Applied Soft Computing Journal 90 (2020) 106176

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

A new metaheuristic optimization algorithm inspired by human dynasties with an application to the wind turbine micrositing problem ☆

Shafiq-ur-Rehman Massan [a,b,*], Asim Imdad Wagan [b], Muhammad Mujtaba Shaikh [c,d]

[a] Shaheed Zulfikar Ali Bhutto Institute of Science and Technology, Sindh, Pakistan
[b] Mohammad Ali Jinnah University, Karachi, Sindh, Pakistan
[c] Department of Basic Sciences and Related Studies, Mehran University of Engineering and Technology, Jamshoro, Pakistan
[d] Supply Chain and Operations Management Research Group, Mehran University of Engineering and Technology, Jamshoro, Pakistan

ARTICLE INFO

Article history:
Received 28 April 2019
Received in revised form 10 February 2020
Accepted 12 February 2020
Available online 24 February 2020

Keywords:
Soft Computing
Genetic Algorithm
Differential Evolution Algorithm
Wind turbine Micrositing
Engineering optimization

ABSTRACT

Optimization is an art that is best performed by a well-tuned algorithm. Nature – instead of being fully deterministic – is evolutionary, vibrant and resourceful. The nature-inspired algorithms use the best combination and evolution strategy in a given situation. In this work, a new metaheuristic algorithm is developed by using social behavior in human dynasties. The motivation, conceptual framework, mathematical model, pseudocode and working of the algorithm are described in this paper and the adjoining papers. The proposed dynastic optimization algorithm (DOA) has evolved with the wind turbine micrositing (WTM) problem in mind. The proposed DOA has been successfully applied to the traditional WTM and encouraging results have been obtained. It is demonstrated that the proposed approach is equally viable as other existing algorithms, like the Genetic algorithm (GA) and Differential evolution algorithm (DEA). The main advantage of the proposed DOA is that it is simple, unique, fast, unbiased and versatile in comparison with others. The validation of results has been made with respect to a few other mainstream algorithms in the literature, besides statistical sensitivity analysis is also performed. The 95% confidence interval forecasts for the power enhancement and cost reduction by using DOA against GA and DEA are encouraging and guarantee an adequate amount of mean increase in power output and a considerable average cost reduction.

1. Introduction

The main target of optimization is to successfully find the maximum or the minimum of a given physical system that is defined by an objective function and is bound by some real-valued constraints. Therefore for any objective function $f(\vec{x})$, the goal is to find the best decision variable vector $\vec{x}$ in R^n so that the lower and upper bound constraints are satisfied. Mathematically, we can write:

$$\text{optimize } f(\vec{x}): \vec{x} \in R^n \tag{1}$$

where the lower bounds and upper bounds are defined as in (2),

$$\vec{l} \leq \vec{x} \leq \vec{u} \tag{2}$$

There exists a solution satisfying the constraints that are optimal for the function and may be approached by a suitable technique.

Some of the most cited algorithms in the stochastic and metaheuristic domain are the Genetic algorithm [1–5], Grey-wolf optimizer [6], Monte Carlo simulation [4], Differential Evolution Algorithm [7–10], the stochastic hill-climbing [11], Ant colony optimization algorithms [11], the Lion pride algorithm [12], the Firefly algorithm [13–16], hybrid algorithms [17], bio-inspired Salp swarm algorithm [18], etc.

These algorithms have some sort of a search strategy that is utilized to navigate the search space. The search strategy is combined with a stochastic component so as to minimize the effect of localization of the search and such methods are known as metaheuristic methods.

There are two classes of algorithms in the metaheuristic domain; either deterministic or stochastic. Thus, the deterministic algorithms are the ones that shall always give a unique and single answer given that the initial conditions and initial guess are the same. However, in the case of stochastic algorithms, the solution

☆ All authors equally contributed in this text.

* Corresponding author at: Mohammad Ali Jinnah University, Karachi, Sindh, Pakistan.

E-mail addresses: srmassan@hotmail.com (S.-u.-R. Massan), aiwagan@gmail.com (A.I. Wagan), mujtaba.shaikh@faculty.muet.edu.pk (M.M. Shaikh).

https://doi.org/10.1016/j.asoc.2020.106176
1568-4946/

may differ with every implementation even if the same initial conditions and initial guess are used [14].

The main reason for choosing a stochastic implementation is that it is less prone to local optima. A deterministic algorithm may be prone to converge at a local optima [14]. This property is evident when we compare the hill climbing and stochastic hill-climbing algorithms [11].

The means of propagation of a stochastic algorithm are as diverse as ant pheromones to genes [11]. These algorithms have two main components, the main engine which is deterministic and the stochastic component. The form and the implementation of the stochastic component may vary, it may work inside the algorithm by changing an allele randomly [8] or it may work separately as a unique solution or a random walk [19].

In the past, researchers have approached multi-objective non-linear optimization problems and solved them by using meta-heuristic techniques and are referenced as [1–6,8–14,16–18,20,21]. There also exist computational fluid dynamics techniques for the solution of the wind turbine Micrositing problem [22,23], etc.

The development of new techniques in the field of Nature-Inspired Algorithms is an interesting and challenging task. It requires analogy from nature, from different ecosystems, to emulate and simulate.

The field of optimizers is an emerging field, and most of the algorithms already available in the literature suffer from one or more of the problems mentioned in [11]. Some techniques are not completely described in the literature, and it is difficult for researchers to piece the parts together to replicate the technique. There are many inconsistencies in techniques and a variety of implementations exist. Thus, a given formulation of pseudocode may vary from one technique to another. Furthermore, new scripts and languages render the older implementations void as the new implementations, such as in Python, are quicker, shorter and less processor intensive. There is a vast difference in the parameterization, operations and data structures for the same technique in a number of papers spanning over its history. Thus, an unambiguous way of writing the technique is important for leading the researchers towards successful application, formulation, further investigation and a possible extension of the technique.

The objective of this paper is to present a unique and robust metaheuristic algorithm that is focused on solving non-linear bound-constrained optimization problems. The idea of the proposal has been a consequence of inspiration from dynasties and revolutions in social sciences. The proposed algorithm is thus named as the Dynastic optimization algorithm (DOA). The proposed algorithm has been established mathematically in a formal way with a description of its working and pseudocode. The performance of the proposed DOA has been checked along-with other existing algorithms on a well-known engineering problem of WTM. The validation of the results of DOA has been made with other approaches, and it is demonstrated that the application of DOA on the WTM problem yields a higher value of the power produced and a lower value of cost per installation per unit turbine after installing 100 turbines. The efficiency of the DOA has been discussed, and results are also encouraging.

2. Proposed algorithm

Here, the proposed optimization algorithm DOA is presented in detail. The motivation for the formulation of this algorithm was to present a simple and unbiased method of solution of optimization problems. The proposed method has a low computational cost and is comparable with other methods.

The proposal, its mathematical model, discussion and example working are discussed. Further, the basics of WTM problem and a corresponding mathematical model is described for the implementation and validation of the performance of DOA against some other algorithms in the literature.

2.1. The proposed dynastic optimization algorithm (DOA)

The DOA has been written in an unambiguous method and is versatile for use in any type of application. It would be very useful to future researchers and is a contribution of its own kind. Further, the analogy may be derived from the synthesis of any data sample in the process of optimization. In every sample, there are troughs, peaks, and clusters. Therefore, the DOA utilizes the peaks to create local optima, then the nearby or adjacent data points move towards these peaks in clusters. Whereas, a third significant portion of the data points is generated stochastically that enriches the search experience.

It should be clearly noted that this algorithm is not a variant of the PSO as it is much simpler in the application. The PSO has to calculate the three directions in search space to update the next position. In DOA, this is even simpler as the Euclidean distance is calculated to the rulers' position where all the followers move. The purpose of randomness is added by randomly moving separate actors known as explorers in the search space.

2.2. Inspiration from dynasties and revolutions

The inspiration for the proposed algorithm is derived from human nature and from the social sciences in particular. It is derived from the society that is a product of nature itself. Therefore, it may also be termed as a nature-inspired algorithm. Here is a brief introduction to the algorithm.

At the very beginning of time mankind was comprised of groups of hunter–gatherers or explorers. They scavenged the wilderness in search of food, shelter, and clothing.

With the passage of time, the concept of belongings and holding on to the belongings for extended periods of time was born. Mankind chose to live in tribes.

