An Improved Rider Optimization Algorithm for Solving Engineering Optimization Problems

GUOHU WANG1, YONGLIANG YUAN2, AND WENWEN GUO1

1School of Mechatronics Engineering, Zhengzhou University of Industrial Technology, Zhengzhou 451100, China
2School of Mechanical Engineering, Dalian University of Technology, Dalian 116024, China

Corresponding author: Yongliang Yuan (yuanyongliang@mail.dlut.edu.cn)

ABSTRACT This paper proposed an improved optimization algorithm for engineering optimization problems. This improved optimization algorithm, namely, gravitational search strategy-assisted rider optimization algorithm (GSSROA), combines the gravitational search strategy (GSS) with the rider optimization algorithm (ROA). In the GSSROA, the search step can be adaptively adjusted in each iteration. The performance of the GSSROA has been verified with three benchmark engineering design problems. The results indicate that the proposed GSSROA has the feasibility and good robustness. Then, the GSSROA is applied to optimize the weight of the front axle of an automobile under the non-linear constraints. The results show that the GSSROA can achieve the optimal structure with a weight that is reduced by 16.82% compared with the initial design.

INDEX TERMS Rider optimization algorithm, gravitational search strategy, engineering design problems, parameter identification.

I. INTRODUCTION

Structural optimization is a challenging and critical research work. The aim of the structural optimization is to maximize profits under the certain conditions. Many optimization problems in real-world are the nonlinear constraint problem, which usually are solved using the classical optimization algorithms [1]–[3]. It is well-known that the classical approaches are often inefficient and need the strong math assumptions, which are usually difficult to establish the mathematical model in the practical problems. With increasing complexity of the problems, the traditional optimization algorithm needs higher computational cost to obtain the global optimal solution. To handle these problems, many meta-heuristic algorithms have been proposed in past decades, such as genetic algorithms (GA) [4], particle swarm optimization (PSO) [5], pattern search (PS) [6], differential evolution (DE) [7], genetic programming (GP) [8], central force optimization (CFO) [9], ant colony optimization (ACO) [10] and rider optimization algorithm (ROA) [11].

Although these optimization algorithms have been successfully applied in the real-world problems, the high computational cost and accuracy are still the challenge work.

According to No Free Lunch (NFL) theorem [12], there is no one algorithm which can solve all the optimization problems. For addressing this challenge, many researchers have focus on improving the performance of the algorithms using different optimization strategies, such as socio evolution and learning optimization (SELO) [13], adaptive genetic algorithms (AGA) [14] and fitness-distance-ratio-based PSO (FDR-PSO) [15], and so forth.

Among these meta-heuristic optimization algorithms, the ROA is a novel optimization algorithm, which is first proposed by Binu in 2017 [11], which considers a few rider groups. ROA has been used for fault diagnosis in analog circuits, the results show that ROA can attain the global optimal solutions. To speed up the convergence rate, the gravity search strategy is applied to the ROA in this paper, which considers the individual inter-relation. The performance of improve rider optimization algorithm (IROA) is verified on three benchmark engineering optimization problems. The results of IROA show that it has strong reliability and robust exploration performances. Then, IROA is applied to evaluate the parameters for automobile front axle. Results show that IROA is more competitive in the computational efficiency and accuracy.

The remainder of the paper is organized as follows. The review of rider optimization algorithm is described in

The associate editor coordinating the review of this manuscript and approving it for publication was Chung Shue Chen.
Section 2. Section 3 develops the gravity search strategy and improved rider optimization algorithm. The benchmark engineering optimization problems are discussed in Section 4. The optimization of the automobile front axle is dealt with in section 5. Section 6 gives the concluding remarks.

II. REVIEW OF RIDER OPTIMIZATION ALGORITHM

The inspiration of ROA is a group of riders, including the bypass rider, follower, overtaker, and attacker. In ROA, all the riders should follow the predefined strategy, which can be described as follows:

1) The bypass rider can reach the target with the leading path.

