coefficient,  $\lambda$ , and added to the WEF as a penalty function. The fitness function value (FFV) of the optimization problem is thus given as Equation 17 and 18:

$$FFV = WEF + PFV \tag{17}$$

$$PFV = \lambda \sum_{m=1}^{nIC+nBC} g_m^2 \tag{18}$$

where  $nIC$  and  $nBC$  are number of initial conditions (IC) and boundary conditions (BC), respectively. The penalty coefficient,  $\lambda$ , is assigned as 10 for all considered ODEs in this study.

### Problem Solution Strategy

The study aims to solve the nonlinear ODE using Fourier series approximation approach, in which the metaheuristic algorithm is used to optimize the Fourier coefficients such that the FFV is minimized. Hence, solving the ODE can be formulated as Equation 19:

$$\text{Minimize } FFV = WRF + PFV \tag{19}$$

subjected to the homogeneous and/or nonhomogeneous initial and/or boundary constraints as in Equation 20 and 21:

$$g_i^{IVP}(x_0) = \begin{cases} 0, & i = 1, 2, \dots, nIC \\ y_0, & i = 1, 2, \dots, nIC \end{cases} \quad (20)$$

$$g_i^{BVP}(x_n) = \begin{cases} 0, & i = 1, 2, \dots, nBC \\ y_n, & i = 1, 2, \dots, nBC \end{cases} \quad (21)$$

This proposed solution strategy is implemented as follows:

- Step 1: Rewrite the ODE in an implicit form, as in Equation 1.
- Step 2: Rewrite the initial and/or boundary condition in the form of Equation 13 and 15.
- Step 3: Select an appropriate value of NT for Fourier series, as in Equation 9.
- Step 4: Assign a variable to each Fourier coefficients.
- Step 5: Apply the metaheuristic algorithm as the optimizer to determine the optimal combination of Fourier coefficients in Equation 9 such that the FFV in Equation 17 is minimized.
- Step 6: Calculate the MSE and GD metric to assess the fitness of the approximate solution in comparison with the exact solution. The GD metric is given as Equation 22 (Sadollah et al., 2015c):

$$GD = \frac{\left( \sum_{i=1}^n d_i^2 \right)^2}{n} \quad (22)$$

where  $n$  is the number of points and  $d$  is the Euclidean distance between point  $i$  in the approximate solution and the nearest point in the exact solution. In this study,  $n$  is chosen as 200 for all case studies.

The flowchart of the proposed strategy in solving the nonlinear ODE using the Fourier series and metaheuristic algorithm is summarized in Figure 3.

## RESULTS AND DISCUSSION

### Numerical Simulation

In this section, the well-known ODEs with different characteristics were used to evaluate the applicability of the approximate solutions obtained by the FPA and MFO algorithm. Specifically, test problems 1 to 3 are the well-known ODEs in science and engineering, test problem 4 is an ODE with Neumann boundary condition, while two mechanical vibration problems (mass-spring system) without (test problem 5) and with damping (test problem 6) were used to evaluate the effectiveness of the proposed approach as well. Also, the ODE of the longitudinal fins was studied in test problem 7. The illustrative comparison between the obtained solution and the actual solution was presented in addition to the obtained GD metric and MSE. Additionally, the relationship of using different NTs to the approximation accuracy was explored.

All simulations were repeated for 30 runs. For each run, the simulation was continued until the maximum number of iteration was reached. Next, the best results of all algorithms among 30 runs were reported. Except for test problem 2, the lower and upper bounds of the unknown Fourier coefficients were set between -1 and 1, while the search space of the unknown Fourier coefficients in test problem 2 was restricted to -2 and 2.

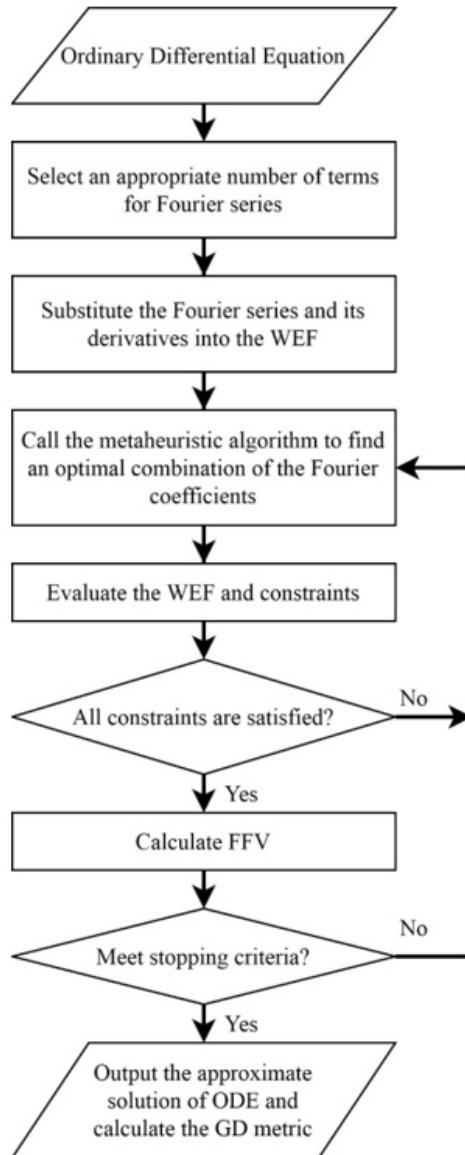


Figure 3. The flowchart of the proposed solution strategy

### Performance Comparison between Different Optimization Algorithms

In this study, the FPA, MFO and GA were used to optimize the coefficients of the Fourier series, with the WEF formulated as the cost function of the optimization algorithm. Since finding the approximate solutions of the aforementioned test problems using PSO, WCA and OCBO has been conducted in (Babaei, 2013; Panda & Pani, 2018; Sadollah et al., 2015b; Sadollah et al., 2015d) and hence, the results were included for comparative purpose. The NT of Fourier series in Equation 9 was assigned in accordance with the reported studies. Table 1 summarizes the NT, population size ( $n$ ) and the maximum number of iteration used in all optimization algorithms. The value was left as NA if the corresponding information was not available (NA) in the respective study. Table 2 lists the parameters setting used in all metaheuristic algorithms. It is pertinent to note that the parameters of OCBO are not provided in this table as they are not stated in the respective study (Panda & Pani, 2018).

Table 1

*Initial parameter of all optimization algorithms for comparative analysis*

Test Problem	NT	WCA		PSO		OCBO		FPA		MFO		GA	
		n	Max Iter	n	Max Iter	n	Max Iter	n	Max Iter	n	Max Iter	n	Max Iter
1	6	50	500	200	1000	50	500	50	700	300	1000	50	700
2	3	200	200	200	200	200	200	200	500	500	500	200	500
3	3	50	500	300	1000	50	500	50	800	50	500	50	800
4	6	200	100	NA	NA	NA	NA	200	500	200	300	200	500
5	6	NA	NA	200	200	NA	NA	200	500	200	400	200	500
6	8	NA	NA	200	200	NA	NA	200	1000	200	600	200	1000
7	3	NA	NA	300	1000	NA	NA	50	1000	50	1000	50	1000

Table 2

*The parameter setting used in all optimization algorithms*

Algorithm	Parameter
WCA	Number of sea and river, $N_{sr} = 8$
PSO	Inertia weight, $\omega = 0.9$
	Cognition component, $c_1 = 0.5$
	Social component, $c_2 = 1.5$
FPA	Switching probability, $P = 0.5$

Table 2 (Continued)

Algorithm	Parameter
FPA	$\lambda = 1.5$
MFO	$b = 1$
GA	Crossover percentage, $pc = 0.7$
	Mutation percentage, $pm = 0.3$
	Mutation rate, $mu = 0.1$
	Random selection
Euler	Number of steps, $N = 200$

The performance indicator of GD metric, which has been used in several studies to evaluate the optimization performance of each algorithm quantitatively, was utilized in this study (Panda & Pani, 2018; Sadollah et al., 2015b; Sadollah et al., 2015c). The GD measures the total Euclidean distance between the approximate solution and the exact solution. The approximate and the exact solutions were divided into 200 points and the Euclidean distance between all the points of the approximate solution and the exact solution was calculated. The lower the GD, the closer the approximate solution to the exact solution. Besides GD metric, the MSE was also used to evaluate the performance of all optimization algorithms.

**Test Problem 1: Integro-Differential Equation.** The integro-differential equation is given by Equation 23 (Sadollah et al., 2015c):

$$y' + 2y + 5 \int_0^x y(t) dt = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{23}$$

with initial condition  $y(0) = 0$ . The solution interval of  $x$  is chosen from 0 to  $\pi$ . The exact solution of this Initial Value Problem (IVP) is given by Equation 24:

$$y(x) = \frac{1}{2} e^{-x} \sin(2x) \tag{24}$$

The NT of the Fourier series was fixed as 6. Hence, the WCA, PSO, OCBO, GA, MFO and FPA optimization approaches have to determine the best combination of  $a_0, a_1, b_1, a_2, b_2, a_3, a_4, b_4, a_5, b_5, a_6$  and  $b_6$  of Equation 25, such that the residual error is minimized.

$$Y_{app}(x) = a_0 + a_1 \cos(x) + b_1 \sin(x) + a_2 \cos(2x) + b_2 \sin(2x) + a_3 \cos(3x) + b_3 \sin(3x) + a_4 \cos(4x) + b_4 \sin(4x) + a_5 \cos(5x) + b_5 \sin(5x) + a_6 \cos(6x) + b_6 \sin(6x) \tag{25}$$

Table 3 summarizes the best approximate solutions obtained by the WCA, PSO, OCBO, GA, MFO and FPA, while Figure 4 presents the comparison between the actual solution and the obtained best approximate solution of each algorithm. As shown in Figure 4, all the approximate solutions were close to the exact solutions. However, by visual inspection, the approximate solutions of MFO, FPA and GA deviate from the actual curve from  $x = 2.5$  onward. Table 4 summarizes the values of the GD metric for all the considered algorithms. It can be observed that Euler's method can approximate the solution more accurately than the others, indicated by the lowest GD, while the GA gave the lowest accuracy. The same finding was reported in Table 5, where the performance evaluation in terms of MSE was made.