This expanse in society gave rise to local nobles or the tribesmen with the best clothing, from the best food gatherers. From amongst these tribesmen rose the Chieftains of the tribes.

With the passage of time, the role of the best hunter–gatherer has never been lost. This explorer has albeit changed its form from Columbus to Bill Gates. They can be very powerful indeed and leave a deep impact on society.

Next, the normal populace is the workers. These workers always gather around the nobles or rich men. They follow the lifestyle of the nobles and emulate their way of life.

The nobles choose the king at the highest strata of society. Thus, the algorithm has two methods of propagation, inspiration and stochastic.

Hence, the trio of explorers, workers, and nobles select the king. The King thrives in the dynasty and succeeds in the next generation whereas the nobles and workers change with every iteration of the population.

In the event of an explorer becoming a king, we can safely say that the new best solution has been attained. The inspiration of this algorithm may be derived from the various dynasties that ruled the Indian sub-continent.

2.3. Mathematical model of the proposed DOA

Step 1. Consider generating a random population N_p, which is then evaluated.

$$\mathrm{Np} = \{1, 2, 3 \ldots \mathrm{m}\} \tag{3}$$

$\forall m \in I$, and

$N_r \subseteq N_p$, $N_w \subseteq N_p$ and $N_e \subseteq N_p$, Where, N_r, N_w and N_e are the number of rulers/kings, workers/followers and explorers, respectively.

Step 2. Now, for any objective function,

$$F(\mathbf{x}) \text{ where } \mathbf{x} = (x_1, x_2, x_3, x_4 \ldots x_\mathrm{m})^\mathrm{T} \tag{4}$$

Step 3. The r_r percentage of the population shall be considered as the Rulers, and they shall be ranked and their positions will be fixed. Then about 60% or a random percentage of the population will be considered as workers. These workers will only perform local search around a particular ruler with a fixed radius. The rest of the remaining population will be considered as the Explorers which will randomly explore the space for optimization. The total ratio of the three percentages should add to 100%. For example, if r_r =20%, then:

$r_\mathrm{r} = 0.2$, $r_\mathrm{w} = 0.6$ and $r_\mathrm{e} = 0.2$

$N_\mathrm{r} = r_\mathrm{r} {}^* \mathrm{N_p}$ $\forall \mathrm{N_r} \in I$ and so on

Step 4. The rulers shall be ranked with each iteration, such that, Rank(N_r) = max | Np| $N_r \subseteq N_p, \forall N_r = 0.2 * N_p$ and so on

Step 5. Rulers' position shall be fixed. The workers shall perform the localized stochastic search around the rulers. The Workers shall move around the closest rulers such that they move in a fixed radius around the rulers,

$$\mathrm{rad_w} = 0.4 s_\mathrm{s} \quad \forall \mathrm{rad_w} \leq 1, \tag{5}$$

s_s is the search space.

The distance between the rulers and the workers shall be governed by the cartesian equation,

$$\text{Distance} = \sqrt{\left| \left| x_{r,i} \right|^2 - \left| x_{w,j} \right|^2 \right|} \tag{6}$$

where, r = Rulers and w = Workers and i,j $\subseteq$I

Step 6. The explorers shall be randomly generated and shall move at random positions in the unexplored space.

$$x_e = rand(x_i) \forall i \in I \tag{7}$$

All the population shall be ranked and the rulers shall again be selected. Then steps 3 to 6 shall be repeated.

Step 7. After the "n" number of iterations or reaching a minimum error, the algorithm will return the ruling class as the best solution.

$$\overrightarrow{x}_{best} = \max F(\overrightarrow{x}) \tag{8}$$

Thus, the best ruler, from the best class of rulers, which is significantly better than the workers and the explorers is selected [24].

2.4. Discussion about the DOA

The proposed DOA has a number of similarities with the GA, however, it is less complex, as it does not have the processes of crossover and mutation.

The DOA followers search for possible optimal points in the vicinity of the rulers in a given society. It introduces the concept of parallel search which has a broader scope as compared to some narrowly tuned algorithms. The DOA can handle multi-modal and volatile optimal functions in an efficient manner. It can be deduced that the DOA is a special class belonging to the swarm type of algorithms.

The DOA is much simpler than the standard particle swarm optimization algorithm [20]. The migration of the workers in the DOA is adjustable so that they may approach a given ruler from multiple directions and converge on it. Varying the number of rulers assures that the search space is sufficiently explored.

It should be noted that the distance between the rulers and the workers is not limited to their Euclidean distance. This is because other measures of distance can also be taken into account such as time lag as in job scheduling problems on the internet by the length of a connection and its associated cost.

It has been experienced that as the iterations are increased, the DOA improves its algorithmic performance. However, the right combination of the mix of rulers, workers, and explorers has to be judged for the problem under consideration. It is evident that the DOA is quick to converge on the local optima as well as on the global optima.

Another advantage of the DOA is that it can search in the vicinity of the global best solution as it approaches it from all sides. This is because the workers can move independently towards their nearest ruler. This is not possible in the implementation of the GA [1] and the PSO [20].

Another advantage of the DOA is that the king or the global best is the penultimate solution.

2.5. The wind turbine micrositing (WTM) problem

It was well quoted by Holland [25], that living organisms are some of the best problem solvers. Hence, it may be attributed to this property of living organisms that we have emulated for the solution of the WTM problem using the proposed DOA.

2.6. The mathematical model of WTM problem

This Jensen model was formulated by N.O Jensen in the year 1983 at the famous Riso National Laboratory Denmark [26]. The Jensen model is known for its simplicity and ease of deployment in mathematical calculations for turbines arranged in open fields. It is possible to emulate and simulate multiple wakes generated one behind the other with this model.

In this work, we make the same assumptions as in [9,10,16,17] such as Radius of the wind turbine rotor = 40 m, Height of the wind turbine hub = 60 m, and Thrust coefficient = 0.88. Thus, we may depict our simulation for a simple scenario as shown in Fig. 1.

In Fig. 1. the initial wind velocity u_0, is incident on a wind turbine having a rotor radius r_r. The effects of the wake can be felt at a distance of x behind the turbine. The radius of the wake effect at this distance is r_1, which is governed by the illustrated equation.

This model is based on the assumption that the total momentum is conserved within a wake. It also stipulates that this wake is spread in a linear fashion downstream from the turbine. Hence, by cascading turbines in this linear model, we may accommodate multiple wakes and their interaction. Here, we may utilize these variables,

r_r = Initial wake radius which is also the Turbine radius

r_1 = Wake radius at a distance of x

x = downstream distance where wake radius is calculated

Thus, Betz's theory gives the wind speed after the rotor as,

$$U = u_0 + \left(1 - \frac{2a}{1 + \alpha(\frac{x}{r_1})}\right) \tag{9}$$

The free stream wind speed or mean speed is denoted by U_0. Moreover, the axial induction factors, a can be found by the following equation as,

$$C_\mathrm{T} = 4a(1 - a) \tag{10}$$

The equation governing the relationship between the turbine radius r_r and the incident radius r_1 located at a distance of x is,

$$r_1 = r_r^2 \sqrt{\frac{1 - a}{(1 - 2a)}} \tag{11}$$

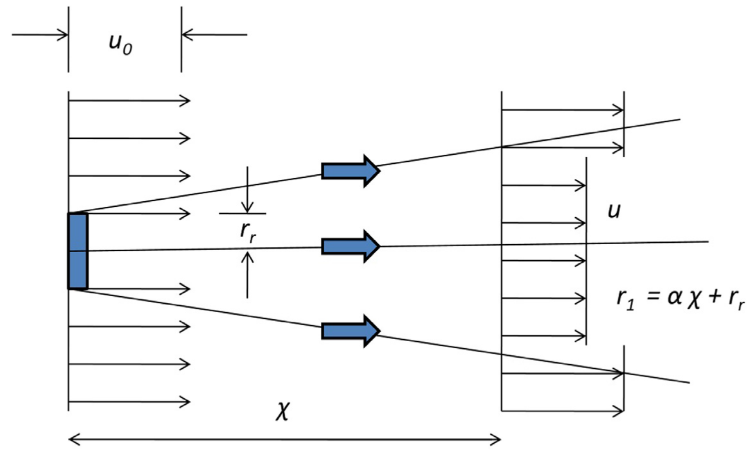


Fig. 1. The simple drawing of the Jensen model illustrating the wake at a distance behind the turbine.