   The group riders initialization can be set as follows:
   \[ X_t = \{X_i(i,j)\}, \quad 1 \leq i \leq R; \ 1 \leq j \leq Q \] (1)
   where \(R\) is the number of the riders, \(Q\) denotes the dimension of the optimization problem, \(t\) denotes the time instant, \(X_t\) is the position of \(i\)th rider at time \(t\). For the bypass rider, it implies that he is the rider in the leading position, and the position can be calculated as follows:
   \[ X_{t+1}^B(i,j) = \delta [X_t(\eta,j)^*\beta(j) + X_t(\xi,j)^*[1-\beta(j)]] \] (2)
   where \(\delta\) is a random within [0,1], \(\eta\) denotes the \(a\) random number within [1, \(R\)], \(\xi\) is a number which can select from 1 to \(R\). In addition, \(\beta\) represents a random within [0,1], but of size \(1 \times Q\).

2) The follower arrives at the destination by following the bypass rider.

   The position of follower is relevant to bypass rider, the main reason for this is that the walk path of the follower depends on the bypass rider. Thus, the position of follower can be updated using Eq. (3).
   \[ X_{t+1}^F(i,k) = X^L(L,k) + \left[ \cos (T_{i,k}^l)^* X^L(L,k)^* d_i^l \right] \] (3)
   where \(k\) represents the coordinate selector, \(X^L\) denotes the position of bypass rider, \(L\) is the index of bypass rider, \(T_{i,k}^l\) denotes the steering angle of the \(i\)th rider in the \(k\)th coordinate, and \(d_i^l\) is the distance to be traveled of the \(i\)th rider.

3) The overtaker not only follows his own position, but also the information are gathered according to the bypass rider.

   The position of overtaker is in dependence on three factors, including the coordinate selector, relative success rate and direction indicator. The main reason for this is that the overtaker needs to gather the information of the bypass rider and his own. Therefore, the overtaker’s position can be updated by Eq. (4).
   \[ X_{t+1}^O(i,k) = X_t(i,k) + [D_i^l(i)^* X^L(L,k)] \] (4)

4) The attacker uses the maximum speed to reach the target point.

   The purpose of the attacker is to obtain the position of the leader, and the travelling strategy is similar to the follower. Notably, all the attackers’ position are updated in this strategy, rather than the selected individual. The position of the attacker can be set as follows:
   \[ X_{t+1}^A(i,j) = X^L(L,j) + \left[ \cos (T_{i,j}^l)^* X^L(L,j)^* d_i^l \right] \] (6)
   where \(X^L(L,j)\) represents the leader’s position, \(T_{i,j}^l\) denotes the steering angle of the \(i\)th rider in the \(j\)th coordinate.

   The pseudo-code of the ROA is shown in Fig. 1.

Algorithm 1. The pseudo-code of ROA.

Input: Random position of the riders \(X_t\).
Output: Leadering rider \(X^L\).

while stopping criteria is not satisfied do
     Initialize the population;
     Initialize the rider parameters: Steering angle \(T\), Gear \(E\), Accelerator \(e\), and Brake \(K\);
     Find the success rate \(r_i\);
     while \(t < T_{off}\) do
         for \(i = 1\) to \(R\) do
             Update the position of bypass rider using Eq. (2);
             Update the position of follower using Eq. (3);
             Update the position of overtaker using Eq. (4);
             Update the position of attacker using Eq. (6);
             Rank the riders based on \(r_i\);
             Select the rider having the maximum \(r_i\) as the leading rider;
             Update \(T, E, e\) and \(K\);
             \(t = t + 1\);
         end for
     end while
end

FIGURE 1. The pseudo-code of the ROA.
III. AN IMPROVED RIDER OPTIMIZATION ALGORITHM