Table 3

*Test Problem 1 - The best approximate solution obtained by various algorithms*

Fourier Coefficient	Algorithm					
	WCA	PSO	OCBO	MFO	FPA	GA
$a_0$	1.9996E-02	3.74E-02	1.9998E-02	5.3744E-01	-1.8435E-01	1.0338E-01
$a_1$	1.1002E-02	4.67E-03	1.1000E-02	1.2014E-01	-3.6568E-01	2.8827E-01
$b_1$	3.8363E-02	5.31E-03	3.8360E-02	-1	3.5412E-01	-1.7517E-01
$a_2$	-7.9942E-04	-3.87E-02	-7.9942E-04	-1	1.0558E-01	-3.1155E-01
$b_2$	1.1888E-01	1.36E-01	1.1888E-01	-2.2236E-02	6.6087E-01	-2.5309E-01
$a_3$	-3.6247E-02	-1.49E-02	-3.6247E-02	-1.2050E-01	4.2466E-01	-2.9862E-01
$b_3$	4.7980E-02	8.24E-02	4.7975E-02	8.7854E-01	1.1973E-01	3.8416E-01
$a_4$	-4.2281E-02	-1.11E-02	-4.2281E-02	4.9884E-01	7.9053E-02	2.2051E-01
$b_4$	6.5364E-02	4.14E-02	6.5365E-02	8.3412E-02	-1.9287E-01	1.5877E-01
$a_5$	2.7610E-02	1.21E-02	2.7610E-02	1.5567E-02	-6.0481E-02	2.6291E-02
$b_5$	4.9891E-02	2.81E-02	4.9890E-02	-1.9518E-01	-3.0025E-02	-8.6205E-02
$a_6$	2.0754E-02	1.06E-02	2.0755E-02	-4.1044E-02	-4.5552E-03	-1.7893E-02
$b_6$	-4.1469E-03	1.01E-03	-4.1470E-03	4.6750E-03	5.7165E-03	7.5348E-03

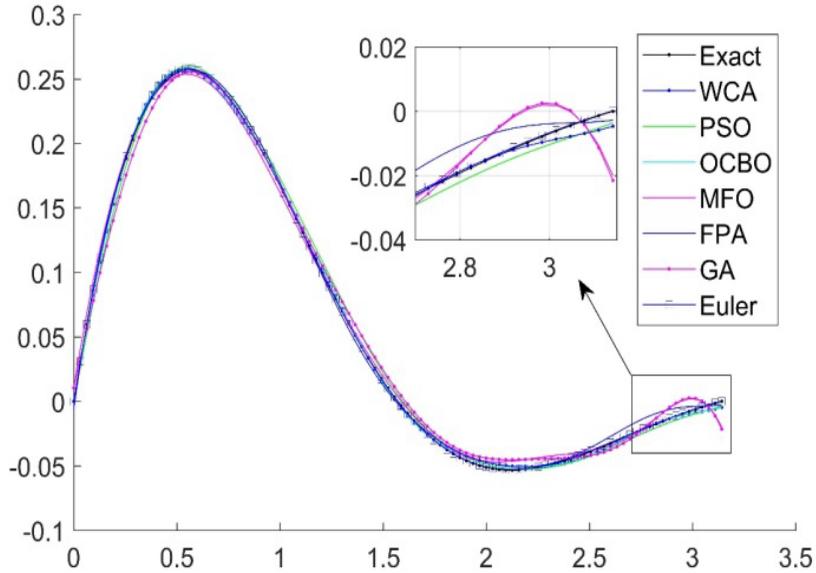


Figure 4. Test Problem 1 – Comparison of the best approximate solutions with the exact solution for WCA, PSO, OCBO, MFO, FPA, GA and Euler

Table 4

The obtained best GD metric of all algorithms for all test problems

Test Problem	Best GD							Best	Worst
	WCA	PSO	MFO	FPA	OCBO	GA	Euler		
1	1.63E-04	3.29E-04	3.66E-04	2.58E-04	1.63E-04	4.76E-04	1.11E-04	Euler	GA
2	5.17E-03	5.15E-03	3.77E-04	1.06E-03	5.15E-04	1.41E-03	8.84E-04	MFO	WCA
3	1.53E-05	2.16E-05	1.47E-05	1.08E-05	6.74E-06	2.40E-04	4.58E-03	OCBO	Euler
4	1.09E-03	-	5.15E-04	8.18E-04	-	2.70E-04	4.45E-03	GA	Euler
5	-	1.77E-03	5.48E-04	1.10E-03	-	5.70E-04	1.37E-02	MFO	Euler
6	-	1.16E-04	4.74E-04	4.77E-04	-	3.02E-04	5.27E-04	PSO	Euler
7	-	3.95E-04	2.81E-04	4.27E-04	-	2.66E-04	4.40E-05	Euler	FPA

Note: There were some blank spaces due to some best approximate solutions were not provided in the respective references

Table 5

The obtained best MSE of all algorithms for all test problems

Test Problem	Best MSE							Best	Worst
	WCA	PSO	MFO	FPA	OCBO	GA	Euler		
1	5.32E-06	2.17E-05	2.67E-05	1.33E-05	5.32E-06	4.53E-05	2.44E-06	Euler	GA
2	2.04E-04	6.94E-03	1.22E-04	4.12E-03	2.00E-04	1.47E-04	4.49E-05	Euler	PSO
3	4.68E-08	9.35E-08	4.32E-08	2.34E-08	9.08E-09	1.15E-05	5.82E-03	OCBO	Euler
4	2.36E-04	-	5.30E-05	1.34E-04	-	1.46E-05	3.97E-03	GA	Euler
5	-	2.22E-03	6.39E-05	1.12E-03	-	7.40E-05	5.37E-02	MFO	Euler
6	-	2.67E-06	6.10E-05	1.21E-04	-	1.82E-05	5.82E-05	PSO	FPA
7	-	5.22E-06	1.58E-05	3.64E-05	-	1.42E-05	3.88E-07	Euler	FPA

Note: There were some blank spaces due to some best approximate solutions were not provided in the respective references

**Test Problem 2: Simple Electrical Circuit.** In this problem, a system of ODEs for a simple circuit was considered (Sadollah et al., 2015c). The ODEs are expressed as Equation 26:

$$\begin{aligned}\frac{dI}{dt} &= -I - V \\ \frac{dV}{dt} &= 2I - V\end{aligned}\tag{26}$$

with initial conditions  $I(0) = 2$  and  $V(0) = 2$

Here,  $I$  and  $V$  represent the current and voltage, respectively. The solution interval of time,  $t$  varies from 0 to 1.5. The exact solutions of current and voltage are given by Equation 27:

$$\begin{aligned}I(t) &= 2e^{-t} \cos(\sqrt{2}t) - \sqrt{2}e^{-t} \sin(\sqrt{2}t) \\ V(t) &= 2\sqrt{2}e^{-t} \sin(\sqrt{2}t) + 2e^{-t} \cos(\sqrt{2}t)\end{aligned}\tag{27}$$

The test problem is unique in which two ODEs are solved simultaneously.

The NT of Fourier series was assigned to 3. Since the test problem is a system of two ODEs, hence, a total of 14 variables needs to be optimized, given as Equation 28 and 29:

$$\begin{aligned}
 I_{app}(t) = & a_0 + a_1 \cos\left(\frac{\pi t}{1.5}\right) + b_1 \sin\left(\frac{\pi t}{1.5}\right) + a_2 \cos\left(\frac{2\pi t}{1.5}\right) + b_2 \sin\left(\frac{2\pi t}{1.5}\right) \\
 & + a_3 \cos\left(\frac{3\pi t}{1.5}\right) + b_3 \sin\left(\frac{3\pi t}{1.5}\right)
 \end{aligned} \tag{28}$$

$$\begin{aligned}
 V_{app}(t) = & c_0 + c_1 \cos\left(\frac{\pi t}{1.5}\right) + d_1 \sin\left(\frac{\pi t}{1.5}\right) + c_2 \cos\left(\frac{2\pi t}{1.5}\right) + d_2 \sin\left(\frac{2\pi t}{1.5}\right) \\
 & + c_3 \cos\left(\frac{3\pi t}{1.5}\right) + d_3 \sin\left(\frac{3\pi t}{1.5}\right)
 \end{aligned} \tag{29}$$

Table 6 and 7 summarize the best approximate solutions for the current function,  $I_{app}$  and the voltage function,  $V_{app}$  obtained by all algorithms. Figure 5 presents the comparison of the obtained best approximate solutions from each algorithm to the actual solution in Equation 27. It can be observed that MFO, GA and Euler’s method can approximate both solutions almost perfectly. The approximate solutions given by the WCA, PSO, OCBO and FPA were only fitted well for one solution ( $I_{app}$  for WCA, PSO and OCBO while  $V_{app}$  for FPA) while having another solution overshoot (such as for WCA, PSO and OCBO while  $I_{app}$  for FPA). Since this test problem needs to solve a system of ODEs and thus, the algorithm which can optimize two functions simultaneously is desirable. In this regard, the MFO outperformed the others, which was further corroborated by the lowest GD of 3.7683E-04 in Table 4. The WCA gave the worst approximate solution, with the highest GD of 5.1703E-03. The unsatisfactory performance of WCA was due to its inability to approximate the  $V_{app}$  precisely, as observed in Figure 5. Though this test problem has 2 curves, the GD values of both curves will be added up.

In terms of MSE, Table 5 reports different finding where the Euler’s method gave the highest accuracy while the PSO was ranked last. This is attributed that for the GD, the Euclidean distance between point  $i$  in the approximate solution with the nearest point in the exact solution was considered, as illustrated in Figure 6. As such, the performance evaluation is different from the MSE in which the same point  $i$  in both the approximate and exact solutions is compared. Referring to Figure 5, it can be seen that comparison in terms of GD metric was more suited to visual assessment where the approximate solution given by the WCA was more significantly deviated from the actual solution.

Table 6

Test Problem 2 - The best approximate solution obtained by various algorithms for  $I_{app}$

Fourier Coefficient	Algorithm					
	WCA	PSO	OCBO	MFO	FPA	GA
$a_0$	0.8381	0.8385	0.8385	9.6502E-01	9.2523E-01	8.3108E-01
$a_1$	1.3621	1.3626	1.3625	1.2481	1.4493	1.2764
$b_1$	-1.0557	-1.0563	-1.0562	-1.2309	-1.2998	-1.0187
$a_2$	-9.8533E-02	-9.8530E-02	-9.8535E-02	-1.8188E-01	-2.9654E-01	-6.8844E-02
$b_2$	-0.4597	-0.4600	-0.4600	-3.1873E-01	-4.7508E-01	-3.4941E-01
$a_3$	-0.1021	-0.1025	-0.1025	-3.6749E-02	-8.8318E-02	-4.8406E-02
$b_3$	1.1915E-02	1.1919E-02	1.1920E-02	2.7173E-02	9.3774E-02	2.9935E-03

Table 7

Test Problem 2 - The best approximate solution obtained by various algorithms for  $V_{app}$

Fourier Coefficient	Algorithm					
	WCA	PSO	OCBO	MFO	FPA	GA
$c_0$	0.6618	0.6622	0.6620	1.0349	1.2179	5.3369E-01
$c_1$	0.8179	0.8184	0.8185	8.0697E-01	7.6768E-01	8.6274E-01
$d_1$	1.3265	1.326	1.3270	7.1052E-01	4.2777E-01	1.5600
$c_2$	0.4821	0.4818	0.4820	1.2034E-01	-3.5610E-02	5.8976E-01
$d_2$	8.5909E-02	-8.5912E-02	8.5915E-02	8.0109E-02	9.6923E-02	3.3360E-02
$c_3$	3.7436E-02	3.7431E-02	3.7430E-02	3.1987E-02	5.1374E-02	6.6653E-03
$d_3$	-0.1137	-0.1136	-0.1135	-8.2506E-04	5.4181E-02	-1.3986E-01

