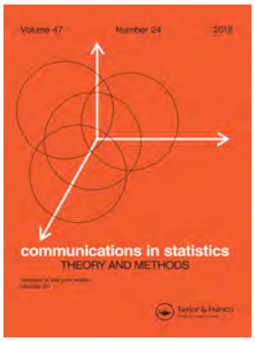

Communications in Statistics - Theory and Methods

ISSN: 0361-0926 (Print) 1532-415X (Online) Journal homepage: https://www.tandfonline.com/loi/lsta20

Hybrid differential evolutionary strawberry algorithm for real-parameter optimization problems

Wali Khan Mashwani, Abdullah Khan, Atila Göktaş, Yuksel Akay Unvan, Ozgur Yaniay & Abdelouahed Hamdi

To cite this article: Wali Khan Mashwani, Abdullah Khan, Atila Göktaş, Yuksel Akay Unvan, Ozgur Yaniay & Abdelouahed Hamdi (2020): Hybrid differential evolutionary strawberry algorithm for real-parameter optimization problems, Communications in Statistics - Theory and Methods, DOI: 10.1080/03610926.2020.1783559

To link to this article: https://doi.org/10.1080/03610926.2020.1783559

Published online: 07 Jul 2020.

Submit your article to this journal

Article views: 52

View related articles

View Crossmark data

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=lsta20

COMMUNICATIONS IN STATISTICS—THEORY AND METHODS
https://doi.org/10.1080/03610926.2020.1783559

Taylor & Francis
Taylor & Francis Group

Check for updates

Hybrid differential evolutionary strawberry algorithm for real-parameter optimization problems

Wali Khan Mashwani[a], Abdullah Khan[a], Atila Göktaş[b], Yuksel Akay Unvan[c], Ozgur Yaniay[d], and Abdelouahed Hamdi[e]

[a]Institute of Numerical Sciences, Kohat University of Science & Technology, Kohat, Pakistan; [b]Department of Statistics, Muğla Sıtkı Koçman University, Bodrum, Turkey; [c]Department of Banking and Finance, Ankara Yildirim Beyazit University, Ankara, Turkey; [d]Department of Statistics, Hacettepe University, Ankara, Turkey; [e]Department of Mathematics, Statistics and Physics, Qatar University, Doha, Qatar

ABSTRACT
Evolutionary algorithms (EAs) is a family of population-based nature optimization methods. In contrast to classical optimization techniques, EAs provide a set of approximated solutions for different test suites of optimization and real-world problems in single simulation. In the last few years, hybrid EAs have received much attention by utilizing the valuable aspects of different nature of search strategies. Hybrid EAs are quite efficient in handling various optimization and search problems having had high complexity, noisy environment, imprecision, uncertainty and vagueness. In this article, a hybrid differential evolutionary strawberry algorithm (HDEA) is suggested to utilize the propagating behavior of the strawberry plant and perturbation process of differential evolution (DE) algorithm in order to evolve their population set of solutions. The proposed algorithm employs DE as a substitute while replacing the runners of the strawberry plant to effectively explore and exploit the search space of the problem at hand. The numerical results found by the proposed algorithm over most benchmark functions after extensive experiments are much promising in terms of proximity and diversity.

ARTICLE HISTORY
Received 20 December 2018
Accepted 10 June 2020

KEYWORDS
Optimization; computing; evolutionary computation

MATHEMATICS SUBJECT CLASSIFICATION
90C59; 68W50

1. Introduction

Optimization finds out the most suitable value for the function with bounded search domain. Optimization problems are naturally posed as real-world problems (Lasisi et al. 2019). Optimization has wide applications in various engineering technologies, mathematics, operations research, economics and medical sciences. In essence, optimization problems can be categorized into constrained and unconstrained ones. In constrained problems, different restrictions are imposed over objective functions while in unconstrained problems the search space is bounded (Mashwani et al. 2019, 2020). The main study in this article is dedicated to the analysis of the unconstrained optimization problems with real parameters. A general optimization problem can be formulated as follows:

CONTACT Abdelouahed Hamdi abhamdi@qu.edu.qa Department of Mathematics, Statistics and Physics, Qatar University, Doha, Qatar.
© 2020 Taylor & Francis Group, LLC

$$\begin{aligned} \text{Minimize } & F(x) = f_1(x), f_2(x), \dots, f_m(x) \\ & g_i(x) \le 0, i = 1, 2, \dots, p \\ & h_j(x) = 0, j = 1, 2, \dots, q \\ & x_l^i \le x^i \le x_u^i, i = 1, 2, \dots, N \end{aligned} \tag{1}$$

where $x = (x_1, x_2, \dots, x_n)^T \in \Omega$ is the candidate solution with n decision variables or real parameters, $F(x)$ is the objective function, $g_i(x) \le 0, i = 1, 2, \dots, p$ and $h_j(x) = 0, j = 1, 2, \dots, q$ are the p inequality constraints and q equality constraints.

If $m = 1$, then problem (1) becomes a single objective optimization problem, while if $m \ge 2$, then problem (1) is a multi-objective problem. Furthermore, if Ω is a closed and connected region in R^n with all objective functions described in real-valued variables, then problem (1) is called a continuous multi-objective problem (Mashwani et al. 2017). In single-objective optimization, problem (1) is said to be continuous, if all decision variables, $x_1, x_2, \dots, x_n$, are expressed in real numbers (Khanum et al. 2018). With this convention, binary or Boolean variables are treated as integer variables (Lasisi et al. 2019; Yoshida 2010).

In general, optimization techniques can be categorized into linear and non linear optimization techniques.[1] Linear optimization techniques are simple and straightforward as compared to non linear optimization techniques. Non linear optimization methods (Fiacco and McCormick 1968; Ruszczynski 2006; Miller 1999) are further subdivided into two classes including non linear local and global search algorithms. The major difference between aforementioned sub-divisions is that local search methods furnish local optimum while global search methods render global optimum solutions for dealing with optimization search problems. In the last few decades, different types of swarm intelligence and nature-inspired techniques (Beni and Wang 1993; Li and Liu 2011; Yu and Gen 2010; Yoshida 2010; Lasisi et al. 2019; Eiben and Smith 2015; Xie, Zhou, and Chen 2013; Khanum et al. 2018; Mashwani and Salhi 2012; Mashwani et al. 2017) were developed and are still developing to cope with various unconstrained and constrained optimization problems.