A. GRAVITATIONAL SEARCH STRATEGY (GSS)

In this section, the gravitational search strategy (GSS) is introduced. Gravitation search algorithm is first proposed by Rashedi et al. in 2009 [16], which has been widely used to resolve the optimization problems in the literature [17]–[21]. The basis of gravitation search algorithm is the Newton gravitational law, i.e., each individual attracts every other individual with a force. In gravitation search algorithm, each individual consist of four parts, including the position of the individual, its inertial mass, its active gravitational mass and passive mass. Correspondingly, the gravitational and inertial mass can be consider as the fitness functions, and the position of the mass is supposed to the solution of the optimization problem. Suppose that mass \( j \) acts mass \( i \), the gravitational force can be stated as follows:

\[
F_{ij}^d = \frac{G M_i \times M_j}{R_{ij} + \varepsilon} (x_j^d - x_i^d) \tag{7}
\]

where \( M_i \) and \( M_j \) are the mass of the \( i \)th and \( j \)th individual, respectively. \( G \) denotes the gravitational constant. \( \varepsilon \) is the a small constant. \( R_{ij} \) denotes the distance of the between \( i \)th and \( j \)th individual, which can be calculated as follows:

\[
R_{ij} = \|X_i, X_j\|_2 \tag{8}
\]

The total force acting on the \( i \)th individual can be stated as follows:

\[
F_i^d = \sum_{j \in k_{best}, j \neq i} N \cdot F_{ij}^d \tag{9}
\]

where \( \text{rand} \) represents a random within \([0, 1]\). \( k_{best} \) is the set of the first individuals with the best function values. According to the Newton’s second law, the acceleration of the inertial mass \( M_i \) can be calculated as follows:

\[
a_i^d = \frac{F_i^d}{M_i} \tag{10}
\]

B. IMPROVED RIDER OPTIMIZATION ALGORITHM

An improved optimization algorithm is proposed in this section, which is named the gravitational search strategy based on ROA (GSSROA). In GSSROA, the correlation between rider was taken into account. To clearly describe the search process, the position of the rider can be defined by Eq. (11).

\[
X_i = (x_i^1, \cdots, x_i^d, \cdots, x_i^N), \quad i = 1, 2, \cdots, N \tag{11}
\]

where \( x_i^d \) is the position of the \( i \)th rider in the \( d \)th dimension.

The acting force can be given by:

\[
F_{ij} (t) = g (t) \frac{m_i \times m_j}{l_{ij} + \varepsilon} (x_j (t) - x_i (t)) \tag{12}
\]

where \( m_i \) and \( m_j \) are the best fitness of the \( i \)th and \( j \)th rider. \( g(t) \) denotes the search constant at the iteration \( t \). \( l_{ij} \) is the distance of the between \( i \)th and \( j \)th rider.

**Algorithm 2. The pseudo-code of GSSROA.**

**Input:** Random position of the riders \( X_i \).

**Output:** Leading rider \( X^L \).

while stopping criteria is not satisfied do

Initilize the population;

Initialize the rider parameters: Steering angle \( T \), Gear \( E \), Accelerator \( e \), and Brake \( K \);

Find the success rate \( r_1 \);

while \( t < T_{off} \) do

for \( i = 1 \) to \( R \) do

Calculate the acceleration of bypass rider, follower, overtaker, and attacker, respectively, using Eq. (14);

Update the position of bypass rider using Eq. (15);

Update the position of follower using Eq. (16);

Update the position of overtaker using Eq. (17);

Update the position of attacker using Eq. (18);

Rank the riders based on \( r_1 \);

Select the rider having the maximum \( r_1 \) as the leading rider;

Update \( T, E, e \) and \( K \);

Return \( X^L \);

\( t = t + 1 \);

end for

end while

end

The total acting force and acceleration of the \( i \)th rider can be calculated as follows:

\[
F_i^d = \sum_{j \in k_{best}, j \neq i} N \cdot F_{ij}^d \tag{13}
\]

\[
a_i^d = \frac{F_i^d}{M_i} \tag{14}
\]

Therefore, the position of bypass rider, follower, overtaker, and attacker can be updated as follows:

\[
X_{t+1}^B (i, j) = \delta [X_t (\eta, j)^* \beta_{(j)} + X_t (\xi_j)^* [1 - \beta_{(j)}]] + a_B \tag{15}
\]

\[
X_{t+1}^F (i, k) = X^L (L, k) + \left[ \cos \left( T_{ik}^L \right)^* X^L (L, k)^* d_i^L \right] + a_F \tag{16}
\]

\[
X_{t+1}^O (i, k) = X_t (i, k) + \left[ D_i^L \left( i^* X^L (L, k) \right) \right] + a_O \tag{17}
\]

\[
X_{t+1}^A (i, j) = X^L (L, j) + \left[ \cos \left( T_{ij}^L \right)^* X^L (L, j)^* d_i^L \right] + a_A \tag{18}
\]

where \( a_B, a_F, a_O \) and \( a_A \) denote the acceleration of the bypass rider, follower, overtaker, and attacker, respectively.