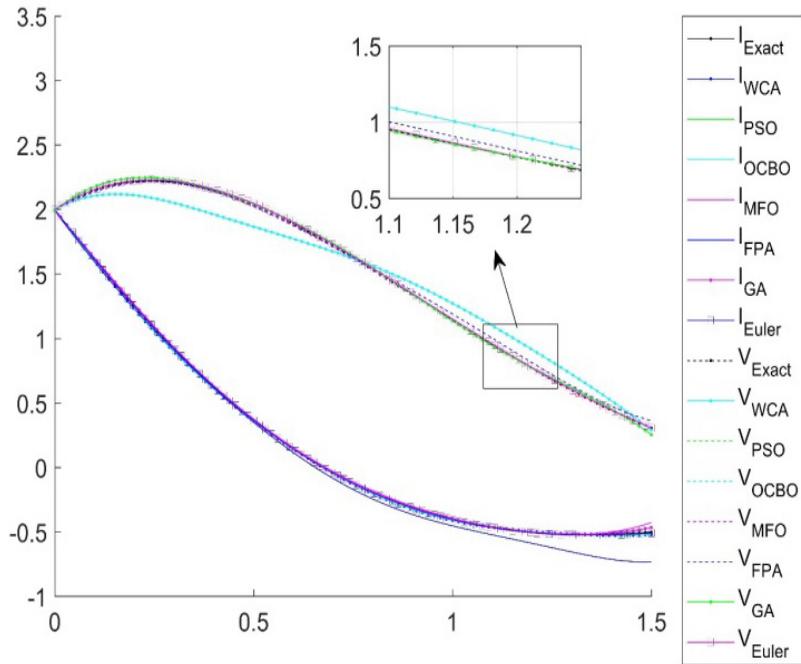


Figure 5. Test Problem 2 – Comparison of the best approximate solutions with the exact solution for WCA, PSO, OCBO, MFO, FPA, GA and Euler

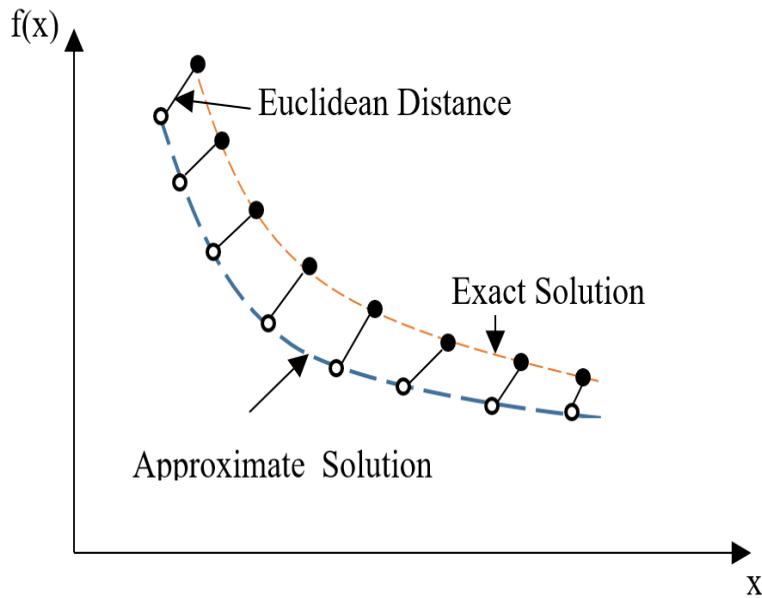


Figure 6. The schematic diagram of GD metric in evaluating the Euclidean distance between the approximate solution and the exact solution

**Test Problem 3: Bernoulli Equation.** The famous Bernoulli equation is considered in this test problem, which is formulated as Equation 30 (Sadollah et al., 2015c):

$$y'' + (y')^2 - 2e^{-y} = 0 \tag{30}$$

with boundary conditions of  $y(0) = 0$  and  $y(1) = 0$ . The exact solution of this second order nonlinear ODE is given by Equation 31:

$$y(x) = \ln \left( \left( x - \frac{1}{2} \right)^2 + \frac{3}{4} \right) \tag{31}$$

with  $x$  lies between 0 and 1.

The NT of the Fourier series was assigned as 3 and thus, the approximate function that needs to be optimized by all algorithms is as in Equation 32. The obtained best approximate solutions by all algorithms are shown in Table 8

$$Y_{app}(x) = a_0 + a_1 \cos(\pi x) + b_1 \sin(\pi x) + a_2 \cos(2\pi x) + b_2 \sin(2\pi x) + a_3 \cos(3\pi x) + b_3 \sin(3\pi x) \tag{32}$$

Table 8

*Test Problem 3 - The best approximate solution obtained by various algorithms*

Fourier Coefficient	Algorithm					
	WCA	PSO	OCBO	MFO	FPA	GA
$a_0$	2.25E-02	2.40E-02	2.2512E-02	2.1661E-02	2.1926E-02	2.8790E-02
$a_1$	8.31E-05	-1.48E-05	8.3125E-05	-2.6965E-04	3.1164E-05	1.9494E-03
$b_1$	-0.3288	-3.32E-01	-0.3290	-3.2742E-01	-3.2808E-01	-3.4499E-01
$a_2$	-2.25E-02	-2.40E-02	-2.2490E-02	-2.1572E-02	-2.2097E-02	-3.1143E-02
$b_2$	1.67E-04	4.09E-05	-1.6735E-04	6.5889E-05	8.9764E-05	-2.8985E-04
$a_3$	-8.31E-05	1.45E-05	-8.3096E-05	1.5966E-05	5.8905E-05	-1.5094E-05
$b_3$	3.69E-03	4.08E-03	3.6882E-03	3.3740E-03	3.5244E-03	7.6365E-03

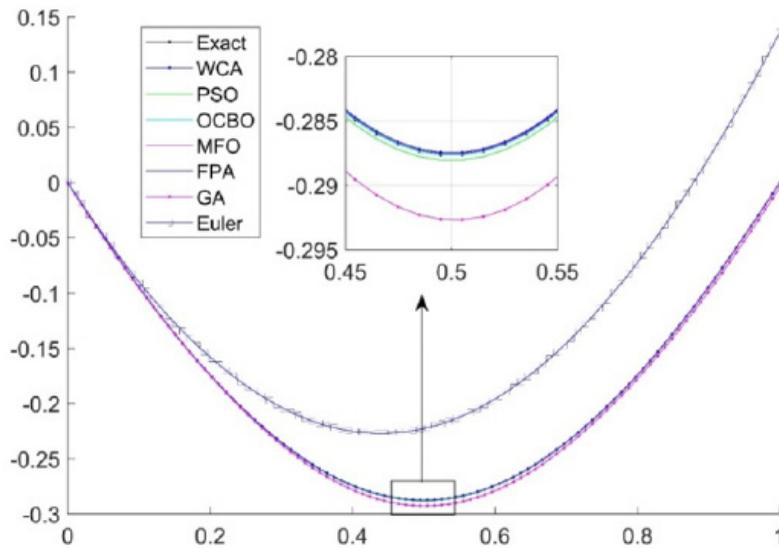


Figure 7. Test Problem 3 – Comparison of the best approximate solutions with the exact solution for WCA, PSO, OCBO, MFO, FPA, GA and Euler

The comparison of the approximate solution to the exact solution (Equation 31) for each algorithm is presented in Figure 7. All algorithms were able to approximate the solution satisfactorily, where the approximate solutions fitted the exact solution well except for the Euler’s method. The approximate solution of the Euler’s method deviated from the exact solution as the  $x$  increases. In terms of the GD metric (Table 4), the GD of the OCBO was the lowest, followed closely by the FPA, MFO, WCA, PSO and GA with differences of  $4.0885E-6$ ,  $7.9635E-06$ ,  $8.5635E-06$ ,  $1.4885E-05$  and  $2.3317E-04$ . Euler’s method gave the highest GD of  $4.5792E-03$ . The same findings were reported in terms of MSE, where the Euler’s gave the lowest approximation accuracy while OCBO was ranked first.

**Test Problem 4: Nonlinear ODE with Neumann Boundary Condition.** A more complex ODE is examined in this test problem, in which the Neumann boundary condition is considered. Moreover, the difficulty of solving the ODE increases as this test problem has a wide range of interval solution, *i.e.*  $x$  varies from 0 to 15. The ODE is expressed as Equation 33 (Sadollah et al., 2015 c)

$$y'' + 0.3y' + y - 1 = 0 \tag{33}$$

with boundary conditions of  $y(0) = 0$  and  $y'(0) = 0$

The exact solution of this test problem is given by Equation 34:

$$y(x) = \left( \frac{-3\sqrt{391} \sin\left(\frac{\sqrt{391}x}{20}\right)}{391 e^{\left(\frac{3x}{20}\right)}} \right) - \left( \frac{\cos\left(\frac{\sqrt{391}x}{20}\right)}{e^{\left(\frac{3x}{20}\right)}} \right) + 1 \tag{34}$$

The obtained Fourier series with NT=6 is expressed as Equation 35:

$$\begin{aligned} Y_{approx}(x) = & a_0 + a_1 \cos\left(\frac{\pi x}{15}\right) + b_1 \sin\left(\frac{\pi x}{15}\right) + a_2 \cos\left(\frac{2\pi x}{15}\right) \\ & + b_2 \sin\left(\frac{2\pi x}{15}\right) + a_3 \cos\left(\frac{3\pi x}{15}\right) + b_3 \sin\left(\frac{3\pi x}{15}\right) \\ & + a_4 \cos\left(\frac{4\pi x}{15}\right) + b_4 \sin\left(\frac{4\pi x}{15}\right) + a_5 \cos\left(\frac{5\pi x}{15}\right) \\ & + b_5 \sin\left(\frac{5\pi x}{15}\right) + a_6 \cos\left(\frac{6\pi x}{15}\right) + b_6 \sin\left(\frac{6\pi x}{15}\right) \end{aligned} \tag{35}$$

Table 9 shows the best approximate solutions given by the WCA, MFO and FPA. The results of PSO and OCBO were not presented due to this problem was not simulated in Babaei (2013) and Panda and Pani (2017). The best approximate solutions were compared against the exact solution in Figure 8. As observed in Figure 8, all the obtained approximate solutions exhibited a slight deviation from the exact solution, but the given approximate solutions were able to capture the behaviour of the exact solution. However, the approximate solution given by the Euler’s method was unsatisfactory at the regions with local minima and local maxima. The comparison in terms of the GD metric in Table 4 shows that the lowest GD was obtained by the GA, and followed by MFO, FPA and WCA, while the highest GD was obtained by the Euler. In terms of MSE, it can be seen in Table 5 that Euler’s method also produced the lowest approximation accuracy.