Whereas, α, the entertainment constant is given by the equation,

$$\alpha = \frac{0.5}{ln\left(\frac{z}{z_0}\right)} \tag{12}$$

Where, z = Hub height
z_0 = Terrain roughness (for flat areas = 0.3)

Hence the Jensen model may be elaborated to yield the equation for wind turbines that have multiple wakes incident on them, given by,

$$u_i = u_0 + \sum_{i=1}^{N_t}\left(1 - \frac{u}{u_0}\right) \tag{13}$$

Where,
N_t = the number of Turbines in incident wake
u_0 = original wind speed,
u_i = wind speed at the ith turbine,
u is the resultant wind speed at incident wind turbine

Hence, we may reach the conclusion that the number of wind turbines and their subsequent placement is equally important.

Further, the Available Power is yielded as per the equation,

$$\text{Available Power } = \frac{1}{2}\rho A u^3 \tag{14}$$

If we multiply the efficiency of the turbine in this equation, we have,

$$\text{Available Power } = \eta\frac{1}{2}\rho A u^3 \tag{15}$$

The maximum power produced is thus,

$$\text{Power Produced } = 0.3u^3 \text{ Kilowatts} \tag{16}$$

The efficiency, η, may be reached by utilizing the aerodynamic efficiency of rotors. From the Betz limit we have,

$$\eta = \frac{\sum_{i=1}^{N_t} 0.3 \times u_i^3}{N_t(0.3 \times u_0^3)} \tag{17}$$

Hence,

$$Efficiency = \frac{Power_{Total}}{N_t\left(0.3 \times u_0^3\right)} \tag{18}$$

The cost model used in this study is dimensionless and empirical and has also been referenced in earlier literature. The cost model predicts a maximum reduction of cost by one third with the installation of every new turbine.

The cost model is as follows,

$$Cost = N_t\left(\frac{2}{3} + \frac{1 \times e^{-0.00174} \times N_t}{3}\right) \tag{19}$$

Thus, the cost reduces by one-third as every new turbine is added to the system.

Consider Fig. 2 showing the positioning of different turbines (nine as an example) with the wake effect of each turbine. The case of two turbines at the same position, for example, turbines “1” and “3” is shown in Fig. 3. In such a case, if both x-distance and y-distance between the turbines are less than 199 then the solution is discarded. Whereas, if either one of x distance or y distance is less than 199, it means that the turbines are adequately separated.

On the other hand, the velocity and power of a turbine experiencing wake can be calculated. For example, consider turbines “1” and “7” as shown in Fig. 4, where “7” is fed by the wake of “1”. The radius of the wake is given by the equation,

$$radius = r_r + \alpha Y \tag{20}$$

If xmin < x and xmax > x, then, at a given distance Y, if x lies between the values of xmin or xmax then it experiences a wake. The velocity is calculated by using initial wind speed u_0, location of the turbine. The position of both turbines is checked and the velocity ratio is calculated by first calculating the vertical distance between the two turbines, the final equation of velocity is given in (13). This velocity ratio is multiplied with the velocity to get the incident velocity at the turbine.

In the case of a cascaded turbine, like “9” with “4” and “2” in Fig. 5, the turbines are eliminated one by one as they approach the final turbine. The initial wind speed u_0 decreases with every wind turbine that it encounters.

3. Results and discussion

3.1. Numerical results and discussion of the performance of DOA for WTM problem

To check the performance of the proposed DOA for the WTM problem, the parameters used for the numerical simulation are given in Table 1 and [27]. In a square-grid of 2 km by 2 km area, an attempt has been made to install 100 turbines. The coordinates in the search region mesh are divided into 100 points for x and for y, thus in total a 100 by 100 mesh. The maximum number of iteration used to search for the optimal positioning of the wind turbines has been set at 10,000. The ratio of the kings, followers, and explorers has been set as 5%: 55%:40%, respectively. The

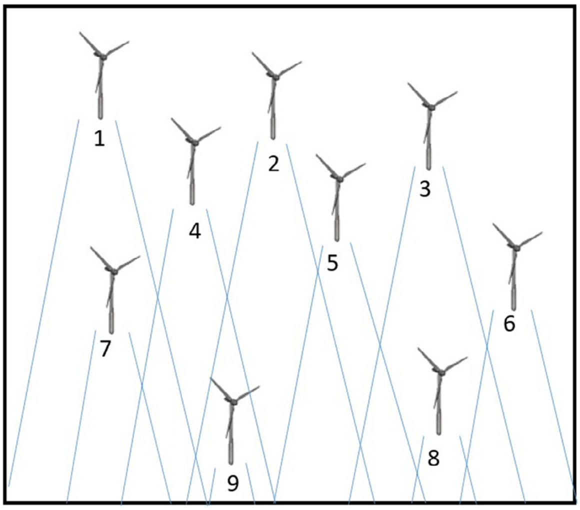


Fig. 2. Wake effect of different turbines.

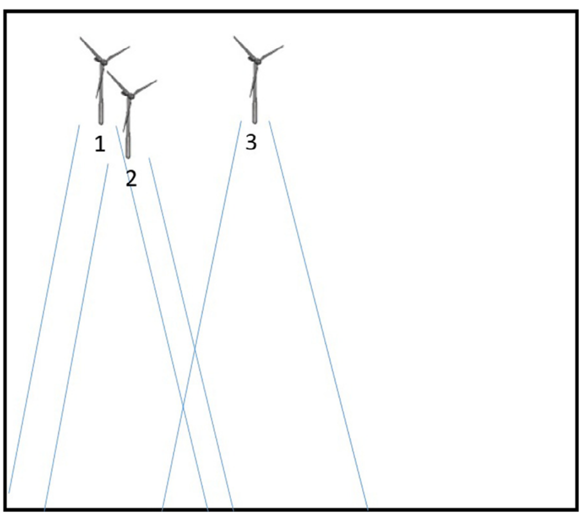


Fig. 3. Two turbines at the same position (1 and 3).

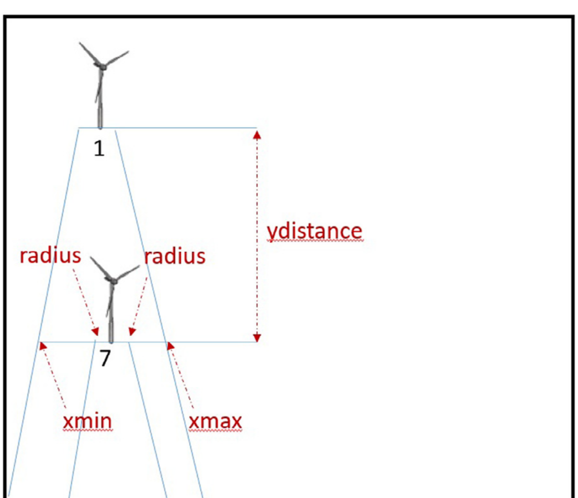


Fig. 4. Velocity and power for a turbine in wake.

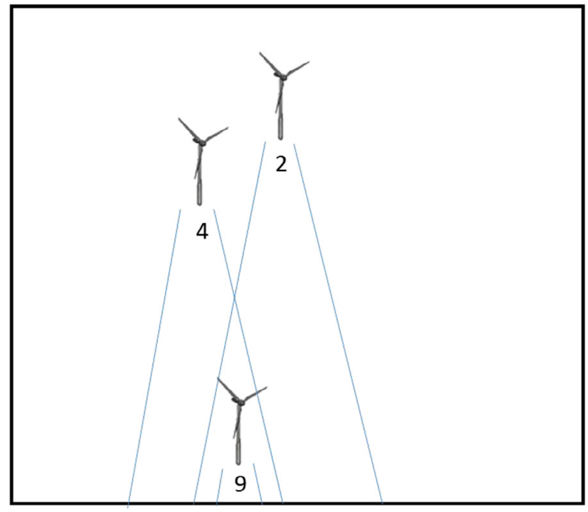


Fig. 5. Cascaded turbines in wake.

Table 1
The parameters for simulation of the DOA.