The pseudo-code of the GSSROA is shown in Fig. 2.
can be described as follows: the mathematical formulation of hydrodynamic thrust bearing discs, actuating force, and number of friction surfaces. The variables, including inner radius, outer radius, thickness of discs, actuating force, and number of friction surfaces. The optimization problem consists of five decision variables, including inner radius, outer radius, thickness of discs, actuating force, and number of friction surfaces. The mathematical formulation of hydrodynamic thrust bearing can be described as follows:

$$\min f(x) = \pi t \left( r_o^2 - r_i^2 \right) (Z + 1) \rho$$

s.t. \( g_1(x) = r_o - r_i - \Delta r \geq 0 \)

$$g_2(x) = t_{\text{max}} - (Z + 1) (t + \delta) \geq 0$$

$$g_3(x) = p_{\text{max}} - p_{r_c} \geq 0$$

$$g_4(x) = p_{r_c} v_{sr_{\text{max}}} - p_{r_c} v_{sr} \geq 0$$

$$g_5(x) = t_{sr_{\text{max}}} - v_{sr} \geq 0$$

$$g_6(x) = T_{\text{max}} - T \geq 0$$

$$g_7(x) = M_h - sM_p \geq 0$$

$$g_8(x) = T \geq 0 \geq 0$$  \hspace{1cm} (19)

where\[ M_h = \frac{2}{3} \mu F Z \left( r_o^3 - r_i^3 \right) p_{r_c} = \frac{F}{\pi \left( r_o^2 - r_i^2 \right)} \]

$$v_{sr} = \frac{2\pi n (r_o^3 - r_i^3)}{90 \left( r_o^2 - r_i^2 \right)}$$

$$T = \frac{1.5 \pi n}{30 \left( M_h + M_f \right)}$$

$$\Delta r = 20 \text{mm}, \quad 1.5 \text{mm} \leq t \leq 3 \text{mm} \quad t_{\text{max}} = 30 \text{mm}$$

$$Z_{\text{max}} = 10, \quad v_{sr_{\text{max}}} = 10 \text{m/s}, \quad \mu = 0.5, \quad s = 1.5$$

$$M_s = 40 \text{Nm}, \quad M_f = 3 \text{Nm}, \quad n = 250 \text{rpm},$$

The mathematical formulation of hydrodynamic thrust bearing can be described as follows:

$$\min f(x) = (2\sqrt{2}x_1 + x_2) \times l$$

s.t. \( g_1(x) = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2} - p - \sigma \leq 0 \)

$$g_2(x) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2} - p - \sigma \leq 0$$

IV. ENGINEERING DESIGN OPTIMIZATION PROBLEMS

To verify the performance of the GSSROA, three engineering optimization problems are adopted, including the multiple disc clutch brake [22]–[23], three-bar truss design [24] and hydrodynamic thrust bearing [22]–[23]. The benchmark optimization problems are described in detail in the following sub-sections.

A. MULTIPLE DISC CLUTCH BRAKE

The multiple disc clutch brake optimization design problem is to find the minimize the power loss [22], [23]. As shown in Fig. 3, the optimization problem consists of five decision variables, including inner radius, outer radius, thickness of discs, actuating force, and number of friction surfaces. The mathematical formulation of hydrodynamic thrust bearing can be described as follows:

$$\min f(x) = \frac{F}{\pi \left( r_o^2 - r_i^2 \right)}$$

where

$$M_h = \frac{2}{3} \mu F Z \left( r_o^3 - r_i^3 \right)$$

$$v_{sr} = \frac{2\pi n (r_o^3 - r_i^3)}{90 \left( r_o^2 - r_i^2 \right)}$$

$$T = \frac{1.5 \pi n}{30 \left( M_h + M_f \right)}$$

$$\Delta r = 20 \text{mm}, \quad 1.5 \text{mm} \leq t \leq 3 \text{mm} \quad t_{\text{max}} = 30 \text{mm}$$