Evolutionary algorithms (EAs) have many characteristics including the population-based collective learning process, self-adaptation and robustness as compared to other global optimization techniques (Zhu and Kwong 2010; Shah et al. 2018; Patwal, Narang, and Garg 2018; Garg 2019). ant colony optimization (ACO) (Kim et al. 2014; Fang et al. 2015), practical swarm optimization (PSO) (Eberhart and Kennedy 1995), firefly algorithms (Yang 2010a), plant propagation algorithms (Nag 2017; Sulaiman et al. 2014), strawberry algorithm (SBA) (Bayat 2014; Sulaiman et al. 2014), plant intelligence-based EAs (Akyol and Alatas 2017) and differential evolution (DE) (Mallipeddi et al. 2011; Mallipeddi and Suganthan 2009) are mostly recently developed EAs and they have efficiently tackled a variety of benchmark functions (Suganthan et al. 2005; Awad et al. 2016) and real-world problems (Yoshida 2010). In Bayat (2014), Sulaiman et al. (2014) and Merrikh-Bayat (2015), plants like strawberry develop both runners and roots to propagate and search for water resources and minerals. Runners and roots of the strawberry plant perform both the local and global searches simultaneously. As discussed in

[1] https://www.britannica.com/science/optimization

Bayat (2014) and Merrikh-Bayat (2015), the agents in SBA do not communicate with each other and duplication-elimination procedure motivates their agents converging toward the global best solution.

In the field of evolutionary computation (Eiben and Smith 2015), hybrid EAs have got popularity due to their capabilities and effective treating with several real-world problems taking into account complexity, noisy environment, imprecision, uncertainty and vagueness (Grosan and Abraham 2007). In this article, the idea of different population differences used in DE (Storn and Price 1997) and propagation behavior of runners and roots strawberry plant (Nag 2017; Sulaiman et al. 2014; Bayat 2014; Merrikh-Bayat 2015) are employed for population evolution and yielding to a hybrid differential evolutionary strawberry algorithm (HDEA) is developed. The algorithmic performance of the suggested HDEA is tried upon 20 benchmark functions with real parameters. The suggested algorithm is much effective and has provided promising optimal solutions for most used test problems. The numerical results provided by the proposed algorithm indicate their effectiveness and strength for dealing with non linear numerical optimization problems.

The rest of the article organized as follows: Section 2 introduces the framework of the proposed hybrid strawberry differential EA and Section 3 demonstrates the experimental results and characteristics of the used benchmark functions. Section 4 finally concludes this article.

2. Hybrid differential evolutionary strawberry algorithm

Traditional optimization techniques (Miller 1999) are unable to deal with non linear and large-scale optimization problems. EAs (Bäck 1996; Eiben and Smith 2015; Mallipeddi and Suganthan 2009) are in general categorized into nine different groups including biology-based (Sulaiman et al. 2014; Akyol and Alatas 2017), physics-based (Siddique and Adeli 2016; Hong et al. 2019), social-based (Farahlina Johari et al. 2013; Yang 2010b; Li and Liu 2011; Yu and Gen 2010), music-based (Jeong and Ahn 2015), chemical-based (Silva, Silva, and Belchior 2019), sport-based (While and Kendall 2014), mathematics-based (Mühlenbein and Mahnig 2002), swarm-based (Eberhart and Kennedy 1995; Shi and Eberhart 1998; Parsopoulos and Vrahatis 2002; Blum 2005; Pham et al. 2005; Yang 2014; Rohan et al. 2017) and hybrid methods (Grosan and Abraham 2007; Qian et al. 2018; Khan 2012; Mashwani 2011a, 2011b, 2013; Mashwani and Salhi 2012, 2014). Among them, hybrid EAs have shown great success in the recent past due to their capabilities in handling several real-world problems involving complexity, noisy environment, imprecision, uncertainty and vagueness. This article presents a HDEA that empolys at same time the recently developed SBA (Bayat 2014; Merrikh-Bayat 2015) and DE (Storn and Price 1997) algorithms. Strawberry plant can model in an effective manner based on three facts including the way strawberry plant propagating by using their runners which rise randomly to perform their global search for resources; each strawberry parent plant develops its roots and root hairs randomly in order to carry on local search process for resources and finally the strawberry offspring plants have access to richer resources that grow faster and generate more runners and roots.

2.1. Differential evolution

DE is a well-known population-based EA that was first introduced by Rainer Storn and Kenneth Price for solving Chebychev polynomial fitting problems (Storn and Price 1997). DE is a stochastic direct search method using population or multiple search points. DE has been successfully applied to the optimization problems including non linear, non differentiable, non convex and multi-modal functions and it perturbs the population by using the idea of difference of different population to perform their search process. To improve the convergence of DE over-complicated constrained and non linear (unconstrained) optimization problems, mathematicians introduced the adaptation schemes in the framework of DE (Mashwani 2014). The important operators of DE are mutation, crossover and selection to generate and select solutions for its next generation of population evolution, while the parameters of DE are NP (population size), F_m (mutation factor) and Cr (crossover ratio). The process to maintain genetic diversity from one generation to the other is called mutation. In each generation of DE, a mutant vector, $\mathbf{v}_{i,g}$ for each individual of the current population, $\{\mathbf{x}_{i,g} | i = 1, 2, \ldots, N\}$ is designed by using, one of the following strategies, which are frequently used in literature:

1. DE/rand/1:

$$\mathbf{v}^{i,t} = \mathbf{x}^{r_1,t} + F \times (\mathbf{x}^{r_2,t} - \mathbf{x}^{r_3,t})$$

2. "DE/rand/1" mutates a random solution with a difference vector.
   DE/best/1:

$$\mathbf{v}^{i,t} = \mathbf{x}^{best,t} + F \times (\mathbf{x}^{r_1,t} - \mathbf{x}^{r_2,t})$$

   "DE/best/1" mutates a best solution with a difference vector,
3. DE/rand-to-best/1:

$$\mathbf{v}^{i,t} = \mathbf{x}^{r_1,t} + F \times (\mathbf{x}^{r_2,t} - \mathbf{x}^{r_3,t}) + F \times (\mathbf{x}^{best,t} - \mathbf{x}^{r_1,t})$$

   "DE/rand-to best/1" mutates a random solution with difference vector of random solution and a best solution.
4. DE/current-to-best/1:

$$\mathbf{v}^{i,t} = \mathbf{x}^{i,t} + F \times (\mathbf{x}^{r_1,t} - \mathbf{x}^{r_2,t}) + F \times (\mathbf{x}^{best,t} - \mathbf{x}^{i,t})$$

"DE/current-to best/1" mutates a current solution with difference vector of random solution and a best solution. In the above equations, $x^{r_1} \neq x^{r_2} \neq x^{r_3}$ are randomly chosen individuals belonging to the set of solutions called population.