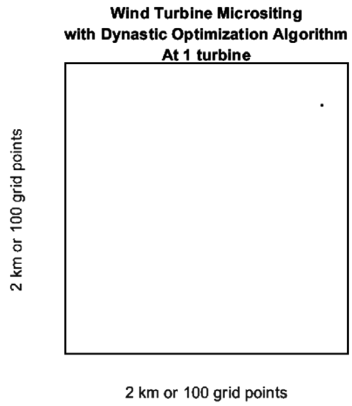
Number	Parameter	Values
(1)	N_{iter}, Number of iterations	10,000
(2)	N_p, Number of population	100
(3)	r_r, Ratio of rulers/kings	0.05
(4)	r_w, Ratio of workers/followers	0.55
5)	r_e, Ratio of explorers	0.4
(6)	rad_w, Radius of workers	0.4
(7)	U_0, wind speed	12 m/s

constant wind speed of 12 m/s has been used from the succession of similar simulations in the literature [9,10,16,17]. The optimal layout of the wind farm after installing 100 turbines with optimal placement of turbines is shown in Fig. 6(f). The successive installation of turbines starting from a single turbine to 25, 50, 54, 75 is also shown in Fig. 6(a)–(e), respectively. The total power produced and the cost of inclusion after installation of every next turbine in the grid have been noted for the proposed DOA, and the results are shown in Figs. 7 and 8, respectively. The efficiency of the proposed DOA has been almost ideal, but due to stochastic nature at instances it dropped slightly, the results are shown in Fig. 9.

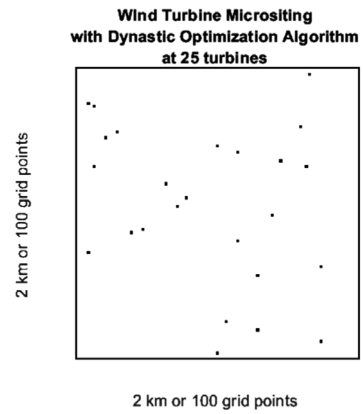
3.2. Comparison and validation of results of proposed DOA against other existing algorithms

The optimal statistics in terms of total power produced and cost per unit turbine have been compared in detail with a comparison to a few similar studies in the literature [1,9,10]. Some other algorithms from literature like the works of Mosetti et al. [2], Grady et al. [3], Mittal et al. [21] and Marmidis et al. [4] have been used for validation. The parameters used for GA [1], DEA [9,10], and other studies [2–4,21] have been taken from the corresponding papers.

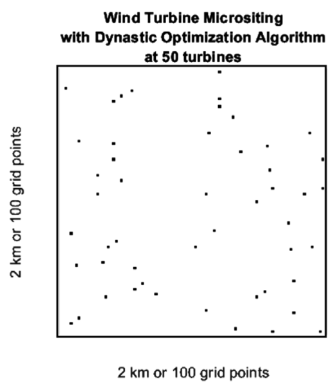
The power produced by using DOA was found higher than the GA method for nearly all the instances of the algorithm run. However, the power produced by the DOA is lesser than the GA at the instance of turbines 18, 19 and 25 [1]. At the installation of the 18th turbine, the GA returns a total power output of 9,331.20 kW whilst the DOA returns a value of 9,328.22 kW [1]. Similarly, at the installation of the 19th turbine, the GA returns the value of 9,849.60 kW and the DOA returns a value of 9,845.28 kW [1]. And, at the installation of the 25th turbine, the GA returns a value of



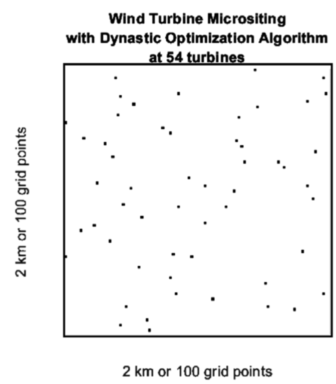
(a) DOA frame after installation of 1 turbine



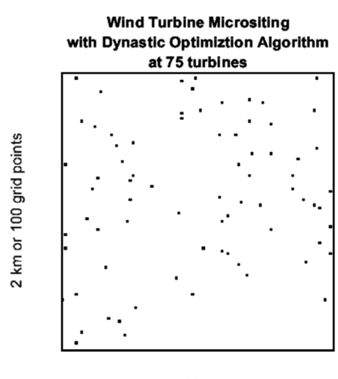
(b) DOA frame after installation of 25 turbines



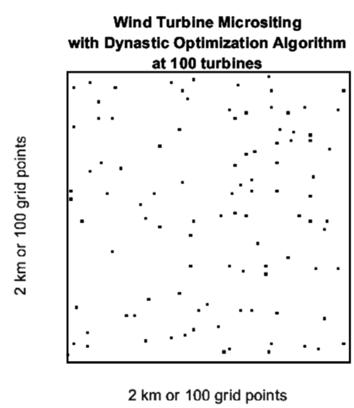
(c) DOA frame with optimal positioning of 50 turbines



(d) DOA frame with optimal positioning of 54 turbines



(e) DOA frame after installation of 75 turbines



(f) DOA frame after installation of 100 turbines

Fig. 6. Optimal layouts of different positioning of successive turbines by using DOA.

12,862.08 kW while the DOA returns a value of 12,805.65 kW [1]. However, the power difference is evident from the installation of the 20th turbine to the 24th turbine and from the 26th turbine to the 100th turbine. Hence, for the 20th turbine, the DOA returns a value of 10,359.23 kW and the GA lags at a value of 10,351.68 kW [1]. Similarly, at the installation of the 24th turbine, the DOA results in a value of 12,429.13 whereas the GA lags at a value of 12,360.00 kW [1]. And at the installation of the 26th turbine, the DOA gives a value of 13,453.78 kW while the GA gives a value of 13,364.16 kW [1]. Finally, at the installation of the 100th turbine, the DOA gives a value of 49,831.45 kW while the GA lags behind at a value of 48,452.26 kW. Similarly, the power produced by the proposed DOA remained higher as compared to DEA [9,10] except in a few instances. The improvement in power by the proposed DOA against GA and DEA are shown in Fig. 10. The negative power differences in Fig. 10 highlight a few cases where the GA and DEA were slightly taking edge over the results of DOA in the numerical simulation. However, after a complete run of algorithms for installation of 100 turbines, the final positioning of turbines by the proposed DOA has been found more efficient than GA and DEA in terms of power produced and

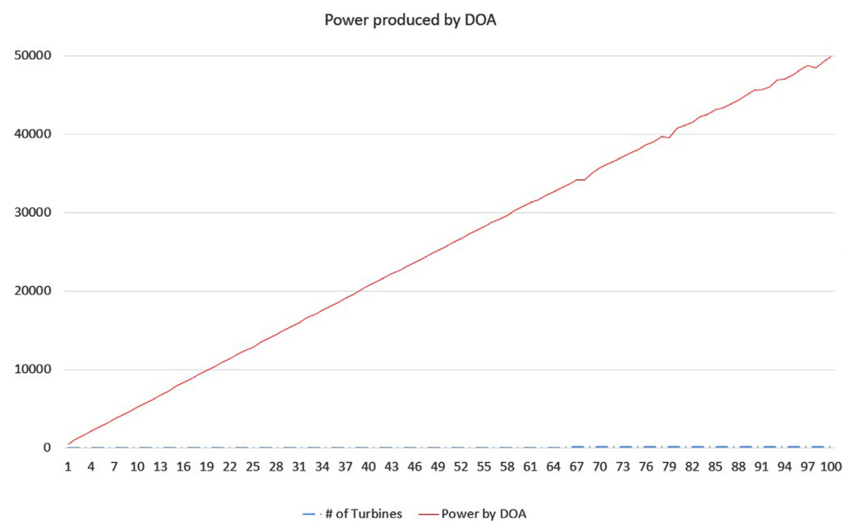


Fig. 7. The power produced by DOA after installing every successive turbine.

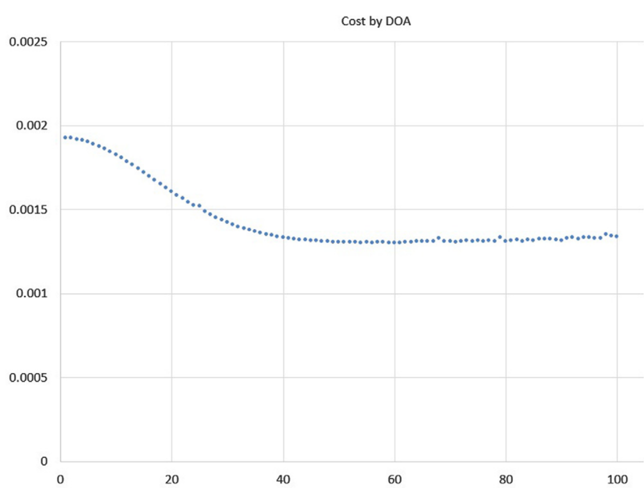


Fig. 8. Cost per installation of the unit turbine using proposed DOA.