$$Z_{\text{max}} = 10, \quad v_{sr_{\text{max}}} = 10 \text{m/s}, \quad \mu = 0.5, \quad s = 1.5$$

$$M_s = 40 \text{Nm}, \quad M_f = 3 \text{Nm}, \quad n = 250 \text{rpm},$$

Table 1 reports the results of the GSSROA and the results from Refs. [22], [25], [26]. Clearly, the GSSROA can obtain the optimal solution with 81859.74, whose optimal value identical to these of the ABC [22] and TLBO [22]. However, the mean values and worst values are different. The main reason for this is that these optimization algorithms have different convergence rate, so that the robustness of the algorithms are different. The performances of WCA [25] and MBA [26] are worse than the GSSROA. Therefore, we can conclude that GSSROA has a good robustness.

B. THE THREE-BAR TRUSS DESIGN

The three-bar truss design optimization problem is chosen from the Ref. [24], which is a widely used benchmark engineering design problem. The purpose of the three-bar truss design is to minimize the weight. The objective function is simple, and it subjects to the highly non-linear constraint, including the stress, deflection, and buckling constraints. The configuration of the three-bar plane framework is shown in Fig. 4.

The mathematical formulation of three-bar truss design problem can be described as follows:

$$\min f(x) = (2\sqrt{2}x_1 + x_2) \times l$$

s.t. \( g_1(x) = \frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2} - p - \sigma \leq 0 \)

$$g_2(x) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2} - p - \sigma \leq 0$$

![FIGURE 3. Multiple disc clutch brake.](image-url)

![FIGURE 4. The three-bar truss design.](image-url)

| Algorithm  | Best       | Mean       | Worst       |
|------------|------------|------------|-------------|
| ABC [22]   | 81859.74   | 81496.00   | 78897.81    |
| TLBO [22]  | 81859.74   | 81438.98   | 80807.85    |
| WCA [25]   | 85539.48   | 83847.16   | 83942.71    |
| MBA [26]   | 85535.96   | 85321.40   | 84440.19    |
| GSSROA     | 81859.74   | 81428.54   | 80987.94    |

![TABLE 1. The results of multiple disc clutch brake.](image-url)
The mathematical formulation of the hydrodynamic thrust optimization problem consists of seven non-linear constraints. Figure 5 shows the hydrodynamic thrust bearing. The optimized results of these popular algorithms for the three-bar truss design are reported in Table 2. Interestingly, these algorithms can obtain the optimal solution, the main reason for this is that the three-bar truss design optimization problem is simple and consists of only two design variables. In view of the results of five algorithms, it can be concluded that the GSSROA has the feasibility and good robustness. The optimization results of GSSROA are compared with the results from Ref. [22], which are listed in Table 3. Obviously, the GSSROA can obtain the optimal solution with 1625.44, whose optimal value identical to these of the ABC [22] and TLBO [22]. By comparing the results of these optimization algorithms, it can be concluded that the GSSROA demonstrates a satisfactory exploration ability and can be used to solve the engineering optimization problems.

V. FRONT AXLE OF AN AUTOMOBILE OPTIMIZATION PROBLEM

The front axle of an automobile is a typical optimization problem in real life [33]. As shown in Fig. 6, this case is a 1-beam structure, which has been widely used in the modern engineering designs because of its high bending strength and light weight. The purpose of the front axle of an automobile is to find the minimum weight. The maximum normal stress and shear stress are \( \sigma = M/W_s \) and \( \tau = T/W_\rho \), respectively, where \( M \) and \( T \) are the time-dependent bending moment and torque, respectively. The values of \( M \) and \( T \) are given by:

\[
M = M_0 \left( \frac{1}{10} \cos \frac{t}{10} + \frac{1}{10} \right)
\]
FIGURE 6. Diagram of the automobile front axle. (a) Diagram of the front axle. (b) Cross section of the front axle.