The mutation strategies as given above are employing three chosen solution vectors to perturb the target vector, where the differences do mimic the gradient descent behavior for guiding the search toward better solutions. These mutation strategies are much robust, stable and highly competitive and have shown strong ability to cope with exploration versus exploitation dilemma while solving scalable and multi-modal optimization problems. The design and algorithmic structure of the suggested algorithm explained here within Algorithm 1. The exploration and exploitation are two major issues for baseline EA. Exploration refers to search the specific region of the search space.

Exploitation is the process to search some areas of land or resources that are more profitable or productive or useful. In the proposed algorithm, DE has been used to explore the best spot for the survival of better offspring solution during the evolution of population and whole course of optimization.

Algorithm 1. The framework of the Hybrid Differential Evolutionary Strawberry Algorithm

```
Define parameters: N, n, M_it, x_l, x_u, d_rr, d_rt, α = 0.5;
Generate N parent solutions: x = x_l^i + (x_u^i − x_l^i) × rand(N, n), i = 1, 2, ..., N
Evaluate objective function values of parent solutions: f(x^i), i = 1, 2, ..., N
Initialize the best function value: f_best = 1e6; x_best = ones(1, n)
for k ← 1 : M_it do
    if mod(k, 5) == 0 then
        r_1^i = randperm(N), r_2^i = randperm(N), r_3^i = randperm(N);
        y = x + d_rr × (rand(n, N) − α) × x(r_1, :) + F_m × [(x(r_2, :) − x(r_1, :))];
    else
        y = [x + d_rr × (rand(n, N) − α) × x + d_rt × (rand(n, N) − α)];
    end if
    Evaluate N offspring solutions y, f(y^i), i = 1, 2, ..., N
    [f(y), I] = sort[f(y)] % sort the objective function values and offspring solutions.
    for j ← 1 : N/2 do
        x(j : , ) = y(I, :);
    end for
    for j ← 1 : 2 × N do
        if f(j) > 0 then
            φ(j) = 1/(β + f(j));
        else
            φ(j) = 1/β + |f(j)|;
        end if
    end for
    for j ← N/2 + 1 : N do
        i_w = f_w(φ(j));
        x(j, :) = y(i_w, :);
    end for
    if min(f) < f_b then
        Output: x_best = [x_1, x_2, ..., x_n] and f_best = min(f);
    end if
end for
```

3. Discussion on experimental results

In essence, optimization functions are also called artificial landscapes having had different characteristics. They are quite useful for carrying out experiments and evaluating the performance of the particular EAs in terms of convergence rate, precision, robustness. Here some test functions are presented with the aim of giving an idea about the different situations that optimization algorithms have to face when coping with these kinds of problems. The Matlab codes and detailed information of the used benchmark functions in our carried out experiments can be found at the link: https://www.mathworks.com/matlabcentral/fileexchange/23147-test-functions-for-global-optimization-algorithms.

The Benchmark functions, namely, f_1: Himmelblan Function, f_2: Rastrigin Function-1, f_3: Rastrigin Function-2, f_4: Rosenbrock Function, f_5: Griewank Function,f_6: Schaffer1 Function,f_7: Schaffer 3 Function, f_8: Sine-Valley Function, f_9: Powell Function, f_{10}: Sphere Function, f_{11}: Haupt F-1 Function, f_{12}: Haupt F-2 Function, f_{13}: Bukin4 Function, f_{14}: Beale's 3 Function, f_{15}: Booth's Function,f_{16}: Helical's Valley Function, f_{17}: Three Hump Camel Function, f_{18}: Level 3 function, f_{19}: Sum of Difference Function,f_{20}: Matyas Function were employed for the purposes to evaluate the performance of the proposed hybrid strawberry evolutionary algorithm (HDEA) in comparison with baseline SBA with same parameter settings and PC configuration as described under:

3.1. PC configuration and platform for the proposed hybrid evolutionary algorithm

- Operating system: Windows XP Professional
- Programing language of the algorithms: Matlab
- CPU: Core 2 Quad 2.4 GHz
- RAM: 4 GB DDR2 1066 MHz
- 25 independent runs were performed to solve each test problem.

3.2. Parameter settings in the proposed hybrid evolutionary algorithm

The experiments were carried out with parameters settings as follow:

- $N = 50$: number of mother plants; N must be an even number;
- $n = 10$: number of decision variables;
- $M_{it} = 500$: maximum number of iterations at each run;
- $d_{rr} = 400$: length of runners;
- $d_{rt} = 10$: length of roots;
- $\beta = 0$: used in the definition of fitness function;
- $F_m = 0.5$: scaling factor of DE
- $CR = 0.1$: probability of crossover
- $\alpha = 0.3, 0.4, 0.5$, respectively.
- $\beta = 0$: adjusts the roulette wheel selection property.

A function with multiple peaks or valleys is called multi-modal function and its landscape is multi-modal. Mostly the optimization problems are comprising many complications like their multi-modal landscapes and in most cases, their derivatives may be either impossible or too computationally expensive. The used benchmark functions are mostly multi-modals including $f_1 - f_8$, f_{10}, f_{13}, $f_{16} - f_{19}$ and the rest are uni-modal functions.

All benchmark functions were optimized by executing the suggested algorithm 25 times independently. We have saved the minimum function values, average function values, standard deviations function values, median functions values and maximum

Table 1. Experimental results of the HDEA versus SBA over 20 benchmark functions with $\alpha = 0.3$ settings.