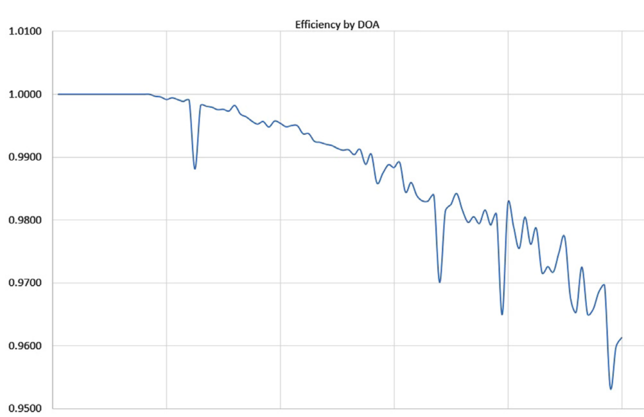


Fig. 9. The efficiency of DOA in the simulations after installing a new turbine successively.

cost reduction. This effect is attributed to the stochastic nature of both the algorithms [1,9,10].

From the results in [1,9,10] and of Fig. 10, it is evident that the DOA gives a lower cost for nearly all instances of the simulation.

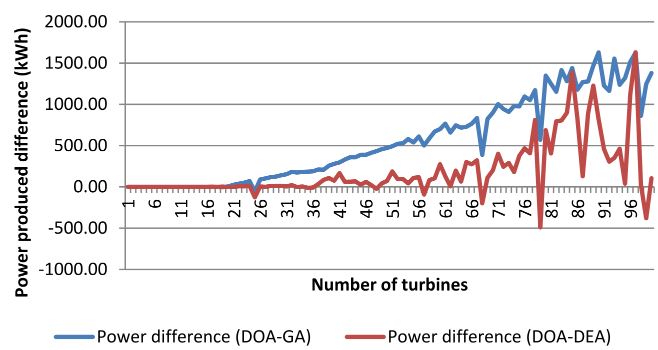


Fig. 10. Increase in power produced by proposed DOA against GA [20] and DEA [7,8].

This holds true with GA for all instances except for the installation of the 18th, 19th, and 25th turbines. At the installation of the 18th turbine, the GA has a lower cost of 0.0016519 whereas the DOA lags at a value of 0.0016524. Similarly, at the installation of the 19th turbine, the GA returns a cost of 0.0016291 whereas the DOA returns a higher value of 0.0016298. And for the installation of the 25th turbine, the GA results in a cost of 0.0015142 and the DOA results in a higher cost of 0.0015209. A similar situation arises when comparing cost reduction by DOA against DEA. The reduction in cost per unit turbine by the proposed DOA against GA and DEA are shown in Fig. 11. The difference in cost is minimum when DOA is compared with both algorithms, but given the magnitude of the lifetime of the project, the cost would become significant. This statement assumes that the operations and maintenance costs do not affect the total cost by a significant factor.

From the results, it is evident that the optimal solution for GA exists at the instance when 54 turbines are installed at a power output of 27,169.52 kW and a cost of 0.0013292. However, the power output of DOA is higher at the installation of 54 turbines at 27,748.88 kW at a lower cost of 0.0013039. These results for DOA are also higher than the results of DEA. Similarly, the optimal point in the case of the DOA occurs at the installation of the 61st turbine at a power output of 31,278.57 kW and at a cost of 0.0013011. Here, the GA lags behind at a power output of 30,512.21 kW and at a higher cost of 0.0013338, and the DEA exhibits a power output of 31,149.41 kW and at a higher cost of 0.0013065.

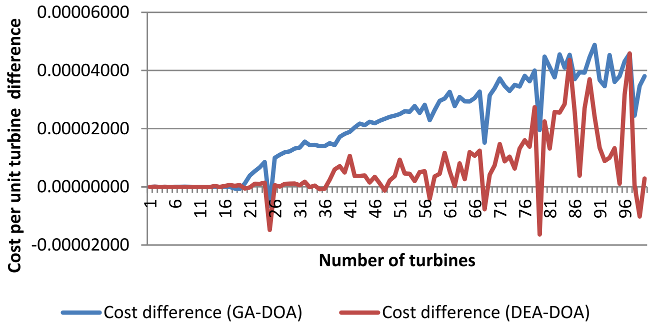


Fig. 11. Decrease in cost per unit turbine by proposed DOA against GA [20] and DEA [7,8].

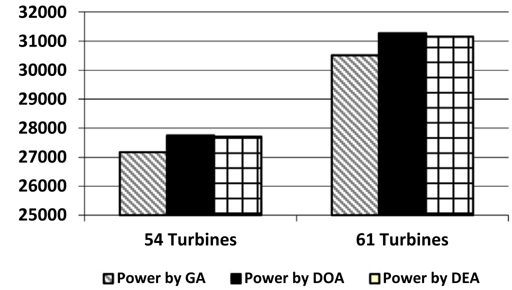


Fig. 12. Comparison of optimal point of GA [20], DEA [7,8] and proposed DOA.

Fig. 12 shows the comparison of power outputs of proposed DOA, GA [1] and DEA [9,10] at optimal points. The power output by the proposed remained highest of all.

Fig. 13(a) illustrates that the DOA gives higher power output as compared to Mosetti et al. [2], GA [1] and DEA [9,10] at the installation of 26 turbines as mentioned limit in Mosetti et al. [2]. Whilst, Fig. 13(b) illustrates that the DOA gives lower cost as compared to Mosetti et al. [2], GA [1] and DEA [9,10] when 26 turbines are installed. Thus, it may be safely concluded that the proposed DOA gives better performance as compared to Mosetti et al. [2], GA [1] and DEA [9,10]. Fig. 14 illustrates that the DOA gives higher power output and lower cost per unit turbine as compared to Grady et al. [3], GA [1] and DEA [9,10] at the installation of 30 turbines. It is evident that the results of the DOA simulation are superior to the results of Grady et al. [3], GA [1] and DEA [9,10], as the power produced is higher and the cost is lower.

Fig. 15(a) and (b) illustrate that the proposed DOA has a superior performance as compared to Marmidis et al. [4], GA [1] and DEA [9,10] at the installation of 32 turbines in terms of higher power production at lower cost per installation of unit turbine. A similar comparison of the performance of proposed DOA with studies of Mittal et al. [21] and other used algorithms GA [1] and DEA [9,10] is shown in Fig. 16. At the installation of 44 turbines, the DOA has a better performance as it results in higher power obtained shown in Fig. 16(a). On the other hand, from Fig. 16(b), it is evident that the lowest cost is reported when the DOA is as compared to Mittal et al. [21], GA [1] and DEA [9,10] at the installation of 44 turbines.

3.3. Contribution

The novel contribution of the proposed DOA indicated that this algorithm gave better results than DEA [9,10], GA [1], Mittal et al. [21], Marmidis et al. [4], Mosetti et al. [2] and Grady et al. [3]. This factor may be attributed to the simplistic, unassuming swarm-based structure and efficient operation of the proposed DOA. It is anticipated that this algorithm shall continue to grow and improve with every successful implementation over its lifetime. Hence, the above results strongly suggest that though this algorithm is a new entrant in this field of study however, it is giving competitive results with regard to the WTM problem.

3.4. Statistical analysis, sensitivity, and forecasts of results

Table 2 summarizes the descriptive statistics for the power produced and cost per unit turbine by using the proposed DOA, DEA and GA. It appears from Table 2 that all methods start with a power of 518.4 kW from the installation of the first turbine whereas the proposed DOA power reaches the highest level after installation of the 100th turbine. The power after installation of the 100th turbine for the DEA and GA are, respectively, the second and third in rank. Similarly, the proposed DOA beats all other methods from the viewpoint of being cost-efficient as the cost required per installation of a unit turbine in the process of installing 100 turbines for the DOA is smaller than those in DEA and GA. While looking at average power and cost values in Table 2, the following rating of methods appears clearly in ascending levels of power: GA, DEA, DOA. On the other hand, from the viewpoint of cost efficiency, the rating of methods is DOA, DEA, GA. The standard errors of estimate and the standard deviations in the power and cost levels for 100 turbines are also reported in Table 2.