Table 9  
Test Problem 4 - The best approximate solution obtained by various algorithms

Fourier Coefficient	Algorithm			
	WCA	MFO	FPA	GA
$a_0$	0.8861	9.5814E-01	9.6552E-01	7.4148E-01
$a_1$	-0.1652	-6.9514E-01	3.3187E-01	-9.9751E-02
$b_1$	0.2533	2.1481E-01	6.1323E-02	5.4197E-01
$a_2$	0.3278	5.1538E-01	5.6435E-02	6.0468E-01

Table 9 (Continued)

Fourier Coefficient	Algorithm			
	WCA	MFO	FPA	GA
$b_2$	0.2068	1	-5.3821E-01	1.0262E-01
$a_3$	0.1606	8.4533E-01	-5.4024E-01	2.5716E-02
$b_3$	-0.4637	-8.0719E-01	-1.5700E-01	-6.9240E-01
$a_4$	-0.6637	-1	-4.0653E-01	-8.1705E-01
$b_4$	-0.1965	-6.1628E-01	2.4779E-01	-1.0015E-01
$a_5$	-0.5218	-6.7200E-01	-3.0362E-01	-4.5059E-01
$b_5$	0.1909	3.8418E-01	5.0291E-02	2.6097E-01
$a_6$	-0.0238	4.0047E-02	-6.2747E-02	-2.1383E-04
$b_6$	0.0926	1.2095E-01	3.6468E-02	7.1086E-02

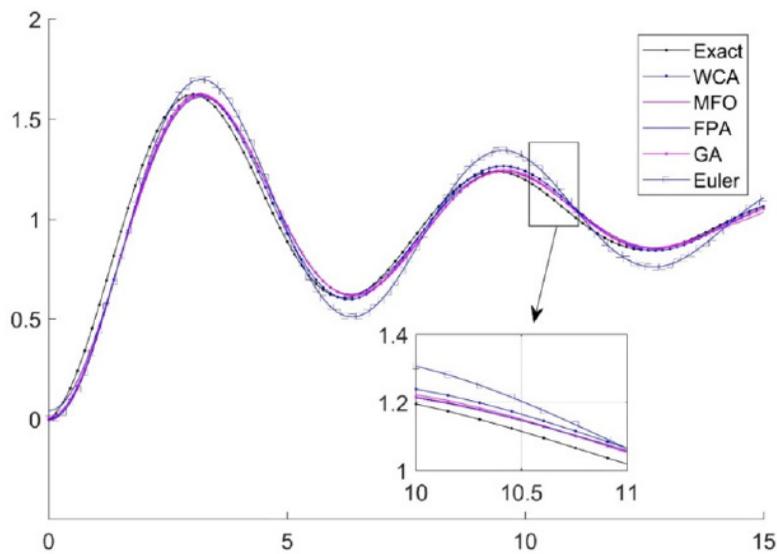


Figure 8. Test Problem 4 – Comparison of the obtained best approximate solutions with the exact solution for WCA, MFO, FPA, GA and Euler

**Test Problem 5: Mass-Spring System without Damping Force.** A mechanical mass-spring system without damping force as illustrated in Figure 9(a) is considered in this test problem. The spring is stretched with a length of 27cm by a 7.25kg object. At the initial position, the spring is displaced 182.88cm upwards from equilibrium, together with an initial downward velocity of 30.48 cm/s. No external and damping forces are involved. By solving this mass-spring system, the ODE as in Equation 36 (Sadollah et al., 2015c):

$$\frac{1}{2}u'' + 18u = 0 \tag{36}$$

with initial conditions of  $u(0) = -1/2$  and  $u'(0) = 1$  is obtained. The analytical solution of this mechanical problem is given by Equation 37:

$$u(t) = 0.52705 \cos(6t - 2.81984) \tag{37}$$

with solution interval of  $t$  varies from 0 to 3.

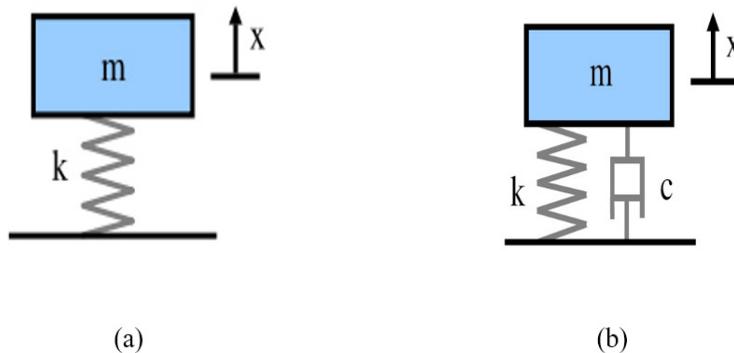


Figure 9 Mass-Spring System (a) without damping force, (b) with damping force

The NT of the Fourier series was chosen as 6. The formulated Fourier series to be optimized by the PSO, MFO and FPA algorithms is as in Equation 38:

$$\begin{aligned}
 Y_{app}(x) = & a_0 + a_1 \cos\left(\frac{\pi x}{3}\right) + b_1 \sin\left(\frac{\pi x}{3}\right) + a_2 \cos\left(\frac{2\pi x}{3}\right) + b_2 \sin\left(\frac{2\pi x}{3}\right) \\
 & + a_3 \cos\left(\frac{3\pi x}{3}\right) + b_3 \sin\left(\frac{3\pi x}{3}\right) + a_4 \cos\left(\frac{4\pi x}{3}\right) + b_4 \sin\left(\frac{4\pi x}{3}\right) \\
 & + a_5 \cos\left(\frac{5\pi x}{3}\right) + b_5 \sin\left(\frac{5\pi x}{3}\right) + a_6 \cos\left(\frac{6\pi x}{3}\right) + b_6 \sin\left(\frac{6\pi x}{3}\right)
 \end{aligned} \tag{38}$$

Table 10 summarizes the best approximate solutions obtained by all algorithms. The comparison between the best approximate solutions and the analytical solution in Equation 36 (Figure 10) shows that all algorithms can capture the changing trend of the exact solution. The best fit of the MFO with the actual solution can be observed, with the lowest GD value as presented in Table 4. The approximate solutions obtained by the GA, PSO

and FPA demonstrated a slight deviation from the exact solution. The worst approximate solution was given by Euler’s method, with the highest GD value of 1.3702E-02. The approximate solution of Euler’s method deviated gradually from the exact solution on each local minimum and maximum. The lowest approximation accuracy in terms of MSE was also reported by Euler’s method, as shown in Table 5.

Table 10

Test Problem 5 - The best approximate solution obtained by various algorithms

Fourier Coefficient	Algorithm			
	PSO	MFO	FPA	GA
$a_0$	0.1867	-4.1899E-01	4.9163E-01	-2.2289E-01
$a_1$	0.1356	2.9464E-01	-6.0758E-02	-1.6153E-01
$b_1$	-0.2727	8.9775E-01	-8.3639E-01	4.3788E-01
$a_2$	-3.3241E-02	1	-4.6687E-01	4.1815E-01
$b_2$	-0.2059	-4.5991E-01	1.0118E-01	2.6953E-01
$a_3$	-0.1890	-4.4598E-01	1.0218E-01	2.9498E-01
$b_3$	-0.2059	-1	5.1262E-02	-3.9466E-01
$a_4$	-0.3420	-8.3637E-01	-2.4337E-01	-3.7587E-01
$b_4$	0.1170	3.0496E-01	-9.4811E-02	-2.4499E-01
$a_5$	-3.4034E-02	1.3113E-01	-9.1927E-02	-1.6475E-01
$b_5$	0.4219	6.4957E-01	4.1332E-01	4.1686E-01
$a_6$	-0.2922	-2.3638E-01	-2.7505E-01	-2.9000E-01
$b_6$	-5.3461E-02	-8.1045E-02	-4.9702E-02	8.5314E-03

**Test Problem 6: Mass-Spring System with Damping Force.** For test problem 5, when the velocity is 60.96 cm/s, a damper which imposes a force of 2.26 kg is added to the mass-spring system (Figure 9 (b)). The second-order ODE derived from this system is given by Equation 39 (Sadollah et al., 2015 c)

$$\frac{1}{2}u'' + \frac{5}{2}u' + 18u = 0 \tag{39}$$

with initial conditions of  $u(0) = -1/2$  and  $u'(0) = 1$ . The analytical solution of this mass-spring system is given by Equation 40:

$$u(t) = 0.502096e^{\frac{5}{2}t} \cos\left(\frac{\sqrt{119}}{2}t - 3.2321\right) \tag{40}$$

with the interval of  $t$  is between 0 and 3.

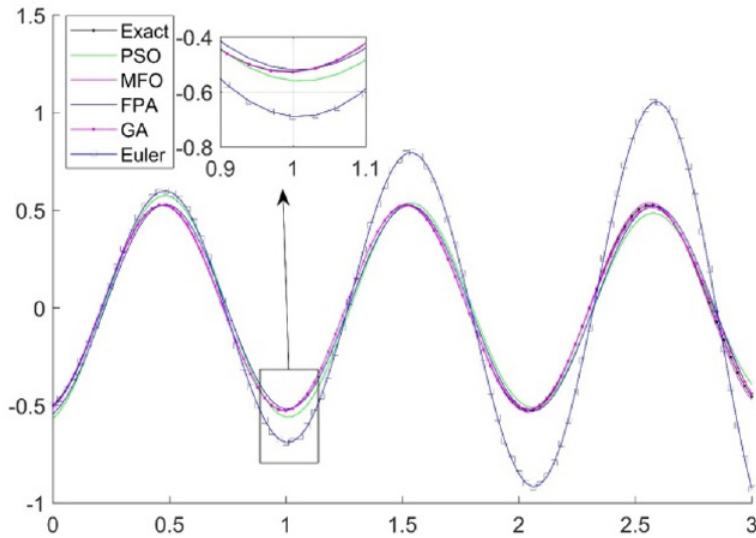


Figure 10. Test Problem 5 – Comparison of the best approximate solutions with the exact solution for PSO, MFO, FPA, GA and Euler

Following the work in (Sadollah et al., 2015c), the NT of 8 was chosen. The Fourier coefficients in Equation 41 were then optimized using the PSO, MFO and FPA algorithms. Table 11 presents the best approximate solutions obtained by each algorithm.

$$\begin{aligned}
 Y_{app}(x) = & a_0 + a_1 \cos\left(\frac{\pi x}{3}\right) + b_1 \sin\left(\frac{\pi x}{3}\right) + a_2 \cos\left(\frac{2\pi x}{3}\right) + b_2 \sin\left(\frac{2\pi x}{3}\right) \\
 & + a_3 \cos\left(\frac{3\pi x}{3}\right) + b_3 \sin\left(\frac{3\pi x}{3}\right) + a_4 \cos\left(\frac{4\pi x}{3}\right) + b_4 \sin\left(\frac{4\pi x}{3}\right) \\
 & + a_5 \cos\left(\frac{5\pi x}{3}\right) + b_5 \sin\left(\frac{5\pi x}{3}\right) + a_6 \cos\left(\frac{6\pi x}{3}\right) + b_6 \sin\left(\frac{6\pi x}{3}\right) \\
 & + a_7 \cos\left(\frac{7\pi x}{3}\right) + b_7 \sin\left(\frac{7\pi x}{3}\right) + a_8 \cos\left(\frac{8\pi x}{3}\right) + b_8 \sin\left(\frac{8\pi x}{3}\right)
 \end{aligned} \tag{41}$$