	Best optimum value		Mean values		Median values	
Problems	HDEA	SBA	HDEA	SBA	HDEA	SBA
f01	**1.916508**	131.339082	**3.764107**	177.989706	**3.673479**	184.431497
f02	**0.000247**	0.021167	**0.007197**	0.035544	0.012106	**0.048077**
f03	0.062359	**0.059809**	**0.352998**	0.415248	**0.322457**	0.524059
f04	**7.242438**	28.239518	**10.281460**	36.521586	**11.542646**	40.449541
f05	**0.000212**	0.000552	**0.014077**	0.052774	**0.014980**	67.218020
f06	**0.001172**	0.002023	0.005958	**0.002754**	0.005808	**0.005803**
f07	24.306000	24.306000	0.012169	**0.010311**	0.017290	**0.012130**
f08	0.003896	**0.002119**	0.011792	**0.003528**	0.014764	**0.003777**
f09	**0.001081**	0.009527	0.003602	**0.027239**	0.005495	**0.049728**
f10	**0.439753**	77041.797390	**2.172962**	308401.435157	**3.228927**	541616.890148
f11	**0.000005**	0.000081	**0.000841**	0.001310	**0.000970**	0.001464
f12	**1.000009**	1.000089	**1.000131**	1.000219	**1.000184**	1.000215
f13	**0.001644**	0.017275	**0.003277**	0.518707	**0.006605**	1.197443
f14	0.000077	**0.006361**	**0.000925**	0.451498	**0.001017**	0.853668
f15	**0.000105**	0.000636	**0.001197**	0.006521	**0.002042**	0.008504
f16	**0.424150**	4.083429	**1.419265**	6.559556	**1.644115**	7.674123
f17	**0.000031**	0.000197	**0.001090**	0.000971	**0.001553**	0.001168
f18	**0.000559**	0.000799	**0.003345**	0.003075	**0.005556**	0.010322
f19	**0.196780**	0.673984	**0.412791**	1.168772	**0.391195**	1.172944
f20	**0.285340**	5.912568	**0.747929**	10.691604	**0.694130**	11.504513

Bold values represent better approximated results as compared to the other values.

Table 2. Experimental results of the MSBA versus SBA for the 20 benchmark functions with $\alpha = 0.4$ settings.

	Best optimum value		Mean values		Median values	
Problems	HDEA	SBA	HDEA	SBA	HDEA	SBA
f01	**1.957371**	40.714728	**3.164092**	44.062940	**3.235599**	45.676136
f02	**0.001044**	0.003140	**0.003249**	0.046030	**0.006297**	0.034004
f03	**0.014202**	0.028694	**0.263798**	0.409307	**0.307297**	0.403738
f04	**2.042091**	9.527664	**7.852659**	12.286566	**8.102214**	13.881178
f05	**0.000683**	0.004013	**0.005653**	0.020932	**0.007191**	0.014502
f06	0.001205	**0.000710**	**0.002744**	0.004183	**0.003734**	0.004958
f07	24.306000	24.306000	0.013923	**0.000561**	0.012541	**0.003327**
f08	**0.001641**	0.001801	0.007359	**0.002693**	0.009289	**0.003167**
f09	**0.000132**	0.000347	**0.001648**	0.006031	**0.002000**	0.004891
f10	**0.570350**	5.134292	**2.519990**	20.908284	**3.366124**	32065.856704
f11	**0.000022**	0.000340	**0.001016**	0.001110	**0.000876**	0.001898
f12	**1.000003**	1.000046	**1.000084**	1.0000162	**1.000215**	1.0000239
f13	**0.000350**	0.002556	**0.005066**	0.019510	**0.006795**	0.104392
f14	0.000311	**0.000094**	0.001813	**0.001606**	**0.002595**	0.106553
f15	**0.000089**	0.000539	**0.000978**	0.001120	**0.001219**	0.001996
f16	**0.055241**	1.084988	**0.544932**	1.199816	**0.530986**	1.324854
f17	0.000143	**0.000028**	**0.000773**	0.001794	**0.000880**	0.002703
f18	**0.000249**	0.002825	**0.004808**	0.008333	**0.006880**	0.013790
f19	**0.082985**	0.170129	**0.278843**	0.343515	**0.283016**	0.354169
f20	**0.143550**	0.616229	**0.373909**	1.604822	**0.410282**	1.547100

Bold values represent better approximated results as compared to the other values.

function values of each benchmark function independently with 25 random seeds. The numerical results of the proposed HDEA versus SBA as summarized in Tables 1–3 were calculated in terms of best, average, standard deviation, median and maximum function

Table 3. Experimental results of the HDEA versus SBA for the 20 benchmark functions with $\alpha = 0.5$ settings.

	Best optimum value		Mean values		Median values	
Problems	HDEA	SBA	HDEA	SBA	HDEA	SBA
f01	**0.013254**	0.068806	**0.041984**	0.086804	**0.039113**	0.083749
f02	0.000448	**0.000118**	0.005625	**0.000875**	0.009081	**0.003745**
f03	**0.003707**	0.044660	0.171438	**0.077780**	**0.277801**	0.388042
f04	4.976185	**6.315056**	**8.237157**	9.615790	**8.071841**	9.426563
f05	**0.001144**	0.006997	**0.008526**	0.008894	**0.008720**	0.010107
f06	**0.000559**	0.001182	**0.007431**	0.008894	0.006355	**0.004545**
f07	24.306000	24.306000	0.011741	**0.000369**	0.015245	**0.000826**
f08	0.001617	**0.001605**	0.007713	**0.001714**	0.007316	**0.001784**
f09	**0.000044**	0.000395	0.003459	**0.002102**	0.004043	**0.002668**
f10	**0.480470**	2.835595	**1.758164**	6.622362	**1.758549**	15.225471
f11	**0.000027**	0.000045	**0.000361**	0.001086	**0.000568**	0.001559
f12	**1.000010**	1.000015	**1.000131**	1.000078	**1.000219**	1.000086
f13	**0.000233**	0.001655	**0.005502**	0.004091	**0.005836**	0.014079
f14	0.000313	**0.000222**	**0.001267**	0.001610	**0.001429**	0.002025
f15	**0.000249**	0.000622	**0.002441**	0.004245	**0.002130**	0.004518
f16	**0.198596**	0.610722	**0.823828**	1.189735	**0.986613**	1.330663
f17	**0.000230**	0.000514	**0.000835**	0.001100	**0.000957**	0.001624
f18	**0.000573**	0.003222	**0.003824**	0.012058	**0.005325**	0.013810
f19	**0.099552**	0.263226	**0.289528**	0.333424	**0.246384**	0.324872
f20	**0.100253**	0.596826	**0.326194**	0.954119	**0.338526**	1.156325

Bold values represent better approximated results as compared to the other values.

values by using min, mean, std, median and max built-in functions of the Matlab environment. The experimental results in Table 1 were found by setting $\alpha = 0.3$, Table 2 represents experimental results by settling $\alpha = 0.4$ in the framework of the suggested hybrid algorithm as outlined in Algorithm 1. Similarly, the numerical results found by the hybrid strawberry differential EA with $\alpha = 0.5$ are compared with existing baseline SBA (Bayat 2014; Merrikh-Bayat 2015). The suggested hybrid algorithm has found the best approximate solutions while solving almost all test problems more efficiently as compared to the baseline SBAs. The comparison results as summarized in Tables 1–3 clearly indicate that the proposed algorithm has shown good results and outperformed SBA (Bayat 2014) over most benchmark functions.