While it appears clearly from descriptive stats of data, i.e. Table 2, that the maximum power level and minimum cost level for DOA are achieved as compared to DEA and GA, we also give results of the t-test to demonstrate our observations and hypotheses about power and cost of all methods more clearly.

In the student's t-test, we compare the average power and average cost by using DOA for the WTM problem against other methods. For this, we set the following hypotheses:

The null hypothesis, H_o: the average power (cost) produced by DOA and any other method X is the same. Mathematically,

$$\mu_{DOA} = \mu_X \tag{21}$$

The alternative hypothesis, H_a: the mean power (cost) by DOA is significantly different than that by any other methods X. Mathematically,

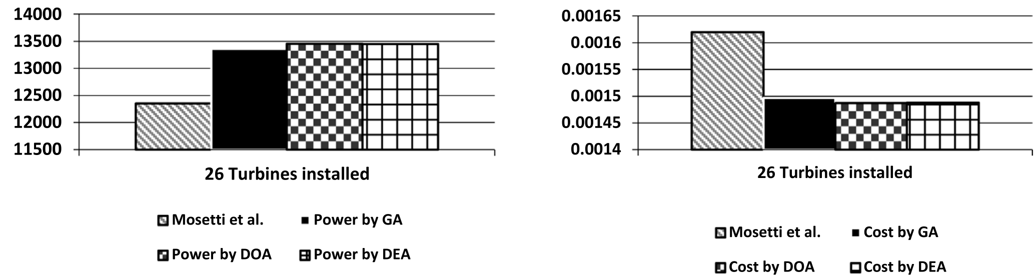
$$\mu_{DOA} \neq \mu_X \tag{22}$$

at 0.05 level of significance. Here X = {GA, DOA}.

Whenever we see a significance level of more than 0.05 in a difference, the null hypothesis will be true, otherwise alternative is accepted. Moreover, if the sign of a difference is positive then the average power (cost) produced by DOA is considered higher than a method X, otherwise lower at the same level of significance. The mathematics of the procedure of statistical comparison in energy saving and cost reduction can be found in [28,29].

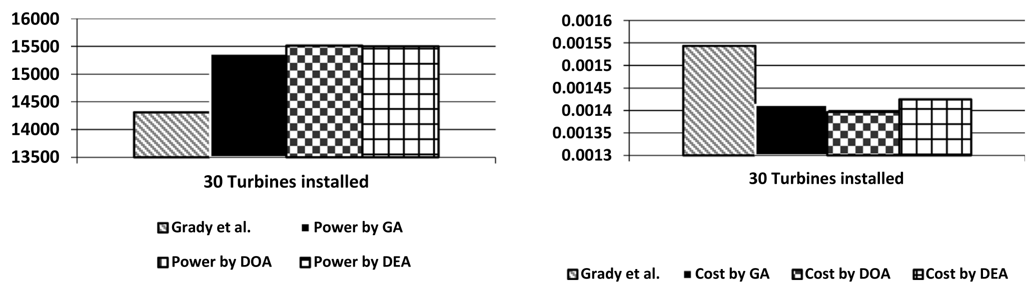
It appears from Table 3, at 0.05 level of significance, that there is a significant difference in the power produced, as well as cost, by using proposed DOA and other methods. So, the alternative hypothesis is accepted. This shows that the average power (cost) levels achieved by using DOA in installing 100 turbines are significantly different than the levels attained by other methods, at a 5% level.

Moreover, it is clear from Table 3 that the power achieved by DOA on average is higher than DEA and GA due to positive differences. Similarly, the average cost level of DOA after installing 100 turbines is lower than DEA and GA due to negative signs. At a 5% level of significance, it can be concluded that the DOA produces higher power output on average than the DEA and GA, and it is also more cost-efficient than these methods. The 95% confidence



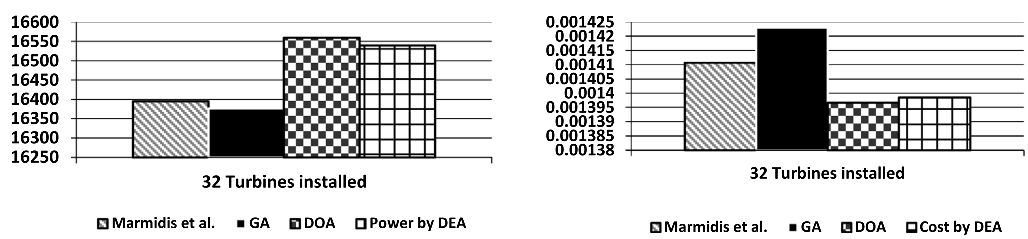
(a) Power produced in kW (b) Cost per unit turbine

Fig. 13. Comparison of power produced and cost per unit turbine at the installation of 26 turbines by Mosetti et al. [2], GA [1], DEA [9,10] and proposed DOA.



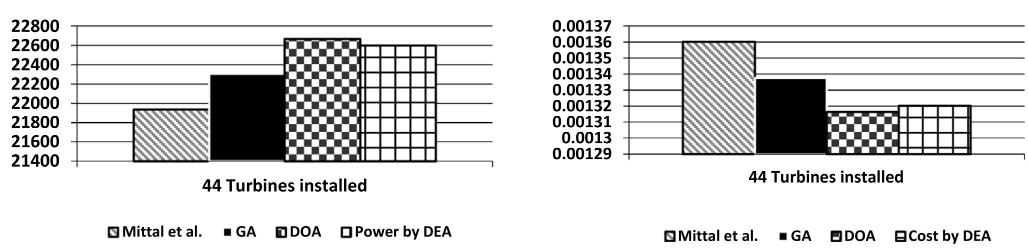
(a) Power produced in kW (b) Cost per unit turbine

Fig. 14. Comparison of power produced and cost per unit turbine at the installation of 30 turbines by Grady et al. [3], GA [1], DEA [9,10] and proposed DOA.



(a) Power produced in kW (b) Cost per unit turbine

Fig. 15. Comparison of power produced and cost per unit turbine at the installation of 32 turbines by Marmidis et al. [4], GA [1], DEA [9,10] and proposed DOA.



(a) Power produced in kW (b) Cost per unit turbine

Fig. 16. Comparison of power produced and cost per unit turbine at the installation of 44 turbines by Mittal et al. [21], GA [1], DEA [9,10] and proposed DOA.

intervals for the expected differences in the power and cost levels of DOA against other methods in the similar installation of 100 turbines in the future are shown more explicitly in Figs. 17 and 18, respectively. We have followed the procedure of forecasts and similar statistical models from [28,29]. Fig. 18 displays the present average increase in power and reduction in cost by using proposed DOA against GA and DEA; the future forecasts of the upper and lower bounds of the confidence intervals are also shown. The confidence interval forecasts for the power enhancement and cost reduction are encouraging, and indicate that, out of the 100 times the next installations of the similar wind turbines under the assumptions of this study, at least 90 times the computed bounds will hold, with a type-I error of only 0.01 (see Fig. 19).

Table 2
Descriptive statistics for the power produced and cost incurred by all algorithms.

Variables		N (size)	Range	Minimum	Maximum	Mean/average	Standard error in mean	Standard deviation	Variance
POWER	DEA	100	49208.22	518.40	49726.62	25510.2400	1433.55896	14335.58962	2.055E+8
	DOA	100	49313.05	518.40	49831.45	25699.3844	1453.41711	14534.17105	2.112E+8
	GA	100	47933.86	518.40	48452.26	25137.0871	1405.11270	14051.12695	1.974E+8
COST	DEA	100	0.0006253	0.0013026	0.0019279	0.001441852	0.00001934	0.00019347	0.000
	DOA	100	0.0006267	0.0013011	0.0019278	0.001435473	0.00001971	0.00019718	0.000
	GA	100	0.0005987	0.0013292	0.0019279	0.001457067	0.00001851	0.00018514	0.000

Table 3
t-test details to see significant difference in power and cost of DOA against other methods.