Table 11

*Test Problem 6 - The best approximate solution obtained by various algorithms*

Fourier Coefficient	Algorithm			
	PSO	MFO	FPA	GA
$a_0$	4.2427E-02	1.6649E-01	2.0069E-02	-2.4357E-01
$a_1$	0.2692	6.3516E-01	-2.9983E-02	-2.2733E-01
$b_1$	-8.1634E-02	-1.8759E-01	-2.0882E-01	3.7190E-01
$a_2$	-3.8633E-02	1.4685E-01	-5.3383E-01	1.4775E-01
$b_2$	-0.4001	-9.2785E-01	-6.1953E-03	2.3527E-01
$a_3$	-0.2950	-7.1377E-01	-2.3839E-02	7.8729E-02
$b_3$	-7.6912E-02	-4.4013E-01	6.8551E-01	-2.1613E-02
$a_4$	-0.1595	-4.6111E-01	6.0477E-01	-8.5710E-04
$b_4$	-3.2617E-02	1.1574E-01	-9.8299E-02	-1.8529E-03
$a_5$	-0.3102	-3.9711E-01	-2.3365E-01	-2.5997E-02
$b_5$	3.6654E-02	1.4303E-01	-5.0980E-01	-9.4478E-02
$a_6$	-0.1159	-1.5763E-01	-3.8550E-01	-1.6933E-01
$b_6$	0.2508	4.1643E-01	1.4217E-01	-3.8009E-02
$a_7$	8.5309E-02	1.9945E-01	1.9615E-02	-7.2358E-02
$b_7$	8.8175E-02	1.5156E-01	1.6755E-01	9.1919E-02
$a_8$	2.2355E-02	4.7600E-02	2.9148E-02	1.7476E-02
$b_8$	-1.3398E-02	-5.1783E-02	6.6004E-03	2.9737E-02

Figure 11 presents the comparison of the best approximate solutions found by all algorithms to the exact solution. It can be observed that the approximate solution obtained by the PSO fitted the exact solution well for all data points. On the other hand, a noticeable deviation from the exact solution after  $x = 2$  can be observed for the approximate solution obtained by the MFO, while the approximate solutions of the FPA and GA deviated from the exact solution before  $x = 0.5$  and at  $x = 1$ . All the algorithms, in general, were able to approximate the exact solution satisfactorily, except significant deviation from the exact solution at  $x = 0.5$  and  $x = 1$  was observed for the Euler's method. Considering the values of GD metric in Table 4 the PSO gave better approximate solution than others, corroborated by the lowest GD metric in addition to the visual inspection. Euler's method, on the other hand, gave the worst approximate solution.

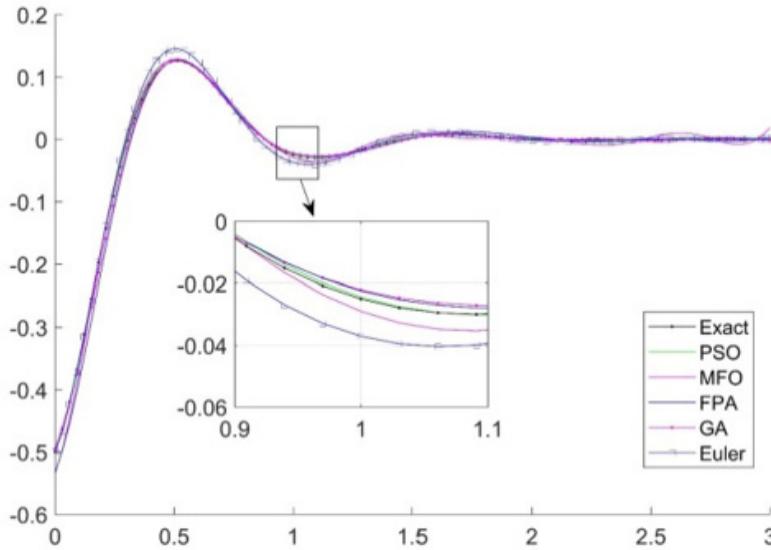


Figure 11. Test Problem 6 – Comparison of the best approximate solutions with the exact solution for PSO, MFO, FPA, GA and Euler

**Test Problem 7: Longitudinal Heat Transfer Fins.** Longitudinal fins are widely utilized in heat exchanger application to increase the rate of heat dissipation to or from the environment. Consider a longitudinal fin with constant rectangular profile as shown in Figure 12, the fin with length  $L$  and semi-base thickness  $t_b$  is attached to a surface with a constant temperature  $T_b$ . The convective heat transfer coefficient,  $h$ , fin thermal conductivity,  $k$  and fin surface emissivity,  $\varepsilon$  are assumed as Equation 42, 43 and 44:

$$h = h_b \left[ \frac{T - T_a}{T_b - T_a} \right]^m \tag{42}$$

$$k = k_a [1 + \alpha(T - T_a)] \tag{43}$$

$$\varepsilon = \varepsilon_s [1 + \beta(T - T_s)] \tag{44}$$

where  $h_b$  is convection heat transfer coefficient according to temperature difference  $T_a$  and  $T_b$ ,  $k_a$  is thermal conductivity at convection sink temperature  $T_a$ ,  $\varepsilon_s$  is fin surface emissivity at radiation sink temp  $T_s$  and constant  $\alpha$  and  $\beta$  are variation of thermal conductivity and surface emissivity with temperature, respectively.

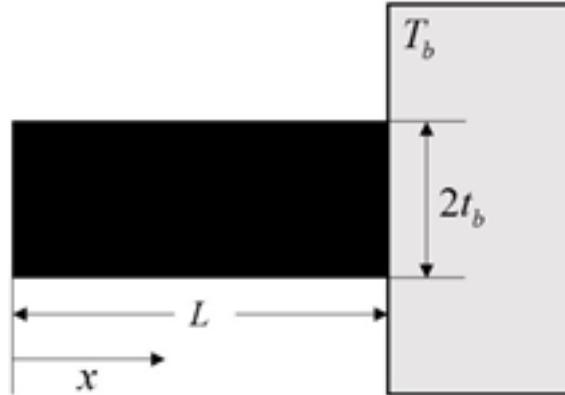


Figure 12. A longitudinal fin with constant rectangular profile

The energy balance of longitudinal fin per unit width based on one-dimensional heat conduction is as in Equation 45 (Torabi et al., 2013)

$$k_a \frac{d}{dx} \left[ [1 + \alpha(T - T_a)] t(x) \frac{dT}{dx} \right] - h_b \left[ \frac{T - T_a}{T_b - T_a} \right]^m (T - T_a) - \sigma \varepsilon_s [1 + \beta(T - T_s)] (T^4 - T_s^4) = 0 \tag{45}$$

where  $t(x)$  is the local semi-fin thicknesses (Equation 46):

$$t(x) = t_b + \delta \left[ (x/L)^n - 1 \right] \tag{46}$$

where exponent  $n = 0, 1$  and  $2$  for rectangular, trapezoidal and concave parabolic fins, respectively and  $\delta$  is the fin taper. The boundary conditions of adiabatic tip and constant base temperature are (Equation 47 & 48):

$$\left. \frac{dT}{dx} \right|_{x=0} = 0 \tag{47}$$

$$T(L) = T_b \tag{48}$$

By assuming these dimensionless parameters (Equation 49, 50, 51 & 52):

$$\theta = \frac{T}{T_b}, \theta_a = \frac{T_a}{T_b}, \theta_s = \frac{T_s}{T_b}, X = \frac{x}{L}, C = \frac{\delta}{t_b}, Nc = \frac{h_b L^2 T_b^m}{k_a t_b (T_b - T_a)^m}, \tag{49}$$

$$A = \alpha T_b, B = \beta T_b, Nr = \frac{\sigma \varepsilon_s L^2 T_b^3}{k_a t_b}, \psi = \frac{t_b}{L}$$

and hence, the fin profile forms the following ODE:

$$\begin{aligned} \frac{d}{dx} \left[ [1 + A(\theta - \theta_a)] [1 + C(X^n - 1)] \frac{d\theta}{dX} \right] - Nc(\theta - \theta_a)^{m+1} \\ - Nr [1 + B(\theta - \theta_s)] (\theta^4 - \theta_s^4) = 0 \end{aligned} \quad (50)$$

with boundary conditions of:

$$\theta(1) = 1 \quad (51)$$

$$\left. \frac{d\theta}{dX} \right|_{X=0} = 0 \quad (52)$$

where parameter  $A$  represents thermal conductivity,  $B$  represents emissivity,  $m$  is associated with convective heat transfer coefficient,  $Nc$  represents convection-conduction parameter,  $Nr$  represents radiation-conduction parameter,  $\theta_a$  represents temperature of convection,  $\theta_s$  represents temperature of radiation sinks and  $C$  represents fin taper ratio. In a longitudinal fin with constant cross-sectional area,  $A_c$ , length,  $L$  and perimeter of cross-section,  $P$ , the following dimensionless parameters in Equation 53 are introduced into Equation 45:

$$\begin{aligned} \theta = \frac{T}{T_b}, \theta_a = \frac{T_a}{T_b}, \theta_s = \frac{T_s}{T_b}, X = \frac{x}{L}, Nc = \frac{h_b L^2 P}{k_a A_c}, \\ A = \alpha T_b, B = \beta T_b, Nr = \frac{\sigma \varepsilon_s L^2 T_b^3 P}{k_a A_c} \end{aligned} \quad (53)$$

The obtained ODE is as in Equation 54:

$$\begin{aligned} \frac{d}{dX} \left[ [1 + A(\theta - \theta_a)] \frac{d\theta}{dx} \right] - Nc(\theta - \theta_a) \left[ \frac{\theta - \theta_a}{1 - \theta_a} \right]^m \\ - Nr [1 + B(\theta - \theta_s)] (\theta^4 - \theta_s^4) = 0 \end{aligned} \quad (54)$$

with boundary conditions in Equation 55 and 56:

$$\theta(0) = 1 \quad (55)$$

$$\left. \frac{d\theta}{dX} \right|_{X=1} = 0 \quad (56)$$

Assuming that the heat conduction occurs only in the longitudinal direction ( $x$ -direction) for a pure convection fin (*i.e.* radiation conduction,  $Nr = 0$ ) with constant thermal conductivity ( $A = 0$ ) operating in a constant convective heat transfer coefficient,  $m$  is assigned to 0,

$Nc$  is assumed to be 1,  $C$  is equal to zero for fin with constant cross-sectional area and  $\theta_a$  is 0.5, the governing ODE can be expressed as Equation 57 (Sadollah et al, 2015a):

$$\frac{d^2\theta}{dx^2} - \theta + 0.5 = 0 \tag{57}$$

with the length of fin  $L$  is assigned to be 1. The analytical solution of this ODE is given by Equation 58:

$$\theta(x) = 0.162e^x + 0.162e^{-x} + 0.5 \tag{58}$$

with  $x$  is within interval 0 to 1.

The NT was chosen as 3 in this test problem. Table 12 summarizes the obtained best approximate solutions for each algorithm, where it can be observed that the obtained coefficients for the Fourier series are similar for all algorithms, except for GA. As shown in Table 4, Euler’s method has surpassed GA, MFO, PSO and FPA with the lowest GD value, albeit in Figure 13 the obtained approximated solutions by all algorithms are close to the actual solution. The performance of all algorithms were considered satisfactorily with the GDs within the resolution of 1E-04. As shown in Table 5, the same performance can be observed in which the Euler’s method gave the highest accuracy while the FPA was ranked last.