TABLE 4. Optimization results for the automobile front axle.

| Design variable | Initial value | Optimal value | Improvement (%) |
|-----------------|---------------|---------------|-----------------|
| $a$ (mm)        | 15.99         | 13.21         | -17.39          |
| $b$ (mm)        | 70.10         | 63.75         | -9.06           |
| $c$ (mm)        | 20.67         | 19.05         | -7.84           |
| $h$ (mm)        | 100.07        | 87.12         | -12.94          |
| $M_0$ (N.mm)    | 40777.26      | 38358.27      | -5.93           |
| $T_0$ (lb-in.)  | 35708.90      | 36054.47      | +0.97           |
| Weight (Kg)     | 51.95         | 43.21         | -16.82          |

\[
T = T_0 \sin \frac{t}{3} \tag{23}
\]

\[
W_x = \frac{a(h - 2c)^3}{6h} + \frac{b}{6h}\left[h^3 - (h - 2c)^3\right] \tag{24}
\]

\[
W_\rho = 0.8bc^2 + \frac{0.4a^3(h - 2c)}{c} \tag{25}
\]

where $a$, $b$, $c$ and $h$ are shown in Fig. 6(b); $M_0$ and $T_0$ are the moment and torque; and $t$ is the time parameter within $[0,1]$. The strength of the front axle can be given by Eq. (26):

\[
g = \sigma_s - \sqrt{\sigma_s^2 + 3\tau^2} \tag{26}
\]

where $\sigma_s$ is the limit yield stress.

The optimization results of the front axle model are listed in the Table 4. Notably, “+” (“-”) denotes the increase (decrease). As seen from Table 4, it is clearly to note that the weight of the automobile front axle decreases from 51.95 Kg to 43.21 Kg, 16.82% lower than the initial value. These results further demonstrate that the GSSROA has a satisfactory exploration ability, and is an efficient algorithm to solve this engineering optimization problem.

VI. CONCLUSION

As no single meta-heuristic algorithm can obtain the global optimal solutions for all optimization problems. To address the challenge, this paper proposes a new optimization algorithm, namely gravitational search strategy assisted rider optimization algorithm (GSSROA). The performance of GSSROA is firstly evaluated on three engineering optimization problems are adopted, including the multiple disc clutch brake, three-bar truss design and hydrodynamic thrust bearing. By comparing the results of these optimization algorithms, it can be concluded that the GSSROA demonstrates a satisfactory exploration ability.

In addition, GSSROA is applied to optimize the front axle of an automobile. Results show that GSSROA is more competitive in the computational efficiency and accuracy. In the future work, GSSROA can be applied to multi-objective optimization problems.

REFERENCES

[1] C. Blum, J. Puchinger, G. R. Raidl, and A. Roli, “Hybrid metaheuristics in combinatorial optimization: A survey,” Appl. Soft Comput., vol. 11, pp. 4135–4151, Sep. 2011.

[2] I. Boussaïd, J. Lepagnot, and P. Siarry, “A survey on optimization meta-heuristics,” Inf. Sci., vol. 237, pp. 82–117, Jul. 2013.

[3] A. Gogna and A. Tayal, “Metaheuristics: Review and application,” J. Exp. Theor. Artif. Intell., vol. 25, no. 4, pp. 26–503, 2013.

[4] M. AlRashidi, M. AlHajri, K. El-Naggar, and A. Al-Othman, “Three, five and seven PV model parameters extraction using PSO,” Sol. Energy., vol. 85, no. 7, pp. 1543–1550, 2011.
[5] M. Ye, X. Wang, and Y. Xu, “Parameter extraction of solar cells using particle swarm optimization,” J. Appl. Phys., vol. 105, no. 9, 2009, Art. no. 094502.

[6] M. AlHajri, K. El-Naggar, M. R. AlRashidi, and A. K. Al-Othman, “Optimal extraction of solar cell parameters using pattern search,” Renew. Energy, vol. 44, pp. 238–245, Aug. 2012.

[7] R. Storn and K. Price, “Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces,” J. Glob. Optim., vol. 11, pp. 341–359, Dec. 1997.