Difference variables		Mean differences	Standard deviation in mean differences	Standard error in mean differences	95% confidence interval forecasts of the mean differences		t-value	Degrees of freedom	p-value
					Lower	Upper			
POWER	DOA–DEA	189.14440	347.11954	34.71195	120.26835	258.02045	5.449	99	0.000
	DOA–GA	562.29730	507.74907	50.77491	461.54887	663.04573	11.074		0.000
COST	DOA–DEA	−6.37810E−6	1.073491E−5	1.07349E−6	−8.5081E−6	−4.2480E−6	−5.941		0.000
	DOA–GA	−2.15931E−5	1.516020E−5	1.51602E−6	−2.4601E−5	−1.8584E−5	−14.243		0.000

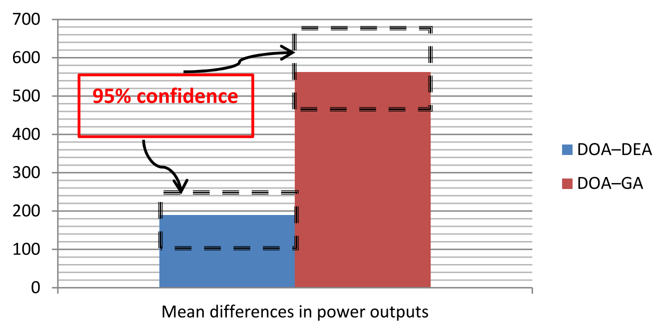


Fig. 17. 95% confidence interval forecasts for the increase in power output by DOA against GA and DEA.

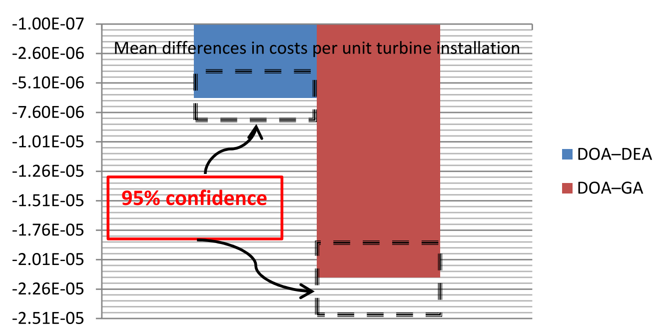


Fig. 18. 95% confidence interval forecasts for the cost difference by using DOA against GA and DEA.

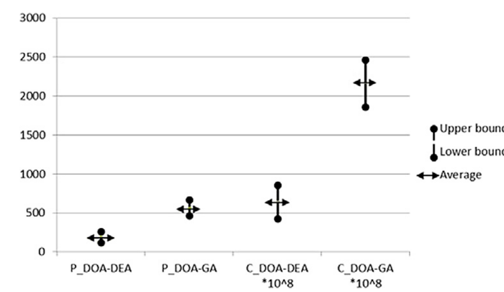


Fig. 19. The average difference, lower and upper bounds of the expected differences in parameters using DOA against GA and DEA.

3.5. Application to test functions

The DOA, DEA and GA were applied to the Rosenbrock's (f_1) and Easom's (f_2) test functions which are considered as conventional test functions for optimization methods.

$$f_1(x_1, x_2) = (1 - x_1)^2 + 100\left(x_2 - x_1^2\right)^2 \tag{23}$$

$$f_2(x_1, x_2) = -\cos x_1 \cos x_2 e^{-\left((x_1-\pi)^2+(x_2-\pi)^2\right)} \tag{24}$$

The minimum values of these functions are respectively 0 and -1 at (1,1) and (π, π).

We used same parameters as used in previous sections for WTO problem.

It is evident from Figs. 20–21 that DOA obtains the minimum value of f_1 more frequently than GA and DEA for a range of number of generation points. Proposed DOA exhibits substantial accuracy with regards to the results closer to expected minimum value for both functions, as in Figs. 20–21. On the other hand, GA and DEA in some cases are less likely to attain the expected minimum value. More on the mathematical insights and application of other widely used test functions can be found in near future.

4. Main achievements, recommendations, and future insights

The DOA algorithm has been developed for the solution of the WTM problem, but it is versatile enough for application to a variety of non-linear optimization problems with a finite number of constraints. It is a stochastic algorithm belonging to the class of nature-inspired metaheuristic algorithms. The term stochastic refers to the fact that two different instances of the same algorithm may yield two entirely different solutions. It has widespread applicability to a vast number of fields of science. The application of the algorithm is open for debate and its further advantages are yet to be discovered. However, it is concluded that the solution provided by the DOA is superior to the solution offered by the GA [1], DEA [9,10], Mittal et al. [21], Marmidis et al. [4], Mosetti et al. [2] and Grady et al. [3]. Further research and innovation would open many more vistas of application of this algorithm.

The DOA may find applications for developing a number of multi-agent systems and robotics, in unmanned vehicles. It has demonstrated that group dynamics can achieve group goals. This

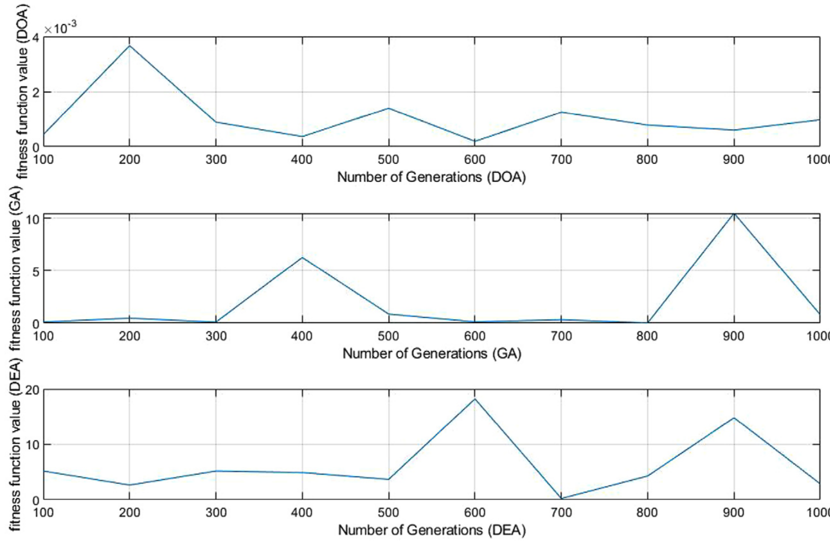


Fig. 20. Comparison of minima attained versus number of generations by all methods for Rosenbrock's function.

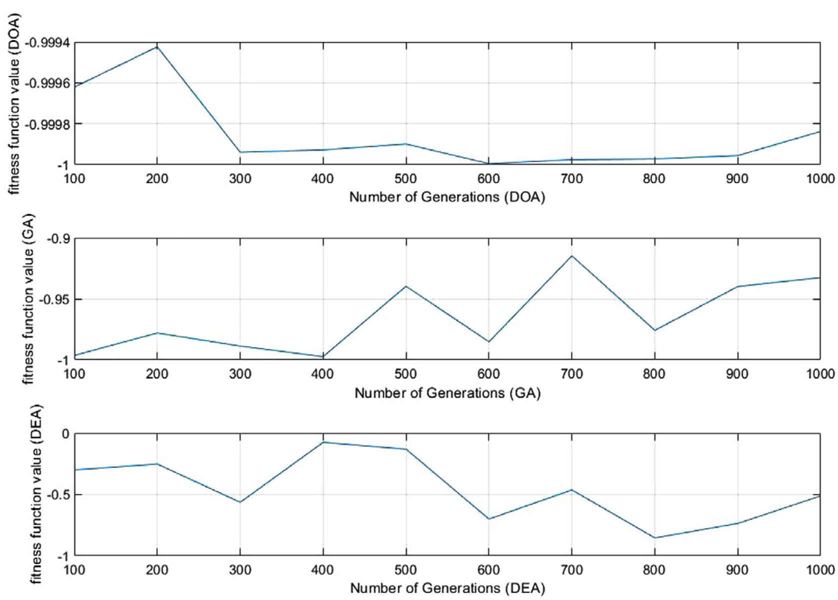


Fig. 21. Comparison of minima attained versus number of generations by all methods for Easom's function.

is because the collective convergence is more favorable as compared to individual excellence. The application of DOA may also be found in training of ANN weights and for the study of interstellar systems where the entropy of the system may change due to unforeseen circumstances and the global optima is volatile.