Table 12  
*Test Problem 7 - The best approximate solution obtained by various algorithms*

Fourier Coefficient	Algorithm			
	PSO	MFO	FPA	GA
$a_0$	0.917338	9.1900E-01	9.1917E-01	9.0374E-01
$a_1$	-0.093953	-8.6519E-02	-8.3184E-02	-8.5112E-02
$b_1$	-0.054890	-5.9048E-02	-5.7407E-02	-3.8041E-02
$a_2$	-0.005338	-7.5052E-03	-6.6807E-03	3.9307E-03
$b_2$	0.026293	2.5749E-02	2.4814E-02	2.2851E-02
$a_2$	0.005953	5.1650E-03	4.9853E-03	3.6390E-03
$b_3$	7.67E-04	7.7181E-04	5.9018E-04	-3.0567E-03

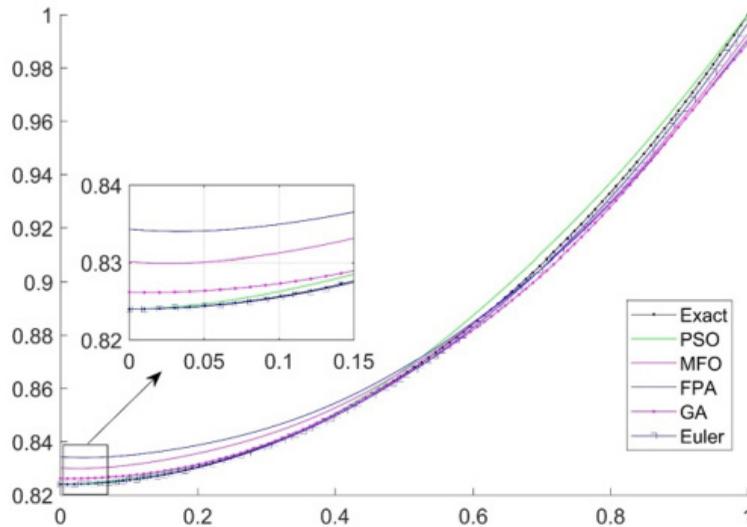


Figure 13. Test Problem 7 – Comparison of the best approximate solutions with the exact solution for PSO, MFO, FPA, GA and Euler

### Comparison of Optimization Performance and Discussion

From the obtained results in Table 4, each algorithm was ranked based on its optimization performance in terms of the GD metric for all considered test problems. The results are summarized in Table 13. The OCBO was ranked first among others, followed by the MFO, FPA and GA with close competition.

Also, since this study attempts to develop an approximate solution of ODE with the formulation of Fourier series expansion based on the MFO and FPA as the optimizers, the results of the average, worst and standard deviation (SD) of GD assessment metric for these optimizers are presented in Table 14. Interestingly, though MFO and GA outperformed the FPA in most of the test problems if considering only the best GD values (Table 4), their average GD values in all test problems were found to be higher than the FPA in Table 14. This implies that the FPA produces more consistent approximate solutions than the MFO and GA optimizers, indicated by the lower SD values in all test problems.

Table 13

*The ranking of all considered algorithms based on the obtained results in Table 4*

Test Problem	Algorithm						
	PSO	WCA	OCBO	MFO	FPA	GA	Euler
1	5	2	3	6	4	7	1
2	6	7	2	1	4	5	3
3	5	4	1	3	2	6	7
4	-	4	-	2	3	1	5
5	4	-	-	1	3	2	5
6	1	-	-	3	4	2	5
7	4	-	-	3	5	2	1
Average Rank	4.17	4.25	2	2.71	3.57	3.57	3.86
Total Rank	6 <sup>th</sup>	7 <sup>th</sup>	1 <sup>st</sup>	2 <sup>nd</sup>	3 <sup>rd</sup> /4 <sup>th</sup>	3 <sup>rd</sup> /4 <sup>th</sup>	5 <sup>th</sup>

Table 14

*Results of the average, worst and standard deviation (SD) of GDs for MFO, FPA and GA algorithms for all test problems*

Test Problem	GD value	Algorithm		
		MFO	FPA	GA
1	Best	3.6552E-04	2.5799E-04	4.7591E-04
	Average	3.8010E-03	5.6645E-04	2.3227E-03
	Worst	2.7739E-2	8.8413E-04	6.9962E-03
	SD	5.6027E-03	1.5857E-04	1.4704E-03
2	Best	3.7638E-04	1.0593E-03	8.3174E-04
	Average	1.5327E-02	4.2612E-03	2.3910E-03
	Worst	6.1616E-02	7.6770E-03	4.3628E-03
	SD	1.5193E-02	1.7392E-03	9.4657E-04
3	Best	1.4702E-05	1.0827E-05	2.3991E-04
	Average	1.5521E-03	1.5522E-04	5.6107E-04
	Worst	4.5131E-02	3.2881E-04	1.3685E-03
	SD	8.0924E-03	7.9581E-05	2.8355E-04
4	Best	5.1476E-04	8.1790E-04	2.7037E-04
	Average	6.3881E-03	1.8513E-03	2.4832E-03

Table 14 (Continued)

Test Problem	GD value	Algorithm		
		MFO	FPA	GA
4	Worst	3.1902E-02	3.0593E-03	5.1182E-03
	SD	7.1452E-03	5.3710E-04	1.1382E-03
5	Best	5.4773E-04	1.1004E-03	5.7002E-04
	Average	1.3354E-02	2.0702E-03	2.4800E-03
	Worst	6.8411E-02	5.3404E-03	4.4761E-03
6	SD	1.6160E-02	9.1795E-04	9.1920E-04
	Best	4.7393E-04	4.7710E-04	3.0154E-04
	Average	2.5800E-03	7.2895E-04	9.9495E-04
	Worst	1.4132E-02	1.0160E-03	2.7513E-03
7	SD	3.0067E-03	1.3648E-04	5.1470E-04
	Best	2.8087E-04	4.2669E-04	2.6636E-04
	Average	1.5787E-02	4.4573E-04	6.2556E-03
	Worst	8.8986E-02	4.6948E-04	2.0060E-02
	SD	2.7739E-02	1.1897E-05	5.5743E-03

Table 15 presents the average iterations used by MFO, FPA and GA to converge for all problems. The simulation of each algorithm was repeated for 30 runs. For each run, the required number of iteration to reach the stopping condition was recorded, and subsequently summed up and averaged. Hence, the average iterations shown in Table 15 are in decimal number instead of natural number. For test problem 1 and 2, in comparison with FPA, more iterations are needed by the MFO algorithm and GA in order to converge. For other test problems, MFO outperformed both FPA and GA in terms of convergence speed. The significant difference can be observed in test problem 6 in which the MFO can converge to the optimal solutions with 400 iterations lesser, as compared to FPA and GA.

Table 15

*Average iterations used to reach convergence by MFO, FPA and GA algorithms for all test problems*

Test Problem	Iteration	Algorithm		
		MFO	FPA	GA
1	Average Iteration	999.4	677.9	699.4
	Maximum Iteration	1000	700	700

Table 15 (Continued)

Test Problem	Iteration	Algorithm		
		MFO	FPA	GA
2	Average Iteration	499.6	488.8	499.5
	Maximum Iteration	500	500	500
3	Average Iteration	499.5	791.8	798.7
	Maximum Iteration	500	800	800
4	Average Iteration	299.6	489.9	499.1
	Maximum Iteration	300	500	500
5	Average Iteration	399.7	481.6	499.3
	Maximum Iteration	400	500	500
6	Average Iteration	599.6	971.6	999.5
	Maximum Iteration	600	1000	1000
7	Average Iteration	936.4	987.2	999.6
	Maximum Iteration	1000	1000	1000

### Relationship of Number of Terms to the Approximation Accuracy

In the previous section, the performances of the optimization algorithms were evaluated in terms of the GD metric and MSE. The NT used in the test problems was chosen according to the studies in Babaei (2013), Sadollah et al. (2015a), and Sadollah et al. (2015c). In this section, the relationship between using different NTs to the accuracy of the approximate solutions was explored. The NTs were varied from 1 to 9, and the obtained Fourier series was then optimized using the MFO, FPA and GA. The number of population and the maximum number of iteration of these algorithms were fixed as in Table 1. All simulations were repeated for 30 trials. For each run, the simulation continues until the maximum iteration was satisfied.

By varying the NTs from 1 to 9, the comparisons of the approximate solutions obtained by the FPA to the actual solutions for test problems 4, 5 and 6 are depicted in Figure 14 to 16. Others are omitted for brevity. A common trend can be observed that for the first few NTs, the algorithms were unable to predict the approximate solutions due to insufficient NTs. When sufficient NT was employed, the algorithms were able to approximate the exact solutions satisfactorily. However, the further increase in NT led to a decline in the approximate accuracy, due to more unknown coefficients were involved. Thus, more iterations and bigger population size were needed such that the MFO, FPA or GA can converge to the optimal solution. However, in this study, the population size and the

maximum number of iteration were fixed. Therefore, these algorithms could not reach the optimal solution within the allowable iterations for  $NT = 7$  and onwards for test problem 4.

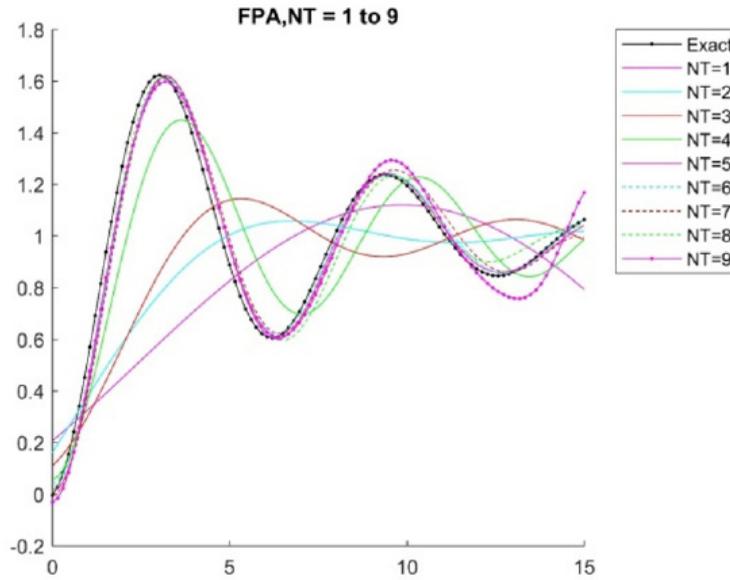


Figure 14. Test Problem 4 – Comparison of the obtained best approximate solutions with the exact solution for FPA by varying the NTs from 1 to 9

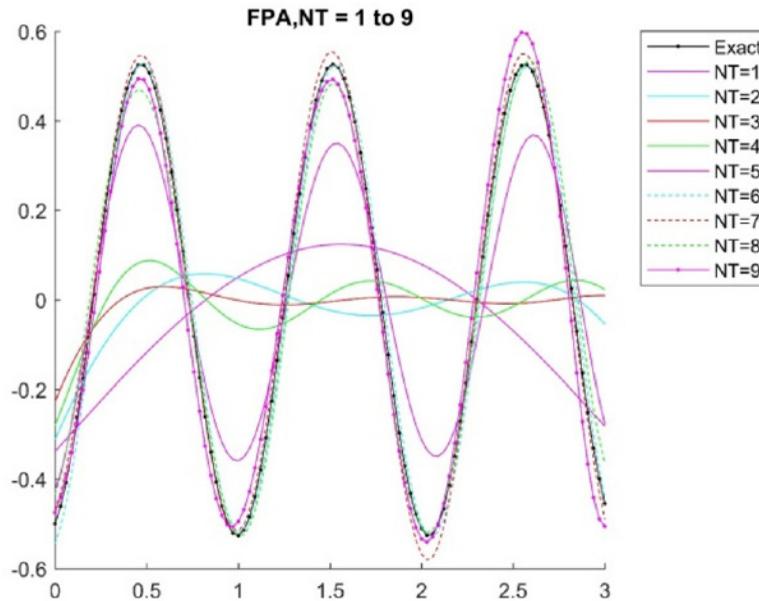


Figure 15. Test Problem 5 – Comparison of the obtained best approximate solutions with the exact solution for FPA by varying the NTs from 1 to 9