Figures 1–3 display the convergence evolution in maximum, average and minimum objective function values of the benchmark functions applied in carried out experiments. These convergence graphs were depicted by settling $\alpha = 0.3, 0.4, 0.5,$ respectively. These three panels of figures demonstrate the capability and creditability of the proposed HDEA while converging toward the optimal solution of each respective benchmark function. As seen in figures, our proposed HDEA avoids the occurrence of premature phenomena during the solution process.

4. Conclusion

Artificial landscapes are quite useful in assessing the best qualities and weakness of the particular optimization algorithms keeping in view the convergence rate, precision, robustness and other general behaviors. Due to the rapid active commotion of EAs in the recent past few years, their performance is trailed upon wide range of optimization problems in the engineering, marketing, operations research and social sciences. The

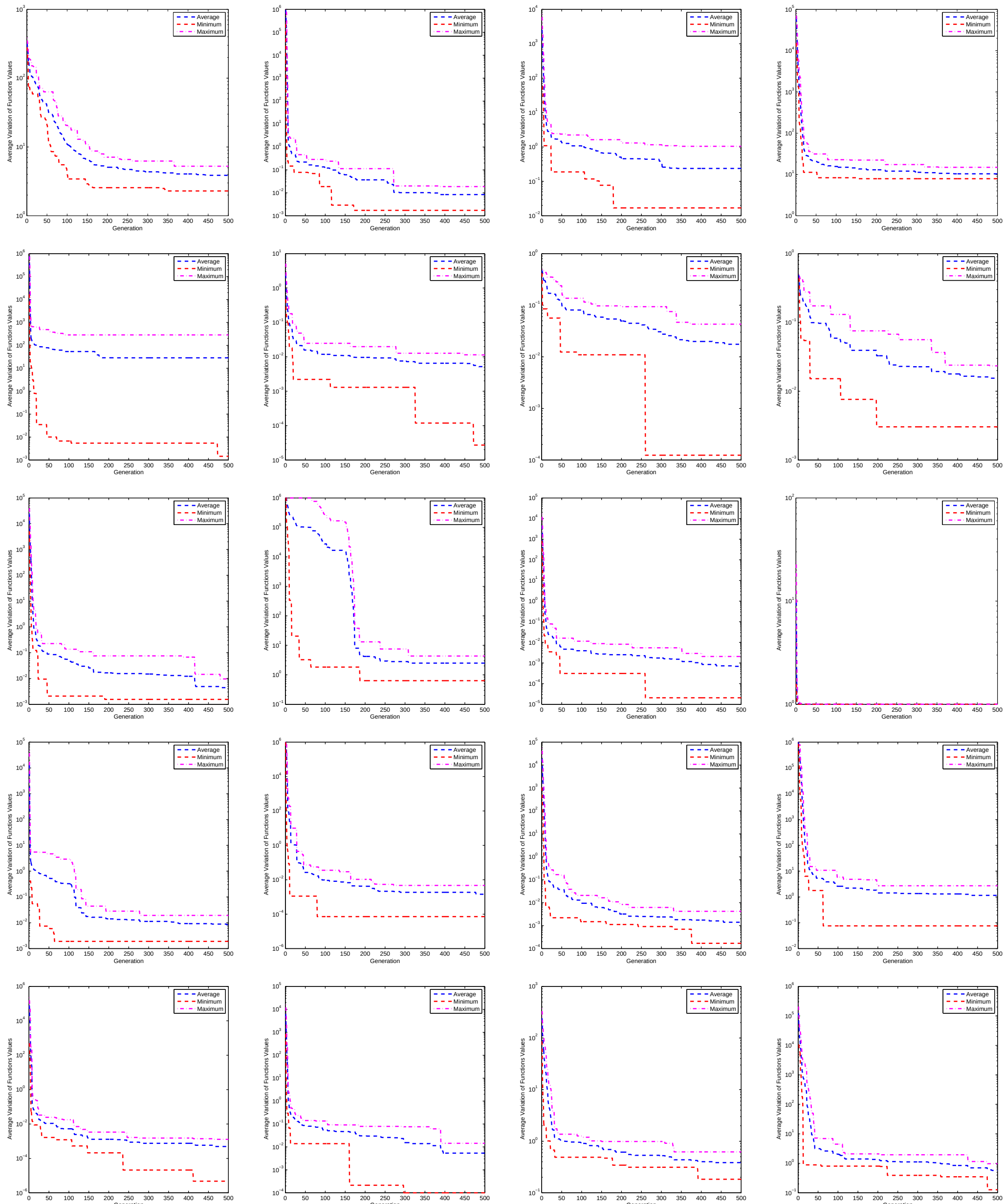

Figure 1. The evolution in minimum, average and maximum function values display by HDEA with $\alpha = 0.3$ for each test problem.

existing literature of evolutionary computing comprises of diverse test suites of unconstrained and constrained problems. In this regard, IEEE conference of evolutionary computation series furnishes every year a test suite of benchmark functions for competition of newly developed EAs. In this article, we have chosen 20 different unconstrained test functions in order to examine the performance of our suggested hybrid EA. Most of the tested functions are multi-modal optimization problems with more than one