The following recommendations have been made during the course of this study,

(1) The DOA can be tested on a wide number of problems in the optimization domain.
(2) The DOA can be utilized in other real-world problems as it is anticipated to do so.
(3) The convergence of the proposed DOA can be further improved by using higher values of the characteristic parameters.
(4) An increase in computational power of the future would enable more extensive search of the solution space and further evaluation of algorithmic efficiency.
(5) Real-world scenarios and uneven terrains can be built and simulated using mainframe computers.
(6) The metaheuristic approach can be combined with the computational fluid dynamics (CFD) approach to yield superior layouts of wind farms.
(7) Presently, the layout of the wind farms is being proposed by humans that are run for a number of simulations to reach the optimal values. In our future work, we would seek to automate this final frontier so as enable the computer to suggest by itself the best placement of the wind turbines.
(8) The DOA can be further tested for getting the best possible combination for such type of problems.
(9) It is also deemed feasible to purchase computing power from the internet cloud to run higher-order simulations.

5. Conclusion

In this paper, a new metaheuristic algorithm has been formulated and applied to the WTM problem. It successfully optimized the placement of 100 wind turbines in a 2 km by 2 km area.

The comparisons were drawn with earlier results obtained by using modern algorithms of GA, DEA, Mittal et al. Monte Carlo–Marmidiset al. Mosetti et al. and Grady et al. It is evident that the stochastic solution of the WTM works out more solutions as compared to the human trial and error method indicated by Rajper et al. [1]. This is due to the sheer number of iterations evaluated by the computer.

The simulation and the results clearly demonstrate that the DOA gives a superior performance as compared to the other approaches in the literature. Moreover, it is a novel and validated contribution in the field of metaheuristics. It is evident at the installation of the 100th turbine the power produced using DOA was recorded as higher than using than the DEA and the GA. Whereas the cost per installation of unit turbine after 100th turbine was smaller than GA and DEA.

It is seen that the results are consistent and follow the results obtained by the GA and other studies. The results of DOA outperform the GA and DEA in terms of success rate and efficiency. The final results suggest that the new algorithm, DOA has good robustness and accuracy as compared to the mainstream algorithms.

The proposed DOA as a new metaheuristic algorithm is expected to open up a new era in stochastic metaheuristic computing. A new class of optimization algorithm with reference to the problem of wind turbine micrositing, has been introduced. This algorithm shall find widespread usage in all fields of science. It has been reported that it is often difficult to evaluate such functions that have uncertainties in their objective functions or the constraints. Such functions may be termed as volatile objective functions and the constraints may be termed as volatile constraints. Therefore, the optimization problem may be termed as a stochastic optimization problem. The DOA shall be very useful in the evaluation of optima for both volatile and non-volatile objective functions. Such types of functions may be found in interstellar studies of nebulae where disruptions in stellar systems change the entropy of the system from one optimal/maximum point to the other.

It is suggested that even more rigorous testing of the DOA shall be carried out in the future with a number of test functions in order to test the efficacy of the algorithm. The efficiency of the DOA for solving other NP-hard problems shall be investigated in the future. In the future, an extended version of this algorithm shall also be presented for the scientific community.

Declaration of competing interest

No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.asoc.2020.106176.

Acknowledgment

The authors wish to thank Dr. Hafiz Abdul Ghani Shaikh for continuous motivation and encouragement while completing this work.

References

[1] S. Rajper, I.J. Amin, Optimization of wind turbine micrositing: A comparative study, Renew. Sustain. Energy Rev. 16 (2012) 5485–5492.
[2] G. Mosetti, C. Poloni, B. Diviacco, Optimization of wind farms positioning in large wind farms by means of a genetic algorithm, Wind Eng. Ind. Aerodyn. (51) (1994) 105–116.
[3] S.A. Grady, M.Y. Hussaini, M.M. Abdullah, Placement of wind turbines using genetic algorithms, Renew. Energy I (30) (2005) 259–270.
[4] G. Marmidis, S. Lazarou, E. Pyrgioti, Optimal placement of wind turbines in a wind park using Monte Carlo simulation, Renew. Energy (33) (2008) 1455–1460.
[5] A. Emami, P. Nougreh, New approach on optimization in placement of wind turbines within wind farm by genetic algorithms, Renew. Energy (35) (2010) 169–178.
[6] S. Mirjalili, S.M. Mirjalili, A. Lewis, Grey wolf optimizer, Adv. Eng. Softw. 69 (2014) 46–61.
[7] K.V. Price, R.M. Storn, J.A. Lampinen, The Differential Evolution Algorithm, Springer, Berlin Heidelberg, 2005.
[8] R. Storn, K. Price, Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces, J. Global Optim. (11) (1997) 341–359.
[9] S.-R. Massan, A.I. Wagan, M.M. Shaikh, M.S. Shah, Application of differential evolution algorithm for wind turbine micrositing, Mehran Univ. Res. J. Eng. Technol. 36 (2) (2017) 133–146.
[10] S.-U.-R. Massan, A.I. Wagan, M.M. Shaikh, M.S. Shah, Numerical data concerning wind farm layout optimization using differential evolution algorithm at different wind speeds, Data in Brief, vol. 15, 2017, pp. 244–248.
[11] J.B. Lee, Clever Algorithms - Nature Inspired Programming Recipes, Jason Brown Lee, Melbourne, 2011.
[12] A. Kaveh, S. Mahjoubi, Lion pride optimization algorithm: A meta-heuristic method for global optimization problems, Sci. Iran. 25 (2018) 3113–3132.
[13] X.-S. Yang, S. Deb, Eagle strategy using levy walk and firefly algorithms for stochastic optimization, in: Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Vol. 284, 2010, pp. 104–111.
[14] X.-S. Yang, Firefly algorithm, stochastic test functions and design optimisation, Int. J. Bio-Inspired Comput (2010) 1–12.
[15] X.-S. Yang, Nature-Inspired Metaheuristic Algorithms, Second. Luniver press, 2010.
[16] S.-R. Massan, A.I. Wagan, M.M. Shaikh, R. Abro, Wind turbine micrositing by using the firefly algorithm, Appl. Soft Comput. 27 (2015) 450–456.
[17] S.-R. Massan, A.I. Wagan, M.M. Shaikh, A new hybrid metaheuristic algorithm for wind farm micrositing, Mehran Univ. Res. J. Eng. Technol. 36 (3) (2017) 635–648.
[18] S. Mirjalili, A.H. Gandomi, S.Z. Mirjalili, S. Saremi, H. Faris, S.M. Mirjalili, Salp swarm algorithm: A bio-inspired optimizer for engineering design problems, Adv. Eng. Softw. 114 (2017) 163–191.
[19] X.-S. Yang, S. Deb, Multiobjective cuckoo search for design optimization, Comput. Oper. Res. 40 (6) (2013) 1616–1624.
[20] J. Kenneddy, R.C. Eberhart, Particle swarm optimization, in: Proceedings of 1995 IEEE International Conference on Neural Networks, 1995, pp. 1942–1948.
[21] A. Mittal, Optimization of the Layout of Large Wind Farms using Genetic Algorithm (MS thesis), Case Western Reserve University, 2010.
[22] Ryanking, Openwindfarm, 2015, 2015, [Online]. Available: https://github.com/ryannking/OpenWindFarm. (Accessed 25 November 2015).
[23] S.-R. Massan, A.I. Wagan, M.M. Shaikh, Power optimization of wind turbines by the adjoint method, Sindh Univ. Res. J. 48 (3) (2016) 559–562.
[24] Wikipedia, Rank of a group, 2016, 2016, [Online]. Available: https://en.wikipedia.org/wiki/Rank_of_a_group.
[25] J.H. Holland, Genetic algorithms, Sci. Am. 267 (1) (1992) 66–72.
[26] N.O. Jensen, A note on wind generator interaction, in: RISO-M-2411, 1983.
[27] S.-R. Massan, A.I. Wagan, M.M. Shaikh, Numerical Data for Wind Turbine Micrositing Inspired By Human Dynasties By Use of the Dynastic Optimization Algorithm (DOA), Data in Brief, 2020, submitted for publication.
[28] A.J. Memon, M.M. Shaikh, Confidence bounds for energy conservation in electric motors: An economical solution using statistical techniques, Energy 109 (2016) 592–601.
[29] M.M. Shaikh, A.J. Memon, M. Hussain, Data on electrical energy conservation using high efficiency motors for the confidence bounds using statistical techniques, Data in Brief, vol. 8, 2016, pp. 529–535.