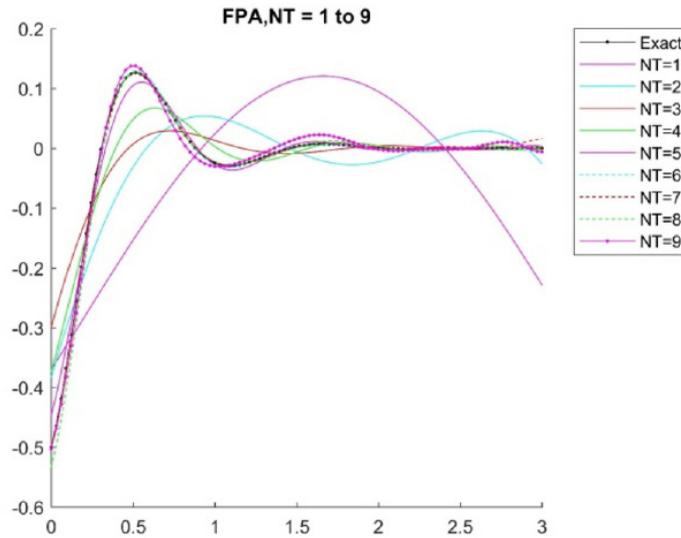


Figure 16. Test Problem 6 – Comparison of the obtained best approximate solutions with the exact solution for FPA by varying the NTs from 1 to 9

Table 16 summarizes the GD values of the best approximate solutions obtained by the MFO, FPA and GA optimizers for all test problems when different NTs were considered and depicted further in Figure 17. It can be observed that similar decreasing and increasing behaviours were shown by the optimizers. Initially, the GD value decreases with increasing NTs until it reaches the minimum. After this point, a further increase in NT was no longer proportional to the obtained approximation accuracies. It can be inferred that neither low nor high NTs guarantee high approximation accuracy. The NT = 3, NT = 5 or 6 and NT = 4 or 7 were recommended for MFO, FPA and GA, respectively, based on the obtained results.

Table 16

The best GD values obtained by MFO, FPA and GA optimizers for all test problems using different NTs

Algorithm	NT	Test Problem			
		1	2	3	4
MFO	1	5.0502E-03	7.7745E-03	3.9283E-04	2.6548E-02
	2	1.2662E-03	5.7775E-04	1.8552E-04	2.2889E-02
	3	2.9339E-04	<b>3.7683E-04</b>	<b>1.4702E-05</b>	2.2890E-02
	4	3.3058E-04	5.4313E-04	1.7945E-04	9.9603E-03

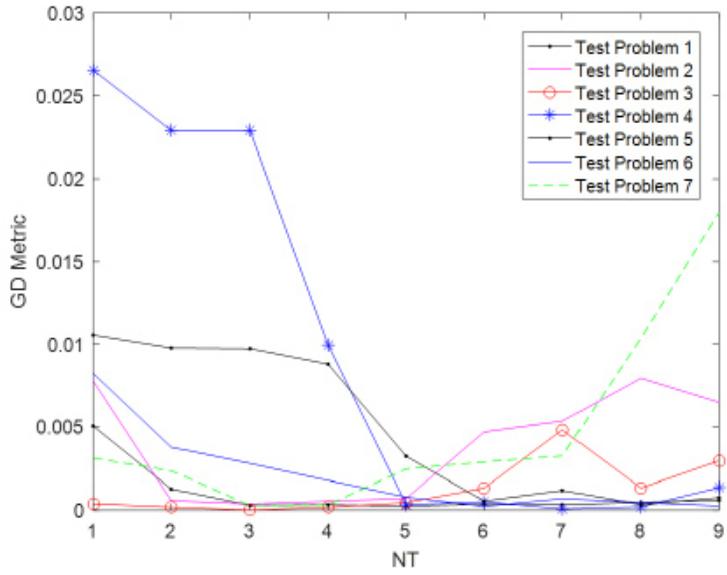
Table 16 (Continued)

Algorithm	NT	Test Problem			
		1	2	3	4
	5	2.4864E-04	7.0202E-04	4.5420E-04	3.4455E-04
	6	3.6552E-04	4.6924E-03	1.3182E-03	5.1476E-04
	7	3.2673E-04	5.3539E-03	4.8136E-03	<b>8.1944E-05</b>
	8	4.8252E-04	7.9415E-03	1.3306E-03	2.0955E-04
	9	5.9491E-04	6.4883E-03	2.9632E-03	1.3481E-03
FPA	1	5.0648E-03	7.5922E-03	3.9793E-04	2.6556E-02
	2	1.4192E-03	<b>4.6520E-04</b>	1.8879E-04	2.2934E-02
	3	2.3567E-04	1.0593E-03	<b>1.0827E-05</b>	2.2727E-02
	4	2.0434E-04	2.8545E-03	1.8994E-04	9.4469E-03
	5	<b>1.8545E-04</b>	6.5862E-03	2.4402E-04	<b>6.4572E-04</b>
	6	2.5799E-04	4.9055E-03	8.1734E-04	8.1790E-04
	7	4.0502E-04	1.1536E-02	6.3980E-04	1.4909E-03
	8	3.8971E-04	1.3597E-02	6.0018E-04	2.3740E-03
	9	7.4868E-04	1.9429E-02	2.5486E-03	2.9649E-03
GA	1	4.1139E-03	7.2000E-03	3.7000E-04	2.6540E-02
	2	1.1776E-03	6.3031E-04	1.2644E-04	2.2604E-02
	3	4.2029E-04	8.3174E-04	2.3991E-04	2.2937E-02
	4	<b>2.9718E-04</b>	5.3528E-04	<b>1.1964E-04</b>	1.2308E-02
	5	8.6889E-04	5.5544E-04	1.7728E-04	4.2365E-03
	6	4.7591E-04	<b>3.0247E-04</b>	1.9681E-04	2.7037E-04
	7	5.0370E-04	1.8568E-03	3.3357E-04	1.2700E-04
	8	3.5526E-04	1.0719E-03	4.0010E-04	<b>1.2113E-04</b>
	9	1.0009E-03	2.2051E-03	2.0830E-04	1.9080E-04
Algorithm	NT	Test Problem			
		5	6	7	
MFO	1	1.0556E-02	8.2317E-03	3.1265E-03	
	2	9.7771E-03	3.7754E-03	2.3641E-03	
	3	9.7323E-03	2.8035E-03	<b>2.8087E-04</b>	
	4	8.7953E-03	1.7855E-03	3.1540E-04	
	5	3.2337E-03	7.8302E-04	2.4732E-03	
	6	5.4773E-04	<b>2.3517E-04</b>	2.8963E-03	

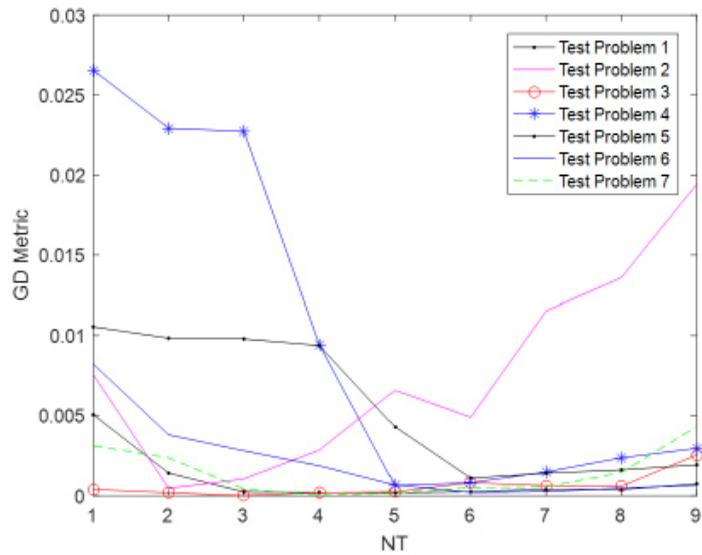
Table 16 (Continued)

Algorithm	NT	Test Problem		
		5	6	7
	7	1.1805E-03	6.9160E-04	3.2579E-03
	8	<b>3.3571E-04</b>	4.7393E-04	1.0370E-02
	9	7.5183E-04	2.6149E-04	1.7981E-02
FPA	1	1.0556E-02	8.2330E-03	3.1273E-03
	2	9.8460E-03	3.8064E-03	2.3766E-03
	3	9.8036E-03	2.8115E-03	4.2669E-04
	4	9.3868E-03	1.8730E-03	<b>3.2487E-05</b>
	5	4.2955E-03	6.8033E-04	1.6666E-04
	6	<b>1.1004E-03</b>	<b>2.0994E-04</b>	5.1030E-04
	7	1.4087E-03	2.5601E-04	5.6284E-04
	8	1.6190E-03	4.7710E-04	1.4553E-03
	9	1.9285E-03	6.5623E-04	4.3279E-03
GA	1	1.0556E-02	8.2328E-03	3.0958E-03
	2	9.8259E-03	3.8128E-03	2.0768E-03
	3	9.7890E-03	2.8533E-03	<b>2.6636E-04</b>
	4	9.2191E-03	2.0325E-03	4.9353E-04
	5	5.4457E-03	9.8756E-04	1.9471E-03
	6	5.7002E-04	4.5320E-04	2.2832E-03
	7	<b>5.5462E-04</b>	<b>2.8517E-04</b>	5.1471E-03
	8	7.3151E-04	3.0154E-04	3.5705E-03
	9	7.4270E-04	2.7269E-04	2.5460E-03

The bolded values are the lowest GD value for each test problem



(a)



(b)

Figure 17. Relationship of using different NTs with respect to approximation accuracies for all test problems (a) MFO; (b) FPA; (c) GA

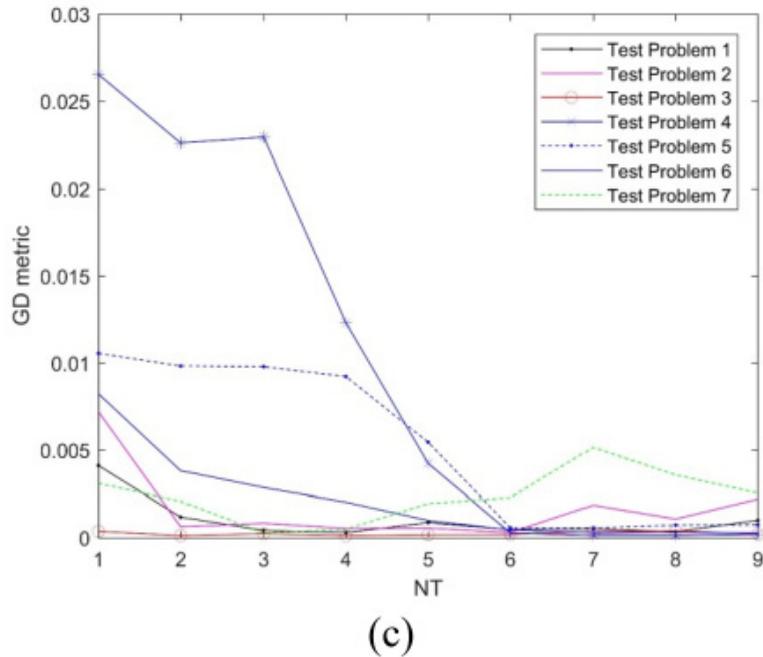


Figure 17. (Continued)

## CONCLUSIONS

In this work, finding the approximate solutions of ODEs with the formulation of Fourier series expansion, calculus of variation and metaheuristic algorithm, specifically, the MFO and FPA, was addressed. The approximate approach was examined through seven test problems, including first and second-order ODEs, system of ODE, ODE with Neumann boundary condition, mechanical engineering problems and longitudinal fins heat transfer problem. It was found that all approximate solutions were able to approximate the exact solutions satisfactorily, with the GD metric within the range  $1E-03$  to  $1E-05$ . Besides having reasonable accuracy, the approximate approach is promising in the sense that (i) it can be applied in a wide range of ODEs regardless of the types and orders, (ii) the procedures used to solve different ODEs can be computed using a general framework, (iii) the Fourier series can be differentiated infinitely and thus, it can be used as the approximate solution for higher-order ODEs, and (iv) the NT is adjustable to suit the personal requirement.