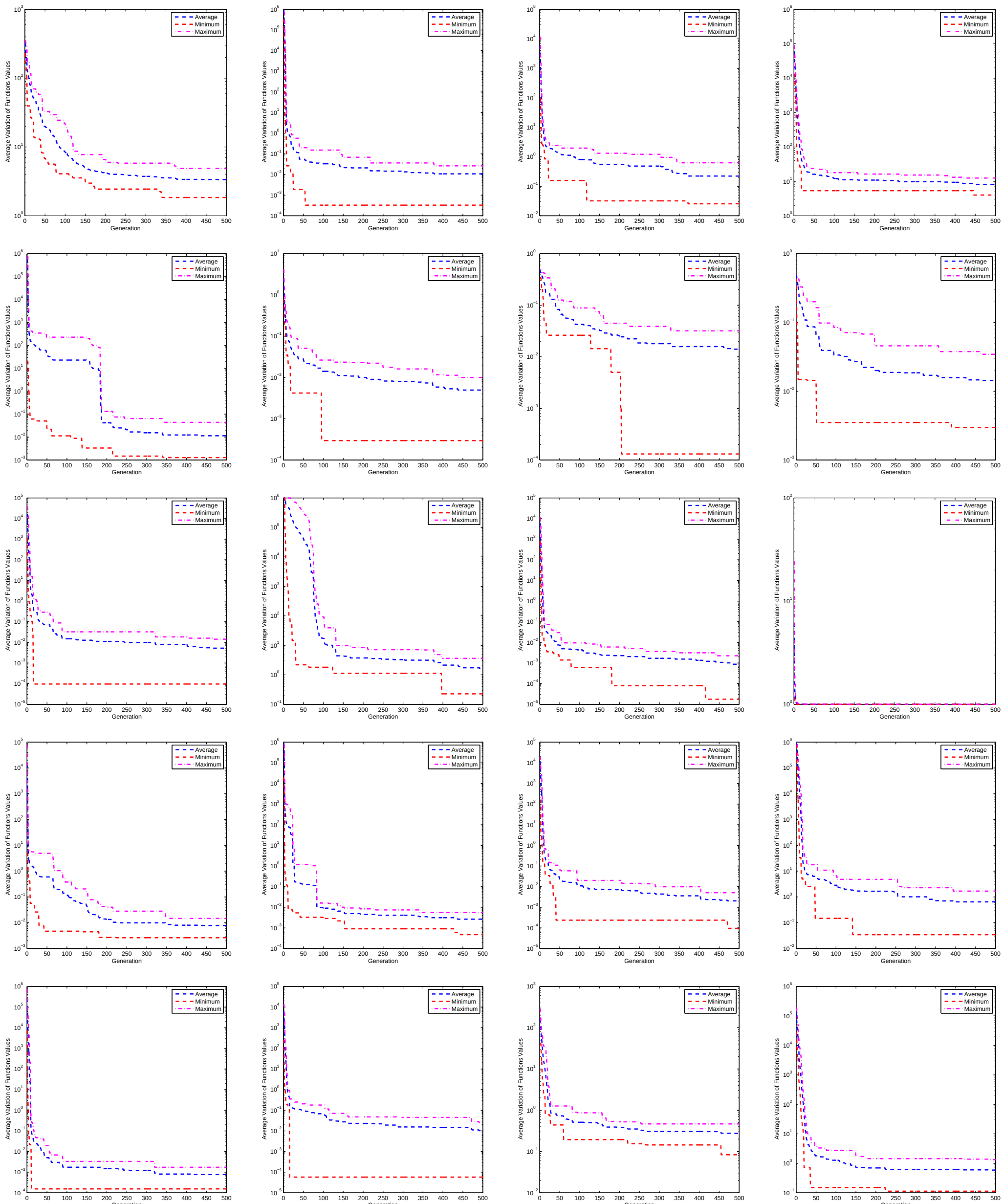

Figure 2. The evolution in minimum, average and maximum function values display by HDEA with $\alpha = 0.4$ for each test problem.

global and local solution. Multi-modal problems are difficult to deal with as compared to the uni-modal problems.

In future, we intend to analyze the intrinsic ss of the suggested algorithm to judge their search ability and credibility in order to build trust of the evolutionary computing communities over this new addition to nature-inspired algorithm paradigm. We also

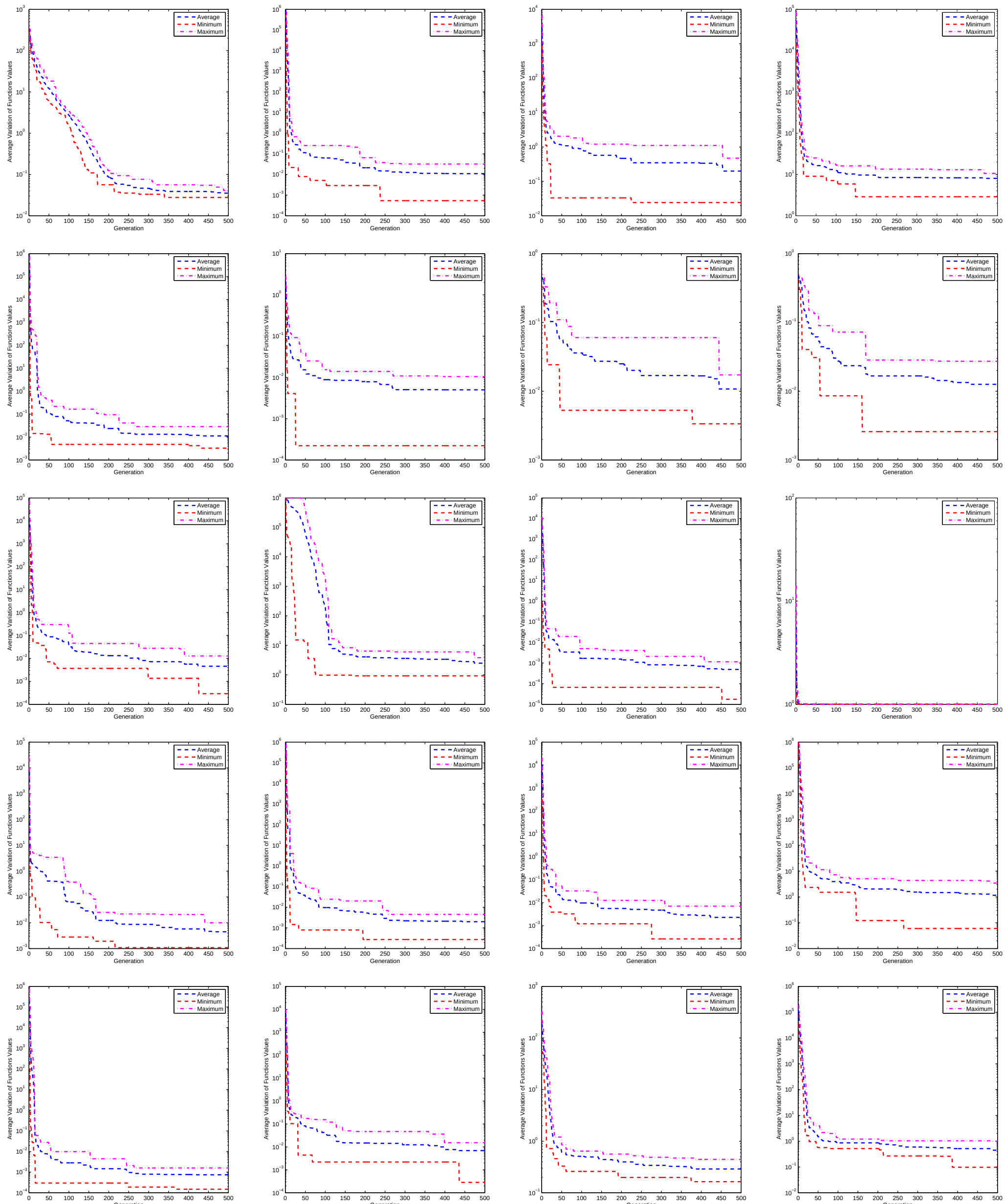

Figure 3. The evolution in minimum, average and maximum function values display by HDEA with $\alpha = 0.5$ for each test problem.

intend to verify and analyze the performance of the proposed algorithm by employing the mutation strategies in combination with nature-inspired algorithm to conduct the numerical experiments using CEC2014 and CEC2013 and CEC2017 Benchmark functions (Awad et al. 2016) http://www.ntu.edu.sg/home/EPNSugan/index-files/CEC2017/CEC2017.htm to establish a fairly comparison with state-of-the-art EAs as developed recently in the field of the evolutionary computation.