[8] J. R. Koza, Genetic Programming II, Automatic Discovery of Reusable Subprograms, Cambridge, MA, USA: MIT Press, 1992.

[9] R. A. Formato, “Central force optimization,” Prog. Electromagn. Res., vol. 77, pp. 425–491, Aug. 2007.

[10] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant system: Optimization by a colony of cooperating agents,” IEEE Trans. Syst. Man, Cybern. B, Cybern., vol. 26, no. 1, pp. 29–41, Feb. 1996.

[11] D. Binu and B. S. Kariyappa, “RideNN: A new rider optimization algorithm-based neural network for fault diagnosis in analog circuits,” IEEE Trans. Instrum. Meas., vol. 68, no. 1, pp. 2–26, Jan. 2018.

[12] D. H. Wolper and W. G. Macready, “No free lunch theorems for optimization,” IEEE Trans. Evol. Comput., vol. 1, no. 1, pp. 67–82, Apr. 1997.

[13] T. Peram K. Veeramachaneni, and C. K. Mohan, “Fitness-distance-ratio based particle swarm optimization,” in Proc. IEEE Swarm Intell. Symp., vol. 6, no. 5, Apr. 2003, pp. 174–181.

[14] Y. L. Yuan, X. G. Song, W. Sun, and X. B. Wang, “Multidisciplinary design optimization of the belt drive system considering both structure and vibration characteristics based on improved genetic algorithm,” AIP Adv., vol. 8, no. 5, 2018, Art. no. 055115.

[15] M. Kumar, A. J. Kulkarni, and S. C. Satapathy, “Socio evolution & learning—optimization algorithm: A socio-inspired optimization methodology,” Future Gene. Comput. Syst., vol. 81, pp. 252–272, Apr. 2018.

[16] E. Rashidi, H. Nezamabadi-Pour, and S. Saryazdi, “GSA: A gravitational search algorithm,” Inform. Sci., vol. 179, no. 13, pp. 2232–2248, 2009.

[17] C. Li and J. Zhou, “ZhouParameters identification of hydraulic turbine governing system using improved gravitational search algorithm,” Energy Convers. Manage., vol. 52, no. 1, pp. 374–381, Jan. 2011.

[18] B. Zibanbehzad, K. Zamanifar, N. Nematzadeh, and F. Mardukhi, “An approach for Web services composition based on QoS and gravitational search algorithm,” in Proc. Innov. Inf. Technol. Conf., Dec. 2010, pp. 340–344.

[19] H. R. Hassanzadeh and M. Rouhani, “A multi-objective gravitational search algorithm,” in Proc. Commun. Syst. Netw. Conf., Jul. 2010, pp. 7–12.

[20] S. R. Balachandar and K. Kannan, “Meta-heuristic algorithm for set covering problem based on gravity,” Int. J. Comput. Math. Sci., vol. 4, pp. 223–228, Jul. 2010.

[21] S. Duman, N. C. Sahu, and P. K. Hota, “Gravitational search algorithm for economic dispatch with valve-point effects,” Int. Rev. Electr. Eng., vol. 5, no. 6, pp. 2890–2895, 2010.

[22] R. V. Rao, V. J. S. Savsani, and D. P. Vakharia, “Teaching-learning-based optimization: A novel method for constrained mechanical design optimization problems,” Comput. Aided Des., vol. 43, no. 3, pp. 303–315, Mar. 2011.

[23] P. Savsani and V. Savsani, “Passing vehicle search (PVS): A novel metaheuristic algorithm,” Appl. Math. Model., vol. 40, nos. 5–6, pp. 3951–3978, 2016.

[24] M. Abdel-Basset, G. G. Wang, A. K. A. K. Sangiah, and E. Rushdy, “Krill herd algorithm based on cuckoo search for solving engineering optimization problems,” Multimed. Tools Appl., vol. 78, no. 4, pp. 3861–3884, Feb. 2019.

[25] H. Eskandar, A. Sadollah, A. Bahreininejad, and M. Hamdi, “Water cycle algorithm—A novel metaheuristic optimization method for solving constrained engineering optimization problems,” Comput. Struct., vols. 110–111, pp. 151–166, Nov. 2012.