In terms of the dependency of approximation accuracy to the NT, it was found that the increment in NT was not proportional to the approximation accuracy. Though an increase of the NT may improve the approximation accuracy, in turn, it causes higher computational complexity due to the increasing number of Fourier coefficients. Thus, a moderate value of NT is suggested.

For future recommendations of this study, the lower and upper bounds of -1 and 1 can be increased to obtain a better approximate solution. Nonetheless, this will increase the computational time needed due to more possible coefficients to consider. Other than that, more newly invented optimization algorithms can be simulated and compared to see the feasibility of this method. Lastly, ODEs with different difficulties and fields should be tested to challenge the limits of this method.

## ACKNOWLEDGEMENT

The authors would like to express the deepest appreciation to the Ministry of Education Malaysia, for funding this project through the Fundamental Research Grant Scheme (FRGS – Vot K070, Reference Code FRGS/1/2018/ICT02/UTHM/02/2). Additional support from Universiti Tun Hussein Onn Malaysia (UTHM) in the form of GPPS Vot H034 is also gratefully acknowledged.

## REFERENCES

- Abassy, T. A., El-Tawil, M. A., & El-Zoheiry, H. (2007). Modified variational iteration method for Boussinesq equation. *Computers and Mathematics with Applications*, 54(7), 955-965.
- Adomian, G. (1988). A review of the decomposition method in applied mathematics. *Journal of Mathematical Analysis and Applications*, 135(2), 501-544.
- Almasi, M. H., Sadollah, A., Kang, S., & Karim, M. R. (2016). Optimization of an improved intermodal transit model equipped with feeder bus and railway systems using metaheuristics approaches. *Sustainability*, 8(6), 1-27.
- Babaei, M. (2013). A general approach to approximate solutions of nonlinear differential equations using particle swarm optimization. *Applied Soft Computing*, 13(7), 3354-3365.
- Babolian, E., Azizi, A., & Saeidian, J. (2009). Some notes on using the homotopy perturbation method for solving time-dependent differential equations. *Mathematical and Computer Modelling*, 50(1), 213-224.
- Beidokhti, R. S., & Malek, A. (2009). Solving initial-boundary value problems for systems of partial differential equations using neural networks and optimization techniques. *Journal of the Franklin Institute*, 346(9), 898-913.
- Biazar, J., & Hosseini, K. (2017). An effective modification of adomian decomposition method for solving emden–fowler type systems. *National Academy Science Letters*, 40(4), 285-290.
- Bouaziz, M., Rechak, S., Hanini, S., Bal, Y., & Bal, K. (2001). Numerical study of nonlinear heat transfer in longitudinal fins. *International Journal of Thermal Sciences*, 40(9), 843-857.
- Butcher, J. C. (2016). *Numerical methods for ordinary differential equations*. Chichester, UK: John Wiley & Sons.
- Chen, C. L., & Liu, Y. C. (1998). Differential transformation technique for steady nonlinear heat conduction problems. *Applied Mathematics and Computation*, 95(2), 155-164.

- Coşkun, S. B., & Atay, M. T. (2007). Analysis of convective straight and radial fins with temperature-dependent thermal conductivity using variational iteration method with comparison with respect to finite element analysis. *Mathematical Problems in Engineering*, 2007, 1-15.
- Coşkun, S. B., & Atay, M. T. (2008). Fin efficiency analysis of convective straight fins with temperature dependent thermal conductivity using variational iteration method. *Applied Thermal Engineering*, 28(17-18), 2345-2352.
- Das, R., & Kundu, B. (2019). Forward and inverse nonlinear heat transfer analysis for optimization of a constructal T-shape fin under dry and wet conditions. *International Journal of Heat and Mass Transfer*, 137, 461-475.
- Dash, P., Saikia, L. C., & Sinha, N. (2016). Flower pollination algorithm optimized pi-pd cascade controller in automatic generation control of a multi-area power system. *International Journal of Electrical Power and Energy Systems*, 82, 19-28.
- Domairry, G., & Fazeli, M. (2009). Homotopy analysis method to determine the fin efficiency of convective straight fins with temperature-dependent thermal conductivity. *Communications in Nonlinear Science and Numerical Simulation*, 14(2), 489-499.
- Ebaid, A. (2013). On a new differential transformation method for solving nonlinear differential equations. *Asian-European Journal of Mathematics*, 6(4), 1-12.
- Ebaid, A. E. (2011). A reliable aftertreatment for improving the differential transformation method and its application to nonlinear oscillators with fractional nonlinearities. *Communications in Nonlinear Science and Numerical Simulation*, 16(1), 528-536.
- El-Sayed, T. A., & El-Mongy, H. H. (2019). Free vibration and stability analysis of a multi-span pipe conveying fluid using exact and variational iteration methods combined with transfer matrix method. *Applied Mathematical Modelling*, 71, 173-193.
- Fouladi, F., Hosseinzadeh, E., Barari, A., & Domairry, G. (2010). Highly nonlinear temperature-dependent fin analysis by variational iteration method. *Heat Transfer Research*, 41(2), 155-165.
- He, J. H. (1999). Variational iteration method—a kind of non-linear analytical technique: some examples. *International Journal of Non-Linear Mechanics*, 34(4), 699-708.
- He, J. H. (2003). Homotopy perturbation method: A new nonlinear analytical technique. *Applied Mathematics and Computation*, 135(1), 73-79.
- Hussain, K., Ismail, F., & Senu, N. (2016). Solving directly special fourth-order ordinary differential equations using Runge–Kutta type method. *Journal of Computational and Applied Mathematics*, 306, 179-199.
- Hussin, C. H. C., Mandangan, A., Kilicman, A., Daud, M. A., & Juhan, N. (2016). Differential transformation method for solving sixth-order boundary value problems of ordinary differential equations. *Jurnal Teknologi*, 78(6-4), 13-19.
- Khani, F., Raji, M. A., & Nejad, H. H. (2009). Analytical solutions and efficiency of the nonlinear fin problem with temperature-dependent thermal conductivity and heat transfer coefficient. *Communications in Nonlinear Science and Numerical Simulation*, 14(8), 3327-3338.

- Lambert, J. (1986). A stable sequence of steplengths for Euler's rule applied to stiff systems of differential equations. *Computers and Mathematics with Applications*, 12(5-6), 1141-1151.
- Lee, Z. Y. (2006). Method of bilaterally bounded to solution Blasius equation using particle swarm optimization. *Applied Mathematics and Computation*, 179(2), 779-786.
- Mastorakis, N. E. (2006). Unstable ordinary differential equations: Solution via genetic algorithms and the method of Nelder-Mead. *WSEAS Transactions on Mathematics*, 5(12), 1-6.
- Mateescu, G. D. (2006). On the application of genetic algorithms to differential equations. *Romanian Journal of Economic Forecasting*, 7(2), 5-9.
- Mirjalili, S. (2015). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-Based Systems*, 89, 228-249.
- Mohammadi, F., & Hosseini, M. M. (2011). A comparative study of numerical methods for solving quadratic Riccati differential equations. *Journal of the Franklin Institute*, 348(2), 156-164.
- Nemati, K., Shamsuddin, S. M., & Darus, M. (2015). Solving initial and boundary value problems using learning automata particle swarm optimization. *Engineering Optimization*, 47(5), 656-673.
- Panda, A., & Pani, S. (2017). Determining approximate solutions of nonlinear ordinary differential equations using orthogonal colliding bodies optimization. *Neural Processing Letters*, 48(1), 219-243.
- Panda, A., & Pani, S. (2018). Determining approximate solutions of nonlinear ordinary differential equations using orthogonal colliding bodies optimization. *Neural Processing Letters*, 48(1), 219-243.
- Perfilov, V., Fila, V., & Marcano, J. S. (2018). A general predictive model for sweeping gas membrane distillation. *Desalination*, 443, 285-306.
- Rudd, K., & Ferrari, S. (2015). A constrained integration (CINT) approach to solving partial differential equations using artificial neural networks. *Neurocomputing*, 155, 277-285.
- Sadollah, A., Choi, Y., & Kim, J. H. (2015a). Metaheuristic algorithms for approximate solution to ordinary differential equations of longitudinal fins having various profiles. *Applied Soft Computing*, 33, 360-379.
- Sadollah, A., Choi, Y., Yoo, D. G., & Kim, J. H. (2015b). Metaheuristic algorithms for approximate solution to ordinary differential equations of longitudinal fins having various profiles. *Applied Soft Computing*, 33, 360-379.
- Sadollah, A., Eskandar, H., & Kim, J. H. (2015c). Approximate solving of nonlinear ordinary differential equations using least square weight function and metaheuristic algorithms. *Engineering Applications of Artificial Intelligence*, 40, 117-132.
- Sadollah, A., Eskandar, H., Yoo, D. G., & Kim, J. H. (2015d). Approximate solving of nonlinear ordinary differential equations using least square weight function and metaheuristic algorithms. *Engineering Applications of Artificial Intelligence*, 40, 117-132.
- Sadollah, A., Yadav, N., Gao, K., & Su, R. (2017). Metaheuristic optimisation methods for approximate solving of singular boundary value problems. *Journal of Experimental and Theoretical Artificial Intelligence*, 29(4), 823-842.

- Tian, X., Pang, W., Wang, Y., Guo, K., & Zhou, Y. (2019). LatinPSO: An algorithm for simultaneously inferring structure and parameters of ordinary differential equations models. *BioSystems*, 182, 8-16.
- Torabi, M., Aziz, A., & Zhang, K. (2013). A comparative study of longitudinal fins of rectangular, trapezoidal and concave parabolic profiles with multiple nonlinearities. *Energy*, 51, 243-256.
- Turkyilmazoglu, M. (2018). A reliable convergent Adomian decomposition method for heat transfer through extended surfaces. *International Journal of Numerical Methods for Heat and Fluid Flow*, 28(11), 2551-2566.
- Yang, X. S. (2012, September 3-7). Flower pollination algorithm for global optimization. In *International Conference on Unconventional Computing and Natural Computation* (pp. 240-249). Orléans, France.