ORCID

Wali Khan Mashwani http://orcid.org/0000-0002-5081-741X

References

Akyol, S., and B. Alatas. 2017. Plant intelligence based metaheuristic optimization algorithms. *Artificial Intelligence Review* 47 (4):417–62. doi:10.1007/s10462-016-9486-6.

Awad, N. H., M. Z. Ali, J. J. Liang, B. Y. Qu, and P. N. Suganthan. 2016. Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective bound constrained real-parameter numerical optimization. Technical Report, Nanyang Technological University, Singapore.

Bäck, T. 1996. *Evolutionary* algorithms in theory and practice*: Evolution* strategies, evolutionary programming, genetic algorithms. New York, NY: Oxford University Press.

Bayat, F. M. 2014. A numerical optimization algorithm inspired by the strawberry plant. arXiv preprint arXiv:1407.7399, pp. 10–36.

Beni, G., and J. Wang. 1993. Swarm intelligence in cellular robotic systems. In *Robots and biological systems: Towards a new bionics*, ed. P. Dario, G. Sandini, and P. Aebischer, 703–12. Berlin/Heidelberg, Germany: Springer.

Blum, C. 2005. Ant colony optimization: Introduction and recent trends. *Physics of Life Reviews* 2 (4):353–73. doi:10.1016/j.plrev.2005.10.001.

Eberhart, R., and J. Kennedy. 1995. A new optimizer using particle swarm theory. Paper presented at the Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan, October 4–6, 39–43.

Eiben, A. E., and J. E. Smith. 2015. *Introduction to evolutionary computing*. 2nd ed. Berlin, Heidelberg: Springer-Verlag.

Fang, W., X. Li, M. Zhang, and M. Hu. 2015. Nature-inspired algorithms for real-world optimization problems. *Journal of Applied Mathematics* 2015:1–2. doi:10.1155/2015/359203.

Farahlina Johari, N., A. Zain, N. Mustaffa, and A. Udin. 2013. Firefly algorithm for optimization problem. *Applied Mechanics and Materials* 421:512–7. doi:10.4028/www.scientific.net/AMM.421.512.

Fiacco, A. V., and G. P. McCormick. 1968. *Nonlinear* programming*: Sequential* unconstrained minimization techniques. New York, NY: John Wiley & Sons. (Reprinted by SIAM Publications in 1990.)

Garg, H. 2019. A hybrid GSA-GA algorithm for constrained optimization problems. *Information Sciences* 478:499–523. doi:10.1016/j.ins.2018.11.041.

Grosan, C., and A. Abraham. 2007. Hybrid evolutionary algorithms: Methodologies, architectures, and reviews. In *Hybrid evolutionary algorithms*, ed. A. Abraham, C. Grosan, and H. Ishibuchi, 1–17. Berlin/Heidelberg, Germany: Springer.

Hong, S., D. Han, K. Cho, J. S. Shin, and J. Noh. 2019. Physics-based full-body soccer motion control for dribbling and shooting. *ACM Transactions on Graphics* 38 (4):1–12. doi:10.1145/3306346.3322963.

Jeong, J. H., and C. W. Ahn. 2015. Automatic evolutionary music composition based on multi-objective genetic algorithm. In *Proceedings of the 18th Asia Pacific Symposium on Intelligent and Evolutionary Systems – Volume* 2, ed. H. Handa, H. Ishibuchi, Y.-S. Ong, and K.-C. Tan, 105–15. Cham, Switzerland: Springer.

Khan, W. 2012. Hybrid multiobjective evolutionary algorithm based on decomposition. PhD diss., University of Essex, Colchester, UK.

Khanum, R. A., M. A. Jan, W. K. Mashwani, N. M. Tairan, H. U. Khan, and H. Shah. 2018. On the hybridization of global and local search methods. *Journal of Intelligent & Fuzzy Systems* 35 (3):3451–64. doi:10.3233/JIFS-17657.

Kim, J., T. Sharma, B. Kumar, G. Tomar, K. Berry, and W. Hyung. 2014. Research article: Intercluster ant colony optimization algorithm for wireless sensor network in dense environment. *International Journal of Distributed Sensor Networks* 10 (4):457402. doi:10.1155/2014/457402.

Lasisi, A., N. Tairan, R. Ghazali, W. K. Mashwani, S. N. Qasem, H. Garg, et al. 2019. Predicting crude oil price using fuzzy rough set and bio-inspired negative selection algorithm. *IJSIR* 10 (4):25–37.

Li, L., and F. Liu. 2011. *Group search optimization for applications in structural design*. Vol. 9. Berlin, Heidelberg: Springer-Verlag.

Mallipeddi, R., and P. Suganthan. 2009. Differential evolution algorithm with ensemble of populations for global numerical optimization. *OPSEARCH* 46 (2):184–213. doi:10.1007/s12597-009-0012-3.

Mallipeddi, R., P. Suganthan, Q. Pan, and M. Tasgetiren. 2011. Differential evolution algorithm with ensemble of parameters and mutation strategies. *Applied Soft Computing* 11 (2):1679–96. doi:10.1016/j.asoc.2010.04.024.

Mashwani, W. K. 2011a. Hybrid multiobjective evolutionary algorithms: A survey of the state-of-the-art. *International Journal of Computer Science Issues* 8 (6):374–92.

Mashwani, W. K. 2011b. MOEA/D with DE and PSO: MOEA/D-DE + PSO. Paper presented at the 31st SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence, Cambridge, UK, December 13–15, 217–21.

Mashwani, W. K. 2013. Comprehensive survey of the hybrid evolutionary algorithms. *International Journal of Applied Evolutionary Computation (IJAEC)* 4 (2):1–19.

Mashwani, W. K. 2014. Enhanced versions of differential evolution: State-of-the-art survey. *International Journal of Computing Science and Mathematics* 5 (2):107–26. doi:10.1504/IJCSM.2014.064064.

Mashwani, W. K., and A. Salhi. 2012. A decomposition-based hybrid multiobjective evolutionary algorithm with dynamic resource allocation. *Applied Soft Computing* 12 (9):2765–80. doi:10.1016/j.asoc.2012.03.067.

Mashwani, W. K., and A. Salhi. 2014. Multiobjective memetic algorithm based on decomposition. *Applied Soft Computing* 21:221–43. doi:10.1016/j.asoc.2014.03.007.

Mashwani, W. K., A. Hamdi, M. Asif Jan, A. Göktaş, and F. Khan. 2020. Large-scale global optimization based on hybrid swarm intelligence algorithm. *Journal of Intelligent & Fuzzy Systems* 38 (6):1–19. doi:10.3233/JIFS-192162.

Mashwani, W. K., A. Salhi, O. Yeniay, H. Hussian, and M. A. Jan. 2017. Hybrid non-dominated sorting genetic algorithm with adaptive operators selection. *Applied Soft Computing* 56:1–18. doi:10.1016/j.asoc.2017.01.056.

Mashwani, W. K., A. Zaib, O. Yeniay, H. Shah, N. Tairan, and M. Sulaiman. 2019. Hybrid constrained evolutionary algorithm for numerical optimization problems. *Hacettepe Journal of Mathematics and Statistics* 48 (3):931–50. doi:10.15672/HJMS.2018.625.

Merrikh-Bayat, F. 2015. The runner-root algorithm: A metaheuristic for solving unimodal and multimodal optimization problems inspired by runners and roots of plants in nature. *Applied Soft Computing* 33:292–303. doi:10.1016/j.asoc.2015.04.048.

Miller, R. E. 1999. *Optimization: Foundations and applications*. New York, NY: John Wiley & Sons.

Mühlenbein, H., and T. Mahnig. 2002. Mathematical analysis of evolutionary algorithms. In *Essays and surveys in metaheuristics*, ed. C. C. Ribeiro and P. Hansen, 525–56. Boston, MA: Springer US.

Nag, S. 2017. Adaptive plant propagation algorithm for solving economic load dispatch problem. CoRR abs/1708.07040.

Parsopoulos, K. E., and M. N. Vrahatis. 2002. Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing* 1 (2/3):235–306. doi:10.1023/A:1016568309421.

Patwal, R. S., N. Narang, and H. Garg. 2018. A novel TVAC-PSO based mutation strategies algorithm for generation scheduling of pumped storage hydrothermal system incorporating solar units. *Energy* 142:822–37. doi:10.1016/j.energy.2017.10.052.

Pham, D., A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim, and M. Zaidi. 2005. The bees algorithm. Technical Note, Manufacturing Engineering Centre, Cardiff University, Cardiff, UK.
Qian, X., X. Wang, Y. Su, and L. He. 2018. An effective hybrid evolutionary algorithm for solving the numerical optimization problems. *Journal of Physics: Conference Series* 1004:012020. doi:10.1088/1742-6596/1004/1/012020.
Rohan, R., C. N. Prasad, J. Jose, and P. Sadiq. 2017. An introduction to the collective behaviour of swarm intelligence. National Conference on Contemporary Research and Innovations in Computer Science, St. Joseph's Evening College, Bangalore, India.
Ruszczynski, A. 2006. *Nonlinear optimization*. Princeton, NJ: Princeton University Press.
Shah, H., N. Tairan, H. Garg, and R. Ghazali. 2018. Global gbest guided-artificial bee colony algorithm for numerical function optimization. *Computer Magazine.* 7 (4):69–17. doi:10.3390/computers7040069.
Shi, Y., and R. Eberhart. 1998. A modified particle swarm optimizer. Paper presented at the IEEE International Conference on Evolutionary Computation Proceedings (Cat. No. 98TH8360), Anchorage, AK, USA, May 4–9, 69–73. doi:10.1109/ICEC.1998.699146.
Siddique, N., and H. Adeli. 2016. Physics-based search and optimization: Inspirations from nature. *Expert Systems* 33 (6):607–23. doi:10.1111/exsy.12185.
Silva, F. T., M. X. Silva, and J. C. Belchior. 2019. A new genetic algorithm approach applied to atomic and molecular cluster studies. *Frontiers in Chemistry* 7:707. doi:10.3389/fchem.2019.00707.
Storn, R., and K. Price. 1997. Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization* 11 (4):341–59.
Suganthan, P. N., N. Hansen, J. J. Liang, K. Deb, Y. P. Chen, and A. Auger. 2005. Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. Technical Report, Nanyang Technological University, Singapore.
Sulaiman, M., A. Salhi, B. I. Selamoglu, and O. B. Kirikchi. 2014. A plant propagation algorithm for constrained engineering optimisation problems. *Mathematical Problems in Engineering* 2014:1–10. doi:10.1155/2014/627416.
While, L., and G. Kendall. 2014. Scheduling the English football league with a multi-objective evolutionary algorithm. In *Parallel Problem Solving from Nature – PPSN XIII*, ed. T. Bartz-Beielstein, J. Branke, B. Filipič, and J. Smith, 842–51. Cham, Switzerland: Springer.
Xie, J., Y. Zhou, and H. Chen. 2013. A novel bat algorithm based on differential operator and lévy flights trajectory. *Computational Intelligence and Neuroscience* 2013:1–13. doi:10.1155/2013/453812.
Yang, X.-S. 2010a. *Nature-inspired meta-heuristic algorithms*. Beckington, UK: Luniver Press.
Yang, X.-S. 2010b. A new metaheuristic bat-inspired algorithm. In *Nature inspired cooperative strategies for optimization (NICSO 2010)*, ed. J. R. González, D. A. Pelta, C. Cruz, G. Terrazas, and N. Krasnogor, 65–74. Berlin, Germany: Springer.
Yang, X.-S. 2014. Swarm intelligence based algorithms: A critical analysis. *Evolutionary Intelligence* 7 (1):17–28. doi:10.1007/s12065-013-0102-2.
Yoshida, Z. 2010. *Nonlinear science: The challenge of complex systems*. London: Springer-Verlag.
Yu, X., and M. Gen. 2010. *Introduction to evolutionary algorithms*. Springer Science & Business Media.
Zhu, G., and S. Kwong. 2010. Gbest-guided artificial bee colony algorithm for numerical function optimization. *Applied Mathematics and Computation* 217 (7):3166–73. doi:10.1016/j.amc.2010.08.049.